

# QUANTITATIVE ANALYSIS OF GEOMASKING METHODS

Von der Fakultät für Gesellschaftswissenschaften der Universität Duisburg-Essen

zur Erlangung des akademischen Grades

Dr. phil.

genehmigte Dissertation

von

Sarah Redlich

aus

Essen

1. Gutachter: Prof. Dr. Rainer Schnell
2. Gutachter: PD Dr. Günther Heller

Tag der Disputation: 15.02.2022



# Contents

<b>List of Figures</b>	<b>v</b>
<b>List of Tables</b>	<b>ix</b>
<b>Acronyms</b>	<b>xi</b>
<b>Preface</b>	<b>xv</b>
<b>Abstract</b>	<b>xvii</b>
<b>1. Introduction</b>	<b>1</b>
1.1. Geocodes . . . . .	2
1.2. The Use of Geographic Masking Methods . . . . .	3
1.3. Contribution . . . . .	3
1.4. Thesis Outline . . . . .	5
<b>2. Data Set and Software</b>	<b>7</b>
2.1. Data Set . . . . .	7
2.2. Coordinate System . . . . .	11
2.3. Software . . . . .	13
2.4. Parameter Choice . . . . .	13
<b>3. Geographic Masking Methods</b>	<b>15</b>
3.1. Aggregation . . . . .	18
3.1.1. Point Aggregation . . . . .	18
3.1.2. Areal Aggregation . . . . .	24
3.2. Adjusting Coordinates . . . . .	25
3.2.1. Affine Transformations . . . . .	25
3.2.2. Grid Masking . . . . .	30
3.2.3. Random Perturbation . . . . .	33
3.2.4. Triangular Displacement . . . . .	36
3.2.5. Donut Geomasking . . . . .	38
3.2.6. Voronoi Masking . . . . .	40
3.2.7. Location Swapping . . . . .	43
3.2.8. Verified Neighbor Approach . . . . .	46
3.2.9. Street Masking . . . . .	47

3.3.	Coordinate Replacement . . . . .	50
3.3.1.	Random Projection . . . . .	50
3.3.2.	Anonymization of Distance Matrices via Lipschitz Embedding . . . . .	51
3.3.3.	Distance Approximation Using Intersecting Sets of Grid Points . . . . .	53
3.4.	Other (Masking) Methods . . . . .	55
3.4.1.	Masking Based on the Military Grid Reference System (MGRS) . . . . .	56
3.4.2.	Alternative Methods . . . . .	58
<b>4.</b>	<b>Risk-Utility Framework</b>	<b>61</b>
4.1.	Utility Evaluation . . . . .	62
4.1.1.	Descriptive Statistics . . . . .	63
4.1.2.	Preserving Distances . . . . .	67
4.1.3.	Preserving Clusters . . . . .	69
4.1.4.	Spatial Autocorrelation . . . . .	73
4.1.5.	Combination of Utility Measures . . . . .	77
4.2.	Risk Evaluation . . . . .	80
4.2.1.	Scenario . . . . .	82
4.2.2.	Reversing Masking Methods . . . . .	83
4.2.3.	Mean of Multiple Releases . . . . .	84
4.2.4.	Minimum Distance . . . . .	84
4.2.5.	Assignment Problem: Hungarian Algorithm . . . . .	85
4.2.6.	Graph Theoretic Linkage Attack . . . . .	87
4.2.7.	Graph Matching Attack on Privacy-Preserving Record Linkage . . . . .	94
4.2.8.	Combination of Risk Measures . . . . .	99
4.3.	Overview of Analysis . . . . .	99
<b>5.</b>	<b>Analysis and Results</b>	<b>105</b>
5.1.	General Comparison of Geomasking Methods . . . . .	105
5.2.	Utility Evaluation . . . . .	109
5.2.1.	Descriptive Statistics . . . . .	109
5.2.2.	Preserving Distances . . . . .	127
5.2.3.	Spatial Autocorrelation . . . . .	141
5.2.4.	Clustering . . . . .	144
5.2.5.	Aggregated Results by Geomasking Methods . . . . .	151
5.3.	Risk Evaluation . . . . .	159
5.3.1.	Reversing Masking Methods . . . . .	160
5.3.2.	Mean of Masked Coordinates . . . . .	160
5.3.3.	Minimum Distance . . . . .	163
5.3.4.	Hungarian Algorithm . . . . .	165
5.3.5.	Graph Theoretic Linkage Attack . . . . .	169
5.3.6.	Graph Matching Attack on Privacy-Preserving Record Linkage . . . . .	171

---

5.3.7. Aggregated Results by Geomasking Methods . . . . .	174
<b>6. R-U Confidentiality Map of Geomasking Methods</b>	<b>181</b>
6.1. Components of the Risk-Utility Maps . . . . .	181
6.1.1. MPR . . . . .	181
6.1.2. GDi and LDi . . . . .	184
6.1.3. MSE . . . . .	187
6.2. Risk-Utility Map Using the GDi and LDi . . . . .	188
6.3. Risk-Utility Map Using the Mean Squared Error . . . . .	189
6.4. Summary of Results of R-U Confidentiality Maps . . . . .	189
<b>7. Discussion and Conclusion</b>	<b>193</b>
7.1. Key Findings . . . . .	193
7.2. Limitations . . . . .	196
7.3. Implications of Results for the Use of Geographic Masking Methods .	197
7.4. Future Research . . . . .	198
<b>Bibliography</b>	<b>200</b>
<b>Appendix</b>	<b>215</b>
<b>A. Code</b>	<b>215</b>
A.1. Aggregation . . . . .	215
A.2. Adjusting Coordinates . . . . .	218
A.3. Coordinate Replacement . . . . .	229
A.4. Overview of Masking Methods (Detailed) . . . . .	231
A.5. Abbreviations of Masking Methods and Parameter Choices . . . . .	232
<b>B. DBSCAN: Parameter Choice</b>	<b>239</b>
<b>C. Frequency Table of Overlap</b>	<b>241</b>
<b>D. Simulation Studies for Parameter Choices</b>	<b>245</b>
D.1. Parameter Choice for Anonymization via Lipschitz Embedding . . . .	245
D.2. Parameter Choice for Distance Approximation Using Intersecting Sets of Grid Points . . . . .	246
<b>E. Execution Time Masking Methods</b>	<b>247</b>
<b>F. Execution Times Risk Analysis</b>	<b>249</b>
<b>G. Detailed Results of the Risk-Utility Analysis</b>	<b>253</b>
G.1. Spatial Mean Center . . . . .	253
G.2. Spatial Median Center . . . . .	255

G.3. Standard Distance . . . . .	256
G.4. Standard Deviation Ellipse . . . . .	258
G.5. Distance Between Coordinates of Data Set . . . . .	264
G.6. Distance Between Original and Masked Coordinates . . . . .	268
G.7. Spatial Autocorrelation . . . . .	274
G.8. Clustering . . . . .	278
G.9. Minimum Distance Risk Method . . . . .	296
G.10. Hungarian Algorithm . . . . .	298
G.11. Hungarian Algorithm Using Additional Variables . . . . .	300
G.12. Graph Theoretic Linkage Attack . . . . .	302
G.13. Graph Matching Attack on Privacy-Preserving Record Linkage . . . . .	303
<b>H. Explanation for Summarizing GD<sub>i</sub> and LD<sub>i</sub></b>	<b>307</b>
<b>I. R-U-Map of Larger Samples</b>	<b>309</b>
I.1. MPR . . . . .	309
I.2. GD <sub>i</sub> and LD <sub>i</sub> . . . . .	311
I.3. MSE . . . . .	313
I.4. Risk-Utility Maps . . . . .	314

# List of Figures

2.1. Map of South Australia. Red points are residents and blue points are sampled points ( $n = 10,000$ ) . . . . .	11
3.1. Categorization of masking methods (oriented at Gutmann et al., 2008). . . . .	16
3.2. Maximum distance to average vector (MDAV). . . . .	19
3.3. Adaptive Point Aggregation (equilateral polygons) . . . . .	21
3.4. Adaptive Random Perturbation (equilateral polygons) . . . . .	22
3.5. Displacement using Translation . . . . .	26
3.6. Change of Scale . . . . .	27
3.7. Rotation (Origin of Coordinate System) . . . . .	28
3.8. Rotation (Arbitrary Point) . . . . .	29
3.9. Global grid masking: horizontal flip (central axis). . . . .	30
3.10. Global grid masking: vertical flip (central axis). . . . .	31
3.11. Global grid masking: horizontal and vertical flip (central axis). . . . .	31
3.12. Local grid masking . . . . .	32
3.13. Random Perturbation (uniform distribution) . . . . .	33
3.14. Random Perturbation (normal distribution) . . . . .	34
3.15. Random Perturbation (within a circle) . . . . .	34
3.16. Donut Geomasking (Hampton et al., 2010, p. 1063). . . . .	39
3.17. Donut Geomasking: maximum radius 0.8, minimum radius 0.3. . . . .	39
3.18. Example of construction of Thiessen-polygon . . . . .	41
3.19. Voronoi masking . . . . .	42
3.20. Location swapping . . . . .	44
3.21. Location swapping with donut . . . . .	45
3.22. Verified Neighbor Approach . . . . .	47
3.23. Random projection example. . . . .	50
3.24. The distance between two points $P$ and $Q$ can be approximated using the area of intersection of two circles (gray area) surrounding the points (figure taken from Schnell, Klingwort, et al., 2021, p. 3). . . . .	53
3.25. Example of regular grid with random numbers laid over the two points (figure taken from Schnell, Klingwort, et al., 2021, p. 3). . . . .	54
3.26. Fifteen digit format of coordinates in MGRS system . . . . .	56
4.1. Risk-Utility Confidentiality Map (based on Duncan and Fienberg, 1999, p. 352; Duncan, Keller-McNulty, et al., 2001, p. 7). . . . .	61
4.2. Overview of risk-utility measures . . . . .	62

4.3. Explanation of relevance of clustering methods to evaluate utility. . . .	69
4.4. DBSCAN example (radius= $\varepsilon$ , MinPts=4). . . . .	72
4.5. Definition of edge weights . . . . .	90
4.6. Definition of maximum radius for points masked using a uniform distribution . . . . .	91
5.1. Average execution time (in minutes) of the masking methods to mask 10,000 coordinates (without the outlier distance approximation using ISGP, 709 minutes). . . . .	108
5.2. Spatial mean centers of masking methods (in EPSG:3107) . . . . .	111
5.3. Mean center zoomed in without outliers . . . . .	112
5.4. Spatial median centers of masking methods (in EPSG:3107) . . . . .	115
5.5. Median center zoomed in without outliers . . . . .	116
5.6. Standard distance of masking methods. . . . .	118
5.7. Standard distance of masking methods without outliers. . . . .	119
5.8. Angle (in degrees) of standard deviation ellipse compared to original data set (red dashed line). . . . .	121
5.9. Major axis difference to original. . . . .	123
5.10. Minor axis difference to original. . . . .	124
5.11. Major axis difference to original without outliers. . . . .	125
5.12. Minor axis difference to original without outliers. . . . .	126
5.13. Mean (orange) and median (red) distance between points averaged over fifty replications . . . . .	129
5.14. Mean distance between points averaged over fifty replications without outliers . . . . .	130
5.15. Median distance between points averaged over fifty replications without outliers . . . . .	131
5.16. Mean and median distances the points are moved for the adaptive areal elimination methods . . . . .	133
5.17. Mean and median distances the points are moved for the donut masking methods using $k$ . . . . .	134
5.18. Mean and median distances the points are moved for the donut masking methods using population density . . . . .	134
5.19. Mean and median distances the points are moved for the random perturbation methods . . . . .	136
5.20. Mean and median distances the points are moved for the location swapping methods . . . . .	136
5.21. Mean and median distances the points are moved for the verified neighbor methods . . . . .	137
5.22. Mean and median distances the points are moved for official statistics grid, microaggregation as well as Voronoi masking . . . . .	138

5.23. Mean and median distances the points are moved for change of scale and rotation . . . . .	138
5.24. Mean and median distances the points are moved for displacement using translation, rotation around an arbitrary point, intersecting grid points as well as Lipschitz embedding . . . . .	139
5.25. Results of Moran's I values for proportion of single households as difference to original value . . . . .	142
5.26. Results of Moran's I values for proportions of full-time working people (right) as difference to original value . . . . .	143
5.27. DBSCAN ( $\varepsilon = 3, 200$ ): Total number of clusters as well as number of clusters with at least 30 points . . . . .	146
5.28. DBSCAN ( $\varepsilon = 9, 500$ ): Total number of clusters as well as number of clusters with at least 30 points . . . . .	147
5.29. Mean of masked coordinates: donut masking method using population density and $k$ -nearest neighbor donut masking. . . . .	161
5.30. Mean of masked coordinates: random perturbation methods. . . . .	162
5.31. Mean of masked coordinates: location swapping, and the verified neighbor approach. . . . .	162
5.32. Average precision and recall for minimum distance. . . . .	164
5.33. Average precision and recall for Hungarian algorithm. . . . .	166
5.34. Average precision and recall for Hungarian algorithm using additional variables. . . . .	167
5.35. Average precision and recall for graph theoretic linkage attack. . . . .	170
5.36. Average precision and recall for graph matching PPRL attack using stable marriage match . . . . .	172
5.37. Average precision and recall for graph matching PPRL attack using symmetric highest match . . . . .	173
6.1. Risk-utility map. Utility using Kounadi and Leitner (2015). . . . .	190
6.2. Risk-utility map. Utility using the MSE. . . . .	191
A.1. Detailed overview of masking methods. . . . .	231
B.1. $k$ -nearest neighbor plot for finding the optimal value of the radius $\varepsilon$ . Distances were sorted in ascending order. . . . .	239
B.2. $k$ -nearest neighbor plot for finding the optimal value of the radius $\varepsilon$ . Distances were sorted in ascending order. Zoomed in for better evaluation. . . . .	240
D.1. Mean of relative difference if 5,000,000 grid points are used. Red dot indicates minimum, located at $r = 40, 000$ . . . . .	246
H.1. Explanation for summarizing the components of $GDi$ and $LDi$ (Kounadi and Leitner, 2015) for every masked data set. . . . .	307
I.1. Risk-Utility Map of the larger subsample ( $n = 2, 000$ ). $GDi$ and $LDi$ as utility measure. . . . .	315

- 
- I.2. Risk-Utility Map of the full sample ( $n = 10,000$ ). GDi and LDi as utility measure. . . . . 316
  - I.3. Risk-Utility Map of the larger subsample ( $n = 2,000$ ). MSE as utility measure. . . . . 317
  - I.4. Risk-Utility Map of the full sample ( $n = 10,000$ ). MSE as utility measure. 318

# List of Tables

2.1. Combinations of sex, employment, and age of the overlap of the masked and identification file ( $n = 1,000$ , sorted by frequency of overlap).	12
3.1. Example of masking based on MGRS.	57
4.1. Example of Hungarian algorithm (Hardwick, 1996, p. 127).	86
4.2. Calculation example of Hungarian algorithm (Hardwick, 1996, pp. 127–129).	87
4.3. Overview of masking methods and if utility and risk measures could be applied (used sample size written below utility/risk measures).	103
5.1. General comparison of masking methods (only tested methods).	106
5.2. Aggregation of descriptive statistics by methods.	151
5.3. Aggregation of standard deviational ellipses by masking methods.	153
5.4. Aggregation of mean and median distance between points by masking methods.	154
5.5. Aggregation of average and median distance between original and masked points by masking methods.	155
5.6. Aggregation of Moran’s I for single households and full-time employment by masking methods.	156
5.7. Aggregation of average number of cluster larger than 30 as well as number of non-clustered points by masking methods.	157
5.8. Aggregation of total of number of points changing from clustered to non-clustered and vice versa.	158
5.9. Precision and recall for affine transformation masking methods.	160
5.10. Results of minimum distance aggregated by masking methods.	175
5.11. Results of Hungarian algorithm aggregated by masking methods.	176
5.12. Results of Hungarian algorithm using additional variables.	177
5.13. Results of graph theoretic linkage attack aggregated by masking methods.	178
5.14. Results of graph matching attack on privacy-preserving record linkage aggregated by masking methods.	179
6.1. MPR for full sample as well as subsamples.	182
6.2. Results of the calculation of the GDi and LDi and its components according to Kounadi and Leitner (2015).	186
6.3. Results of the calculation of the MSE	188
C.1. Combinations of sex, employment, and age of the overlap of the masked and identification file ( $n = 10,000$ , sorted by frequency of overlap).	241

D.1. Precision and recall for simulation study for parameter choice for the anonymization via Lipschitz embedding masking method. . . . .	245
E.1. Execution time of masking method applications (in minutes). . . . .	247
F.1. Average execution times of risk methods for one replication (in minutes). . . . .	249
F.2. Execution times of risk methods for each iteration (in minutes). . . . .	250
G.1. Spatial mean (mean center) comparison of masked data with original data (in meter). . . . .	253
G.2. Spatial median (median center) comparison of masked data with original data (in meter). . . . .	255
G.3. Standard distance of the masked coordinates (in meter). . . . .	256
G.4. Angle of rotation of the standard deviational ellipses of the masked coordinates (in degree). . . . .	258
G.5. Difference in the length of the major axis (x-axis) of the standard deviational ellipse of the original coordinates and of the masked coordinates (in meter). . . . .	261
G.6. Difference in the length of the minor axis (y-axis) of the standard deviational ellipse of the original coordinates and of the masked coordinates (in meter). . . . .	262
G.7. Mean and median distance between the coordinates. . . . .	264
G.8. Detailed results for mean and median distances points are moved. . . . .	268
G.9. Minimum, maximum, mean and median standard deviations of the distances over all iterations. . . . .	272
G.10. Results of Moran's I values for proportion of single households . . . . .	274
G.11. Results of Moran's I values for proportion of full-time working people . . . . .	276
G.12. Results of DBSCAN clustering algorithm ( $\epsilon = 3200$ ). . . . .	278
G.13. Results of DBSCAN clustering algorithm ( $\epsilon = 9500$ ). . . . .	286
G.14. Number of points changing from clustered to non-clustered and vice versa. . . . .	294
G.15. Average precision and recall for minimum distance. . . . .	296
G.16. Average precision and recall of the Hungarian Algorithm. . . . .	298
G.17. Average precision and recall of the Hungarian Algorithm with blocking by sex, age, and employment status . . . . .	300
G.18. Average precision and recall of the graph attack . . . . .	302
G.19. Average precision and recall of the pprl attack using SMM . . . . .	303
G.20. Average precision and recall of the pprl attack using SHM . . . . .	304
I.1. MPR for different parameter choices of masking methods . . . . .	309
I.2. Results of the GDi, LDi and its average for different parameter choices of masking methods . . . . .	311
I.3. Results of the MSE for different parameter choices of masking methods . . . . .	313

# Acronyms

AAE	Adaptive Areal Elimination
AAM	Adaptive Areal Masking
AGNES	Agglomerative Nesting
APA	Adaptive Point Aggregation
ARP	Adaptive Random Perturbation
BIRCH	Balanced Iterative Reducing and Clustering using Hierarchies
CLARANS	Clustering Large Applications based on Randomized Search
CLIQUE	Clustering In Quest
ClusNoise	Number of Points Remaining Clustered/Unclustered
ClusNum	Number of Clusters
COVID-19	Coronavirus Disease 2019
CS	Change of Scale
DBCLASD	Distribution Based Clustering of Large Spatial Databases
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DD	Donut Masking using Population Density
DENCLUE	Density-based Clustering
DIANA	Divisive Analysis
Dk	Donut Masking using $k$ -nearest Neighbors
DkData	Donut Masking using $k$ -nearest Neighbors, reference file is data set itself
DUT	Displacement Using Translation
EM	Expectation-Maximization Algorithm
EPSG	Geodetic Parameter Registry of the European Petroleum Survey Group (now: International Association of Oil & Gas Producers)
ESRI	Environmental Systems Research Institute
ETRS89	European Terrestrial Reference System 1989
FN	False Negatives
FP	False Positives

---

FSDP	Fast Search by Density Peak
GDi	Global Divergence Index
GIS	Geographic Information System
G-NAF	Geocoded National Address File
GDA94	Geocentric Datum of Australia 1994
Grid	Official Statistics Grid
GRS80	Geodetic Reference System 1980
INSPIRE	INfrastructure for SPatial InfoRmation in Europe
ISGP	Intersecting Sets of Grid Points
LAEA	Lambert Azimuthal Equal-Area projection
LDi	Local Divergence Index
LGA	Local Government Areas
LifBi	Leibniz-Institut für Bildungsverläufe e.V
Lipschitz	Anonymization of Distance Matrices via Lipschitz Embedding
LS	Location Swapping
LSdonut	Location Swapping with donut
LSH	Locality Sensitive Hashing
MAdi	Major Axis' Divergence
MdAV	Maximum Distance to Average Vector
Mdi	Mean's Divergence
MDS	Multidimensional Scaling
MGRS	Military Grid Reference System
MPR	Mean of Precision and Recall
MSE	Mean Squared Error
MWM	Maximum Weight Match
NEM	Neighborhood Expectation-Maximization Algorithm
Odi	Orientation's Divergence
OPTICS	Ordering Points To Identify the Clustering Structure
OSM	OpenStreetMap
P	Positives
PAM	Partitioning Around Medoids
PPRL	Privacy-Preserving Record Linkage
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses

---

PRMSE	Percent Root Mean Squared Error
PSMA	Public Sector Mapping Agency
RandProj	Random Projection
RatSWD	Rat für Sozial- und Wirtschaftsdaten
RMSE	Root Mean Squared Error
Rot	Rotation around origin of coordinate system
RotArb	Rotation around arbitrary point
RPC	Random Perturbation within a circle
RPN	Random Perturbation using a normal distribution
RPU	Random Perturbation using uniform distribution
SHM	Symmetric Highest Match
SMM	Stable Marriage Match
SpatAutCorr	Spatial Autocorrelation
STING	Statistical Information Grid-Based
StreetMask	Street Masking
SVD	Singular Value Decomposition
TP	True Positives
UTM	Universal Transverse Mercator
VBAS	Voronoi-Based Aggregation Systems
VNE	Verified Neighbor Approach/Method using employment status as variable of interest
VNS	Verified Neighbor Approach/Method using sex as variable of interest
Voronoi	Voronoi Masking
WGS84	World Geodetic System 1984



# Preface

This thesis is the result of my research of the past years at the social science department of the University of Duisburg-Essen. The idea for this thesis came from Prof. Dr. Schnell. He gave me the first overview of geographic masking methods (Armstrong et al., 1999) and introduced me to a research topic that I would not have made contact with otherwise. He also suggested using a risk-utility map to distinguish between masking methods. His work in record linkage is the reason for the scenario considered for the risk analysis. Further, he introduced me to his contributions in the field, namely to two masking methods and two risk methods. Additionally, his collaboration with Prof. Dr. Peter Christen led to the choice of the data set used in this thesis.

Given the starting point and the idea, I researched the geographic masking methods currently existing. Then, unless code was available for the geographic masking methods, I implemented each geographic masking method (in the open-source software *R*). The code will allow other researchers who are either unfamiliar or do not have access to software usually used in this field (such as ArcGIS) to use geographic masking methods.

Based on recommendations in the geomasking literature, I chose the parameters for the particular geographic masking methods. I also identified the utility measures used in this thesis. In agreement with Prof. Dr. Schnell, the scenario for the risk analysis was defined. Two of the risk measures were his contributions to research in this field as well as record linkage. The other risk measures are the results of approaches found in literature and solutions for similar problems in other fields. Finally, I conducted the analysis and summarized the results in this thesis.

## Acknowledgements

First, I want to thank Prof. Dr. Rainer Schnell for the encouragement to pursue a scientific career. His idea for this thesis, his contributions, and his comments on my work helped me finish this thesis. Second, I also want to thank PD Dr. Günther Heller for taking the time to be the second supervisor. I also want to thank the other committee members for their time and effort. In addition, I would like to thank the committee members, in particular Prof. Dr. Rainer Schnell, for their comments, which led to this editorially revised publication version.

Furthermore, I want to thank Dr. Yanling Chen for the many discussions about the implemented risk methods and her comments on my work. Her wise words throughout

the final two years of this dissertation helped in finishing this work. In addition, I want to thank my former colleagues Jonas and Christian for the discussions whenever I got stuck on a problem in the early stages of this thesis.

Lastly, I want to thank my family and friends for their support throughout this process. Mainly, I would like to thank Daniel and Vio for their moral support. I could not have finished this thesis without them.

# Abstract

Spatial information at the most detailed level (coordinates) has been recognized as important information in various fields such as epidemiology, medicine, and social science. Geographic coordinates are used to identify the relationship between behaviors and environmental factors called social-spatial linkage. However, the release of geographic coordinates makes the identification of the respondents' address relatively easy. Therefore, geographic masking methods (short: geomasking methods) have been proposed to preserve the privacy of the respondents' true location.

Geographic masking methods can be divided into three categories: aggregation, adjusting coordinates, and coordinate replacement. Since the second category contains most geomasking methods, it can be further divided into three subcategories. The first subcategory contains methods that scale, rotate, displace or flip coordinates. The second subcategory contains methods that move points into a random direction and random distance. The last subcategory contains methods that also move locations into a random direction and a random distance but need additional information to be applied.

Decreasing the risk of identifying an address is often accompanied by a decreasing utility of the spatial information. This risk-utility-trade-off can be visualized in a risk-utility map. Various risk and utility measures are used to perform a quantitative analysis of geographic masking methods. Utility measures are preserving descriptive statistics, preserving distances, spatial autocorrelation, and preserving clusters.

Risk, which is commonly assessed using  $k$ -anonymity, is addressed in more detail in this thesis. As common in the field of record linkage, it was assumed that an intruder has a data set containing the true location of some respondents. Several methods are used to identify the correct matches between the data set containing the masked locations and the data set containing the true locations. These methods are the minimum distance approach, the Hungarian algorithm, the graph theoretic linkage attack, and the graph matching attack on privacy-preserving record linkage. In addition, taking the mean of several masked coordinates is used, and, for some methods, attempts are made to reverse the displacement of the coordinates caused by the masking methods.

The risk-utility maps show that most masking methods succeed in preserving the utility but also show a high risk of re-identification. The maps also reveal that using  $k$ -anonymity alone is not appropriate, as coordinates masked with geomasking methods based on the  $k$ -anonymity approach were still re-identified. The only geomasking

methods that succeed in hiding the true location while showing good utility preserving properties fall into the category of replacing the coordinates by only allowing the publication of a distance matrix.

# 1. Introduction

The importance of spatial data is widely recognized in research, and official statistics (see, e.g., RatSWD, 2011).<sup>1</sup> Spatial data has become very important in the research of health-related topics. The earliest example of the use of geographic information is John Snow's cholera outbreak map to detect disease outbreaks (the map can be found in Snow, 1854 or, e.g., in McLeod, 2000 taken from Frost, 1936). He used the addresses of infected people to draw a map of all known cases and detected the relation to the water system. Even nowadays, the need for geographic information to discover the spread of diseases is emphasized, for example, with the COVID-19 pandemic (see, e.g., infas 360, 2020).

Rushton et al. (2006) showed how geographic information is used to attach socio-economic, demographic, and environmental variables to a data set of cancer patients. With this information, research questions such as the distance to the nearest health service can be answered. Nearly a decade earlier, Vine et al. (1997) showed how geographic information is used in epidemiology to predict people's level of exposure to environmental factors such as water contamination. Kamel Boulos (2004) provides a short overview of other studies using geographic information in the health research field.

Throughout the years, the advantages of using spatial data have been noticed in other fields as well (Rushton et al., 2008, p. 1). For example, Olligschlaeger (1997) shows how geographic information is used to map criminal activity based on police calls. Furthermore, geographic information is used to forecast crime and analyze crime patterns. Another field is political science, where spatial information is needed to analyze, for example, reasons not to vote. The idea is that the distance between a resident and the voting poll could influence the individual's cost and benefit analyses to vote (Haspel and Knotts, 2005).

Overall, spatial data enables the researchers to identify additional (risk) factors and environmental factors that might influence certain behaviors and characteristics of respondents (Rushton et al., 2008, p. 3). This is also termed "Social-Spatial Linkage" and has been recognized as an important factor in analyzing respondents' behavior and characteristics (VanWey et al., 2005, p. 15337). Furthermore, it has been recognized that it is important to make the linkage of spatial information available to third parties (VanWey et al., 2005, p. 15337; National Research Council, 2007).

---

<sup>1</sup>In 2019, the German Federal Statistical Office held a scientific colloquium about the benefits of geographic information (<https://www.destatis.de/DE/Ueber-uns/Kolloquien-Tagungen/Kolloquien/2019/28WissenschaftlichesKolloquium.html>).

## 1.1. Geocodes

Spatial information is available in external data sets which are linked to the given data using so-called *geocodes* (VanWey et al., 2005, p. 15337). The term *geocodes* is comprised of the words *geo*, Latin for earth and *coding* “applying a rule for converting a piece of information into another” (Goldberg et al., 2007, p. 35). Geocodes assign addresses and places the equivalent of a geographic unit (Dueker, 1974, p. 319). This geographic unit can be nominal, ordinal, or cardinal indices. Nominal indices refer to the names of streets, buildings, or cities. Ordinal indices comprise zip codes or census area codes. Cardinal indices represent coordinate systems (Dueker, 1974, p. 319).

This classification shows an important feature of geocodes: different levels are available. For example, one can work with city levels that assign the same spatial information to people living in the same city. Working with such large geographic levels, also referred to as lattice data (Cressie, 1992, p. 614), especially in large cities, poses the problem of assigning the same environmental and social factors to respondents from the same city in very different districts. For example, the same information is assigned to two people, even though one person may live in the city, and one person may live on the outskirts and thus in very rural areas.

Authors such as Openshaw (1983) and Openshaw and Taylor (1981) pointed out in their work in the early 1980s that results of analyses can vary depending on the level of the spatial information available (e.g., states, postal code, coordinates). Aggregating data may lead to different conclusions than point data because the aggregation of information may hide the underlying structure. Furthermore, different ways of aggregating data can lead to different conclusions. This problem is known as the *modifiable areal unit problem* (Openshaw and Taylor, 1981; Openshaw, 1983).

The use of geographic coordinates makes it possible to account for small spatial differences (Armstrong et al., 1999, p. 498; VanWey et al., 2005, p. 15337). In this dissertation, only cardinal indices are considered, i.e., where the spatial location is available at the most detailed level (geographic coordinates).

The downside of releasing geographic coordinates lies in privacy concerns (see, e.g., National Research Council, 2007; Bridwell, 2007). These privacy concerns have been particularly raised when publishing maps of respondents’ locations (see, e.g., Sherman and Fetters, 2007; Loenen et al., 2016; Kim et al., 2020), but also with regard to the idea of releasing geographic locations with other collected information. Nowadays, the underlying address of geographic coordinates can be easily identified using web mapping services such as Google maps by using so-called *reverse geocoding* (see, e.g., Zandbergen, 2014, p. 2; VanWey et al., 2005, p. 15337). Reverse geocoding uses the geographic coordinate as an input and gives the corresponding address as an output.

Therefore, methods are needed that allow the use of spatial information without releasing private information. Such methods are subsumed under the term *geographic(al) masking methods* (short: *geomasking methods* or *masking methods*).

## 1.2. The Use of Geographic Masking Methods

Geographic masking methods claim to preserve respondents' privacy while maintaining spatial information (Armstrong et al., 1999, p. 501). In general, these methods alter geographic coordinates, for example, by aggregating, modifying coordinates, or releasing only contextual data (Gutmann et al., 2008). Alternative approaches, not usually referred to as geomasking methods, include spatial smoothing, multiple imputations, linear programming, use of synthetic data, controlled access to data, and flexible aggregation (Zandbergen, 2014, p. 11). These will only be discussed briefly in this thesis. If geographic masking methods are applied, reverse geocoding will lead to an incorrect address.

However, reverse geocoding is not the only approach for testing if the geomasking method is privacy-preserving. Current practice in the geomasking literature, when assessing the risk of revealing private information that can be used to identify the individual, is the concept of  $k$ -anonymity as a measure for risk of re-identification (see, e.g., Ghinita et al., 2010; Broen et al., 2021).  $k$ -anonymity states that there must be at least  $k - 1$  other records with the same attributes (Sweeney, 2002, p. 564; Samarati, 2001, p. 1013) or as applied in some masking methods as  $k - 1$  other coordinates within a certain distance for the method to be considered low-risk (Hampton et al., 2010). The latter is also referred to as spatial  $k$ -anonymity. However, as will be shown, (spatial)  $k$ -anonymity is not an appropriate measure of the risk of re-identification.

On the other hand, geographic masking methods that greatly alter coordinates can change the original spatial information associated with the data. This is commonly referred to as not preserving the utility (Duncan, Keller-McNulty, et al., 2001, p. 6). Contrary to the situation of assessing the risk of re-identification, several methods have been proposed to evaluate utility. These can be broadly categorized as descriptive statistics, preserving distance, clustering, and spatial autocorrelation (see, e.g., Armstrong et al., 1999, Seidl, Paulus, et al., 2015). Additional methods have been developed over the years, but they are not widely used. For example, visual comparisons of plotted geographic coordinates by respondents in an experiment (Leitner and Curtis, 2004; Leitner and Curtis, 2006; Seidl, Jankowski, and Nara, 2019).

This risk-utility relationship of preserving as much information as possible while eliminating the risk of re-identification can be visualized in a risk-utility map (Duncan and Fienberg, 1999; Duncan, Keller-McNulty, et al., 2001).

## 1.3. Contribution

While the benefits of releasing geographic coordinates and the need for geographic masking methods are clear, a comprehensive analysis of the risk-utility relationship of geomasking methods is missing.

Examples of previous attempts of comparing geographic masking methods, which

do not try to prove that the new proposed geomasking method is superior to others, are the publications of Armstrong et al. (1999), Kwan et al. (2004), and Broen et al. (2021). Armstrong et al. (1999) were the first to provide an overview and compare different masking methods. They compared the then-existing geomasking methods called *affine transformations*, *random perturbation*, *aggregation*, and a *nearest-neighbor method*. The utility measures used include clusters, distances, and descriptive statistics. As risk measures, they used the proportion of masked points whose distance to the original location<sup>2</sup> is closer than to any other location in the unmasked data set and the number of records needed to reveal the displacement of other geographic locations, if correctly identified. In addition, they described if knowledge about the geographic region and the masking method could be used for re-identifying the original location. The authors showed that random perturbation outperforms affine transformations in many of the measures. Moreover, for larger displacement distances the utility results of the masked data set deviated more from the results using the unmasked data set (Armstrong et al., 1999).

Kwan et al. (2004) attempted to clarify the trade-off between privacy (risk) and utility. They tested different variants of one method (called *random perturbation*), yielding different displacement distances. Their utility was mainly evaluated by comparing the clustering and by visual comparisons. For privacy, it was assumed that increasing radii correspond to increasing privacy. Their approach showed that there is indeed a risk-utility trade-off.

The recent analysis by Broen et al. (2021) used spatial  $k$ -anonymity as a measure of re-identification risk. Utility measures included descriptive statistics, clustering, and spatial autocorrelation. They analyzed geomasking methods called *Voronoi masking*, *donut masking*, *random perturbation methods*, and a *rotation method* and concluded that Voronoi masking outperforms the other tested geomasking methods.

The examples all have in common that not all of the geomasking methods available at that time are considered and that the focus lies mainly on the utility aspects rather than an in-depth analysis of the capabilities to keep respondents' location private. Therefore, the following question remains: Can we disclose the respondents' masked location to make the most use of the data without compromising their privacy?

This thesis aims to answer this question by providing an overview of all currently existing geomasking methods, comparing their utility by using several dimensions, and doing a comprehensive risk analysis beyond spatial  $k$ -anonymity. Further, the detected risk-utility relationship will be examined using risk-utility maps.

---

<sup>2</sup>In the following, the terms “original” and “unmasked” are used to refer to the location/coordinates before being masked using a geographic masking method.

## 1.4. Thesis Outline

The following chapter presents the data set used for the analysis of geomasking methods. In the subsequent chapter a framework of currently existing geomasking methods is presented in which geomasking methods are classified into three categories. Then, a description of each geomasking method follows. The description also contains a brief description of how the geomasking method is implemented and the parameter choices.

Chapter 4 presents the idea of using a risk-utility map, followed by an overview of how risk and utility are measured in this thesis using a variety of methods. Afterward, a description is given for each utility method. Before the risk methods are explained further, the scenario considered is described (section 4.2.1). The chapter closes with an overview of the analysis, containing which risk method can be applied to which geomasking method.

The results for the different utility methods as well as the risk methods are summarized in chapter 5. First, a general comparison between geomasking methods is made. Next, the results for each of the utility measures for each parameter choice of the geomasking methods are described. Then the utility results are aggregated by masking methods for a better comparison. After that, the results for the risk measures are shown. Concluding again with the aggregated results by the geomasking method for better comparison.

Finally, chapter 6 shows the results for aggregating the utility into one value and risk assessment results into one value for the risk-utility map. As explained in chapter 4, the utility results are aggregated in this thesis using two measures, resulting in two risk-utility maps. These maps are shown in subsections 6.2 and 6.3. The thesis concludes with a discussion, an answer to the above-stated question, limitations, and an outlook on future research.<sup>3</sup>

---

<sup>3</sup>In this thesis, the data sets (and their preprocessing), the geomasking methods, and the utility and risk methods used are described in detail. The functions written for the geomasking methods and the detailed results can be found in the appendix of this thesis. For other supplementary material, such as the code for the risk and utility methods and the data sets (due to the large amount of this material), please contact [sarah.redlich@uni-due.de](mailto:sarah.redlich@uni-due.de).



## 2. Data Set and Software

A data set is needed to which the masking methods can be applied and subsequently evaluated. The data set should contain a large number of real addresses since some of the methods require knowledge of all residential addresses in the study area. It should also include additional information to implement some of the masking methods and is needed for utility and risk measures.

This chapter first provides a detailed description of the data set used. The following section describes the sample used for applying the masking methods (as well as subsamples due to size limitations of some risk methods) and the identification file used for assessing the risk. The following section briefly explains the data sets' coordinate system and the coordinate system the data set is converted to, which is required for some masking methods. A short description of the software *R* that is used for the analysis is then given.

### 2.1. Data Set

Due to strong data protection restrictions, a data set that meets the above requirements is not available for Germany. An alternative was, therefore, searched for and found with the *PSMA Geocoded National Address File* (G-NAF; August 2019; Department of Industry, Innovation and Science (2019b)).<sup>1</sup> G-NAF contains all residential addresses for Australia, divided by state. This data set is updated every few months. The information is divided into 19 data sets for each state, all of which can be combined based on different identifiers. In addition, census data packs and the Mesh Block 2016 data set are used to enrich the data set with variables needed for utility and risk analyses and some masking methods. The coordinates are in the datum<sup>2</sup> GDA94 using the ellipsoid GRS80.<sup>3</sup>

Since considering all of Australia would have resulted in a much too large data set for the masking methods, it was decided only to use South Australia (population size as of 30. June 2021: 1,773,200).<sup>4</sup> As described below, this still resulted in a large

---

<sup>1</sup>For finding the G-NAF I have to thank Prof. Dr. Peter Christen and Prof. Dr. Rainer Schnell.

<sup>2</sup>A geodetic datum is a reference frame that provides information about the used earth model, the origin, the orientation and the scale (Flacke et al., 2015, pp. 22, 86–87).

<sup>3</sup>Later releases of the data set were updated to the GDA2020 datum, which is more accurate than the GDA94. However, due to the time already invested in preprocessing the data set, applying the masking methods, and analyzing, the August 2019 data set was used.

<sup>4</sup>See, e.g., <https://www.abs.gov.au/statistics/people/population/national-state-and-territory-population/latest-release#states-and-territories> (retrieved 16.02.2022).

number of residential addresses to consider for the respective masking methods.

The data set was preprocessed by first, merging the individual files for South Australia to obtain a data set containing the latitude and longitude, the postcode, the mesh block ID, the locality name, the building name, and a variable indicating whether other sources could validate the address.

Using the Mesh Block 2016 data set (Australian Bureau of Statistics, 2016a), the Statistical Area 1 ID is merged.<sup>5</sup> The Statistical Area 1 divides Australia into areas of approximately 200 to 800 people. It provides information on the use of the property at the respective address, e.g., whether it is used commercially, residentially, medically, educationally. When conducting a survey, often only residential addresses are considered. Therefore, addresses with the information that they were not used for residential purposes were deleted.

The variable building name indicates whether the building at the respective address has a name, e.g., is a high school, museum, or hospital. Furthermore, there are also residential addresses with a building name. These buildings appear to be predominately retirement buildings or villages.<sup>6</sup> People living in retirement buildings (institutional population) are usually excluded from the surveyed population (see, e.g., Schnell, 1991). Therefore, these addresses are also excluded.

There is also a variable that indicates whether the address could be verified using other sources. According to the G-NAF product description, addresses that could not be verified were usually deleted over time and are currently only kept to show changes throughout the years. Since this is not of interest, addresses that could not be verified by at least one other source are also deleted. Before other variables could be added, the data set had to be cleaned further by removing duplicate addresses.<sup>7</sup> In total, the G-NAF is reduced from 1,153,801 lines to 756,509 lines.<sup>8</sup>

Since some of the masking methods require the knowledge of the population density, using the census data packs (Australian Bureau of Statistics, 2016b), the population on postcode level and the population on local government area level is added to the data set (area level for which such an information is usually available). The size of the area is also added, using the shapefiles provided with the data packs, which is necessary to obtain the population densities.<sup>9</sup>

For calculating Moran's I (used among other measures to evaluate utility) and

---

<sup>5</sup>For all added information, the most recent available data set at the time of the preprocessing was used.

<sup>6</sup>This was verified by manually looking at all residential addresses with a building name.

<sup>7</sup>The excluded lines were not an indication of multiple people living at the same address, but simply the same coordinates but different information on variables that were not needed.

<sup>8</sup>Thus, 34.43% of the lines were deleted: 20.63% of non-residential addresses, an additional 1.91% of buildings with a name, suggesting that the address is not used as residential address, 2.56% unverified addresses, 9.34% duplicates, and one address, because the population information added at postcode level was zero.

<sup>9</sup>No information could be found in the Census packs for the postal code 5611. This information was obtained using the Australian Bureau of Statistic's Quickstats based on the census 2016 (Australian Bureau of Statistics, 2017).

applying certain masking methods additional variables were then added to the data set. In the following, for each variable, the source of the variable and how it was added are described.

Based on the Census data packs, the proportion of single households and the proportion of full-time workers on the Statistical Area 1 level is added as the continuous variable for Moran's I, as this was the best variable available.

Additional variables are needed for the verified neighbor masking method and some risk measures, which identifies residents with the same characteristics as the person living at the sampled address. Since such detailed information is not available, an approximation is used. The full file of 756,509 addresses is treated as the residential population, from which a sample is drawn.<sup>10</sup>

The Census data packs provide several variables at different regional levels. The variables needed should be variables that can be expected to be found in any data set. Therefore, demographic variables were taken. Also, the number of people per category should be large enough to ensure  $k$ -anonymity in the data set, but at the same time, enough categories are needed to distinguish between people. Usually, sex and age categories are available. However, if only sex and age categories are considered, the individual groups (combinations of the categories of the variables) would be too large to distinguish between individuals. As a third variable, to create more groups, different variables could be chosen: educational level, marital status, employment status, country of birth, and so on. Many of these variables have a high percentage of the Australian population in one category, such as country of birth being Australia, or have too many categories to ensure  $k$ -anonymity (for the sample needed), such as educational level. For employment status, the population was more evenly distributed among the categories. Therefore, employment status was chosen.

The number of men and women, employment status, and age categories are known at the postcode level. The ratios are used to randomly assign the addresses in each postcode to the values of the variable in approximately equal proportions. Note that the variables were not assigned independently, but the combination of the values was preserved. For sex the categories "female" and "male" are used, for employment status "full-time", "part-time", "away from work", "unemployed", and "undefined" are used. The last category ("undefined") is the result of the sum of the other categories, not adding to the number of people in the postcode, e.g., children or the elderly. The age categories are "15-19 years", "20-24 years", "25-34 years", "35-44 years", "45-54 years", "55-64 years", "65-74 years", and "75+ years".<sup>11</sup> For example, if 5% of the population in the postcode 5611 were 25-34 year old, female part-time workers, randomly chosen

---

<sup>10</sup>A true residential population would consider multiple people living at the same coordinates due to a shared household or multiple households in the same building. However, such a detailed data set is not available. But for most cases, third variables should provide enough information to identify the correct person once the address is known.

<sup>11</sup>Note that the census data packs have the categories "75-84 years" and "85+ years". However, since at that age the categories of the employment status will show very few people still working, making it more difficult to ensure  $k$ -anonymity in the sample, the age category was limited to "75+ years".

5% of the addresses in the postcode 5611 would be assigned the sex “female”, the age “25-34 years” and the employment status “part-time”.

A sample is then drawn to which the masking methods are to be applied. In reviewing the literature, especially articles focusing on testing and comparing masking methods, there is no consensus on the number of addresses needed to perform a comprehensive analysis of the masking methods. Several authors use only up to 500 addresses (see, e.g., Seidl, Jankowski, and Clarke, 2018, Richter, 2017). In one article where the authors report the size of the data set and the size of the sample, it shows that less than 1% (between 640 and 1465 cases) of the data set was used (Zhang, Freundsuh, et al., 2015, p. 5). The sample size considered here is arbitrarily set to 10,000 (see figure 2.1 for population and sampled points).<sup>12</sup> From these  $n = 10,000$  addresses a subsample of  $n = 1,000$  and  $n = 2,000$  was drawn as needed for subsequent risk analysis.

For the risk analysis, another sample was drawn from the residential address file with an overlap of 10% to the masked file.<sup>13</sup> All samples were drawn as a simple random sample. Due to the comparatively few categories for the additional variables and the large numbers of individuals with the same combination,  $k$ -anonymity was easily maintained without needing a more complex sampling such as a stratified sampling.

### Data Set to be Masked

The data set to be masked contains 10,000 points, and the combination of sex, age, and employment status (referred to as groups) results in 80 different groups. The minimum number per group is three, as suggested for example by the official statistics as minimum number for the reported number of cases in a table (Rothe, 2015, p. 299) so that there are at least three individuals with the same characteristics in the data set. For some risk methods, a smaller sample is needed. Therefore, a subsample of 1,000 points, as well as 2,000 points, was drawn. The combination of sex, age, and employment results in 74 distinct groups for the subsample  $n = 1,000$ , and the  $k$ -anonymity is preserved with at least four individuals per group. The subsample with  $n = 2,000$  addresses has 72 distinct groups with at least 3 individuals per group.

### Identification File

The identification file contains 10,000 points with 78 different groups. Again, there are at least three individuals with the same characteristics in the data set. For a subset of 1,000 points, the number of groups is 74, with at least five individuals per group, and for  $n = 2,000$  there are 74 groups with at least three individuals per group.

---

<sup>12</sup>The larger sample size, compared to other analyses in literature, was chosen to also review how masking methods perform for larger sample sizes.

<sup>13</sup>An explanation for the choice of the overlap and a more detailed explanation of the scenario considered can be found in chapter 4.2.1.

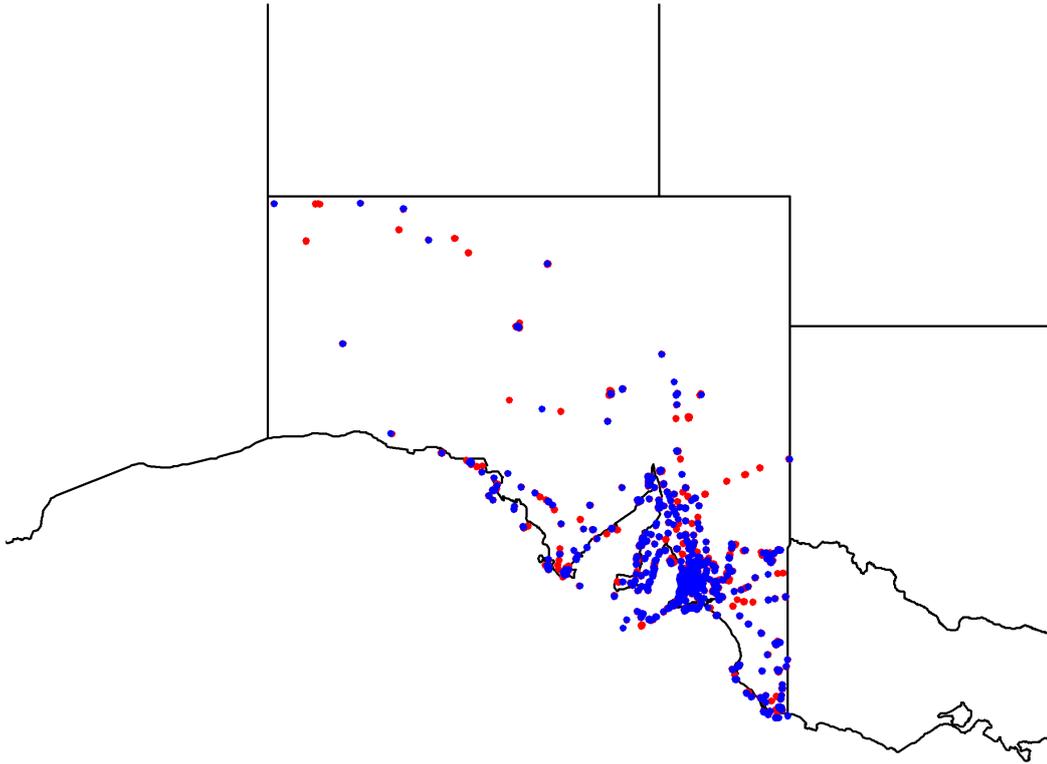


Figure 2.1.: Map of South Australia. Red points are residents and blue points are sampled points ( $n = 10,000$ ). Points, which seem to be located in the ocean, are located on islands.

### Overlap Between Masked and Identification File

The overlap (see section 4.2.1 for more information) between the masked and identification file is 10% of the data sets, i.e., 1,000 for  $n = 10,000$ , 100 for  $n = 1,000$ , and 200 for  $n = 2,000$ . The overlap for  $n = 10,000$  results in 75 different combinations of sex, age, and employment (see table C.1 in appendix C). The overlap does not have at least three individuals per group. However, this is not a requirement of the scenario. For  $n = 1,000$  the overlap results in 19 different combinations (see table 2.1 for more information) and for  $n = 2,000$  there are 36 different combinations.

## 2.2. Coordinate System

The earth is an ellipsoid and is usually represented in a simplified form as a globe (Flacke et al., 2015, p. 96). A projection is used to display the three-dimensional earth surface in a two-dimensional flat surface (Flacke et al., 2015, p. 96). If a globe is cut open and laid flat, it can be seen that some parts of the map are subject to distortion (Flacke et al., 2015, p. 96). The theoretical goal is to find a projection that is conformal, equivalent, and equidistant (Flacke et al., 2015, p. 99). A map is conformal if angles are preserved when projecting a round surface onto a flat surface, which is necessary for navigation. It is equivalent if areas are preserved (scale reductions

Table 2.1.: Combinations of sex, employment, and age of the overlap of the masked and identification file ( $n = 1,000$ , sorted by frequency of overlap). The fifth and sixth column contains the number of people with the respective combinations in the masked and identification file.

sex	employment	age	freq overlap	freq masked	freq ident.
F	full time	45-54	10	25	22
F	undefined	15-19	9	25	27
F	part time	25-34	7	19	16
F	part time	45-54	7	20	23
F	part time	35-44	6	24	21
F	undefined	25-34	6	21	22
M	full time	55-64	6	23	19
M	undefined	65-74	6	25	18
F	part time	55-64	5	23	18
F	undefined	55-64	5	20	19
M	undefined	15-19	5	18	17
F	full time	55-64	4	19	8
M	full time	25-34	4	24	15
M	full time	35-44	4	19	12
M	full time	45-54	4	17	23
M	undefined	55-64	4	11	14
F	undefined	65-74	3	15	21
M	undefined	75+	3	20	15
F	undefined	75+	2	18	14

are allowed), which is necessary for geographical comparisons. A map is equidistant if distances are preserved, necessary for measuring distance. In reality, none of the projections achieve all three goals, although many projections have been proposed. There are conic projections, azimuthal projections, cylindrical projections, and many more (Flacke et al., 2015, pp. 96–97).

As mentioned before, the data set is in the GDA94 system using the GRS80 ellipsoid, and latitude and longitude coordinates in decimal degrees (Department of Industry, Innovation and Science, 2019a). The problem with latitude and longitude coordinates is that they can not easily be moved by a certain distance (i.e. adding the same number to all coordinates to move all coordinates by the same amount), which is required for some masking methods (e.g., for affine transformation). Thus, a transformation of the coordinates in meters is needed. The coordinates are converted to the GDA94 / SA Lambert System with the ellipsoid GRS80 (EPSG:3107), where the coordinates are in easting and northing.<sup>14</sup> The masked coordinates will always be converted back to the coordinate system specifications of the original data set.

<sup>14</sup>In theory, an alternative would be to use the UTM system. However, using the UTM system is not feasible for large study areas covering more than one zone (Bertici et al., 2014), as given in this thesis. This was also verified in a personal conversation with a research assistant of the Institute of Geography of the University of Duisburg-Essen (Birgit Sattler, 17.10.2018).

## 2.3. Software

A common software in the field of geography is *ArcGIS*, a commercial software that allows people without knowledge in programming to work with geographic information. A free alternative, which is the software used for the implementation of the masking methods and analysis, is *R*. However, the majority of the masking methods are not implemented in *R* (not all are implemented in *ArcGis* either).<sup>15</sup> The corresponding code for each masking method is provided in appendix A.<sup>16</sup>

## 2.4. Parameter Choice

Since most of the masking methods will yield different results each time they are applied, applying each masking method only once does not reveal the variability of the masking methods in terms of their utility and risk. With only one application, one may get the one application out of many that preserves the utility the most or the least. Therefore, the masking methods are applied multiple times for every input parameter (set) chosen. It was arbitrarily chosen to apply each masking method per input parameter (set) 50 times, e.g., when a random number was needed, a new random number was drawn for each replication.<sup>17</sup> The input parameter (sets) are described after each explanation of the geomasking methods in the following chapter.

---

<sup>15</sup>Just recently Swanlund, Schuurman, and Brussoni (2020) developed the tool *MaskMy.XYZ* to make geomasking methods more accessible to use. The tool can apply affine transformations, random perturbation, donut masking, verified neighbor approach, location swapping, and adaptive areal elimination. Also, Fronterré (2018) provided *R* code for the implementations used in his corresponding PhD thesis.

<sup>16</sup>See chapter 3 for brief descriptions on how each masking method was implemented.

<sup>17</sup>50 seemed to yield reasonable stable results and allowed to choose multiple different input parameters (parameter sets) without the problem of excessive computational times.



### 3. Geographic Masking Methods

In this chapter, current classifications of masking methods are briefly presented, followed by a description of the classification used. Subsequently, each masking method is presented in detail. After each masking method, a brief description of the implementation and the parameter choice is given.<sup>1</sup>

Over the years, many masking methods have been proposed to protect geographic information. The first overview of masking methods was given by Armstrong et al. (1999). Since few masking methods existed, the authors proposed to categorize them into five categories: individual and concatenated affine transformations, random perturbation, aggregation, neighbor information, and contextual information (Armstrong et al., 1999).

In 2008, Gutmann et al. proposed using only three categories: aggregation, adjusting coordinates, and attaching contextual variables (and removing coordinates). Gupta and Rao (2020) provide a more recent overview. They expanded the number of categories to six: aggregation, random perturbation, blurring, affine transformations, flipping, and encryption. Although Gupta and Rao (2020) attempted to give a complete overview of all existing masking methods, they failed to do so. Missing masking methods (which existed long before) include Voronoi masking (Seidl, Paulus, et al., 2015), location swapping (Zhang, Freundsuh, et al., 2015), and verified neighbor approach (Richter, 2017). Furthermore, the missing masking methods do not fit in any category of this classification, since each masking method is more or less classified as its own category. Similarly, Kounadi and Leitner (2015) used six categories: aggregation, affine transformations, random perturbations, flipping, neighbor information, and “other”. Again, their overview does not include all masking methods and classifies many masking methods in their own category.

All methods must first be found for the categorization. To find masking methods, guidelines such as PRISMA (see, e.g., Page et al., 2021) can be used for literature searches. However, in the present case, many papers mention geographical masking methods, but as can be seen by the number of methods in this chapter, very few present a new idea for a masking method. As a starting point, the first overview by Armstrong et al. (1999) was used. Based on this article and the publications that cited it, other reviews were found and the masking methods they contained were gathered. Additionally, Google scholar and the databases PubMed, Web of Science,

---

<sup>1</sup>It should be noted that in the following the words “points” and “location” are used interchangeably when talking about “coordinates”. Furthermore, this chapter and the overview are current as of July 2021 (submission of this thesis).

and Scopus were used to search for the terms “geomasking method”, “geographical masking method”, and “geographic masking method”, which obtained the masking method based on the military grid reference system (MGRS) (Clarke, 2016) and triangular displacement (Murad et al., 2014). Later this search strategy was repeated to include methods that have been published after the initial search. Apart from the method street masking (Swanlund, Schuurman, Zandbergen, et al., 2020), this only yielded small variations of existing methods.

The masking methods found are categorized oriented at Gutmann et al. (2008) (see figure 3.1).<sup>2</sup> Similar methods, not usually referred to as geomasking methods (Zandbergen, 2014, p. 11), are not included in the overview. However, they are briefly described at the end of this chapter. The first category *aggregation* aggregates coordinates to a certain level. Hence, multiple coordinates will receive the same masked coordinate. Both, *point aggregation* as well as *areal aggregation* can be subsumed under this category.

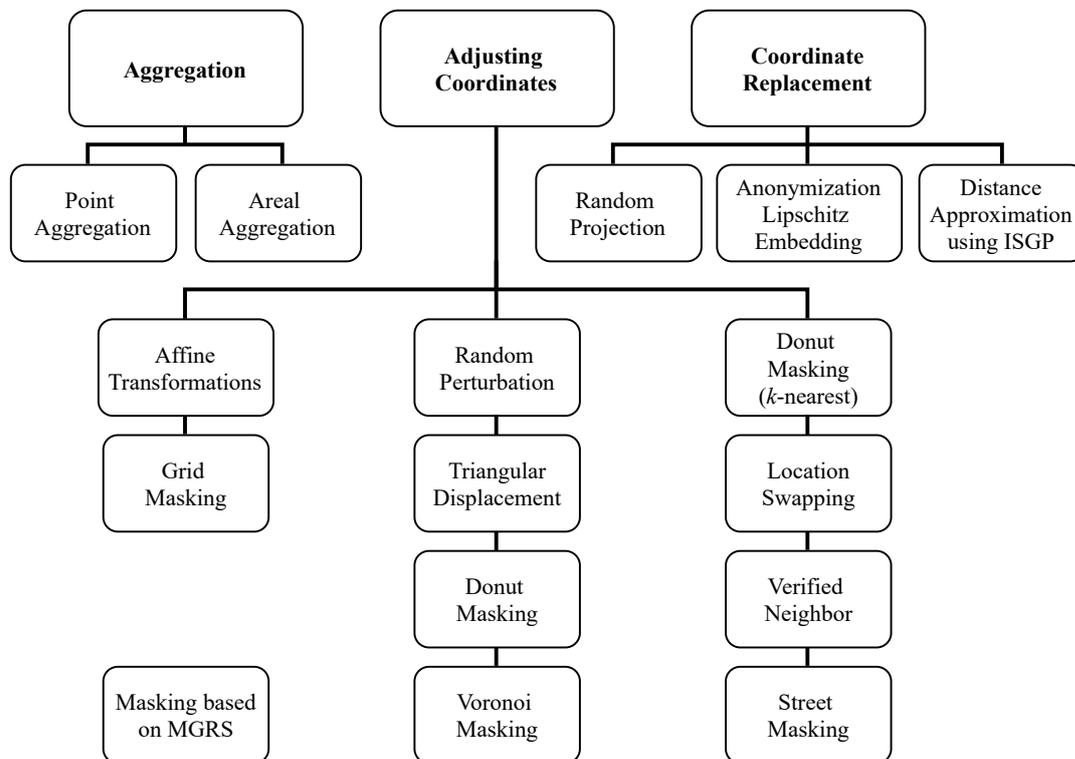


Figure 3.1.: Categorization of masking methods (oriented at Gutmann et al., 2008).

The second category is termed *adjusting coordinates*. The majority of the masking methods take the approach of adjusting coordinates rather than aggregating or replacing them with other information. Therefore, this category is divided into subcategories.

The first subcategory contains masking methods that scale, rotate, displace or flip coordinates. This subcategory includes *affine transformations* (Armstrong et al., 1999)

<sup>2</sup>A more comprehensive overview of masking methods can be found in appendix A.1.

and *grid masking* (Leitner and Curtis, 2004).

The second subcategory comprises methods that move points into a random direction by a random distance. This subcategory includes *random perturbation* (Armstrong et al., 1999), *triangular displacement* (Murad et al., 2014), *donut masking* (Stinchcomb, 2004; Hampton et al., 2010), and *Voronoi masking* (Seidl, Paulus, et al., 2015). For Voronoi masking, this is not quite correct, since the displacement depends on the coordinates' distance towards each other. However, due to the lack of a fixed input parameter for the coordinates for the displacement distance, Voronoi masking fits best in this category.

The last subcategory, similar to the previous one, also displaces coordinates into a random direction by a random distance, but in addition it requires the information of the surrounding residents. This subcategory contains the method *k-nearest neighbor donut masking* (Hampton et al., 2010), *location swapping* (Zhang, Friendschuh, et al., 2015), *verified neighbor approach* (Richter, 2017), *street masking* (Swanlund, Schuurman, Zandbergen, et al., 2020), and *adaptive random perturbation* (Kounadi and Leitner, 2016). The latter is one of two methods subsumed under the masking method *adaptive areal elimination* which falls into the category aggregation. Therefore, it is not shown in figure 3.1 and will be explained in the subsection of aggregation. Nonetheless, it fits better in the category adjusting coordinates (if viewed on its own). The only masking method, that does not quite fit into this framework is *masking based on the military grid reference system* (Clarke, 2016). As will be seen, this is a masking method that is only applicable under certain conditions that are rarely met when using data sets that cover larger regions.

The third category of Gutmann et al. (2008) is called *attaching contextual variables*, where additional information is added to the data set based on the coordinates, and then coordinates are removed. This category also includes methods that generate a modified distance matrix without releasing any coordinates. Since this is not readily apparent from the name, the category name is replaced by the name *coordinate replacement*. This category comprises *random projection* (Henecka, 2019), *anonymization of geographical distance matrices via Lipschitz embedding* (Kroll and Schnell, 2016) as well as *distance approximation using intersecting sets of grid points* (Schnell, Klingwort, et al., 2021).

In the following, each of the masking methods is described in the order of their occurrence in the categorization shown in figure 3.1. Whenever possible, an example is provided.<sup>3</sup> Over the years, many variants of the presented masking methods have been proposed, e.g., using different distributions or different area definitions. These are summarized in their respective section. Also, a brief description of the implementation (including the parameter choice) is given after each masking method.

---

<sup>3</sup>An overview of the majority of the masking methods was presented at the LIfBi by Prof. Dr. Schnell (Schnell and Redlich, 2019).

## 3.1. Aggregation

Over time, several different aggregation methods have been proposed. A rough distinction can be made between point and areal aggregation. These two differ mainly in that in point aggregation, a point (e.g. the average) represents several other points, while in areal aggregation, the coordinates are replaced by the areal identifier (Armstrong et al., 1999, pp. 506–507).

### 3.1.1. Point Aggregation

Methods that are classified as point aggregation methods are microaggregation, blurring, and adaptive areal elimination. The latter can be further subdivided into adaptive point aggregation and adaptive random perturbation.<sup>4</sup>

#### 3.1.1.1. Microaggregation

*Microaggregation* is a method proposed by Wolf (1988). For this method, a proximity measure is needed, which defines the distance between points. For example, the Euclidean distance can be used. Then the following steps are performed for each of the points separately. First, all neighboring points are aggregated. A neighboring point has a distance to the target point that is less than a predefined threshold. The newly created record of the aggregated points also receives aggregated values for the data sets' other variables. These are calculated, e.g., using a weighted average. Subsequently, either all of the points that have been aggregated are deleted, or they remain in the data set, and the next point is viewed (Wolf, 1988, 355 f.).

The method is not limited to coordinates. The description of the method of Wolf (1988) is wide so that variables with other information could be masked similarly. Furthermore, the distance measure and the aggregation method can be chosen freely (Wolf, 1988, p. 355). However, this method intends to aggregate not only coordinates but values of other variables as well. A variant of microaggregation is *maximum distance to average vector* (MDAV; see figure 3.2). A comprehensive description of this method can be found in Domingo-Ferrer and Torra (2005). In general, there are two steps (Domingo-Ferrer and Torra, 2005, p. 203): first, similar elements are grouped into clusters with at least  $k$  elements using a distance measure. Second, the information within the cluster is aggregated. The authors propose to use the arithmetic mean as aggregation method and the Euclidean distance as distance measure.<sup>5</sup>

---

<sup>4</sup>Although adaptive random perturbation is placed in the category “adjusting coordinates” of the framework, it is part of adaptive areal elimination, therefore, it will be explained in this subsection.

<sup>5</sup>There are many variants for microaggregation. In the following, the word microaggregation is used for the method maximum distance to average vector (MDAV).

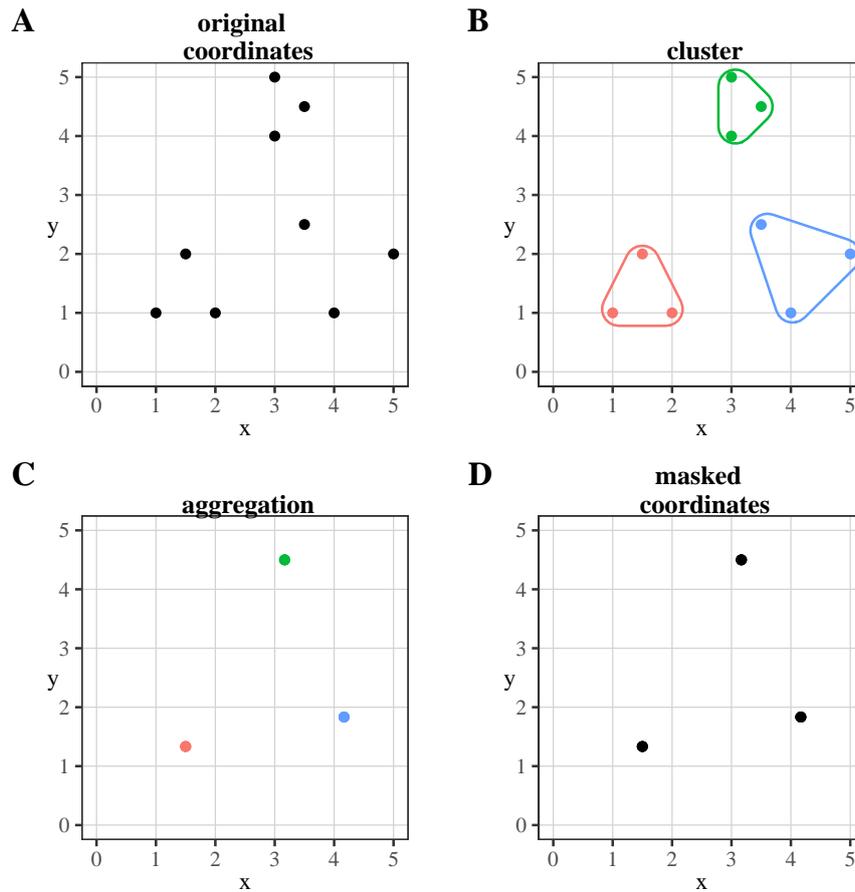


Figure 3.2.: Maximum distance to average vector (MDAV).

### Implementation of Maximum Distance to Average Vector (short: MDAV)

MDAV has been implemented in the *sdcMicro*-Package (Templ et al., 2021) in *R*. However, MDAV is usually not used for location data. The variables to be aggregated are considered independently. It also uses the Euclidean formula for the distance measure, which in turn requires the conversion of the coordinates into meters (such as easting and northing).<sup>6</sup> The corresponding code for this and the other implementations is found in appendix A.

For the number of points per cluster ( $k$ ) the minimum number of three, common in official statistics, was chosen (see, e.g., Rothe, 2015, p. 299). In addition, larger cluster sizes of 25 and 50 are considered, resulting in 400 and 200 clusters, respectively.

#### 3.1.1.2. Blurring

*Blurring* was introduced by Strudler et al. (1986) and applied to spatial data sets by Armstrong et al. (1999). In blurring, variables are categorized (Strudler et al., 1986, p. 377). For spatial data, the latitude and longitude are independently sorted from

<sup>6</sup>More information on coordinate systems can be found in chapter 2, where the data set used is described. The phrase “coordinates in meters” is used whenever a coordinate system is meant for which the unit is meters instead of (decimal) degrees (latitude-longitude coordinates).

the smallest to the largest. Then each of the coordinates is replaced by its group average. The groups can be defined as fixed, non-overlapping intervals or as sliding intervals that vary for each point (Armstrong et al., 1999, p. 506). For the sliding interval, one possibility would be to take the  $n$ -nearest points. Another option is to take the points within a certain distance (Armstrong et al., 1999, p. 506). The latter requires finding the maximum distance to the nearest neighbor so that every interval contains at least two points.

Blurring is similar to MDAV, as it aggregates near coordinates. Blurring is very flexible in its definition of the interval and its size. Apart from an arbitrary number for the example, no suggestion is made by the authors on the size of the interval (length of the interval or number of points). Due to the strong similarities to MDAV and the fact that microaggregation methods are commonly used for point aggregation, only MDAV is tested in this thesis.

### 3.1.1.3. Adaptive Areal Elimination

*Adaptive Areal Elimination (AAE)* is a masking method by Kounadi and Leitner (2016) that uses the idea of grid masking, as described in a subsequent chapter. First, the area of interest is divided into smaller sections, for example, street segments or administrative units, using an external data set. As an alternative to polygons, a grid can be laid over the data points. Then, using a predefined  $k$ -anonymity threshold,<sup>7</sup> polygons are merged by the largest shared border if the number of (residential) addresses in a polygon is below the threshold. For equilateral polygons (using a grid), they are merged with all adjacent polygons. Polygons are merged until each polygon has the required number of points (Kounadi and Leitner, 2016, 61 f.). Additionally, the authors propose to disclose which parameters and which masking method was used (Kounadi and Leitner, 2016, p. 61).

Croft et al. proposed 2015, 2016, and 2017 the *Voronoi-Based Aggregation Systems (VBAS)*. This method creates regions by choosing the number and location of so-called sites, which are then used to form Voronoi polygons. Points that fall within a Voronoi polygon are then aggregated into a region. The number of sites can be freely chosen or based on any method, such as population divided by the dynamic geographic area population size cutoff approximation as described in El Emam et al. (2009). For the location of sites, for example, the authors propose using the balanced density approach by creating equally sized (in terms of population) regions and using the spatial median center as site location. The resulting areas can then be used for further analysis (Croft et al., 2015; Croft et al., 2016; Croft et al., 2017).<sup>8</sup>

Kounadi and Leitner (2016) then describe two methods for masking the points

<sup>7</sup>The authors call this the “RoRi” value, but for point data, this seems to be equivalent to  $k$ -anonymity (Kounadi and Leitner, 2016, p. 62).

<sup>8</sup>If no subsequent steps are taken after regions are created, this would be considered an areal aggregation method.

within the polygons. *Adaptive Point Aggregation (APA)* computes the centroid for each polygon separately and replaces the coordinates of the points that fall within this area with the centroid of the area (Kounadi and Leitner, 2016, p. 61). An example with equilateral polygons and a threshold of  $n = 2$  is shown in figure 3.3. The other method of adaptive areal elimination is *Adaptive Random Perturbation (ARP)*; see figure 3.4), which defines that the points should be moved randomly within the boundaries of the polygons (Kounadi and Leitner, 2016, p. 61). As opposed to the method grid masking (see chapter 3.2.2) of having equally sized areas (squares), the areas can be polygons of different sizes.

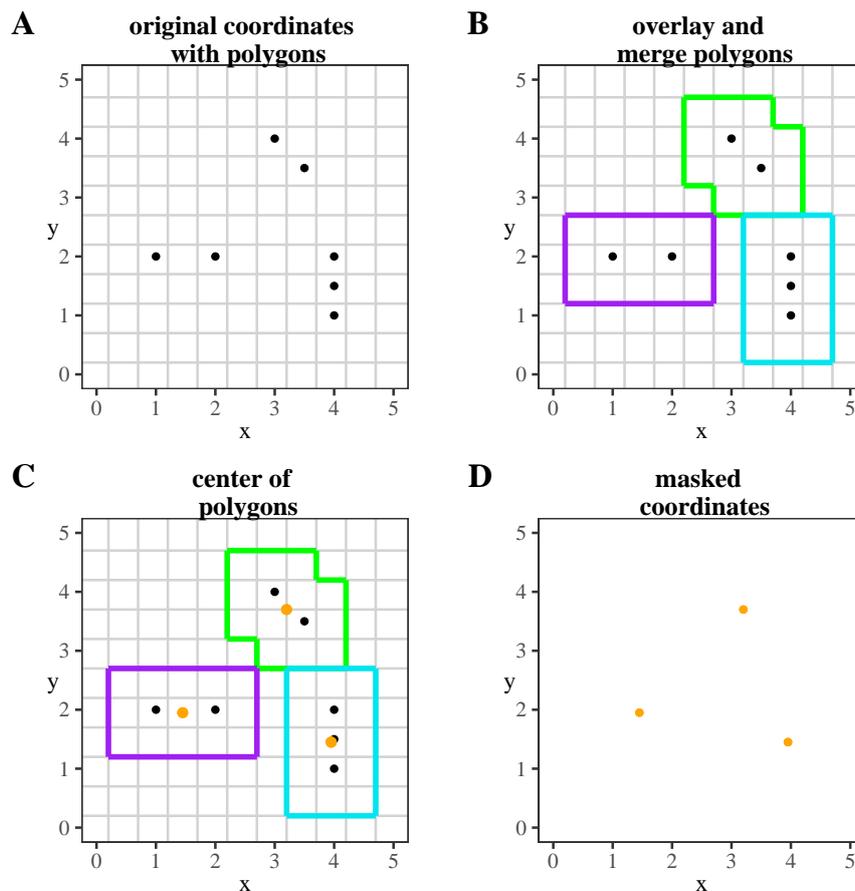


Figure 3.3.: Adaptive Point Aggregation. Threshold: 2 points per polygon. Grid are the overlaid “polygons” (equilateral polygons). Purple, green and blue lines are borders of the merged polygons. Orange points are masked points.

More recently, Charleux and Schofield (2020) have pointed out a problem of adaptive areal elimination and attempted to solve it with a new method. The problem they found is that polygons with the number of points below the  $k$ -anonymity threshold need to be merged and are potentially merged with a polygon with points above the  $k$ -anonymity threshold. However, this makes the polygon with enough points much larger than it needed to be and thus, increases the potential displacement distance. Furthermore, processing time increases when polygons need to be merged since the

largest shared border has to be found. The solution is called *adaptive areal masking (AAM)* and merges polygons not by the largest shared border but with the polygon with the closest centroid (Charleux and Schofield, 2020, pp. 538–539). Thus, the only difference to adaptive areal elimination is the rule according to which polygons are merged. For the given data set, all polygons had at least  $k$ -points, and therefore, this method was not needed.

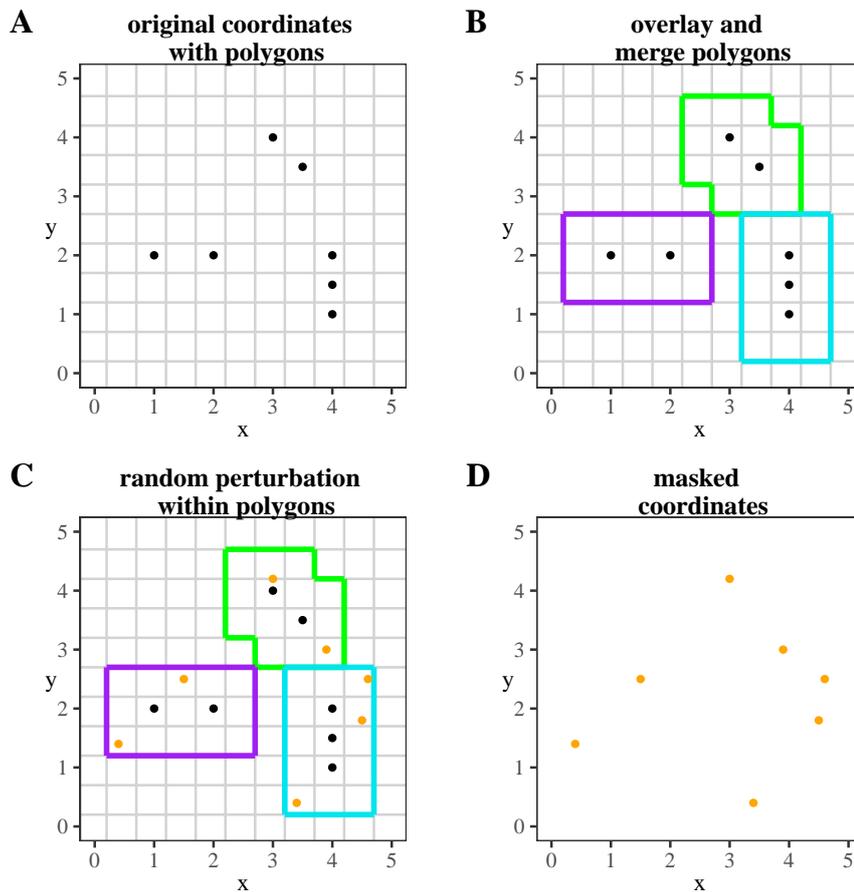


Figure 3.4.: Adaptive Random Perturbation. Threshold: 2 points per polygon. Grid are the overlaid “polygons” (equilateral polygons). Purple, green and blue lines are borders of the merged polygons. Orange points are masked points.

Moving coordinates within a grid satisfying  $k$ -anonymity was also proposed as an idea by Chen et al. (2017) and implemented in the *GeoMasker tool*. Here, a grid is placed over the data set. In grid cells with at least  $k$  number of points, the coordinates are moved randomly within the grid cell. Grid cells with less than  $k$  number of points are aggregated until the resulting grid cell contains at least  $k$  points. Then these points are moved randomly within the grid cell. This is exactly what Kounadi and Leitner (2016) describes as adaptive random perturbation when using a grid as polygons.

Houfah-Khoufah and Touya (2020) provide a “new” approach<sup>9</sup> (called: *simulated*

<sup>9</sup>When this thesis was submitted the approach could only be found in a preprint of an article. It has recently been published (October 2021).

*crowding*) that is similar to the adaptive random perturbation. In their case, they use the polygons identified in the original data by the clustering algorithm DBSCAN (Ester et al., 1996). Within these polygons, as many new points as original points are randomly drawn. In addition, a buffer around each point can be used to ensure that points are moved at least a certain distance. Again, this is a method that defines the polygons differently but is similar to adaptive random perturbation and donut masking (see subsection 3.2.5).

In the following, only adaptive point aggregation and adaptive random perturbation are analyzed since all other methods in this subsection are very similar variants in terms of how polygons are created, used, or merged. In this thesis, different polygons are used, to ensure that the influence of the chosen polygons is viewed.

### **Implementation of Adaptive Areal Elimination** (short: APA and ARP)

First, the coordinates are assigned to their respective polygon. Some points may not fall within the borders of the polygons due to some inaccuracies. This can be compensated for by assigning points to the nearest polygon. For adaptive point aggregation, the centroids of the polygons are calculated and used as masked coordinates.

For the adaptive random perturbation, several random points are drawn within the polygons. Since the data set used in this thesis is from a region also consisting of islands, the randomly generated points can be located between the mainland and an island. Therefore, more random points than needed are drawn and checked to see if they are within the polygons' boundaries within the larger polygon. If not, indicating that the sampled location is within an ocean or a sea, the corresponding random points are not considered as possible displacement locations. A random sample is drawn from the remaining points according to the number of points needed in the polygon. This is equivalent to randomly moving a point within the boundaries of the polygon without having to solve the problem of defining maximum displacement distances in polygons of arbitrary shapes.<sup>10</sup> Adaptive point aggregation and adaptive random perturbation can be applied using latitude-longitude coordinates, thus, without the need to transform coordinates into a system fulfilling special requirements.

For the polygons, the state electorates from 2018 and the local government areas (LGA) of August 2016 were chosen.<sup>11</sup> Since polygons always contained at least three residents, no polygons were merged.<sup>12</sup> No other input parameters are required. Unlike other masking methods that use the regional levels of zip code, state electorates were chosen to show the impact of a few polygons of larger sizes.

---

<sup>10</sup>This approach has also been suggested by Zandbergen (2014) for the random perturbation method.

<sup>11</sup>The shapefile for the local government areas is part of the census data packs. For state electorates, an updated version (of 2018) could be found.

<sup>12</sup>The official statistics considers records to be indistinguishable if three records share the same value for a variable of interest (Rothe, 2015, p. 299).

### 3.1.2. Areal Aggregation

Methods classified as areal aggregation methods are contextual information and the official statistics grid.

#### 3.1.2.1. Contextual Information

*Contextual Information* is a method proposed by Saalfeld et al. (1992). It aggregates the values of variables of all points within a predefined area. This area is called *context*. An area is divided into contexts by using partitions or windows. Partitions are non-overlapping such as the states of a country. All points within a partition are given the same value (Saalfeld et al., 1992, p. 691). Windows can overlap because they are defined by a fixed radius around each point. Due to the possibility of overlapping windows, points that are close to each other get similar but not necessarily identical values (Saalfeld et al., 1992, p. 692). It is not clear from the description of Saalfeld et al. (1992) whether this should be done to coordinates at all, as Armstrong et al. (1999, pp. 509–510) suggest, or whether the coordinates should instead be replaced by a variable containing information about the neighborhood. Giving all points within a region the same value is similar to adaptive point aggregation. The “similar value” could also be an identifier of the region, such as in the official statistics grid (as explained below). Since adaptive point aggregation and the official statistics grid are analyzed, the contextual information method is not tested in this thesis.

#### 3.1.2.2. Official Statistics Grid

For official statistics in Germany (and the European Union), a specific form of aggregating coordinates by area is used. A grid is overlaid on the data, and the coordinates to be masked are replaced by an ID formed from the lower-left corner coordinates or center coordinates of the cell in which the coordinates to be masked fall. The cell width can vary, but usually, a cell width of at least 100 meters is considered (see, e.g., Gebers and Graze, 2019).

This grid method was developed as part of the European initiative INSPIRE to establish a spatial data infrastructure (Bundesamt für Kartographie und Geodäsie, 2020). The coordinate system used is ETRS89-LAEA (EPSG:3035). However, INSPIRE refers to national coordinate systems that can be used, such as UTM, Zone 32 (EPSG:25832) for Germany. The proposed ID is built from the following components: the size of the grid, then a symbol for the coordinate axis facing north (e.g., *N* for “north”), the first half of the coordinate (e.g., northing), a symbol for the coordinate axis facing east (e.g., *E* for “east”), and the second half of the coordinate (e.g., easting). Furthermore, unnecessary trailing zeros should be deleted. An example is 10kmN579E47. However, this can be altered as done in Germany for grid sizes of, e.g., 250, 500 (Bundesamt für Kartographie und Geodäsie, 2020, pp. 5–6).

In this thesis, the data set used covers South Australia (as explained in section 2.1), which is not covered by the INSPIRE grid, nor could a similar approach for Australia be found. The next best solution was taken, to use a comparable coordinate system i.e. the coordinates were converted to coordinates in meters. The official statistics grids show that the lower-left corner is rounded to the nearest 100 for a 100 meter cell width. Hence, this is done as well. This method could also be classified under point aggregation. However, since the result is a cell identifier and thus an area identifier, the method was subsumed under areal aggregation methods.

### **Implementation of Official Statistics Grid** (short: grid)

For the grid, the unit of the coordinates must be in meters. Assuming that the leftmost edge starts at 0, the grid is defined as multiples of the cell width. One way would be to create a grid and identify in which cell each point falls. A simpler approach is to round the easting and northing coordinates. For example, if a cell width of 1,000 is considered, the coordinates are rounded to the lower 1,000, e.g., for 415555 (easting) and 2630904 (northing), the coordinates of the lower-left corner are 415000 (easting) and 2630000 (northing). For the center coordinates, the cell width divided by two must be added to the easting and northing. The resulting coordinates can then be concatenated into an ID for the cell, which is used as the geocode remaining in the data set.

Both 100 meters and 1,000 meters are considered as the cell width for the grid. For both, the bottom left corner of the grid is at the origin of the coordinate system.

## **3.2. Adjusting Coordinates**

The category *adjusting coordinates* contains several masking methods which scale, rotate, displace, flip, or move points into a random direction by a random distance, some needing additional information.

### **3.2.1. Affine Transformations**

The first paper providing an overview of geomasking methods and comparing them based on predefined characteristics was written by Armstrong et al. (1999). They summarized several geomasking methods under the term *affine transformations*, namely *displacement using translation*, *change of scale*, and *rotation*, which are described in the following.

#### **3.2.1.1. Displacement Using Translation**

The first affine transformation is called *displacement using translation*. The original idea goes back to Aldrich and Krautheim (1995). Given a coordinate system in which

a coordinate is located at  $(x,y)$ , a displacement constant is added to  $x$  and  $y$ . A different constant can be chosen per axis, but it remains the same for all coordinates (Armstrong et al., 1999, p. 502). For example, 0.5 is added to  $x$  and 1 is added to  $y$  (see figure 3.5). This method moves each coordinate to a different location while preserving the distance between coordinates.<sup>13</sup>

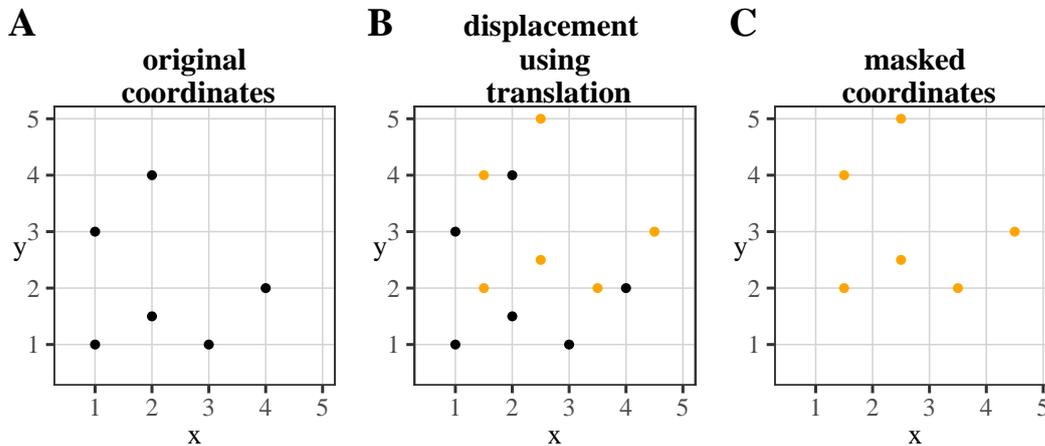


Figure 3.5.: Displacement using Translation: 0.5 is added to  $x$  and 1 is added to  $y$ .

### Implementation of Displacement Using Translation (short: DUT)

With displacement using translation, the main problem to be solved is that latitude and longitude coordinates cannot easily be moved by a certain distance. For example, if 100 is added, the distance between coordinates is not preserved, nor does it correspond to a specific number of meters that is equal for all coordinates.

Therefore, for this masking method, the coordinates are converted into easting and northing, which is in meters, and a certain number of meters can be added to the existing coordinates. The displacement distance is randomly chosen. If the random value is positive, it is added; a negative random value causes subtraction. The masked coordinates are then converted to the original coordinate format of latitude and longitude. This introduces a small additional displacement, especially for systems covering large areas, i.e., when the distance between the original and the masked coordinates is calculated, the distance may not exactly equal the number of meters the coordinates are moved. But the error introduced by converting the coordinates is far smaller than trying to move latitude-longitude coordinates (degrees).<sup>14</sup>

For each iteration, two random numbers of a uniform distribution with the bounds  $[-10,000 \text{ meters}, 10,000 \text{ meters}]$  were added to the coordinate. The first random number is used to move the points on the  $x$ -axis, and the second random number is used to move the points on the  $y$ -axis.

<sup>13</sup>As will be seen, this is only true if coordinates are in meters, to begin with, and not in latitude-longitude form.

<sup>14</sup>This is true whenever the coordinates are converted between coordinate formats and will not be mentioned again.

### 3.2.1.2. Change of Scale

The second affine transformation, *change of scale* is similar to the previous masking method in that it uses an arbitrary constant to change a coordinate. But the coordinates are multiplied by the chosen constant (Armstrong et al., 1999, p. 503). Again, the coordinates are all multiplied by the same constant. While *displacement using translation* can use different constants for the axes, *change of scale* uses the same constant for both parts of a coordinate (Armstrong et al., 1999, p. 503). As shown in figure 3.6, this masking method changes not only the position of the coordinates but also the distances between them.

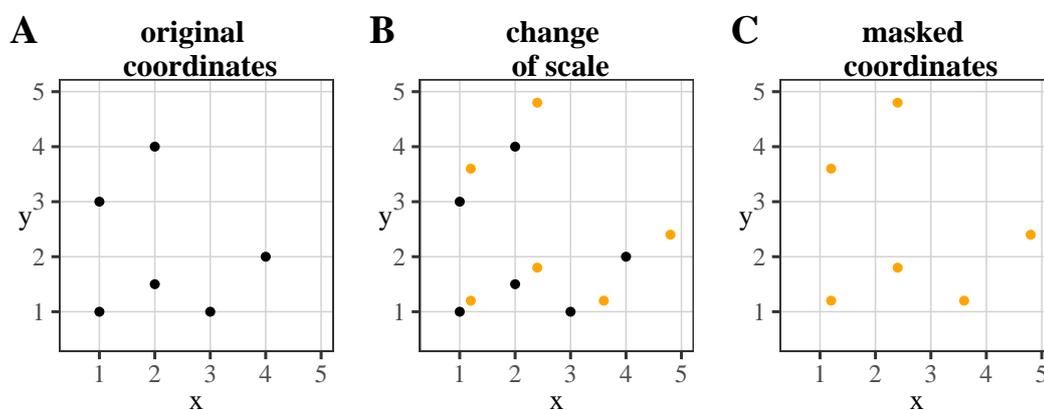


Figure 3.6.: Change of Scale: multiply x and y with 1.2.

### Implementation of Change of Scale (short: CS)

For change of scale, a similar approach as displacement using translation is taken. Instead of adding or subtracting a random number, the random number is multiplied. Although the same constant is used for both axes, the code written allows setting a different value for each axis.

A random number (decimal numbers, rounded to the nearest fifth decimal place) between 0 and 2 was chosen for multiplication. 0 and 2 were chosen because the number had to be reasonably small so that points are not moved unrealistically far from their origin.

### 3.2.1.3. Rotation

Another masking method subsumed under the term affine transformations is called *rotation*. Each point is rotated by a fixed angle  $\theta$  between 0 and  $2\pi$  around a pivot point (see figure 3.7). The pivot point can be either the coordinate system's origin or an arbitrary point (Armstrong et al., 1999, p. 503).

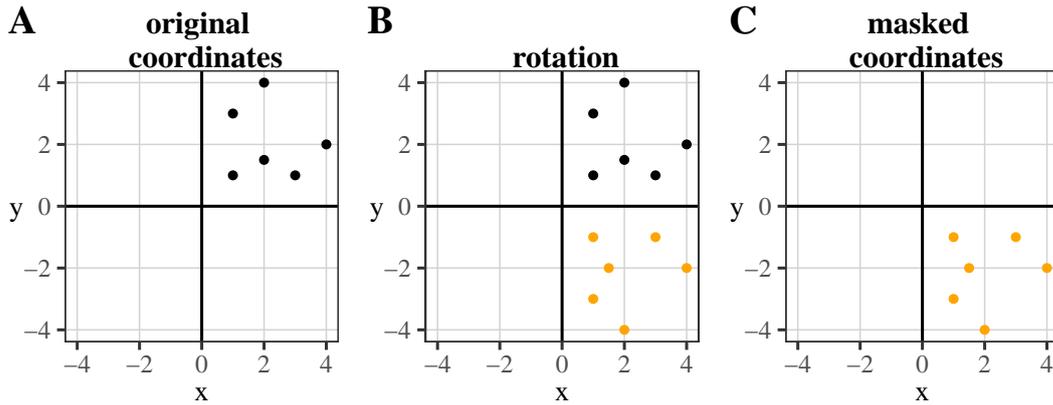


Figure 3.7.: Rotation (Origin of Coordinate System): rotate  $90^\circ$  clockwise.

If an arbitrary point is chosen, the coordinates are first moved so that the arbitrary point matches the origin of the coordinate system (see figure 3.8). After that each point is rotated by a fixed angle  $\theta$  about the origin using the rotation matrix,  $R$ .

$$R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (3.1)$$

The coordinates are then moved again so that the origin of the coordinate system matches the arbitrary point originally chosen (Armstrong et al., 1999, p. 503).

#### Implementation of Rotation (short: Rot and RotArb)

The rotation masking method also uses the converted coordinates. For the use of an arbitrary pivot point, the coordinates are moved, so the origin corresponds to the pivot point. This is done by subtracting the coordinates of the arbitrary point from the original coordinates. To rotate, one can use angles or radians. Both methods lead to the same results. The actual rotation is done by multiplying the coordinates with the rotation matrix. As an example the coordinates  $(x, y)$  are rotated around the spatial mean center  $(\bar{x}, \bar{y})$  using the rotation matrix  $R$ , yielding the masked coordinates  $(x_m, y_m)$

$$\begin{pmatrix} x_m \\ y_m \end{pmatrix} = R \cdot \begin{pmatrix} x - \bar{x} \\ y - \bar{y} \end{pmatrix} + \begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix} \quad (3.2)$$

with  $\theta$  being the angle or radian. Note that this rotates points counterclockwise. The rotation of  $90^\circ$  clockwise is achieved by rotating  $270^\circ$  counterclockwise. Then the coordinates are converted to latitude and longitude coordinates.

Rotation can be realized with the coordinate system's origin as the pivot point (Rot) or an arbitrary point (RotArb). In addition, random values for the angle for the rotation are needed. Therefore, a random angle (integer) of a uniform distribution with the bounds  $[0, 360]$  was chosen to rotate the coordinates around the origin of the

coordinate system.<sup>15</sup> Another random number was drawn as the angle for rotation around an arbitrary point. But this time, the coordinates are rotated around the spatial mean center.

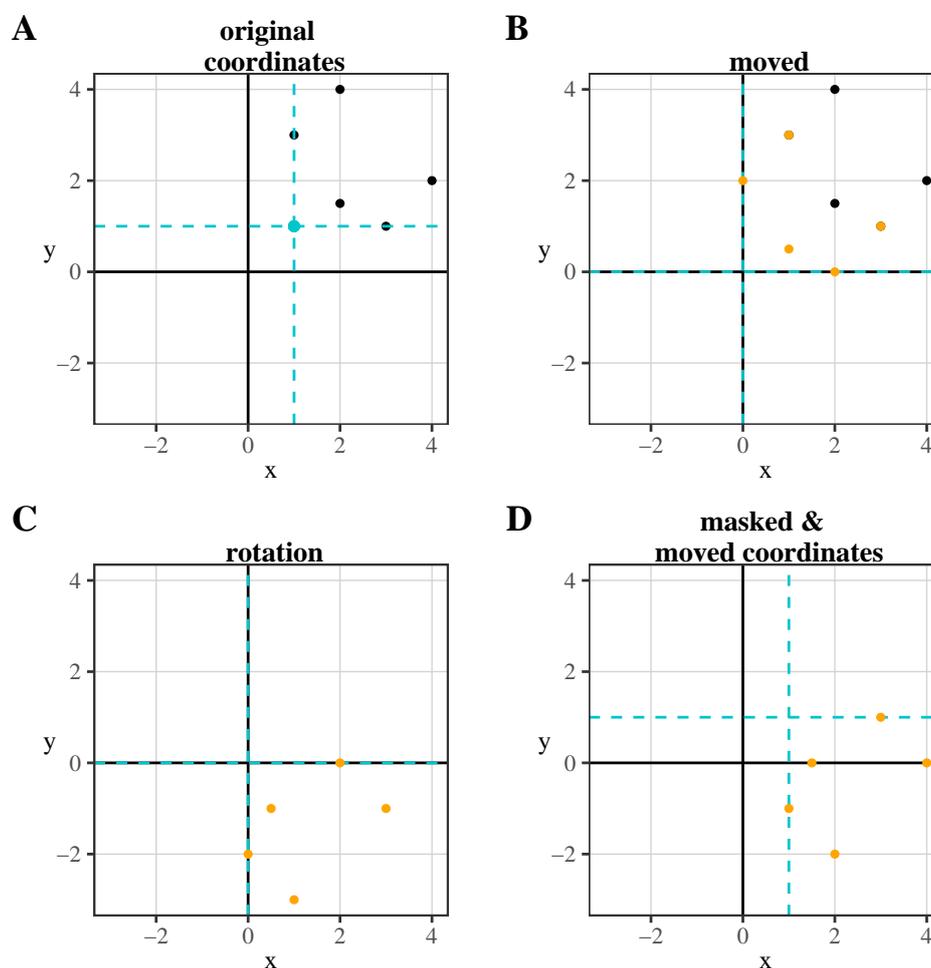


Figure 3.8.: Rotation (Arbitrary Point): rotate 90° clockwise, arbitrary point is at (1,1) indicated by the blue point/lines.

#### 3.2.1.4. Combination of Affine Transformations

Armstrong et al. (1999) also proposed the idea of combining affine transformations, e.g., by first rotating each point and then displacing it by a chosen constant (Armstrong et al., 1999, p. 503). This would make it more difficult for an intruder to identify a coordinate system's original position since, theoretically, it lowers the chance of finding the correct re-identification method. Moreover, instead of choosing the constant or angle needed to apply the mask, the parameter(s) can be set randomly (Armstrong et al., 1999, p. 504).

<sup>15</sup>The origin of the coordinate system, when converted to easting and northing, is the point (0,0). This point is located in the middle of the ocean between South Australia and Antarctica.

### 3.2.2. Grid Masking

In 2004, Leitner and Curtis proposed several methods which they subsumed under the term grid masking. The methods are intended to be used for publishing coordinates on maps. They distinguish between *global geographic masking methods* and *local geographic masking methods*. The former applies the masking method to all points and uses the same parameters for masking for each point (Leitner and Curtis, 2004, p. 24). The latter places a grid over the area, and in each cell, the parameter or masking method used is changed (Leitner and Curtis, 2004, p. 24).

Although the authors point out that there are several masking methods that can be applied globally or locally, they give five examples of each category. In addition, the masking methods require knowledge of street courses. The first global masking method presented (see figure 3.9) flips each point by taking the horizontal central axis as a mirror and moving the data point to the closest street segment. A street segment is the part of the street between the two adjacent street intersections, which is defined as an intersection of three or more streets (Leitner and Curtis, 2004, p. 24).

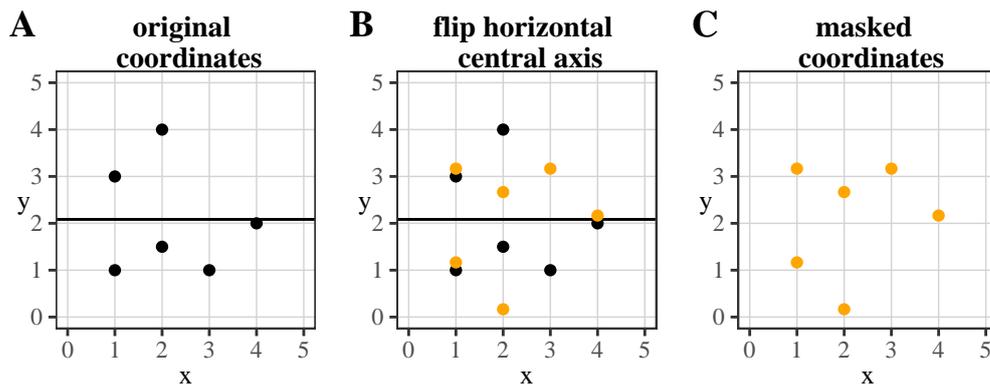


Figure 3.9.: Global grid masking: horizontal flip (central axis).

The second uses the vertical axis as a mirror and then moves the flipped data points on the closest street segment (see figure 3.10). In the third global masking method, each point is flipped about the central horizontal and vertical axis before being moved to the closest street segment (see figure 3.11).

The last two masking methods use rotation.<sup>16</sup> The first rotates the data points by  $60^\circ$  clockwise, where the center of the coordinates is the pivot point, and then, the points are moved to the closest street segment. In the last method, the points are moved  $120^\circ$  counterclockwise and then moved to the closest street segment (Leitner and Curtis, 2004, p. 24).

<sup>16</sup>For a visual example using an arbitrary rotation angle see figure 3.8.

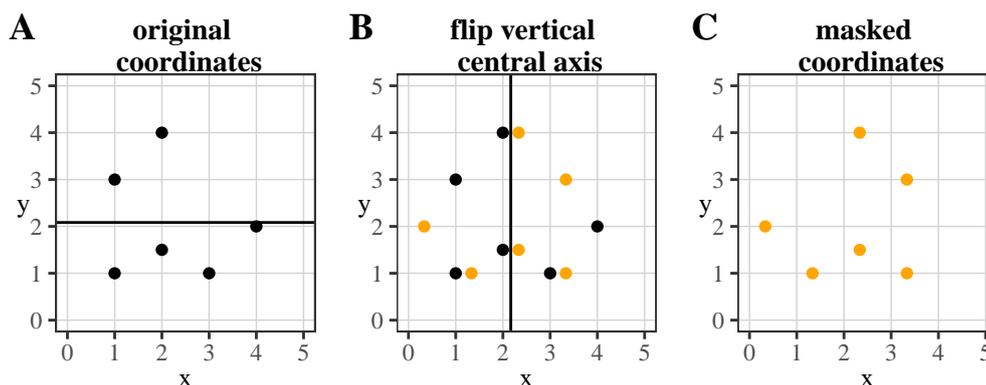


Figure 3.10.: Global grid masking: vertical flip (central axis).

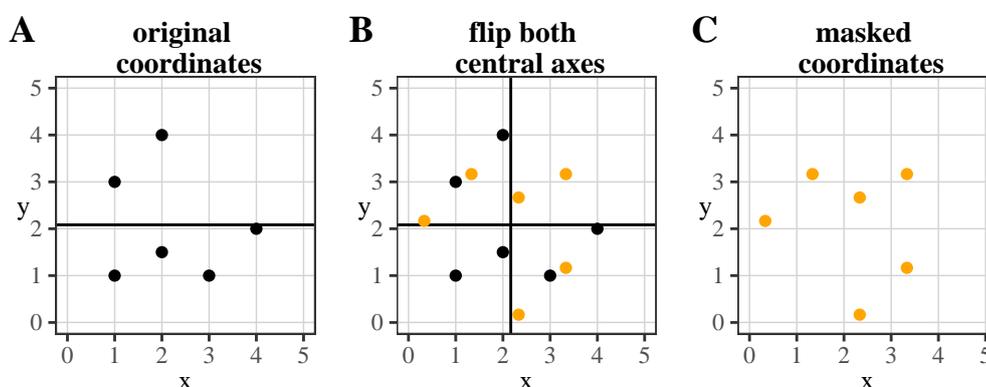


Figure 3.11.: Global grid masking: horizontal and vertical flip (central axis).

The first of the five local masking methods (see figure 3.12) aggregates the data points at the midpoint of their respective street segments (Leitner and Curtis, 2004, p. 24). The second local masking method aggregates the data points to the closest street intersection. The third uses the flipping methods described above (vertical, horizontal, both) and varies between them in each cell. The fourth uses the rotation methods described above, but randomly varies the direction and degree of rotation in each cell. In the last local masking method, the data points in each cell are moved a random distance and then moved to the closest street segment (Leitner and Curtis, 2004, p. 24; see also *random perturbation* in chapter 3.2.3). In their example, the authors used a 500 meter grid, and a different method or parameter choice was used for each cell.

In addition, the authors analyzed their masking methods by presenting the maps of coordinates to students who were asked to identify areas with a high concentration of points. The students were also asked to compare masked and original maps in terms of areas with a high density of points (Leitner and Curtis, 2004, p. 27). The authors concluded that all masking methods except aggregating points to the midpoint of their street segment and aggregating points to the nearest street intersection should

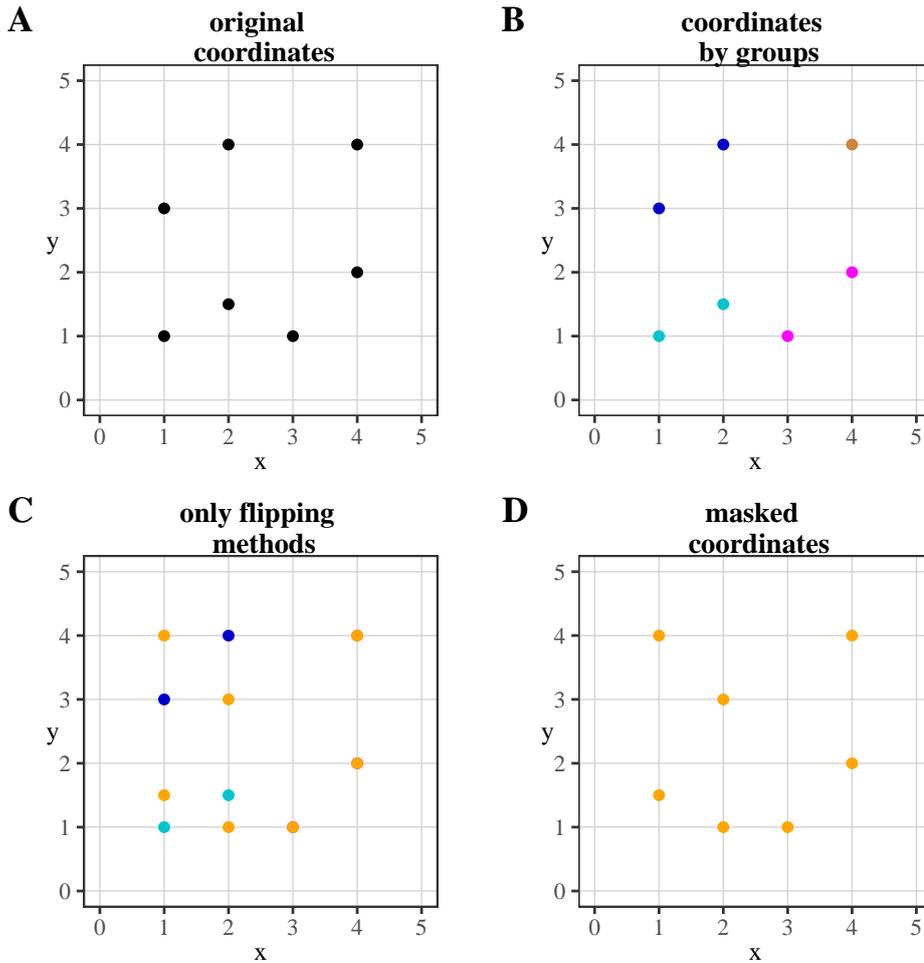


Figure 3.12.: Local grid masking: Blue points are flipped vertically (central axis). Turquoise points are flipped horizontally. Purple and brown points are flipped horizontally and vertically. Orange points are masked points.

not be used (Leitner and Curtis, 2004, p. 37). In addition, if there are fewer than seven points (including the location to be masked) on either side of the street segment, the points should be aggregated to the nearest street intersection rather than to the midpoint of the road segment (Leitner and Curtis, 2004, p. 37).

None of the proposed displacement methods are new, except for the placement on the nearest street segment and the possibility of using different methods for different areas. The authors conclude that most of their proposed methods are unsuitable (Leitner and Curtis, 2004, p. 30). In addition, a map of streets is not always available or requires long calculation times and computing power, especially if the area of interest is large.<sup>17</sup>

Since the authors themselves state that most variants (e.g. flipping horizontally, flipping vertically) of this masking method are unsuitable and this masking method

<sup>17</sup>Unlike the other method that uses roads (called street masking), no more specification is given on the map, i.e., whether all roads or only drivable roads are included.

is designed for publishing maps and not for releasing the coordinates (Leitner and Curtis, 2004; Leitner and Curtis, 2006), this method is not tested in this thesis.

### 3.2.3. Random Perturbation

In *random perturbation* the coordinates are displaced in a random direction by a random distance. This is done by adding a random number to each part of the coordinate. For each coordinate new random numbers are drawn (Armstrong et al., 1999, p. 504). The distributions considered by Armstrong et al. (1999) for the random numbers are the bivariate uniform and the bivariate normal distribution. When the bivariate uniform distribution is chosen, a maximum limit is set for the displacement. The bivariate normal distribution can sometimes move the coordinates far away from the original location (Armstrong et al., 1999, p. 505). However, for the random numbers in random perturbation, any distribution can be chosen.

Figures 3.13 and 3.14 show examples of masking coordinates by random perturbation using uniform and normal distribution. As can be seen, for the normal distribution, there is the possibility of points being far displaced (see, e.g., point  $x = 3, y = 1$ ). Additionally, figure 3.15 shows a random perturbation within a circle where the coordinate is randomly displaced within the boundaries of the circle, as also proposed by Armstrong et al. (1999).

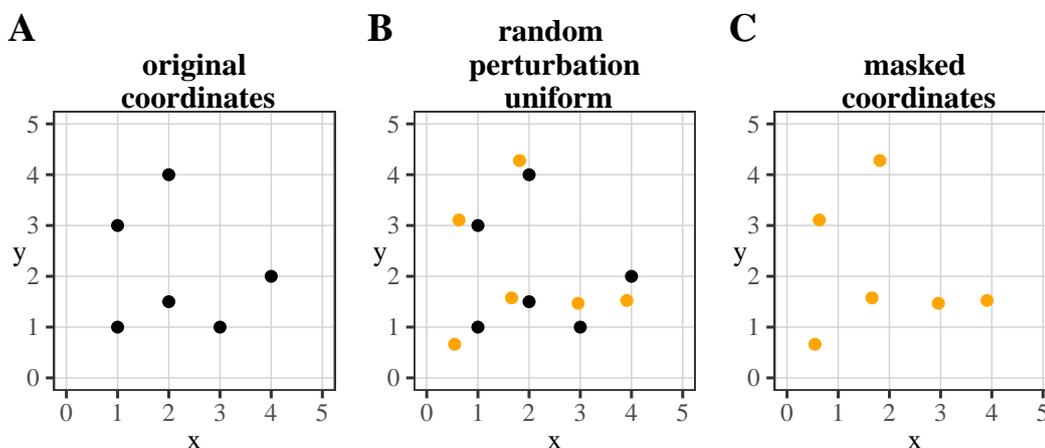


Figure 3.13.: Random Perturbation (uniform distribution): intervall  $[-0.5, 0.5]$ .

In the geomasking literature, also the name *weighted random perturbation* can be found (Allshouse et al., 2010). This states that the population density determines the parameter choice (size of the range or the variance) for the chosen distribution.<sup>18</sup> If the underlying population density is high, the variance or range is chosen to be small. If the population density is low, the variance or range is larger (Armstrong et al.,

<sup>18</sup>The need for a variable maximum displacement distance by point was also proposed by Lu et al. (2012). They also emphasized that larger displacement distances will result in large noise for spatial analysis (Lu et al., 2012, p. 176).

1999, p. 505). Stinchcomb (2004) proposed using the average distance between people given by  $\sqrt{1/\text{PopDensity}}$  and multiplying the value by 1, 2, 3, 4, or 5 as parameter choice for the chosen distribution (e.g. for the range, variance, etc.).

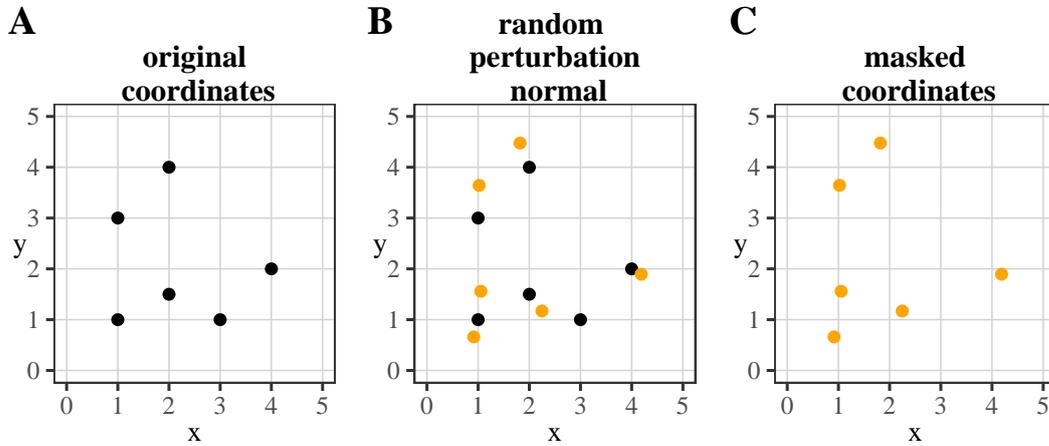


Figure 3.14.: Random Perturbation (normal distribution): mean = 0, sd = 0.5.

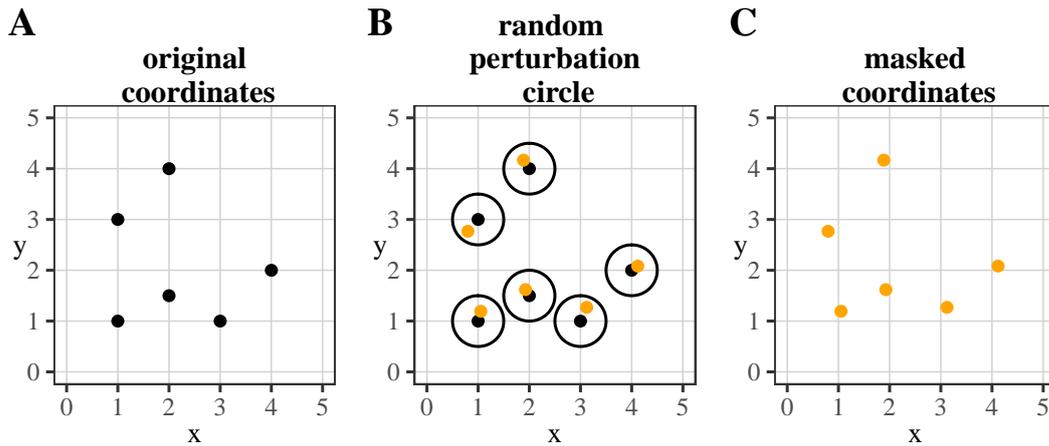


Figure 3.15.: Random Perturbation (within a circle): maximum radius 0.5.

In addition, the names *Gaussian displacement* (Zandbergen, 2014) and *Population-density-based Gaussian spatial blurring* (Kounadi and Leitner, 2015) can be found, whose description is the same as for the random perturbation using a normal distribution. Alternatively, a bimodal Gaussian distribution can be used (Gupta and Rao, 2020). More recently, Gao et al. (2019) named *Gaussian perturbation* as a masking method in which the distribution of the masked points follow a two-dimensional Gaussian distribution. However, the papers cited as the references for this (namely Zandbergen, 2014 and Fronterré, 2018) refer to the fact that the displacement distance must follow a Gaussian distribution, not the masked points.

**Implementation of Random Perturbation: Uniform distribution** (short: RPU)

For the random perturbation using a uniform distribution, the coordinates in easting and northing format were used. For each coordinate considered separately, an independent random number within the bounds of the uniform distribution is chosen which is the displacement distance. Then, a second random number between 1 and 10 is drawn. If the second random number is below 5, the first random number of the uniform distribution is subtracted from the coordinate, and if it is above five, the first random number of the uniform distribution is added to the coordinate. This ensures that the coordinates can move up and down as well as left and right.

The masked coordinates are then converted back to latitude and longitude. Ensuring that coordinates can be displaced in all directions can also be solved by considering a range that includes numbers below zero. However, if the probability of adding a number or subtracting a number should be equal, the range would always have to contain zero as the midpoint, and a minimum displacement distance cannot be guaranteed.

For the uniform distribution, the bounds were chosen based on the underlying population density. Stinchcomb (2004) provided a formula to set the minimum and maximum distances using an estimation of the average distance between people based on the population density. The population density is calculated by dividing the population by the area in square kilometers.<sup>19</sup> Then the estimate of the average distance between people is  $\sqrt{1/\text{PopDensity}}$ . Stinchcomb then proposed multiplying the estimate of the average distance between people based on the population density by a number between one and five. The lower bound was always set to the estimate of the average distance between people (Stinchcomb, 2004) multiplied by 2. The upper bound was set to the estimate of the average distance between people multiplied by 3, 4, and 5. The population density based on postcode areas and based on local government areas (LGA) was used, to show differences between different area levels used. As the population, the number of people based on the census 2016 was used.<sup>20</sup>

**Implementation of Random Perturbation: Normal distribution** (short: RPN)

Again, for this variant, coordinates are needed in easting and northing format. As with uniform distribution, a random number is drawn, but this time from a normal distribution with zero as the mean. For the standard deviation, any number can be chosen, e.g., based on the population density. For the easting and northing, different random numbers are drawn. Depending on the random number, the coordinates are moved up or down as well as left or right.

---

<sup>19</sup>In the original work, square miles are used, but this is not feasible for countries using the metric system.

<sup>20</sup>Another idea is to use the number of addresses as the basis for the population. This will increase the displacement distances. However, this did not influence the risk results considerably (see subsection 5.3.4).

Random perturbation based on a normal distribution was always carried out with a mean of zero. The standard deviation was set to the estimate of the average distance between people (Stinchcomb, 2004) without multiplying it since, unlike other masking methods, it is not used as an upper limit. For each coordinate part, a random number was drawn from this distribution. If both random numbers were zero new random numbers were drawn to ensure that no point remains at their unmasked location. Again, the population density per postcode and local government area was used.

#### **Implementation of Random Perturbation: Within a Circle** (short: RPC)

For the random perturbation within a circle, an angle (or radian) is needed to define the direction, and a random number below a predefined threshold (the radius of the circle) is required. The latter random number is the distance a point is moved. Using trigonometry, the masked point can be found by multiplying the random number (*rand*) by the cosine (for the easting) or sine (for the northing) of the angle ( $\theta$ ). Then this is added to the original coordinate.

$$\begin{pmatrix} x_m \\ y_m \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + rand \cdot \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix} \quad (3.3)$$

This way, the point on a circle with a smaller radius than the maximum radius is found. The masked coordinates  $(x_m, y_m)$  are then converted to latitude and longitude.<sup>21</sup>

Postcode areas and local government areas are used for the population density as described above. Then, the result of the formula by Stinchcomb (2004) is multiplied by 3, 4, or 5.

#### **3.2.4. Triangular Displacement**

More recently, Murad et al. (2014) proposed a geomasking method that did not just use the population density or a random value to define the distance each point is moved. Instead, they used multiple criteria.<sup>22</sup> This masking method consists of a displacement distance algorithm and a triangular displacement algorithm (Murad et al., 2014, p. 5).

The displacement distance algorithm is used to determine the needed minimum distance and maximum distance by which a point must be displaced. First, the risk

<sup>21</sup>Zandbergen (2014) points out that alternatively, a random point inside the circle could be drawn and used as the new location. Since this is more difficult to implement, a random angle and random displacement distance are preferred. This approach, of drawing a sample from all possible points of the area considered for displacement, could also be used for donut masking and is used for adaptive random perturbation.

<sup>22</sup>It should be noted that until now, no application of the masking method has yet been found, possibly due to the various ambiguities regarding the implementation of this method, as shown in the following. The first author (corresponding author) of the paper was contacted via e-mail to clarify questions. However, he did not reply.

factor is evaluated. If the risk factor to identify a respondent is below 1:20 within a one-mile radius (probability of less than 0.05), the person is considered as “cannot be identified” (Murad et al., 2014, p. 5). It is not specified what will be done with the coordinates of these respondents.

For the respondents that are not considered as “cannot be identified”, the sensitivity of the data is evaluated. If it is considered low, the displacement distance ( $dd$ )<sup>23</sup> is set to 300 m. If the sensitivity is considered as high, the distance is set to 500 m (Murad et al., 2014, pp. 4–5). No criteria are mentioned for evaluating the sensitivity of the data, making this a subjective decision of the data holder. There is also no explanation of why 300 meters or 500 meters is chosen.<sup>24</sup>

Next, the complexity of the data analysis to be performed is evaluated. When considered advanced, 100 m is added to the displacement distance. If only traditional analyses are performed, the displacement distance remains the same (Murad et al., 2014, pp. 4–5). Again, no criteria can be found for which analyses are considered traditional or advanced, so this criterion is also somewhat subjective.

The next category is the availability of quasi-indicators. If such are available, the distance is increased by 100 m, and the indicators must be categorized. If quasi-indicators are not available, the displacement distance remains the same (Murad et al., 2014, pp. 4–5). Then, the distance is increased by 100 m if public media reported about the data collection. If not, the displacement distance remains the same (Murad et al., 2014, pp. 4–5). Again, no definition of public media is provided. Lastly, the end-users of the data set must be considered. If the end-user is the general public, the displacement distance is increased by 100 m (Murad et al., 2014, pp. 4–5). If the end-user type is “other” (researchers are also categorized as “other”) the displacement distance remains the same (Murad et al., 2014, pp. 4–5). Using these categories, the final displacement distance is between 300 and 900 m. This distance is used in the second algorithm (triangular displacement algorithm) as the lower and upper bound (200 m added) for a random variable,  $r$ , which is used for the actual displacement.

After the displacement distance algorithm, the triangular displacement algorithm is needed for the actual displacement of the coordinates. This algorithm is based on the Pythagorean equation  $x^2 + y^2 = r^2$ , where  $x$  and  $y$  are the so-called offsets (displacement distances) for the coordinates  $(x,y)$  and  $r$  is the result of the displacement distance algorithm, ranging from  $dd$ , as lower bound, to  $dd + 200$  as upper bound (Murad et al., 2014, p. 5).

Using the Pythagorean equation as a starting point, a random number  $r^2$  is drawn from the interval  $[dd, dd + 200]$ . Then a random number is drawn from the interval  $[1, r^2 - 1]$  as the value for  $x^2$ . Based on the Pythagorean equation,  $y^2$  is calculated by subtracting  $x^2$  from  $r^2$  (Murad et al., 2014, p. 5). The offsets are then calculated as

<sup>23</sup>The original paper uses the abbreviation  $DD$ . However, this abbreviation will be used for a masking method. Therefore,  $dd$  is used here.

<sup>24</sup>This applies to the entire masking method for all displacement distances and is not mentioned again.

(Murad et al., 2014, p. 5):

$$\begin{aligned} xOffset &= \sqrt{x} \\ yOffset &= \sqrt{y} \end{aligned} \tag{3.4}$$

It is unclear whether the square root is taken twice since the input according to the formula is  $x$  and  $y$ , and up to this point,  $x^2$  and  $y^2$  are given.

Another step in the triangular displacement algorithm is to evaluate whether the calculated number is added or subtracted from the original coordinate (Murad et al., 2014, p. 5). For  $x$ , a random number from the interval  $[dd, dd + 200]$  is divided by two. If the modulo is zero,  $xOffset$  remains positive. If it is not zero, it is set negative, so the  $yOffset$  value is subtracted from the original coordinate. For  $y$ , another random number is drawn, but this time from the interval  $[1, 10]$ , and divided by two. As with  $x$  the modulo decides whether the  $yOffset$  value is added or subtracted (Murad et al., 2014, p. 5). There seems to be no reason why an interval between one and ten is used for  $y$ . Lastly, the new coordinates are calculated by adding or subtracting (depending on the sign of the values) to the original coordinates (Murad et al., 2014, p. 5). It is uncertain that  $xOffset$  and  $yOffset$  vary between the coordinates of the data set.

In summary, triangular displacement is nothing more than a displacement using translation or a random perturbation. Since many questions remain unanswered and no applications could be found to help answer these questions, this masking method will not be considered in the later analysis.

### 3.2.5. Donut Geomasking

The *donut geomasking method* was originally presented by Stinchcomb (2004) at the ESRI International Health GIS Conference. It was not until 2010 that Hampton et al. explained this method in detail and completed it with an analysis of its efficiency.

Donut geomasking is similar to random perturbation within a circle (see figures 3.16 and 3.17). A circle is placed around the considered location, indicating the maximum distance a point should be moved. Additionally, another circle is placed around the location where the radius is a fraction of the radius of the first circle (Stinchcomb, 2004, pp. 10–11). Within these two circles, which form a “donut” around the coordinate, the point is moved a random distance in a random direction (Hampton et al., 2010, p. 1064; Stinchcomb, 2004).

The minimum and maximum distance indicated by the circles is inversely proportional to the population density given in the area or is defined as the distance to the nearest  $k$  residential point, where two values are set for  $k$  to find the minimum and maximum displacement distance. For example, the distance to the fifth nearest neighbor is set as the minimum distance and the distance to the 50th nearest neighbor as the maximum distance. Again, by taking population density into account, points in an area with high population density are not displaced as far as points in an area with

low population density (Stinchcomb, 2004, pp. 10–11). In the following, these two variants for choosing the minimum and maximum distance are called donut masking (using population density) and  $k$ -nearest neighbor donut masking.

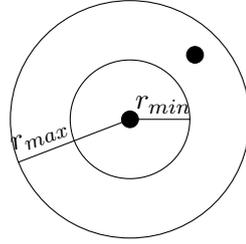


Figure 3.16.: Donut Geomasking (Hampton et al., 2010, p. 1063).

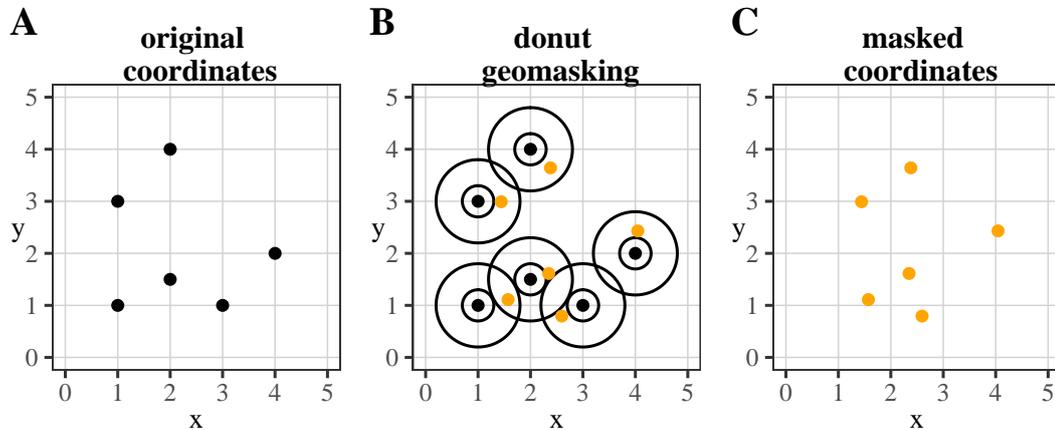


Figure 3.17.: Donut Geomasking: maximum radius 0.8, minimum radius 0.3.

### Implementation of Donut Masking (short: DD, Dk, and DkData)

For donut masking (DD), the same procedure as for random perturbation within a circle is performed with the difference that not only an upper limit but also a lower limit is set for the random number. In  $k$ -nearest neighbor donut masking, an additional function determines the minimum and maximum distance. The additional function calculates the distance of the coordinates to all residential coordinates and takes the distance of the nearest  $k$  residential coordinate as minimum/maximum distance. The residential population was used because it is explicitly referred to in the article (Hampton et al., 2010, p. 1064). This could be modified to find the  $k$ -nearest point in the data set, which might be a better option if relatively few points are selected from a large population. Thus, this was also done in this thesis.

As already explained for the random perturbation methods, Stinchcomb (2004) provided a formula to set the minimum and maximum distance using an estimation of the average distance between people based on the population density, which will be used for donut masking (DD). The minimum radius was always kept at the population

density multiplied by 2 to limit the number of parameter choices to be tested. The population density times 3, 4, and 5 were used for the maximum radius. The population density based on postcode areas and based on local government areas (LGA) was used, to show differences between different area levels used. As the population, the count of people based on the census 2016 was used.<sup>25</sup>

For  $k$ -nearest neighbor donut masking (Dk), a minimum  $k$ -nearest neighbor and a maximum  $k$ -nearest neighbor has to be chosen. For the maximum  $k$ -nearest neighbor, the numbers 5, 25, 50, 100, 500, and 1,000 were chosen. For the minimum  $k$ -nearest neighbor, 10% of the maximum was chosen. However, for 25 and 5, the minimum number was set to 2 other points. Note that  $k$ -nearest in the present case calculates the  $k$ -nearest address of the full data set for South Australia<sup>26</sup> and not the  $k$ -nearest person since there is no information on how many residents live at the address. Another tested idea was to set the maximum number to 20 and the minimum number to 2, but considering the neighbors of the given data set and not the resident file (DkData).<sup>27</sup>

### 3.2.6. Voronoi Masking

*Voronoi masking* is an idea introduced by Seidl, Paulus, et al. (2015). It is a masking method that differs from the previous ones in that it does not use population density or moves points into random directions. Voronoi masking uses Voronoi polygons (also called Thiessen-polygons) to change coordinates (Seidl, Paulus, et al., 2015, p. 256).

Thiessen-polygons are placed over the area of interest, with the data points being at the center. Thiessen-polygons are constructed by setting the borders so that every point within the border of a polygon is closer to the center (point of the data set) than to any other point of the data set in the area (Haggett et al., 1977, p. 437). Haggett et al. (1977, p. 436) propose drawing lines between each point and their adjacent points and to use the midpoints of these lines to draw orthogonal lines as borders for the polygons (see figure 3.18). This method can leave some of the lines to the adjacent points unused (Haggett et al., 1977, pp. 436–437).

Kopec (1963) proposed an alternative method where circles are drawn around each point, with the radius being the distance to the nearest data point. A line is then drawn through the intersection points of the two circles as one border of the polygon (Kopec, 1963, p. 25). Another alternative is to use Delaunay triangulation, which is, along with other methods, described in more detail in Aurenhammer (1991).

<sup>25</sup>Another idea is to use the number of addresses as the basis for the population. This will increase the displacement distances. However, this did not influence the risk results considerably (see subsection 5.3.4).

<sup>26</sup>The sampled coordinates of the data set were excluded from the residential coordinates. This is especially important for other masking methods that use the residential file (verified neighbor approach and location swapping) to ensure that a coordinate of the data set is not an eligible masked coordinate. Furthermore, the residential file should be the same for all geomasking methods using a residential file.

<sup>27</sup>This idea is only applicable for reasonably large data sets with a certain density of the points. For the given example of  $n = 10,000$ , 20 and 2 were reasonable.

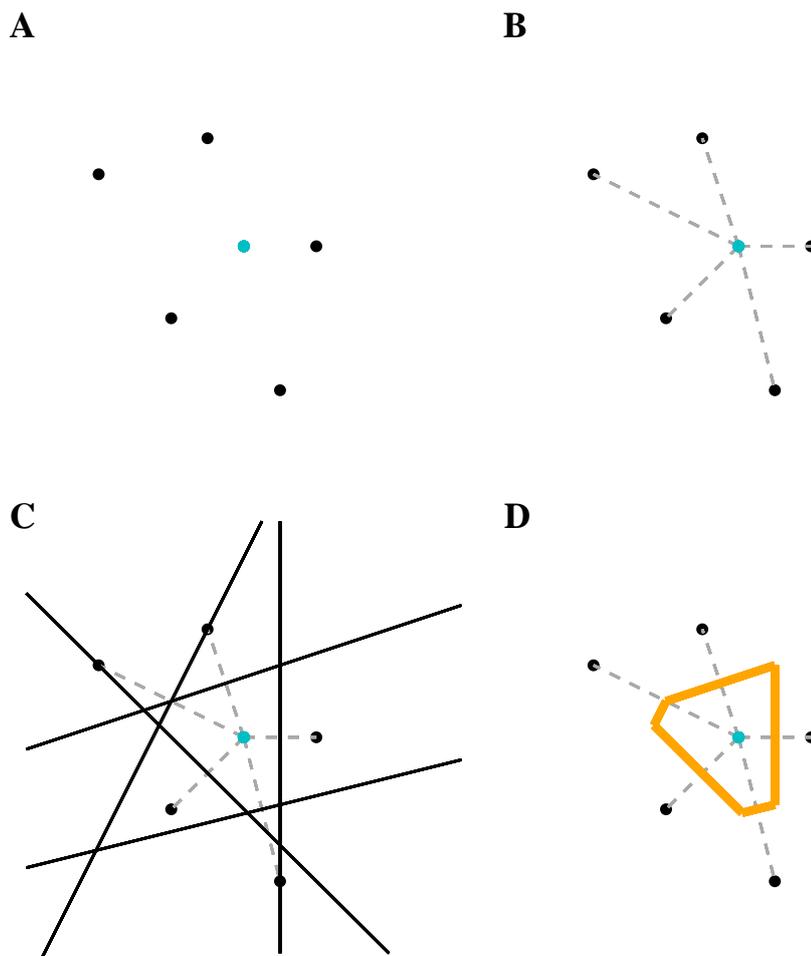


Figure 3.18.: Example of construction of Thiessen-polygon as described by Haggett et al. (1977, p. 436). Blue point is point for which Thiessen-polygon is constructed. Orange lines show borders of Thiessen-polygon.

In locations with many data points, small polygons are constructed, while in areas with fewer points, large polygons are created. After the Thiessen-polygons are constructed, the data points are moved to the closest point on the nearest border of their polygon (Seidl, Paulus, et al., 2015, p. 256). Potentially, points can be moved to the same target point. Then the Thiessen-polygons are removed, leaving only the masked coordinates (Seidl, Paulus, et al., 2015, p. 256). An example is shown in figure 3.19.

With this masking method, points closely surrounded by other points, thus resulting in smaller Thiessen-polygons, are moved less than points with distant neighboring points resulting in larger Thiessen-polygons (Seidl, Paulus, et al., 2015, p. 256).

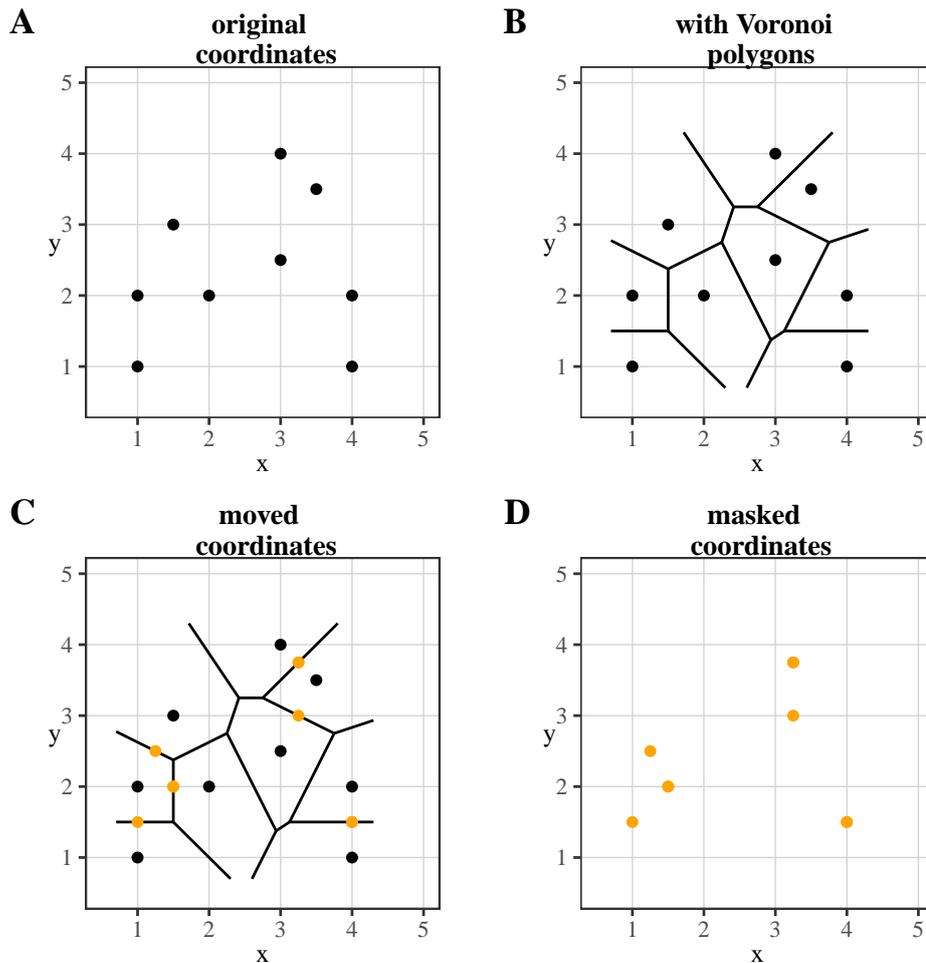


Figure 3.19.: Voronoi masking. Points can be moved to the same location, e.g. points (3,4) and (3.5,3.5) are moved to the same location (3.25,3.75).

Gupta and Rao (2020) proposed a method very similar to Voronoi masking called *three layer RDV masking*. The three layers are abbreviated as *RDV*. In layer *R*, the points are randomly distributed over the region of interest. It is not explained whether constraints must be placed on how far points are moved or whether the points are actually distributed randomly over the region, thus making it questionable what the layers *D* and *V* are for.<sup>28</sup> In layer *D*, Delaunay triangles are placed over the data. Delaunay triangles are formed by drawing triangles so that three points are the corners of the triangle, and no other point lies within the triangle (see, e.g., Lee and Lin, 1986). Using these triangles, Voronoi polygons are created in the third layer (layer *V*). Depending on the region size, for large regions, points are moved to the nearest segment of the edges of the polygon (just like Voronoi masking). For moderate/small regions, the center of the Voronoi polygon is taken as the masked point. There is no explanation why a distinction is made between large, moderate, and small areas.

<sup>28</sup>Therefore, the first author (corresponding author) was contacted by e-mail. The corresponding author did not reply.

This method can be extended by using *nested RDV*. An iterative indicator ( $I$ ) is used, with  $1 \leq I \leq n$  and  $n$  the maximum level.<sup>29</sup> Depending on the value of  $I$ , layers  $D$  and  $V$  are repeated  $I$  times. If  $I = n$ , it is defined that the coordinates are aggregated to one point by taking the average. The authors state that the value of the iterative indicator can be determined by so-called index factors (Gupta and Rao, 2020, p. 190): the population density, the type of investigation (type of analysis done), statistics responsiveness (“seriousness of the data”), degree of quasi-identifiers existence, target user (researchers, general public). No further guidelines are given on how to set the value of  $I$ .<sup>30</sup> Unless masking is done for a specific project where information such as the type of analysis and the type of data is known in advance, intuitively, the highest iterative indicator would be considered, i.e., the points would be aggregated into a single point. However, this would not be considered a geomasking method because the remaining geographic information is mostly useless.<sup>31</sup>

Due to the fact that questions remain unanswered that are essential for the implementation of this variant of Voronoi masking, this method was not implemented in this thesis. Theoretically, without the layer  $R$  and the fact that the data set used in this thesis is considered large, the results would correspond to Voronoi masking. With nested RDV, as mentioned before, it should always be assumed that the data set, for which complex analyses are likely, contains sensitive information with many quasi-identifiers. Therefore, the maximum value of  $I$  would be considered, i.e., aggregation of the data to the spatial mean center.

### Implementation of Voronoi Masking (short: Voronoi)

In Voronoi masking, coordinates can remain as latitude-longitude coordinates. Thiessen-polygons are created using the *deldir*-function (Turner, 2019) in  $R$ . The smallest distance between a point and the surrounding lines is found and gives the masked coordinate. No input information is required for Voronoi masking.

### 3.2.7. Location Swapping

A similar geomasking method to *random perturbation* was published by Zhang, Freundsuh, et al. (2015) and is called *location swapping*. Until now, each geomasking method changed the coordinate without considering the target point of the displacement. Therefore, a coordinate can be moved, e.g., into a sea, an ocean, a park. To prevent this, a circle is first placed around each coordinate (see figure 3.20) whose radius is inversely proportional to the local population density (Zhang, Freundsuh, et al., 2015, p. 3). If the population density is high, a small radius is sufficient. If the population density is low, a larger radius is needed.

<sup>29</sup>According to the code provided in the article,  $n \in N$  with  $N$  the number of points (Gupta and Rao, 2020, p. 192).

<sup>30</sup>This was also asked when contacting the author.

<sup>31</sup>For example, the distance between the points is zero.

Each point is then moved to another residential address within the boundaries of the circle (Zhang, Freundschuh, et al., 2015, p. 3). It is presumed that the potential new residential addresses have similar geographic characteristics. It should be noted that it is not checked whether the radius contains a minimum number of residents. Thus, if there is only one other resident, the other resident's geographic coordinate is always used.

Location swapping is similar to random perturbation in that it modifies the radius, respectively, the maximum displacement distance according to the underlying population density. Also, each potential target location has the same probability of being randomly drawn (Zhang, Freundschuh, et al., 2015, p. 3). The difference between the two masking methods is that in location swapping, only residents are considered as target locations (Zhang, Freundschuh, et al., 2015, p. 3). In addition, this method assumes that all residents in the area are known.

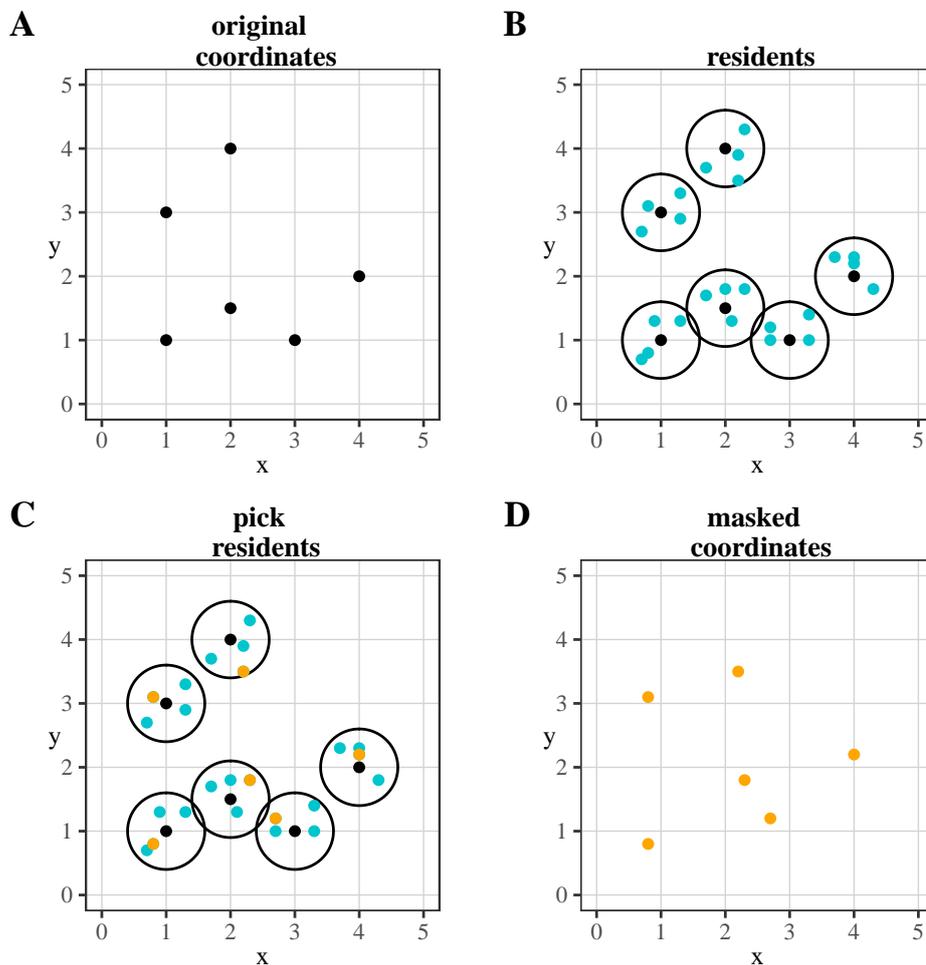


Figure 3.20.: Location swapping: maximum radius 0.6 (not varied for the example).

Another variant of this geomasking method is called *location swapping with donut* (see figure 3.21). While regular location swapping only sets a maximum radius, the donut variant of this method also sets a minimum radius (Zhang, Freundschuh, et al.,

2015, p. 3).

As in donut geomasking, only the points within these two circles are considered as target locations. Therefore, in location swapping with donut, only the residential addresses between the two boundaries are considered as possible target locations. The authors suggest using half the radius of the maximum radius (Zhang, Freundsuh, et al., 2015, p. 3).

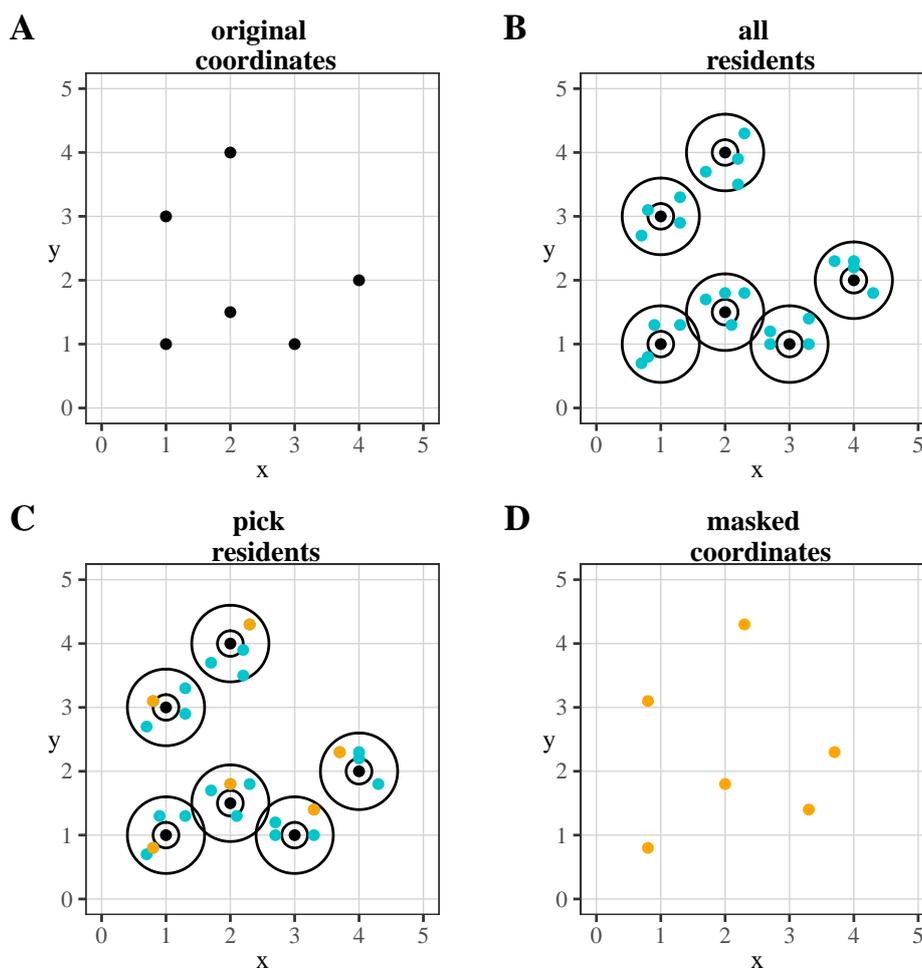


Figure 3.21.: Location swapping with donut: maximum radius 0.6, minimum radius 0.2 (not varied for the example).

### Implementation of Location Swapping (short: LS, LSdonut)

First, the distance to the residents is calculated for each point. In the following step, only points from the residential file are considered whose distance is smaller than the radius defined for the point. From these, a point is randomly drawn as the masked coordinate. If population density is used to set the radius, locations in rural areas may not have a neighbor within the distance based on the population density. In this case, the nearest point is used. In the original paper (Zhang, Freundsuh, et al., 2015) no solution to such a problem is given. For location swapping with donut, an additional

minimum radius is defined as half the radius of the maximum radius, as suggested by Zhang, Freundsuh, et al. (2015, p. 3).

For location swapping (LS), the population density is needed. As stated above, the formula of Stinchcomb (2004) is used and multiplied by 3, 4, or 5. The population density was calculated at the postcode level and local government area level. For the donut variant of location swapping (LSdonut), the minimum radius was set to half of the maximum radius, as suggested by Zhang, Freundsuh, et al. (2015, p. 3).

### 3.2.8. Verified Neighbor Approach

A recently published geomasking method was presented by Richter (2017) and is called *verified neighbor approach* (also: verified neighbor method). This geomasking method is similar to location swapping, but it considers more information in order to find more similar potential target locations.

To apply this masking method, first, a minimum number of surrogate locations  $k$  must be specified as well as a maximum distance (Richter, 2017, p. 3). In addition, a publicly available file containing information on the residential status and other variables of interest is necessary (Richter, 2017, pp. 2–3). If available, polygons of administrative or environmental variables can also be used. In the second step, residential addresses within the defined maximum distance are evaluated using available external information to find similar residents to the original point (Richter, 2017, p. 3). These are considered as the “neighbors”. In the third step, one of the “neighbor” residents is randomly drawn (see figure 3.22). It must be ensured that at least  $k$  residents are left to select from in the third step (Richter, 2017, p. 3).

This geomasking method requires some decisions: the minimum number of surrogate locations must be defined, as well as the maximum distance of surrogate locations. In addition, the variables of interests used to decide whether another point is similar or not must be carefully selected (Richter, 2017, pp. 2–3) and must be known and available for all residents.

#### **Implementation of Verified Neighbor Approach** (short: VNS and VNE)

The method is implemented similarly to the location swapping method, but for this, instead of considering all points within the radius, the points with the same characteristic of a given variable within the radius are considered. In case there are not at least  $k$  people within the radius, the radius is widened to contain  $k$  people with the same characteristic, out of which one is randomly drawn, and the coordinates are used as masked coordinates.

As with location swapping, the estimate of the average distance between people (Stinchcomb, 2004) is used using the population per postcode and population per local government area to calculate population density. This is again multiplied by 3 or 5. The number 4 was skipped to reduce the number of tested parameter choices as it

takes comparatively long to implement. The variables used to identify residents with the same characteristics were the variable sex (VNS) and the variable employment status (VNE). These variables were chosen because sex contains the minimum number of categories and the variable employment status consists of five categories, which is an example of a moderate number of categories. Thus, the influence of the number of categories of the variable can be seen. As minimum number of residents with the same characteristics  $k = 50$  and  $k = 100$  were used.

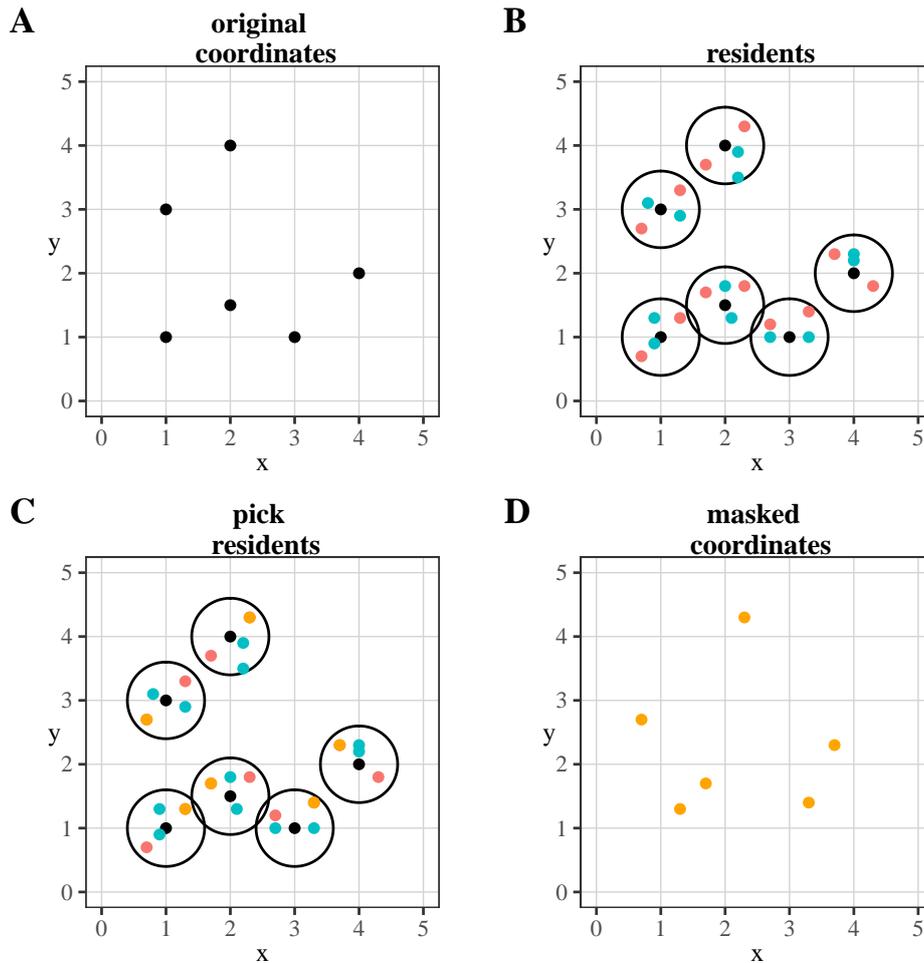


Figure 3.22.: Verified Neighbor Approach. Red: with same characteristics; Blue: different; Orange: sampled; maximum radius 0.6.

### 3.2.9. Street Masking

Swanlund, Schuurman, Zandbergen, et al. (2020) criticized that most masking methods require population density, which may not always be readily available for researchers. Therefore, they proposed using road density, assuming that areas with a high population density also have high road density. Areas with low population density are assumed to have low road density (Swanlund, Schuurman, Zandbergen, et al., 2020, pp. 4–5). In addition, the reason for using roads is to avoid moving locations to

a different residential address, potentially suggesting that the resident of the new address is actually associated with the information in the data set, which was also criticized by Seidl, Jankowski, and Clarke (2018).

The masking is done in several steps (Swanlund, Schuurman, Zandbergen, et al., 2020, pp. 3–5). Using the OpenStreetMap (OSM) road network data, all drivable roads for the area of interest are retrieved and stored as a graph. The nodes of this graph are all dead-ends of roads, “the point from which an edge self-loops” (Boeing, 2017, p. 132), intersections of multiple roads with at least one road continuing through or two roads with different OSM-IDs ending at an intersection (Boeing, 2017, pp. 132–133). The edges are all drivable public roads, excluding service roads (Boeing, 2017, p. 131). Then all nodes are deleted that are not intersections or dead-ends or have no edges, to exclude nodes that, e.g., indicate a curve in a road (Swanlund, Schuurman, Zandbergen, et al., 2020, p. 3).

Using the processed OpenStreetMap data, the locations to be masked are moved to the nearest node. Then the  $n$ -nearest nodes to the node, the points are moved to, are found, and the distance to those nodes is calculated and averaged. The value for  $n$ , in the article called “depth value”, can be any number. The average of the distances to the  $n$ -nearest nodes is then used as the maximum displacement radius. Within this radius, one node is randomly selected as the masked location (Swanlund, Schuurman, Zandbergen, et al., 2020, pp. 3–5).

As the value for depth, the authors suggested using 10, 20, or 30, with larger numbers achieving a lower risk of re-identification of the original location. They also state that this depends on the sensitivity of the data as well as other unspecified factors (Swanlund, Schuurman, Zandbergen, et al., 2020, p. 11). Further, they state that different values could be tried, and the  $k$ -anonymity could be assessed. The authors then suggest looking at the proportion of points achieving a  $k$ -anonymity with  $k$  being 25, 50, 100, and 200. However, it is not stated what the threshold for an acceptable proportion is nor why these values were chosen. But according to the table provided in the article and the statement that the results in the table indicate a low re-identification risk, achieving a proportion of less than 50% seems appropriate (Swanlund, Schuurman, Zandbergen, et al., 2020).

Street masking is very similar to location swapping in that the original location is replaced with coordinates of a point sampled from a set of points within a predefined radius. However, roads are used instead of residential addresses. In addition, points that are not close to a drivable road (e.g., in rural areas) are additionally moved, as all points are first moved to the nearest node.

For large study areas, as the one in this thesis, the application of the masking method will take far longer compared to most methods presented in this thesis. Therefore, depending on the road density of the area (number of nodes of the OpenStreetMap road network data) and the number of points to be masked, this method might not

be applicable.<sup>32</sup>

### Implementation of Street Masking (short: StreetMask)

The authors provide the Python code to apply the masking method (Swanlund, Schuurman, Zandbergen, et al., 2020, p. 5). Although not explicitly stated, the coordinates must be in a format to calculate the Euclidean distance. If not, the Python code causes an error and terminates. If the coordinates are in a different format and need to be converted, additional distortion occurs.<sup>33</sup> Therefore, it has been implemented in *R* for this thesis using the part of the Python code to download the road network and preprocess it as desired.

First, the streets are downloaded from OpenStreetMap for the region of interest according to the Python code provided, but modified to use the coordinates in a latitude-longitude format. This included cleaning the data according to the provided code. Then the nodes are saved and used for masking.

Second, the nearest node is found for every location. Third, the distance from these nodes to all other nodes is calculated. The distance of the nearest nodes (number of nearest nodes defined by the depth value), excluding the node itself, is averaged. Lastly, one node of the nodes with a smaller distance than the average is randomly drawn, and the coordinates are used as the masked location.

For this method, all streets from South Australia were retrieved using the coordinates of the area provided in the shapefile. The depth value, i.e., the number of nearest nodes for calculating the average distance, was set to 30, as this is the recommendation when a low risk of re-identification is the goal (Swanlund, Schuurman, Zandbergen, et al., 2020, p. 11). Additionally, *k*-anonymity was evaluated as the authors proposed, i.e., the proportion of points with 25, 50, 100, 200 points closer to the original location than the masked point was viewed. However, instead of including all possible addresses (including non-residential addresses), the residential file was used. For depth = 30 and *k* = 200 only about 10% fulfilled *k*-anonymity. According to the results of the article, a low risk of re-identification is achieved if there are more than 50% of the points fulfilling *k* = 200 (Swanlund, Schuurman, Zandbergen, et al., 2020, p. 6). Therefore, another depth value (depth = 100) was chosen that meets this criterion.

---

<sup>32</sup>Similarly to street masking, the published crime data by *Police.uk* use a not publicly available list of predefined points (such as the midpoint of a street or point above a public place) and moves the coordinate to the closest point in that list (Police.uk, n.d.). If the crime location is more than 20 kilometers away from the closest point in the list, or the area does not contain at least eight postal addresses, the coordinates will not be used in maps (Police.uk, n.d.).

<sup>33</sup>In addition, the code causes several errors due to the fact that the required packages were updated. An attempt to use this code resulted in the code terminating after about five hours.

### 3.3. Coordinate Replacement

The last category of masking methods contains those methods that propose to only disclose an anonymized distance matrix, or a string of zero and ones, as opposed to geographic coordinates. These are described in the following.

#### 3.3.1. Random Projection

*Random projection* is a technique that originated in the field of record linkage where it seeks to solve the problem of linking people with changing addresses (Henecka, 2019). Although the problem it attempts to solve was not originally masking geographic coordinates, and the idea is only briefly described within a blog post, it is explained sufficiently to implement it.

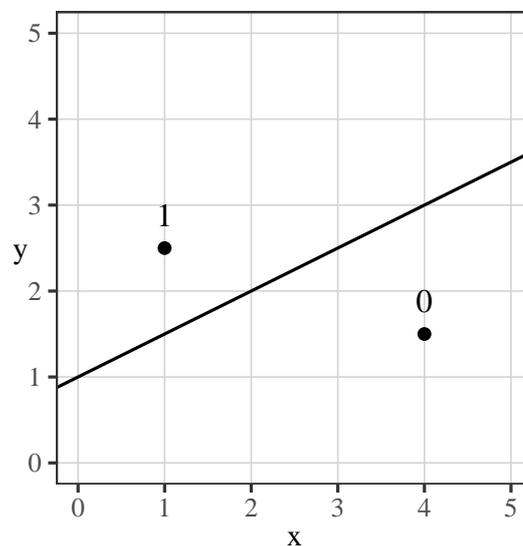


Figure 3.23.: Random projection example.

The area of the points is divided by so-called random projections, which is a straight line dividing the area into a positive side and a negative side, e.g., right and left, top and bottom (see figure 3.23). If the points fall into the upper or right area, they get the number one, if not the number zero. This is done  $n$ -times, so that  $n$  numbers are given for each point. The masked coordinate is then the  $n$  concatenated numbers, which is called a bit vector (Henecka, 2019).

It is not stated whether the random projections must be orthogonal and thus producing a grid-like structure or whether all lines fall randomly within the area. Therefore, it is assumed that random projections do not have to be orthogonal and that each line is independent of the other lines. Also, there is no specification as to how many random projections should be used to subdivide the area.

### Implementation of Random Projection (short: RandProj)

To create random lines that cut through the area of interest, two points within the area of interest are drawn, and the slope and intercept of the line passing through the points are calculated. The area of interest is defined by the maximum and minimum value of the easting and northing since this masking method can only be applied to a coordinate system with coordinates in meters. Adapted from linear regression, the residuals are calculated to show whether points fall on the positive or the negative side of the line. That is, the difference between the observed northing and the predicted value of the northing, given the easting value, is calculated. If the difference is positive, the points receive a one, and if it is negative, the points receive a zero. This is repeated as many times as needed to reach the predefined number of digits for the bit vector.

The area was randomly divided by 100 lines. This was repeated using 200 lines, 300, 500, and 1,000 lines.

### 3.3.2. Anonymization of Distance Matrices via Lipschitz Embedding

Another approach by Armstrong et al. (1999) involves neighbor information. The basic idea is that instead of moving the coordinates, they are used to calculate a distance matrix, and then coordinates are removed from the data set. With the so-called nearest-neighbor mask for each point, only the distance to the closest  $n$  neighbors is considered (Armstrong et al., 1999, p. 509).

This idea of not revealing the coordinates' information but only a distance matrix was further developed by Kroll and Schnell (2016), who used a variant of Lipschitz embedding<sup>34</sup> to anonymize geographical distance matrices since non-anonymized distance matrices still provide enough information for potential attacks. First the parameters  $d$  (dimension) and  $k$  (size) must be specified. The result is influenced by the choice of  $k$  and  $d$  in the following way: "Increasing values of  $k$  and decreasing values of  $d$  will increase the variance of approximated distances" (Kroll and Schnell, 2016, p. 5).

First,  $d$  reference sets of size  $k$  are created by selecting  $k$  uniformly distributed points within the considered geographical area (Kroll and Schnell, 2016, p. 3). Second, the distance between each point in the data set and the reference set is calculated using the Great-Circle-Distance (Haversine formula). The minimum distance for each point to any reference point is taken and set as the respective  $d$  column of the Lipschitz matrix.

$$p \mapsto f(p) := (f_1(p), \dots, f_d(p)) \in \mathbb{R}^d \quad (3.5)$$

$$f_i(p) := \min_{j=1, \dots, k} d(p, r_{ij}) \quad (3.6)$$

<sup>34</sup>For more information on Lipschitz embedding see Bourgain (1985).

With  $r_{ij}$  being the elements of the random reference sets, and  $p$  each point location. Lastly, the maximum metric (Chebyshev-distance) is calculated to yield the approximated distance matrix  $\tilde{D}$  (Kroll and Schnell, 2016, p. 4).

$$\tilde{d}(p, q) := \|f(p) - f(q)\|_\infty = \max_{i=1, \dots, d} |f_i(p) - f_i(q)| \quad (3.7)$$

$$\tilde{D} = (\tilde{d}_{ij}) \quad \text{with} \quad \tilde{d}_{ij} := \tilde{d}(p_i, p_j) \quad (3.8)$$

The resulting distance matrix is released together with the data set but without the coordinates. The distance matrix allows many calculations, such as spatial autocorrelation and cluster detection. On the other hand, this masking method does not allow the adding of more information to the data set since the actual coordinates are deleted.

To choose appropriate values for  $k$  and  $d$ , the authors proposed implementing the method multiple times with different values for  $k$  and  $d$ . The results should then be evaluated by attacking the method with another proposed method described in the article (Kroll and Schnell, 2016, p. 12). Since the evaluation of the attack is based on precision and recall, the parameters resulting in the lowest precision and the lowest recall should be taken. Furthermore, the authors note that this masking methods preserves smaller distances better than larger distances (Kroll and Schnell, 2016, p. 12).

### **Implementation of Anonymization via Lipschitz embedding** (short: Lipschitz Embedding / Lipschitz)

For this masking method, Martin Kroll's code is used (Kroll and Schnell, 2016). The optimal parameters for this masking method are found by performing a short simulation study based on a subsample of  $n = 200$  for the masked as well as the identification file. Both files contain at least three points for each combination of sex, age, and employment category. The data set for the simulation study was masked with each possible combination of the dimension  $d = \{20, 60, 100\}$  and the size  $k = \{5, 10, 20, 30\}$  with 20 replications each. Then for each of the masked data sets, the attack by Kroll (2015) was applied to find the optimal parameters for this masking method. For each of the combinations of  $d$  and  $k$ , the arithmetic mean of the 20 replications was calculated for precision and recall. For a table of results, see appendix D.1. The parameter combination with acceptable (low) precision and recall considering the influence of  $d$  and  $k$  was chosen. This was  $d = 60$  and  $k = 20$ .

### 3.3.3. Distance Approximation Using Intersecting Sets of Grid Points

In 2014, Farrow proposed and Schnell, Klingwort, et al. (2021) enhanced the idea of using intersecting sets of grid points (ISGP) as distance approximation between two locations.<sup>35</sup> This can also be used to mask geographic coordinates. Instead of releasing the geographic locations themselves, only approximated distances are released. Without any alterations, the distance matrix can easily be converted back to coordinates. Therefore, minimal alterations to the distance matrix are needed. These minimal differences in the true distance are achieved by approximating the distance using intersecting sets of grid points. An application of this method can be found in Klingwort et al. (2020). In the following, the method of approximating distances is described.

Given two points  $P$  and  $Q$  the true distance  $d$  can be approximated by the intersection of two circles of radius  $r$  surrounding the two points (see figure 3.24), if  $0 \leq d \leq 2r$  (Schnell, Klingwort, et al., 2021, p. 3). The area of intersection between two circles  $A(d)$  can be calculated using geometry leading to the following formula (Schnell, Klingwort, et al., 2021):<sup>36</sup>

$$A(d) = 2r^2 \cdot \arccos\left(\frac{d}{2r}\right) - \frac{1}{2}d \cdot \sqrt{4r^2 - d^2} \quad (3.9)$$

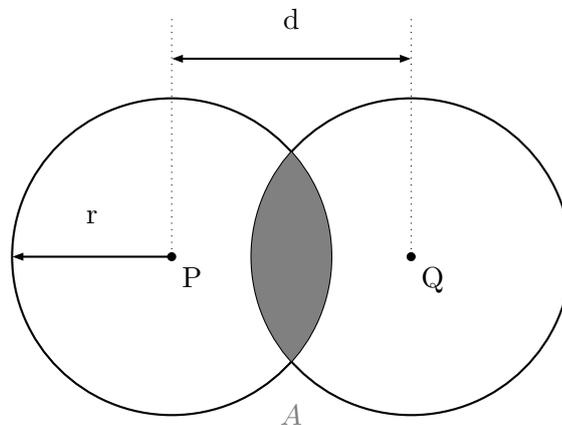


Figure 3.24.: The distance between two points  $P$  and  $Q$  can be approximated using the area of intersection of two circles (gray area) surrounding the points (figure taken from Schnell, Klingwort, et al., 2021, p. 3).

If  $A(d)$  is known,  $d$  can be calculated. The distance  $d$  between two points is approximated by laying a regular grid over the coordinates so that several grid points lie within the circles surrounding the points (see figure 3.25). Based on the number of grid points in each circle and the number of grid points in the area of intersection, a

<sup>35</sup>The idea was patented by Farrow (2015).

<sup>36</sup>A proof of the formula can be found in the appendix of the article of Schnell, Klingwort, et al. (2021).

suitable similarity measure can be calculated. The authors proposed the use of the Dice coefficient (Dice, 1945).

$$s = \frac{2|\mathcal{G}_P \cap \mathcal{G}_Q|}{|\mathcal{G}_P| + |\mathcal{G}_Q|} \quad (3.10)$$

$|\cdot|$  denotes the number of elements in the respective set of grid points  $\mathcal{G}_P$  and  $\mathcal{G}_Q$ .

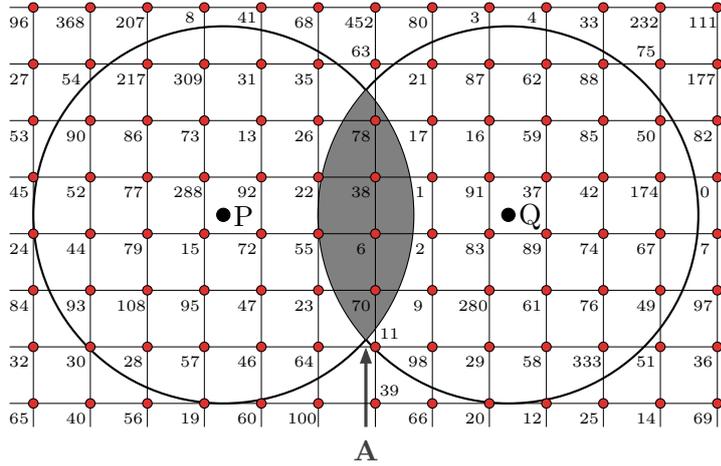


Figure 3.25.: Example of regular grid with random numbers laid over the two points (figure taken from Schnell, Klingwort, et al., 2021, p. 3).

The area of intersection can then be approximated using:

$$\hat{A} = s \cdot \pi r^2 \quad (3.11)$$

The approximation for  $d$  is the result of solving the equation  $A(d) = \hat{A}$ . In the case where the distance between two points exceeds twice the radius, i.e., no area of intersection exists, the distance  $2r$  is assigned (Schnell, Klingwort, et al., 2021, p. 12).

A simulation study performed by Schnell, Klingwort, et al. (2021) showed that the choice of the number of grid points, as well as the size of the radius, affects the quality of the results. If too few grid points are defined, areas of intersection may not contain any grid point and give the false impression of long distances (Schnell, Klingwort, et al., 2021, p. 11). Therefore, the more grid points are used, the more accurate the results.

An unsuitable choice for the size of the radius leads to high mean errors. To find a suitable size for the radius, the authors recommend performing a simulation study with varying radii and choose the radius that leads to the smallest absolute relative error (Schnell, Klingwort, et al., 2021, pp. 10–11). The absolute relative error is obtained by calculating the difference between the true distance and the approximated distance divided by the true distance. This is averaged over all points, and the absolute value

is taken.<sup>37</sup>

Another approach to this method is to use a random grid. However, using a random grid gives results that are less accurate than using a regular grid (Schnell, Klingwort, et al., 2021, p. 6). Therefore, the masking method was applied using a regular grid (Klingwort et al., 2020).

Also, the method is intended to mask the distance to a landmark rather than the distance of locations towards each other. For the latter, the choice of the radius is even more difficult. Choosing radii too small results in many larger distances having to be set equal to  $2r$ . Another disadvantage of this method is that a large number of grid points are needed for large areas, which both slows the computational time and takes up more internal memory. Moreover, this masking method is limited to using the Euclidean distance (Schnell, Klingwort, et al., 2021, p. 14).

### **Implementation of Distance Approximation Using Intersecting Sets of Grid Points** (short: ISGP)

For the implementation, the code provided by the authors is used and modified to fit the given data. The code in this thesis calculates the distance of the points to each other and not to a point of interest, as originally intended. Additionally, instead of using a polygon of the study area, the furthest coordinates minus the largest tested radius are used as borders of the area of interest, to ensure that for points close to the border, the radius is full of grid points.

No recommendations for the number of grid points are given. The more points used, the more accurate is the distance approximation. Therefore, it was arbitrarily set to 5,000,000, resulting in points being about 651 meters apart. A different seed for the random grid was set for each of the 50 iterations. The furthest coordinates minus the maximum tested radius (110,000 meters) rounded to the nearest hundred were used as borders for the grid points. For the radius, a simulation study was performed on a 5% sample ( $n = 500$ ). The range for the radius was based on the mean and median distance between points, similar to Klingwort et al. (2020, p. 6). Thus, the radii 30,000 to 110,000 meters were tested in 1,000 m increments. The result is shown in appendix D.2. The chosen radius is  $r = 40,000$ .

## **3.4. Other (Masking) Methods**

One of the masking methods cannot easily be placed in one of the categories. The method is called “masking based on the Military Grid Reference System” (Clarke, 2016) and will be explained below. Furthermore, this section also provides details on the methods which are currently not viewed as geomasking methods (Zandbergen, 2014, p. 11).

---

<sup>37</sup>According to the code provided by the authors, the absolute relative error is calculated only for the approximated distances smaller than  $2r$ .

### 3.4.1. Masking Based on the Military Grid Reference System (MGRS)

In 2016, Clarke presented a masking method based on the military grid reference system (MGRS; see, e.g., Hager et al., 1992). First, the given coordinates are translated to the MGRS system. The coordinates are then masked using the method described below. Afterward, the coordinates are translated to the original coordinate system.

The MGRS system translates the coordinates into a fifteen digit format (see figure 3.26). The first two digits indicate the zone number, the third digit is a letter indicating the grid cell designator, and the fourth and fifth digits, which are letters as well, represent the cell square identifier. The last ten digits are the easting (digit 6-10) and northing (11-15). The masking method does not change the first five digits/characters (Clarke, 2016, p. 303).

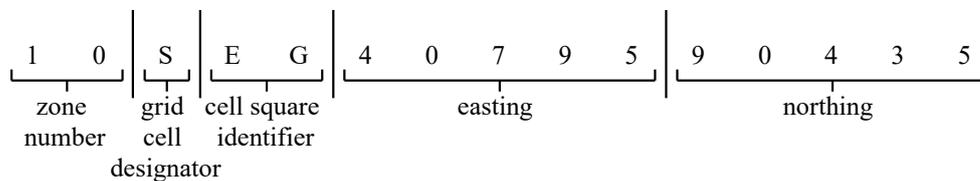


Table 3.1.: Example of masking based on MGRS.

Original:	0	1	2	3	4	5	6	7	8	9
Encoded:	9	7	5	8	6	4	0	3	2	1

	easting					northing				
	4	0	7	9	5	9	0	4	3	5
Level 1:	4	0	7	9	4	9	0	4	3	4
Level 2:	4	0	7	1	4	9	0	4	8	4
Level 3:	4	0	3	1	4	9	0	6	8	4
Level 4:	4	9	3	1	4	9	9	6	8	4
Level 5:	6	9	3	1	4	1	9	6	8	4

(Clarke, 2016, p. 304). The descriptive statistics are the minimum, maximum, mean, median, and standard deviation of the easting and northing, and the square root of the sum of the squared standard deviations of the easting and northing (Clarke, 2016, p. 304).<sup>38</sup>

First, the following descriptive statistics are calculated separately for the easting and northing: minimum, maximum, mean, median, and (corrected) standard deviation. The standard distance is calculated by adding the easting standard deviation and the northing standard deviation and taking the square root of this sum.<sup>39</sup>

The nearest neighbor index, going back to Clark and Evans (1954), is used to compare the given pattern with a random pattern. It results from the following formula (Pinder, Shimada, et al., 1979, p. 431; Pinder and Witherick, 1972, pp. 278–279):

$$R_n = \frac{\bar{d}_{obs}}{\bar{d}_{ran}} \quad (3.12)$$

with

$$\bar{d}_{obs} = \frac{d_1 + d_2 + d_3 + \dots + d_n}{n}$$

$$\bar{d}_{ran} = 0.5\sqrt{(a/n)}$$

with  $a$  the study area,  $n$  the number of points, and  $d$  the distance between a point

<sup>38</sup>Although Clarke (2016) provides an explanation of how the fit value is calculated, he did not explain terms such as “normalized distance difference” which left room for interpretation. Therefore, the author was contacted and asked for a more detailed explanation. He provided parts of the code, and the following description is based on that code.

<sup>39</sup>Note that Clarke (2016, p. 304) says in his article the standard distance is the “root of the sum of the squared standard deviations”, which is incorrect according to the example in his article and the code provided. In fact, squaring the standard deviation, i.e., using the variance, is unnecessary.

and its nearest neighbor.<sup>40</sup>

For each statistic, the value for the masked data is subtracted from the original data, taking the absolute value so that the subtraction does not result in negative numbers. This is then multiplied by the absolute difference of the original nearest neighbor index and the masked nearest neighbor index (Clarke, 2016, p. 304).

A critical aspect of the masking method is that it does not acknowledge crossing zones, i.e., different grid cell designator and cell square identifier: “As yet no allowance is made for point sets that cross MGRS zone or cell boundaries” (Clarke, 2016, p. 304). This also impacts the nearest neighbor index because the maximum area size is limited to a  $99,999 \times 99,999$  meters area since all points must belong to the same zone.

Within an MGRS cell, if level 1 is chosen, points can be moved 9 meters up or down, left or right. For level 2, the maximum displacement by changing the numbers is 99 meters up or down and 99 meters left or right. For the other levels the displacements are accordingly (Clarke, 2016, p. 302).

Due to these area restrictions, this masking method cannot be applied to the chosen data set, which is a major drawback of this method since it first has to be checked whether the coordinates of the data set of interest fall into one cell when transformed to the MGRS.

### 3.4.2. Alternative Methods

As mentioned before, some methods are not commonly referred to as geomasking methods but can be used alternatively to protect the privacy of spatial information (Zandbergen, 2014, p. 11). These methods are briefly described below.

#### 3.4.2.1. Controlled Access

*Controlled access* uses a virtual organization into which sensitive information, including spatial data, is transferred so that the data user can perform the analysis with other data sets within the virtual organization (Kamel Boulos et al., 2006). This is not viewed as a geomasking method because it does not change coordinates. Similarly, Ajayakumar et al. (2019) proposed a tool named *Privy* which enables data holders to share their data without confidentiality concerns. First, the geographic coordinates are displaced and rotated (storing information on the parameters). Then the data set is sent to a collaborator who can analyze the data set but does not know how the spatial coordinates have been modified. Finally, the results of the analysis are then sent back to the data holder. If needed, the coordinates can be moved to their original location using the stored parameter choices.<sup>41</sup>

---

<sup>40</sup>In literature, a wrong formula of the nearest neighbor index is found, which was pointed out by Pinder, Shimada, et al. (1979, p. 430).

<sup>41</sup>A similar approach of allowing access to spatial data without releasing the data set can be found in Rao, Gao, Li, et al. (2021).

#### 3.4.2.2. Flexible Aggregation

*Flexible aggregation* is used for aggregated data and not at the coordinate level. Two methods are described, namely random record swapping and local density swapping (Young et al., 2009). Both are very similar to location swapping and verified neighbor, both of which are designed to mask coordinates (as opposed to aggregated data) and are included in this thesis.

#### 3.4.2.3. Spatial Smoothing

*Spatial smoothing* is a method that was not originally designed for spatial data. Instead, it calculates a weighted average of non-geographic variables of records that are close in terms of their spatial location (Zhou et al., 2010). However, it led to multiple imputation approaches designed for spatial data. Here, a model is estimated to simulate multiple new coordinates for each record using other variables. These multiple coordinates are then released (Wang and Reiter, 2012). Again, this method does not fall directly under the definition of geomasking methods and, therefore, is not tested here.

#### 3.4.2.4. Synthetic Data

Similarly, the method *synthetic data* proposes drawing randomly from a distribution of a variable. This method is not intended to be applied to coordinates but can be extended to draw coordinates from the joint distribution of, e.g., the latitude and longitude (Huckett, 2008). More recently, Rao, Gao, Kang, et al. (2020) propose using deep learning approaches for creating synthetic data.

#### 3.4.2.5. Linear Programming

*Linear programming* replaces the coordinates with coordinates from another data set by drawing a new location using a multinomial distribution with transition probabilities solved by linear programming (Wieland et al., 2008). The authors propose that the other data set (out of which a new location is sampled) should contain points arranged in a grid (for individual addresses as input) or small administrative units (for areas as input). Part of the constraint equations for linear programming is that the risk of linking the masked location to the original location is small. The method states that a small risk of re-identification is achieved if three assumptions are met: the underlying population size is known at each location (i.e., 1 for addresses), the intruder has no knowledge of any non-random membership to the data set if the membership to the data set is not random, and no other information (e.g., demographic variables) about the individuals are available to help re-identify locations (Wieland et al., 2008). As

will be seen, the scenario considered in this thesis assumes that additional information is known.

## 4. Risk-Utility Framework

Geomasking methods were developed to allow researches access to the social-spatial-relationship, while maintaining privacy (Armstrong et al., 1999, p. 501). This can be translated into Duncan and Fienbergs (1999) *R-U Confidentiality Map* (see figure 4.1; also referred to as *risk-utility map*), which was further elaborated on in the papers of Duncan, Fienberg, et al. (2001) and Duncan, Keller-McNulty, et al. (2001).

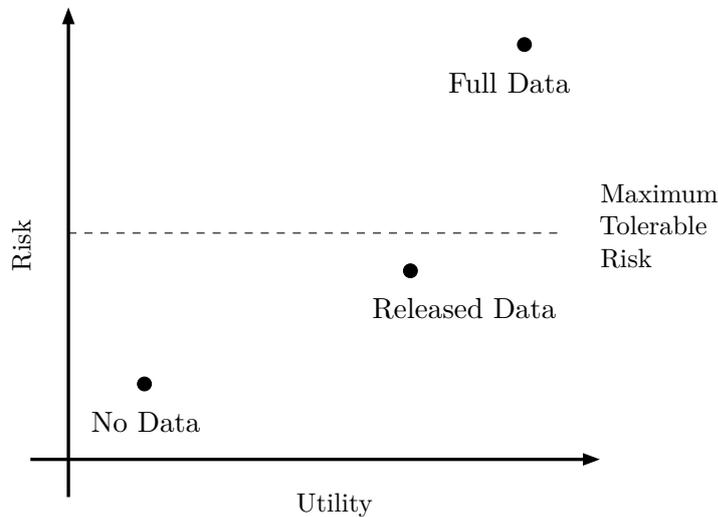


Figure 4.1.: Risk-Utility Confidentiality Map (based on Duncan and Fienberg, 1999, p. 352; Duncan, Keller-McNulty, et al., 2001, p. 7).

The map, respectively, the graph allows for a quantitative analysis of masking methods. The idea is that every imposed masking method reduces the risk of disclosing respondents,  $R$ , but at the same time lowers the data utility,  $U$  (Duncan, Fienberg, et al., 2001, p. 139). Therefore, the map is the visual description of the trade-off between the risk and the utility of a masking method (Duncan, Keller-McNulty, et al., 2001, p. 6; Elliot, Mackey, et al., 2016, p. 16). The threshold (dashed line) indicates the maximum tolerable risk. However, Duncan and his co-authors only provided a brief description of what the utility and risk comprise. Namely, risk can be calculated as the percentage of easily identifiable respondents and utility as preserving statistical information (Duncan, Keller-McNulty, et al., 2001, p. 6). However, it lacks a proper definition (suitable for geographic masking methods) of how risk is calculated and how preserving statistical information is quantified. In the following, the used measures for utility and risk are described. An overview is found in figure 4.2 and serves as a guide for this chapter.

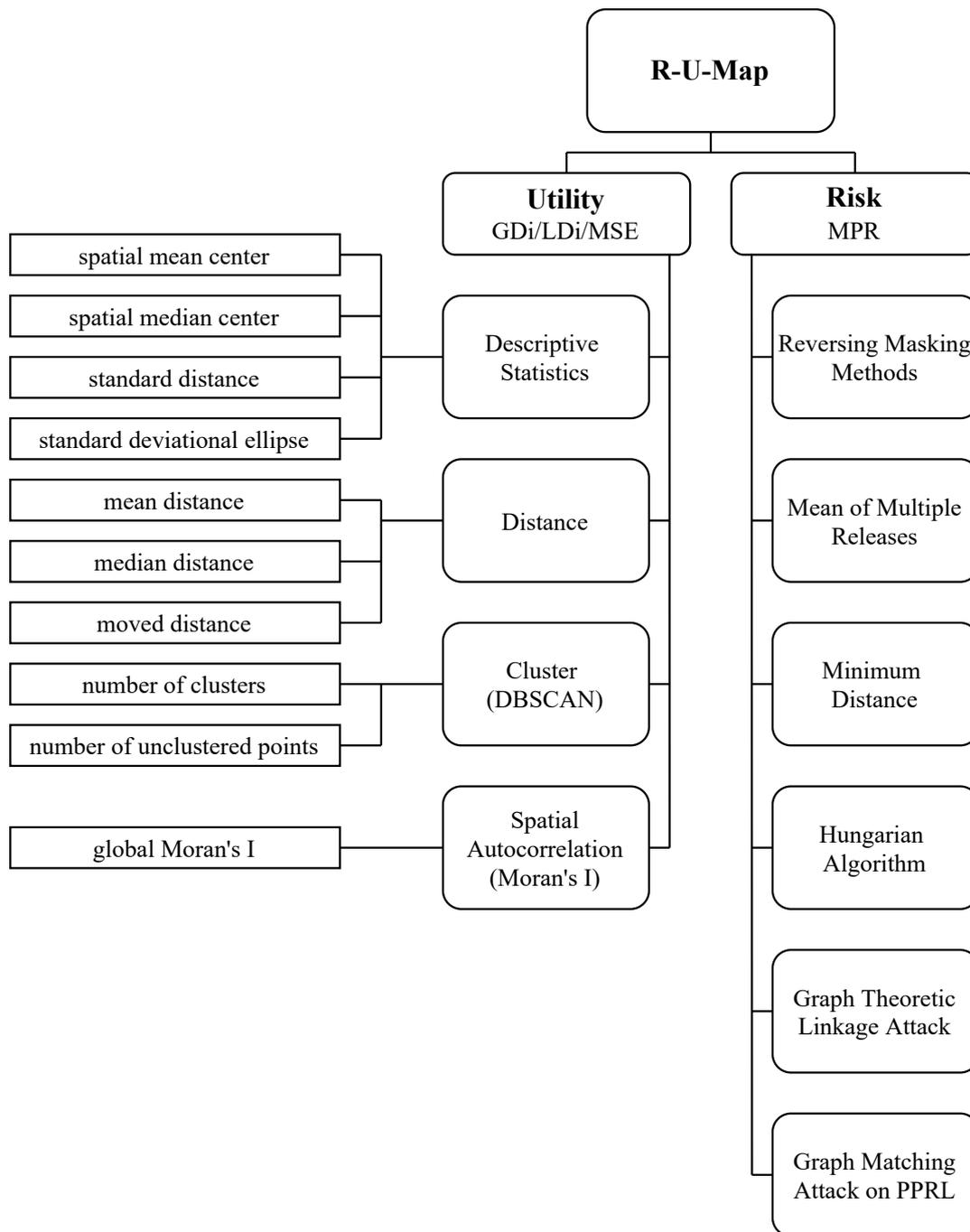


Figure 4.2.: Overview of risk-utility measures.<sup>1</sup>

## 4.1. Utility Evaluation

In the geomasking literature, several criteria for utility have been proposed. In general, the utility is defined as the usefulness of the data after applying the masking method (Duncan, Keller-McNulty, et al., 2001, p. 6). A data set is useful if the same analysis

<sup>1</sup>The individual components of this figure will be described in the following sections. See list of acronyms for explanation of abbreviations.

can be performed with the masked data as using the original data without losing information such as results or the quality or significance of results.

Based on previous analysis of the utility of masking methods, the dimensions for the usefulness of the masked data set can be summarized as preserving distances, preserving clusters, preserving descriptive statistics of coordinates, and, including third variables, spatial autocorrelation/patterns, maintaining relationships between variables / inference results and preserving trends (see, e.g., Armstrong et al., 1999, p. 510; Cassa, Grannis, et al., 2006, p. 163; Richter, 2017, p. 112; Seidl, Paulus, et al., 2015, p. 255; Rushton et al., 2008, pp. 133–134).

Authors such as Leitner and Curtis (2004/2006) have proposed using visualization to evaluate the utility of a masking method. They plotted the original and the masked coordinates and let students decide whether these maps looked similar or different. Since this is based on a subjective evaluation, visualization is not used as a criterion in this thesis. Clifton and Gehrke (2013) suggested using the percent root mean squared error (percent RMSE or PRMSE) to evaluate the utility. The percent root mean squared error is based on several so-called “built environment measures” such as the number of people per acre, jobs per acre, and distance to a point of interest (nearest grocery store, nearest bus stop) (Clifton and Gehrke, 2013, p. 44). However, the rationale for choosing these “built environment measures” by the author remains unknown. Moreover, less popular criteria in the geomasking literature include event-geography relations, trends, anisotropies (direction-dependent spatial correlation; Armstrong et al., 1999), and semivariograms (Seidl, Paulus, et al., 2015).

For the present analysis, the dimensions on which the literature agrees are used. Namely, preserving descriptive statistics, preserving distances, spatial autocorrelation (also used to evaluate whether the relationship to other variables is preserved), and preserving clusters. Each dimension has multiple measures, which are explained in the respective subsection of this chapter.

#### **4.1.1. Descriptive Statistics**

Descriptive statistics of a data set are used to understand the data and their distribution (see e.g. Oyana and Margai, 2016, p. 56; Panigrahi, 2014, p. 169; Lach Arlinghaus and Kerski, 2014, p. 187). Accordingly, for spatial data, descriptive statistics are used to understand the spatial structure. The spatial data must be in a coordinate system allowing the Euclidean distance formula.<sup>2</sup>

##### **4.1.1.1. Spatial Measures of Central Tendency**

When considering descriptive statistics in general, the following are usually of interest: mean, weighted mean, median, and mode, as measures of central tendency and range,

---

<sup>2</sup>For a more detailed explanation of calculating geographic distances see chapter 4.1.2.

standard deviation and variance as measures of dispersion (Oyana and Margai, 2016, pp. 59–62). Some authors also use standard error, skewness, kurtosis (Panigrahi, 2014, p. 174). For spatial data, much fewer descriptive statistics are used in the literature. The spatial measures of central tendency used are the mean, the weighted mean, and the median.

### (Weighted) Spatial Mean

The spatial mean is the “average value of observed points for each of the [...] coordinates” (Oyana and Margai, 2016, p. 64).

$$\bar{x} = \sum_{i=1}^n x_i/n \quad (4.1)$$

$$\bar{y} = \sum_{i=1}^n y_i/n$$

In the weighted mean, the weights represent the frequency or magnitude of a third variable at a given location. To calculate frequencies, the data points usually need to be grouped to obtain an area (Oyana and Margai, 2016, p. 59). The center of the group is then used as the coordinate input for the formula.<sup>3</sup>

$$\bar{x}_w = \sum_{i=1}^n x_i w_i / \sum_{i=1}^n w_i \quad (4.2)$$

$$\bar{y}_w = \sum_{i=1}^n y_i w_i / \sum_{i=1}^n w_i$$

Since the weighted mean is applied to areas rather than the individual points, it will not be calculated in the subsequent analysis.

### Spatial Median and Mode

The spatial median, like the regular median, is the middle value of a sorted list of coordinates, i.e., the 50th percentile (Oyana and Margai, 2016, p. 57). The mode is the value that occurs most often in the data (Oyana and Margai, 2016, p. 57). For the spatial mean center and the spatial median center, the coordinates have to be in a coordinate system that allows the calculation of the Euclidean distance.

---

<sup>3</sup>Using the centroid of areas is common if lattice data is given. Lattice data is the term for spatial data, which is not in the form of point data but rather an area, e.g., counties instead of coordinates (Cressie, 1992, p. 614).

#### 4.1.1.2. Spatial Measures of Dispersion

As spatial measures of dispersion, the standard distance, weighted standard distance, and standard deviational ellipse are used (Oyana and Margai, 2016, p. 69).

##### (Weighted) Standard Distance

The standard distance ( $SD$ ), like the standard deviation, is the measurement for the dispersion around the coordinates' spatial mean.

$$SD = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2 + \sum_{i=1}^n (y_i - \bar{y})^2}{n}} \quad (4.3)$$

The same weight is used for the weighted standard distance as for the spatial mean, i.e., the frequency of another variable in a given area.

$$SD_w = \sqrt{\frac{\sum_{i=1}^n w_i (x_i - \bar{X})^2 + \sum_{i=1}^n w_i (y_i - \bar{Y})^2}{\sum_{i=1}^n w_i}} \quad (4.4)$$

As with the weighted spatial mean, the weighted standard distance is not calculated. Again, the standard distance requires a coordinate system that allows the use of the Euclidean distance.

##### Standard Deviational Ellipse

The standard deviational ellipse to measure geographic coordinates' concentration is an idea proposed by Lefever (1926). First, the origin of the coordinate system must be moved to the mean center, and the respective coordinates must also be moved. Then the following formula is used to find the angle of rotation ( $\theta_m$ ) for which the standard deviation along the x-axis is the maximum and along the y-axis the minimum (Lefever, 1926, pp. 90–91).

$$\tan \theta_m = \frac{-A \pm \sqrt{A^2 + 4B^2}}{2B} \quad (4.5)$$

with

$$A = \sum x^2 - \sum y^2 \quad (4.6)$$

$$B = \sum xy$$

and  $x$  and  $y$  being the deviations of the mean center.

Hereafter, the standard deviation along the x-axis ( $a$ ; major axis) and the y-axis ( $b$ ;

minor axis) is calculated (Lefever, 1926, pp. 90, 93).

$$a = SD_{x-axis} = \sqrt{\frac{\cos^2 \theta_m \sum x^2 + 2 \sin \theta_m \cos \theta_m \sum xy + \sin^2 \theta_m \sum y^2}{N}} \quad (4.7)$$

$$b = SD_{y-axis} = \sqrt{\frac{\sin^2 \theta_m \sum x^2 - 2 \sin \theta_m \cos \theta_m \sum xy + \cos^2 \theta_m \sum y^2}{N}}$$

Then the standard deviational ellipse can be drawn. Usually, the length of the axes is defined as one standard deviation. This can be extended with two or three standard deviations. To calculate the standard deviational ellipse, the coordinates must be projected into a coordinate system that allows the use of the Euclidean distance formula.

Years after Lefever (1926) and multiple critiques regarding the shape of the standard deviational ellipse (see, e.g., Furfey, 1927 and Yuill, 1971), Ebdon (1977) proposed formulas for calculating the standard deviational ellipse with the same results as Lefever but from a different perspective. He calculated the rotation angle  $\theta$  with the formula (Ebdon, 1977, p. 114):

$$\tan \theta = \frac{(\sum x'^2 - \sum y'^2) + \sqrt{(\sum x'^2 - \sum y'^2)^2 + 4(\sum x'y')^2}}{2 \sum x'y'} \quad (4.8)$$

which is the same formula used by Lefever, but without the preceding minus sign. He then modified the calculation of the standard deviations along the x-axis and y-axis by switching sinus and cosine. Thus, the standard deviation along the x-axis in Ebdon's formula corresponds to the standard deviation along the y-axis in Lefever's formula and vice versa. These formulas give the same result because Lefever uses the x-axis as the reference axis and rotates the axes counterclockwise. In contrast, Ebdon uses the y-axis as the reference axis and rotates the axes clockwise. This also explains why Lefever's  $SD_x$  value is Ebdon's  $\sigma_y$  value.<sup>4</sup>

$$\sigma_x = \sqrt{\frac{(\sum x'^2) \cos^2 \theta - 2(\sum x'y') \sin \theta \cos \theta + (\sum y'^2) \sin^2 \theta}{N}} \quad (4.9)$$

$$\sigma_y = \sqrt{\frac{(\sum x'^2) \sin^2 \theta + 2(\sum x'y') \sin \theta \cos \theta + (\sum y'^2) \cos^2 \theta}{N}}$$

<sup>4</sup>The corresponding function *calc\_sde* of the R-package *aspace* (Bui et al., 2012) uses a different formula and refers to Ebdon's second edition of the book from 1988. However, even the reprinted and corrected second edition of this book does not show the formulas used in the R-package (Ebdon, 1990). Therefore the R-package was not used, and the code was written based on the formulas stated by Lefever.

### 4.1.2. Preserving Distances

The calculation of distances between spatial points is necessary to discover relationships between geographic coordinates (Lawhead, 2015, p. 158). The idea that the values of a variable can be related to their spatial location has its roots in the *laws of geography*.

The first law of geography states “everything is related to everything else, but near things are more related than distant things” (Tobler, 1970, p. 236).<sup>5</sup> The second law of geography states that “geographic variables exhibit uncontrolled variance” (Goodchild, 2004, p. 302). The third law of geography was recently proposed by Zhu et al. and states: “The more similar geographic configurations of two points (areas), the more similar the values (processes) of the target variable at these two points (areas)” (Zhu et al., 2018, p. 230). Even though they take different perspectives, all three laws show that values can be associated with their geographic location, and such association can and should be analyzed.

The problem regarding the measurement of distance in geographic coordinate systems is that although maps of the earth exist, they do not accurately project the earth (Lawhead, 2015, p. 158). Therefore, distances between two spatial points based on map projections do not correspond to the actual distance. The solution for measuring distances depends on the earth model considered, the accuracy needed, and what is being measured (Lawhead, 2015, p. 158).

In general, there are four different earth models. The first is the flat earth model, for which Euclidean geometry is used to calculate distances. Using a flat earth model results in substantial differences in the calculated and actual distances. Other models consider the earth to be spherical or ellipsoid. For these, the Haversine formula and Vincenty’s formula have been proposed (Lawhead, 2015, p. 160). The last and most accurate earth model is the geoid model, for which no formula has yet been provided (Lawhead, 2015, p. 160).

The Haversine formula for calculating the distance between two points was proposed by Sinnott (1984).<sup>6</sup> A more comprehensible notation provides Panigrahi (2014, p. 213):

$$\text{haversin}\left(\frac{d}{R}\right) = \text{haversin}(\Delta\phi) + \cos(\phi_1)\cos(\phi_2)\text{haversin}(\Delta\lambda) \quad (4.10)$$

with the coordinate pairs  $(\phi_1, \lambda_1)$  and  $(\phi_2, \lambda_2)$ ,  $\Delta\lambda = \lambda_2 - \lambda_1$ ,  $\Delta\phi = \phi_2 - \phi_1$ ,  $d$  the distance, and the mean radius of the earth  $R$ .

<sup>5</sup>Although Tobler is considered the inventor of the first law of geography, similar proposals can be found earlier, for example, in Krige (1951, p. 135). Also, Goodchild proposed that Tobler’s law should be second and his law of spatial heterogeneity the first (Goodchild, 2004, p. 302).

<sup>6</sup>Sinnott (1984) is not the inventor of the usage of half the versed sine, but he was the first to use this for distance calculations of two spatial points.

Given  $\text{hav}\sin\theta = \sin^2(\theta/2)$  (Panigrahi, 2014, p. 214), solved for  $d$  yields:

$$d = R \cdot 2\sin^{-1} \left( \sqrt{\text{hav}\sin(\Delta\phi) + \cos(\phi_1)\cos(\phi_2)\text{hav}\sin(\Delta\lambda)} \right) \quad (4.11)$$

The mean radius of the earth is 6,371 km on average (equatorial radius 6,378.137 km and polar radius 6,356.752 km).

The Vincenty formula proposed in 1975 is a more complex but more precise method for calculating the distance between two points. For the calculations in this formula, the major ( $a$ ) and minor semi-axes ( $b$ ) of the ellipsoid (according to the projection) are needed as well as the flattening  $f$  (Vincenty, 1975).

Lawhead (2015) proposed that for spherical earth models, the Haversine formula should be used, and for ellipsoid earth models, Vincenty's formula should be used. However, the latter is far more complex, and other authors pointed out that the Haversine formula is accurate to about 0.3%, which is sufficient (see, e.g. Panigrahi, 2014, p. 214). Further, the accuracy of Vincenty's formula applies to the ellipsoid earth model and not the more accurate geoid model (Panigrahi, 2014, p. 214).<sup>7</sup> Although there is the possibility of using a projection that allows the calculation of the Euclidean distance such as UTM, other problems exist, such as inaccurate results when points are located in different zones.

As measures for preserving distances, the distance between the original location and the masked location, and between the original points and between the masked points is of interest (Armstrong et al., 1999, p. 510; Cassa, Grannis, et al., 2006, p. 163; Richter, 2017, p. 112; Seidl, Paulus, et al., 2015, p. 255). This is measured by comparing the distances of the original points to each other with the distances of the masked points to each other, as well as measuring the average distance of the points from their original location. The utility is high if the difference of the average distance between points is as small as possible, and points are moved as little as possible (Richter, 2017, p. 112). In addition, since outliers strongly influence the average, the median distance between points is calculated as well.

Maintaining the distance between the coordinates and keeping the distance to their original location small are two essential goals one wants to achieve when masking coordinates. But simply measuring how far points are moved does not give information on how the spatial structure might be changed, for instance, as seen in figure 4.3. The original point structure shows no clustering in the data. However, four points were moved closer to each other. And the masked coordinates show clustering of the points, even though they were not moved very far. Therefore, the data set's clusters and spatial autocorrelation has to be measured.

<sup>7</sup>For the data set used, the difference between the Haversine formula and Vincenty's formula for the two most distant points in the sample is 0.19%. Therefore, the Haversine formula is considered sufficient.

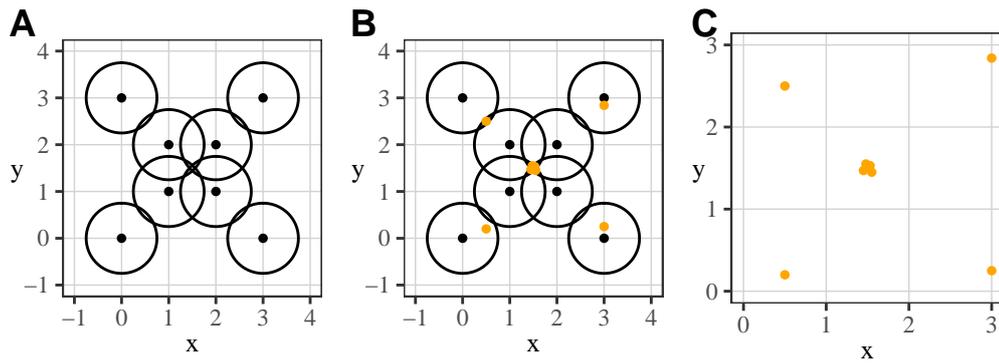


Figure 4.3.: Explanation of relevance of clustering methods to evaluate utility.

### 4.1.3. Preserving Clusters

Clustering methods seek to detect whether data contain underlying patterns (Han, Lee, et al., 2009, p. 150). For (spatial) data, clustering methods can be roughly divided into four categories: partitioning based algorithms, hierarchical algorithms, density-based algorithms, and grid-based algorithms (see e.g., Han, Lee, et al., 2009; Chitra and Maheswari, 2017; Chauhan et al., 2010).

#### Partitioning Algorithms

Partitioning algorithms randomly assign cluster centers to the data as an initial clustering. Then points are added, and the cluster membership is changed based on a specific criterion until the value for the criterion does not change (e.g. Han, Lee, et al., 2009, pp. 154–155). These algorithms are based on the assumption that each cluster must contain at least one element. However, some algorithms allow points to belong to more than one cluster, which is called fuzzy clustering.

A famous example of a partitioning algorithm is k-means.<sup>8</sup> The k-means algorithm first randomly selects points as cluster centers, e.g., if two clusters should be the result, two points are randomly chosen. Then the remaining points are allocated to the clusters based on their distance to the cluster centers. In an iterative process, a new cluster center is calculated based on the points in each cluster (arithmetic mean of the coordinates), the distances between the points and the cluster centers are calculated, and points are assigned to the nearest cluster. If no change in the cluster center or assignment of the points to clusters is detected, the final allocation of points to clusters is reached (Han, Lee, et al., 2009, p. 156). In the case that no final solution can be found, the iteration is stopped at a predefined threshold, e.g., after 1,000 iterations (Han, Lee, et al., 2009, pp. 157–158).

Other partitioning algorithms are CLARANS (Clustering Large Applications based

<sup>8</sup>The name and idea go back to MacQueen (1967). The algorithm goes back to Lloyd in 1957 but was not published until 1982. The description follows Han, Lee, et al. (2009).

on RANdomized Search; Ng and Han, 2002), Neighborhood EM algorithm<sup>9</sup> (NEM; Ambroise and Govaert, 1998), PAM (Partitioning Around Medoids) and the related method  $k$ -medoids (Kaufman and Rousseeuw, 2005). A major drawback of this or other partitioning algorithms for detecting clusters is that the number of clusters must be known beforehand, and clusters are spherical and not arbitrarily shaped (Chauhan et al., 2010, p. 9; Han, Lee, et al., 2009, p. 158). Moreover, these algorithms are prone to outliers. In  $k$ -means, the centers are calculated as arithmetic means, and outliers strongly influence the result (Han, Lee, et al., 2009, p. 158).

### Hierarchical Algorithms

Hierarchical algorithms group data in the form of trees either agglomerative (from bottom to top) or divisive (from top to bottom) (Han, Kamber, et al., 2012, p. 449; Han, Lee, et al., 2009, p. 155; Kaufman and Rousseeuw, 2005, p. 199). From bottom to top, such as AGglomerative NESTing (AGNES), means that the algorithm starts with each point belonging to a single cluster. Then clusters are successively merged until one large cluster remains. Top to bottom algorithms, such as DIvisive ANALysis (DIANA), start with one large cluster containing all points and successively split the clusters until single point clusters are reached (Han, Kamber, et al., 2012, p. 449; Han, Lee, et al., 2009, p. 155; Kaufman and Rousseeuw, 2005, p. 253). The criteria used to merge or split clusters differ between hierarchical algorithms (Chauhan et al., 2010, p. 10; Han, Lee, et al., 2009, p. 155; Chitra and Maheswari, 2017, p. 112).

As an example, AGNES will be presented briefly (Kaufman and Rousseeuw, 2005). First, a distance matrix is calculated. The points with the smallest distance are clustered. Then, the distance matrix is updated to include the distances between the points and between the points and the cluster of the two merged points. Several options are possible here. Complete linkage calculates the distance between a point and all the clusters' points and uses the maximum. Single linkage calculates the distance between a point and all the clusters' points and takes the minimum distance. Average linkage takes the average of the distances, and centroid linkage calculates the distance between a point and the center of the cluster. Then, the points or clusters with the smallest distance are merged again. This process is repeated until a single large cluster containing all points is the result (Han, Lee, et al., 2009, pp. 163–165).

The process can be presented in a dendrogram (Han, Kamber, et al., 2012, p. 460), which can also be used to decide what the final number of clusters should be. A major drawback is that once a decision is made to combine or separate points, it cannot be revised in most algorithms (Han, Lee, et al., 2009, p. 155). Algorithms that are hierarchical and do not have this problem are BIRCH (Zhang, Ramakrishnan, et al.,

<sup>9</sup>The NEM algorithm is the EM algorithm (Dempster et al., 1977) modified to cluster spatial data (Ambroise and Govaert, 1998). The EM Algorithm is used to assign points to a cluster according to a weight (in this case, the weight is the membership probability), where the likelihood is penalized by a term that takes spatial information into account. For more details, see Ambroise and Govaert (1998). A short overview is given by Han, Lee, et al. (2009, pp. 160–162).

1996) and Chameleon (Karypis et al., 1999). Other problems include long processing times (Han, Lee, et al., 2009, p. 165).

### Grid-based Algorithms

Grid-based algorithms differ from the previous algorithms in that they do not work with the individual data points but use the regional space (Chitra and Maheswari, 2017, p. 113; Chauhan et al., 2010, p. 10; Han, Lee, et al., 2009, p. 156). These algorithms lay a grid over the data and use a density threshold to decide whether a cell contains (part of) a cluster. Neighboring cells that contain clusters are merged.

An example is CLIQUE (CLustering In QUEst; Agrawal et al., 1998). First, a grid with equal-sized cells is laid over the data. Using a density threshold, e.g., two points, each cell is categorized as “dense” or “not dense”. When two dense cells are adjacent, they merge into one cluster (Han, Kamber, et al., 2012, pp. 481–483). Other grid-based algorithms are STING (STatistical INformation Grid-Based; Wang, Yang, et al., 1997) and WaveClusters (Sheikholeslami et al., 1998). A major advantage is the fast processing time, and that outliers are taken into account (Chitra and Maheswari, 2017, p. 114; Han, Lee, et al., 2009, p. 156). The drawback is that the quality strongly depends on the grid size used (Han, Kamber, et al., 2012, p. 483). Related to this is also the disadvantage that if a border point of a cluster lies in a neighboring cell that does not exceed the density threshold, this point is not considered to belong to the cluster.

### Density-based Algorithms

Density-based algorithms use the density of the points to determine whether they form a cluster. A famous example is DBSCAN (Ester et al., 1996). In DBSCAN (Density-Based Spatial Clustering of Applications with Noise), data points are categorized into core points, directly density-reachable and density-reachable. Core points are points that have at least a predefined number of minimum points  $MinPts$  within a predefined radius  $\varepsilon$  (see figure 4.4 red point).

All points within the radius ( $\varepsilon$ ) are considered the (Eps-)neighborhood of a point (Ester et al., 1996, p. 227):

$$N_{Eps}(p) = \{q \in D \mid dist(p, q) \leq \varepsilon\} \quad (4.12)$$

Directly density-reachable points are points within the radius  $\varepsilon$  of a core point, but are not themselves core points (see figure 4.4 orange points; Ester et al., 1996, p. 228):

$$p \in N_{Eps}(q) \quad (4.13)$$

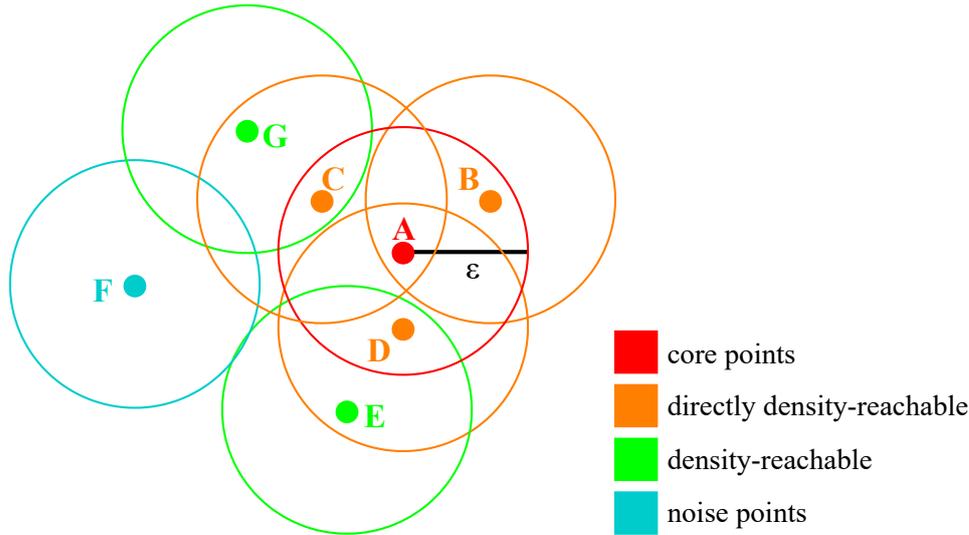


Figure 4.4.: DBSCAN example (radius= $\varepsilon$ , MinPts=4).

$$|N_{Eps}(q)| \geq \text{MinPts} \quad (4.14)$$

Density-reachable points are points that are not within the radius  $\varepsilon$  of a core point but are within the radius  $\varepsilon$  of a point that is within the radius  $\varepsilon$  of a core point (see figure 4.4 green points). Noise points are points that are not core points and are not (directly) density-reachable (see figure 4.4 blue point). Further, two points are density-connected if they are density-reachable from a point (Ester et al., 1996, pp. 227–228). In figure 4.4, points E and G are density-connected over A because E is density-reachable from A over D, and G is density-reachable from A over C.

To find adequate values for the input parameters  $\varepsilon$  and MinPts, Ester et al. (1996) and again Schubert et al. (2017, 19:11-19:12) suggest that the minimum number of points should be twice the number of dimensions, thus for geographic coordinates  $\text{MinPts} = 4$ . For  $\varepsilon$ , they propose plotting a  $k$ -dist graph. This is done by calculating the distance of each point to its  $k$  nearest neighbor, where  $k$  is MinPts. Then the distances are plotted in ascending order. The knee in the plot, i.e., the large change along the  $k$ -distance curve, is the value to be set for  $\varepsilon$  (Ester et al., 1996, p. 230). In addition, these values can also be set based on knowledge about the data.

Other density-based algorithms are OPTICS (Ordering Points To Identify the Clustering Structure; Ankerst et al., 1999), DENCLUE (DENsity-based CLUstEring; Hinneburg and Keim, 1998) and DBCLASD (Distribution Based Clustering of LARge Spatial Databases; Xu et al., 2017).

The disadvantage of this clustering method is that it does not perform well with high-dimensional data (Chitra and Maheswari, 2017, p. 113). Also, the need to initially set parameters is criticized (see, e.g., Chauhan et al., 2010, p. 10), although this is true for most clustering methods. On the other hand, these clustering methods are highly recommended when the goal is to find clusters with arbitrary shapes. They are also

recommended when the algorithm should not be prone to outliers, and the number of clusters is not known in advance (see, e.g., Han, Kamber, et al., 2012, p. 449). More recently, Schoier and Gregorio (2017) compared the DBSCAN algorithm to the k-Means algorithm and the Fast Search by Density Peak (FSDP) algorithm and were able to show that DBSCAN is superior to the other clustering methods for spatial data. They found that DBSCAN is especially useful to identify clusters in data sets containing outliers, needs few parameters, and is fast even for large data sets (Schoier and Gregorio, 2017, p. 581).

### Choice of Clustering Algorithm

Of interest is whether the spatial points form clusters and whether these are preserved when the coordinates are masked. Since the number of clusters is unknown, and outliers and arbitrary shapes of clusters are to be expected, partitioning algorithms and hierarchical algorithms are unsuitable. Although grid algorithms have the important advantage of fast processing time, they heavily depend on the initial cell width set, and border points might be ignored (Han, Kamber, et al., 2012, p. 483). Moreover, they cannot work with a distance matrix as input, which is required for some masking methods. Therefore, in the following analysis, the density-based algorithm DBSCAN is used. The parameters MinPts and  $\varepsilon$  are set, as suggested by Ester et al. (1996).

The resulting clusters are then compared based on their existence (number of clusters), and the size/ density and the number of correctly identified clusters (Armstrong et al., 1999, p. 511; Cassa, Grannis, et al., 2006, p. 162; Richter, 2017, p. 112; Seidl, Paulus, et al., 2015, p. 255).<sup>10</sup> Furthermore, Kounadi and Leitner (2016) point out to consider the clusters' specificity as the "percentage of masked points that originate from non-clustered original points and are still non-clustered" (Kounadi and Leitner, 2016, p. 65). This highlights the importance of obtaining the number of non-clustered points. Therefore, the number of non-clustered points is also compared, as well as if points that were clustered using the unmasked coordinates remain clustered and points that were non-clustered remain non-clustered.

#### 4.1.4. Spatial Autocorrelation

A major advantage of geographic information, i.e., geographic coordinates, is that it allows finding an underlying spatial pattern that explains the differences in values for a given variable. The spatial relation of a variable is called spatial autocorrelation (e.g. Griffith, 2003, p. 3). It should be analyzed and the results should be the same for the unmasked and masked data.

Common to all of the spatial autocorrelation measures is that a similarity coefficient

---

<sup>10</sup>There are also suggestions to compare locations of clusters (see, e.g., Seidl, Paulus, et al., 2015, p. 260). However, this would assume that it is known which cluster of the original file corresponds to which cluster based on the masked file.

(similarity measurement) is multiplied by a matrix of spatial connectivity (Dubé and Legros, 2014, p. 60):

$$\Gamma = \sum_{i=1}^N \sum_{j=1}^N w_{ij} \times c_{ij} \quad (4.15)$$

with  $w_{ij}$  the elements of a weight matrix, and  $c_{ij}$  a measure of similarity (or dissimilarity). The chosen similarity coefficient is often the correlation coefficient. Moran's I is an example of using a similarity measure and is more commonly used in literature (Dubé and Legros, 2014, p. 59), it will be the measure used in the subsequent analysis. Moran's I is a measure of spatial autocorrelation proposed by Moran in 1950. Moran's I calculates the correlation of a variable with itself. The basis is the Bravais-Pearson correlation coefficient which is calculated as follows (Fahrmeir et al., 2016, p. 126):

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}} \quad (4.16)$$

For Moran's I, the formula is modified to show the correlation in a variable among nearby locations (Panigrahi, 2014, p. 179).

The formula for Moran's I is the following (Dubé and Legros, 2014, p. 68):

$$I = \frac{N}{\sum_{i=1}^N \sum_{j=1}^N w_{ij}} \frac{\sum_{i=1}^N \sum_{j=1}^N w_{ij} (y_i - \bar{y})(y_j - \bar{y})}{\sum_{i=1}^N (y_i - \bar{y})^2} \quad (4.17)$$

where  $w_{ij}$  is the weight matrix. Therefore, Moran's I can be viewed as the spatial weighted Bravais-Pearson correlation coefficient (Waller and Gotway, 2004, p. 228).<sup>11</sup>

Most important for Moran's I is the weight matrix or spatial proximity matrix ( $w_{ij}$ ), which indicates whether two points or areas are close to each other (neighbors) or farther apart. This can be quantified either by distance or adjacency (e.g. Pfeiffer et al., 2008, p. 34; Waller and Gotway, 2004, pp. 224–225).

If “distance” is chosen, the distance between the points is calculated, and the inverse of the resulting distances are used (see, e.g., Ebdon, 1987, p. 163). Another option is to consider a point as a neighbor only if the distance falls below a predefined threshold. More complicated definitions of neighbors based on distance include row-standardization (Pfeiffer et al., 2008, p. 34). To obtain row-standardized weight matrices, the cell weight is divided by the row sum. This is done to better interpret the result of statistical tests and coefficients (Dubé and Legros, 2014, p. 50). In this thesis, the distance matrix is used as a weight matrix for the given data by using  $1/(d_{ij})$  (with  $d_{ij}$ , the distance between two points), so that short distances correspond

<sup>11</sup>A simplified example of the calculation of Moran's I is provided by Oyana and Margai (2016, p. 195).

to a larger weight, and then row-standardizing the result.<sup>12</sup>

“Adjacency” uses several methods and degrees (Pfeiffer et al., 2008, p. 34). The methods are *rook contiguity* and *queen contiguity*. Rook contiguity considers two areas as neighbors if they share a border. Queen contiguity defines two areas as neighbors even if they only share a corner. When two areas are considered neighbors, the cell of the respective weight matrix is set to one and zero if two areas are not neighbors. Regarding the degree, there is first-order adjacency or second-order adjacency. First-order adjacency means that two areas have to be adjacent to each other, while second-order adjacency means that the neighbors of a neighbor are also considered (Pfeiffer et al., 2008, p. 34). Although one can also consider third-order, fourth-order, and so on, going too far would result in a weight matrix with all ones.

Which neighbor definition should be used depends on the type of spatial data given. If point data is given, distances can be easily calculated. If only areas are given, it is easier to use adjacency. Although it is possible to use the centroids of the regions to calculate distances, centroids of oddly-shaped polygons might give the impression that two areas are far apart, even though they share a border.<sup>13</sup>

Moran’s I is usually between -1 and 1, but it is not restricted to these bounds, especially when the product of the difference from the mean is high and heavily weighted (Waller and Gotway, 2004, p. 228). Furthermore, high values of Moran’s I do not necessarily mean that there is a spatial pattern. Waller and Gotway (2004, p. 229) state that: “Under the constant risk hypothesis, regions with higher-than (overall)-average population sizes will tend to have higher-than-average observed counts, elevating the value of Moran’s I”. In social science, variables tend to be slightly positively correlated, e.g., demographic and socioeconomic characteristics, and negative correlations are rare (Griffith, 2003, p. 5).

The significance test for Moran’s I depends on the assumption of the underlying distribution. There are two options, the normality assumption and not making any assumption about the distribution, which is referred to as “randomization” (Cliff and Ord, 1981, p. 14). Neither of these affects the calculation of Moran’s I or the expected value. But they do affect the calculation of the variance and thus the test statistic (Cliff and Ord, 1981, p. 21). However, the authors point out that as  $n$  increases, the distribution is approximately normal (Cliff and Ord, 1981, pp. 46, 51). Alternatively, the significance can be tested using Monte Carlo test (Cliff and Ord, 1981, p. 63).

For the Moran’s I test under the normality assumption, the expected value is given by (Dubé and Legros, 2014, p. 70):

$$E(I) = -\frac{1}{N-1} \quad (4.18)$$

<sup>12</sup>The used R package *ape* (Paradis et al., 2019) already automatically row-standardizes input weight matrices.

<sup>13</sup>A more in-depth explanation is given by Pfeiffer et al. (2008).

with  $N$  the total number of observations, and the variance is given using the following formula (Dubé and Legros, 2014, p. 70):

$$Var(I) = \frac{N^2 S_1 - N S_2 + 3S_0^2}{S_0^2(N^2 - 1)} - (E(I))^2 \quad (4.19)$$

with

$$S_0 = \sum_{i=1}^N \sum_{j=1}^N w_{ij} \quad (4.20)$$

$$S_1 = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (w_{ij} + w_{ji})^2 \quad (4.21)$$

$$S_2 = \sum_{i=1}^N (w_{i.} w_{.i})^2 \quad (4.22)$$

$$w_{i.} = \sum_{j=1}^N w_{ij} \quad (4.23)$$

$$w_{.i} = \sum_{j=1}^N w_{ji} \quad (4.24)$$

The formula for the test statistic is given by:

$$t = \frac{I - E(I)}{\sqrt{Var(I)}} \quad (4.25)$$

under the null hypothesis of the absence of spatial autocorrelation (Dubé and Legros, 2014, p. 70).

Moran's I is also named global Moran's I because in 1995 Anselin introduced a local Moran's I, which is used to identify points whose value of the variable of interest is above or below the values of surrounding points (Anselin, 1995). In other words, local Moran's I identifies hotspots and cold spots because spatial autocorrelation is not necessarily distributed over the entire region but rather locally.

#### 4.1.5. Combination of Utility Measures

To locate each masking method in a risk-utility map, one value for utility is needed. Combining the results into one value proves not to be straightforward. One approach in the literature is, even if various dimensions of utility are calculated, to only evaluate them separately, which does not allow drawing a risk-utility map (see, e.g., Seidl, Paulus, et al., 2015; Armstrong et al., 1999). Another approach is to use the percent root mean square error (percent RMSE) as proposed by Clifton and Gehrke (2013). The square of the RMSE, the mean squared error (MSE), is also used in the field of social research as a measure of the Total Survey Error (Biemer, 2010, pp. 825–826). The MSE can be decomposed into the squared bias (difference between an estimate  $\hat{\theta}$  and the parameter  $\theta$  it estimates) and variance (Biemer, 2010, p. 826).

$$\text{MSE}(\hat{\theta}) = \text{B}^2(\hat{\theta}) + \text{Var}(\hat{\theta}) \quad (4.26)$$

Clifton and Gehrke (2013) used additional information such as jobs per acre for the calculation. Another possibility would be to use the mean distance between points.<sup>14</sup>

Kounadi and Leitner (2015) defined two indices, inspired by the idea of the RMSE, based on descriptive statistics (global measure) and the identification of hotspots (local measure) using clustering and spatial autocorrelation. The global measure (GDi) is calculated as the arithmetic mean of the Mdi, Odi and MAdi,<sup>15</sup> with (Kounadi and Leitner, 2015, pp. 745–746):

$$\text{Mdi} = \frac{\text{distance original mean to masked mean}}{\text{distance of original mean to the farthest point away in study area}} \times 100 \quad (4.27)$$

$$\text{Odi} = \frac{|\text{orientation of original ellipse} - \text{orientation of masked ellipse}|}{180} \times 100 \quad (4.28)$$

and if

$$\text{original major axis} \leq \frac{\text{maximum major axis}}{2} \quad (4.29)$$

$$\text{MAdi} = \frac{|\text{masked major axis} - \text{original major axis}|}{\text{maximum major axis} - \text{original major axis}} \times 100 \quad (4.30)$$

<sup>14</sup>Unless stated otherwise, the words “average” and “mean” are used for the arithmetic mean.

<sup>15</sup>Kounadi and Leitner (2015, p. 744) describe the GDi and LDi as a “equally weighted composite indicator” of the indices used. Seidl, Jankowski, and Clarke (2018, p. 289) more clearly state that the indices are averaged.

else

$$\text{MAAdi} = \frac{|\text{masked major axis} - \text{original major axis}|}{\text{original major axis}} \times 100 \quad (4.31)$$

All measures follow one simple idea: The difference between a dimension of the masked and the original data set is considered in relation to the maximum possible difference to the original data set e.g., for the major axis, the difference of the original major axis to the maximum major axis is used. The maximum major axis is found by moving the points to the outermost edges of the study area where the points are the furthest from each other (Kounadi and Leitner, 2015, p. 746).

For the local measure (LDi) based on hotspot identification, the nearest neighbor hierarchical spatial clustering and the Getis-Ord  $G_i^*$  statistic are averaged. As explained in more detail in subsection 4.1.3, hierarchical clustering has many drawbacks and therefore was not used. Instead, the density-based clustering algorithm DBSCAN was used (Ester et al., 1996).

The Getis-Ord  $G_i^*$  statistic is a local spatial autocorrelation measure used to identify so-called “hotspots”. The problem with this local measure is that the given data set does not provide additional information at the coordinate level, which is needed for the calculation. The variables used, as described in chapter 2, are based on the smallest available regional level, which includes between 200 and 800 people. Therefore, points belonging to the same area will receive the same value and unintentionally form a hotspot. Moreover, the local spatial autocorrelation measure does not provide one summarized value that can be easily compared between different masking methods. Therefore, the global measure (Moran’s I) was used here.

To yield one index for spatial autocorrelation and one for clustering, Kounadi and Leitner (2015, p. 747) proposed calculating the symmetric difference between the areas of the masked hotspots and the original hotspots and divide this by the sum of the areas of both sets of hotspots and multiply by 100 (Kounadi and Leitner, 2015, p. 747), i.e.:

$$\text{Index} = \frac{\text{symmetric difference of A and B}}{A + B} \times 100 \quad (4.32)$$

with  $A$  the original hotspots and  $B$  the masked hotspots, and the symmetric difference being the areas of the hotspots that do not overlap between the original and masked data set. However, this is not calculable for some masking methods that provide only a distance matrix.

A key advantage of the chosen clustering algorithm is that outliers are considered noise points and will not be clustered. This allows comparing two properties: (1) the number of clusters, (2) whether points that are non-clustered in the original data set remain non-clustered and whether clustered points remained clustered. To compare the number of clusters, the difference between the number of clusters of the original

data set and the masked data set is divided by the difference of the original number of clusters to the maximum number of clusters, and this is multiplied by 100.

$$\text{ClusNum} = \frac{|\text{original cluster number} - \text{masked cluster number}|}{(n/\text{minimum cluster size}) - \text{original cluster number}} \times 100 \quad (4.33)$$

For DBSCAN (Ester et al., 1996), the maximum number of clusters can be evaluated by dividing the number of points ( $n$ ) by the minimum cluster size (MinPts).

For evaluating whether points remained clustered (or non-clustered), the symmetric difference is used as Kounadi and Leitner (2015) proposed, but not regarding the areas the points cover, but the points assigned to a cluster.<sup>16</sup> The symmetric difference is calculated by finding the number of points that were clustered in the original data set and are non-clustered in the masked data set and vice versa. This is divided by the maximum difference and multiplied by 100.

$$\text{ClusNoise} = \frac{|\text{orig. points} \setminus \text{masked points}| \cup |\text{masked points} \setminus \text{orig. points}|}{10000} \times 100 \quad (4.34)$$

The maximum difference would be reached if all non-clustered points in the original data set are clustered, and all clustered points are considered non-clustered in the masked data set, i.e., a maximum of 10,000. The two measures for clustering are averaged before combining the results with the spatial autocorrelation to give equal weights to spatial autocorrelation and clustering results.

For spatial autocorrelation, the maximum value can be -1 or 1 for maximum dispersion or maximum clustering. Thus, the maximum difference depends on whether the original value is positive or negative. If positive, the maximum difference is  $| -1 - \text{original value} |$ , if negative the maximum difference is  $1 - \text{original value}$ . The spatial autocorrelation was measured for two different variables (proportion of single households, proportion of full-time working people). Therefore, the measure (SpatAut-Corr) for both is calculated and averaged before then being averaged with the measure for clustering to give equal weights to spatial autocorrelation and clustering results. The local measure (LDi) is the arithmetic mean of Clus and SpatAutCorr, with:

$$\text{Clus} = (\text{ClusNum} + \text{ClusNoise})/2 \quad (4.35)$$

<sup>16</sup>Similarly, Houfak-Khoufak and Touya (2020, p. 5) evaluated the similarity between clusters by comparing if clusters  $A$  and  $B$  contained the same address points. For the comparison, he used the Jaccard similarity  $\left( \text{Jaccard}(A, B) = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \right)$ , so that the maximum value (one) indicates that clusters are the same, as opposed to Kounadi and Leitner (2015) who used dissimilarity.

and

$$\text{SpatAutCorr} = \frac{|\text{original Moran's I} - \text{masked Moran's I}|}{\text{maximum difference to original Moran's I}} \times 100 \quad (4.36)$$

GDi and LDi range from zero to 100. Zero corresponding to the masked data set yielding the same results as the original data and 100 corresponds to the expected largest difference of the masked results to the original results (Kounadi and Leitner, 2015, p. 744). To yield one value for the risk-utility map, GDi and LDi are combined using the average.

It should be noted that the GDi and LDi only consider the distance between points in spatial autocorrelation and the clustering algorithm, but not as a measure on its own. Therefore, for the risk-utility-map, two approaches were applied, the average of the GDi and LDi, and the MSE, to show the difference between considering only distance as a relevant utility measure or considering all other dimensions (GDi and LDi). For the MSE, the mean distance between points is used, i.e., the squared difference between the original and masked mean distance is calculated, adding the variance of the mean distances between points of the multiple applications.

## 4.2. Risk Evaluation

Assessing the risk of re-identification has hardly been performed using multiple risk measures. Usually, the  $k$ -anonymity is evaluated. The idea is that there must be at least  $k - 1$  other records with the same attributes so that they cannot be distinguished from each other (Sweeney, 2002, p. 564; Samarati, 2001). In official statistics, for example, a  $k = 3$  is considered sufficient, so that there are always at least three people with the same characteristic (see, e.g., Rothe, 2015, p. 299).

Spatial  $k$ -anonymity can be defined in various ways. First, it can be defined as the number of residents within distance  $d$  from the point considered (Seidl, Jankowski, and Clarke, 2018, p. 284). However, this imposes the question of defining the value of  $d$ . Similarly, as a second option, it can be defined as the average distance to  $k$ -nearest neighbors (Seidl, Paulus, et al., 2015, p. 255; Seidl, Jankowski, and Clarke, 2018, p. 284). Lastly, it can be defined as the number of residents closer to the point considered than the masked point (Allshouse et al., 2010, p. 446; Hampton et al., 2010, p. 1064; Broen et al., 2021, p. 6). The first and the second option say more about the distance than  $k$ -anonymity. The last option is already taken into account in some of the masking methods, such as  $k$ -nearest neighbor donut masking (Hampton et al., 2010) and street masking (Swanlund, Schuurman, Zandbergen, et al., 2020). As will be shown in this thesis, this does not guarantee that the risk of re-identification is low.

Moreover, similarly to  $k$ -anonymity, authors such as Armstrong et al. (1999, p. 516) proposed considering the distance of the masked coordinates to their original co-

ordinates compared to distances to other points. More specifically, they used the proportion of masked points which are closer to the original location than any other point in the unmasked data set.<sup>17</sup>

Another attempt was made by Gao et al. (2019) who used the DBSCAN algorithm (Ester et al., 1996) to identify home and work cluster points and if the masking method prevented the algorithm from identifying home and work cluster points correctly. However, this is not a risk measure that will be considered here due to the fact that it does not attempt to identify the residential locations of the respondents correctly.

Attempts to identify the original location by linking the records of the masked and original file are common in the field of record linkage. As will be shown, such an approach is necessary because they prove that masking methods can satisfy (spatial)  $k$ -anonymity but are unable to withstand these attacks.<sup>18</sup> Typically, an intruder is assumed to have an *identification file* with direct identifiers, in this case, unmasked geographic coordinates, as well as *key variables* that are also released with the masked coordinates (e.g., Duncan, Elliot, et al., 2011, p. 30). Once the records are matched, it is evaluated how many of the matches are correct. In literature, this is termed (Fawcett, 2003, p. 2):

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad \text{and} \quad \text{recall} = \frac{\text{TP}}{\text{P}} \quad (4.37)$$

calculated by comparing the true positives (TP), false positives (FP), as well as the sum of the true positives and false negatives (P). Recall provides the answer to the question of how many records were correctly identified, while precision answers the question about the proportion of identified records that are actually correct. In record linkage, false positives are of great interest because it is important to know whether two records are incorrectly matched. In the case of geographic masking, even false matches can pose a threat because an intruder thought it might be possible to re-identify the sampled people (Elliot and Dale, 1999; Elliot, Mackey, et al., 2016). Further, falsely matched points pose the threat that the data intruder will associate false characteristics with respondents (Seidl, Jankowski, and Clarke, 2018, p. 281).

In this thesis, the approach of linking the data set with an identification file was chosen (as opposed to spatial  $k$ -anonymity) to evaluate the re-identification risk. Simply trying to find the overlap between those data sets will not be successful due to the applied masking methods. Therefore, methods are needed to find an overlap between data sets even if there is some perturbation. The used methods in this thesis for evaluating the risk were reversing masking methods (Armstrong et al., 1999)

<sup>17</sup>This measure is based on Spruill's measure (Spruill, 1982).

<sup>18</sup>In the following, "attack method" and "risk measure" are used as a synonym for "re-identification method".

and mean of multiple releases (Zimmerman and Pavlik, 2008; Cassa, Wieland, et al., 2008) and for linking the data set with an identification file the minimum distance, Hungarian algorithm (Kuhn, 1955), graph theoretic linkage attack (Kroll, 2015; Kroll and Schnell, 2016), and graph matching attack on privacy-preserving record linkage (Vidanage et al., 2020) were used.

The scenario considered for the re-identification methods is described below. Afterward, each method used to assess the re-identification risk is explained in detail.

#### 4.2.1. Scenario

An example of a scenario for risk analyses is given by Kroll (2015) and Kroll and Schnell (2016). In their scenario, there are two files (one with correct coordinates and one with masked coordinates) and an overlap of 10%. Therefore, not all coordinates that were masked are in the identification file. Also, additional variables that help to eliminate possible matches are given. However,  $k$ -anonymity is met. This type of scenario is also considered here. The masked file contains as many records as the identification file, and the overlap is 10%.<sup>19</sup>

Simply trying to find matching coordinates between two data sets will not be successful due to the displacements of the coordinates. Therefore, methods were found that, in theory, should find the correct overlap between the two data sets even though coordinates have been displaced in one of the files. The methods tested are minimum distance, Hungarian algorithm (Kuhn, 1955), graph theoretic linkage attack (Kroll, 2015; Kroll and Schnell, 2016), and graph matching attack on privacy-preserving record linkage (Vidanage et al., 2020). In addition, two methods were tested usually used for identification of the original location when testing geomasking methods, namely reversing masking methods (Armstrong et al., 1999) and mean of multiple releases (Zimmerman and Pavlik, 2008; Cassa, Wieland, et al., 2008).

Since some attack methods, such as the graph theoretic linkage attack (Kroll, 2015; Kroll and Schnell, 2016), are computationally complex, it is necessary to draw a subsample of the given sample and test the attack method only on a data set with fewer records. Therefore, a sample of  $n = 1,000$  with an overlap of  $n_{overlap} = 100$  and a sample of  $n = 2,000$  with an overlap of  $n_{overlap} = 200$  were drawn. Further, additional variables were included (sex, age, and employment status) to limit the number of possible matches. Regardless of the sample size, at least three people had the same combination of sex, age, and employment status, so that correct matches cannot be identified based on the additional variables alone.<sup>20</sup> Further, based on Kerckhoffs' principle, it is assumed that the masking method used and the parameter(s) of the

<sup>19</sup>A more detailed description of the data set can be found in section 2.1.

<sup>20</sup>Table C.1 in appendix C provides an overview of the characteristics as well as the number of people in the masked and identification file with the respective characteristic. More information on the data set characteristics and how the variables were chosen can be found in section 2.1.

masking method can be known without severe privacy concerns (Petitcolas, 2011).<sup>21</sup>

### 4.2.2. Reversing Masking Methods

In the first paper summarizing geomasking methods, the authors pointed out that the affine transformation methods can be easily reversed to find the original location, by trying all possible parameters (Armstrong et al., 1999, p. 516). For the rotation masking method around the coordinate system's origin, every angle between 1 and 360 should be applied and then checked if there are overlapping points between the two data sets. Since the conversion may cause minor changes in the coordinate, the coordinates of the masked file and the identification file were rounded to the nearest fourth decimal place. To shorten the runtime, as soon as an angle resulted in an overlap, this angle was chosen as the correct angle, and the overlapping coordinates were set as matches.

The rotation masking method around the spatial mean center works the same way. Every angle between 1 and 360 has to be tried until one is found, resulting in overlapping points between the two data sets. However, in this case, the points must be moved so that the spatial mean center corresponds to the origin of the coordinate system. After a rotation by a certain angle, the points have to be moved back by the same distance the coordinates have been moved before so that they are located in the correct area again. When considering the smaller subsamples, the spatial mean center of the full sample must be taken instead of the spatial mean center of the subsample.

For displacement using translation, moving the coordinates in every direction by all possible values for the displacement distance will eventually result in overlapping points of the two data sets. However, this is very time-consuming. A much faster approach is to consider the coordinate axes separately. Due to possible minor displacements of the coordinates when converting them, the coordinates are rounded to the fourth decimal place. Based on the coordinates, the two data sets are merged to find overlapping points. For smaller data sets with unique coordinates, it may be sufficient to consider only part of the coordinates (e.g., only the easting). Further, the interval searched can be shortened by knowledge about the parameter choices (interval from which random number is drawn). Even if the interval of the random integer is not known, the differences in the spatial mean centers and spatial median centers of the two data sets can be used as guidance. Also, plotting both data sets can be helpful to determine the range of values to be tested.

Change of scale can be reversed using a similar approach as for displacement using translation. However, instead of adding or subtracting the respective value, it is multiplied. Again, if the interval for the random number is unknown, it can be approximated by plotting the two data sets' coordinates.

Another possible attack method for affine transformations would be to use the

---

<sup>21</sup>The original paper from Kerckhoffs (1883) was written in French.

Procrustes analysis. Given two matrices, the basic idea is that one matrix can be translated, rotated, and scaled to yield the other (see, e.g., Mardia et al., 1995; Schnell, 1994). However, the method assumes that the order of the records is preserved so that this method will not be very successful for data sets whose matching entries have different row-indices and for data sets for which the overlap is not 100%. Therefore, this method was not used as an attack method. However, it was used to calculate descriptive statistics for those masking methods, which only release a distance matrix (see section 4.3 for more details).

### 4.2.3. Mean of Multiple Releases

One idea of Zimmerman and Pavlik (2008) as well as Cassa, Wieland, et al. (2008) is to use multiple releases of data sets to obtain the position of the original coordinates. The idea is based on the central limit theorem, which states that as the number of observations increases, here the number of masked coordinates, the average of the coordinates will fall closer to the original point (Cassa, Wieland, et al., 2008, pp. 3–5). With regard to the scenario given, the expectation would be that the mean of the coordinates of multiple releases is less than one meter away from the original location. This re-identification method does not allow points to be classified as not matched because it does not use an identification file. Therefore, in chapter 5.3 only the percent of correctly identified matches is reported.

However, this attack can easily be prevented by masking the data set only once and then making the same data set available to every scientist instead of masking the coordinates every time the data set is requested. Moreover, for displacements that do not consider a random perturbation and masking methods that will always displace coordinates to the same location, the assumptions are not met, and the method's inefficiency becomes more apparent.

### 4.2.4. Minimum Distance

Another intuitive approach is to calculate the distances between each point of the masked and the identification file and define the points with the smallest distance as matching.<sup>22</sup> A disadvantage of this method is that all points of the masked data will be assigned a match of the identification file, and thus the precision will be small due to a large number of false positives. Another disadvantage of this method is that the same point of the identification file might be assigned to multiple points of the masked file. Thus, there is no one-to-one correspondence. Since the intruder cannot know which match is correct, these are removed, and only the one-to-one matches are considered. This, in turn, reduces the number of correctly identified points.

<sup>22</sup>Similarly, Seidl, Jankowski, and Clarke (2018) has criticized that it might be assumed that the closest residential address to a masked point will be associated with the information of the record of the data set.

Therefore, a method is needed that only allows a one-to-one correspondence. Since an assignment based on the individual minimum distance is not possible, the overall minimum distance should be found. This problem is known in the literature as assignment problem.

#### 4.2.5. Assignment Problem: Hungarian Algorithm

Intuitively a data intruder will take the locations with the smallest distance between them as a possible match (see, e.g., Seidl, Jankowski, and Clarke, 2018). However, in some cases, two locations of the masked data set may have the same nearest neighbor in the identification data set or vice versa, which again is not very helpful to determine whether two points correspond to the same data entry. Therefore, the idea is to find a one-to-one correspondence between two data sets based not on the minimum distance between individual points but the overall minimum distance.

In mathematics, this problem is known as *assignment problem*. Typically, the problem is stated as assigning workers to jobs with varying qualifications of each worker for each job (Kuhn, 1955, pp. 83–84). The goal is to assign each worker exactly one job optimally. There are several solutions to this problem. One, very common in the literature, is the Hungarian method (Kuhn, 1955).<sup>23</sup> To run the algorithm, a matrix (here: distance matrix) with the distance between each point of one data set to each point of the other data set is needed. Usually, both data sets have the same number of rows (balanced). However, the Hungarian method can also handle unbalanced assignment problems, as will be explained.

To find the solution with the smallest overall costs (here lowest overall distance), the algorithm performs the following steps based on the distance matrix (see, for example Hardwick, 1996, pp. 127–129). First, the minimum of each row is subtracted from its respective row (see example below, illustrated in table 4.2). This is repeated for the columns as well, resulting in a matrix containing some zeros. In a second step, horizontal and vertical lines are drawn across columns and rows containing at least one zero, with the goal of using as few lines as possible. If the minimum number of lines is smaller than the number of rows in the data set, additional steps must be taken. For the entire matrix, the lowest number not covered by a line is subtracted from each number not covered by a line. Additionally, for numbers that are covered by two lines, the smallest number is added to the respective number.

Then again, horizontal and vertical lines are drawn across rows and columns containing zeros with as few lines as possible. These two steps (subtracting or adding the smallest number and drawing lines) are repeated until the smallest number of lines needed to cover zeros is equal to the number of rows in the data set. Lastly, the

---

<sup>23</sup>This is also sometimes referred to as the Kuhn-Munkres-algorithm. Kuhn proposed the term “Hungarian method” because the idea was based on earlier work by two Hungarian men D. König and E. Egerváry. In literature, also the name “Hungarian algorithm” is common (see, e.g., Hardwick, 1996), which is the term used here.

zeros indicate the one-to-one correspondence between the two data sets. Note that there is a possibility of multiple solutions due to multiple zeros in a row/column.<sup>24</sup>

As an example, taken from Hardwick (1996), five jobs should be assigned to five people who take different amounts of time to complete each job (see table 4.1). If only the minimum times for each person's job are taken, job two would be assigned to person one, three, four, and five. Job three would be assigned to person two, and jobs one, four, and five would not be assigned. The Hungarian method achieves a one-to-one correspondence where each person takes on a job, and each job is assigned.

Table 4.1.: Example of Hungarian algorithm (Hardwick, 1996, p. 127).

	J1	J2	J3	J4	J5
P1	13	8	12	21	14
P2	17	23	10	16	18
P3	14	13	15	15	16
P4	17	8	11	16	14
P5	12	7	15	20	11

First, the minima 8, 10, 13, 8, and 7 are subtracted from rows one to five (see table 4.2 a). Then the column minima are subtracted from each element of the respective column. For jobs two and three, the minimum is zero; thus, nothing changes (b). The minimum number of lines to cover the zeros is three (indicated by the arrows in (c)). Since the minimum number of lines is less than the number of rows/columns, the uncovered minimum (P5,J5: 1) is subtracted from the other uncovered numbers and added to numbers covered by two lines (P2,J2: 13; P3,J2: 0). The resulting matrix (d) requires a minimum of four lines to cover all zeros. Since this is again smaller than the number of rows/columns, the uncovered minimum number must be subtracted respectively added (P1,J1: 3; P5,J1: 3). The final result is shown in table 4.2 (e). The minimum number of lines required to cover all zeros is five. Taking the zeros as matches and assigning only one job to one person results in the following assignments: P1→J1, P2→J3, P3→J4, P4→J2, P5→J5. Based on the original times from table 4.1, the overall minimum is 13+10+15+8+11=57 minutes.

The Hungarian algorithm will assign each person in file A to a person in file B. However, as stated in the scenario, there is just a 10% overlap between the two files. Therefore, this approach will automatically result in a large number of false positives. As a solution, criteria can be defined, which decide in a second step whether the matches found are really considered as matches. The criterion chosen is the distance between the potential matches. So, the distances between the matches are calculated and sorted. Only the first  $n$  matches with the shortest distance are considered, where  $n$  is the number of matches of the known overlap.<sup>25</sup>

<sup>24</sup>Existing implementations select the solution differently, so different results (one-to-one correspondences) may be obtained depending on the implementation.

<sup>25</sup>Even if the overlap is not known, it can be assumed and set accordingly.

Table 4.2.: Calculation example of Hungarian algorithm (Hardwick, 1996, pp. 127–129).

a)	J1	J2	J3	J4	J5		b)	J1	J2	J3	J4	J5	
P1	5	0	4	13	6	(-8)	P1	4	0	4	11	3	
P2	7	13	0	6	8	(-10)	P2	6	13	0	4	5	
P3	1	0	2	2	3	(-13)	P3	0	0	2	0	0	
P4	9	0	3	8	6	(-8)	P4	8	0	3	6	3	
P5	5	0	8	13	4	(-7)	P5	4	0	8	11	1	
								(-1)			(-2)	(-3)	
c)	J1	J2	J3	J4	J5		d)	J1	J2	J3	J4	J5	
P1	4	0	4	11	3		P1	3	0	3	10	2	
P2	6	13	0	4	5	←	P2	6	14	0	4	5	
P3	0	0	2	0	0	←	P3	0	1	2	0	0	←
P4	8	0	3	6	3		P4	7	0	2	5	2	
P5	4	0	8	11	1		P5	3	0	7	10	0	
		↑							↑	↑		↑	
e)	J1	J2	J3	J4	J5		solution:						
P1	<b>0</b>	0	3	7	2	←	P1→J1						
P2	3	14	<b>0</b>	1	5	←	P2→J3						
P3	0	4	5	<b>0</b>	3	←	P3→J4						
P4	4	<b>0</b>	2	2	2	←	P4→J2						
P5	0	0	7	7	<b>0</b>	←	P5→J5						

Another approach to improve and simultaneously speed up this method is to consider only people with the same characteristics when running the Hungarian algorithm. Although not originally defined for application to different sized data sets, implementations for unequal sized data sets can be found (referred to as *rectangular assignment problem*). For example, a rectangular assignment problem can be solved by extending the matrix into a square matrix by adding dummy variables (Burkard et al., 2012, p. 165). Thus, in this thesis, the Hungarian algorithm with and without considering additional characteristics about people will be considered.

#### 4.2.6. Graph Theoretic Linkage Attack

A rather recently published attack method is from Kroll (2015). It is also explained in Kroll and Schnell (2016) with a corresponding masking method.<sup>26</sup> The attack makes use of graph theory and formulates the problem of linking masked coordinates

<sup>26</sup>The following description follows Kroll (2015) and Kroll (2014a), which is more mathematical but also more detailed. Another version of this article can be found in Kroll (2014b). Kroll and Schnell (2016) give a shorter overview.

with their original counterpart as a maximum clique problem of the product graph of the two data sets. The attack assumes that the intruder has an identification file with the direct identifiers. The intruder can also calculate the distances between the coordinates given in the identification file (Kroll, 2015, p. 7). Also, the masking method and the parameters are known.

Given the file with the masked coordinates and an identification file with the original coordinates, both can be represented by an undirected graph.

$$G_1 = (V, E, \lambda_V, \omega_E) \quad \text{and} \quad G_2 = (W, F, \lambda_W, \omega_F) \quad (4.38)$$

with  $V$  and  $W$  the vertices sets,  $E$  and  $F$  the edge sets,  $\lambda_V$ , and  $\lambda_W$  the labels of the vertices, and  $\omega_E$  and  $\omega_F$  the weights of the edges. An apparent strategy is to use other variables (labels of the vertices) to find matching records, for example, sex and age. Considering  $k$ -anonymity, this leads to ties meaning that several points of the masked file are possible matches for several points in the identification file. Therefore, Kroll (2015) suggests using the edge weights as additional information.

Given the two graphs, a common subgraph between these is given by  $S \subseteq V$  and  $T \subseteq W$ , and a bijection  $\varphi : S \rightarrow T$  with the following characteristics (Kroll, 2014a, p. 8; Kroll, 2015, pp. 224–225):

$$\begin{aligned} (i) \quad & \lambda_V(s) = \lambda_W(\varphi(s)) \quad \text{for all } s \in S \\ (ii) \quad & \text{For all } s_1, s_2 \in S \text{ we have either} \\ & (a) \quad s_1 s_2 \in E, \varphi(s_1)\varphi(s_2) \in F \quad \text{and} \quad \omega_E(s_1 s_2) \approx \omega_F(\varphi(s_1)\varphi(s_2)) \quad \text{or} \\ & (b) \quad s_1 s_2 \notin E \quad \text{and} \quad \varphi(s_1)\varphi(s_2) \notin F \end{aligned} \quad (4.39)$$

In words, the approximate common subgraphs have the same vertex labels. Either the edge between two points of the first graph also exists in the corresponding points of the other graph, and the weights of the edges must be approximately equal. Or there is no edge between two points and thus no edge between the corresponding points in the graph of the other file (Kroll, 2014a, p. 8; Kroll, 2015, pp. 224–225).

To evaluate whether points are matches or not, the idea is to take the product graph of the two given graphs with the vertex set and the edge set be defined as following (Kroll, 2014a, p. 9; Kroll, 2015, p. 225):

$$\begin{aligned} V_{\otimes} &= \{(v, w) \in V \times W : \lambda_V(v) = \lambda_W(w)\} \quad \text{and} \\ E_{\otimes} &= \{ \{(v_1, w_1), (v_2, w_2)\} : v_1 \neq v_2, w_2 \neq w_1 \text{ and either} \\ & (a) \quad v_1 v_2 \in E, w_1 w_2 \in F \quad \text{and} \quad \omega_E(v_1 v_2) \approx \omega_F(w_1 w_2) \quad \text{or} \\ & (b) \quad v_1 v_2 \notin E \quad \text{and} \quad w_1 w_2 \notin F \} \end{aligned} \quad (4.40)$$

The vertices consist of the combination of the vertices of the first graph with the second graph for all vertices with the same vertex label. For this product graph, two vertices are connected by an edge if the edge existed in both the first and the second graph, and the weights are approximately equal. If the vertices were not connected by an edge in the first graph, they are not connected in the second graph and are not connected in the product graph (Kroll, 2014a, p. 9; Kroll, 2015, p. 225). Thus, the number of possible matches is not only limited by common vertex labels but also by approximate weights.

Approximate weights are defined as the difference between the edge weights being smaller than a threshold  $\varepsilon$  (Kroll, 2015, p. 10). Given the product graph, the maximum clique graph is found, and the maximum clique graph's vertices are considered as matches. The clique in a graph is a complete subgraph of an undirected graph (see, e.g. Valiente, 2002, p. 299). The maximum clique is defined as the complete subgraph of a graph with the largest number of vertices (see, e.g. Valiente, 2002, pp. 299–300).<sup>27</sup>

Further, Kroll (2015) provides guidelines to define when the weights can be considered approximately equal. The weights of the edges are the distances between the coordinates, and the author proposed using empirical quantiles. Thus, two edges are approximately equal if the difference in deviation of the distances is:

$$\frac{1 - \alpha}{2}\text{-quantile} < d - d' < \frac{1 + \alpha}{2}\text{-quantile} \quad (4.41)$$

with  $\alpha \in (0, 1)$  a threshold parameter, defining the probability that a common edge is detected (Kroll, 2015, p. 231). The empirical quantiles are obtained by sampling some points of the area of interest and simulating the distribution of the deviation of the distances (Kroll, 2015, p. 230). The threshold parameter should be chosen carefully because as  $\alpha$  increases, the overlap increases, but for too large values of  $\alpha$ , the precision decreases.

Another way to define the weights as approximately equal could be based on the masking methods' properties. For example, if two points are masked using random perturbation within a circle, the distances can differ at most by the sum of each point's radius (see figure 4.5). As can be seen in figure 4.5 the distance between points (indicated by the black line) can increase at most by  $r_1 + r_2$  (red line) and decrease by  $-(r_1 + r_2)$  (blue line). Even if the radius is based on local information such as population density, the masked coordinate can be used to assign the population density (as done in this thesis), which in most cases should be similar to the population density originally used. Then two edge weights are equal if the distance differs by less than the radius of point A plus the radius of point B. This train of thought can be repeated for each masking method.

<sup>27</sup>To find the maximum clique graph, the maximum clique algorithm as implemented by Konc and Janežič (2007) was used, as proposed by Kroll (2015). The implementation of the algorithm can be found at the respective website <http://insilab.org/maxclique/>.

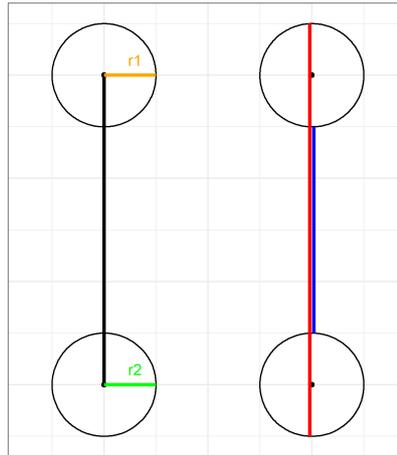


Figure 4.5.: Definition of edge weights: Two points are moved within the radius of  $r$ . Red line shows maximum possible distance between moved points. Blue line indicates smallest possible distance between moved points.

A major shortcoming of the attack is that it is computationally complex. Graph matching and finding the maximum clique are NP-hard problems (Kroll, 2015, p. 238). The maximum file size Kroll (2015) was able to work with was  $N_1 = N_2 = 2,000$  with an overlap of  $n = 200$ . More precisely, the maximum product graph he was able to work with consisted of 30,000 nodes and 44,994,803 edges (Kroll, 2015, p. 236). Because of the much larger sample size considered in this thesis, only a subsample could be considered for this method.

A sample of  $n = 1,000$  was taken from each file with an overlap of  $n = 100$ , yielding a product graph with 14,352 vertices. Increasing by 1,000 people ( $n = 2,000$ ) yields a product graph with 105,144 vertices. If all  $n = 10,000$  are considered, the result is a product graph with 2,515,147 vertices.

In the following, for each masking method, a short description is given on how the lower and upper value was defined. Following Kerckhoffs' principle (Petitcolas, 2011), it is assumed that parameter choices for the masking methods are known.

### Random Perturbation Within a Circle, Location Swapping, Verified Neighbor Approach

These masking methods are based on population density which is assigned using easily accessible files. Thus, based on the location of the masked coordinate, the population density is assigned.<sup>28</sup> The sum of the radii (based on the population density) is taken as the lower and upper limits for the difference in distance between points (see also figure 4.5).

<sup>28</sup>If the masked coordinate was displaced to a region in which the population is zero, the number three was assigned to be able to calculate a population density.

### Random Perturbation Using a Uniform Distribution, Official Statistics Grid

With a uniform distribution, the point is moved within a square region instead of a circular region. The size of the square is the difference between the upper and lower bound (range). If half of the range is used as the maximum displacement distance, only part of the square is covered (see figure 4.6 (a) gray area). Another approach would be to consider the distance of the point to the corner of the square as the radius. This distance can be calculated using the Pythagorean formula ( $\sqrt{2r^2}$ ).<sup>29</sup> However, this includes areas outside the area of interest (see gray area in figure 4.6 (b)). Using (a) could lead to false negatives, while (b) could lead to false positives. Therefore, both approaches were considered. The latter approach proved to be the better solution, so only the results using this approach are reported. Similarly, for the official statistics grid the Pythagorean formula is used as well.

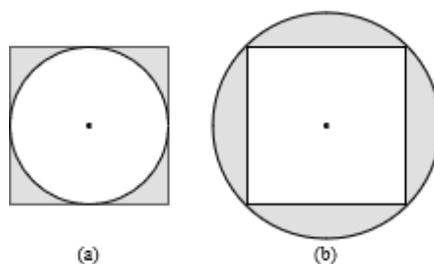


Figure 4.6.: Definition of maximum radius for points masked using a uniform distribution. Maximum radius either defined as half the range (a) or  $\sqrt{2}r$  (b) with  $r$  as half the range.

### Random Perturbation Using a Normal Distribution

Using a normal distribution when relocating points imposes the problem that there is no maximum distance. However, further distances are less likely. As a solution, the maximum distance can be approximated by setting it lower than the distance given a certain probability, similar to the proposed simulation study by Kroll and Schnell (2016). However, in this case, a sample of 10,000 points was drawn from a normal distribution with the respective parameters. Of these, the 90th-percentile was set as the maximum displacement distance.

### Donut Masking

The donut masking method has a maximum and a minimum radius. However, since the minimum radius does not help reduce the allowed difference between the distances of the masked coordinates and the distance of the original coordinate, the same approach as for RPC is considered. For  $k$ -nearest neighbor donut masking the distance to the  $k$ -nearest point based on the masked location was calculated and used as the radius.

<sup>29</sup>Half of the range is the length of the adjacent leg and the length of the opposite leg. The distance of interest is the hypotenuse.

### Street Masking

In street masking, the displacement of points varies based on street density. Again, the lower and upper value is based on an approximation of the maximum displacement distance of each point. The maximum displacement distance is defined by the average of the distances to a predefined number of neighboring nodes of the graph. Therefore, for each point, the distances to the predefined number of nearest neighbors (depth value) are averaged. The average is set as the maximum radius. It is then proceeded as with random perturbation within a circle.

### Voronoi Masking

In Voronoi masking, the displacement distance is not limited to a maximum displacement and does not follow a specific distribution. Instead, in Voronoi masking, the displacement distance depends on the distance to the neighboring points. Furthermore, it will always yield the same result when given the same points as input information. For the simulation study, a sample of  $n = 10,000$  points was drawn from the residential address file.<sup>30</sup> If a smaller sample had been taken, the distance to the nearest points and thus to the nearest border of the Voronoi polygon would potentially be much larger because there would be fewer points. The Voronoi masking method was then applied, and the distances between the original points and the masked points were compared. Then, the lower  $\frac{1-\alpha}{2}$ -quantile and upper  $\frac{1+\alpha}{2}$ -quantile were calculated using  $\alpha = \{0.1, 0.5, 0.75, 0.9\}$  of the differences in distance and were used for the risk method. The highest precision and recall were achieved using  $\alpha = 0.9$ , which is the result reported in the following.<sup>31</sup>

### Adaptive Point Aggregation

APA is also a masking method where the displacement distance is not limited to a maximum displacement and does not follow a specific distribution. Instead, the maximum displacement distance of APA is half the distance of the two furthest points of the region's border. Especially for larger regions, this leads to large potential distances. Using the recommended simulation approach (Kroll, 2015, pp. 230–231) performs poorly in terms of precision and recall (almost zero), due to the strongly varying displacement distances. As an alternative, the polygon variable was also used to limit the number of vertices. As mentioned above, additional variables drastically limit the number of possible matches.<sup>32</sup> Then, either all of the edges of the product graph are considered in the search for the maximum clique, or the difference in distance can further limit it (as needed for APA LGA). When using the small subsample, this

<sup>30</sup>If a residential address file is not available, the identification file could be used, or random points from the region of interest could be drawn. The lower and upper values using the identification file showed only minor differences to the sample drawn from the residential address file.

<sup>31</sup>Results for tested  $\alpha$  not reported are available in the online supplementary material.

<sup>32</sup>Note that  $k$ -anonymity is not given when the polygon variable is used. However, according to the authors, it is information that can be released (Kounadi and Leitner, 2016, p. 61) and thus, it is valid to use it.

approach yields large values in precision and recall. However, with increasing size and an increasing number of records with the same combinations in the selected variables, precision and recall decrease.

Furthermore, for APA using local government areas as regions, the product graph still contains too many edges to find the maximum clique within a reasonable time.<sup>33</sup> Therefore, the number of edges is limited by taking the distribution of the given edge weights and consider only those edge weights larger than the  $\frac{1-\alpha}{2}$ -quantile and smaller than the  $\frac{1+\alpha}{2}$ -quantile with  $\alpha = \{0.1, 0.5\}$ . Larger values of  $\alpha$  again result in too many product graph edges for finding the maximum clique graph. For the local government areas, the best result was achieved using an  $\alpha = 0.5$ , which is the basis for the result reported in the following.

### **Adaptive Random Perturbation**

ARP randomly moves the coordinates within the predefined area. Since the polygons here are of arbitrary shape and can vary drastically, only a simulation study can define a possible displacement distance value. However, the results were poor due to the strongly varying displacement distances, so again, as with APA, the polygons were used as additional information to limit the number of possible matches. For ARP LGA the best result was achieved using  $\alpha = 0.5$ .

### **Random Projection**

In the case of random projection, the output are bit vectors, and although the similarity between bit vectors can be calculated, the geographic distance remains unknown. Even using different models (e.g., linear regression, local polynomial regression) to predict the values in meter given the Jaccard distances results in large differences to the original distance, and too many edges remain in the product graph to find the maximum clique graph. Hence, the graph theoretic linkage attack could not be applied.

### **Anonymization of Distance Matrices via Lipschitz Embedding, Distance Approximation Using ISGP**

For anonymization of distance matrices via Lipschitz embedding and distance approximation using ISGP, only the masked distance matrix is available. The graph theoretic linkage attack can handle distance matrices, but a major problem remaining is the proper definition of the lower and upper difference of distances. As proposed by Kroll (2015, pp. 230–231), a simulation study was performed with a sample of the residential address file of size  $n = 1,000$ .<sup>34</sup> Then, the masking methods were applied (for distance approximation using ISGP, it was assumed that the region is also known), and the masked distances were compared to the true distances of the

<sup>33</sup>For most masking methods, the maximum clique graph could be found in less than a minute. Thus, a reasonable time was set to one hour.

<sup>34</sup>Using a sample of  $n = 10,000$  did not improve the results but increased the execution time drastically.

sample. From the resulting distribution the lower and upper value were defined using the  $\frac{1-\alpha}{2}$ -quantile and the  $\frac{1+\alpha}{2}$ -quantile, as suggested by Kroll (2015, p. 231).

For anonymization of distance matrices via Lipschitz embedding  $\alpha$  was set to  $\alpha = \{0.1, 0.5, 0.75, 0.9\}$ . However, at  $\alpha = 0.9$ , too many edges remain, so a maximum clique is not found within a reasonable time. Also, an  $\alpha = 0.1$  leaves no edges in the product graph, so only  $\alpha = \{0.5, 0.75\}$  were used. The highest precision and recall were achieved using  $\alpha = 0.5$ , which is the basis for the reported results in this thesis.

When distances are approximated using ISGP, an  $\alpha > 0.3$  will result in too many edges to find a maximum clique within a reasonable time. As an alternative, only those differences in distances were considered for which the masked distance was below twice the radius ( $< 80,000$ ). The lower and upper values were then defined as above using  $\alpha = \{0.1, 0.5, 0.75, 0.9\}$ . The highest precision and recall were achieved using  $\alpha = 0.9$ . Thus, the results reported here are based on using  $\alpha = 0.9$ .

### MDAV

For setting the lower and upper values for MDAV also a simulation study had to be conducted. Two approaches were tested. First, a sample of the residential address file of size  $n = 1,000$  was used, the masking method applied, and distances compared. The lower and upper values were then defined using  $\alpha = \{0.1, 0.25, 0.5, 0.75, 0.9\}$ . For all cluster sizes (3, 25, 50), an  $\alpha = 0.9$  leaves too many edges of the product graph to obtain the maximum clique graph within a reasonable time. Therefore, for cluster size 3 an  $\alpha = \{0.1, 0.25, 0.5, 0.75\}$  were tested. For cluster size 25 and 50 only  $\alpha = \{0.1, 0.25\}$  are tested.

Due to the large variation in differences in distances, especially for larger values of  $k$ , as a second approach, multiple samples of the residential address file of size  $n = 10,000$  were used. Again, the masking method was applied, and distances were compared. The lower and upper values were defined using  $\alpha = \{0.1, 0.25, 0.5, 0.75, 0.9\}$  and the results of the multiple samples for the lower and upper values were averaged. While for cluster size three, all lower and upper values could be used, increasing the cluster size to 25 allows only to use  $\alpha = \{0.1, 0.25, 0.5, 0.75\}$ . For cluster size 50, a maximum clique could not be found in a reasonable time.

The best result for cluster size three was achieved with the second approach using  $\alpha = 0.9$ . For the other two cluster sizes (25 and 50), the best results were achieved with the first approach using  $\alpha = 0.25$ .

#### 4.2.7. Graph Matching Attack on Privacy-Preserving Record Linkage

The latest attack method applicable to geomasking is located in the more general field of record linkage. Vidanage et al. (2020) propose to use graph matching procedures and limit the number of possible matches using various record linkage techniques. The advantage of this method over the others is that it allows distance matrices as input

and can also be applied to masking methods that always yield the same result (e.g., Voronoi masking).

The attack method is designed as follows (Vidanage et al., 2020, pp. 1488–1491): A graph is created for both the encoded as well as the non-encoded data set. The nodes are the respective information, and the edges are a similarity measure such as the Dice coefficient. Only nodes that have a similarity above a predefined threshold ( $s_m$ ) are connected by an edge. As an additional but optional step, the similarity values of the edges of the non-encoded graph are adjusted using polynomial regression. To a sample of the similarities of the non-encoded data set, the respective masking method is applied using the known parameter. Polynomial regression is applied to find the best model between the non-encoded similarities and the encoded similarities. Using the resulting model, the non-encoded similarity measures are replaced by their predicted values (Vidanage et al., 2020, p. 1493).

For each graph, the procedure for limiting the number of options is the same. First, features are defined for each node with a connection to at least a predefined minimum number of nodes ( $c_m$ ). The authors propose to use features that can be categorized as node-based features, edge-based features, and structural features (Vidanage et al., 2020, p. 1489). The node-based features are the number of records with the same information as the node, the length of the node (since applied to q-grams, it is the length of the q-gram set), and the number of edges connected to the node. Edge-based features are the maximum similarity, the minimum similarity, the arithmetic mean, and the standard deviation of the similarities. Lastly, the structural features are the egonet degree, egonet density, between centrality, degree centrality, neighborhood node degree histograms (Vidanage et al., 2020, p. 1489). In the provided code,<sup>35</sup> the authors also state that not all features have to be calculated, but rather features that have large standard deviations and allow differentiation.

Second, the features are normalized and then translated into bit vectors. The latter is achieved by applying cosine locality-sensitive hashing (Cosine LSH). In cosine locality-sensitive hashing, random numbers are drawn from a Gaussian distribution (as proposed by the authors of the article) to generate the required random hyperplanes. Depending on whether the features are above or below the hyperplane, a zero or a one is assigned (Vidanage et al., 2020, p. 1490; Leskovec et al., 2014, pp. 99–100), forming bit vectors of the length equal to the number of hyperplanes.

As a third step, the bit vectors are grouped using Hamming locality-sensitive hashing (Hamming LSH). This step is optional but reduces the number of possible matches between the two graphs. Given two bit vectors, a sample of bits<sup>36</sup> is drawn, and the Hamming distance is calculated. The Hamming distance is the number of bits that

<sup>35</sup>Retrieved 10.10.2020: <https://dmm.anu.edu.au/pprlattack/cikm-2020-paper-demo-code.tar.gz>

<sup>36</sup>This is especially helpful for large data sets. However, for the given data set, the computation time using the entire bit vectors is not much higher, and since it is more accurate, the Hamming distance of the entire bit vector was calculated.

differ between  $x$  and  $y$  (Hamming, 1950, p. 155). The probability ( $p$ ) that the samples agree is

$$p = 1 - \frac{h(x, y)}{d} \quad (4.42)$$

with  $h(x, y)$  being the Hamming distance between the samples and  $d$  the number of bits. If two bit vectors' Hamming distance is above a threshold ( $b_m$ ), they are compared in the following step. If not, they are no longer considered as potential matches.

The fourth step is the actual comparison of the two graphs. For the comparison, the cosine similarity, the similarity confidence, and the degree confidence are calculated. The cosine similarity is calculated as (Heimann et al., 2018, notation from Vidanage et al., 2020, p. 1490):

$$cs(v, u) = \frac{f(v) \cdot f(u)}{\|f(v)\| \cdot \|f(u)\|} \quad (4.43)$$

with  $f(v)$  and  $f(u)$  the normalized feature vectors of nodes  $v$  and  $u$ . The similarity confidence is given by:

$$sc(v, u) = \frac{cs(v, u) \cdot (p + q - 2)}{\sum_{i=1}^{p-1} cs(v, u_i) + \sum_{j=1}^{q-1} cs(v_j, u)} \quad (4.44)$$

with  $cs(v, u)$  being the cosine similarity of nodes  $v$ , and  $u$ . Lastly, the degree confidence is given by:

$$dc(v, u) = \frac{1}{p + q - 1} \quad (4.45)$$

with  $p$  being the degree of node  $v$  and  $q$  being the degree of node  $u$ .

Either one of the similarity measures is chosen, or a weighted sum of the measurements is considered. The weights for the three similarity measures should sum to one. The authors propose to use  $w_{cs} = 0.5$ ,  $w_{sc} = 0.3$ , and  $w_{dc} = 0.2$ . Also, the similarity measures have to be normalized prior to calculating the weighted sum (Vidanage et al., 2020, p. 1488).

The result of calculating the similarity measures between the two graph nodes is a bipartite graph. The matching pairs can then be identified using the symmetric highest match (SHM), the stable marriage match (SMM), the maximum weight match (MWM) as proposed by the authors or any other preferred method (Vidanage et al., 2020, p. 1491). Symmetric highest match successively finds the largest similarity. It then deletes all other edges connected to the nodes as well as the edges of their neighbors (Vidanage et al., 2020, p. 1491), until no more unmatched nodes remain. Stable marriage match has a similar approach, but only deleted edges that are connected to

the node of interest and not their neighbors as well (Vidanage et al., 2020, p. 1491). Maximum weight match can be performed with the Hungarian algorithm as described above, but instead of finding the overall minimum, the overall maximum is found, since, in this case, a larger number indicates are higher similarity. A simulation by Vidanage et al. (2020, p. 1493) showed stable marriage match to be the most successful method, followed by symmetric highest match and maximum weight match.

The authors themselves see two limitations with this approach. The graph matching attack on privacy-preserving record linkage method will not perform well when the difference between similarities of the encoded and not-encoded data is large, and the relationship is not linear. Second, the accuracy decreases when more attributes are encoded (Vidanage et al., 2020, pp. 1493–1494). The second limitation is not present for geomasking methods because only the attribute “geographic coordinates” is encoded. Concerning the first limitation, the distances between the masked coordinates are compared with the original distances. A limitation not explicitly stated as such in the paper is that large overlaps between data sets are considered. This is not the case for the given data. Therefore, lower correct classifications are likely. Also, only the top  $t$  (e.g., 10, 100, 1000) matches, based on their similarity, are considered (Vidanage et al., 2020, p. 1492). For the data given here, the known size of the overlap will be used as a value for  $t$ .

Additionally, some adjustments must be made to apply this method to the identification of matches if the encoding is done via geomasking methods. The authors primarily use this attack method to compare hashed information to clear text information and use the Dice coefficient or the Jaccard similarity as the initial similarity measure ( $s_m$ ). The similarity measure here is the Haversine distance, so in this case, a smaller value indicates similarity rather than a larger value. Thus only nodes with a distance smaller than  $s_m$  were considered. As nodes, each coordinate pair is considered separately since all coordinate pairs are unique. Therefore, the features “number of records with the same information as the node”, and “length of the node” are irrelevant here. Further, the degree distribution of a node’s one-hop and two-hop neighborhoods showed barely any variation and was therefore not used as a feature. Lastly, before calculating the similarity measures, a step is added. Namely, the attributes sex, age, and employment status are used to eliminate matches further.

The question of the parameter choice remains. For this attack, several parameter settings must be made. First, the minimum similarity threshold  $s_m$  has to be set, which limits the number of connections between nodes of each graph. Other than the choice for the performed simulation ( $s_m = \{0.2, 0.3, 0.4\}$ ) by the authors, no recommendation is given. Second, the minimum connected component size ( $c_m$ ) must be set for choosing whether features are calculated or not. Again, no recommendation are given other than what the authors used for the simulation ( $c_m = \{5, 10, 50, 100\}$ ). Third, the minimum similarity of pairs for the bipartite graph ( $b_m$ ) must be determined. Recommendations taken from the simulation study show that high values should be

chosen ( $b_m = \{0.8, 0.9\}$ ). Fourth, the number of hyperplanes used in Cosine LSH must be specified. No recommendations are given here. Therefore, a simulation study was performed (see paragraph below). Lastly, the similarity measures and the identification methods for the matched pairs must be chosen. Each of the similarity measures, as well as the weighted sum, are calculated according to the authors' recommendations, and the stable marriage match and the symmetric highest match are used to find matching pairs since these are the two best performing, according to Vidanage et al. (2020, p. 1493).

On a sample of  $n = 100$  of the identification file, three masking methods, randomly selected from the list of all parameter choices of all masking methods, were chosen for which the parameter options were evaluated. The masking methods and parameters applied were: RPC with population density based on postcode and the number 4 as the multiplier (RPC 4), Voronoi masking, and VNE with at least 100 people in the area based on 3 times the estimate of the average distance between people (VNE 100 3). For each masking method three replications were used.<sup>37</sup>

The minimum connected component size ( $c_m$ ) was arbitrarily set to ten. Larger numbers would have eliminated nodes (here coordinates) located in rural areas with few and distant neighboring points. The weights for the three different similarity measures, cosine similarity ( $cs_w$ ), similarity confidence ( $sc_w$ ), and degree confidence ( $dc_w$ ), were kept the same as the recommendation weights by the authors: for cosine similarity 0.5, for similarity confidence 0.3, and degree confidence 0.2 (Vidanage et al., 2020). Because the attack has to be applied to each replication of each masking method, the methods used to identify matches were limited to the two best performing of the paper, symmetric highest match (SHM) and stable marriage match (SMM). The parameters tested in the simulation study were the minimum similarity threshold<sup>38</sup>  $s_m$  (5,000; 10,000; 20,000; 30,000), the number of hyperplanes for cosine LSH (10,000; 5,000), and the minimum similarity of pairs for the bipartite graph  $b_m$ . For the latter, the attack was performed until  $b_m$  is needed, and the similarities achieved for the correctly matching nodes between the masked coordinates and the original coordinates were examined manually. The result was that a minimum similarity threshold of  $b_m = 0.6$  should be considered to avoid losing too many true matches.

Considering a  $b_m = 0.6$ ,  $c_m = 10$ ,  $cs_w = 0.5$ ,  $sc_w = 0.3$ ,  $dc_w = 0.2$ , each combination of  $s_m$  and the number of hyperplanes was applied. The resulting values for the precision and recall of the attacks were then compared. The results showed that the highest values for precision and recall were obtained when the minimum similarity threshold was  $s_m = 10,000$ . The number of hyperplanes did not show different results. Since fewer hyperplanes reduce the computation time, the number was set to 5,000.

In summary, the following parameter values were chosen:  $s_m = 10,000$ ,  $c_m = 10$ ,

<sup>37</sup>Due to the fact that many parameters have to be defined and the risk method needs a lot of computational power for larger samples, only a small sample could be considered.

<sup>38</sup>As stated in chapter 4.2.7, the similarity measure used here is the Haversine distance. Thus, a smaller value indicates higher similarity.

$b_m = 0.6$ , number of hyperplanes was set to 5,000,  $cs_w = 0.5$ ,  $sc_w = 0.3$ ,  $dc_w = 0.2$ , and  $t = 100$  ( $t = 200$  for  $n = 2,000$ ). The methods for the bi-partite matching are SMM and SHM. Moreover, the simulation study also showed that better precision and recall was achieved for SMM when only the cosine similarity is used and for SHM the weighted combination.

#### 4.2.8. Combination of Risk Measures

For the R-U confidentiality map, one value for risk is needed. As the risk measure, the highest achieved combination of precision and recall is taken. This combination could be expressed by the F-measure (also called F-score) (Fawcett, 2003, p. 2):

$$F = \frac{2}{1/\text{precision} + 1/\text{recall}} \quad (4.46)$$

More recently Hand and Christen (2018, p. 542) have shown that the F-score can be rewritten, using the notation introduced above, as:

$$F = p \cdot \text{recall} + (1 - p) \cdot \text{precision} \quad (4.47)$$

with

$$p = \frac{FN + TP}{FN + FP + 2 \cdot TP} = \frac{\text{positives}}{\text{positives} + \text{predicted}}$$

This shows that precision and recall can be unequally weighted if the number of overlapping records (positives) is not equal to the number of the predicted overlap (predicted; Hand and Christen, 2018, p. 546). In the present case, some attack methods result in the unequal weighting of precision and recall, so not all of the F-scores can be easily compared. As a solution to enforce equal weighting of precision and recall, the average of precision and recall can be taken instead of the harmonic mean as the F-score does (Hand and Christen, 2018, p. 546). To avoid confusing this with the definition of the F-score, the notation of Borgs (2019, p. 31) is used and called MPR (mean precision/recall) accordingly:

$$\text{mean prec./rec. (MPR)} = \frac{1}{2}(\text{recall} + \text{precision}) \quad (4.48)$$

### 4.3. Overview of Analysis

First, the masking methods are applied to the full sample of coordinates of the chosen region ( $n = 10,000$ ). Each parameter choice of the masking methods is applied 50

times to show possible variation.

In a second step, the utility of each replication is evaluated using all utility measures (see figure 4.2). When anonymization of distance matrices via Lipschitz embedding and distance approximation using ISGP is used, descriptive statistics can only be evaluated if the coordinates are approximated given the distance matrix (see also table 4.3). This can be done using Procrustes analysis and multidimensional scaling.

Multidimensional scaling (MDS) is used to approximate the coordinates based on the distance matrix (see, e.g., Mardia et al., 1995, pp. 394–402). There are multiple solutions to this problem, which can be divided into non-metric methods, which use only the rank order of the distances, and metric methods, which directly use the given distances. In the following, for MDS, only the so-called classical solution, used in this thesis, is explained, which first uses matrix  $D$ , which contains the distances, to calculate matrix  $A$ :

$$A = \left( -\frac{1}{2}d_{rs}^2 \right) \quad (4.49)$$

where  $d_{rs}$  is the Euclidean distance between point  $P_r$  and  $P_s$ . Matrix  $B$  is then obtained with the elements:

$$b_{rs} = a_{rs} - \bar{a}_{r.} - \bar{a}_{.s} + \bar{a}_{..} \quad (4.50)$$

with

$$\bar{a}_{r.} = \frac{1}{n} \sum_{s=1}^n a_{rs} \quad \bar{a}_{.s} = \frac{1}{n} \sum_{r=1}^n a_{rs} \quad \bar{a}_{..} = \frac{1}{n^2} \sum_{r,s=1}^n a_{rs} \quad (4.51)$$

Then the largest  $k$  eigenvalues  $\lambda_1 > \dots > \lambda_k$  of  $B$  are found with the corresponding eigenvectors  $X = (x_{(1)}, \dots, x_{(k)})$ . These are then normalized by

$$x'_{(i)}x_{(i)} = \lambda_i \quad \text{with } i = 1, \dots, k \quad (4.52)$$

and the points  $P_r$  are

$$x_r = (x_{r1}, \dots, x_{rp})' \quad \text{with } r = 1, \dots, k \quad (4.53)$$

the rows of  $X$  (Mardia et al., 1995, pp. 394–402).

The coordinates are moved to the study area using Procrustes analysis. Given two matrices  $x$  and  $y$ , in the Procrustes analysis, matrix  $y$  is translated, rotated, and

scaled to yield the values in matrix  $x$ :

$$y_r^* = cA'y_r + b \quad (4.54)$$

where  $y_r^*$  should ideally result in the values of  $x$ ,  $c$  is the scale factor,  $A'$  is the rotation matrix, and  $b$  is the translation. The following description and notation are based on Mardia et al. (1995, pp. 416–418). Easier to understand is Schnell (1994, pp. 209–211). To yield  $A'$ ,  $c$ , and  $b$ , first, the matrices  $x$  and  $y$  are centered column-wise by subtracting the column-means. In the following, the centered matrices are referred to as  $X$  and  $Y$ . To yield the rotation matrix, a singular value decomposition (SVD) of  $Y'X$  is performed (equation (4.55)), and the orthogonal matrices  $U$  and  $V$  are used to calculate the rotation matrix (equation (4.56)).

$$\text{SVD}(Y'X) = VTU' \quad (4.55)$$

$$A = VU' \quad (4.56)$$

The translation is calculated by multiplying the column-wise means of  $y$  and subtracting it from the means of  $x$ .

$$b = \bar{x} - A'\bar{y} \quad (4.57)$$

The scale factor is calculated as

$$c = (\text{trace } \Gamma)/(\text{trace } YY') \quad (4.58)$$

If a scale factor is used, the translation is altered by multiplying  $\bar{y}$  with the scale factor (Mardia et al., 1995, pp. 416–418). Given  $A'$ ,  $c$ ,  $b$ , and the points from the multidimensional scaling, formula (4.54) is used to yield the approximated coordinates.

The utility measures could not be calculated for random projection because random projection results in a bit vector. Using the Jaccard index, researchers can say whether points are close or far away on a scale from 0 to 1, but a distance in the unit meters cannot be calculated.

In a third step, the risk of re-identification is assessed for each masking method. As stated above, some methods could not be applied to the entire sample.<sup>39</sup> For these, subsamples are used, as explained in more detail in the following chapter. In addition, reversing the masking methods is performed only for the affine transformations. Since locations masked with affine transformations could be easily re-identified, the other,

<sup>39</sup>This was mainly due to the long run times necessary with the computing power available.

more complex attack methods are not needed and, therefore, not used for affine transformations. Some attack methods require the input of coordinates. Therefore not all attack methods could be applied to the masking methods anonymization of distance matrices via Lipschitz embedding, distance approximation using ISGP, and random projection. An overview of which attack method is applied to which masking method, as well as the sample size used, is given in table 4.3.<sup>40</sup>

In a fourth step, the utility measures and the risk measures for each application of each parameter choice are aggregated to one value for each masking method and are then presented in a risk-utility map. Based on the assumption that a lower risk of re-identification also lowers data utility (Duncan, Fienberg, et al., 2001, p. 139), masking methods that move the points further from their original location (lower utility), should have a lower re-identification risk. For example, donut masking sets a minimum displacement distance that, on average, should displace coordinates further than random perturbation within a circle. Therefore, donut masking should have lower risk and lower utility values.<sup>41</sup>

---

<sup>40</sup>The attack methods are abbreviated as follows: reversing = reversing masking methods; mean = mean of multiple releases; MinDist = minimum distance; Hungarian = Hungarian algorithm; HungBlock = Hungarian algorithm using additional variables; Graph = graph theoretic linkage attack; PPRL = graph matching attack on privacy-preserving record linkage.

<sup>41</sup>However, because the utility evaluation is complex and the success of re-identification can only be guessed, no hypotheses are made about the position of the geomasking methods in a risk-utility map.

Table 4.3.: Overview of masking methods and if utility and risk measures could be applied (used sample size written below utility/risk measures).

$n$	Utility		Risk					
	10,000	reversing 10,000	mean 10,000	MinDist 1,000;2,000	Hungarian 1,000;2,000	HungBlock all sizes	Graph 1,000	PPRL 1,000,2,000
Aggregation								
MDAV	yes	no	yes	yes	yes	yes	yes	yes
Official Statistics Grid	yes	no	yes	yes	yes	yes	yes	yes
Adap. Point Aggregation	yes	no	yes	yes	yes	yes	yes	yes
Adjusting Coordinates								
Displace. Using Translation	yes	yes	no	no	no	no	no	no
Change of Scale	yes	yes	no	no	no	no	no	no
Rotation	yes	yes	no	yes	yes	yes	yes	yes
Random Perturbation	yes	no	yes	yes	yes	yes	yes	yes
Adap. Random Perturbation	yes	no	yes	yes	yes	yes	yes	yes
Donut Masking	yes	no	yes	yes	yes	yes	yes	yes
k-nearest neighbor Donut	yes	no	yes	yes	yes	yes	yes	yes
Voronoi Masking	yes	no	yes	yes	yes	yes	yes	yes
Location Swapping	yes	no	yes	yes	yes	yes	yes	yes
Verified Neighbor	yes	no	yes	yes	yes	yes	yes	yes
Street Masking	yes	no	yes	yes	yes	yes	yes	yes
Random Projection	no	no	no	no	no	no	no	yes
Coordinate Replacement								
Lipschitz Embedding	yes; approx. coordinates for descriptive statistics	no	no	no	no	no	yes	yes
Approx. using ISGP								
Approx. using ISGP	yes; approx. coordinates for descriptive statistics	no	no	no	no	no	yes	yes



## 5. Analysis and Results

In the following chapter the masking methods are evaluated using the utility measures described in chapter 4.1 and the risk measures described in chapter 4.2. There are three main sections in this chapter. The first, focuses on a general comparison of the masking methods, e.g., format of output and required additional information. The second and third sections present the results for the utility and risk measures. For each utility measure the masked data set is compared with the original data set.

### 5.1. General Comparison of Geomasking Methods

The masking methods differ not only in terms of their utility and risk, although this is of utmost importance, but they also show differences in terms of additional information or data sets required and whether identical coordinates would be displaced to the same new location. As can be seen in table 5.1 except for APA, MDAV, Grid, and Voronoi masking, the results are different for each replication because the remaining masking methods rely on the generation of random numbers. For distance approximation using ISGP, identical results can be obtained if the same seed is used to generate the grid.

With regards to whether identical coordinates would be moved to the same new location, APA, CS, DUT, distance approximation using ISGP, MDAV, random projection, rotation, and Voronoi masking satisfy this property. Thus, if identical coordinates should be displaced to the same location, the other masking methods would have to be applied to a unique set of coordinates. After that the same masked location has to be assigned to identical unmasked coordinates.

Most masking methods require some additional information. Masking methods that base the possible displacement distance on the population density need this information as an additional variable (e.g., DD). For Dk, LS, VNE, and VNS, a file with all residential addresses is needed. APA and ARP, and anonymization of distance matrices via Lipschitz embedding require the shapefile of the region of interest. For street masking, the OpenStreetMap road network data is needed for the region of interest.

Distance approximation using ISGP, in theory, uses the region of interest. However, the region can also be defined without needing a shapefile. Only the affine transformations, random projection, MDAV, and Voronoi masking do not require additional information to be applied.

Table 5.1.: General comparison of masking methods (only tested methods).

method	same displacement of same coordinates	displacement of same coordinates	same results for multiple replications	additional information needed	output
Adaptive Point Aggregation (APA)	yes		yes	yes, shapefile of subregions	coordinates
Adaptive Random Perturbation (ARP)	no		no	yes, shapefile of subregions	coordinates
Change of Scale (CS)	yes		no	no	coordinates
Donut Masking (DD)	no		no	yes, population density	coordinates
$k$ -nearest neighbor Donut (Dk)	no		no	yes, residential address file	coordinates
Displacement Using Translation (DUT)	yes		no	no	coordinates
Distance Approximation Using ISGP	yes		no	shapefile of region optional	distance matrix
Location Swapping (LS/LSdonut)	no		no	yes, residential address file and population density	coordinates
Lipschitz embedding	yes		no	yes, shapefile of region	distance matrix
Microaggregation (MDAV)	yes		yes	no	coordinates
Official Statistics Grid (Grid)	yes		yes	no	IDs of grid cells
Random Projection (RandProj)	yes		no	no	bit vector
Rotation (Rot/RotArb)	yes		no	no	coordinates
Random Perturbation Circle (RPC)	no		no	yes, population density	coordinates
Random Perturbation Normal (RPN)	no		no	yes, population density	coordinates
Random Perturbation Uniform (RPU)	no		no	yes, population density	coordinates
Street Masking (StreetMask)	no		no	yes, drivable streets as network graph	coordinates
Verified Neighbor (VNS/VNE)	no		no	yes, population density, residential address file and characteristics	coordinates
Voronoi	yes		yes	no	coordinates

The method used by the official statistics (Grid) does not need any further information. However, the coordinates should be in easting-northing format.

Another major difference between masking methods is their output. Most masking methods, as originally intended, give the output in the form of coordinates, i.e., another location. The major advantage is that researchers can link additional information based on the location, such as local unemployment rates and distances to points of interest. Two of the masking methods only allow the release of a modified distance matrix (anonymization of distance matrices via Lipschitz embedding and distance approximation using ISGP). Another masking method only gives bit vectors (random projection), and a further masking method only gives a cell identifier (official statistics grid). Since some masking methods do not yield coordinates, some utility measures cannot be calculated easily. For the official statistics grid, the ID is a concatenated string of direction information, and the coordinates are in easting and northing format. The ID of the center of the grid is taken for the following analysis.

For distance approximation using ISGP and anonymization of distance matrices via Lipschitz embedding, the result is a distance matrix, and the actual location remains unknown. Therefore, descriptive statistics and the distance between original and masked locations cannot be calculated directly based on the distance matrix. As a solution, solely to evaluate utility, the distance matrix can be transformed into coordinates by means of multidimensional scaling. The resulting coordinates must then be moved to the region of interest, for which Procrustes analysis is used. In the following, the resulting coordinates will be referred to as approximated coordinates. This approach is not without consequences. The distance of the spatial mean center to the original spatial mean center is zero because the Procrustes analysis is based on the original coordinates. Furthermore, the transformation of a distance matrix into coordinates by means of multidimensional scaling leads to an additional distortion.

Lastly, because random projection results in bit vectors, random projection does not allow calculating any utility measure chosen here. In addition, researchers can only tell on a scale from 0 to 1 whether points are close or far away with the Jaccard similarity measure, but they lack information about the unit of length.

While the advantage of these masking methods of not releasing coordinates is primarily the additional security of not revealing the coordinates, researchers themselves cannot add any additional information. Random projection, which seems to be more of an idea than a fully developed masking method, leaves the researcher the least amount of information.

In addition to the information shown in table 5.1, the time required to apply a masking method should also be compared. Ideally, the masking method can be applied fairly quickly ensuring fast access to the data once it is available. However, the code for the masking method was not optimized to be as fast as possible but was written to be reasonably fast enough.<sup>1</sup>

---

<sup>1</sup>For detailed results on the execution time and system specifications see table E.1 in appendix E.

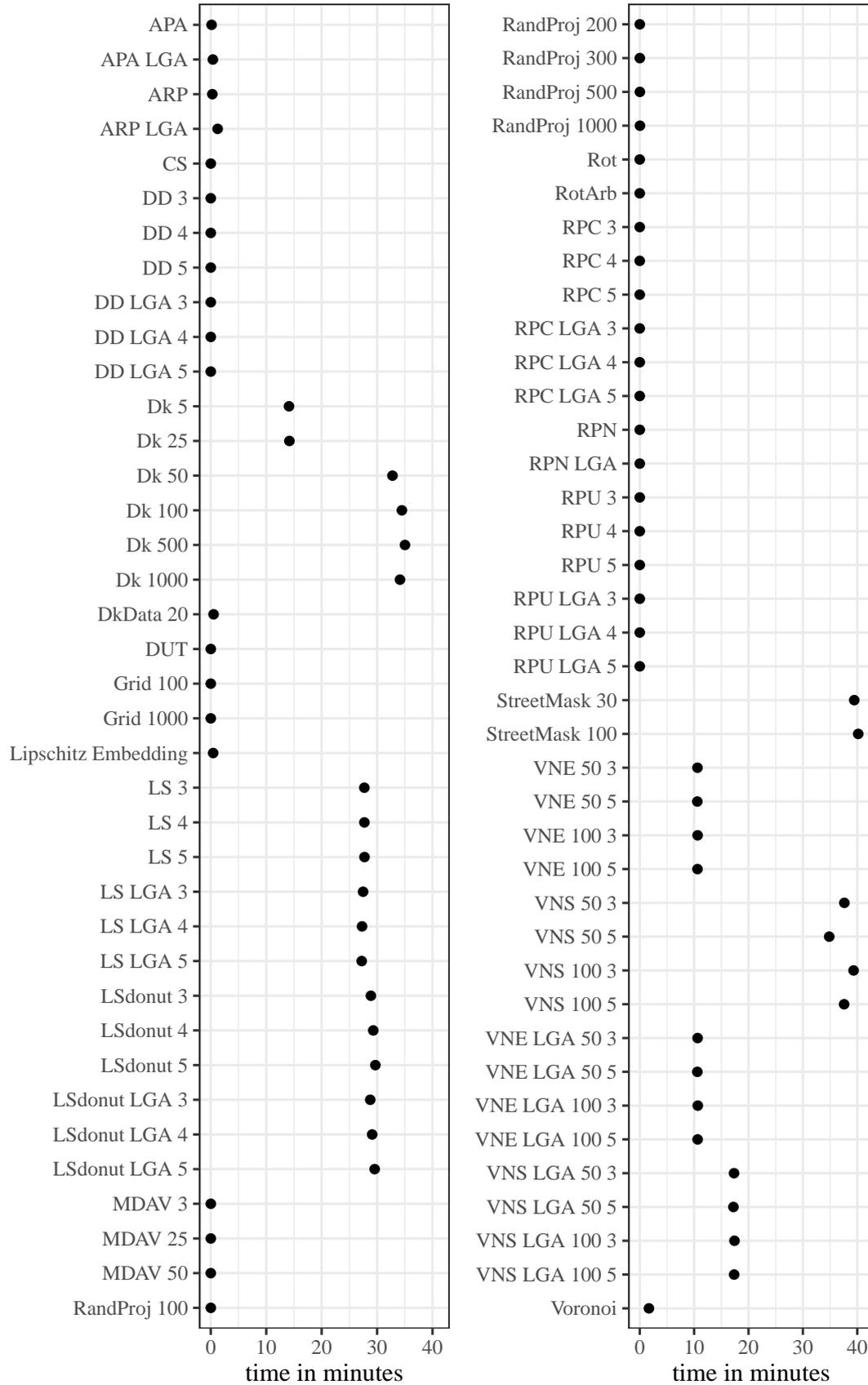


Figure 5.1.: Average execution time (in minutes) of the masking methods to mask 10,000 coordinates (without the outlier distance approximation using ISGP, 709 minutes).

Most masking methods are not computationally intensive, and even with  $n = 10,000$ , the masking methods are applied very quickly (see figure 5.1).<sup>2</sup> The masking methods that take more time are the  $k$ -nearest neighbor donut masking method, location swapping, verified neighbor approach, and street masking. In donut masking, this is caused by the fact that the distances to each neighbor must be calculated and sorted to find the  $k$ -nearest neighbor.

Similarly, in street masking, the nearest node to the location must first be found by calculating the distance to all nodes, and then the distances to the nearest neighboring nodes must be found for these nodes. However, the time would be drastically reduced if more computing power is used. In addition, the time spent downloading the OpenStreetMap data is not included because it varies greatly, and the file only needs to be downloaded once and can be reused for the area.

Location swapping and verified neighbor method require much more time for the same reason. With location swapping, points within a certain radius have to be identified, which is the main reason for the comparatively larger time needed.<sup>3</sup> With the verified neighbor approach, the time decreases because not so many points have to be considered at the same time. However, if no locations within a predefined radius satisfy the characteristic, the distance to all points must be calculated to find the  $k$ -nearest neighbors.

Of the given masking methods, distance approximation using ISGP takes the longest (on average, 709 minutes; excluded in figure 5.1). The longer running time was also pointed out as a drawback by the authors (Schnell, Klingwort, et al., 2021). However, this can be solved using more computing power.

## 5.2. Utility Evaluation

In the following, the masking methods' utility is evaluated using the utility methods discussed in the chapter 4.3.

### 5.2.1. Descriptive Statistics

First, the descriptive statistics were compared between the masked and the original data set. Namely, the position of the spatial mean center as well as the spatial median center were compared between the original and masked data sets. Furthermore, the dispersion around the spatial mean center, the standard distance, was compared (see,

---

<sup>2</sup>Masking methods are abbreviated. A list of abbreviations can be found at the beginning of this thesis, in section 2.4 or with explanation in appendix A.5. For a better comparison of the different parameter choices for each masking method, the y-axis of this figure and those of the other figures have been sorted alphabetically. Also, the positioning of the masking methods will be the same for most figures, allowing easier comparison.

<sup>3</sup>Both methods needed more computing power. VNS using postcode areas shows run time if less computing power is available.

e.g., Seidl, Paulus, et al., 2015). In addition, the standard deviational ellipse was calculated to see if the direction and dispersion of the data changed.

The spatial mean center of the original data set is located at the coordinates (longitude=138.556289328368, latitude=-34.8419447758913). Since outliers strongly influence the spatial mean center, the spatial median center was calculated and is located at the coordinates (longitude=138.602314364691, latitude=-34.8966561617305). The standard distance is 124026.651 meters. In the following, the masking methods are evaluated on whether they preserve the spatial centers and standard distance.

#### 5.2.1.1. Spatial Mean Center

For each masking method and each replication of the masking method, the spatial mean center was calculated. After that the distances between the spatial mean of the masked coordinates for all replications of a masking method and the spatial mean of the original coordinates were calculated.

Figure 5.2 shows the position of the spatial mean center of the masking method when the multiple replications are averaged (plotted in EPSG:3107<sup>4</sup>). It can be seen that using the rotation method moves the spatial mean center far from its original location. All other spatial mean centers remain much closer to the original. Table G.1 in appendix G.1 shows in more detail the smallest distance by which the spatial mean was moved as well as the largest distance if multiple replications are compared. Also, the arithmetic mean of all replications of the distances was calculated. The last column shows the standard deviation.

The rotation of the coordinates around the origin causes the largest distance between the spatial mean of the masked coordinates and the spatial mean of the original coordinates with an average of 2,545.551 km and a standard deviation of 1,178.613 km. Also, the minimum is by far larger than any other masking method. When the coordinates are rotated around the spatial mean center, the location of the masked center and the location of the original center do not differ.

As shown in more detail in table G.1 in appendix G.1, APA and ARP also do not preserve the location of the center of the coordinates very well. Using APA always moves the coordinates to the same new coordinate given the same shapefile as input. Therefore, each replication results in a spatial mean far from the original mean (22.796 km). For ARP, the results vary between replications. Of the given fifty replications using state electorates as polygons, the farthest mean center is 25.259 km away and the closest 22.208 km. On average, the distance of the masked mean centers to the original mean center is 23.481 km, with a standard deviation of 648 meters. The use of local government areas (LGA) leads to smaller values due to more regions, and thus, shorter possible displacement distances, but still, the spatial mean is comparatively far displaced. For APA, the distance to the original mean is 582 meters; for ARP, the

<sup>4</sup>See section 2.2 for an explanation of the used coordinate system.



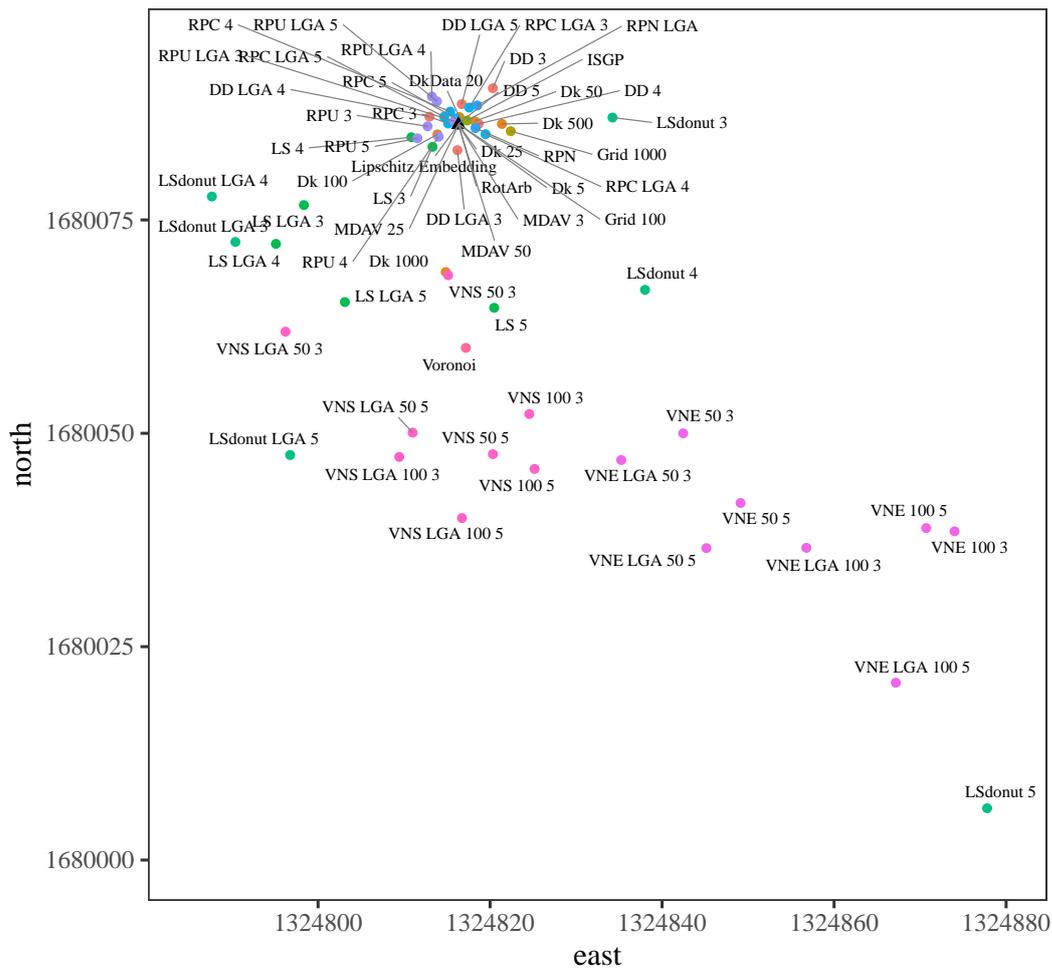


Figure 5.3.: Mean center zoomed in without the methods rotation, change of scale, adaptive areal elimination, street masking, and displacement using translation. Average of mean centers for masking methods (in EPSG:3107; area 95 x 95 meters).

closer look without these outliers (see figure 5.3) shows that many masking methods preserve the original location (indicated by a black triangle) reasonably well (distance from original spatial mean center less than 104 meters, see table G.1 in appendix G.1).

The donut masking method was implemented using population density<sup>6</sup> (DD) and  $k$ -nearest neighbors (Dk). As shown in more detail in table G.1 in appendix G.1 when the estimate of the average distance between people based on postcode area is multiplied by a higher number, the average distance between the masked and the original mean center increases slightly (7-8 meters), which was to be expected. Nevertheless, the mean center remains close to the original mean center. For population density based on the local government areas, the displacements of the spatial means are approximately the same as when using the postcode population density.

Similar results are seen for the  $k$ -nearest neighbors variant. As the number of  $k$

<sup>6</sup>In the following, the term “population density” is used when referring to the estimate of the average distance between people based on the population density. See section 3.2.3 for a detailed explanation.

increases, the average moved distances and the standard deviations of the mean center increase. For small values of  $k$  ( $k = 5, k = 25$ ), barely any displacements of the spatial mean center are seen (on average less than one meter). A  $k$  of 50 is sufficient to displace the spatial mean center by over 10 meters. Larger values of  $k$  result in larger displacements of the spatial mean centers, e.g., an average of 103 meters with  $k = 1,000$  as maximum boundary and  $k = 100$  as the minimum boundary. If the data set is used as a reference for the distance to the nearest neighbor (DkData), a slightly smaller displacement of the spatial mean center than  $k = 1,000$  is obtained, i.e., 96 meters on average. However, the variation between multiple replications is larger.

Population density at postcode level and local government area level was again used for location swapping. The average distances between the masked spatial means and the original spatial means show a comparatively large difference between multiplying the estimate of the average distance between people by 4 (8 meters on average) and 5 (25 meters on average) for population density at postcode level. The distance between the spatial mean of masked coordinates and the original spatial mean is between 10 and 45 meters for multiplications by 5. In comparison, it is between less than one meter and 20 meters for the multiplication of 4. When population density based on local government areas is used, the spatial mean displacement is on average the same for multipliers 4 and 5 (26 meters) and slightly smaller for 3 (21 meters). When location swapping is combined with the idea of donut masking and a minimum radius is also imposed, the mean center is displaced slightly further. For example, for the estimate of the average distance between people based on postcode population density multiplied by 3 (LSdonut 3) the displacement is on average 6 meters compared to 19 meters for the donut variant (DD 3).

Since the verified neighbor approach extends the idea of location swapping by considering only points with matching characteristics, the spatial mean is displaced further on average. The lowest displacement (26 meters) is seen for VNS with  $k = 50$  and the estimate of the average distance between people (based on postcode areas) multiplied by 3 (VNS 50 3). The largest displacement (90 meters) is seen for VNE with  $k = 100$  and the estimate of the average distance between people (based on LGA) multiplied by 5 (VNE LGA 100 5). Both show that an increase in possible displacement distance will increase the distance between the spatial mean centers. Table G.1 in appendix G.1 also shows that, of the remaining masking methods, the verified neighbor masking method moved the spatial mean center the furthest. Increasing the distance, either by minimum neighbors or multipliers of the estimate of the average distance between people, increases the average displacement of the spatial mean and the variation between replications. Increasing the minimum neighbors is more influential than increasing the multiplier of the estimate of the average distance between people. It also shows that the chosen variable strongly influences the possible displacement of the spatial mean center. A variable with just two categories results in smaller displacement distances than if a variable with more categories is used. This

can be explained by the fact that if not enough people with the same characteristic are found within the radius, it is enlarged, and so even further displacements are possible.

The spatial mean center of MDAV is not displaced even when using larger values for the number of clusters. RPC moves the mean center between 15 and 25 meters for population density based on postcode areas and 16 and 26 meters for population density based on local government areas. Since there is no minimum radius, the displacement of the mean center in RPC is, on average, slightly smaller than in donut masking using population density (DD). RPU moves the spatial mean center further than RPC, even if the estimate of the average distance between people is multiplied by the lowest number. However, the average displacement distance of the mean centers only is between 26 and 43 meters. RPN displaces the mean center only by about 10 meters. The difference between population density based on postcode and population density based on local government areas is negligible.

Voronoi masking is a masking method that will always yield the same result when the same coordinates are given as input. The masked mean center is 26 meters away from the original mean center.

Only a distance matrix is given for distance approximation using ISGP and anonymization of distance matrices via Lipschitz embedding. Solely based on a distance matrix, spatial mean centers cannot be calculated. Based on the distance matrix and multidimensional scaling, the coordinates are found, which are then moved to the study area using the Procrustes analysis and the original points as input. After that the spatial mean center can be calculated. However, for distance approximation using ISGP and anonymization of distance matrices via Lipschitz embedding, the spatial mean center always lies exactly at the original spatial mean center. With the random projection method, such as solution is not possible because the output is a bit vector and a distance measurement in meters is not calculable.

Using the idea of a grid as it is currently used in official statistics, the coordinates are not replaced by another coordinate but by an area identifier. For the evaluation of their utility, the coordinates of the center of the cells were taken. Neither the 100 meters grid nor the 1,000 meters grid results in large displacements of the spatial mean center. Moreover, the official statistics grid always yields the same results; less than one meter for 100 meters grid and about 6 meters for 1,000 meters grid.

#### **5.2.1.2. Spatial Median Center**

For the spatial median center, the same procedure was performed as for the spatial mean center. The position of the average spatial median center of the masking methods' fifty replications (plotted in EPSG:3107) are shown in figure 5.4 and the detailed results can be found in table G.2 in appendix G.2. Again, due to the large displacement of the spatial median center of rotation, the differences of other masking methods cannot be seen. For that reason, figure 5.5 shows the placement of the

average of the spatial median centers without outliers.

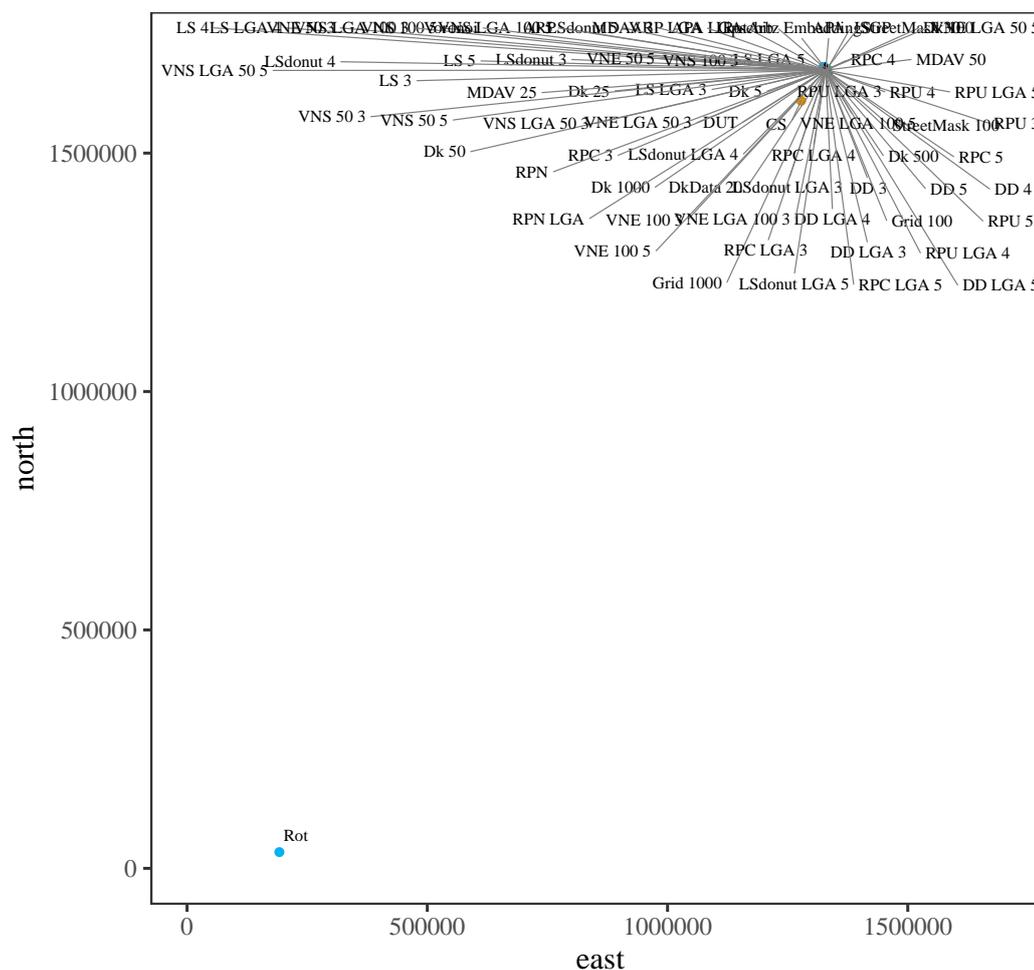


Figure 5.4.: Spatial median centers of masking methods (in EPSG:3107). Without outliers shown in figure 5.5.

Similar to the spatial mean, the spatial median differs greatly from the original median center for the masking methods APA, ARP, change of scale, displacement using translation, rotation, and using a 1,000 meters grid. The main difference is that APA and ARP displace the spatial median further when local government areas are used (APA 1 km and ARP 894 meters) as opposed to using state electorates (APA 804 meters and ARP 175 meters). With the masking method change of scale, the spatial median center is displaced from the original median center as far as the spatial mean center from the original spatial mean center, i.e., 1,168 km on average. For anonymization of distance matrices via Lipschitz embedding, distance approximation using ISGP the spatial median center differs greatly from the original median center as well. Displacement using translation and rotations about the origin replaces the spatial median approximately as far as the spatial mean.

When the rotation is about the spatial mean, the spatial median is displaced by 10 km on average. The official statistics grid method displaces the median center by 50

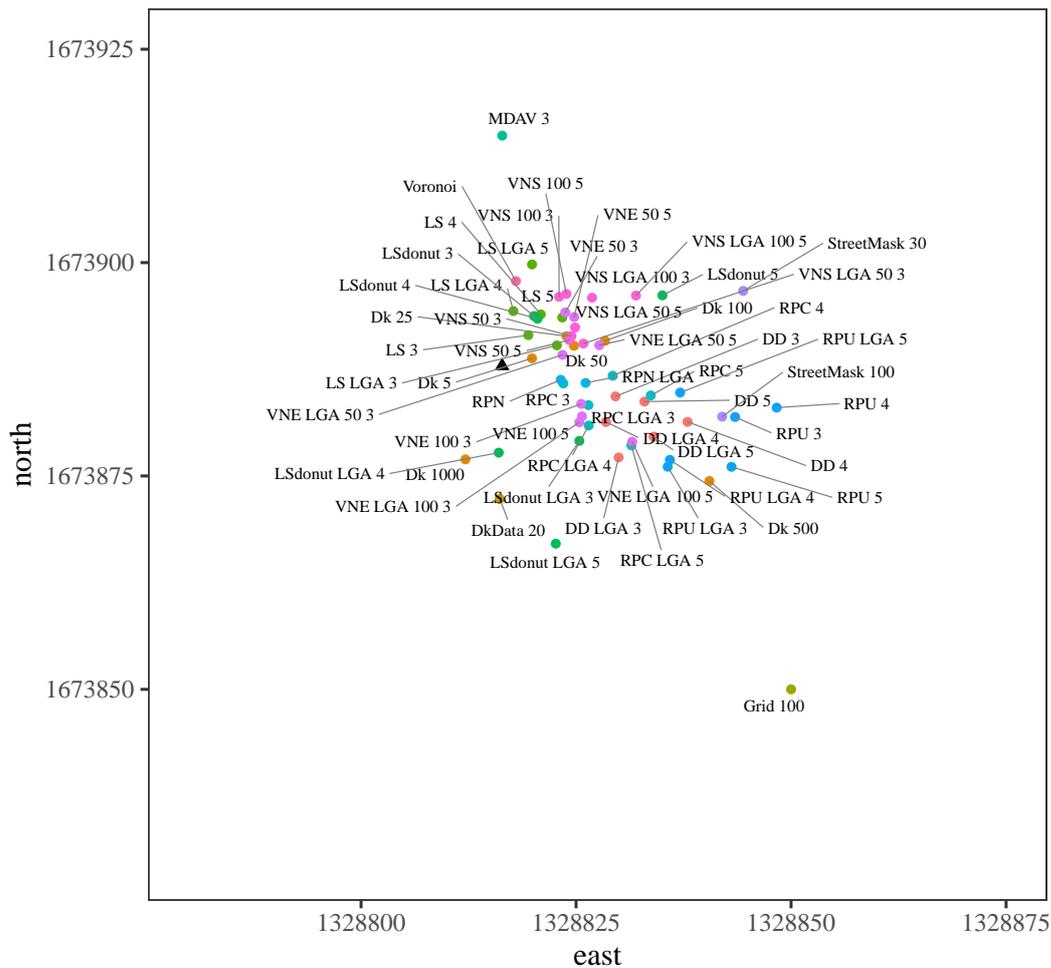


Figure 5.5.: Median center zoomed in without the rotation methods, change of scale, adaptive areal elimination, displacement using translation, microaggregation (cluster sizes 25 and 50), Lipschitz embedding and using a 1,000 meter grid. Average of median centers of the fifty replications for each masking method is plotted (in EPSG:3107; area 95 x 95 meters).

meters and 500 meters, respectively. The microaggregation method (MDAV) displaces the spatial median center by 27 meters for cluster size 3, 57 meters for cluster size 25, and 133 meters for cluster size 50. For anonymization of distance matrices via Lipschitz embedding and distance approximation using ISGP, the same procedure as for the spatial mean was used. This showed large displacements of the spatial median (on average, 10 km for anonymization of distance matrices via Lipschitz embedding and 4 km for distance approximation using ISGP).

Figure 5.5, therefore, shows the plot without the masking methods that cause large displacements. The black triangle indicates the original spatial median center. The overall displacement of the spatial median is not as large as for the spatial mean center (within 40 meters of the original spatial median center, see table G.2 in appendix G.2).

Donut masking using population density displaces the spatial median center by approximately the same distance as the spatial mean center (between 20 meters and

27 meters on average). However, there are smaller differences in the displacement distance of the spatial median center when multiplying the estimate of the average distance between people by 3, 4, or 5 than the differences in the displacement distance of the spatial mean center when multiplying the estimate of the average distance between people by 3, 4, or 5. Depending on the value of  $k$ , the spatial median center is moved about the same distance as the spatial mean center ( $k = 50, k = 100$ ), moved further for small values of  $k$  ( $k = 5, k = 25$ ) or moved less for larger numbers of  $k$  ( $k = 500, k = 1,000$ ). Again, using the data set as the reference file results in the same displacements as for the larger values of  $k$  (on average 35 meters for the spatial median center). Street masking only displaces the spatial median center for both depth values minorly by about 32 meters.

The spatial median center is displaced further in location swapping than the spatial mean when the postcode population density is used (between 11 and 19 meters on average). The verified neighbor method, in turn, displaces the spatial median center slightly further than location swapping (between 17 meters and 25 meters on average). However, the spatial median center of the verified neighbor method is displaced much less than the spatial mean center. In random perturbation, the spatial median center is displaced similarly to the spatial mean center. Voronoi masking causes the spatial median center to be displaced by less than half the displacement as for the spatial mean center (10 meters).

### 5.2.1.3. Standard Distance

The standard distance of the original data set is 124,026.651 meters. Figures 5.6 and 5.7 show the average of the standard distance of the fifty replications of each masking methods' parameter choice. Table G.3 in appendix G.3 shows the detailed results.

As can be seen in figure 5.6, APA and ARP that use state electorates as polygons result in much larger standard distances (APA 190,671 meters and ARP 207,488 meters, average of 50 replications). Thus the points are even more spread out than in the original data set. For ARP using the local government areas, the standard distance is, on average, 1,000 meters larger than the standard distance of the original data set. In comparison, the standard distance is well preserved for APA using the local government areas (about 80 meters larger).

Anonymization of distance matrices via Lipschitz embedding and distance approximation using ISGP show much smaller standard distances than the original data (108 km and 24 km). Again, it should be noted that these results are based on the approximated coordinates using multidimensional scaling and Procrustes analysis.

A closer look at the standard distances without the outliers mentioned above can be seen in figure 5.7. For the remaining masking methods, the standard distance differs by a maximum of 692 meters. Also, the differences between the parameter choices for each masking method can be seen.

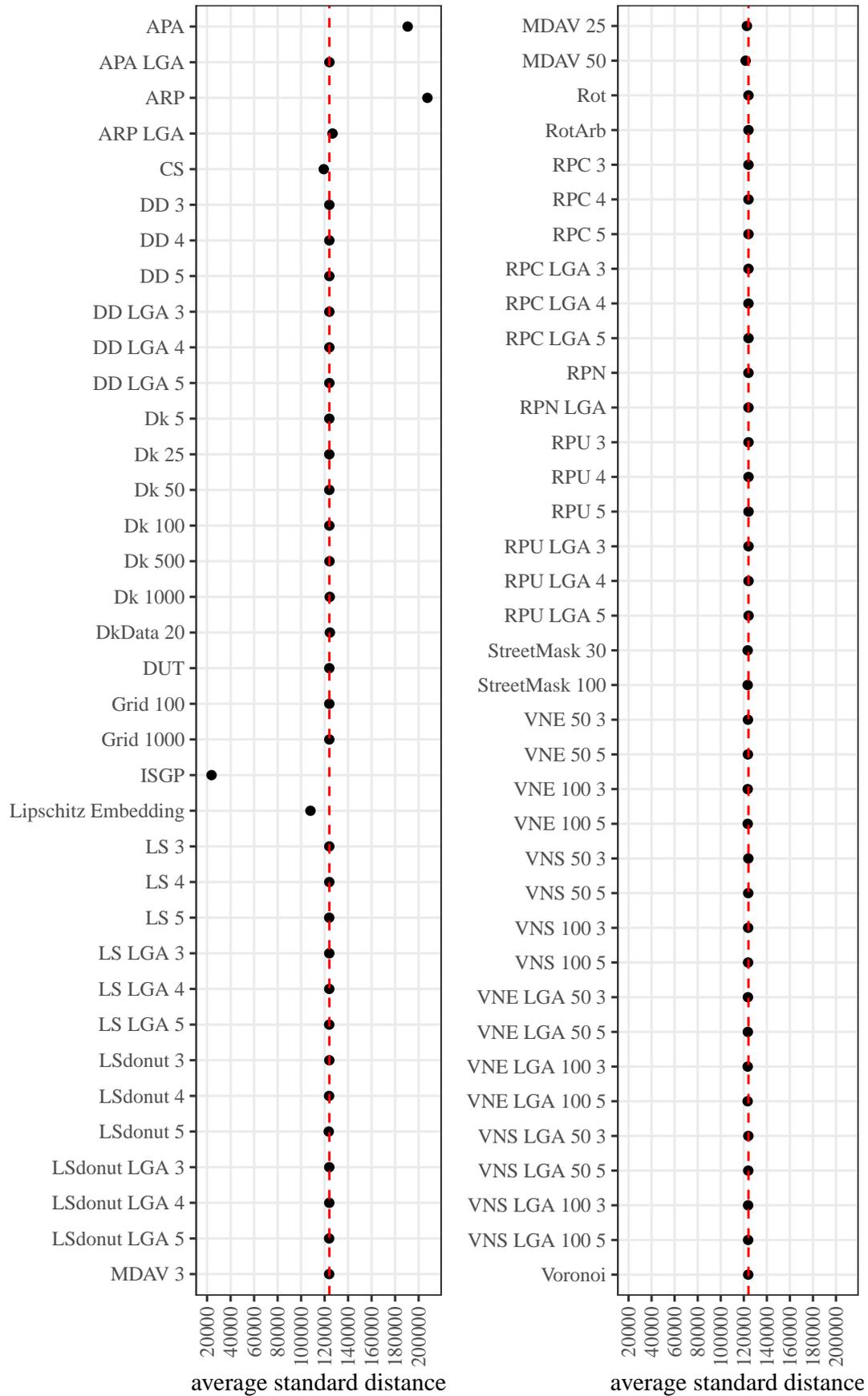


Figure 5.6.: Standard distance of masking methods.

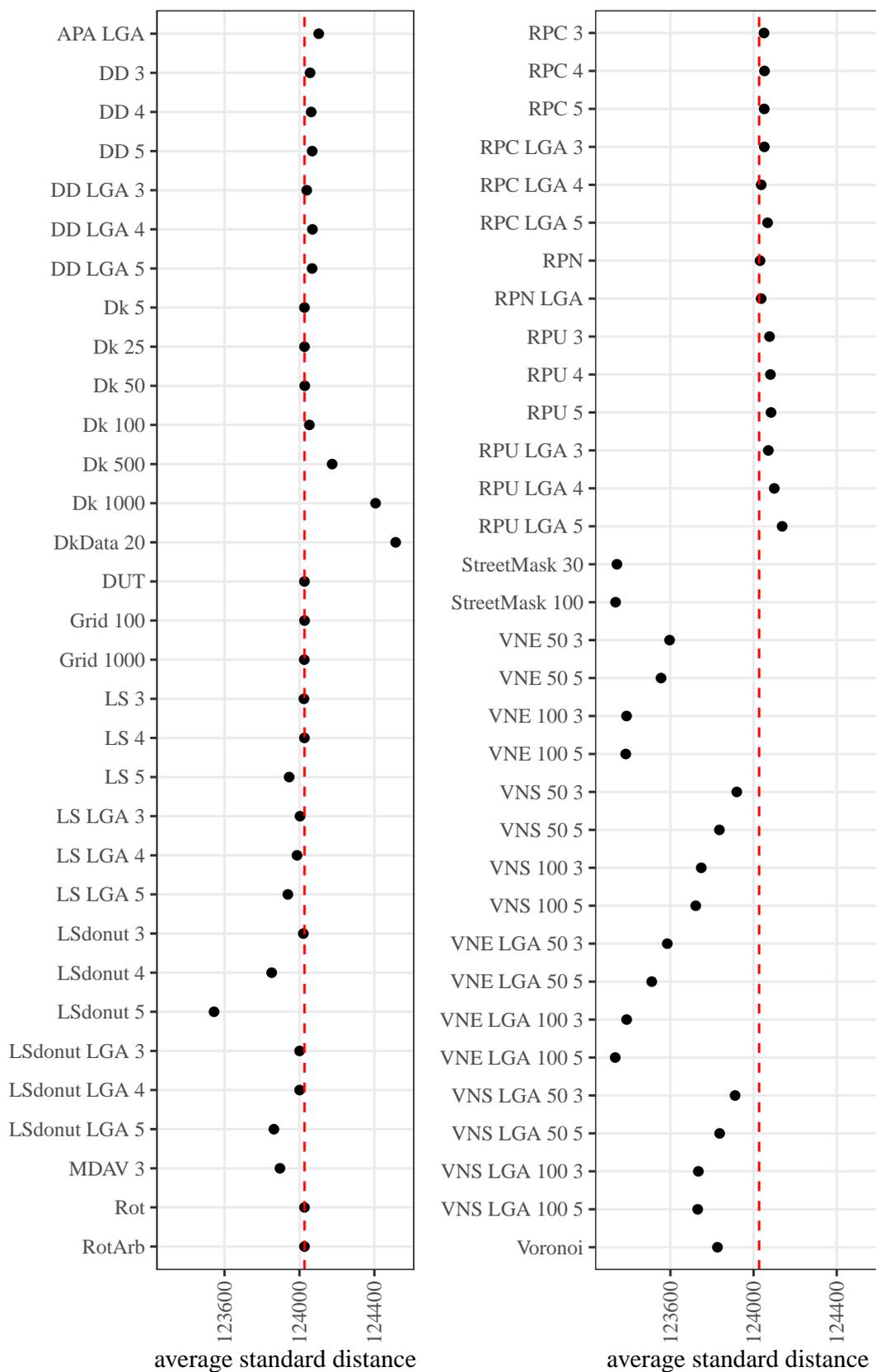


Figure 5.7.: Standard distance of masking methods without outliers.

For donut masking using population density at the postcode level, the standard distance between enlargements of the radius varies by 11 meters. For local government areas, the standard distance of the masked data set is also slightly larger than the standard distance of the original data set. Larger differences are seen for  $k$ -nearest neighbor donut masking. The standard distance is almost identical to the original data set's standard distance for small values of  $k$  ( $k = 5, k = 25$ ). For  $k = 50$ , the standard distance is, on average, two meters larger than that of the original data set, and for  $k = 100$ , on average, 26 meters larger. The average standard distance is even more enhanced for  $k = 500$  and  $k = 1,000$ , and, especially for larger values of  $k$ , the standard distance varies to a larger extent for the given replications. Using the data set as the reference instead of the residential address file, the difference to the original standard distance is even larger (124,513 meters).

The standard distances for displacement using translation and the rotation methods remain the same. For the official statistics grid, the standard distance is almost identical to the original standard distance (difference of one meter).

Location swapping shows slightly smaller standard distances for the estimate of the average distance between people based on postcode population density multiplied by 3 (124,024 meters), almost no difference for multiplication by 4 (124,027 meters), and smaller again for multiplication by 5 (123,945 meters). For local government areas, the standard distance of the masked points is always smaller than the original standard distance. The difference is less than 100 meters. Location swapping in the donut variant also reduces the standard distance, here by a maximum of 175 meters. Street masking, VNS, and VNE result in even smaller standard distances on average, up to about 700 meters smaller than the original standard distance. Again, the results of the two chosen depth values for street masking are very similar.

On average, RPC's standard distance is slightly larger than the original standard distance (about 25 meters). This observation can also be made for RPU at postcode level population density (about 50 meters). For population densities based on local government areas, the standard distance of RPU increases between 74 and 109 meters when the estimate of the average distance between people is multiplied by 4 or 5.

Voronoi masking reduces the standard distance by 200 meters, which is caused by the fact that points in less dense areas are moved closer to other points. Similarly, MDAV reduces the standard distance, especially when larger cluster sizes are chosen.

#### 5.2.1.4. Standard Deviational Ellipse

The original standard deviation ellipse has an angle of  $121.032^\circ$ . The length of the major axis is 114,384.840, while the length of the minor axis is 47,944.954.

Figure 5.8 shows the average angle of the orientation of the standard deviation ellipse for each masking method. Detailed results, such as the minimum, maximum, and the standard deviation of the angles, can be seen in table G.4 in appendix G.4.

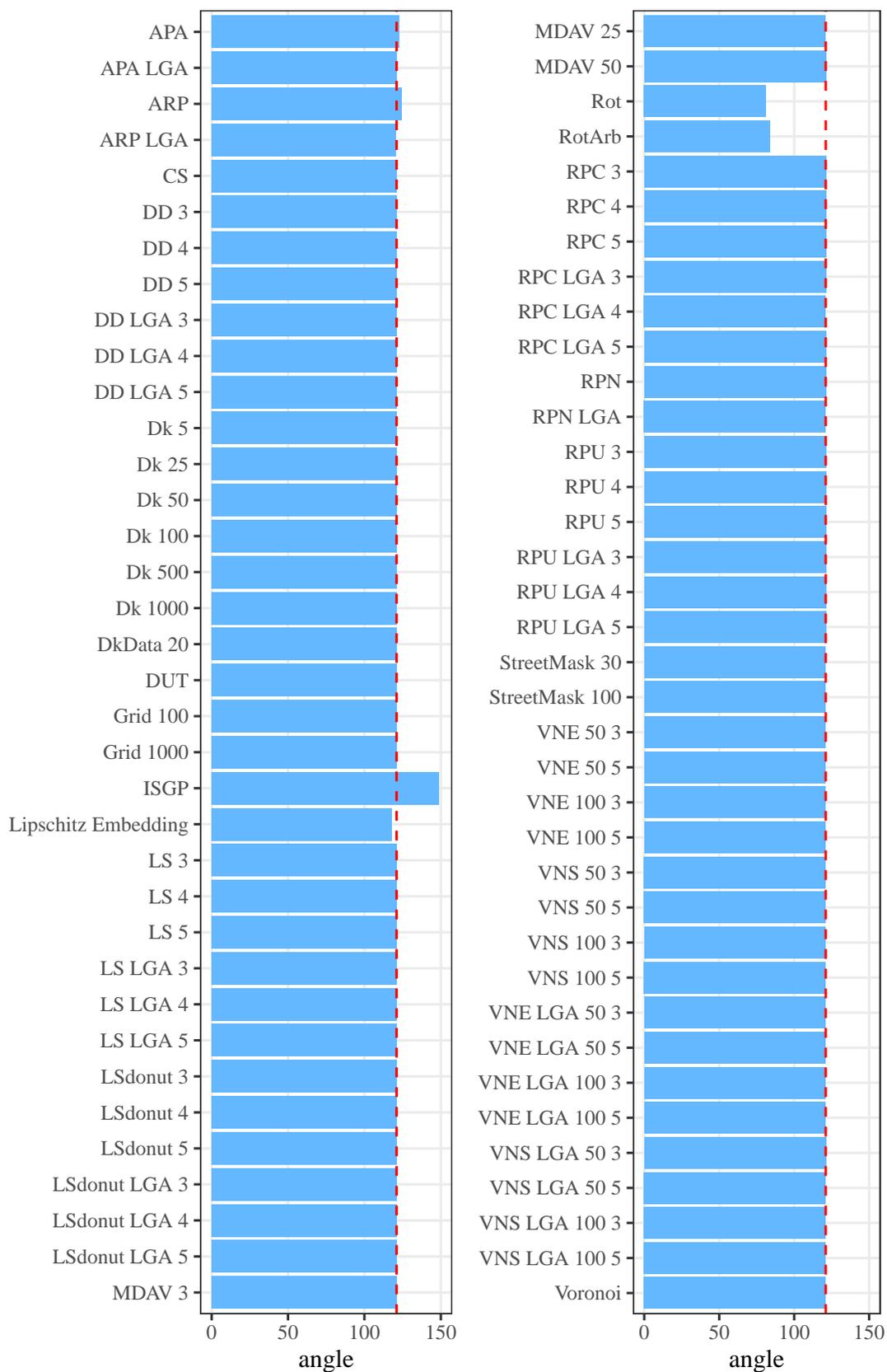


Figure 5.8.: Angle (in degrees) of standard deviation ellipse compared to original data set (red dashed line).

The angles of most masking methods vary by less than one degree, suggesting that the data's overall orientation is preserved (see figures 5.8). Exceptions are APA and ARP (using state electorates), the rotation method, anonymization of distance matrices via Lipschitz embedding, and distance approximation using ISGP.

The angle differs by  $1.48^\circ$  for APA using state electorate polygons. Thus, the orientation of the standard deviational ellipse is turned further counterclockwise, which is still an acceptable change. Comparing the results of APA using state electorate polygons with APA using local government areas (difference to original is  $0.1^\circ$ ), the strong influence of the choice of regional level on preserving the orientation becomes clear. The difference of using state electorates or local government areas as polygons becomes more apparent for ARP. The state electorate polygons cause an average difference in axis rotation of  $3.15^\circ$ . In comparison, for local government areas, the average difference is  $0.29^\circ$ .

When the rotation masking methods are used, the standard deviational ellipse rotates according to the rotation angle. Therefore, large chosen degrees for the rotation change the overall orientation of the data enormously. The angle only differs on average by about  $3^\circ$  for the method anonymization of distance matrices via Lipschitz embedding. However, this varies more (a standard distance of  $5^\circ$ ). For the method distance approximation using ISGP, the angle differs by  $27.6^\circ$ . The results of both masking methods apply to the approximated coordinates.

For the standard deviational ellipse, the major and minor axes' lengths must also be calculated. The length depends on the standard deviation of the points around the mean. The standard distance has already shown that certain masking methods, such as APA or change of scale, show a larger difference in this regard. Figure 5.9 and 5.10 show that most masking methods preserve the length of the axes. For APA and ARP, which use state electorates as polygons, the axes are lengthened on average (APA 60.3 km and ARP 73.9 km larger for major axis and APA 28.5 km and ARP 39.3 km larger for minor axis). Thus, the overall data set is more dispersed.

A shortening of the major and minor axis can be seen for the masking methods change of scale (4.4 km shorter for major axis and 1.8 km shorter for minor axis), anonymization of distance matrices via Lipschitz embedding (15.7 km shorter for major axis and 4.4 km shorter for minor axis), and distance approximation using ISGP (95.1 km shorter for major axis and 34.0 km for minor axis). For anonymization of distance matrices via Lipschitz embedding and distance approximation using ISGP, the results are based on the approximated coordinates. With the method change of scale, it depends only on the number with which the coordinates are multiplied.

Figures 5.11 and 5.12 show the average differences of the major and minor axes without the aforementioned masking methods.<sup>7</sup> These masking methods shorten or lengthen the major axis between 0 and 600 meters and 0 and 1,220 meters for the minor axis.

<sup>7</sup>For more detailed results see table G.5 and G.6 in appendix G.4.

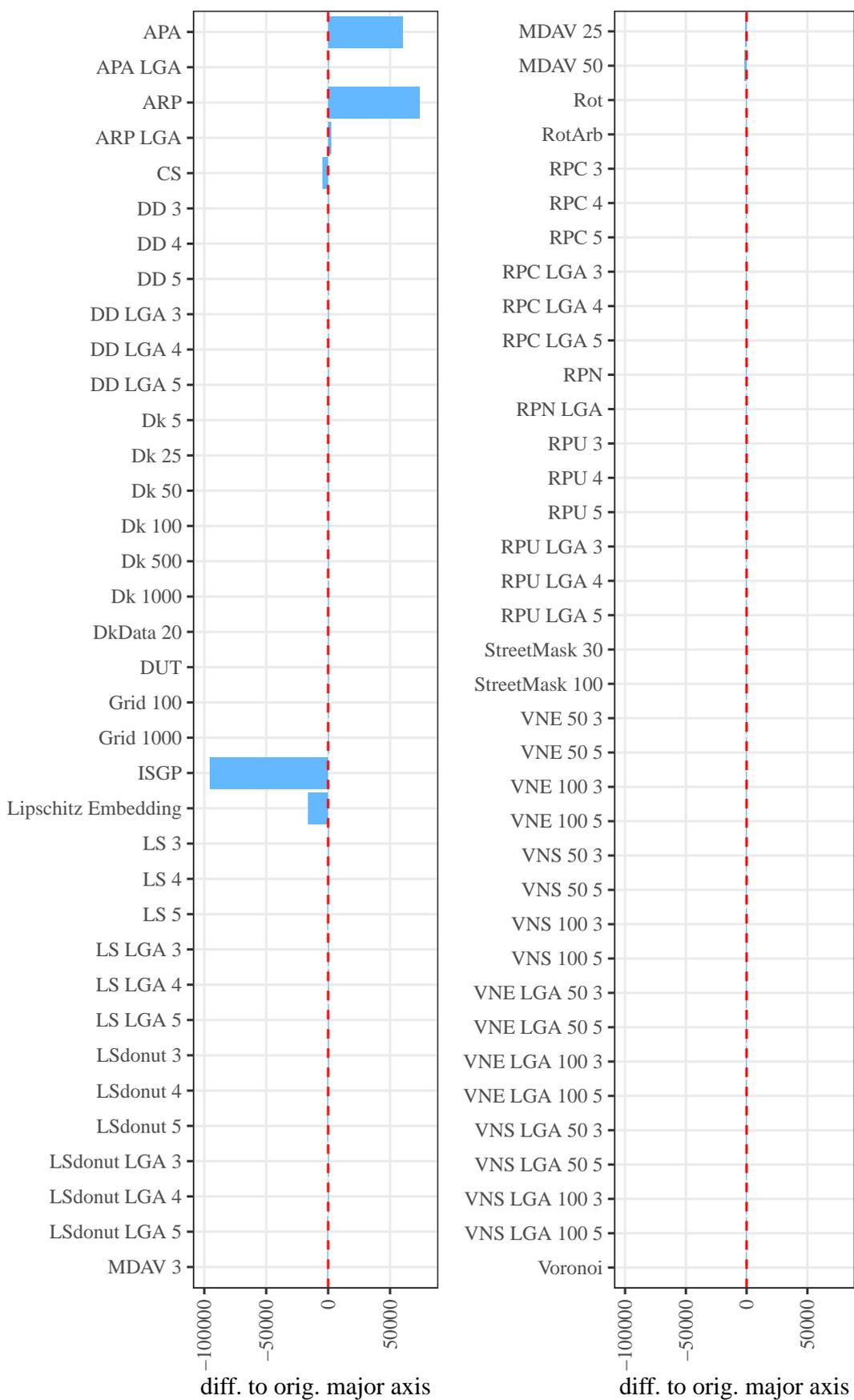


Figure 5.9.: Major axis difference to original (red dashed line).

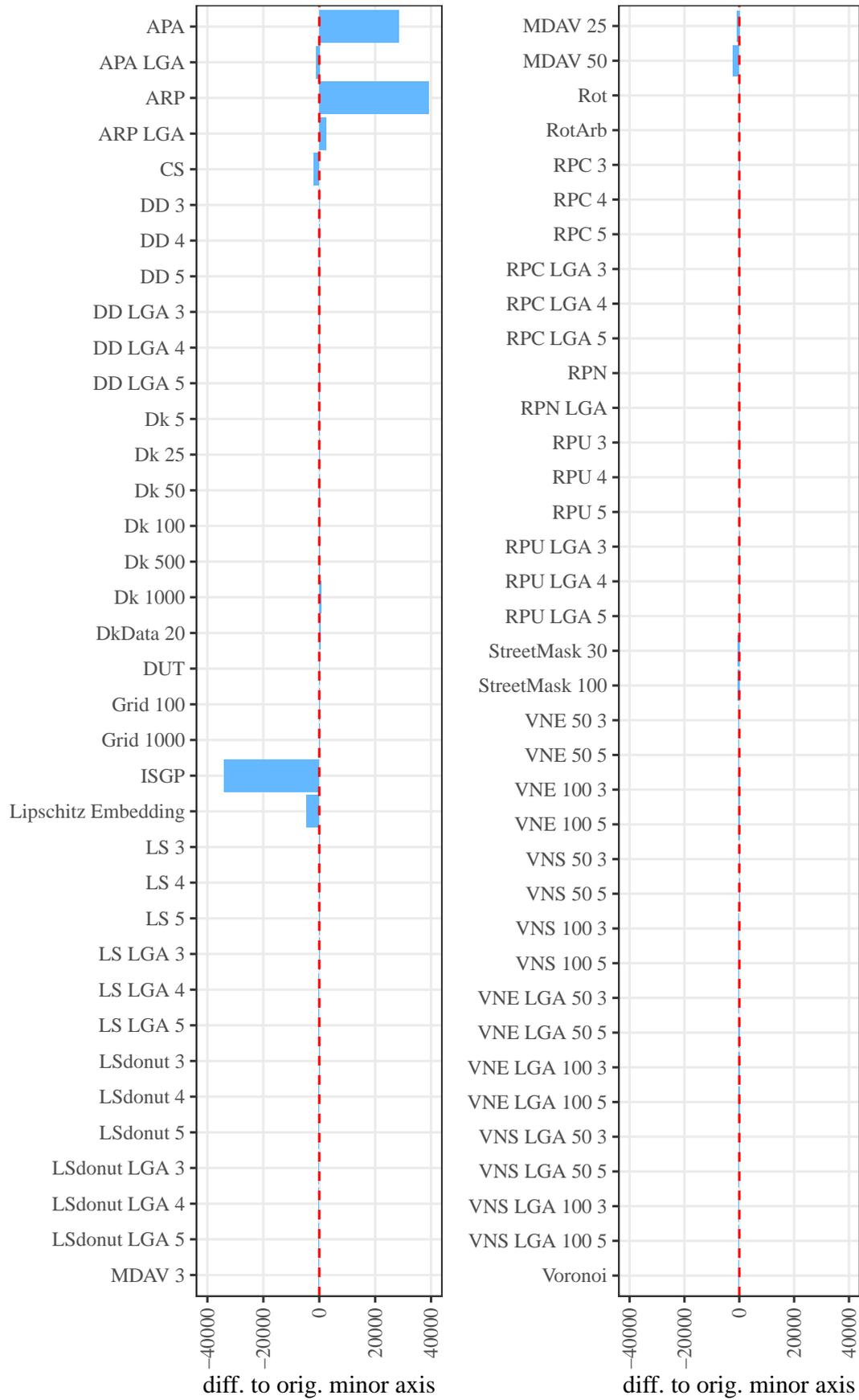


Figure 5.10.: Minor axis difference to original (red dashed line).

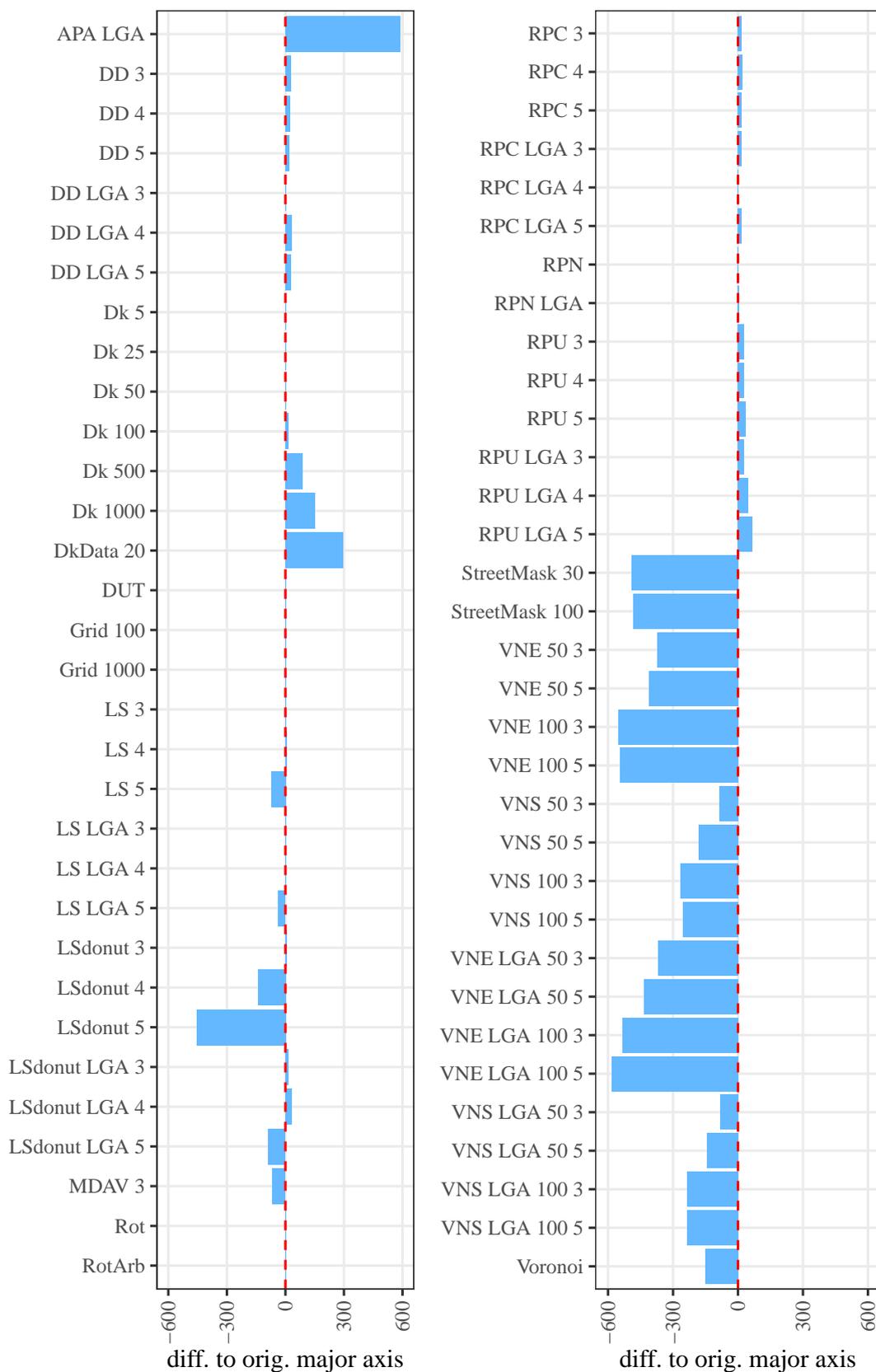


Figure 5.11.: Major axis difference to original (red dashed line) without outliers.

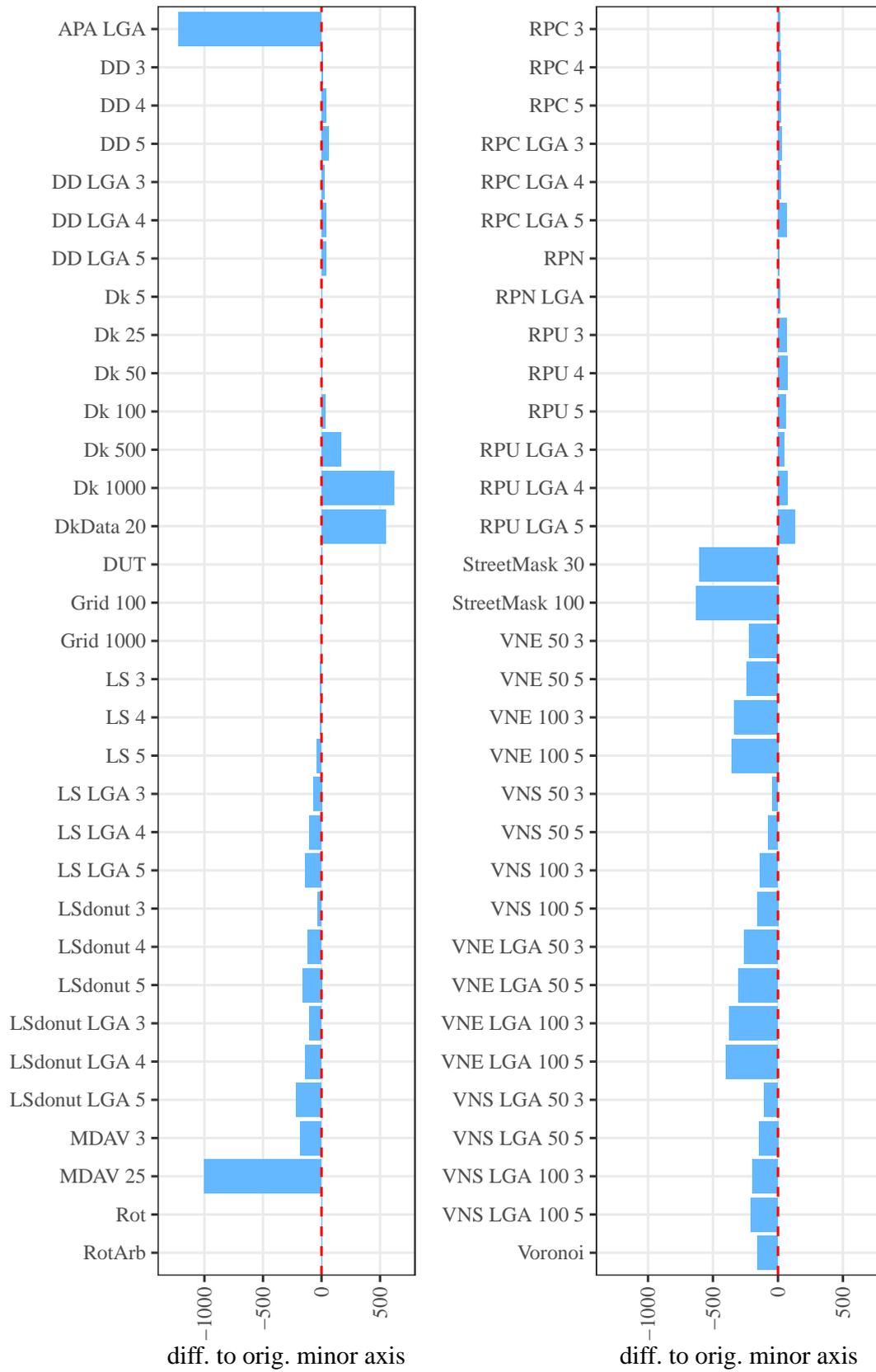


Figure 5.12.: Minor axis difference to original (red dashed line) without outliers.

Donut masking based on population density, RPN, and RPC result in major and minor axes that are close to the original major and minor axes length (difference less than 30 meters for the major axis and 67 meters for the minor axis). The major axis difference is less than 30 meters for donut masking and less than 40 meters for RPN, RPU, and RPC. Only for RPU LGA 4 and RPU LGA 5, the length of the major axis is longer by 65 meters (RPU LGA 5) and 48 meters (RPU LGA 4), respectively. For the minor axis, the difference is slightly larger but still less than 80 meters except for RPU with local government area population density multiplied by 5, which is 131 meters on average.

For  $k$ -nearest neighbor donut masking, the major and minor axes are identical on average for small values of  $k$  (less than 50), with larger values of  $k$  the difference increases. For example, for the major axis the difference to the original major axis is 149 meters for  $k = 1,000$  and 89 meters for  $k = 500$ , and for the minor axis 620 meters for  $k = 1,000$  and 169 meters for  $k = 500$ .

Location swapping with an estimate of the average distance between people multiplied by 3 or 4 shows almost no difference in the length of the major axis to the original axis. When multiplied by 5, the difference is about 73 meters (38 meters for LGA population density). The donut variant of location swapping shows larger differences in length for multipliers of 4 or more, e.g., multiplying the estimate of the average distance between people by 4 shows a difference to the original major axis by 141 meters on average (30 meters for LGA). The minor axis is always smaller than the original minor axis, with an increasing difference if local government areas are used compared to postcode areas.

The verified neighbor approach shows even greater differences, up to 600 meters for the major axis and 400 meters for the minor axis. The difference always causes a shortening of the axes. For street masking, the amount of shortening of the axes is the opposite of the verified neighbor method. The major axis is 480 meters smaller than the original, while the minor axis is 600 meters shorter (630 meters for depth = 100). Figures 5.11 and 5.12 also show how larger possible displacement distances increase the difference in the length of the axes.

In MDAV, the higher the number of clustered points, the shorter the average difference of the major and minor axis. For cluster size 50, the major axis is 1,738 meters shorter, and the minor axis 2,208 meters compared to a shortening of 66 meters for the major axis and 180 meters for the minor axis for cluster size 3. Lastly, Voronoi masking shortens the major axis by 151 meters and the minor axis by 159 meters.

### 5.2.2. Preserving Distances

Preserving distances can be divided into two aspects: preserving the coordinates' distances from each other and keeping the distance between the new location of the coordinates and the original location small so as not to cause too much change

when additional information should be assigned. While the main interest is on the difference of the mean distance between points, the median distance is also considered. Furthermore, the variation of the mean and median distance between the fifty replications are also taken into consideration.

### 5.2.2.1. Distance Between Coordinates of the Data Sets

The average distance between points in the original data set is 104,527.220 meters (with a median distance of 36,225.976 meters). Figure 5.13 shows the mean and median distance of the masked coordinates, averaged over the fifty replications. The dashed line indicates the original distance. Table G.7 in appendix G.5 shows the corresponding difference from the original mean and median distance and the standard deviation of the fifty replications.

The adaptive areal elimination methods using state electorate polygons increase the mean distance between points by about 47.6 km (APA) and 52.6 km (ARP). In contrast, the median distance between points is increased by 1.2 km (APA) and 3.3 km (ARP). If local government area polygons are used, the differences in the mean distance is only 335 meters for APA LGA and 2.3 km for ARP LGA. The median distance differs by 1.4 km (APA LGA) and 2.7 km (ARP LGA). For anonymization of distance matrices via Lipschitz embedding, the difference to the mean distance is 11.4 km and to the median distance only 123 meters, and the standard deviation of the mean distance is large (2 km). Both distances were shortened as opposed to the enlargement of the mean and median distance when using APA and ARP. Distance approximation using ISGP shows similar effects as anonymization of distance matrices via Lipschitz embedding. The median distance is only decreased by 126 meters. However, the mean distance is decreased by 60.2 km. Also, the standard deviation is near zero.

For rotation, the difference to the original masking method is due to the fact that the rotation around the coordinate system's origin is done with the coordinates in meters. As the points have been moved to a different area of the globe and are converted from latitude and longitude to easting and northing and back, differences in distances occur. If the coordinates remain in the same area as when rotating around the spatial mean center, distances are (nearly) preserved. Change of scale also shows large differences on average to the mean and median distance, i.e., on average 5.1 km for the mean distance and 1.7 km for the median distance. Again, this strongly depends on the multiplier as seen by the large standard deviation of the fifty replications.

A closer look at the average of the fifty replications of the mean distance between points without the aforementioned masking methods (see figures 5.14 and 5.15) shows that most masking methods preserve the mean and median distance between points well. Moreover, the median distance varies much less between replications (see table G.7 in appendix G.5).

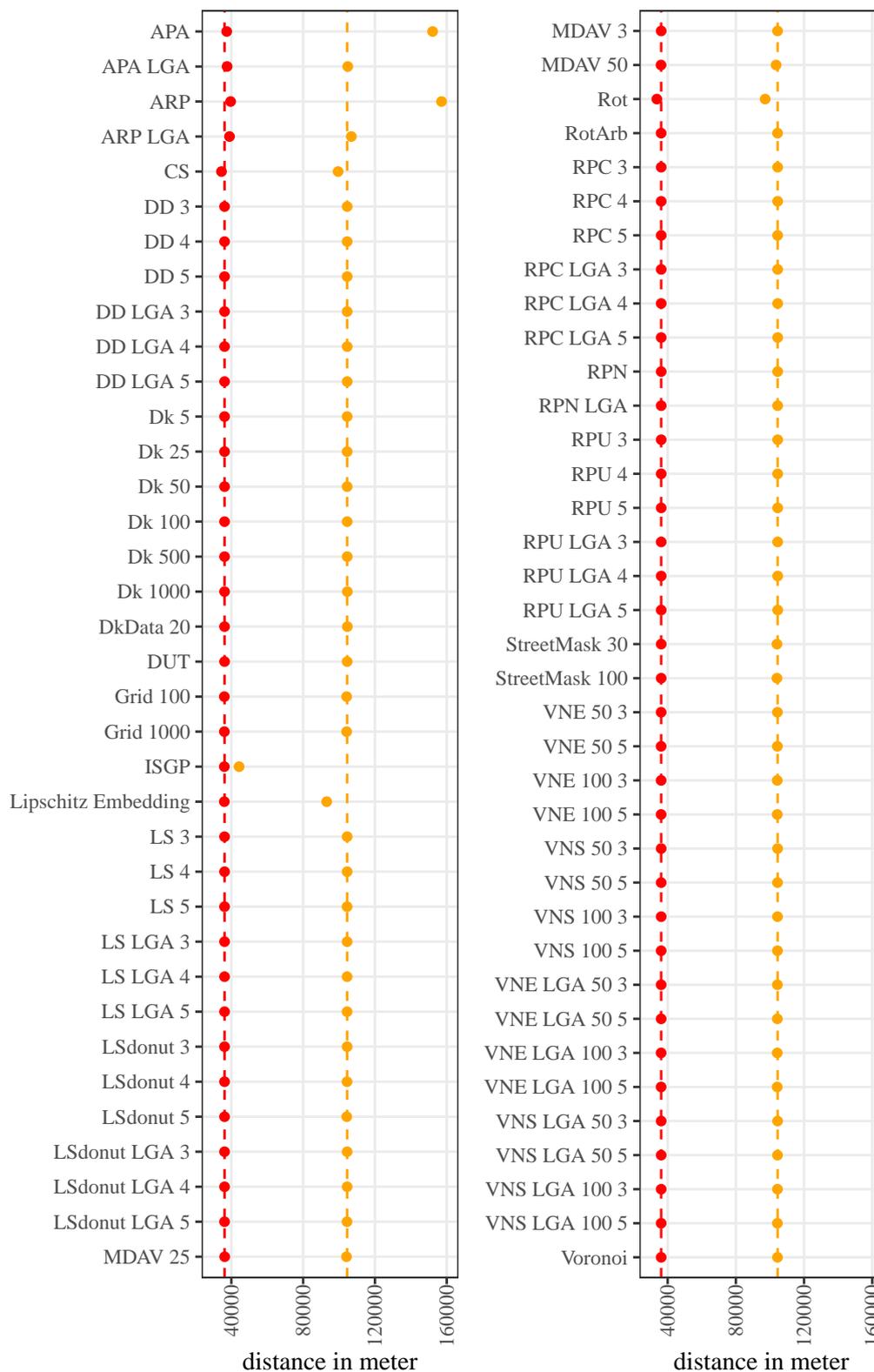


Figure 5.13.: Average mean (orange) and median (red) distance between points over fifty replications. Dashed line shows value of the unmasked data.

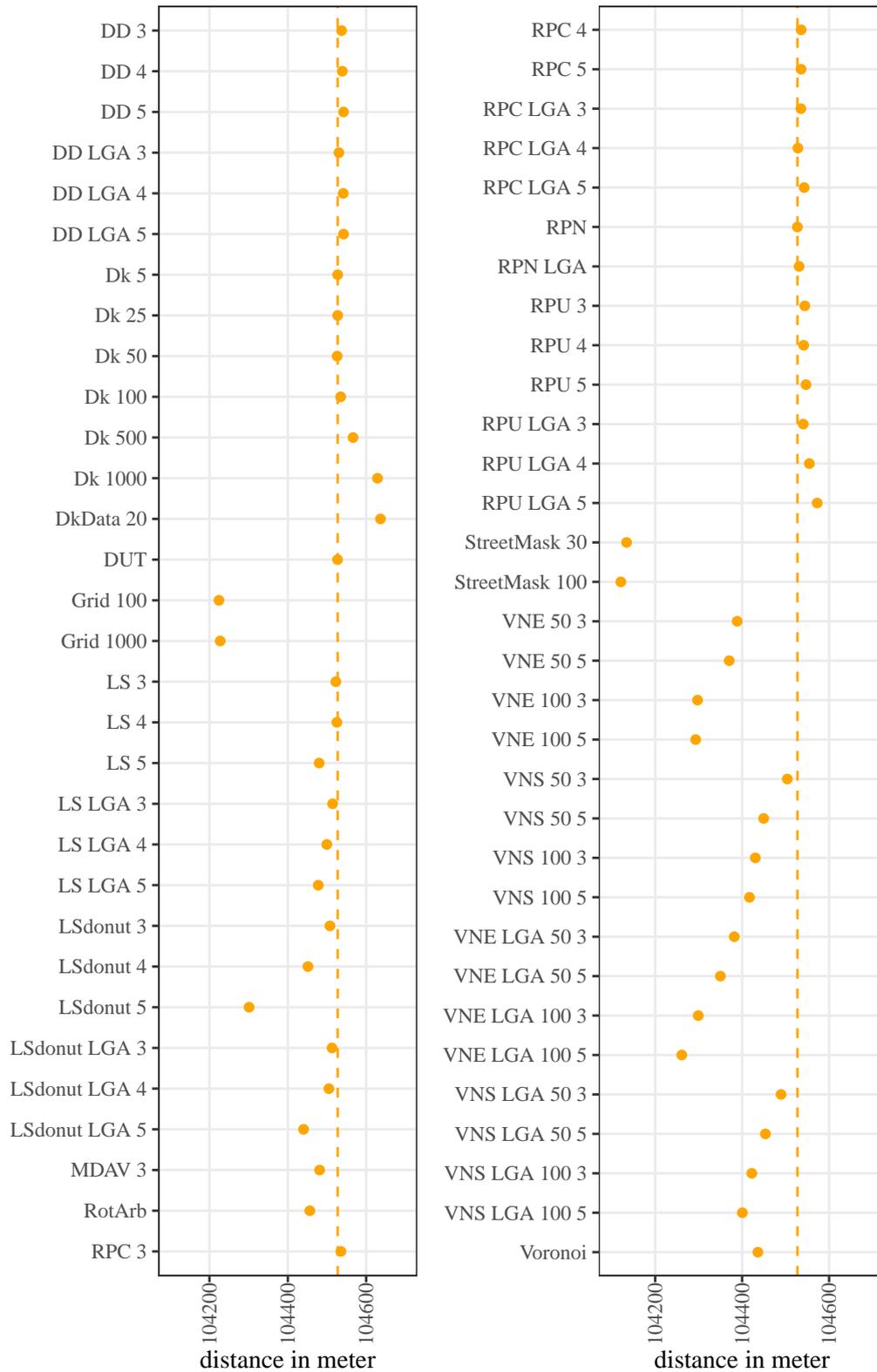


Figure 5.14.: Mean distance between points averaged over fifty replications without outliers. Dashed line shows value of the unmasked data.

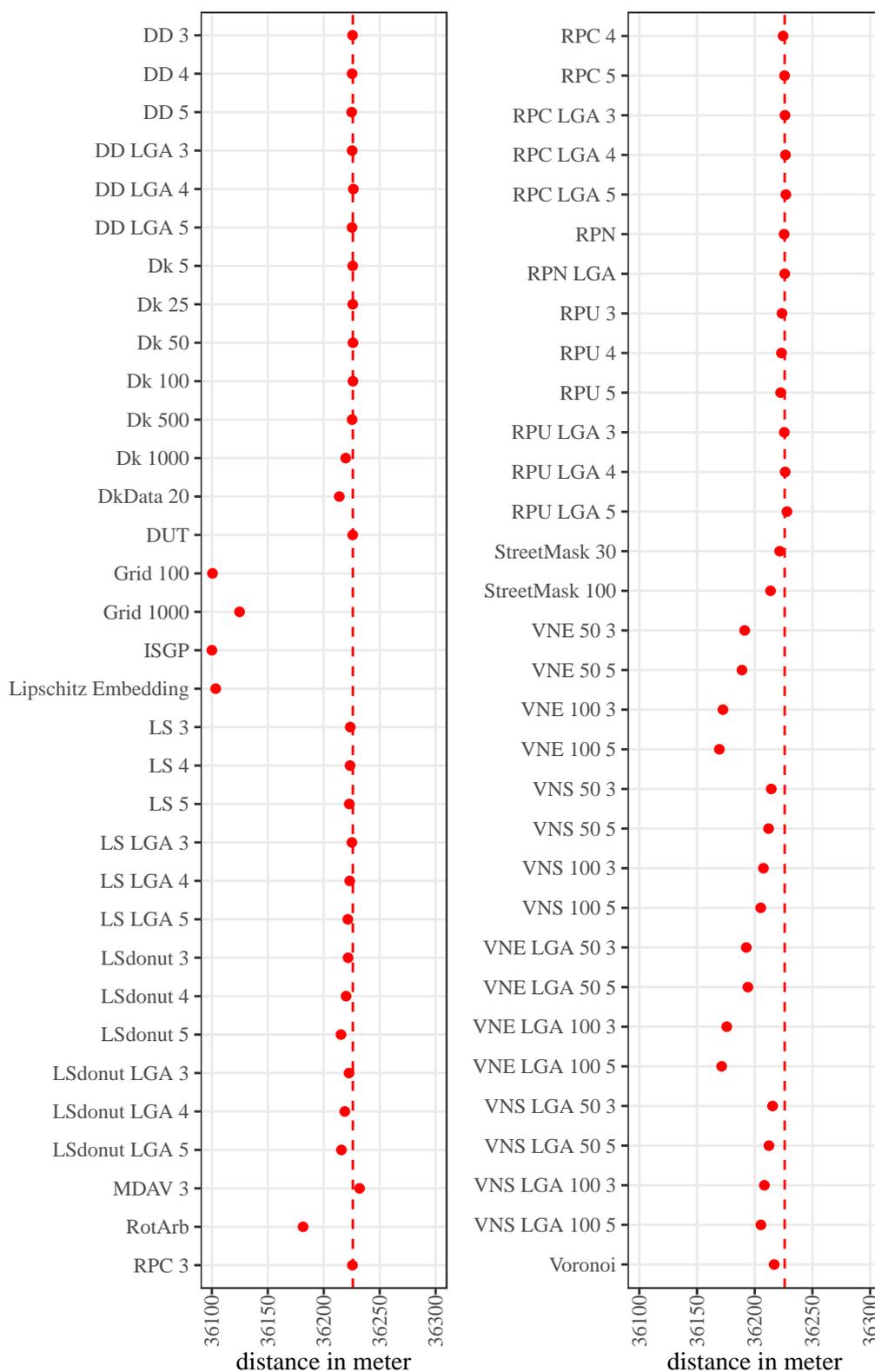


Figure 5.15.: Median distance between points averaged over fifty replications without outliers. Dashed line shows value of the unmasked data.

However, it can be seen that allowing larger displacement distances result in larger differences in distances. For example, the verified neighbor method with a  $k$ -anonymity

of 50 versus 100 shows a difference in the mean distance of about 100 meters (difference to original is up to 266 meters for  $k = 100$ ). The median distance does not show such extreme behavior, but differences in the median distance between points are still seen for large values of  $k$ . This is also true for the selected variable for the characteristics. The influence of the chosen parameter is again evident when considering the  $k$ -nearest neighbor donut masking method. The increase of the possible displacement distances and the increase of the minimum radius lead to comparatively larger differences from the original mean respectively median distance (less than one meter for  $k = 5$  and 102 meters for  $k = 1,000$  for the mean distance).

The official statistics grid lowers the average distance between points by about 300 meters and the median distance by 100 (1,000 meters grid) to 125 meters (100 meters grid). MDAV shows a larger median distance for cluster size 25 (89 meters larger) and a smaller median distance for cluster size 50 (102 meters smaller).

The mean distance for MDAV is shortened by 366 meters for cluster size 25 compared to 919 meters for cluster size 50. Using street masking causes a smaller mean distance between points (about 400 meters). However, for the median distance the difference is small, 4 meters for depth = 30 and 12 meters for depth = 100.

For random projection, the median and mean distance calculation is not comparable to the other masking methods because the distance matrix can only show string distances and not in a unit of length.

### 5.2.2.2. Distance Between Original and Masked Coordinates

When coordinates are masked, they are moved to a different location. Ideally, points in urban areas are not moved very far, while points in rural areas are moved a little further but still at a reasonable distance. Also, a masking method should not have a high deviation in the distance by which the points are moved between multiple replications.

Therefore, the distance between each original and masked point is calculated for each replication to see how far the points are moved (average and median) and at the same time to see how the result may vary. Figures 5.16-5.24 show, grouped by masking methods, the mean and median distance by which the masked coordinates were moved from their original location, with the results for each replication sorted in ascending order showing possible variations between replications.<sup>8</sup> For all figures containing multiple masking methods or multiple variants, the same color was used for the same masking methods and different symbols for different parameter choices, but the symbol was kept the same for the same parameter choice. For example, APA and APA LGA receive the same color as does ARP and ARP LGA, and APA LGA and ARP LGA receive the same symbol as does APA and ARP.

<sup>8</sup>The values of the figures are summarized in table G.8 in appendix G.5. The figures are grouped by masking methods and similar distances.

As expected, the median distance by which the points are moved is much smaller than the mean distance because most masking methods intend to move locations in dense areas much less than in areas with few residential addresses, and more points are located in dense areas. APA always moves the points to the same location; thus, the points' displacement distance is the same for all replications (see figure 5.16). Comparing the use of state electorate polygons to local government areas again shows the importance of choosing an appropriate regional level. For state electorate polygons, the mean distance between original and masked coordinates is more than four times larger than when using local government areas (39.7 km APA vs. 8.5 km for APA LGA). On the other hand, the median distance for local government areas is 1,300 meters larger than for state electorate polygons (3 km APA vs. 4.3 km for APA LGA). The same trends hold for ARP, but the mean and median displacements are larger than in APA. Moreover, ARP moves the points by varying distances.

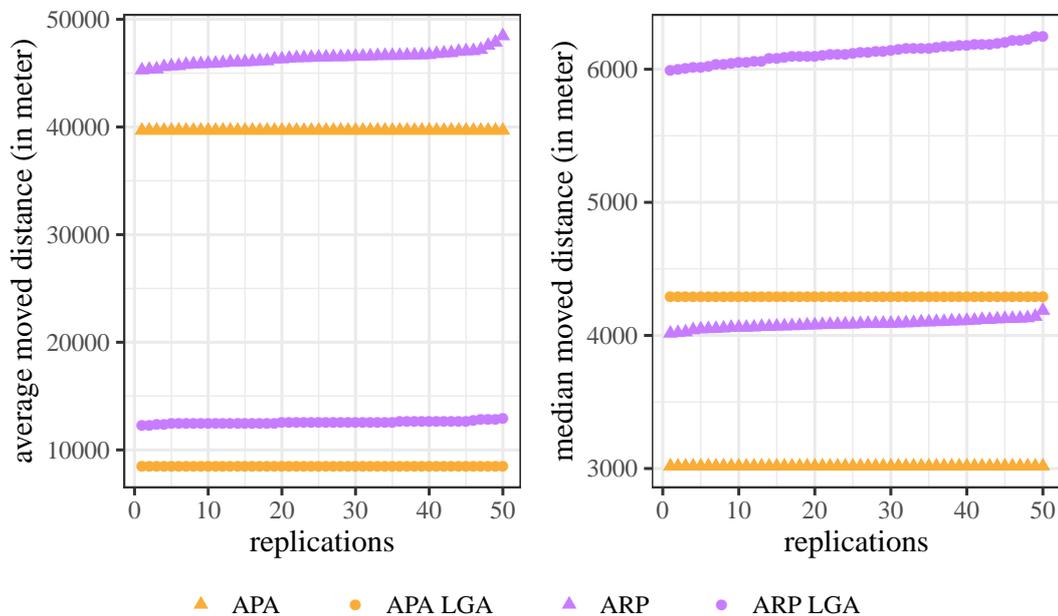


Figure 5.16.: Mean and median distances the points are moved for the adaptive areal elimination methods. Individual points are the fifty replications.

Donut masking using  $k$  (see figure 5.17) again shows that larger  $k$  lead to larger displacements and larger dispersions of displacement distances. Furthermore, as  $k$  increases, the difference between the mean and median displacements increases. The mean and median displacements for  $k = 5$  are close to each other (34 meters and 31 meters on average) and close to zero. For  $k = 1,000$ , the median is about 1/4 of the mean displacement with a median displacement distance of 436 meters and a mean displacement distance of 1,809 meters. It should be noted that for  $k$ -nearest neighbor donut masking, the displacement distance highly varies between data sets and regions considered. Data sets that cover rural areas with a low population density will result in large displacements, even for smaller values of  $k$ .

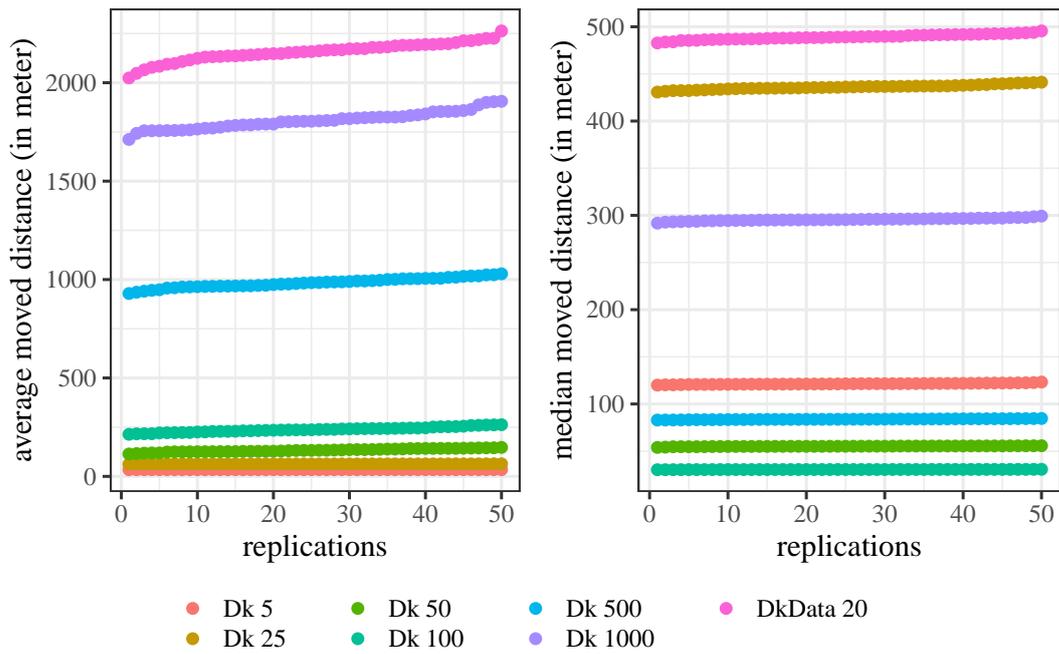


Figure 5.17.: Mean and median distances the points are moved for the donut masking methods using  $k$ . Individual points are the fifty replications.

If donut masking based on population density is used, a distinction must be made between which regional levels are used and whether the mean or median displacement is considered (see figure 5.18).

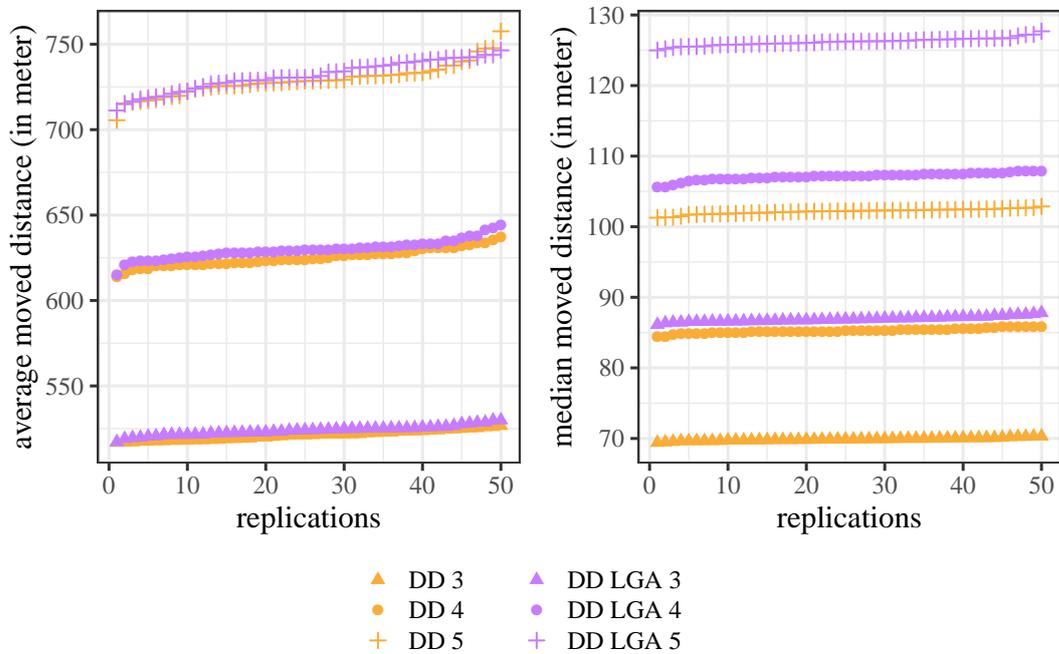


Figure 5.18.: Mean and median distances the points are moved for the donut masking methods using population density. Individual points are the fifty replications.

There is little difference between the regional levels used for the mean distance. Nevertheless, it can be seen that larger allowed displacement distances result in a larger average displacement distance. For the median displacement distance, a difference of about 20 meters can be seen between regional levels for population density (and not just because of the smaller range of the y-axis in figure 5.18).

Figure 5.18 also shows that this can be compensated for by not using larger numbers to multiply the estimate of the average distance between people. Multiplying the estimate of the average distance between people based on local government areas by 3 yields roughly the same median distance as using postcode areas multiplied by 4. It can also be seen that using the number 5 as a multiplier leads to greater variations in the average moved distance. In contrast, the median moved distance remains the same for the replications.

RPC (see figure 5.19) always leads to smaller mean and median displacements than donut masking using population density since no minimum distance is defined. The comparison between RPC to RPU shows that RPU yields mean and median displacement distances, which are about twice as large as for RPC. For example, for multiplying the estimate of the average distance between people based on postcode level by 3, a mean displacement distance of 315 meters can be seen for RPC and 740 meters for RPU. The increase in displacement distance when using larger numbers to multiply by is also slightly larger for RPU than for RPC. It can once again be observed that there is little difference between population density based on postcode and based on local government areas, when focusing only on the average distance. The median displacement distances show that local government areas result in greater displacements than postcode population density. This can be regulated by the number chosen to multiply by. The average and median moved distance is relatively stable for all replications.

Using location swapping to mask coordinates results in small displacements on average compared to random perturbation and donut masking. But more variation in the average moved distances is seen between replications (see figure 5.20), especially when larger radii have been considered. The median displacement is again smaller and shows less variation. Similar to RPC, location swapping based on local government areas population density multiplied by 3 yields approximately the same values as location swapping based on postcode population density multiplied by 4. Setting a minimum displacement as in the donut variant of location swapping leads to further displacements for the average as well as median moved distance.

Verified neighbor methods' mean displacement distance increases by about 100 meters when changing the multiplier (see figure 5.21) and up to 400 meters when increasing the minimum number of points with the same characteristics. Comparing employment and sex as a variable for selecting individuals with the same characteristic also shows how increasing the number of categories for a variable increases the displacement distance of the points.

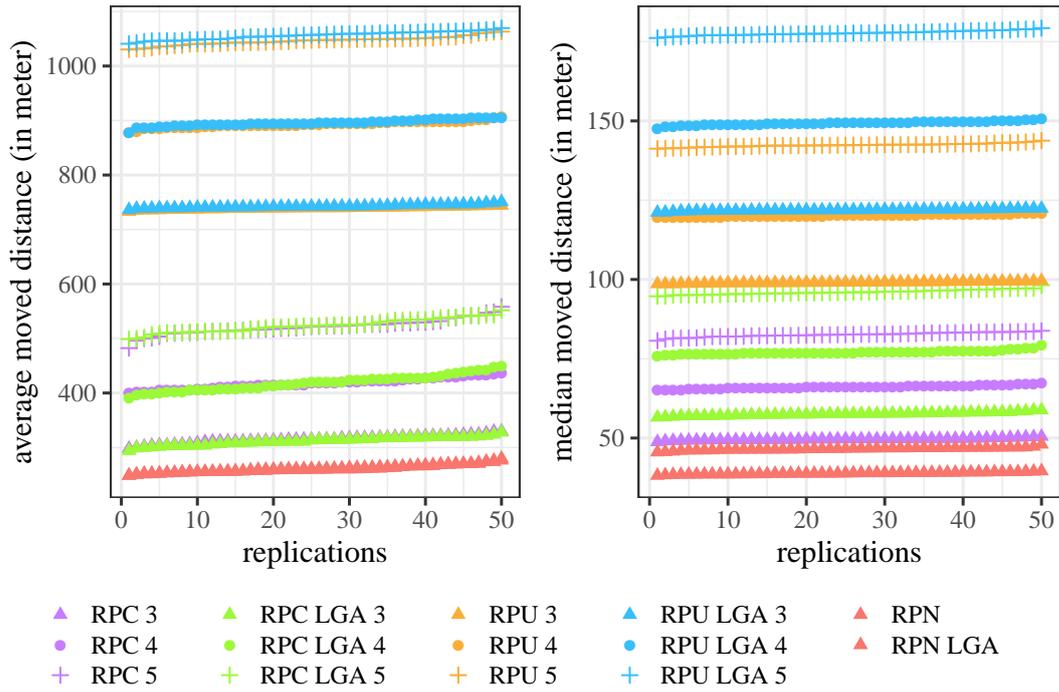


Figure 5.19.: Mean and median distances the points are moved for the random perturbation methods. Individual points are the fifty replications.

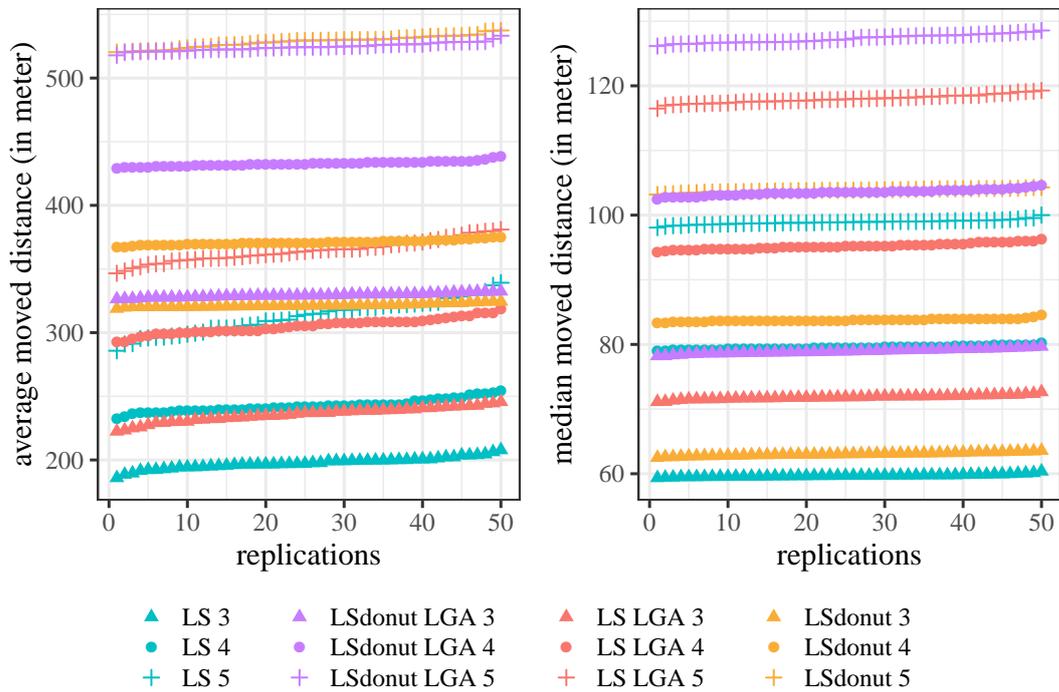


Figure 5.20.: Mean and median distances the points are moved for the location swapping methods. Individual points are the fifty replications.

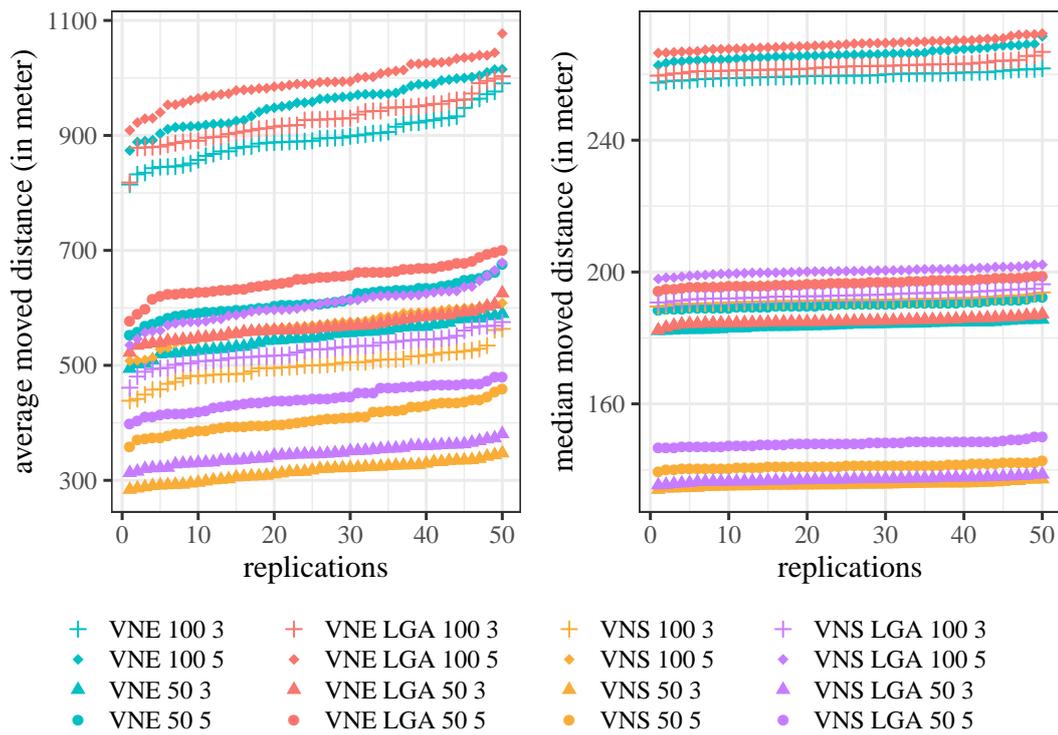


Figure 5.21.: Mean and median distances the points are moved for the verified neighbor methods. Individual points are the fifty replications.

Voronoi masking, the official statistics grid, and microaggregation are masking methods that show no variation between replications (see figure 5.22). For the given data, the average displacement distance lies at 178 meters for Voronoi masking (median 65 meters). For the official statistics grid, the average displacement lies at 38.25 meters for cell size 100 meters (383 meters for cell size 1,000 meters) and the median displacement at 40 meters for cell size 100 meters (398 meters for cell size 1,000 meters).

For MDAV (see figure 5.22), the results also depend on the cluster size considered. Larger cluster sizes result in larger average and median displacements. Also, large differences between mean and median displacements can be seen. For the smallest possible cluster size (3), the mean distance by which points are displaced is about 800 meters, while the median distance is 153 meters. In comparison, the mean distance is about 6,875 meters, and the median distance is about 1,000 meters for the largest cluster size considered (50).

Street masking shows almost no variation between replications. On average, points are moved by 524 meters for depth = 30 and 730 meters for depth = 100. The median moved distance is 170 meters and 296 meters, respectively.

Rotation (around the origin) and change of scale show large mean and median displacement distances. The magnitude of the displacement is highly dependent on the angle and the multiplier chosen (see figure 5.23). However, for both, the mean

and median moved distance is very similar.

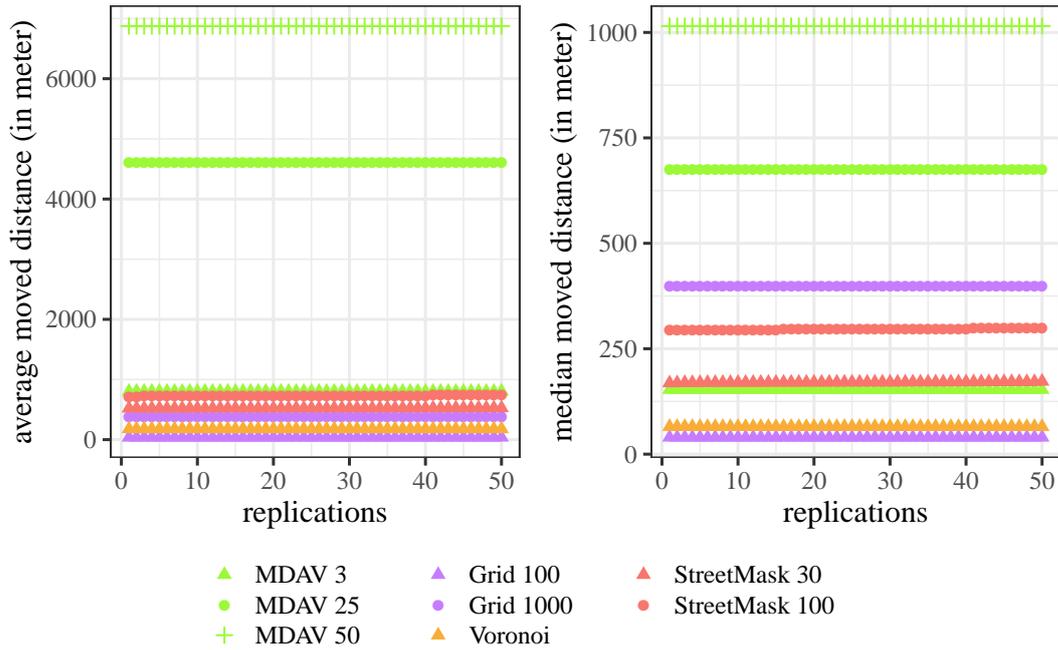


Figure 5.22.: Mean and median distances the points are moved for official statistics grid, microaggregation as well as Voronoi masking. Individual points are the fifty replications.

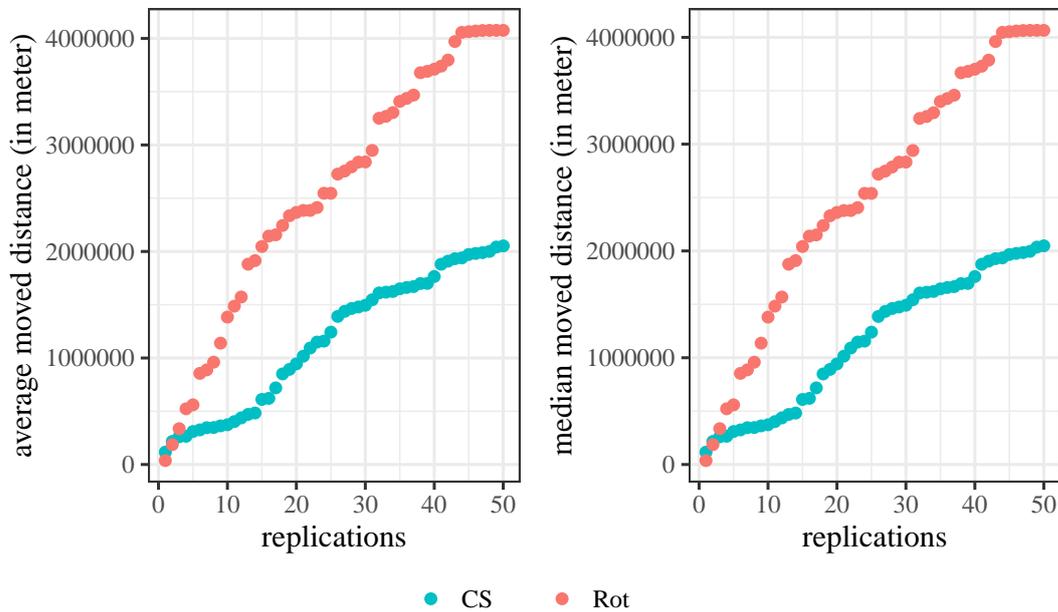


Figure 5.23.: Mean and median distances the points are moved for change of scale and rotation. Individual points are the fifty replications.

A rotation around the spatial mean center (see figure 5.24) displaces the coordinates on average between 1.1 km and 124.3 km (median between 301 meters and 34.6 km). This is much less than using the origin of the coordinates system for rotation.

Similar tendencies show displacement using translation (see figure 5.24) with an average and median displacement distance between 1.7 km and 12.5 km. Displacement using translation is the only masking method where the mean and median distance is almost identical.

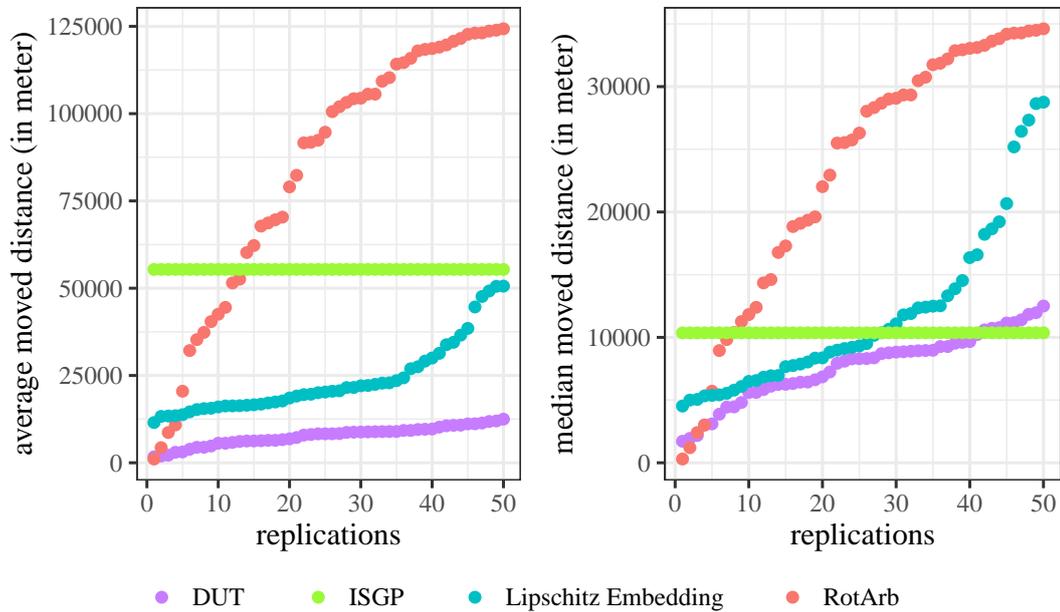


Figure 5.24.: Mean and median distances the points are moved for displacement using translation, rotation around an arbitrary point, intersecting grid points as well as Lipschitz embedding. Individual points are the fifty replications.

Anonymization of distance matrices via Lipschitz embedding and distance approximation using ISGP do not allow calculating the distance between the original and the new location since only the distance matrix is provided. However, using the approximation (MDS and Procrustes analysis), a distance can be calculated. Figure 5.24 shows large deviations in the average and median moved distance of the points for anonymization of distance matrices via Lipschitz embedding (10.3 km for mean distance and 6.5 km for median distance) while the results of distance approximation using ISGP are similar for all replications (standard deviation of less than 2 meters). For random projection, calculation of the distance between original and masked coordinates is not possible.

Instead of just looking at the variation of the average and median moved distance, the consistency of displacement distances for each point and each replication is also evaluated. This is done by calculating the distance between the original and the masked location. After that for each point, the standard deviations of the distances over all replications are calculated to assess whether the points are always moved approximately the same distance when the masking method is applied. While some masking methods always move points to the same masked location (no variation in the displacement distance), other masking methods can displace points by greatly

varying distances or show little variation in the distance by which points are moved.

The detailed results can be found in table G.9 in appendix G.6, which shows the smallest standard deviation (point displaced the most similar over replications), the largest standard deviation (displacement varies strongly over replications), the average standard deviation (average variation of distances the points are moved), and median standard deviation (median variation of distances the points are moved) of the 10,000 points for each masking method.

APA (for state electorates and local government areas), Voronoi masking, the official statistics grid, and MDAV always move the points by the same distance. On the other hand, the rotation method fails in displacing points by approximately the same amount for different replications if the angle is chosen randomly. When the points are rotated around the spatial mean center, there are also large standard deviations in the distances by which the points are moved. The point that shows the least standard deviation in its displacement distance has a standard deviation of 540.6 meters. If points are far from the spatial mean center, even small angles result in large changes in location.

The donut masking methods all show small to moderate standard deviations when looking at the mean and median standard deviation (mean standard deviation 60 to 180). Donut masking based on the population density (postcode and local government areas) shows a linear relationship between multiplying by a larger number and the standard deviation (e.g., 60 meters on average for multiplying by 3, 120 meters for multiplying by 4, 180 meters for multiplying by 5). For  $k$ -nearest neighbor donut masking, there is no linear relationship between increasing  $k$  and increasing standard deviation, but a relationship that could be defined by a logarithmic function. Large standard deviations for the distances between original and masked coordinates are seen with the masking method that uses the  $k$ -nearest points of the data set rather than the residential address file. Again, this can be explained by the increased allowed distance by which points can be moved.

Displacement using translations standard deviation of 2,551.9 meters (mean) is explained by the fact that the distance is defined by a random value between -10 km and 10 km and thus can vary strongly. In location swapping using donut masking, there is a point that is always moved the same distance. This is caused by the fact that if no point is found within the radius based on population density, the nearest point is taken.<sup>9</sup>

RPC shows similar results when the postcode population or the local government area population is used. Most points show small deviations in their distances, but some show larger deviations (mean standard deviation between 180 and 300 meters; median standard deviation between 24 and 50 meters). RPU causes smaller standard

---

<sup>9</sup>Zhang, Freundsuh, et al. (2015) may not have intended this, but since no explanation was given what to do when no or just one point can be found within the radius, the decision was made to choose the nearest point.

deviations in terms of the distance by which the points are moved between replications (mean standard deviation between 59 and 180 meters). RPN also causes less variation in the displacement of the distances of points than RPC (mean standard deviation about 135 meters).

The verified neighbor approach shows, on average, similar results as RPC. Because at least  $k$  individuals with the same characteristics must be within the considered region, no point is always moved to the same location. If the radius does not contain another resident, it is enlarged until it does. Street masking also shows that points are moved by very similar distances when several replications are compared. For the distance approximation using ISGP and the anonymization of distance matrices via Lipschitz embedding again, only the approximated coordinates can be used. For the latter, this could be the reason for the large deviations of the displacement distances.

### 5.2.3. Spatial Autocorrelation

The overall assessment of the spatial autocorrelation is evaluated using Moran's I. In addition to the coordinates needed for the weight variable, a variable is required to assess spatial autocorrelation. For the given data, the proportion of single households and the proportion of full-time employed people was assigned based on the Statistical Area 1 level since a continuous variable is difficult to find for the used data set at the coordinate level. For the proportion of single households, Moran's I is 0.129, and for the proportion of full-time working people, it is 0.170. Both values indicate that the overall data for both variables show a slight tendency to a clustered data set. Both values are statistically significant ( $p < 0.01$ ) and have an expected value of -0.00010001.

The difference of the Moran's I value to the original value is shown in figures 5.25 and 5.26. Additionally, the dashed orange line shows a Moran's I of zero. The mean, standard deviation of the mean, minimum, and maximum of the Moran's I values for each masking method for all replications is shown in tables G.10 and G.11 in appendix G.7. All masking methods obtain approximately the original Moran's I for both variables, as seen from the near-zero standard deviations. Furthermore, they all yield a positive Moran's I value.

For the variable proportions of single households, in contrast to the results of the original data, APA and ARP show that the variable is randomly dispersed (random dispersion equals a Moran's I of zero indicated by the orange dashed line). This is also true for MDAV. Even the minimum cluster size of 3 already substantially reduces Moran's I (0.09).

The masking methods change of scale, displacement using translation, and rotation preserve, on average, the original Moran's I. Donut masking in the  $k$ -nearest neighbor variant and street masking result in a Moran's I close to the original value.

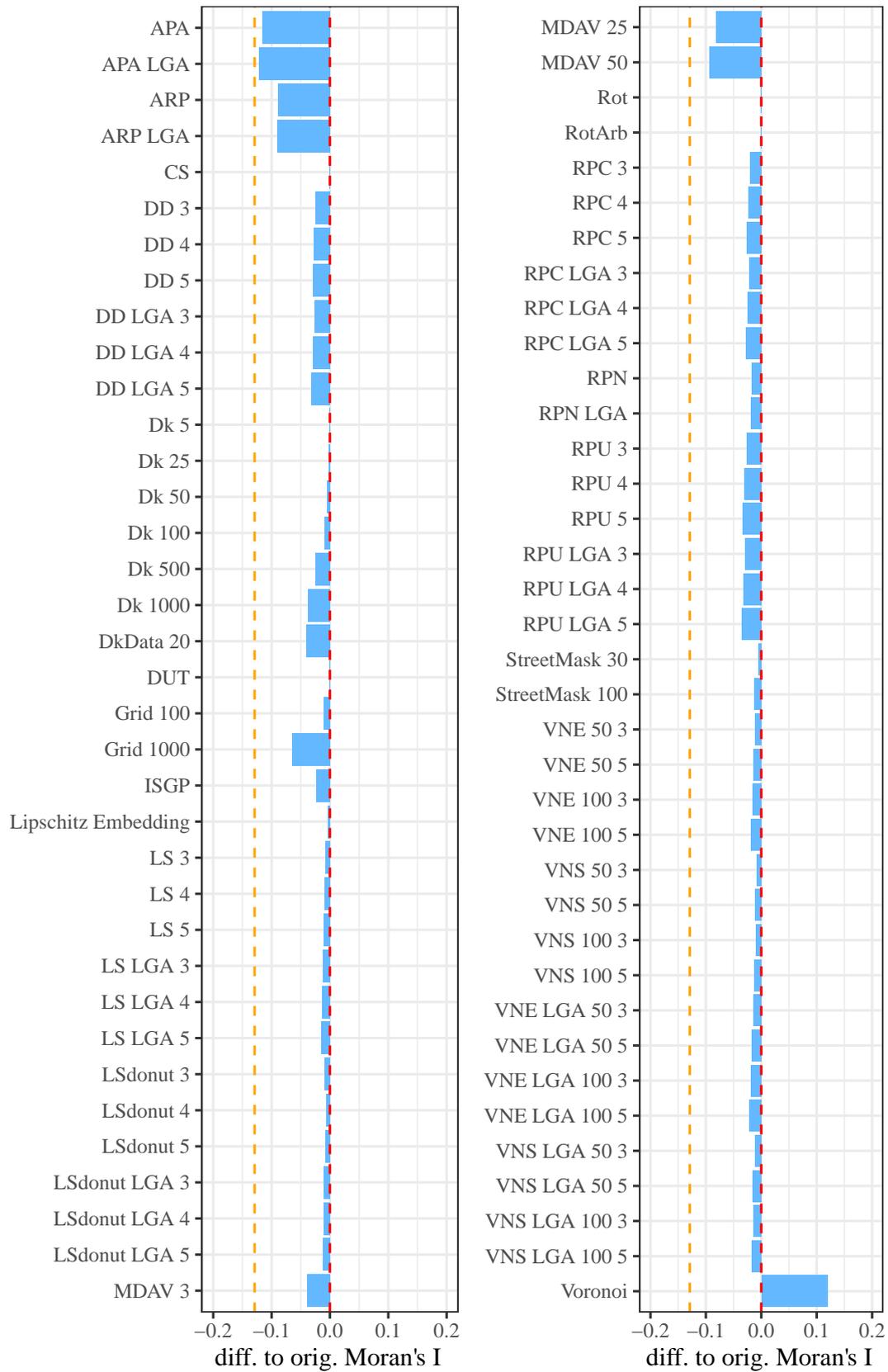


Figure 5.25.: Results of Moran's I values for proportion of single households as difference to original value (red dashed line). Orange dashed line shows random dispersion (Moran's I equals 0).

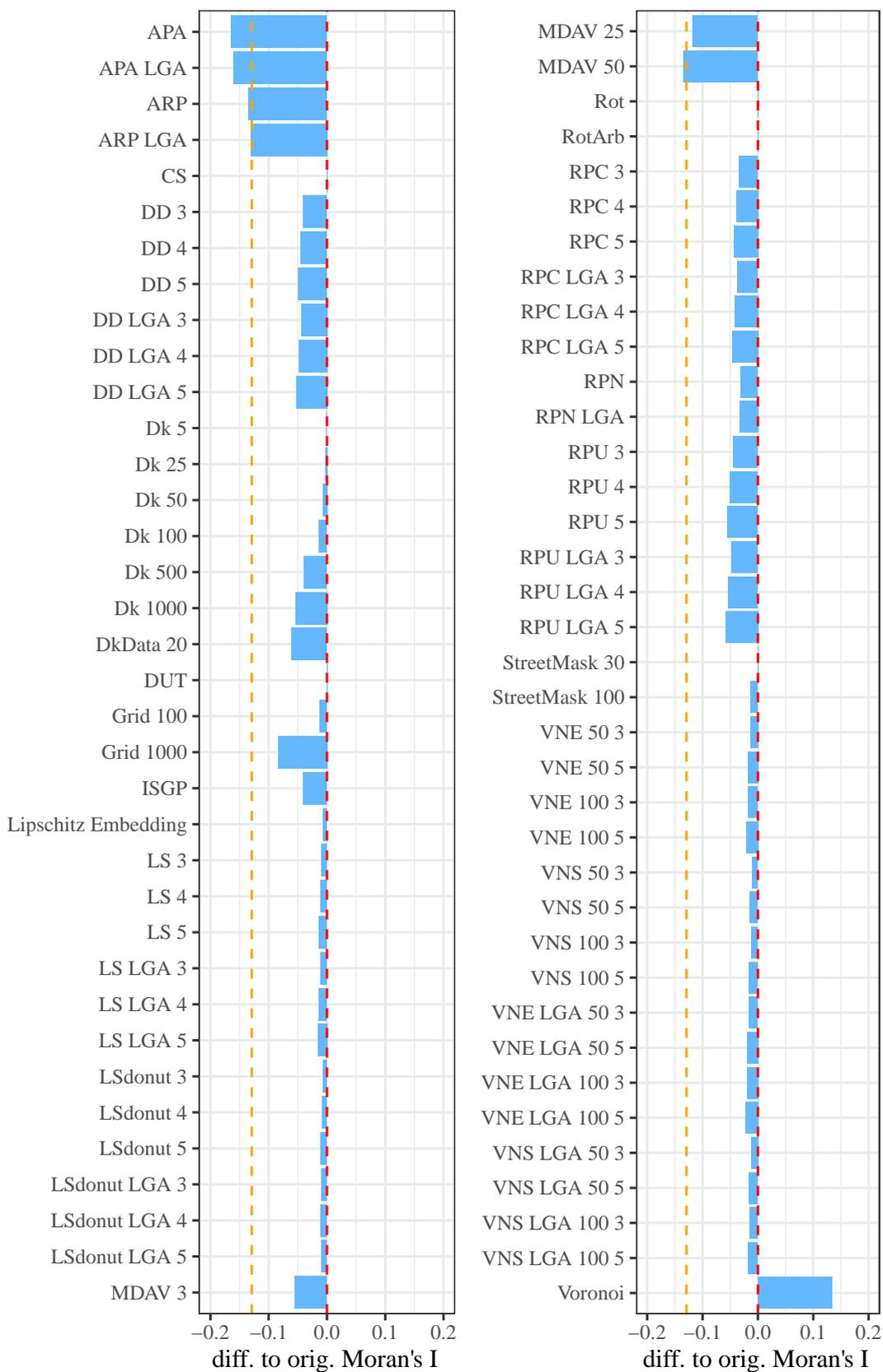


Figure 5.26.: Results of Moran's I values for proportions of full-time working people (right) as difference to original value (red dashed line). Orange dashed line shows random dispersion (Moran's I equals 0).

It can also be seen that as  $k$  and the depth value increase, the Moran's I value decreases, e.g., for  $k = 5$  Moran's I of 0.128 and  $k = 1,000$  Moran's I of 0.092. Using the underlying population density in donut masking, Moran's I is slightly lower (0.10) than the original value. There is a minor difference between using postcode population density and local government areas (LGA results in a value 0.01 smaller).

Random perturbation methods are similar to the donut masking method that uses population density. Compared with donut masking, the Moran's I value is closer to the original Moran's I value. The regional levels show the same small differences as in donut masking using population density. In location swapping, the Moran's I value is nearly preserved, as it is with the verified neighbor approach. Different variables, multipliers, and different minimum population sizes do not affect the resulting Moran's I value.

The anonymization of distance matrices via Lipschitz embedding and the distance approximation using ISGP yield only distance matrices. However, these are sufficient for the calculation of Moran's I since only the distances between points are needed. Here it can be seen that the Moran's I value is preserved when using the anonymization of distance matrices via Lipschitz embedding (0.125) but underestimated when using distance approximation using ISGP (0.106).

Like APA and ARP, Voronoi masking shows larger deviations from the original Moran's I, but in the direction of higher clustering (0.248). This is easily explained by the fact that points in non-dense areas are moved much closer to other points.

Similar results are seen for the variable full-time employees as for the single households variable. The variation of Moran's I between replications is slightly larger than with the variable proportions of single households.

#### 5.2.4. Clustering

In addition to spatial autocorrelation, a closer look is taken at the number and size of clusters. As mentioned in the previous chapters, the DBSCAN algorithm (Ester et al., 1996) is used because it does not require a predefined number of clusters, allows arbitrarily shaped clusters, and allows points to be non-clustered (noise points). DBSCAN requires two input parameters: the number of minimum points (MinPts) and the radius ( $\epsilon$ ), which evaluate whether a point is a core point and is eligible as a starting point.

As suggested by Ester et al. (1996, p. 230), the number of minimum points is set to 4, and a  $k$ -dist graph is used to define the radius ( $\epsilon$ ). The radius is found by finding the "valley" in the graph, where there is a sudden increase in the distances. The graph shows that there are two sudden increases in distances: the first at 3,200 meters and the second at 9,500 meters. Consequently, the radii 3,200 meters and 9,500 meters were used (see appendix B for a detailed explanation of the chosen value for the radius ( $\epsilon$ ) and for the graph itself). The same input parameters (MinPts = 4;  $\epsilon = 3,200$  and

$\varepsilon = 9,500$ ) were used to evaluate the clustering for the masking methods to assess whether the input parameters were preserved. An evaluation for each replication of each masking method would be too time-consuming as it would have to be done manually.<sup>10</sup>

The original data with input parameters  $\text{MinPts} = 4$  and  $\varepsilon = 3,200$  yield 130 clusters in the data, and 248 points remain non-clustered. Some clusters contain only the minimum or a few more points than the minimum. The total number of clusters as well as the number of clusters containing at least 10 points, 20 points, and 30 points are, therefore, also considered.<sup>11</sup> For the original data, the number of clusters containing at least 10 points is 60. 30 clusters contain at least 20 points, and 17 clusters contain at least 30 points.

When the radius is increased to 9,500 meters, the total number of clusters decreases to 97, and 134 points remain non-clustered (noise points). There are 53 clusters with at least 10 points, 29 clusters contain at least 20 points, and 16 clusters contain at least 30 points. Reducing the radius leads to more non-clustered points because they are not within the radius of one another. However, larger radii lead to fewer clusters because the larger radius allows groups of points in close proximity to be viewed as one cluster. The number of clusters with at least 30 points differs only by one.

Figures 5.27 and 5.28 show the results of the DBSCAN clustering algorithm for the different masking methods. More detailed information can be found in tables G.12 to G.13 in appendix G.8. As each masking method was replicated 50 times, the average, standard deviation, minimum, and maximum number of clusters were obtained for the total number of clusters, as well as for the clusters with at least 10, 20, and 30 points, and the number of points that were not clustered. Further, table G.14 in appendix G.8 shows a comparison of the number of points that are clustered in the original clustering but are noise points in the masked data set and vice versa. Below, the results are described for each masking method.

At first glance, it can be seen that most masking methods show results close to the original clustering regarding the total number as well as the number of clusters with at least 30 points. Displacement using translation and the rotation around the spatial mean center preserve the clustering of the data. The official statistics grids with 100 meter grid size, distance approximation using ISGP, anonymization of distance matrices via Lipschitz embedding, and  $k$ -nearest neighbor donut masking with a  $k$  of 5, 25, and 50 on average preserve the clustering.

<sup>10</sup>It should be noted that DBSCANs input is the coordinates in a system allowing for Euclidean distance or a distance matrix. Distances based on easting-northing-coordinates may have minor differences to the distances based on latitude-longitude coordinates. Also, anonymization of distance matrices via Lipschitz embedding and distance approximation using intersecting sets of grid points only yield distance matrices. Therefore, the distance matrix was calculated and used as input to the DBSCAN algorithm for all masking methods.

<sup>11</sup>The decision to focus on clusters with up to 30 points was made because cluster sizes increase slowly up to about 30. Then only a few clusters with rapidly increasing sizes remain.

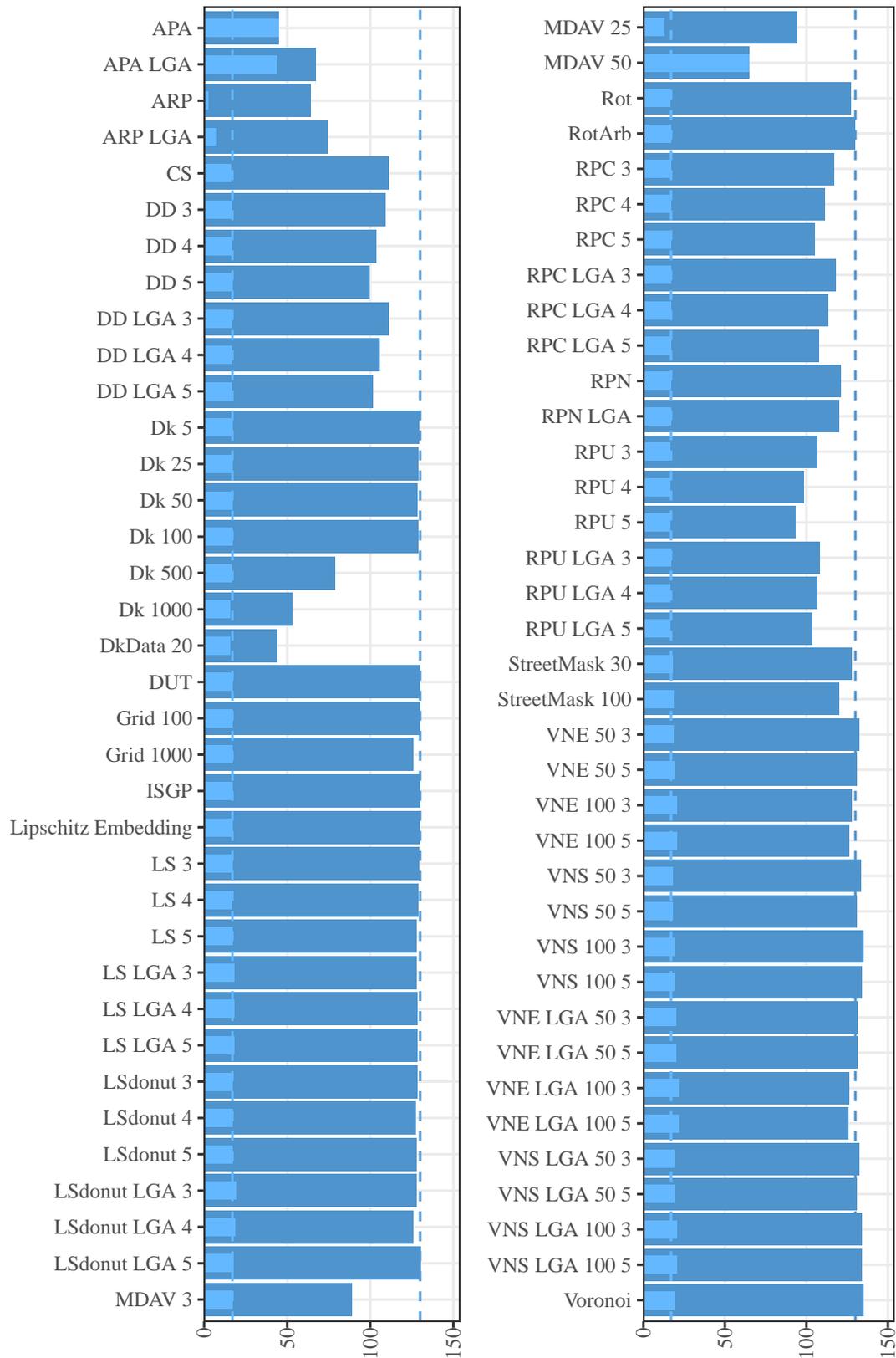


Figure 5.27.: DBSCAN ( $\epsilon = 3, 200$ ): Total number of clusters as well as number of clusters with at least 30 points (light blue bars). Dashed lines shows the number of clusters in the original data (total: 130; larger 30: 17).

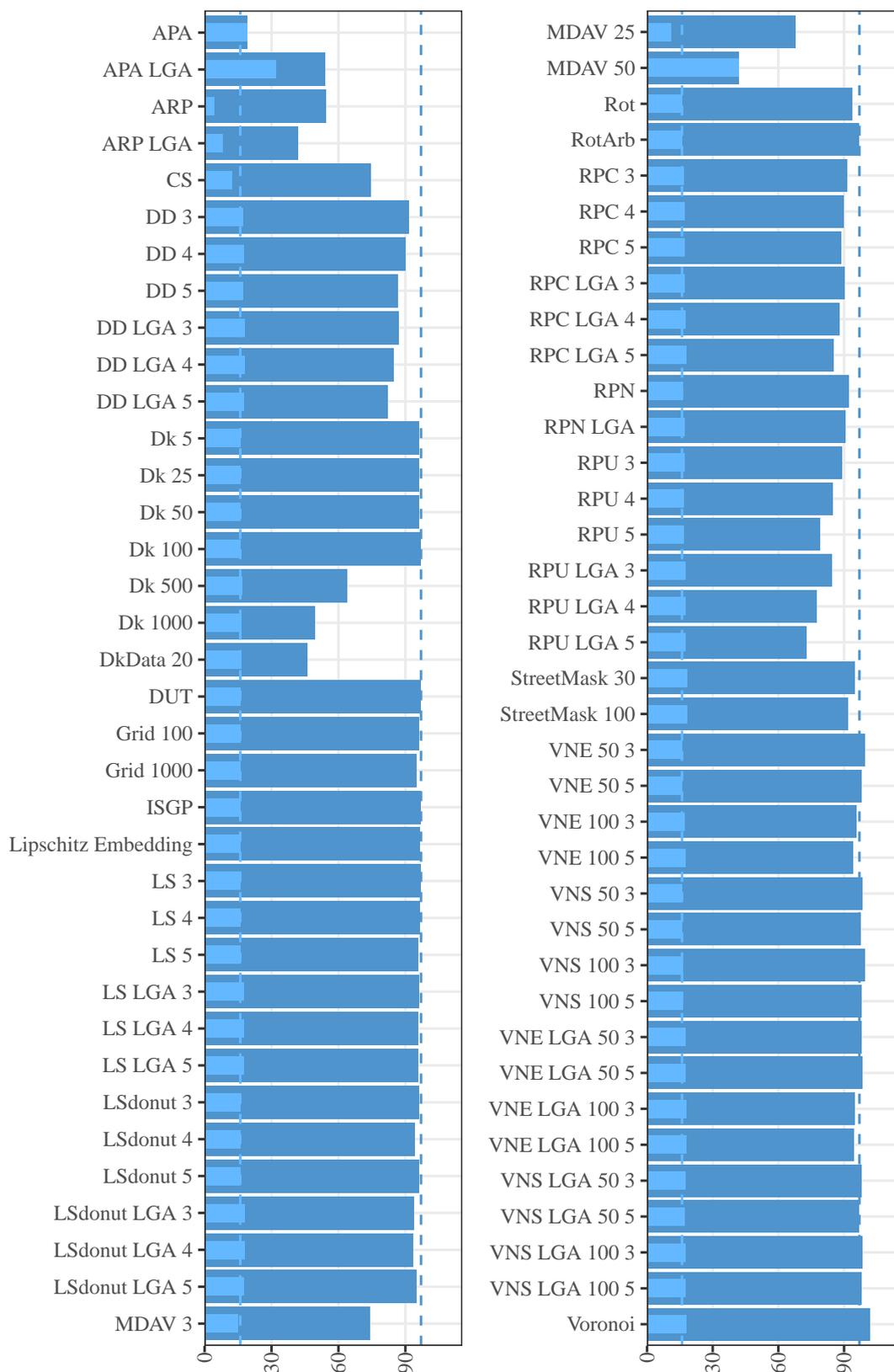


Figure 5.28.: DBSCAN ( $\epsilon = 9, 500$ ): Total number of clusters as well as number of clusters with at least 30 points (light blue bars). Dashed lines shows the number of clusters in the original data (total: 97; larger 30: 16).

Figures 5.27 and 5.28 also show that the number of clusters with at least 30 points is better preserved than the total number of clusters. The 1,000 meter official statistics grid results in a lower total number of clusters.

For APA, all points within a region are moved to the center of the region. Thus, clusters of points are formed. In addition, all clusters contain at least 30 points (45 clusters for  $\varepsilon = 3, 200$  and 19 clusters for  $\varepsilon = 9, 500$ ) for state electorates as polygons. For local government areas, the number of clusters with at least 30 points is larger than for the original data (44 clusters for  $\varepsilon = 3, 200$  and 32 clusters for  $\varepsilon = 9, 500$ ), but the total number of clusters is smaller (67 clusters for  $\varepsilon = 3, 200$  and 54 clusters for  $\varepsilon = 9, 500$ ). Since the points are always moved to the same location, and there is no variation between replications, there is no variation in the number of clusters. As for the noise points, points that were not clustered before are now clustered.

In ARP, the points are randomly moved within the boundaries of their region. Consequently, the points are more dispersed than in the original data. Thus, the total number of clusters (for  $\varepsilon = 3, 200$  on average 64.02 for ARP and 74 for ARP LGA, for  $\varepsilon = 9, 500$  on average 54.42 for ARP and 41.80 for ARP LGA) and the number of clusters with at least 30 points (for  $\varepsilon = 3, 200$  on average 2.64 for ARP and 7.5 for ARP LGA, for  $\varepsilon = 9, 500$  on average 4.32 for ARP and 8.20 for ARP LGA) are lower. The number of non-clustered points increases drastically (for  $\varepsilon = 3, 200$  on average 1,991.26 for ARP and 1,285.60 for ARP LGA, for  $\varepsilon = 9, 500$  on average 858.82 for ARP and 365.74 for ARP LGA). Very few points that were noise points in the original data are now clustered points. The majority of non-clustered points are points that have been displaced so far that they are no longer considered clustered. There is also a large variation in the results between replications.

MDAV also shows large differences in the clustering compared to the clustering of the original data. For the smallest cluster size considered (3), the total number of clusters is smaller than the original number of clusters (89 for  $\varepsilon = 3, 200$  and 68 for  $\varepsilon = 9, 500$ ), and the number of non-clustered points is about twice as large (531 for  $\varepsilon = 3, 200$  and 255 for  $\varepsilon = 9, 500$ ). Many of these points are points that were clustered in the original data set but are not when masked. This is explained by the fact that the cluster size for MDAV 3 is smaller than the minimum cluster size for DBSCAN. MDAV 25 and MDAV 50 show no non-clustered points, i.e., all points that were noise points in the original data set are included in a cluster. Correspondingly, for MDAV 25, the clusters all contain at least 20 points. Only the number of clusters with at least 30 elements is considerably lower (68 compared to 11 for  $\varepsilon = 9, 500$  and 94 compared to 13 for  $\varepsilon = 3, 200$ ). MDAV 50 results in all clusters containing at least 30 elements, with the number of clusters being about half of the original total cluster size.

The masking method change of scale also shows large deviations from the result of the clustering algorithm for the original data set with a large variation between replications. The points are moved closer together when the coordinates are multiplied by a number between zero and one. Since the radius for the DBSCAN algorithm has

not been changed for every masking method, the radius thus covers many more points than in the original data set. When numbers greater than one are used, the points are dispersed further apart so that existing clusters are dissolved. Although a proper adjustment of the radius would still yield approximately the number of clusters in the original data set, it is questionable whether the larger radius size required would allow points to be called “adjacent”.

The  $k$ -nearest neighbor donut masking method shows the influence of displacement distances on the correct specification of the number of clusters. With an increasing number of  $k$  and thus increasing the distance by which the points can potentially be shifted, the number of clusters varies more. The total number of clusters is, on average, smaller than the original number of clusters. Once a  $k = 500$  is reached, the total number of clusters starts to deviate more from the original total (78.52 for  $\varepsilon = 3,200$  and 63.92 for  $\varepsilon = 9,500$  compared to  $k = 5$ : 129.54 for  $\varepsilon = 3,200$  and 96.22 for  $\varepsilon = 9,500$ ), while the number of clusters with at least 30 points remains close (between 16 and 17 for both radii for all values of  $k$ ).

The number of non-clustered points increases greatly for large values of  $k$  (500, 1,000). Again, many of these points are points that are clustered in the original data set. Even worse results compared to the original clustering are obtained by the variant where the displacement distance is calculated from the data set itself (DkData). The number of non-clustered points is about three times higher (3.4 times higher for  $\varepsilon = 3,200$  and 2.5 times higher for  $\varepsilon = 9,500$ ), and the total number of clusters is much lower (46.16 for  $\varepsilon = 9,500$  and 43.88 for  $\varepsilon = 3,200$ ).

With donut masking using population density, even the smallest radius considered (DD 3) results in a much smaller total number of clusters, e.g., for postcode population density 109.08 ( $\varepsilon = 3,200$ ) and 91.64 ( $\varepsilon = 9,500$ ). However, the number of clusters with at least 30 points remains about the same. The variation between replications is also larger compared to other methods. In addition, more points remain non-clustered. The difference between the population density based on local government areas and based on postcode areas is minor.

RPC shows similar results to DD in terms of variation across replications. The total number of clusters is lower but closer to the results of the original data set compared to DD. RPC preserves the number of clusters with at least 30 points. However, there are many more non-clustered points than in the unmasked data (over 300 non-clustered points for  $\varepsilon = 3,200$  and over 140 for  $\varepsilon = 9,500$ ). RPU performs worse than RPC, with the largest variation in the total number of clusters between replications similar to Dk 500 and Dk 1,000. Of the three random perturbation methods, RPN preserves the number of clusters the most. However, in all three variants (RPC, RPN, RPU), the much larger number of non-clustered points can be observed. The number of points that are clustered in the original data set and are now noise points is larger for  $\varepsilon = 3,200$  than for  $\varepsilon = 9,500$ .

Location swapping is a masking method that shows, for the given data, that a larger

radius does not necessarily lead to a larger variation in detected clusters. The total number of clusters is almost identical to the original number of clusters. However, for clusters with at least 30 points, local government area population density increases the number of clusters by one on average and up to two for the donut variant. A closer look at the points shows that about the same number of points that are clustered in the original data set are now noise points, and about the same number of noise points are now clustered.

For the verified neighbor approach, the number of clusters increases for most parameter choices, especially when only clusters with at least 30 elements are considered (up to 18 clusters for  $\varepsilon = 9,500$  and up to 20 clusters for  $\varepsilon = 3,200$ ). Multiplying the estimate of the average distance between people by 3 or 5 does not give different results; this is also true for the area considered for population density (LGA vs. postcode). For the total number of clusters, increasing the minimum number of people with the same characteristics (50 vs. 100) resulted, on average, in a lower number of clusters for employment status and a higher number of clusters for sex. Again, there is greater variation across the 50 replications. For all parameter choices, there are more clusters with 30 points than in the original data.

When comparing the points that are now noise points in the masked data set and points that are now clustered in the masked data set, more points are included in clusters than are removed from clusters. Also, when looking at the average of the 50 replications, fewer points change from clustered to non-clustered and vice versa for  $k = 50$  than for  $k = 100$ . This is also true for multiplying by 3 compared to 5 and using sex as a variable compared to employment status. However, no difference can be seen when comparing using LGA to postcode areas to calculate population density.

For street masking, the total number of clusters is smaller than the original number of clusters (127.96 for  $\varepsilon = 3,200$  and 94.78 for  $\varepsilon = 9,500$ , for a depth value of 30), especially for a depth value of 100 (120.26 for  $\varepsilon = 3,200$  and 91.98 for  $\varepsilon = 9,500$ ). However, two additional clusters reach a minimum cluster size of 30 points compared to the original data set.

Voronoi masking is a masking method that gives the same results whenever applied. The number of clusters increases (a total of 135 for  $\varepsilon = 3,200$  and 102 for  $\varepsilon = 9,500$ ). This can be explained by the fact that points are moved to the border of their polygon and thus closer to other points. In turn, fewer non-clustered points are found (235 for  $\varepsilon = 3,200$  and 116 for  $\varepsilon = 9,500$ ). Many of the points that change from clustered to non-clustered and vice versa are points that are noise points in the data set and are now included in a cluster.

Although DBSCAN allows the calculation based on a distance matrix, random projection yields bit vectors for which only a similarity matrix can be calculated. As a consequence, the clustering of points cannot be evaluated for random projection with  $\varepsilon$  in meters.

### 5.2.5. Aggregated Results by Geomasking Methods

In addition to the individual results (for every parameter choice), the aggregated results of utility measures by masking method should also be considered.

#### 5.2.5.1. Descriptive Statistics

The random perturbation methods and the donut methods remain quite close to the original descriptive statistics. Furthermore, there are only minor differences between different parameter choices as seen by the standard deviation of table 5.2.

Table 5.2.: Aggregation of descriptive statistics by methods. Sd shows variation between different parameter choices for the respective masking method.

method	mean center		median center		standard distance	
	avg. dist.	sd	avg. dist.	sd	average	sd
orig	0.00	0.00	0.00	0.00	124,026.65	0.00
APA	11,689.13	15,707.24	912.07	152.94	157,387.13	47,071.28
ARP	12,114.16	16,075.75	534.68	508.45	167,136.32	57,065.82
CS	1,169,524.71		1,168,196.01		119,287.20	
DD	26.52	6.40	22.76	2.86	124,060.41	11.56
Dk	32.02	40.30	20.95	11.15	124,119.20	151.75
DkData	96.19		35.63		124,513.47	
DUT	7,701.91		7,701.50		124,026.65	
Grid	3.12	4.25	276.40	319.09	124,026.35	1.03
ISGP	0.00		4,265.42		23,772.16	
Lipschitz	0.00		10,019.75		107,974.08	
LS	18.68	9.24	14.49	3.31	123,987.26	38.26
LSdonut	42.33	30.11	18.08	4.97	123,880.19	180.63
MDAV	0.00	0.00	72.39	54.39	122,709.93	1,160.08
Rot	2,545,551.79		2,542,308.90		124,026.65	
RotArb	0.00		10,384.65		124,026.65	
RPC	19.80	4.53	17.81	4.31	124,051.60	9.73
RPN	11.51	1.40	13.00	1.99	124,033.83	3.84
RPU	32.89	6.80	29.79	3.72	124,091.64	24.46
StreetMask	244.00	6.34	32.53	0.16	123,339.50	4.62
VNE	68.50	16.28	23.51	3.45	123,468.03	104.46
VNS	41.83	8.66	20.04	3.01	123,804.68	81.76
Voronoi	26.27		10.12		123,826.14	

Even location swapping methods, verified neighbor methods, and Voronoi masking, whose displacement is based on the position of other points, show good results. Street masking shows an average distance of 244 meters from the original location of the spatial mean center. The spatial median center remains close to the original. In contrast, the official statistics grid shows minor displacements of the spatial mean center but an average of 276 meters difference for the spatial median center with a large standard deviation of different parameter choices (100 meters grid vs. 1,000 meters grid).

The affine transformations preserve the standard distance (except for change of scale). But these methods do not preserve the location of the spatial median center and the spatial mean center, except for rotation around the center of the coordinate system, which preserves the spatial mean center. Voronoi masking reduces the standard distance by about 200 meters as the points are moved closer together. This is even more noticeable for the verified neighbor approach and street masking.

APA and ARP do not preserve descriptive statistics and also show large differences between parameter choices. Distance approximation using ISGP and anonymization of distance matrices via Lipschitz embedding rely on approximated points and lack reliable descriptive statistics. MDAV preserves the spatial mean center but shows a small difference in the position of the median center. The standard distance is much smaller than the original standard distance and shows a large variation for the different parameter choices considered.

Table 5.3 shows the aggregated results by masking method for the standard deviational ellipse. All parts of the ellipse are preserved for displacement using translation, and the official statistics grid shows only minor deviations. For most masking methods, the orientation is preserved. Only the rotation methods and distance approximation using ISGP fail to preserve the orientation. For the distance approximation using ISGP, the evaluations are based on the approximated points. However, the rotation methods keep the major and minor axis's length, while most masking methods have larger deviations.

Overall, displacement using translation and the official statistics grid remain the closest to the original data set, followed by random perturbation, donut masking, and location swapping. This is followed by location swapping using donut, Voronoi masking, and the verified neighbor approach. In the latter, the influence of the number of categories for the variable of interest can be seen. MDAV shortens the distance of the axes. However, large differences in the results between chosen cluster sizes can be seen. The verified neighbor approach and street making show negligible difference in the angle of rotation. But the minor and major axis are shortened. For street masking, the minor axis is shortened even more than the major axis.

Table 5.3.: Aggregation of standard deviational ellipses by masking methods. For major and minor axis the difference to the original major and minor axis is shown. Sd shows variation between different parameter choices for the respective masking method.

method	angle		major		minor	
	avg.	sd	avg.	sd	avg.	sd
orig	121.03	0.00	0.00	0.00	0.00	0.00
APA	121.72	1.12	30,430.31	42,205.92	13,659.05	21,043.39
ARP	122.46	2.44	37,933.67	50,823.06	20,817.90	26,120.02
CS	121.03		-4,371.01		-1,832.13	
DD	121.03	0.01	21.44	10.12	36.10	14.86
Dk	121.04	0.01	42.16	63.02	137.94	245.08
DkData	121.01		296.06		549.44	
DUT	121.03		0.00		0.00	
Grid	121.03	0.00	0.68	0.27	-2.41	3.29
ISGP	148.63		-95,119.28		-34,018.24	
Lipschitz	118.02		-15,699.02		-4,440.21	
LS	121.07	0.03	-16.52	32.05	-62.54	51.81
LSdonut	121.03	0.09	-105.27	184.17	-127.83	60.30
MDAV	121.15	0.26	-956.94	841.31	-1,130.74	1,019.76
Rot	81.19		0.00		0.00	
RotArb	83.55		0.00		0.00	
RPC	121.03	0.00	14.74	7.02	29.30	18.59
RPN	121.03	0.00	3.51	2.58	10.16	3.77
RPU	121.04	0.01	38.69	15.29	75.60	28.66
StreetMask	120.71	0.01	-487.09	3.57	-617.02	20.63
VNE	120.79	0.03	-475.11	86.89	-311.83	66.36
VNS	120.98	0.05	-185.08	74.35	-133.65	57.60
Voronoi	121.00		-150.68		-159.31	

### 5.2.5.2. Distances

The mean and median distance between points by masking methods is shown in table 5.4. Comparing the mean and median distance between points by masking methods shows an important aspect of analyzing the utility of geomasking methods. Relying solely on the average or median distance between points incorrectly leads to many masking methods being considered utility-preserving even though points have been moved far (see table 5.5, e.g., DUT).

The donut methods, location swapping methods, verified neighbor methods, displacement using translation, rotation about the spatial mean center, random perturbation methods, and Voronoi masking all show minor differences to the mean and median distance between points (table 5.4). The official statistics grid shows a difference from

the original mean distance of about 300 meters and about 110 meters for the median distance. The difference is less than 500 meters and almost identical to the median distance for MDAV and street masking, if averaged over the different parameter choices.

Table 5.4.: Aggregation of mean and median distance between points by masking methods. Sd shows variation between different parameter choices for the respective masking method.

method	mean distance		median distance	
	avg.	sd	avg.	sd
orig	104,527.22	0.00	36,225.98	0.00
APA	128,489.42	33,413.98	37,531.25	153.40
ARP	131,978.28	35,519.55	39,281.61	442.98
CS	99,463.25		34,521.49	
DD	104,538.78	4.57	36,225.52	0.52
Dk	104,551.81	40.77	36,224.83	2.56
DkData	104,636.44		36,213.93	
DUT	104,526.92		36,225.86	
Grid	104,225.87	2.24	36,112.67	17.13
ISGP	44,326.35		36,100.10	
Lipschitz Embedding	93,126.27		36,103.43	
LS	104,503.23	20.88	36,223.33	1.19
LSdonut	104,452.94	80.59	36,219.06	3.00
MDAV	104,083.41	441.57	36,223.71	96.03
Rot	97,182.00		33,674.45	
RotArb	104,456.14		36,181.56	
RPC	104,535.44	4.57	36,226.00	0.84
RPN	104,529.02	2.77	36,225.76	0.38
RPU	104,549.98	12.16	36,224.86	2.12
StreetMask	104,127.62	9.35	36,217.67	5.60
VNE	104,330.20	48.22	36,181.98	10.68
VNS	104,445.76	35.93	36,210.07	4.01
Voronoi	104,436.20		36,216.86	

Change of scale, distance approximation using ISGP, anonymization of distance matrices via Lipschitz embedding, and APA and ARP show large deviations. In the case of distance approximation using ISGP, this can be explained by the fact that this masking method was designed to mask distances to points of interest rather than a full distance matrix of the individual points. Therefore, many of the larger distances are set to the maximum considered value. Moreover, large differences between parameter choices can be seen for APA and ARP, thus, the parameter choice strongly influences the results.

Table 5.5 shows the average distance by which the points are moved. Donut masking based on population density, donut masking based on the distance to the  $k$ -nearest neighbors, random perturbation methods, the verified neighbor approach, street masking, the official statistics grid, location swapping methods, and Voronoi masking do not move the points very far, on average. As previously shown, some points, especially points in areas with few neighboring points, are moved much further than in dense areas. Again, street masking and the verified neighbor approach show similar results. However, there is a larger variation between parameter choices for  $k$ -nearest neighbor donut masking.

Table 5.5.: Aggregation of average and median distance between original and masked points by masking methods. Sd shows variation between different parameter choices for the respective masking method.

method	mean	sd	median	sd
orig	0.00	0.00	0.00	0.00
APA	24,083.54	22,053.22	3,653.51	899.29
ARP	29,490.13	23,938.37	5,102.91	1,438.45
CS	1,170,831.22		1,167,795.25	
DD	626.80	92.93	96.27	19.76
Dk	543.30	713.72	170.45	160.63
DkData	2,156.50		489.27	
DUT	7,700.96		7,701.50	
Grid	210.46	243.54	218.90	253.15
ISGP	55,383.18		10,366.29	
Lipschitz Embedding	23,726.11		11,771.12	
LS	276.37	61.40	87.19	20.96
LSdonut	417.92	92.86	93.41	22.73
MDAV	4,092.38	3,068.87	614.64	434.33
Rot	2,548,618.34		2,541,755.21	
RotArb	83,589.94		23,263.83	
RPC	418.17	93.58	71.51	17.05
RPN	261.07	0.61	42.79	5.48
RPU	895.56	138.69	135.09	27.43
StreetMask	627.12	146.11	233.41	89.00
VNE	767.93	190.29	226.63	40.56
VNS	463.41	103.79	167.88	29.70
Voronoi	177.87		65.21	

APA and ARP move points far from their original location. This is strongly influenced by the definition of each region considered. The cluster size strongly influences how far points are moved in MDAV. Most rotation angles and most multipliers for

change of scale also lead to large displacements. In the distance approximation using ISGP and the anonymization of distance matrices via Lipschitz embedding, the distance was calculated using approximated points and therefore are not reliable.

### 5.2.5.3. Spatial Autocorrelation

The global evaluation of spatial autocorrelation using Moran's I (see table 5.6) shows that, on average, most masking methods preserve the original value or show only minor differences, even for different parameter choices. The overall conclusion of a slight tendency to a clustered data set, remains the same.

Only APA, ARP, MDAV, and Voronoi could potentially lead to different conclusions regarding overall clustering/dispersion. While APA, ARP, and MDAV show a much lower Moran's I than the original data set, Voronoi masking almost doubles the value.

Table 5.6.: Aggregation of Moran's I for singles and employment status by masking methods. Sd shows variation between different parameter choices for the respective masking method.

method	singles		employment	
	avg.	sd	avg.	sd
orig	0.13	0.00	0.17	0.00
APA	0.01	0.00	0.01	0.00
ARP	0.04	0.00	0.04	0.00
CS	0.13		0.17	
DD	0.10	0.00	0.12	0.00
Dk	0.12	0.01	0.15	0.02
DkData	0.09		0.11	
DUT	0.13		0.17	
Grid	0.09	0.04	0.12	0.05
ISGP	0.11		0.13	
Lipschitz Embedding	0.13		0.16	
LS	0.12	0.00	0.16	0.00
LSdonut	0.12	0.00	0.16	0.00
MDAV	0.06	0.03	0.07	0.04
Rot	0.13		0.17	
RotArb	0.13		0.17	
RPC	0.10	0.00	0.13	0.00
RPN	0.11	0.00	0.14	0.00
RPU	0.10	0.00	0.12	0.01
StreetMask	0.12	0.01	0.16	0.01
VNE	0.11	0.00	0.15	0.00
VNS	0.12	0.00	0.16	0.00
Voronoi	0.25		0.30	

#### 5.2.5.4. Clustering

The aggregated results for the number of clusters identified (see table 5.7) by masking method show that most masking methods yield approximately the same number of clusters when larger clusters (at least 30 points in a cluster) are considered.

Displacement using translation, rotation around an arbitrary point, distance approximation using ISGP preserve the number of clusters. Rotation and anonymization of distance matrices via Lipschitz embedding remain very close to the original number of clusters. In the case of rotation around the origin, the small difference can be explained by the fact that coordinates had to be transformed before rotating around the pivot point, and a small additional distortion was introduced.

Table 5.7.: Aggregation of average number of cluster larger than 30 as well as number of non-clustered points by masking methods. Sd shows variation between different parameter choices for the respective masking method.

method	$\varepsilon = 3, 200$				$\varepsilon = 9, 500$			
	total		larger 30		total		larger 30	
	avg.	sd	avg.	sd	avg.	sd	avg.	sd
orig	130.00	0.00	17.00	0.00	97.00	0.00	16.00	0.00
APA	56.00	15.56	44.50	0.71	36.50	24.75	25.50	9.19
ARP	69.01	7.06	5.07	3.44	48.11	8.92	6.26	2.74
CS	111.16		16.02		74.40		12.48	
DD	105.02	4.42	16.79	0.07	87.00	3.48	17.65	0.32
Dk	107.79	33.66	16.69	0.49	83.12	21.03	16.30	0.34
DkData	43.88		15.56		46.16		16.64	
DUT	130.00		17.00		97.00		16.00	
Grid	128.00	2.83	17.00	0.00	95.50	0.71	16.00	0.00
ISGP	130.00		17.00		96.36		16.00	
Lipschitz	130.00		17.00		96.68		16.00	
LS	128.37	0.55	17.54	0.73	96.13	0.44	16.86	0.74
LSdonut	127.92	1.35	17.72	0.93	94.80	1.23	17.04	0.95
MDAV	82.67	15.50	31.67	28.94	61.33	17.01	22.67	16.86
Rot	127.08		16.46		93.92		16.22	
RotArb	130.00		17.00		96.90		16.00	
RPC	112.00	5.09	16.83	0.11	88.89	2.21	17.33	0.32
RPN	120.55	0.47	16.87	0.01	91.43	0.89	16.86	0.48
RPU	102.66	5.76	16.53	0.21	81.28	5.88	17.35	0.41
StreetMask	124.11	5.44	18.24	0.28	93.38	1.98	18.37	0.01
VNE	128.84	2.83	20.19	1.08	96.58	2.09	17.25	0.71
VNS	133.06	1.52	19.07	0.93	98.08	0.72	16.93	0.66
Voronoi	135.00		19.00		102.00		18.00	

Donut masking using  $k$ -nearest neighbors using the data set as reference file (DkData 20) does not preserve the total number of clusters but shows similar results when only clusters with at least 30 points are considered.

Donut masking using population density and random perturbation show similar results. The number of clusters with at least 30 elements remains close to the original,

but the total number of clusters is smaller. Aggregating the results of the different parameter choices shows even more clearly how RPU preserves the number of clusters less than RPC and RPN. RPN preserves the number of clusters the most.

The location swapping methods and the official statistics grid remain close to the original when averaging the results over the different parameter choices. However, there is a small variation among the parameter choices. The verified neighbor methods yield a similar total number of clusters but with a difference of up to three clusters.

Street masking shows a smaller total number of clusters but a larger number of clusters with at least 30 points. Finally, Voronoi masking is the only masking method that yields more clusters than the original data set. Change of scale shows very different results depending on the number by which the coordinates are multiplied.

The majority of methods that show differences in the number of clusters leave more points non-clustered. In particular, ARP,  $k$ -nearest neighbor donut masking with large values of  $k$ , and MDAV 3 show much more non-clustered points. APA, MDAV with cluster sizes 25 and 50 leave (almost) no points non-clustered. Street masking and verified neighbor methods leave fewer points non-clustered.

Table 5.8.: Aggregation of total of number of points changing from clustered to non-clustered and from non-clustered to clustered by masking methods. Sd shows variation between different parameter choices for the respective masking method.

method	$\varepsilon = 3, 200$		$\varepsilon = 9, 500$	
	mean	sd	mean	sd
APA	246.00	2.83	132.00	2.83
ARP	1,491.75	487.21	595.30	350.78
CS	60.78		65.46	
DD	189.18	44.62	57.82	14.77
Dk	137.88	226.20	65.06	94.79
DkData	641.20		276.50	
DUT	0.00		0.00	
Grid	1.00	1.41	0.00	0.00
ISGP	0.00		0.00	
Lipschitz	0.04		0.08	
LS	15.64	4.91	11.76	3.86
LSdonut	21.58	8.21	11.74	3.95
MDAV	277.00	50.23	143.00	15.59
Rot	1.52		7.16	
RotArb	0.00		0.00	
RPC	121.48	43.17	42.38	13.41
RPN	60.45	20.21	24.90	2.83
RPU	309.98	48.66	80.40	17.86
StreetMask	39.99	31.38	16.85	11.70
VNE	119.98	37.78	68.94	20.49
VNS	57.69	23.54	41.53	14.93
Voronoi	13.00		34.00	

Finally, table 5.8 shows the aggregated results of the number of points that were either assigned to a cluster in the original data set and are now non-clustered (noise points), or were non-clustered in the original data set, and are assigned to a cluster in the masked data set. As can be seen, for APA, ARP, and  $k$ -nearest neighbor donut masking using the data set as reference file (DkData), the number of points that change from being clustered to noise points and vice versa is much larger than for the other methods. As previously stated, APA reduces the number of noise points while ARP (and DkData) enlarges it.

For  $k$ -nearest neighbor donut masking using the residential file as the reference file, only large values of  $k$  (especially for the smaller radius) lead to differences compared to the original clustering. More points change from being clustered to noise points. This can also be seen for the random perturbation methods. For MDAV, the number of points shown in table 5.8 are noise points in the original data set and are now assigned to a cluster if the cluster size exceeds the minimum number of points (MinPts).

The location swapping methods, Voronoi masking, and street masking show minor differences to the original clustering in terms of preserving the clustering and non-clustering of points. Only DUT, anonymization of distance matrices via Lipschitz embedding, distance approximation using ISGP, and rotation methods show the desired small difference to the original clustering.

### 5.3. Risk Evaluation

The risk of the masking methods is evaluated by applying the strategies described in chapter 4.2 to each masking method, and calculating the precision and recall. The average run times of each re-identification method can be found in appendix F.

For the affine transformation methods, an individual attack was used, reversing the masking process. The execution takes on average 3.42 minutes for DUT for the full sample and less than one minute for the subsample of  $n = 1,000$  and  $n = 2,000$ . The displacement caused by change of scale can be reversed in 6.88 minutes for the full sample and 1.62 respectively 0.93 minutes for the subsamples  $n = 2,000$  and  $n = 1,000$ . The rotation methods need less than 0.27 minutes for the full sample regardless of the pivot point chosen.

Using the mean of multiple releases takes 0.04 minutes on average. Slightly faster is the minimum distance approach with the smallest sample of  $n = 1,000$ . However, if the sample size is increased to  $n = 2,000$ , the execution time increases to 0.03-0.06 minutes. Similarly, without additional variables, the Hungarian algorithm shows a more considerable difference between a sample size of  $n = 1,000$  and  $n = 2,000$ . The smaller sample takes on average between 0.16 and 0.45 minutes, the larger subsample needs between 1.9 and 8.5 minutes. Using additional variables reduces the execution times to 0.02 minutes for both subsamples. The full sample takes up to 0.25 minutes to execute. The graph theoretic linkage attack takes between 4 and 8 minutes. Although

this seems quite fast, applying it to 50 replications of 74 different parameter choices for 12 different masking methods (excluding affine transformation methods) results in a large total runtime. The graph matching attack on privacy-preserving record linkage takes about 2 minutes for the small subsample and about 15 minutes for the  $n = 2,000$  sample. However, the original code provided by Vidanage et al. (2020) was not used due to the changes necessary. Therefore, the attack method might require more time than with the original code written in Python.

### 5.3.1. Reversing Masking Methods

The reversal of masking methods was only applied to the affine transformation methods. A precision of one and a recall of one is achieved for all replications (see table 5.9). When rotating around an arbitrary point, care must be taken that the arbitrary point (center of the masked coordinates) is calculated with the full sample, even when using a smaller sample.

In displacement using translation, a random integer is added or subtracted to each of the coordinates. The random integer is drawn from the interval -10,000 to 10,000 meters and was added to the coordinates in a planar system (easting and northing). Thus, the numbers between -10,000 and 10,000 are tried until an overlap between the data sets is reached.

Change of scale is another masking method that can easily be reversed, similar to displacement using translation. The multiplier between 0 and 2 must be found. In this case, the multiplier is limited to the fifth decimal place. But even if more decimal places are allowed, this only increases the time needed, but not the result. The coordinates of the two data sets are rounded to the fourth decimal place because the coordinates are converted, possibly introducing additional noise.

Table 5.9.: Precision and recall for affine transformation masking methods.

masking method	$n = 1,000$		$n = 2,000$		$n = 10,000$	
	precision	recall	precision	recall	precision	recall
CS	1	1	1	1	1	1
DUT	1	1	1	1	1	1
Rot	1	1	1	1	1	1
RotArb	1	1	1	1	1	1

### 5.3.2. Mean of Masked Coordinates

For the method “mean of the masked coordinates”, the arithmetic mean of the first five replications is calculated. This is repeated in steps of five up to 50 replications. The resulting coordinates are compared with the original locations. Since a distance of zero between the mean of the masked coordinates and the original coordinate may

not be reached, a distance of less than one meter is defined to be sufficient. Taking the mean of masked coordinates does not involve an identification file and does not quite fit the described scenario. However, since it is discussed in the geomasking literature as one of the few approaches to re-identify coordinates (Zimmerman and Pavlik, 2008), it will be applied. Also, the results cannot be evaluated in terms of precision and recall because no distinction can be made between true positives, false positives, true negatives, and false negatives. Therefore, only the “correctly identified” points (in percent) is reported (see figures 5.29, 5.30, and 5.31).

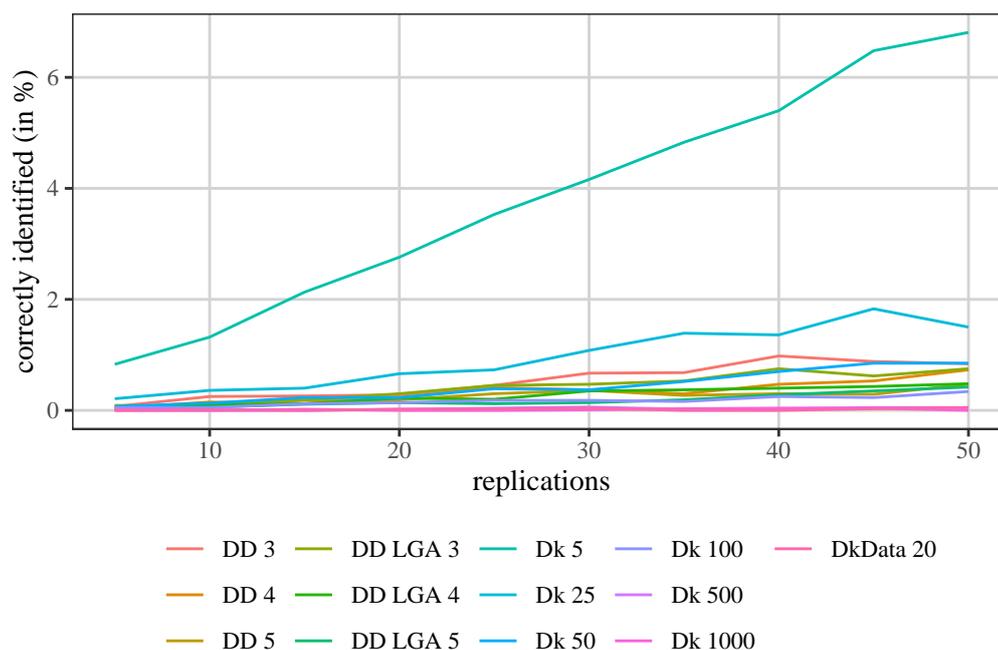


Figure 5.29.: Mean of masked coordinates: donut masking method using population density and  $k$ -nearest neighbor donut masking.

Figures 5.29 and 5.30 show that donut masking based on population density and random perturbation methods yield less than 3% correctly identified records. However, it can be seen that the percentage of correctly identified records decreases when increasing the multiplier for the estimate of the average distance between people. Donut masking using 5-nearest neighbor (see figure 5.29) shows only about 6% correctly identified records. With larger values for  $k$ , the percentage of correctly identified points decreases. Close to zero records could be identified when masked with the verified neighbor approach and location swapping (see figure 5.31). For the official statistics grid, MDAV, street masking, Voronoi masking, APA, and ARP, the masked coordinates’ mean results in zero correctly identified coordinates (not shown in a figure).

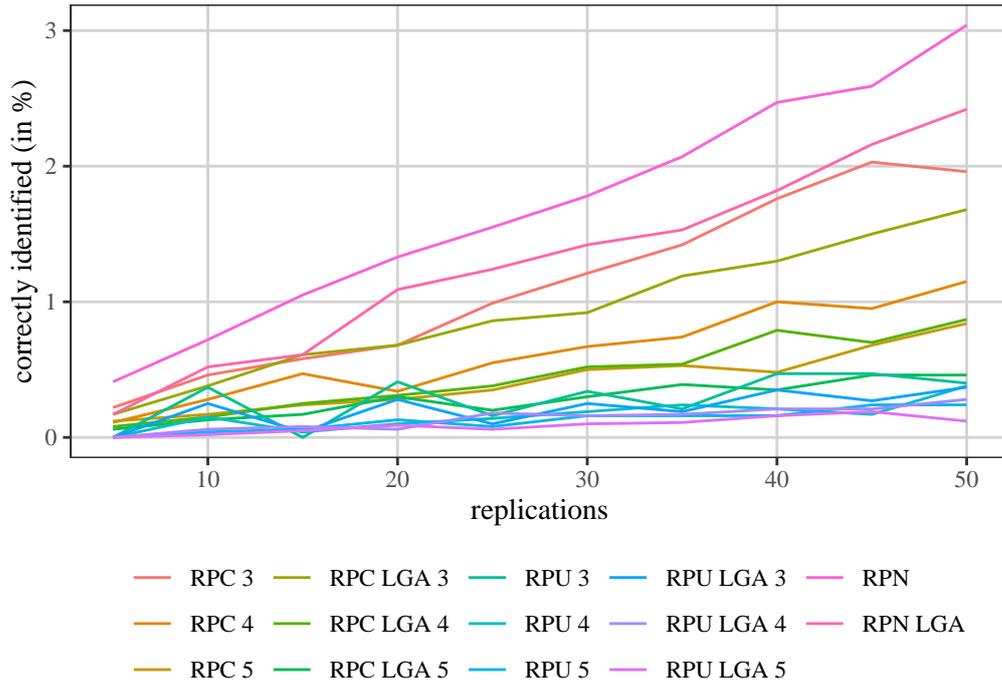


Figure 5.30.: Mean of masked coordinates: random perturbation methods.

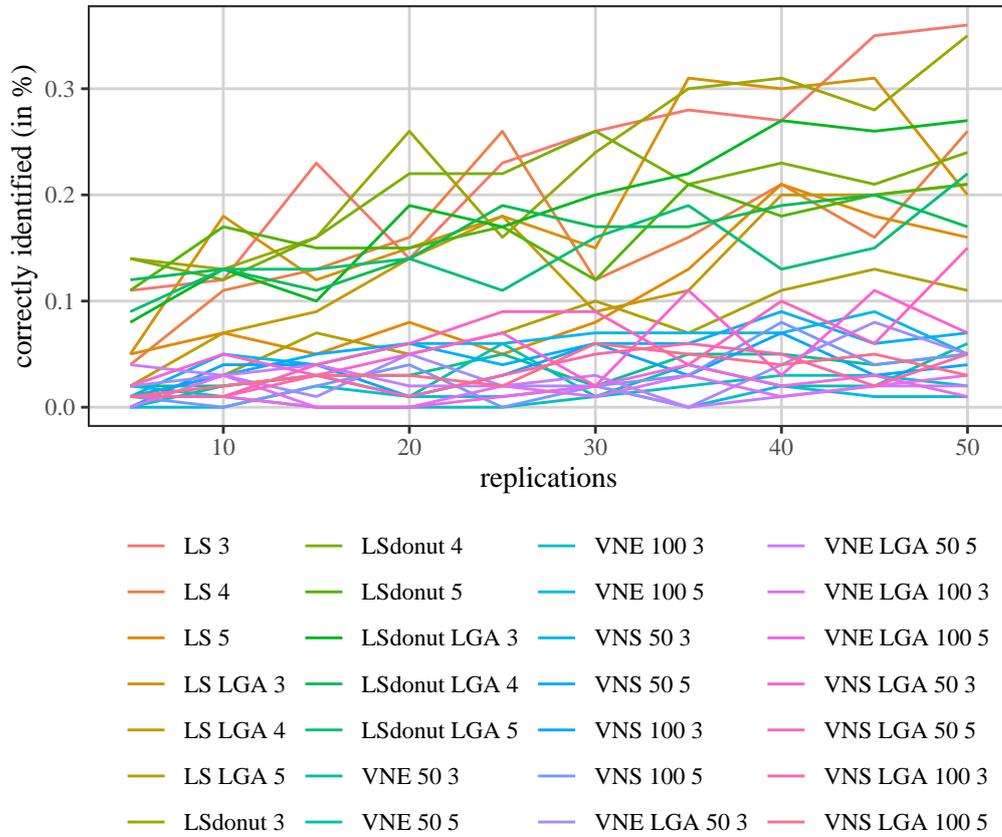


Figure 5.31.: Mean of masked coordinates: location swapping, and the verified neighbor approach.

Overall, using the mean of multiple replications is not a suitable method to re-identify the original location of coordinates. It will be shown throughout this chapter that there are more suitable attack methods.<sup>12</sup>

### 5.3.3. Minimum Distance

For the minimum distance, a  $n = 1,000$  subsample and a  $n = 2,000$  subsample were considered.<sup>13</sup> For each replication of the masking method, the closest point in the identification file was considered a potential match. If the location of an identification file was assigned twice, these matches were deleted as it is assumed that there are no duplicate entries. Precision and recall were calculated and then averaged over the replications.

Figure 5.32 and table G.15 in appendix G.9 show that using the minimum distance between records alone yields a recall of up to 40% for a subsample of  $n = 1,000$  and up to 48% for a subsample of  $n = 2,000$ . However, this is accompanied by a larger proportion of false positives, as this attack method is designed to match each point in the masked data set with a point in the identification data set. Therefore, a much lower precision (approximately 0.11) is achieved.

The recall for  $n = 1,000$  for most masking methods shows little difference between parameter choices. The exceptions are the official statistics grid, MDAV, and street masking. For  $k$ -nearest neighbor donut masking, small values of  $k$  (up to 100) give the same results for precision and recall (recall of 0.39 and precision of 0.12). Larger values of  $k$  decrease precision and recall, e.g., for  $k = 1,000$  recall of 0.25 and precision of 0.08.

For  $n = 2,000$ , the difference between parameter choices that is, the decrease in precision and recall as the possible displacement distances increase is more apparent. However, most parameter choices of masking methods differ regarding precision and recall by less than 0.1, e.g., for multiplying by 4 compared to 3. Again, the official statistics grid, MDAV, and street masking show larger differences between parameter choices.

The only masking methods for which low recall is seen are APA, ARP, and MDAV (for cluster sizes 25 and 50). For distance approximation using ISGP, random projection, and anonymization of distance matrices via Lipschitz embedding, no points can be re-identified since the attack method does not take distance matrices as input.<sup>14</sup>

---

<sup>12</sup>No difference could be found when using the coordinates in latitude-longitude format or easting-northing format.

<sup>13</sup>Due to the ineffectiveness of this attack method and the much longer run times for the full sample only the subsamples were considered.

<sup>14</sup>Using the approximation, as for some utility measures, is not realistic since the original coordinates must be known for the Procrustes analysis and it is assumed that a potential intruder does not know the original coordinates.

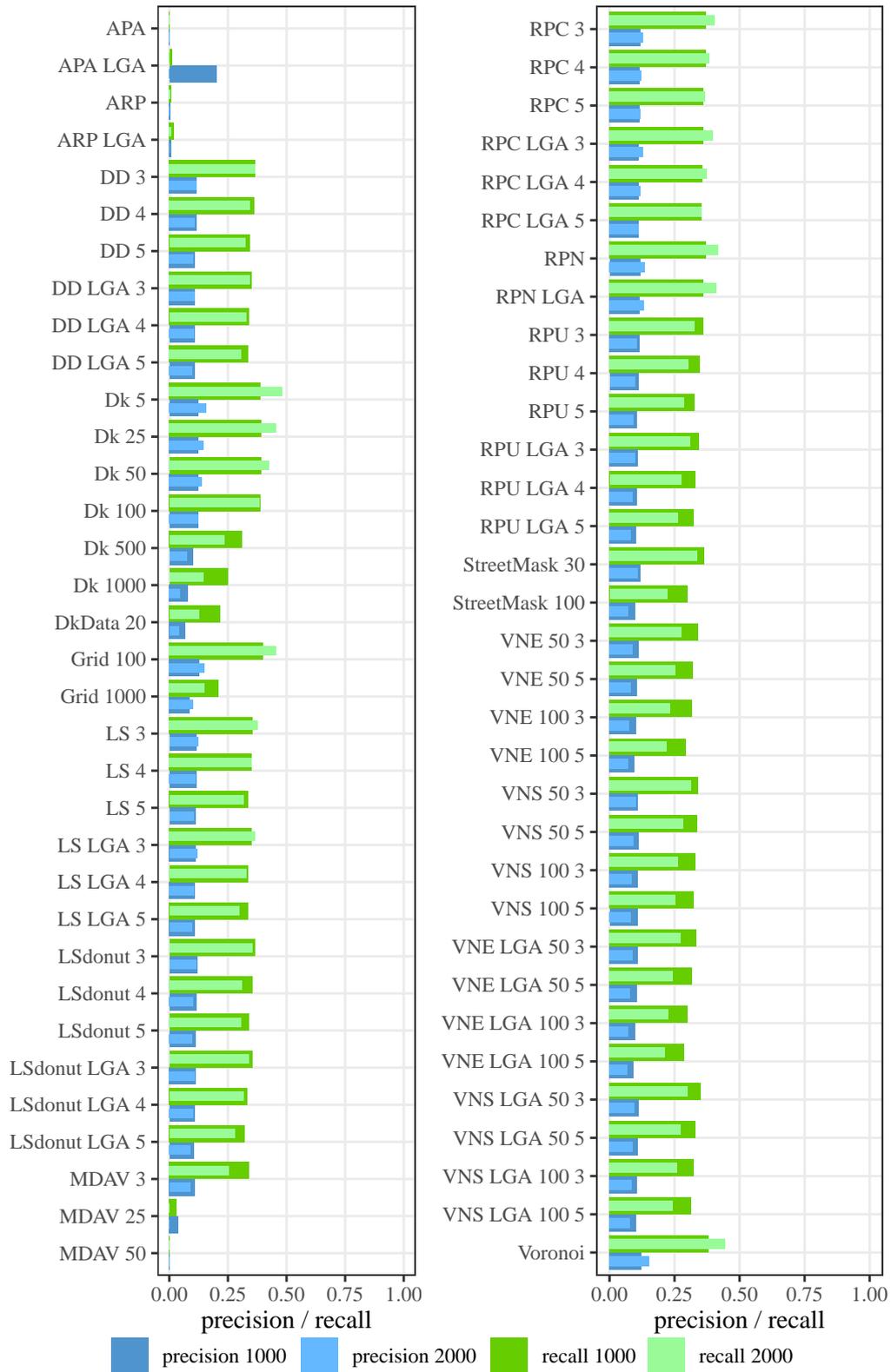


Figure 5.32.: Average precision and recall for minimum distance.

### 5.3.4. Hungarian Algorithm

The Hungarian algorithm was applied to find a one-to-one correspondence between the masked coordinates and the coordinates of the identification file. Since it is known that only 100 coordinates are matches, the distance between the matched coordinates was calculated. Then only the coordinate pairs with the 100 smallest distances were considered matches (for  $n = 2,000$  the first 200 matches). Again the precision and recall are calculated for each replication and then averaged. The result is shown in figure 5.33; the detailed results can be found in table G.16 in appendix G.10. The Hungarian algorithm without using third variables could not be applied to the full sample due to the large computation times.

Using the Hungarian algorithm and only considering the closest 100 (200 for  $n = 2,000$ ) matches shows an increase in the recall by 0.1 to 0.2. Furthermore, precision increases largely, as many of the previously considered false positives are eliminated.

Increasing the multiples decreases precision and recall when comparing different parameter choices for the same masking methods by up to 0.1. In addition, increasing  $k$  ( $k$ -nearest neighbor donut masking, VNE, VNS) decreases precision and recall. For example, the minimum requirement of at least two other points between the masked and original location (as done in 5-nearest neighbor donut masking) results in a precision and recall of 0.84 for  $n = 1,000$  and 0.72 for  $n = 2,000$ . For  $k = 1,000$ , precision and recall decrease to 0.07 for  $n = 1,000$  and 0.02 for  $n = 2,000$ . Small differences (no difference for verified neighbor approach) are seen when using postcode areas compared to local government areas for the population density. Large differences between parameter choices are seen for the official statistics grid, MDAV, and street masking.

Comparing the masking methods, it is found that donut masking shows a lower precision and recall (precision and recall between 0.42 and 0.53 for  $n = 1,000$  and between 0.28 and 0.38 for  $n = 2,000$ ) than RPC (precision and recall between 0.50 and 0.63 for  $n = 1,000$ , and between 0.37 and 0.49 for  $n = 2,000$ ), and RPN (0.60 for LGA and 0.63 for postcode areas for  $n = 1,000$  and 0.48 for LGA and 0.50 for postcode areas for  $n = 2,000$ ). However, even lower precision and recall are achieved for RPU (precision and recall between 0.32 and 0.49 for  $n = 1,000$  and between 0.20 and 0.32 for  $n = 2,000$ ). For location swapping and the donut variant of location swapping similar results are seen. Precision and recall are between 0.43 and 0.58 for  $n = 1,000$  and 0.27 and 0.42 for  $n = 2,000$  for location swapping, and for the donut variant between 0.40 and 0.58 for  $n = 1,000$  and 0.25 and 0.37 for  $n = 2,000$ . It can be seen that for the verified neighbor approach using a variable with two categories shows greater precision and recall than using a variable with more categories (about 0.1 larger for precision and recall). Again, the larger sample size ( $n = 2,000$ ) results in lower precision and recall.

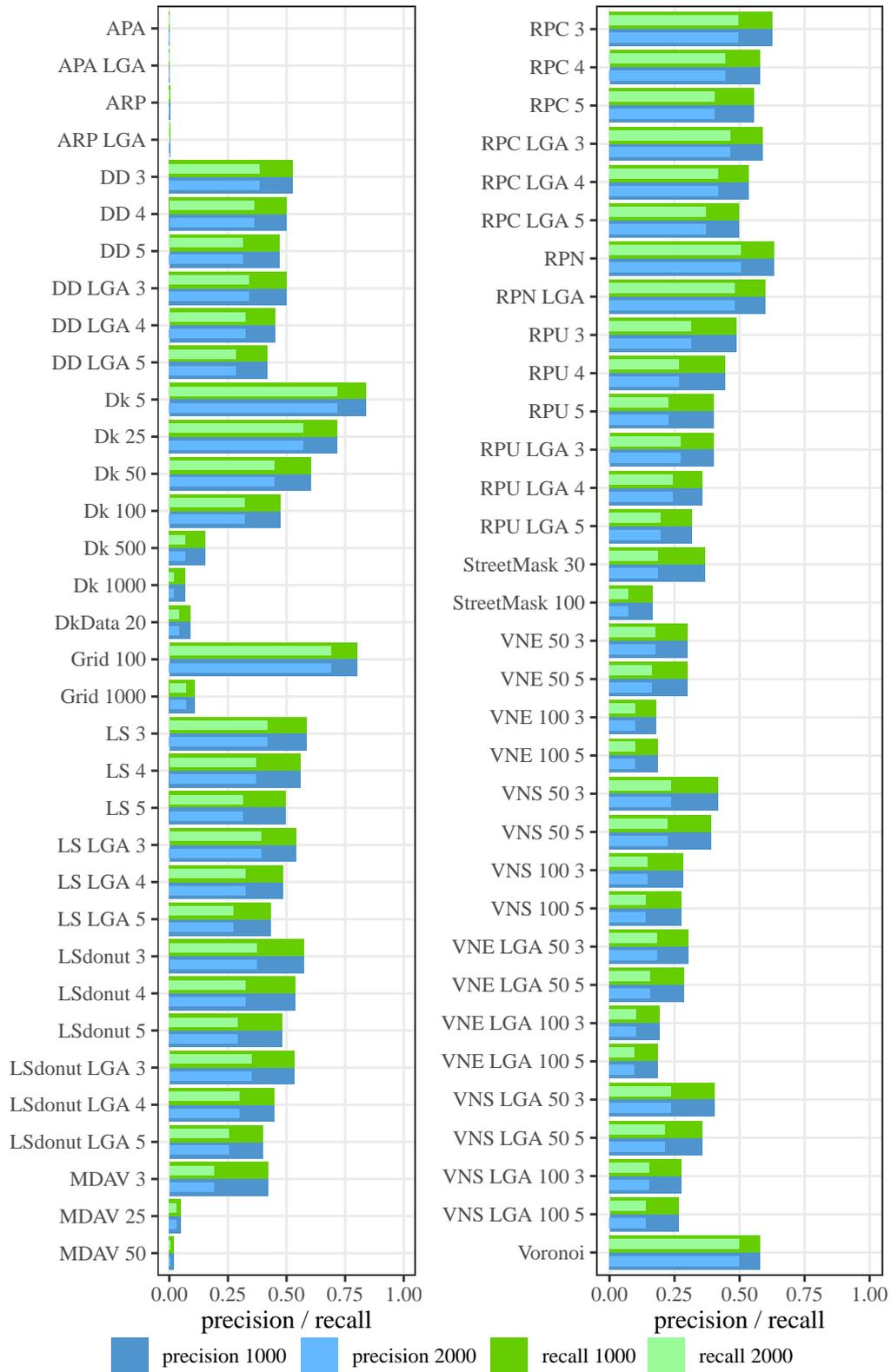


Figure 5.33.: Average precision and recall for Hungarian algorithm.

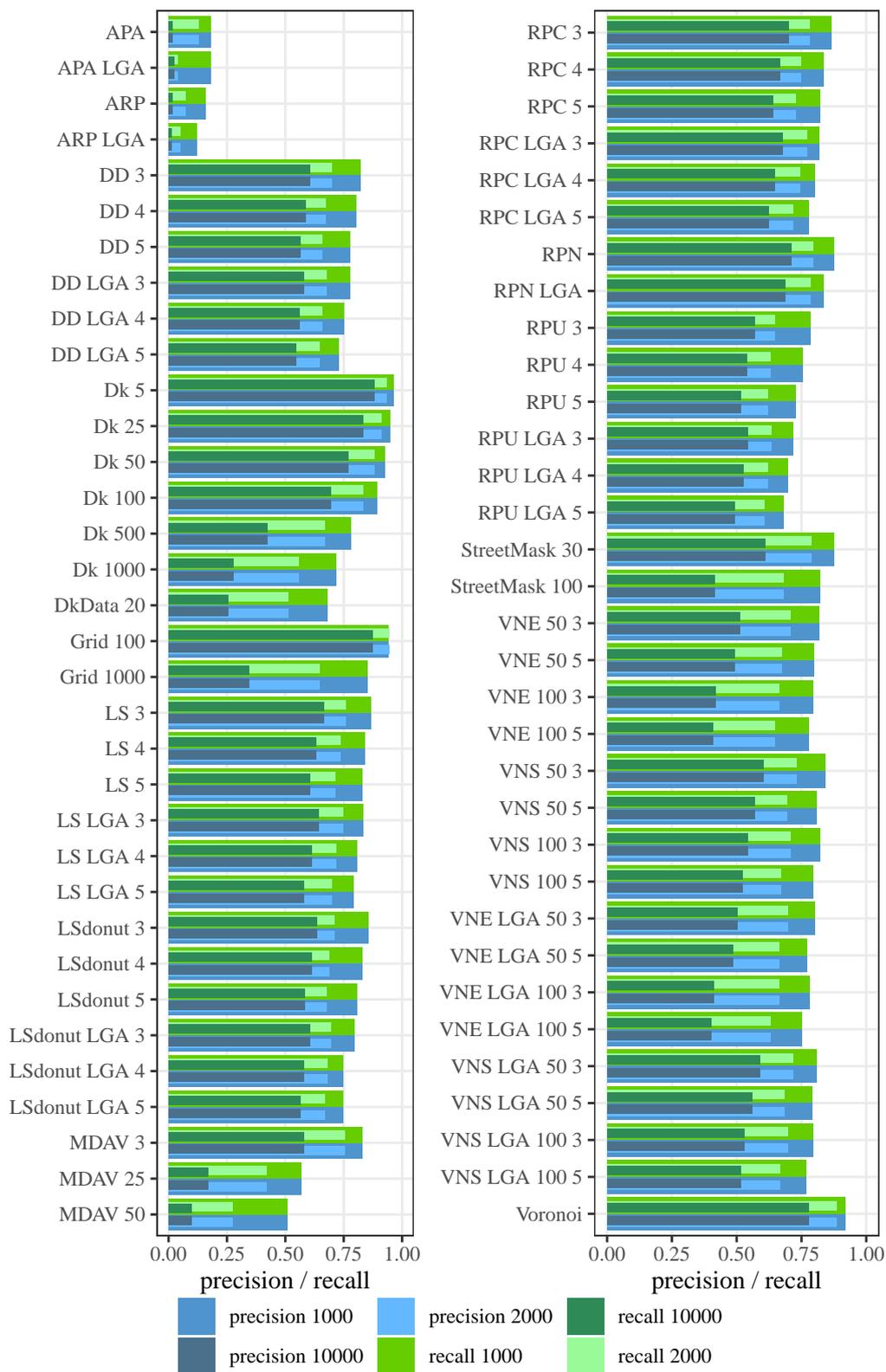


Figure 5.34.: Average precision and recall for Hungarian algorithm using additional variables.

For APA, ARP, and MDAV with cluster sizes of 25 and 50, still (almost) no original location could be identified (precision and recall of zero for AAE and up to 0.05 for MDAV). Furthermore, the Hungarian algorithm cannot be applied to the anonymization of distance matrices via Lipschitz embedding, distance approximation using ISGP, and random projections, since the input must be coordinates and not a distance matrix.

The variables sex, age, and employment status were used to limit the possible combinations to improve the results.<sup>15</sup> Thus, instead of finding a one-to-one correspondence between all coordinates of each file, a one-to-one correspondence is searched for each combination of categories for sex, age, and employment status. After assigning each coordinate in the masked file to a coordinate in the identification file, only the first 100 smallest distances between masked and identification coordinates are considered matches. For the subsample  $n = 2,000$ , the first 200 matches with the smallest distance are taken. With this modification, also the full sample of  $n = 10,000$  can be considered (thus, the 1,000 smallest distances are matches). The average precision and recall of each masking method's replications are shown in figure 5.34, the corresponding values can be found in table G.17 in appendix G.11.

As shown in figure 5.34, this attack method achieves a precision and recall of 0.7 to 0.8 for the majority of masking methods for the smallest subsample. However, for the full sample, about 0.2-0.3 lower values are seen. This shows that a drawback of this attack method is that the number of combinations, i.e., the number of groups, strongly affects precision and recall. As more people share the same combination of the three variables and the number of possible matches increases, precision and recall decrease.<sup>16</sup>

The precision and recall relationships between parameter choice and masking methods noted above remain. That is, lower precision and recall for larger multipliers, larger values for  $k$ , more categories for additional variables, and minimal differences of precision and recall between using LGA compared to postcode areas.

The  $k$ -nearest neighbor donut masking shows how much precision and recall decrease with increasing possible displacement distance. Small values such as  $k = 5$  result in almost all points of the smallest subsample being correctly identified. For the highest parameter value considered ( $k = 1,000$ ), this drops to about 0.70 for the smallest sample considered ( $n = 1,000$ ). The newly tested approach of taking the data itself as reference data for the  $k$ -nearest neighbors (Dk Data 20) shows even less precision and recall.

As for the comparison of masking methods, donut masking shows lower precision

<sup>15</sup>In record-linkage literature, limiting the number of possible matches using additional variables is termed "blocking" (see, e.g., Christen, 2012). Therefore, in tables the term "blocking" will be used when referring to limiting the possible number of matches using additional information.

<sup>16</sup>Another idea to make re-identification of coordinates more difficult was to use the residential address file as a reference for the population density instead of the true population density in South Australia. However, this reduced precision and recall for the full sample by only about 0.03-0.07 (tested for RPC 5, DD 5, and RPU 5 for ten replications).

and recall than RPC and RPN but higher precision and recall than RPU. Extending the idea of location swapping with donut masking gives similar values as using no minimum displacement distance.

With the help of additional variables, large precision and recall values for street masking are achieved for both depth values ( $n = 1,000$ : 0.87 for depth = 30 and 0.82 for depth = 100). However, with increasing sample size, precision and recall decreases for the larger depth value more ( $n = 10,000$ : 0.61 for depth = 30 and 0.41 for depth = 100).

Even these improvements to the original idea of using the minimum distance as a benchmark show that only 18% of the coordinates for the smallest subsample could be identified for APA. The recall and precision decrease to 0.13 for the subsample of size  $n = 2,000$  and to 0.02 for the complete sample. For ARP, slightly lower values for precision and recall are seen for all sample sizes.

For MDAV, precision and recall of 0.57 and 0.51 for  $n = 1,000$  could be achieved. Larger sample sizes drastically reduce the precision and recall to 0.17 and 0.10 ( $n = 10,000$ ).

### 5.3.5. Graph Theoretic Linkage Attack

For the graph theoretic linkage attack, the key to successfully identifying matches is defining the range of possible distance differences accurately. Kroll (2015, pp. 230–231) proposed using a simulation study to identify the distribution of distance differences and then using predefined quantiles to find the lower and upper values. However, this is not required for all masking methods as presented in section 4.2.6.

The average precision and recall achieved with this attack can be found in figure 5.35 and the corresponding values in table G.18. The graph theoretic linkage attack is able to identify almost all locations (of the overlap) of the given subsample for most masking methods (see figure 5.35, and table G.18 in appendix G.12). A recall of above 0.95 is achieved, and the precision is just below 0.95 (0.04-0.05 below 1.00 for most masking methods).

Exceptions are the verified neighbor approach, APA, ARP, official statistics grid using 1,000 meters, donut masking using the 5-nearest neighbor, RPN, MDAV, street masking, and Voronoi masking. Recall ranges from 0.35 to 0.85 for the verified neighbor approach, depending on the possible displacement distance and the number of categories considered for the variable of interest. However, the precision approaches up to 0.92. Multiplying the estimate of the average distance between people by 3 instead of 5 reduces recall, as does  $k = 100$  compared to  $k = 50$ , postcode population density compared to LGA, and using employment status compared to sex.

APA and ARP show major differences between using state electorates (recall of 0.69, precision of 0.24) compared to local government areas (recall of 0.17-0.18, precision of 0.14). Again it should be noted that these results were only achieved by using the

polygons as additional information.

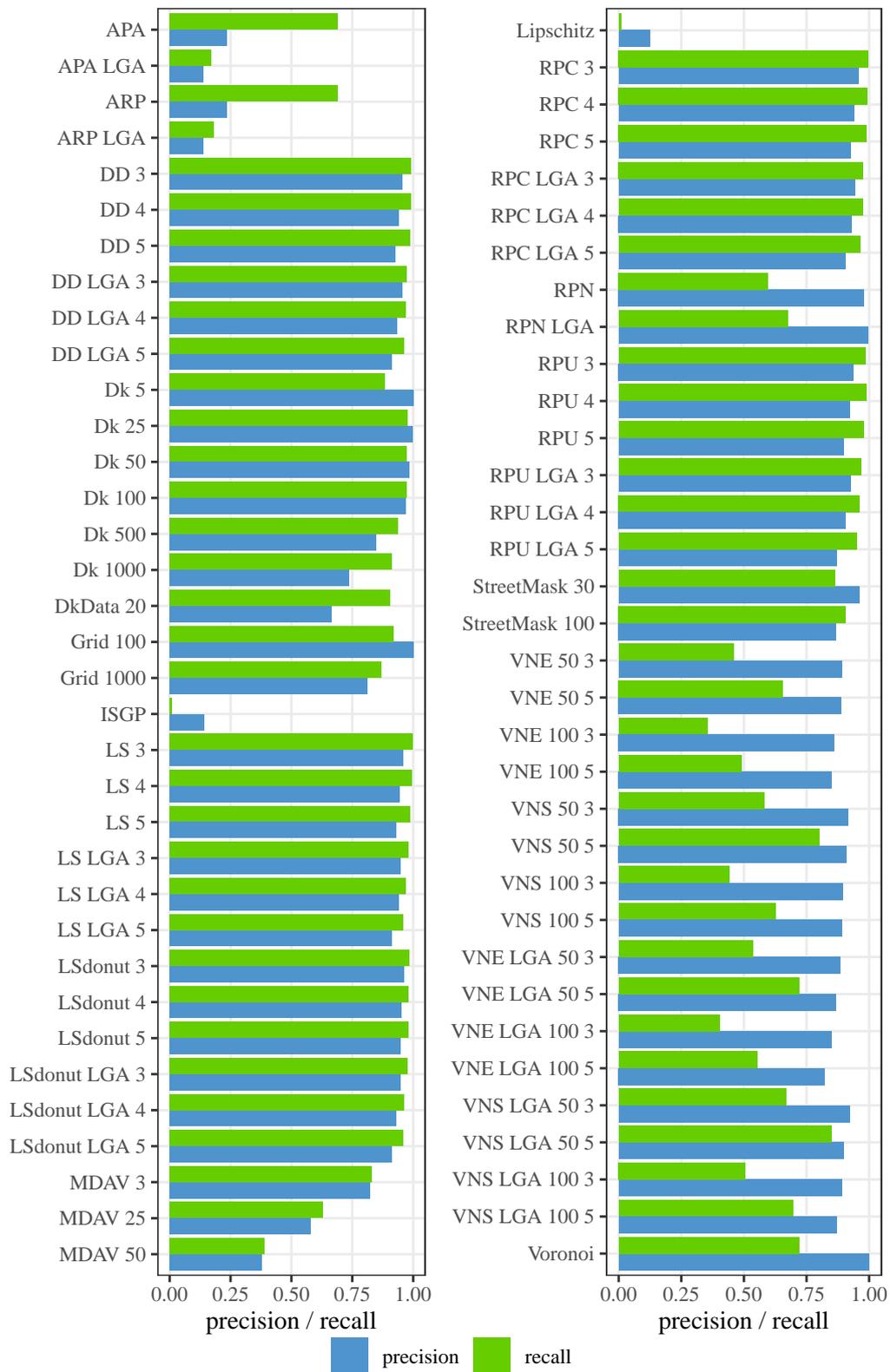


Figure 5.35.: Average precision and recall for graph theoretic linkage attack.

The difference between parameter choices of masking methods can also be seen for the official statistics grid. A 100 meters grid results in a recall of 0.92 and a precision of 1.00. A 1,000 meters grid reduces the recall to 0.87 and the precision to 0.81.

As with the other methods, MDAV also shows large differences in the results between cluster sizes. A recall of 0.83 and a precision of 0.82 are achieved for the minimum cluster size (3). Increasing the cluster size to 25 reduces recall to 0.63 and precision to 0.58. Further increasing the cluster size to 50 reduces recall to 0.39 and precision to 0.38. For street masking, increasing the depth value decreases precision (0.96 for 30 and 0.87 for 100), but increases recall (0.86 for 30 and 0.91 for 100).

Only a recall of 0.60 (0.67 if LGA is used) can be achieved for RPN, but the precision is high (0.98, 0.99 for LGA). Similarly, for Voronoi masking, a recall of 0.72 is reached with a precision of 1.00.

The graph theoretic linkage attack is one of the two attack methods that allow the input of a distance matrix. While this attack method proves to identify matching records between two data sets successfully, a major problem arises when the lower and upper values for the distance difference cannot be determined by knowing the maximum displacement distance of the masking method. Therefore, for distance anonymization via Lipschitz embedding a recall of 0.01 and a precision of 0.13 is achieved and for the approximation of distances using ISGP, a recall of 0.004 is achieved with a precision of 0.06.

### 5.3.6. Graph Matching Attack on Privacy-Preserving Record Linkage

In the description of the graph matching attack on privacy-preserving record linkage, which can, in theory, also be used for geographic masking methods, it was pointed out that several parameter choices have to be made. However, a major problem is that for most of them, there are no guidelines on how to choose them. Therefore, a small simulation study was performed for some of the parameters (see section 4.2.7).

The simulation study performed showed that the following parameters should be chosen:  $s_m = 10,000$ ,  $c_m = 10$ ,  $b_m = 0.6$ , number of hyperplanes was set to 5,000,  $cs_w = 0.5$ ,  $sc_w = 0.3$ ,  $dc_w = 0.2$ , and  $t = 100$  ( $t = 200$  for  $n = 2,000$ ). SMM (using cosine similarity) and SHM (using weighted combination) should be chosen as the methods for the bi-partite matching.

The results of this attack method for the given masking methods can be seen in figure 5.36 and 5.37, and the corresponding values in table G.19 and G.20 in appendix G.13. The results show that many true matches are lost when the results are trimmed (taking only the top 100 or 200). Moreover, SHM is not suitable to find correct matching pairs for geographic masking methods (see figure 5.37). In comparison, SMM performs better (see figure 5.36).

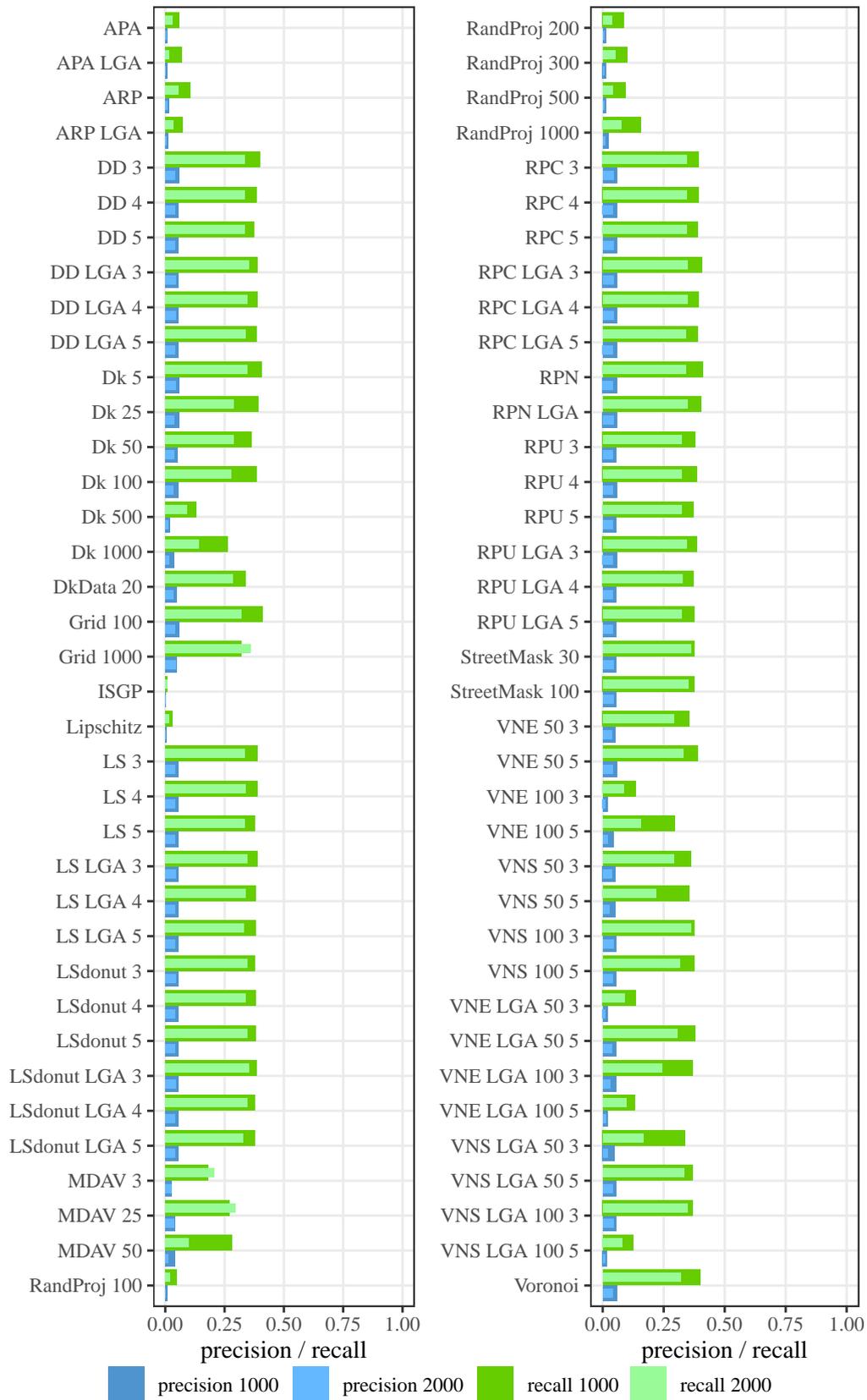


Figure 5.36.: Average precision and recall for graph matching PPRL attack using stable marriage match.

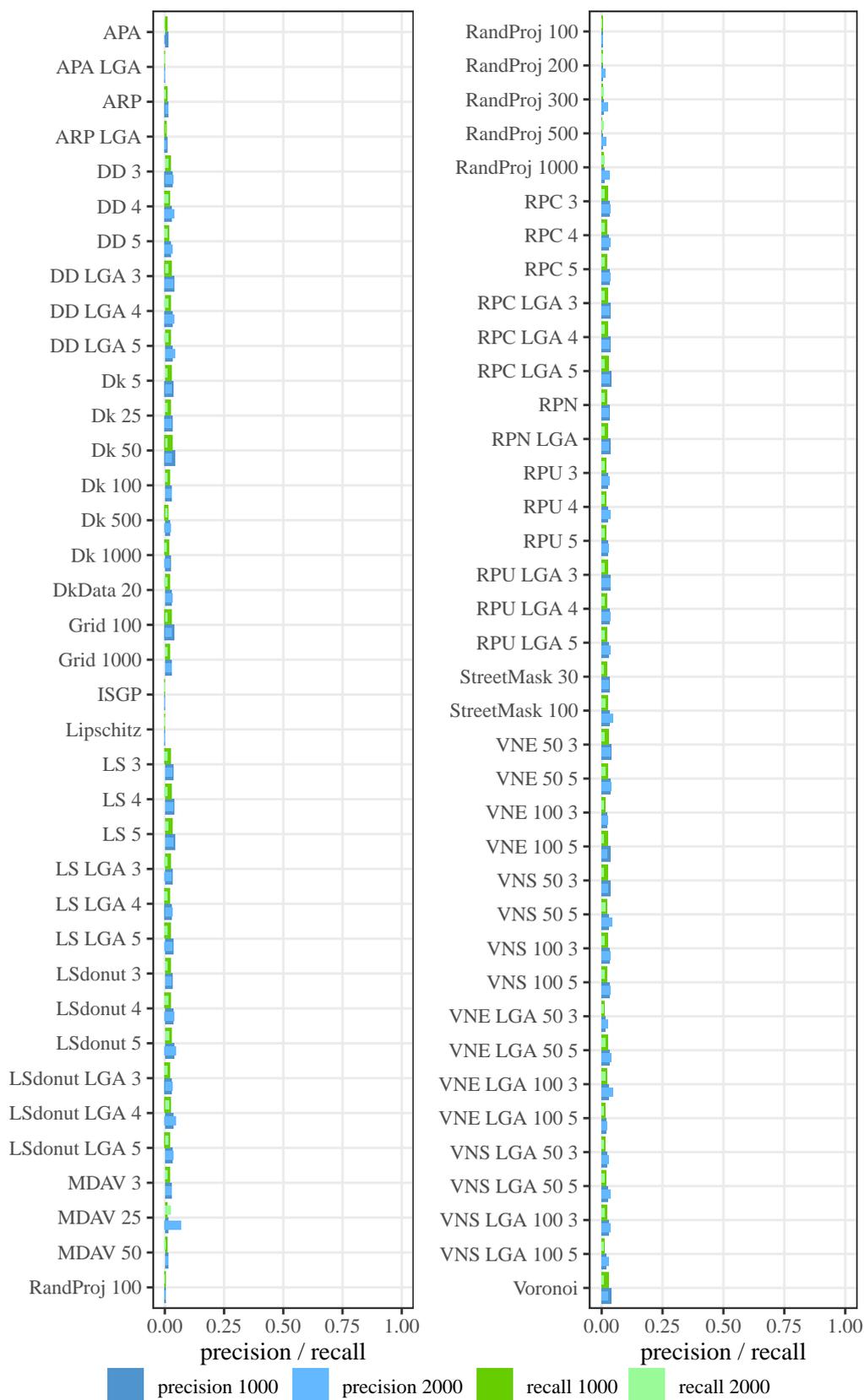


Figure 5.37.: Average precision and recall for graph matching PPRL attack using symmetric highest match.

For the smaller subset of the sample ( $n = 1,000$ ), a maximum recall of 0.41 is achieved, but the precision is close to zero. Thus, the attack method can correctly identify several locations, but many more are false positives. Moreover, this attack method does not work well for the methods anonymization of distance matrices via Lipschitz embedding, distance approximation using ISGP, APA, and ARP.

Unlike other attack methods, there is only a small difference in results between the  $n = 1,000$  sample and the  $n = 2,000$  sample.<sup>17</sup> Another advantage is that the attack method can work with the bit vector output of the random projection method.

### 5.3.7. Aggregated Results by Geomasking Methods

A major drawback of most attack methods is that much larger resources (computing power and time) are required for large data sets, such as  $n = 10,000$  coordinates.<sup>18</sup> The Hungarian algorithm with the additional use of third variables is the only attack method that can easily handle even larger data sets. However, this depends on the number of categories that can be formed using third variables. Furthermore, only the graph matching attack on privacy-preserving record linkage and the graph theoretic linkage attack allows the input of a distance matrix. The latter then shows problems in defining a suitable value for the lower and upper values for the difference in distances. In turn, the graph matching attack on privacy-preserving record linkage does not perform as well if the overlap between data sets is small. Moreover, various parameters have to be defined for this attack method.

For most of the masking methods, at least one attack method can correctly re-identify most locations. Only APA, ARP, anonymization of distance matrices via Lipschitz embedding, distance approximation using ISGP, and random projection resist almost complete re-identification with these attack methods. Most of these masking methods only allow the release of a distance matrix rather than coordinate pairs. Lastly, the affine transformation methods can easily be reversed, yielding a precision and recall of one. In the following, the results of the attack methods aggregated by masking methods are reported.<sup>19</sup>

#### 5.3.7.1. Minimum Distance

The aggregated results of the minimum distance by masking method is shown in table 5.10. Donut density,  $k$ -nearest neighbor donut masking, location swapping

<sup>17</sup>Due to the long computation times, the graph matching attack on privacy-preserving record linkage for the  $n = 2,000$  sample was only applied on the first 30 replications. However, the first 30 replications showed very little variation in the results, so that it is not expected that this would change when using all 50 replications.

<sup>18</sup>For this thesis, in particular, the large number of masking methods and replications of masking methods do not allow for larger sample sizes. One reason could be the use of  $R$ .

<sup>19</sup>The results for the mean of multiple releases and reversing masking methods are not reported again. The former showed poor results, while the latter was applied only to the affine transformations.

(both variants), random perturbation method, verified neighbor methods, and Voronoi masking show very similar results for the smallest sample size considered. The recall ranges from 0.3 to 0.37, and the precision ranges from 0.1 to 0.12. Thus, only 30-37% of the original locations could be re-identified. However, this attack method is not very precise in doing so and yields many false positives.

For the larger subsample of  $n = 2,000$ , precision and recall decrease only slightly. Even when the data set is used as a reference file for donut masking, a recall of 0.22 and a precision of 0.07 are achieved, which are reduced to 0.13 and 0.04 for the larger subsample. MDAV also shows low values for precision and recall. The recall of 0.11 is only achieved by the variant with a minimum cluster size of 3. APA and ARP are masking methods that do not allow re-identification with the minimum distance method.

Table 5.10.: Results of minimum distance aggregated by masking methods.

method	$n = 1,000$		$n = 2,000$	
	precision	recall	precision	recall
APA	0.10	0.00	0.00	0.00
ARP	0.01	0.01	0.00	0.01
DD	0.11	0.35	0.11	0.34
Dk	0.11	0.35	0.11	0.35
DkData	0.07	0.22	0.04	0.13
Grid	0.11	0.30	0.13	0.30
LS	0.11	0.34	0.11	0.34
LSdonut	0.11	0.34	0.10	0.32
MDAV	0.05	0.12	0.03	0.08
RPC	0.11	0.36	0.12	0.38
RPN	0.12	0.37	0.13	0.41
RPU	0.11	0.34	0.09	0.29
StreetMask	0.11	0.33	0.09	0.28
VNE	0.10	0.31	0.08	0.24
VNS	0.11	0.33	0.09	0.27
Voronoi	0.12	0.38	0.15	0.44

### 5.3.7.2. Hungarian Algorithm

Using the overall minimum distance (the Hungarian algorithm) and including a threshold excludes many false positives and increases precision (see table 5.11). The Hungarian algorithm yields better results for donut masking, location swapping, random perturbation methods, and Voronoi masking. However, in terms of recall, it does not work as well for verified neighbor methods,  $k$ -nearest neighbor donut masking using the data set as the reference file, and street masking. Better results are obtained

for MDAV, as locations can be identified even for cluster sizes 25 and 50. APA and ARP still preserve the privacy of the respondents. Again, for most masking methods larger differences in precision and recall are seen between the two subsamples (about 0.1-0.2 decrease).

By using additional information to limit the number of potential matches further, the Hungarian algorithm can identify some locations for APA and ARP (see table 5.12). However, precision and recall decrease rapidly with increasing sample size. For Voronoi masking and official statistics grid, precision and recall increase to at least 0.9. For  $k$ -nearest neighbor donut masking and MDAV precision and recall values above 0.6 are achieved. The other masking methods show precision and recall of 0.8-0.9.

Table 5.11.: Results of Hungarian algorithm aggregated by masking methods.

method	$n = 1,000$		$n = 2,000$	
	precision	recall	precision	recall
APA	0.00	0.00	0.00	0.00
ARP	0.00	0.00	0.00	0.00
DD	0.48	0.48	0.34	0.34
Dk	0.48	0.48	0.36	0.36
DkData	0.09	0.09	0.04	0.04
Grid	0.46	0.46	0.38	0.38
LS	0.52	0.52	0.35	0.35
LSdonut	0.50	0.50	0.32	0.32
MDAV	0.16	0.16	0.07	0.07
RPC	0.56	0.56	0.43	0.43
RPN	0.62	0.62	0.49	0.49
RPU	0.40	0.40	0.25	0.25
StreetMask	0.27	0.27	0.13	0.13
VNE	0.24	0.24	0.13	0.13
VNS	0.33	0.33	0.19	0.19
Voronoi	0.58	0.58	0.50	0.50

Increasing the sample size from  $n = 1,000$  to  $n = 2,000$  decreases precision and recall values by 0.17 for  $k$ -nearest neighbor donut masking, by 0.1-0.11 for location swapping, official statistics grid, donut masking using population density, RPU, and the verified neighbor approach using sex as a variable. For the latter, using employment status as the variable of interest and the methods APA, ARP, and street masking precision and recall decrease by 0.12. For MDAV, precision and recall decrease by 0.16 when the sample size is increased. Voronoi masking, RPU, and RPN decrease precision and recall by 0.04-0.07.

When the sample size is further increased ( $n = 10,000$ ), APA and ARP decrease again by 0.04 to 0.06. Donut masking using population density, random perturbation

methods, Voronoi masking, and location swapping show a decrease of 0.09-0.11 for precision and recall. VNS and  $k$ -nearest neighbor donut masking show a reduction of 0.15 for precision and recall. Using the data set as the reference file for  $k$ -nearest neighbor reduces precision and recall to 0.26. Lastly, for the official statistics grid precision and recall decrease by 0.18, MDAV by 0.2, and VNE and street masking by 0.22.

Table 5.12.: Results of Hungarian algorithm using additional variables.

method	$n = 1,000$		$n = 2,000$		$n = 10,000$	
	precision	recall	precision	recall	precision	recall
APA	0.18	0.18	0.08	0.08	0.02	0.02
ARP	0.14	0.14	0.06	0.06	0.02	0.02
DD	0.78	0.78	0.67	0.67	0.57	0.57
Dk	0.87	0.87	0.80	0.80	0.65	0.65
DkData	0.68	0.68	0.51	0.51	0.26	0.26
Grid	0.90	0.90	0.79	0.79	0.61	0.61
LS	0.83	0.83	0.73	0.73	0.62	0.62
LSdonut	0.80	0.80	0.69	0.69	0.60	0.60
MDAV	0.64	0.64	0.48	0.48	0.28	0.28
RPC	0.82	0.82	0.75	0.75	0.66	0.66
RPN	0.86	0.86	0.79	0.79	0.70	0.70
RPU	0.73	0.73	0.63	0.63	0.53	0.53
StreetMask	0.85	0.85	0.73	0.73	0.51	0.51
VNE	0.79	0.79	0.67	0.67	0.45	0.45
VNS	0.80	0.80	0.70	0.70	0.55	0.55
Voronoi	0.92	0.92	0.88	0.88	0.78	0.78

### 5.3.7.3. Graph Theoretic Linkage Attack

Table 5.13 shows the results for the graph theoretic linkage attack aggregated by masking methods.

The recall increases drastically for APA and ARP, but only due to the use of the polygons as additional information. This attack method can identify almost all locations for the donut masking methods, the official statistics grid, RPC, RPU, and location swapping methods. Also, this attack method obtains a higher precision and reduces the number of false positives even more. However,  $k$ -nearest neighbor donut masking with the data set as reference file yields similar precision to the Hungarian algorithm with additional variables.

While this attack method is applicable for distance approximation using ISGP and anonymization of distance matrices via Lipschitz embedding, it does not work well. MDAV also does not show as good results as the previous attack method because of

the difficulty in defining the lower and upper bounds for the difference in distance. Compared to the Hungarian algorithm worse results are obtained for recall for Voronoi masking, the verified neighbor approach, and RPN, but better precision results are obtained.

Table 5.13.: Results of graph theoretic linkage attack aggregated by masking methods.

method	precision	recall
APA	0.19	0.43
ARP	0.19	0.43
DD	0.94	0.98
Dk	0.92	0.94
DkData	0.66	0.90
Grid	0.91	0.90
ISGP	0.06	0.00
Lipschitz Embedding	0.13	0.01
LS	0.94	0.98
LSdonut	0.94	0.97
MDAV	0.47	0.37
RPC	0.93	0.98
RPN	0.99	0.64
RPU	0.91	0.97
StreetMask	0.91	0.88
VNE	0.86	0.52
VNS	0.90	0.65
Voronoi	1.00	0.72

#### 5.3.7.4. Graph Matching Attack on Privacy-Preserving Record Linkage

Table 5.14 shows the results for the graph matching attack on privacy-preserving record linkage.<sup>20</sup>

As can be seen, this attack method does not work well for any of the masking methods. While it can achieve moderate results for recall, the precision is very low. And while it is the only attack method that can, in theory, identify records when they have been masked using random projection, precision and recall are close to zero.

<sup>20</sup>Results when using only the top 100 (200 for  $n = 2,000$ ) matches based on the similarity obtains even worse results and thus are not reported.

Table 5.14.: Results of graph matching attack on privacy-preserving record linkage aggregated by masking methods.

method	$n = 1,000$				$n = 2,000$			
	SMM		SHM		SMM		SMM	
	prec	rec	prec	rec	prec	rec	prec	rec
APA	0.01	0.06	0.01	0.00	0.00	0.02	0.00	0.00
ARP	0.01	0.09	0.01	0.01	0.01	0.04	0.01	0.00
DD	0.06	0.39	0.03	0.02	0.04	0.34	0.04	0.01
Dk	0.05	0.32	0.03	0.02	0.03	0.24	0.03	0.01
DkData	0.05	0.34	0.03	0.02	0.04	0.28	0.03	0.01
Grid	0.05	0.36	0.03	0.02	0.04	0.34	0.03	0.01
ISGP	0.00	0.01	0.00	0.00	0.00	0.01	0.00	0.00
Lipschitz Embedding	0.00	0.03	0.00	0.00	0.00	0.01	0.00	0.00
LS	0.06	0.38	0.04	0.03	0.04	0.34	0.03	0.01
LSdonut	0.06	0.38	0.03	0.02	0.04	0.34	0.04	0.01
MDAV	0.04	0.24	0.02	0.01	0.02	0.20	0.04	0.01
RandProj	0.01	0.10	0.01	0.00	0.01	0.05	0.02	0.01
RPC	0.06	0.39	0.03	0.03	0.04	0.35	0.04	0.01
RPN	0.06	0.41	0.03	0.02	0.04	0.34	0.03	0.01
RPU	0.06	0.38	0.03	0.02	0.04	0.33	0.04	0.01
StreetMask	0.06	0.38	0.03	0.02	0.04	0.36	0.04	0.01
VNE	0.04	0.27	0.03	0.02	0.02	0.20	0.03	0.01
VNS	0.05	0.33	0.03	0.02	0.03	0.26	0.03	0.01
Voronoi	0.06	0.40	0.04	0.03	0.04	0.32	0.03	0.01

To conclude, the success of fairly simple attack methods such as the minimum distance and the Hungarian algorithm shows that most masking methods do not displace the coordinates far enough to be not re-identified. This conclusion is also supported by the fact that parameter choices that allow smaller displacements show higher recall and precision values. Moreover, it is shown that re-identification is also possible even when  $k$ -anonymity for spatial data is a requirement for masking methods. For  $k$ -nearest neighbor donut masking and street masking, re-identification is possible if an identification file is available.



## 6. R-U Confidentiality Map of Geomasking Methods

The relationship between the risk of re-identification and the utility of the spatial information is shown in R-U confidentiality maps. As described in chapter 4, in a R-U confidentiality map (also referred to as risk-utility map) the utility is shown on the x-axis and the risk on the y-axis (Duncan and Fienberg, 1999, p. 352; Duncan, Fienberg, et al., 2001, p. 139; Duncan, Keller-McNulty, et al., 2001, p. 7). Ideally, a masking method would be located in the lower right quadrant, which would indicate high utility and low risk. Since in the GDi, LDi and MSE values close to zero indicate high utility, in this thesis, the x-axis maps the difference to the utility of the original data set (utility loss).<sup>1</sup>

### 6.1. Components of the Risk-Utility Maps

First, the MPR is calculated to obtain one value for each masking method used in the risk-utility map. Afterward, the results of the two utility measures (average of GDi and LDi, and MSE) are shown.

#### 6.1.1. MPR

The results for precision and recall are averaged, yielding the MPR (see, e.g., Borgs, 2019). The MPR aggregated by masking methods can be seen in table 6.1.<sup>2</sup> For each risk method (and each sample size), the highest MPR of the different sample sizes was taken for each masking method for the risk-utility map. Some of the risk methods can only be applied to a subset of the data. Therefore, the results based on the full sample and subsets  $n = 1,000$  and  $n = 2,000$  are shown. The MPR for the full sample could only be calculated using the Hungarian algorithm with the help of additional attributes and a threshold to limit the matches to the size of the overlap. Therefore, no MPR for the full sample is shown for anonymization of distance matrices via Lipschitz embedding, random projection, and distance approximation using ISGP since the only applicable risk method does not allow the input of a distance matrix. For the subsets,  $n = 1,000$  and  $n = 2,000$ , the minimum distance method, the Hungarian

---

<sup>1</sup>Choosing to use negative values to measure utility has also been done by Kroll (2014b, pp. 18–19).

<sup>2</sup>The MPR of the individual parameter choices for each masking method is shown in table I.1 in appendix I.

methods, the graph matching attack on PPR, and the graph theoretic linkage attack ( $n = 1,000$  only) could be used. The reversing masking methods approach was used for affine transformations (DUT, CS, Rot, RotArb). Since this proved to be successful, the other attack methods were not considered.

As expected, the MPR decreases with increasing sample size even when additional variables are used because the number of possible matches per group (combinations of attributes of the additional variables) increases. The decrease of the MPR, considering the size differences, between  $n = 1,000$  and  $n = 2,000$  is much larger than to the full sample ( $n = 10,000$ ). As a comparison, sample sizes of data sets in social science is usually around 3,000 completed questionnaires (e.g., Allbus, ESS Germany). Large differences of the MPRs between sample sizes are seen for all masking methods except affine transformations and the masking methods that do not provide coordinates as output. With affine transformations, all locations could be re-identified despite an increase in sample size. On the other hand, distance approximation using ISGP, anonymization of distance matrices via Lipschitz embedding, and random projection resulted in almost no re-identified locations for all sample sizes.

Table 6.1.: MPR for full sample as well as subsamples.

method	MPR		
	$n = 1,000$	$n = 2,000$	$n = 10,000$
orig	1.00	1.00	1.00
APA	0.32	0.08	0.02
ARP	0.31	0.06	0.02
CS	1.00	1.00	1.00
DD	0.96	0.67	0.57
Dk	0.94	0.80	0.65
DkData	0.78	0.51	0.26
DUT	1.00	1.00	1.00
Grid	0.90	0.79	0.61
ISGP	0.03	0.01	
Lipschitz Embedding	0.07	0.01	
LS	0.96	0.73	0.62
LSdonut	0.96	0.69	0.60
MDAV	0.65	0.48	0.28
RandProj	0.06	0.03	
Rot	1.00	1.00	1.00
RotArb	1.00	1.00	1.00
RPC	0.96	0.75	0.66
RPN	0.86	0.79	0.70
RPU	0.94	0.63	0.53
StreetMask	0.90	0.73	0.51
VNE	0.79	0.67	0.45
VNS	0.82	0.70	0.55
Voronoi	0.92	0.88	0.78

Most masking methods for  $n = 1,000$  show MPRs between 0.9 and 1, namely, affine transformations (CS, DUT, Rot, RotArb), donut masking, the official statistics grid, location swapping, RPC, RPU, street masking, and Voronoi masking. RPN yields a slightly lower value of 0.86. The verified neighbor methods yield an MPR of 0.82 for a binary variable (VNS) and 0.79 for more categories (VNE). When  $k$ -nearest neighbor donut masking is applied using the data set as the reference file (DkData), the MPR decreases to 0.78. MDAV shows a value of 0.65. The lowest MPRs and thus, masking methods that preserve the privacy of the original locations the most are APA (0.32) and ARP (0.31) as well as distance approximation using ISGP (0.03), anonymization of distance matrices via Lipschitz embedding (0.07), and random projection (0.06).

For the subsample of  $n = 1,000$ , the graph theoretic linkage attack achieves the largest MPR for most masking methods. Only for RPN, verified neighbor approach, Voronoi masking, MDAV, and random projection did another risk method yield higher values for precision and recall. Masked points with RPN can be identified more successfully with the Hungarian algorithm with additional variables. This is due to the fact that an upper bound needed for the graph theoretic linkage attack is more difficult to define than with the same masking method but using a uniform distribution (RPU). This is also true for MDAV. Points masked with random projection could not be identified with the graph theoretic linkage attack because the difference in distances could not be adequately defined. The given MPR was achieved with the graph matching attack on privacy-preserving record linkage. The verified neighbor approach shows similar MPR values for the graph theoretic linkage attack and the Hungarian algorithm with additional variables. Therefore, some parameter choices show larger MPR values for the Hungarian algorithm, while other parameter choices show larger MPR values for the graph attack. This is also true for APA using local government areas, 5-nearest neighbor donut masking, and the official statistics grid with a 1,000 meters grid size. For Voronoi masking, the lack of ability to define differences in distances results in the graph theoretic linkage attack not identifying as many records as the Hungarian algorithm with additional variables.

For the larger sample size of  $n = 2,000$ , the affine transformations (CS, DUT, Rot, RotArb) again do not preserve privacy. But, APA, ARP, distance approximation using ISGP, anonymization of distance matrices via Lipschitz embedding, and random projection still prove to preserve privacy well (MPR of 0.01-0.08). Large MPR values are seen for  $k$ -nearest neighbor donut masking, the official statistics grid, and Voronoi masking (0.80, 0.79, and 0.88). Slightly lower but still large values for MPR are found for the random perturbation methods (0.63-0.79), verified neighbor approach (0.67-0.70), location swapping (0.73), and street masking (0.73). A more substantial reduction in MPR is found for location swapping using the donut variant (0.69) and donut masking based on population density (0.67). The  $k$ -nearest neighbor variant with the data set as reference (DkData) and MDAV only show an MPR of about 0.5.

For the larger subsample of  $n = 2,000$ , only a handful of risk methods were

used: the minimum distance method, the Hungarian algorithm variants, and the graph matching attack on privacy-preserving record linkage. Here, the Hungarian algorithm using additional variables proves to be superior to the other risk methods. Only distance approximation using ISGP, random projection, and anonymization of distance matrices via Lipschitz embedding (masking methods for which the Hungarian algorithm cannot be applied) show the largest MPR with the graph matching attack on privacy-preserving record linkage. But the “largest” MPR is close to zero.

The total sample still shows a near-zero MPR for APA and ARP. The MPR is decreased by about 0.1 for donut masking using population density, location swapping, random perturbation, and Voronoi masking. A decrease of 0.15 to 0.2 is found for  $k$ -nearest neighbor donut masking using the resident file as reference, the official statistics grid, MDAV, street masking, and the verified neighbor approach. For the total sample ( $n = 10,000$ ) only the Hungarian algorithm with additional variables could be applied, since other attack methods are too computationally intensive.

### 6.1.2. GD<sub>i</sub> and LD<sub>i</sub>

According to the formulas given in chapter 4, the GD<sub>i</sub> (M<sub>di</sub>, O<sub>di</sub>, MA<sub>di</sub>) and LD<sub>i</sub> (Clus, SpatAutCorr) were calculated for each masking method. The GD<sub>i</sub> and LD<sub>i</sub> are the averages of the individual indices (Kounadi and Leitner, 2015, p. 744). The first step was to calculate each element of the GD<sub>i</sub> and LD<sub>i</sub> for every replication. The results are then averaged over the fifty replications.<sup>3</sup> Finally, to yield one value for the individual indices for each masking method, the individual indices were averaged over the different parameter choices. To position the masking methods in a risk-utility map, the GD<sub>i</sub> and LD<sub>i</sub> are averaged.<sup>4</sup>

For SpatAutCorr, two variables were considered (proportion of single households and proportion of full-time working people). Therefore, the SpatAutCorr was calculated separately for each variable and then averaged. Similarly, for Clus, the number of clusters was evaluated and whether points remained clustered or non-clustered. Moreover, the total number of clusters and clusters with at least 30 elements were considered to show whether masking methods preserve the total number of clusters and clusters with a reasonable size (at least 30 elements). Furthermore, two different radii were considered for the clustering algorithm. Thus for every replication the total number of clusters as well as the number of clusters with at least 30 elements (ClusNum), and the number of points clustered in the original data set but not in the masked data set and vice versa (ClusNoise) was evaluated for  $\varepsilon = 3, 200$  and  $\varepsilon = 9, 500$ . The results were then averaged over the different radii to yield one value (ClusNum) for the total number of clusters, one value for clusters with at least 30 elements

<sup>3</sup>Results for individual parameter choices for each masking method can be seen in table I.2 in appendix I.2.

<sup>4</sup>See figure H.1 in appendix H for an overview.

(ClusNum), and one value for the comparison of the number of points that remained clustered respectively non-clustered (ClusNoise). After that the two ClusNum values were averaged before combining it with ClusNoise to yield the index Clus.

For the calculation of the Mdi, the distance of the original spatial mean center to the farthest point of the study area must be found (Kounadi and Leitner, 2015, p. 745). For this, the shapefile was used, and the distance between the spatial mean center and the points forming the polygon of South Australia (border of South Australia) was calculated. This results in a maximum distance of 1,344,030.676 meters. In addition for the MAdi, the maximum major axis must be found. This is done by finding the two most distant points of the study area and then using these points to calculate the major axis of the standard deviational ellipse (Kounadi and Leitner, 2015, p. 746). This resulted in a maximum major axis of 873,197.433 meters.

Using the idea of Kounadi and Leitner (2015) of a global and a local utility measure shows that most masking methods have a GD<sub>i</sub> and LD<sub>i</sub> of less than 10 (table 6.2). GD<sub>i</sub> and LD<sub>i</sub>, as well as the individual indices are usually between zero and 100, where zero indicates that no difference to the results of the original data is found. 100 shows the maximum difference between the original and masked data sets results (Kounadi and Leitner, 2015, p. 744). However, as seen in table 6.2, the indices can be greater than 100 if the difference between the original and masked measure is even further than the definition of the maximum difference by Kounadi and Leitner (2015). For example, for the Mdi (spatial mean center), it is assumed that the maximum difference is reached when the spatial mean center is displaced to the furthest point of the study area (Kounadi and Leitner, 2015, p. 745). However, if the spatial mean center is positioned outside of the study area as when rotating around the origin of the coordinate system, the distance between the spatial mean centers is even larger.<sup>5</sup>

Viewing the individual components of the GD<sub>i</sub> and LD<sub>i</sub> shows that setting the difference from the original results in relation to the maximum possible value results in many of the differences between masking methods not being seen. Except for APA and ARP, none of the masking methods show differences from the original results in all dimensions. Mdi, MAdi, and Clus show that the maximum value is too large to reveal differences between the masking methods.

The Mdi shows high values only for CS and rotation. For all others, there is a Mdi of less than 1%. This includes the masking methods APA, ARP, and DUT, which displace the mean center on average by about 12 km (APA, ARP) and 7.7 km (DUT).

Odi is based on the angle of the standard deviational ellipse. As shown in chapter 5, the orientation is preserved for most methods. Only for the rotation methods and distance approximation using ISGP the orientation of the ellipse is not preserved. For the latter, the standard deviational ellipse could only be calculated using approximated

---

<sup>5</sup>A solution would be to set the maximum distance between the spatial mean center to the found maximum distance (if larger than to the furthest point of the study area). However, this would cause that here almost no difference in the indices between masking methods can be seen.

Table 6.2.: Results of the calculation of the GDi and LDi and its components according to Kounadi and Leitner (2015) (sorted by the average of the GDi and LDi, referred to as utility).

method	GDi				LDi			
	Mdi	Odi	MAdi	GDi	Clus	SpatAut	LDi	utility
orig	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DUT	0.57	0.00	0.00	0.19	0.00	0.00	0.00	0.10
LSdonut	0.00	0.04	0.02	0.02	0.19	0.82	0.50	0.26
LS	0.00	0.02	0.00	0.01	0.15	1.03	0.59	0.30
StreetMask	0.02	0.18	0.06	0.09	0.33	0.69	0.51	0.30
VNS	0.00	0.04	0.03	0.02	0.39	1.16	0.78	0.40
Dk	0.00	0.05	0.02	0.02	0.74	1.43	1.09	0.56
VNE	0.01	0.13	0.06	0.07	0.68	1.53	1.10	0.59
RPN	0.00	0.01	0.00	0.01	0.33	2.19	1.26	0.63
Lipschitz	0.00	2.22	2.07	1.43	0.00	0.43	0.22	0.82
RPC	0.00	0.02	0.01	0.01	0.61	2.81	1.71	0.86
Grid	0.00	0.00	0.00	0.00	0.02	3.73	1.88	0.94
DD	0.00	0.03	0.01	0.01	0.88	3.25	2.07	1.04
RPU	0.00	0.04	0.01	0.02	1.28	3.59	2.43	1.23
DkData	0.01	0.18	0.07	0.09	3.11	4.41	3.76	1.93
MdAV	0.00	0.10	0.13	0.07	2.59	7.56	5.07	2.57
Voronoi	0.00	0.02	0.02	0.01	0.33	10.99	5.66	2.84
APA	0.87	0.44	4.01	1.77	3.11	12.20	7.66	4.72
RotArb	0.00	28.62	0.00	9.54	0.00	0.01	0.00	4.77
ARP	0.90	1.03	5.00	2.31	6.65	9.65	8.15	5.23
ISGP	0.00	15.33	12.54	9.29	0.00	2.80	1.40	5.35
CS	87.02	0.00	8.27	31.76	0.89	0.01	0.45	16.10
Rot	189.40	31.09	0.00	73.50	0.08	0.01	0.05	36.77

coordinates.

The major axis of the standard deviational ellipse is viewed for the MAdi. Here, too, hardly any differences between masking methods can be seen since the maximum possible value is very large. Moreover, the GDi for distance approximation using ISGP and anonymization of distance matrices via Lipschitz embedding is based on the approximated coordinates since these methods yield a distance matrix as output.

For clustering, most masking methods show a distortion from the original clustering of less than 1%. APA, ARP, and DkData are the masking methods that preserve the original clustering the least. With APA, the total number of clusters is less than half of the original, twice as much for clusters with at least 30 elements and no non-clustered points, which is a difference of only 3% according to the measure proposed by Kounadi and Leitner (2015). It also shows no difference between APA and DkData. DkData at least preserves the number of clusters with at least 30 points, which APA does not.

The spatial autocorrelation (SpatAut) allows for better differentiation between masking methods, although the results are quite similar. APA and Voronoi masking show a difference of over 10% compared to the spatial autocorrelation of the original

data set. ARP and MDAV show values just below 10% and 8%.

The arithmetic mean of the GDi and LDi shows that displacement using translation yields the best utility value of the masking methods tested. This is closely followed by location swapping methods and street masking. The anonymization of distance matrices via Lipschitz embedding is rather in the middle of the list of masking methods (sorted by utility values) since the descriptive statistics are not preserved well. However, this could be due to the approximated coordinates used.

Similarly, distance approximation using ISGP would be higher placed if the results of the descriptive statistics were not so different from the original. Again, this may be due to the fact that approximated coordinates had to be used, as well as the fact that this method is designed to mask distances to points of interest rather than points to each other. Rotation and change of scale preserve the utility the least, caused by the large displacement of the spatial mean center.

### 6.1.3. MSE

Another idea is to use the mean squared error as a measure of utility, as opposed to Kounadi and Leitner (2015) who do not use distances between points as an individual measure. The individual parts bias and variance can be defined as the squared difference between the original mean distance and the average of the masked mean distance of the coordinates (in meter). The variance is the variance of these distances since each masking method was replicated 50 times. Therefore, the first step was to calculate the average of the 50 replications and the variance of the mean distances. The MSE was then calculated by squaring the difference between that average and the original mean distance and adding the variance. Finally, to obtain one value for each masking method, the MSE was averaged for the different parameter choices.<sup>6</sup>

As opposed to Kounadi and Leitner (2015) the MSE is not limited by a maximum value. Therefore, it is challenging to evaluate whether a certain MSE is high or moderate. As can be seen in table 6.3, the difference between the MSE of the masking methods displacement using translation and change of scale is very large. However, it can also be seen that only a few methods show a very large MSE (e.g., rotation around the origin, distance approximation using Lipschitz Embedding, APA, ARP, and approximating distances using ISGP).

Moreover, considering only the mean distance between points shows different rankings compared to the GDi and LDi (if sorted by utility value) of the masking methods. Only displacement using translation is always placed as the masking method that preserves the utility the most. Similar rankings are seen for  $k$ -nearest neighbor donut masking (using the residential file or the data set as reference file), location swapping, APA, ARP, MDAV, distance approximation using ISGP, and change of scale.

---

<sup>6</sup>Results for individual parameter choices for each masking method can be seen in table I.3 in appendix I.

The random perturbation methods, donut masking using population density, and Voronoi masking are placed higher in the sorted list of utility values than for the average of the GDi and LDi. On the other hand, anonymization of distance matrices via Lipschitz embedding is placed near the bottom of the list, suggesting that the utility measured by MSE is much less preserved than shown by GDi and LDi. The difference in ranking remains even when using the median distance between coordinates.

Table 6.3.: Results of the calculation of the MSE (shown in four columns to reduce space; sorted by MSE).

method	MSE	method	MSE
orig	0		
DUT	22.77	DkData	35,720.65
RPN	304.84	VNE	44,950.36
LS	1,120.61	Grid	90,814.64
RPC	1,120.82	StreetMask	159,841.68
DD	1,679.43	MDAV	326,956.27
RPU	3,181.14	Rot	113,198,247.70
Dk	6,178.70	Lipschitz Embedding	134,156,982.26
Voronoi	8,284.46	APA	1,132,433,850.24
VNS	9,818.61	ARP	1,385,054,549.42
RotArb	10,856.97	ISGP	3,624,144,345.49
LSdonut	11,118.69	CS	4,252,206,780.73

## 6.2. Risk-Utility Map Using the GDi and LDi

The relationship between risk and utility is visualized in figure 6.1. Risk-utility maps using the MPR of the subsample  $n = 2,000$  and the full sample ( $n = 10,000$ ) are shown in appendix I.<sup>7</sup>

To obtain the map with an increasing utility on the right side of the graph, the average of the GDi and LDi was plotted as utility loss, i.e., the average of GDi and LDi subtracted from the average GDi and LDi of the unmasked data set. Most masking methods are located in the upper (right) corner (x-axis constrained to show differences between masking methods). Therefore, most masking methods preserve utility but not privacy. Rotation performs the worst (lowest utility, highest risk). Change of scale also proves unsuitable (high risk, much lower utility compared to others).

To see more clearly the difference between the masking methods, figure 6.1 also shows the risk-utility map without the rotation and change of scale methods. Again, the tendency can be seen that the risk value decreases when the utility decreases

<sup>7</sup>For random projection, the utility could not be evaluated with the given measures. Furthermore, the only applicable risk method is the graph matching attack on PPRL. Therefore, the masking method random projection will not be shown in the risk-utility maps.

(except for the excluded masking methods). The masking method with the best risk-utility relationship is the anonymization of distance matrices via Lipschitz embedding and the distance approximation using ISGP. These two show the desired low (close to zero) MPR. Again, it should be mentioned that for the GDi, approximated coordinates had to be used. Furthermore, distance approximation using ISGP originally intends to mask distances to points of interest (Schnell, Klingwort, et al., 2021).

### 6.3. Risk-Utility Map Using the Mean Squared Error

Figure 6.2 shows the result if the MSE is used as utility measure. To ensure that larger utility values are shown on the right side of the graph, the utility loss is shown, i.e., the MSE of the masking method is subtracted from the original MSE (0).

Again, change of scale, distance approximation using ISGP, APA, and ARP perform comparatively poorly in preserving utility, even when only distances are considered when measuring the utility. Change of scale, in this case, also shows a high risk of re-identification, while for APA, ARP the MPR is about 0.3 and for distance approximation using ISGP 0.03. Anonymization of distance matrices via Lipschitz embedding shows a high MSE but also the desired low MPR.

A closer look at the masking methods positioned in the upper right corner can be seen in the bottom map of figure 6.2. MDAV shows a much greater loss of utility than the other masking methods. Again, the risk-utility-trade-off can be seen. As the utility loss increases, the re-identification risk decreases.

### 6.4. Summary of Results of R-U Confidentiality Maps

All maps show that the majority of the masking methods are located in the upper right quadrant of a risk-utility map. Thus, although these methods preserve the utility, they also show a high risk of re-identification. The worst risk-utility relationship can be seen for change of scale, which shows the worst results for the GDi and LDi, and MSE while all coordinates could be re-identified correctly. Furthermore, both maps show that a decrease in the risk of re-identification also decreases utility. The masking method which is the closest to the bottom right corner of the map, i.e., high utility and low risk, is anonymization of distance matrices via Lipschitz embedding.

If using the MSE and the combination of GDi and LDi as utility measures are compared, some masking methods are evaluated differently. Using the GDi and LDi, Voronoi masking shows a lower utility than other masking methods, but not when using the MSE. For street masking, the utility of the data set seems much more preserved when using the GDi and LDi compared to the MSE. These results clearly show that the choice of the utility measure strongly influences the evaluation of the performance of a geomasking method compared to other geomasking methods.

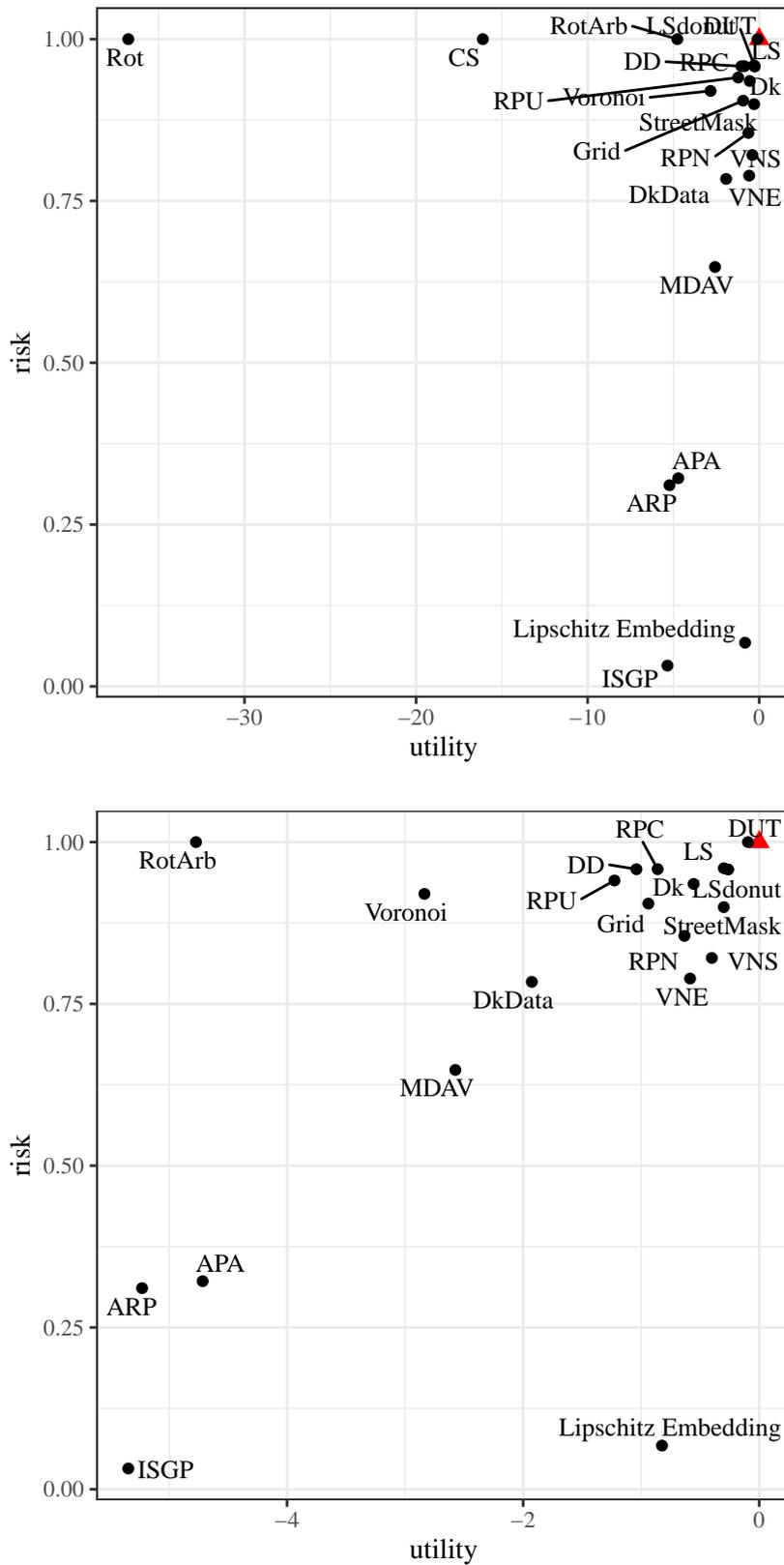


Figure 6.1.: Risk-utility map. Utility using mean of GD<sub>i</sub> and LD<sub>i</sub> (Kounadi and Leitner, 2015) shown as loss of utility compared to original data set (utility of zero). Risk is largest MPR. Red triangle shows original data. Bottom map shows risk-utility map without the outliers (rotation and change of scale).

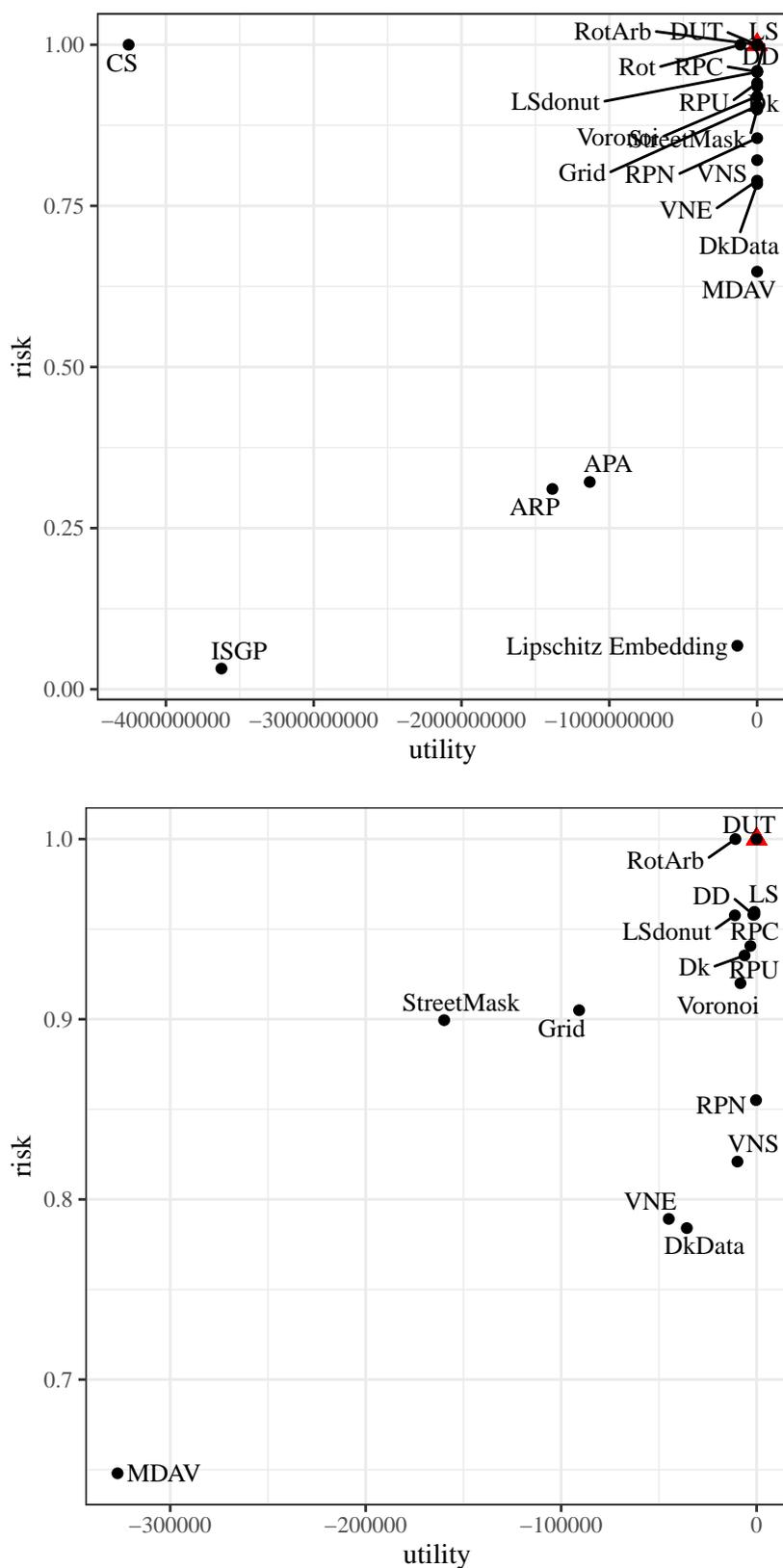


Figure 6.2.: Risk-utility map. Utility using only the mean distance between points, shown as loss of utility compared to original data set (utility of zero). Risk is largest MPR. Red triangle shows original data. Bottom map shows risk-utility map of methods with MPR > 0.6 excluding the method change of scale.



## 7. Discussion and Conclusion

Geographic information has proven useful in a variety of fields, including epidemiology, criminology, and social science (Rushton et al., 2006; Rushton et al., 2008; VanWey et al., 2005). Therefore, it seems reasonable to make such information available to other researchers. However, when geographic coordinates are released as-is, reverse geocoding can easily identify the location associated with the geographic coordinate (Zandbergen, 2014). Therefore, geographic masking methods have been proposed which displace the coordinates to hide the original location (see, e.g., Armstrong et al., 1999). Current practices in research to evaluate geomasking methods limit the number of tested masking methods without or with little explanation and lack an appropriate risk evaluation beyond  $k$ -anonymity and visual examination (see, e.g., Seidl, Jankowski, and Clarke, 2018; Broen et al., 2021). This thesis provides an overview of all currently existing geomasking methods, compares their utility using several dimensions, and makes a comprehensive risk assessment.

In the following, the key findings of this thesis are summarized.

### 7.1. Key Findings

#### Geographic Masking Methods

A complete review of all existing masking methods cannot be found in the geomasking literature at present. This is mainly due to the fact that many variants of each masking method exist and minor changes and a new name give the impression that many masking methods have been proposed. For example, displacements using translation moves coordinates up and down, left and right (Armstrong et al., 1999). A subsequent approach, where each coordinate is moved individually in a random direction, was random perturbation. Limiting the possible displacement distance was then called random perturbation within a circle (Armstrong et al., 1999). Adding a minimum displacement distance resulted in donut masking (Hampton et al., 2010). Considering only residential addresses within the area of the maximum displacement distance was named location swapping (Zhang, Freundsuh, et al., 2015). Further limiting the number of possible locations for displacement by using additional characteristics was introduced as verified neighbor approach (Richter, 2017).

Previous reviews (such as Armstrong et al., 1999; Gupta and Rao, 2020; Kounadi and Leitner, 2015) either placed almost every masking method in its own category or did not include all of the masking methods available. The exception is Gutmann et al. (2008)

whose three category idea was used here. The categories are “aggregation”, “adjusting coordinates”, and (renamed) “coordinate replacement”. The second category “adjusting coordinates” contains the most masking methods and can be further subdivided into:

1. methods that scale, rotate, displace or flip
2. methods that move points into a random direction and a random distance
3. methods that move points into a random direction and a random distance, and require additional information of the surrounding residents.

While most masking methods provide a detailed explanation of how to use the masking procedure in general, most masking methods lack a corresponding guide for the optimal choice of parameters for the given data set. Exceptions are anonymization of distance matrices via Lipschitz embedding and distance approximation using ISGP, as the authors propose a simulation study to find the optimal parameters for the provided data (Kroll and Schnell, 2016; Schnell, Klingwort, et al., 2021). Moreover, the majority of masking methods descriptions do not take into account possible problems in their replication. As an example, location swapping proposes defining the radius based on the population density (Zhang, Freundschuh, et al., 2015, p. 3). However, no solution is given for situations where no other residential address is found within the radius.

### **Risk-Utility Relationship**

All geomasking methods attempt to preserve the utility of the data while keeping the risk of re-identification low. In general, geomasking methods that preserve much of the utility are presumed to impose a high risk of re-identification. In contrast, geomasking methods that have a low risk of re-identification may not preserve much utility. Although this is well known, a comprehensive risk-utility-analysis of currently proposed geomasking methods cannot be found.

The focus of research on the risk-utility relationship of masking methods is primarily on utility and preserving as much information as possible from the unmasked data. But, the dimensions of utility are not fixed. Research agrees on descriptive statistics, preserving distances, spatial autocorrelation, and preserving clusters (see, e.g., Armstrong et al., 1999; Seidl, Paulus, et al., 2015), as used here. However, which measures are chosen within these dimensions and to what extent they are compared remains variable. For example, there are several clustering methods, sometimes hierarchical clustering (see, e.g., Seidl, Jankowski, and Clarke, 2018) and sometimes density-based clustering (see, e.g., Gao et al., 2019) are the preferred method. Even if the same clustering approach is taken, it may be of interest whether the number, the size or the location of clusters are preserved (see, e.g., different approaches of Kounadi and Leitner, 2015 and Seidl, Paulus, et al., 2015). Moreover, approaches to combine different utility dimensions in one value are sparse. Current approaches are the GDi and LDi (Kounadi and Leitner, 2015) or the RMSE (see, e.g., Clifton and Gehrke,

2013) or MSE. Not all utility measures (e.g. spatial median center) used in this thesis are necessary to calculate these combined values. However, they allow a better insight into which information is preserved.

As has been shown, the choice of utility measure can influence whether or not the masking method is considered to preserve the utility. For example, displacement using translation does not preserve the spatial mean center or spatial median center very well, but it does preserve the distance between points. Therefore, researches must be well aware that for masking methods that have shown to preserve utility well in other papers, this might just be due to the choice of the utility measure.

The risk component is most commonly evaluated in terms of  $k$ -anonymity (Sweeney, 2002; Samarati, 2001; Hampton et al., 2010; Broen et al., 2021) or whether the masked coordinate can be visually associated with the original location (see, e.g., Seidl, Jankowski, and Clarke, 2018). Neither of these is suitable because they make it seem that the risk of re-identification is much lower than proven in this thesis. The (worst-case) scenario, as commonly used in record linkage to evaluate whether methods are privacy-preserving, is the scenario used in this thesis. An intruder has the unmasked coordinates, and some additional information about attributes associated with the coordinates, as well as the knowledge about the applied masking method (see, e.g., Kroll, 2015). The problem of re-identifying locations can also be considered as a linkage problem. Two data sets are linked, but some information (coordinates in this case) does not match due to introduced noise. For this reason, in addition to some obvious attack methods such as the (overall) minimum distance, methods in the field of record linkage have been successfully used to assess the risk of re-identification in this thesis.

The summarized results of the previous chapter have shown that none of the masking methods that fall into the category “adjusting coordinates” displaces the coordinates far enough to avoid being re-identified. Most successful in re-identifying the unmasked coordinates is using the overall minimum distance (Hungarian algorithm) with the help of additional attributes and the graph theoretic linkage attack. As additional attributes, demographic characteristics can be used, which are usually available in (social science) data sets. Even when the sample size is increased, and more respondents share the same additional attributes, the risk of re-identification is still too high.

For the masking methods that fall into the category “aggregation” the results vary depending on the sample size considered and the acceptable loss of utility. As sample size increases, and thus the number of people with the same characteristics, precision and recall approach zero – the smaller the number of polygons or clusters, the higher precision and recall, due to larger displacement distances. The achieved value for the MPR for APA and ARP could only be reached if the polygons are used as additional information. However, APA and ARP show low utility with either measure (GD<sub>i</sub> and LD<sub>i</sub>; MSE) compared to other masking methods.

A major advantage in preserving the privacy of the masking methods that fall into

the category “coordinate replacement” is that many attack methods cannot be used if coordinates are not given as input. However, this also makes calculating utility difficult and sometimes impossible, i.e., descriptive statistics are not calculable from the distance matrix alone. Except for random projection, these methods allow the calculation of clusters and spatial autocorrelation. But additional information based on the coordinates cannot be added.

This thesis was able to show that affine transformations are not suitable masking methods. Furthermore, the thesis showed that the most promising risk methods are the Hungarian algorithm using additional information and the graph theoretic linkage attack. The graph matching attack on privacy-preserving record linkage was promising as it is able to handle various inputs (coordinates, distance matrix, concatenated string of zeros and ones). However, it fails for data sets with a small overlap and should be improved in this regard. In terms of utility measures, it could be seen that the standard deviational ellipses orientation shows little difference between the masking methods, except if the masking methods completely distort the underlying spatial structure of the data. It should also be noted that for rotations of  $180^\circ$ , the standard deviational ellipses orientation is preserved even if most of the coordinates have been displaced far. Also, spatial autocorrelation did not help differentiate between masking methods, which could be due to the fact that there were no better variables available for calculating spatial autocorrelation.

Before the implications of the results for the use of geographic masking methods are stated, the limitations of this work should be considered.

## 7.2. Limitations

The first limitation of this work is the lack of an appropriate test file. The most suitable file was the G-NAF, which contains every address in Australia. However, the G-NAF does not include the number of dwellings or the number of people living at each address. Therefore, each coordinate could only be considered as one person, even if multiple people live there. Moreover, the data set does not contain additional information at the person level, which had to be added based on known proportions of demographic characteristics in the population.

It was also assumed that the data set did not contain any missing or incorrect information. If information were missing or incorrect, this would make it harder to identify the original location. But it is in the interest of the data collectors to avoid missing or incorrect information. Also, the masking methods were only applied to one simple random sample of the data set, and different samples could yield different results. The influence of missing or incorrect information and different samples on the results should be tested in future research with real-world data.

Another limitation of this work is that not all of the proposed masking methods could be applied. This is due to the limitations of the masking methods themselves.

For example, the masking method based on the MGRS can only be used within a maximum area size of  $99,999 \times 99,999$  meters (Clarke, 2016). Another example is triangular displacement (Murad et al., 2014), which leaves too many questions unanswered. Another reason is the many variations that exist for each masking method, such as different distributions considered, different variables, different ways of forming polygons, e.g., in APA and ARP (Kounadi and Leitner, 2016).

Due to the necessary computational time (and computing power), only a selection of parameter choices per masking method could be applied. Each parameter choice for each masking method was replicated 50 times. Then each of the 50 replications had to be evaluated in terms of their risk and utility. The decisions of the parameter choices made were all based on common approaches in research or suggestions made by the inventors of the masking method. Similarly, parameter choices necessary for utility and risk measures were also based on common approaches found in the literature or, if available, author suggestions.

Furthermore, a choice had to be made on the utility and risk measures used. For utility, multiple dimensions were considered. The measures were chosen based on previously conducted analysis of geomasking methods. However, as has been shown, the choice of utility measure influences the utility evaluation of the geomasking methods. Thus, different utility measures might lead to different conclusions. For example, for clustering, different algorithms can be used. Moreover, the number of clusters, the size, or the density can be compared. In the geomasking literature, common approaches do not consider clusters individually but compare the clustering overall (see, e.g., Kounadi and Leitner, 2015). More meaningful would be to compare each cluster regarding their location, area size, the number of points, and if points remain in the same cluster. However, when comparing many masking methods, this is not feasible.

Lastly, the work is limited to single-point data. Point traces may also be of interest for geomasking methods, which requires that the relationship between the points forming tracks is preserved while preserving privacy (see, e.g., Scheider et al., 2020). Deep learning methods to re-identify masked point tracks can be found in Mol (2019).

### **7.3. Implications of Results for the Use of Geographic Masking Methods**

This work aimed to answer the following question: can we disclose the respondents' masked location to make the most use of the data without compromising their privacy?

With the currently existing masking methods and their intended replication that are summarized in the categories “adjusting coordinates”, and “aggregation”, the masked location cannot be disclosed, especially if information about the masking method and its parameter choices are released. The most promising attack methods

for an intruder are the Hungarian algorithm with blocking (considering additional information about respondents) and the graph theoretic linkage attack. In both, the additional information help reduce the number of potential matches. For the graph theoretic linkage attack, knowledge of the masking method also helps to set the required bounds for differences in distances without the need for a simulation study. Even with larger sample sizes, where the graph theoretic linkage attack requires more resources, the Hungarian algorithm using additional variables can still identify many locations. Furthermore, there are usually more variables available that can be used to make the attack more successful.

Geomasking methods that release the masked distance matrix instead of coordinates (anonymization of distance matrices via Lipschitz embedding, distance approximation using ISGP) do not impose a privacy risk (grouped in the category “coordinate replacement”). However, for these, descriptive statistics based on the distance matrix alone are not calculable. Also, researchers cannot add other information available in additional files based on the distance matrix. But, inferences based on the coordinates’ distance towards each other or towards points of interest can still be made (as in the spatial autocorrelation and the clustering algorithm).

Of the two, anonymization of distance matrices via Lipschitz embedding should be preferred when the full distance matrix is needed. Distance approximation using ISGP can be used when only distances to points of interest are masked since the masking method censors distances that exceed twice the considered radius. But it should be noted that anonymization of distance matrices via Lipschitz embedding preserves large distances not as well as small distances (Kroll and Schnell, 2016, p. 12).

Should geographic coordinates remain in the data set, an alternative is that the data set should only be used in a secure environment, such as a research data center (Kamel Boulos et al., 2006). Thus the data owner can control which data sets are linked to the data set of interest. Then simple masking methods such as  $k$ -nearest neighbor donut masking are sufficient. In addition, the risk of re-identification should always be assessed for data sets (and also their utility) prior to release to ensure a low re-identification risk.

## 7.4. Future Research

Future research should focus on tests with real-world data sets of different sizes (sample size, region size, overlap size) and consider missing or incorrect information. Then also masking methods that had to be excluded (see, e.g., masking based on the MGRS) can be tested.

In addition, an in-depth analysis is needed to determine whether some points are always or never identified and how these points differ, such as different population densities. Moreover, an even broader spectrum of parameter choices should be tested. Using simulation studies can help in finding the optimal parameter choice for the data

set. Future research should also focus on creating guidelines on how to find the optimal parameter choice for the masking methods for the given data set. Furthermore, it should be evaluated how much information about a masking method is needed for the risk methods to still achieve a high precision and recall.

The results of this thesis can be used to limit the number of masking methods tested and the risk measures to be applied. Moreover, the individual masking methods were compared in terms of additional information requirements, regional limits, etc. This comparison helps decide whether a masking method is suitable for the given data set. Finally, this thesis also provides the code for implementing each masking method in the open-source software *R* so that only minimal knowledge of *R* is required to use geographic masking methods.



## Bibliography

- Agrawal, Rakesh, Johannes Gehrke, Dimitrios Gunopulos, and Prabhakar Raghavan (1998). “Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications”. In: *SIGMOD '98 Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data*. New York: ACM, pp. 94–105.
- Ajayakumar, Jayakrishnan, Andrew J. Curtis, and Jacqueline Curtis (2019). “Addressing the Data Guardian and Geospatial Scientist Collaborator Dilemma: How to Share Health Records for Spatial Analysis While Maintaining Patient Confidentiality”. In: *International Journal of Health Geographics* 18 (30), pp. 1–12.
- Aldrich, T. E. and K. R. Krauthaim (1995). *Protecting Confidentiality in Small Area Studies. Paper presented at the CDC Symposium on “Statistical Methods: Small Area Statistics in Public Health”, Atlanta, GA*.
- Allshouse, William B., Molly K. Fitch, Kristen H. Hampton, Dionne C. Gesink, Irene A. Doherty, Peter A. Leone, Marc L. Serre, and William C. Miller (2010). “Geomasking Sensitive Health Data and Privacy Protection: An Evaluation Using an E911 Database”. In: *Geocarto International* 25 (6), pp. 443–452.
- Ambroise, Christophe and Gérard Govaert (1998). “Convergence of an EM-type Algorithm for Spatial Clustering”. In: *Pattern Recognition Letters* 19 (10), pp. 919–927.
- Ankerst, Mihael, Markus M. Breunig, Hans-Peter Kriegel, and Jörg Sander (1999). “OPTICS: Ordering Points To Identify the Clustering Structure”. In: *SIGMOD '99 Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data*. New York: ACM, pp. 49–60.
- Anselin, Luc (1995). “Local Indicators of Spatial Association – LISA”. In: *Geographical Analysis* 27 (2), pp. 93–115.
- Armstrong, Marc P., Gerard Rushton, and Dale L. Zimmerman (1999). “Geographically Masking Health Data to Preserve Confidentiality”. In: *Statistics in Medicine* 18 (5), pp. 497–525.
- Aurenhammer, Franz (1991). “Voronoi Diagrams: A Survey of a Fundamental Geometric Data Structure”. In: *ACM Computing Surveys* 23 (3), pp. 345–405.
- Australian Bureau of Statistics (2016a). *Australian Statistical Geography Standard (ASGS) Volume 1: Main Structure and Greater Capital City Statistical Areas: South Australia Mesh Blocks ASGS Edition 2016 in .csv Format*. Retrieved: 25. October 2019. URL: <https://www.abs.gov.au/AUSSTATS/abs@.nsf/Lookup/1270.0.55.001Explanatory%20Notes1July%202016?OpenDocument>.

- Australian Bureau of Statistics (2016b). *Census DataPacks: 2016 General Community Profile: Statistical Area 1 / Postal Areas*. Retrieved: 25. October 2019. URL: <https://datapacks.censusdata.abs.gov.au/datapacks/>.
- Australian Bureau of Statistics (2017). *2016 Census Community Profiles*. Retrieved: 25. October 2019. URL: [https://quickstats.censusdata.abs.gov.au/census\\_services/getproduct/census/2016/communityprofile/SSC40621](https://quickstats.censusdata.abs.gov.au/census_services/getproduct/census/2016/communityprofile/SSC40621).
- Bertici, R., M. Herbei, Silvică Onciă, and Laura Smuleac (2014). “Comparative Analysis of Mercator and U.T.M. Map Projections”. In: *Research Journal of Agricultural Science* 46 (2), pp. 14–24.
- Biemer, Paul P. (2010). “Total Survey Error: Design, Implementation, and Evaluation”. In: *Public Opinion Quarterly* 74 (5), pp. 817–848.
- Boeing, Geoff (2017). “OSMnx: New Methods for Acquiring, Constructing, Analyzing, and Visualizing Complex Street Networks”. In: *Computers, Environment and Urban Systems* 65, pp. 126–139.
- Borgs, Christian (2019). “Optimal Parameter Choice for Bloom Filter-Based Privacy-Preserving Record Linkage”. PhD thesis. University of Duisburg-Essen.
- Bourgain, J. (1985). “On Lipschitz Embedding of Finite Metric Spaces in Hilbert Space”. In: *Israel Journal of Mathematics* 52 (1–2), pp. 46–52.
- Bridwell, Scott A. (2007). “The Dimensions of Locational Privacy”. In: *Societies and Cities in the Age of Instant Access*. Ed. by Harvey J. Miller. Dordrecht: Springer, pp. 209–225.
- Broen, Kelly, Rob Trangucci, and Jon Zelner (2021). “Measuring the Impact of Spatial Perturbations on the Relationship Between Data Privacy and Validity of Descriptive Statistics”. In: *International Journal of Health Geographics* 20 (3), pp. 1–16.
- Bui, Randy, Ron N. Buliung, and Tarmo K. Remmel (2012). *Package “aspace”: A Collection of Functions for Estimating Centographic Statistics and Computational Geometries for Spatial Point Patterns: Version 3.2*. Retrieved: 11. December 2019. URL: <https://mran.microsoft.com/snapshot/2017-05-16/web/packages/aspace/aspace.pdf>.
- Bundesamt für Kartographie und Geodäsie (2020). *Geographical Grids for Germany: GeoGitter*. Retrieved: 17.05.2021. URL: <https://gdz.bkg.bund.de/index.php/default/geographische-gitter-fur-deutschland-in-lambert-projektion-geogitter-inspire.html>.
- Burkard, Rainer, Mauro Dell’Amico, and Silvano Martello (2012). *Assignment Problems*. Philadelphia: Society for Industrial and Applied Mathematics.
- Cassa, Christopher A., Shaun J. Grannis, J. Marc Overhage, and Kenneth D. Mandl (2006). “A Context-Sensitive Approach to Anonymizing Spatial Surveillance Data: Impact on Outbreak Detection”. In: *Journal of the American Medical Informatics Association* 13 (2), pp. 160–165.

- Cassa, Christopher A., Shannon C. Wieland, and Kenneth D. Mandl (2008). “Re-Identification of Home Addresses from Spatial Locations Anonymized by Gaussian Skew”. In: *International Journal of Health Geographics* 7 (45), pp. 1–9.
- Charleux, Laure and Katherine Schofield (2020). “True Spatial k-anonymity: Adaptive Areal Elimination vs. Adaptive Areal Masking”. In: *Cartography and Geographic Information Science* 47 (6), pp. 637–649.
- Chauhan, Ritu, Harleen Kaur, and M. Afshar Alam (2010). “Data Clustering Method for Discovering Clusters in Spatial Cancer Databases”. In: *International Journal of Computer Applications* 10 (6), pp. 9–14.
- Chen, Chien-Chou, Jen-Hsiang Chuang, Da-Wei Wang, Chien-Min Wang, Bo-Cheng Lin, and Ta-Chien Chan (2017). “Balancing Geo-Privacy and Spatial Patterns in Epidemiological Studies”. In: *Geospatial Health* 12 (2), pp. 294–209.
- Chitra, K. and D. Maheswari (2017). “A Comparative Study of Various Clustering Algorithms in Data Mining”. In: *International Journal of Computer Science and Mobile Computing* 6 (8), pp. 109–115.
- Christen, Peter (2012). “A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication”. In: *IEEE Transactions on Knowledge and Data Engineering* 24 (9), pp. 1537–1555.
- Clark, Philip J. and Francis C. Evans (1954). “Distance to Nearest Neighbor as a Measure of Spatial Relationships in Populations”. In: *Ecology* 35 (4), pp. 445–453.
- Clarke, Keith C. (2016). “A Multiscale Masking Method for Point Geographic Data”. In: *International Journal of Geographical Information Science* 30 (2), pp. 300–315.
- Cliff, Andrew D. and J. Keith Ord (1981). *Spatial Processes: Models & Applications*. London: Pion Limited.
- Clifton, Kelly J. and Steven R. Gehrke (2013). “Application of Geographic Perturbation Methods to Residential Locations in the Oregon Household Activity Survey”. In: *Transportation Research Record: Journal of the Transportation Research Board* 2354 (1), pp. 40–50.
- Cressie, Noel (1992). “Statistics For Spatial Data”. In: *Terra Nova* 4 (5), pp. 613–614.
- Croft, William Lee, Wei Shi, Jörg-Rüdiger Sack, and Jean-Pierre Corriveau (2015). “A Novel Geographic Partitioning System for Anonymizing Health Care Data”. In: *arXiv:1505.06939*, pp. 1–26.
- Croft, William Lee, Wei Shi, Jörg-Rüdiger Sack, and Jean-Pierre Corriveau (2016). “Location-Based Anonymization: Comparison and Evaluation of the Voronoi-based Aggregation System”. In: *International Journal of Geographical Information Science* 30 (11), pp. 2253–2275.
- Croft, William Lee, Wei Shi, Jörg-Rüdiger Sack, and Jean-Pierre Corriveau (2017). “Comparison of Approaches of Geographic Partitioning for Data Anonymization”. In: *Journal of Geographical Systems* 19, pp. 221–248.

- Dempster, Arthur P., Nan M. Laird, and Donald B. Rubin (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm". In: *Journal of the Royal Statistical Society: Series B* 39 (1), pp. 1–38.
- Department of Industry, Innovation and Science (2019a). *G-NAF: Data Product Description: August 2019*. URL: <https://data.gov.au/data/dataset/geoscape-geocoded-national-address-file-g-naf-previous-versions>.
- Department of Industry, Innovation and Science (2019b). *PSMA Geocoded National Address File (G-NAF) (August 2019 Release)*. Retrieved: 04. September 2019. URL: <https://data.gov.au/data/dataset/geoscape-geocoded-national-address-file-g-naf-previous-versions>.
- Dice, Lee R. (1945). "Measures of the Amount of Ecologic Association Between Species". In: *Ecology* 26 (3), pp. 297–302.
- Domingo-Ferrer, Josep and Vicenç Torra (2005). "Ordinal, Continuous and Heterogeneous  $k$ -Anonymity Through Microaggregation". In: *Data Mining and Knowledge Discovery* 11, pp. 195–212.
- Dubé, Jean and Diégo Legros (2014). *Spatial Econometrics Using Microdata*. London: ISTE Ltd.
- Dueker, Kenneth J. (1974). "Urban Geocoding". In: *Annals of the Association of American Geographers* 64 (2), pp. 318–325.
- Duncan, George T., Mark Elliot, and Juan-José Salazar-González (2011). *Statistical Confidentiality: Principles and Practice*. New York: Springer.
- Duncan, George T. and Stephen E. Fienberg (1999). "Obtaining Information While Preserving Privacy: A Markov Perturbation Method for Tabular Data". In: *Statistical Data Protection: Proceeding of the Conference: Lisbon, 25 to 27 March 1998*. Luxembourg: European Commission, Statistical Office of the European Communities, pp. 351–362.
- Duncan, George T., Stephen E. Fienberg, Ramayya Krishnan, Rema Padman, and Stephen F. Roehrig (2001). "Disclosure Limitation Methods and Information Loss for Tabular Data". In: *Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies*. Ed. by Pat Doyle, Julia I. Lane, and Jules J. M. Theeuwes. Amsterdam: Elsevier, pp. 135–166.
- Duncan, George T., Sallie A. Keller-McNulty, and S. Lynne Stokes (2001). *Disclosure Risk vs. Data Utility: The R-U Confidentiality Map*. Tech. rep. 121. National Institute of Statistical Science.
- Ebdon, David (1977). *Statistics in Geography: A Practical Approach*. Oxford: Basil Blackwell.
- Ebdon, David (1987). *Statistics in Geography*. 2nd ed. Oxford: Basil Blackwell.
- Ebdon, David (1990). *Statistics in Geography*. 2nd ed. Oxford: Basil Blackwell.
- El Emam, Khaled, Ann Brown, and Philip AbdelMalik (2009). "Evaluating Predictors of Geographic Area Population Size Cut-offs to Manage Re-Identification Risk". In: *Journal of the American Medical Informatics Association* 16 (2), pp. 256–266.

- Elliot, Mark and Angela Dale (1999). “Scenarios of Attack: the Data Intruder’s Perspective on Statistical Disclosure Risk”. In: *Netherlands Official Statistics: Special Issue: Statistical Disclosure Control* 14 (Spring 1999), pp. 6–10.
- Elliot, Mark, Elaine Mackey, Kieron O’Hara, and Caroline Tudor (2016). *The Anonymisation Decision-Making Framework*. Manchester: UKAN.
- Ester, Martin, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu (1996). “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise”. In: *KDD’96 Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. Menlo Park: AAAI Press, pp. 226–231.
- Fahrmeir, Ludwig, Christian Heumann, Rita Künstler, Iris Pigeot, and Gerhard Tutz (2016). *Statistik: Der Weg zur Datenanalyse*. 8th ed. Berlin: Springer.
- Farrow, James (2014). *Privacy Preserving Distance-Comparable Geohashing*. International Health Data Linkage Conference 2014, 28–30 April 2014, Vancouver, Canada.
- Farrow, James Matthew (2015). “Method and System for Comparative Data Analysis”. Patent Number PCT/AU2015/000251; International Publication Number WO 2015/164910 A1.
- Fawcett, Tom (2003). *ROC Graphs: Notes and Practical Considerations for Data Mining Researchers*. Tech. rep. HPL-2003-4. Intelligent Enterprise Technologies Laboratory: HP Laboratories Palo Alto.
- Flacke, Werner, Birgit Thomsen, Uta Griwodz, and Mareike Dietrich (2015). *Koordinatensysteme in ArcGIS: Praxis der Transformationen und Projektionen*. 3rd ed. Berlin: Wichmann.
- Fronterré, Claudio (2018). “Spatial Analysis of Geomasked and Aggregated Data”. PhD thesis. Università Degli Stude di Padova. Dipartimento di Scienza Statistiche.
- Frost, W. H. (1936). *Snow on Cholera: Being a Reprint of Two Papers by John Snow, M.D. Together with a Biographical Memoir by B.W. Richardson and an Introduction by Wade Hampton Frost, M.D.* New York: The Commonwealth Fund.
- Furfey, Paul Hanly (1927). “A Note on Lefever’s ’Standard Deviational Ellipse’”. In: *American Journal of Sociology* 33 (1), pp. 94–98.
- Gao, Song, Jinneng Rao, Xinyi Liu, Yuhao Kang, Qunying Huang, and Joseph App (2019). “Exploring the Effectiveness of Geomasking Techniques for Protecting the Geoprivacy of Twitter Users”. In: *Journal of Spatial Information Science* 19, pp. 105–129.
- Gebers, Kathrin and Philip Graze (2019). “Statistische Datengewinnung durch die Nutzung geografischer Informationen”. In: *WISTA* 4, pp. 11–18.
- Ghinita, Gabriel, Keliang Zhao, Dimitris Papadias, and Panos Kalnis (2010). “A Reciprocal Framework for Spatial  $K$ -Anonymity”. In: *Information Systems* 35 (3), pp. 299–314.
- Goldberg, Daniel W., John P. Wilson, and Craig A. Knoblock (2007). “From Text to Geographic Coordinates: The Current State of Geocoding”. In: *Journal of the Urban and Regional Information Systems Association* 19 (1), pp. 33–46.

- Goodchild, Michael F. (2004). “The Validity and Usefulness of Laws in Geographic Information Science and Geography”. In: *Annals of the Association of American Geographers* 94 (2), pp. 300–303.
- Griffith, Daniel A. (2003). *Spatial Autocorrelation and Spatial Filtering: Gaining Understanding Through Theory and Scientific Visualization*. Berlin: Springer.
- Gupta, Ruchika and Udai Pratap Rao (2020). “Preserving Location Privacy Using Three Layer RDV Masking in Geocoded Published Discrete Point Data”. In: *World Wide Web* 23, pp. 175–206.
- Gutmann, Myron P., Kristine Witkowski, Corey Colyer, JoAnne McFarland O’Rourke, and James McNally (2008). “Providing Spatial Data for Secondary Analysis: Issues and Current Practices Relating to Confidentiality”. In: *Population Research and Policy Review* 27 (6), pp. 639–665.
- Hager, John W., Larry L. Fry, Sandra S. Jacks, and David R. Hill (1992). *Datums, Ellipsoids, Grids, and Grid Reference Systems*. Tech. rep. ADA247651. Defence Mapping Agency. URL: <https://apps.dtic.mil/sti/citations/ADA247651>.
- Haggett, Peter, Andrew D. Cliff, and Allan Frey (1977). *Locational Analysis in Human Geography*. 2nd ed. London: Edward Arnold.
- Hamming, R. W. (1950). “Error Detecting and Error Correcting Codes”. In: *The Bell System Technical Journal* 29 (2), pp. 147–160.
- Hampton, Kristen H., Molly K. Fitch, William B. Allshouse, Irene A. Doherty, Dionne C. Gesink, Peter A. Leone, Marc L. Serre, and William C. Miller (2010). “Mapping Health Data: Improved Privacy Protection with Donut Method Geomasking”. In: *American Journal of Epidemiology* 172 (9), pp. 1062–1069.
- Han, Jiawei, Micheline Kamber, and Jian Pei (2012). *Data Mining: Concepts and Techniques*. 3rd ed. Waltham: Elsevier, Morgan Kaufmann.
- Han, Jiawei, Jae-Gil Lee, and Micheline Kamber (2009). “An Overview of Clustering Methods in Geographic Data Analysis”. In: *Geographic Data Mining and Knowledge Discovery*. Ed. by Harvey J. Miller and Jiawei Han. 2nd ed. Boca Raton: CRC, pp. 149–187.
- Hand, David and Peter Christen (2018). “A Note on Using the F-Measure for Evaluating Record Linkage Algorithms”. In: *Statistics and Computing* 28, pp. 539–547.
- Hardwick, Ian (1996). *Decision and Discrete Mathematics: Maths for Decision-Making in Business and Industry*. Sawston: Woodhead Publishing.
- Haspel, Moshe and H. Gibbs Knotts (2005). “Location, Location, Location: Precinct Placement and the Costs of Voting”. In: *The Journal of Politics* 67 (2), pp. 560–573.
- Heimann, Mark, Wei Lee, Shengjie Pan, Kuan-Yu Chen, and Danai Koutra (2018). “HashAlign: Hash-Based Alignment of Multiple Graphs”. In: *Advances in Knowledge Discovery and Data Mining: 22nd Pacific-Asia Conference, PAKDD 2018, Melbourne, VIC, Australia, June 3-6, 2018, Proceedings, Part III*. Ed. by Dinh Phung,

- Vincent S. Tseng, Geoffrey I. Webb, Bao Ho, Mohadeseh Ganji, and Lida Rashidi. Springer, pp. 726–739.
- Henecka, Wilko (2019). *Distance Aware Address Encoding for Privacy-Preserving Record Linkage*. Retrieved: 09. January 2020. URL: <https://medium.com/@wilko.henecka/distance-aware-address-encoding-for-privacy-preserving-record-linkage-a6cecdadc22>.
- Hinneburg, Alexander and Daniel A. Keim (1998). “An Efficient Approach to Clustering in Large Multimedia Databases with Noise”. In: *KDD’98 Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*. Menlo Park: AAAI Press, pp. 58–65.
- Houfah-Khoufah, Walid and Guillaume Touya (2020). *Geographically Masking Addresses to Study COVID-19 Clusters*. Retrieved: 29. January 2021. Preprint. URL: <https://doi.org/10.21203/rs.3.rs-128679/v1>.
- Huckett, Jennifer C. (2008). “Synthetic Data Methods for Disclosure Limitation”. PhD thesis. Iowa State University.
- infas 360 (2020). *Wo sind die Neuinfektionen in der Stadt? Ein Standard bietet sofort Hilfe*. Retrieved: 30. April 2020. URL: <https://blog.infas360.de/2020/04/29/georaster-hilft-bei-neuinfektionen/>.
- Kamel Boulos, Maged N. (2004). “Towards Evidence-based, GIS-driven National Spatial Health Information Infrastructure and Surveillance in the United Kingdom”. In: *International Journal of Health Geographics* 3 (1).
- Kamel Boulos, Maged N., Qiang Cai, Julian A. Padget, and Gerard Rushton (2006). “Using Software Agents to Preserve Individual Health Data Confidentiality in Micro-Scale Geographical Analyses”. In: *Journal of Biomedical Informatics* 39 (2), pp. 160–170.
- Karypis, George, Eui-Hong Han, and Vipin Kumar (1999). “Chameleon: Hierarchical Clustering Using Dynamic Modeling”. In: *Computer* 32 (8), pp. 68–75.
- Kaufman, Leonard and Peter J. Rousseeuw (2005). *Finding Groups in Data: An Introduction to Cluster Analysis*. Hoboken: Wiley.
- Kerckhoffs, Auguste (1883). “La Cryptographie Militaire”. In: *Journal des Sciences Militaires* IX (January, February), pp. 5–38, 161–191.
- Kim, Junghwan, Mei-Po Kwan, Margaret C. Levenstein, and Douglas B. Richardson (2020). “How Do People Perceive the Disclosure Risk of Maps? Examining the Perceived Disclosure Risk of Maps and Its Implications for Geoprivacy Protection”. In: *Cartography and Geographic Information Science* 48 (1), pp. 2–20.
- Klingwort, Jonas, Rainer Schnell, and Michaela Sixt (2020). *Geo-Masking von Koordinaten der BiLO Befragten für zukünftige datenschutzgerechte Distanzberechnungen*. LIfBi Working Paper No. 87. Bamberg: Leibniz-Institut für Bildungsverläufe.
- Konc, Janez and Dušana Janežič (2007). “An Improved Branch and Bound Algorithm for the Maximum Clique Problem”. In: *MATCH: Communications in Mathematical and in Computer Chemistry* 58, pp. 569–590.

- Kopec, Richard J. (1963). "An Alternative Method for the Construction of Thiessen Polygons". In: *The Professional Geographer* 15 (5), pp. 24–26.
- Kounadi, Ourania and Michael Leitner (2015). "Spatial Information Divergence: Using Global and Local Indices to Compare Geographical Masks Applied to Crime Data". In: *Transactions in GIS* 19 (5), pp. 737–757.
- Kounadi, Ourania and Michael Leitner (2016). "Adaptive Areal Elimination (AAE): A Transparent Way of Disclosing Protected Spatial Datasets". In: *Computers, Environment and Urban Systems* 57, pp. 59–67.
- Krige, D. G. (1951). "A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand". In: *Journal of the Chemical Metallurgical & Mining Society of South Africa* 52 (6), pp. 119–139.
- Kroll, Martin (2014a). "A Graph Theoretic Linkage Attack on Microdata in a Metric Space". In: *arXiv:1402.3198v1*, pp. 1–24.
- Kroll, Martin (2014b). *A Graph Theoretic Linkage Attack on Microdata in a Metric Space*. Working Paper WP-GRLC-2014-01. German Record Linkage Center.
- Kroll, Martin (2015). "A Graph Theoretic Linkage Attack on Microdata in a Metric Space". In: *Transactions on Data Privacy* 8 (3), pp. 217–243.
- Kroll, Martin and Rainer Schnell (2016). "Anonymisation of Geographical Distance Matrices via Lipschitz Embedding". In: *International Journal of Health Geographics* 15 (1), pp. 1–14.
- Kuhn, H. W. (1955). "The Hungarian Method for the Assignment Problem". In: *Naval Research Logistics Quarterly* 2 (1-2), pp. 83–97.
- Kwan, Mei-Po, Irene Casas, and Ben C. Schmitz (2004). "Protection of Geoprivacy and Accuracy of Spatial Information: How Effective Are Geographical Masks?" In: *Cartographica* 39 (2), pp. 15–28.
- Lach Arlinghaus, Sandra and Joseph J. Kerski (2014). *Spatial Mathematics: Theory and Practice through Mapping*. Boca Raton: CRC Press.
- Lawhead, Joel (2015). *Learning Geospatial Analysis with Python: An Effective Guide to Geographic Information System and Remote Sensing Analysis using Python 3*. 2nd ed. Birmingham: Packt Publishing.
- Lee, Der-Tsai and Arthur K. Lin (1986). "Generalized Delaunay Triangulation for Planar Graphs". In: *Discrete & Computational Geometry* 1 (3), pp. 201–217.
- Lefever, D. Welty (1926). "Measuring Geographic Concentration by Means of the Standard Deviation Ellipse". In: *American Journal of Sociology* 32 (1), pp. 88–94.
- Leitner, Michael and Andrew Curtis (2004). "Cartographic Guidelines for Geographically Masking the Locations of Confidential Point Data". In: *Cartographic Perspectives* 49, pp. 22–39.
- Leitner, Michael and Andrew Curtis (2006). "A First Step Towards A Framework for Presenting the Location of Confidential Point Data on Maps: Results of an Empirical Perceptual Study". In: *International Journal of Geographical Information Science* 20 (7), pp. 813–822.

- Leskovec, Jure, Anand Rajaraman, and Jeffrey David Ullman (2014). *Mining of Massive Datasets*. 2nd ed. Cambridge: Cambridge University Press.
- Lloyd, Stuart P. (1982). "Least Squares Quantization in PCM". In: *IEEE Transactions on Information Theory* 28 (2), pp. 129–137.
- Loenen, Bastiaan van, Stefan Kulk, and Hendrik Ploeger (2016). "Data Protection Legislation: A Very Hungry Caterpillar: The Case of Mapping Data in the European Union". In: *Government Information Quarterly* 33 (2), pp. 338–345.
- Lu, Yongmei, Charles Yorke, and F. Benjamin Zhan (2012). "Considering Risk Locations When Defining Perturbation Zones for Geomasking". In: *Cartographica: The International Journal for Geographic Information and Geovisualization* 47 (3), pp. 168–178.
- MacQueen, J. (1967). "Some Methods for Classification and Analysis of Multivariate Observations". In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*. Ed. by Lucien M. Le Cam and Jerzy Neyman. Berkeley: University of California Press, pp. 281–297.
- Mardia, Kanti V., John T. Kent, and John M. Bibby (1995). *Multivariate Analysis*. London: Academic Press.
- McLeod, Karis S. (2000). "Our Sense of Snow: the Myth of John Snow in Medical Geography". In: *Societal Science & Medicine* 50 (7–8), pp. 923–935.
- Mol, Maarten (2019). "Investigation of Attack Strategies on Geoprivacy with Spatial Obfuscation". MA thesis. Utrecht University.
- Moran, Patrick A. P. (1950). "Notes on Continuous Stochastic Phenomena". In: *Biometrika* 37 (1/2), pp. 17–23.
- Murad, Abdullah, Brian Hilton, Thomas Horan, and John Tangenberg (2014). "Protecting Patient Geo-Privacy Via a Triangular Displacement Geo-Masking Method". In: *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Privacy in Geographic Information Collection and Analysis*. New York: ACM.
- National Research Council (2007). *Putting People on the Map: Protecting Confidentiality with Linked Social-Spatial Data*. Washington, D.C.: The National Academies Press.
- Ng, Raymond T. and Jiawei Han (2002). "CLARANS: A Method for Clustering Objects for Spatial Data Mining". In: *IEEE Transactions on Knowledge & Data Engineering* 14, pp. 1003–1016.
- Olligschlaeger, Andreas M. (1997). "Artificial Neural Networks and Crime Mapping". In: *Crime Prevention Studies*. Ed. by David Weisburd and Tom McEwen. Vol. 8. Monsey: Willow Tree Press, pp. 313–347.
- Openshaw, Stan (1983). *The Modifiable Areal Unit Problem*. Norwich: Geo Books.
- Openshaw, Stan and Peter J. Taylor (1981). "The Modifiable Areal Unit Problem". In: *Quantitative Geography: A British View*. Ed. by N. Wrigley and R. J. Bennett. London: Routledge & Kegan Paul, pp. 60–69.

- Oyana, Tonny J. and Florence M. Margai (2016). *Spatial Analysis: Statistics, Visualization, and Computational Methods*. Boca Raton: CRC Press.
- Page, Matthew J., Joanne E. McKenzie, Patrick M. Bossuyt, Isabelle Boutron, Tammy C. Hoffmann, Cynthia D. Mulrow, Larissa Shamseer, Jennifer M. Tetzlaff, Elie A. Aki, Sue E. Brennan, Roger Chou, Julie Glanville, Jeremy M. Grimshaw, Asbjørn Hróbjartsson, Manoj M. Lalu, Tianjing Li, Elizabeth W. Loder, Evan Mayo-Wilson, Steve McDonald, Luke A. McGuinness, Lesley A. Stewart, James Thomas, Andrea C. Tricco, Vivian A. Welch, Penny Whiting, and David Moher (2021). “The PRISMA 2020 Statement: An Updated Guideline for Reporting Systematic Reviews”. In: *Systematic Reviews* 10 (89), pp. 1–11.
- Panigrahi, Narayan (2014). *Computing in Geographic Information Systems*. Boca Raton: CRC Press.
- Paradis, Emmanuel, Simon Blomberg, Ben Bolker, Joseph Brown, Julien Claude, Hoa Sien Cuong, Richard Desper, Gilles Didier, Benoit Durand, Julien Dutheil, R. J. Ewing, Olivier Gascuel, Thomas Guillerme, Christoph Heibl, Anthony Ives, Bradley Jones, Franz Krahe, Daniel Lawson, Vincent Lefort, Pierre Legendre, Jim Lemon, Eric Marcon, Rosemary McCloskey, Johan Nylander, Rainer Opgen-Rhein, Andrei-Alin Popescu, Manuela Royer-Carenzi, Klaus Schliep, Korbinian Strimmer, and Damien de Vienne (2019). *Package ape: Analyses of Phylogenetics and Evolution: Version 5.3*. Retrieved: 17. March 2020. URL: <https://cran.r-project.org/web/packages/ape/index.html>.
- Petitcolas, Fabien A. P. (2011). “Kerckhoffs’ Principle”. In: *Encyclopedia of Cryptography and Security*. Ed. by Henk C. A. van Tilborg and Sushil Jajodia. 2nd ed. New York: Springer, p. 675.
- Pfeiffer, Dirk U., Timothy P. Robinson, Mark Stevenson, Kim B. Stevens, David J. Rogers, and Archie C. A. Clements (2008). *Spatial Analysis in Epidemiology*. Oxford: Oxford University Press.
- Pinder, David A. and M. E. Witherick (1972). “The Principles, Practice and Pitfalls of Nearest-neighbour Analysis”. In: *Journal of the Geographical Association* 57 (4), pp. 277–288.
- Pinder, David, Izumi Shimada, and David Gregory (1979). “The Nearest-Neighbor Statistic: Archaeological Application and New Developments”. In: *American Antiquity* 44 (3), pp. 430–445.
- Police.uk (n.d.). *Privacy and Anonymisation*. Retrieved: 19. July 2021. URL: <https://data.police.uk/about/#anonymisation>.
- Rao, Jimeng, Song Gao, Yuhao Kang, and Qunying Huang (2020). “LSTM-TrajGAN: A Deep Learning Approach to Trajectory Privacy Protection”. In: *11th International Conference on Geographic Information Science, GIScience 2021, September 27–30, 2021, Poznań, Poland - Part I*. Ed. by Krzysztof Janowicz and Judith Anne Verstegen. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 12:1–12:17.

- Rao, Jinneng, Song Gao, Mingxiao Li, and Qunying Huang (2021). “A Privacy-Preserving Framework for Location Recommendation Using Decentralized Collaborative Machine Learning”. In: *Transactions in GIS* 25 (3), pp. 1153–1175.
- RatSWD (2011). *Endbericht der AG “Georeferenzierung von Daten” des RatSWD: Bericht der Arbeitsgruppe und Empfehlung des Rates für Sozial- und Wirtschaftsdaten* (RatSWD. Retrieved: 30. January 2021. URL: [http://ratswd.de/Geodaten/downloads/RatSWD\\_Endbericht\\_Geo-AG.pdf](http://ratswd.de/Geodaten/downloads/RatSWD_Endbericht_Geo-AG.pdf)).
- Richter, Wayne (2017). “The Verified Neighbor Approach to Geoprivacy: An Improved Method for Geographic Masking”. In: *Journal of Exposure Science and Environmental Epidemiology* 28 (2), pp. 109–118.
- Rothe, Patrick (2015). “Statistische Geheimhaltung – Der Schutz vertraulicher Daten in der amtlichen Statistik: Teil 1: Rechtliche und methodische Grundlagen”. In: *Bayern in Zahlen* 5, pp. 294–303.
- Rushton, Gerard, Marc P. Armstrong, Josephine Gittler, Barry R. Greene, Claire E. Pavlik, Michele M. West, and Dale L. Zimmerman (2006). “Geocoding in Cancer Research: A Review”. In: *American Journal of Preventive Medicine* 30 (2S), S16–S24.
- Rushton, Gerard, Marc P. Armstrong, Josephine Gittler, Barry R. Greene, Claire E. Pavlik, Michele M. West, and Dale L. Zimmerman (2008). “Introduction”. In: *Geocoding Health Data: The Use of Geographic Codes in Cancer Prevention and Control, Research, and Practice*. Ed. by Gerard Rushton, Marc P. Armstrong, Josephine Gittler, Barry R. Greene, Claire E. Pavlik, Michele M. West, and Dale L. Zimmerman. Boca Raton: CRC Press, pp. 1–10.
- Saalfeld, Alan, Laura Zayatz, and Erik Hoel (1992). “Contextual Variables via Geographic Sorting: A Moving Averages Approach”. In: *Proceedings of the Survey Research Methods Section*. American Statistical Association, pp. 691–696.
- Samarati, Pierangela (2001). “Protecting Respondents Identities in Microdata Release”. In: *IEEE Transactions on Knowledge and Data Engineering (TKDE)* 13 (6), pp. 1010–1027.
- Scheider, Simon, Jiong Wang, Maarten Mol, Oliver Schmitz, and Derek Karssenber (2020). “Obfuscating Spatial Point Tracks with Simulated Crowding”. In: *International Journal of Geographical Information Science* 34 (7), pp. 1398–1427.
- Schnell, Rainer (1991). “Wer ist das Volk? Zur faktischen Grundgesamtheit bei ‘allgemeinen Bevölkerungsumfragen’: Undercoverage, Schwererreichbare und Nichtbefragbare”. In: *Kölner Zeitschrift für Soziologie und Sozialpsychologie* 43 (1), pp. 106–154.
- Schnell, Rainer (1994). *Graphisch gestützte Datenanalyse*. München: Oldenbourg.
- Schnell, Rainer, Jonas Klingwort, and James Matthew Farrow (2021). “Locational Privacy-Preserving Distance Computations with Intersecting Sets of Randomly Labeled Grid Points”. In: *International Journal of Health Geographics* 20 (14), pp. 1–16.

- Schnell, Rainer and Sarah Redlich (2019). *Statistische Methoden des Datenschutzes für georeferenzierte Daten der empirischen Sozialforschung*. Leibniz-Institut für Bildungsverläufe e.V (LIfBi), 06.11.2019, Bamberg, Germany.
- Schoier, Gabriella and Caterina Gregorio (2017). “Clustering Algorithms for Spatial Big Data”. In: *Computational Science and Its Applications – ICCSA 2017: 17th International Conference, Trieste, Italy, July 3-6, 2017, Proceedings, Part IV*. Ed. by Osvaldo Gervasi, Beniamino Murgante, Sanjay Misra, Giuseppe Borruso, Carmelo M. Torre, Ana Maria A.C. Rocha, David Taniar, Bernady O. Apduhan, Elena Stankova, and Alfredo Cuzzocrea. Springer, pp. 571–583.
- Schubert, Erich, Jörg Sander, Martin Ester, and Hans Peter Kriegel (2017). “DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN”. In: *ACM Transactions on Database Systems* 42 (3), 19:1–19:21.
- Seidl, Dara E., Piotr Jankowski, and Keith C. Clarke (2018). “Privacy and False Identification Risk in Geomasking Techniques”. In: *Geographical Analysis* 50 (3), pp. 280–297.
- Seidl, Dara E., Piotr Jankowski, and Atsushi Nara (2019). “An Empirical Test of Household Identification Risk in Geomasked Maps”. In: *Cartography and Geographic Information Science* 46 (6), pp. 475–488.
- Seidl, Dara E., Gernot Paulus, Piotr Jankowski, and Melanie Regenfelder (2015). “Spatial Obfuscation Methods for Privacy Protection of Household-Level Data”. In: *Applied Geography* 63, pp. 253–263.
- Sheikholeslami, Gholamhosein, Surojit Chatterjee, and Aidong Zhang (1998). “WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial Databases”. In: *VLDB '98 Proceedings of the 24th International Conference on Very Large Data Bases*. Morgan Kaufmann, pp. 428–439.
- Sherman, Jill E. and Tamara L. Feters (2007). “Confidentiality Concerns with Mapping Survey Data in Reproductive Health Research”. In: *Studies in Family Planning* 38 (4), pp. 309–321.
- Sinnott, Roger W. (1984). “Virtues of the Haversine”. In: *Sky & Telescope* 68 (2), p. 159.
- Snow, John (1854). *On the Mode of Communication of Cholera*. 2nd ed. London: John Churchill.
- Spruill, Nancy L. (1982). “Measures of Confidentiality”. In: *Proceedings of the Survey Research Methods Section*. ASA, pp. 260–265.
- Stinchcomb, Dave (2004). *Procedures for Geomasking to Protect Patient Confidentiality*. Presented at the ESRI International Health GIS Conference held in Washington, DC, October 17-20, 2004.
- Strudler, Michael, H. Lock Oh, and Fritz Scheuren (1986). “Protection of Taxpayer Confidentiality with Respect to the Tax Model”. In: *Proceedings of the Survey Research Methods Section*. American Statistical Association, pp. 375–381.

- Swanlund, David, Nadine Schuurman, and Mariana Brussoni (2020). “MaskMy.XYZ: An Easy-to-Use Tool for Protecting Geoprivacy Using Geographic Masks”. In: *Transactions in GIS* 24 (2), pp. 390–401.
- Swanlund, David, Nadine Schuurman, Paul Zandbergen, and Mariana Brussoni (2020). “Street Masking: A Network-Based Geographic Mask for Easily Protecting Geoprivacy”. In: *International Journal of Health Geographics* 19 (26), pp. 1–11.
- Sweeney, Latanya (2002). “ $k$ -Anonymity: A Model for Protecting Privacy”. In: *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems* 10 (5), pp. 557–570.
- Templ, Matthias, Bernhard Meindl, and Alexander Kowarik (2021). *Package “sdcmicro”: Statistical Disclosure Control Methods for Anonymization of Data and Risk Estimation: Version 5.6.0*. Retrieved: 19. February 2021. URL: <https://cran.r-project.org/web/packages/sdcMicro/sdcMicro.pdf>.
- Tobler, W. R. (1970). “A Computer Movie Simulating Urban Growth in the Detroit Region”. In: *Economic Geography* 46, pp. 234–240.
- Turner, Rolf (2019). *Package “deldir”: Delaunay Triangulation and Dirichlet (Voronoi) Tessellation: Version 0.1-23*. Retrieved: 11. December 2019. URL: <https://cran.r-project.org/web/packages/deldir/index.html>.
- Valiente, Gabriel (2002). *Algorithms on Trees and Graphs*. Berlin: Springer.
- VanWey, Leah K., Ronald R. Rindfuss, Mayron P. Gutmann, Barbara Entwisle, and Deborah L. Balk (2005). “Confidentiality and Spatially Explicit Data: Concerns and Challenges”. In: *Proceedings of the National Academy of Sciences* 102 (43), pp. 15337–15342.
- Vidanage, Anushka, Peter Christen, Thilina Ranbaduge, and Rainer Schnell (2020). “A Graph Matching Attack on Privacy-Preserving Record Linkage”. In: *CIKM '20: Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. ACM, pp. 1485–1494.
- Vincenty, Thaddeus (1975). “Direct and Inverse Solutions of Geodesics on the Ellipsoid with Application of Nested Equations”. In: *Survey Review* 23 (176), pp. 88–93.
- Vine, Marilyn F., Darrah Degnan, and Carol Hanchette (1997). “Geographic Information Systems: Their Use in Environmental Epidemiologic Research”. In: *Environmental Health Perspectives* 105 (6), pp. 598–605.
- Waller, Lance A. and Carol A. Gotway (2004). *Applied Spatial Statistics for Public Health Data*. Hoboken: Wiley.
- Wang, Hao and Jerome P. Reiter (2012). “Multiple Imputation for Sharing Precise Geographies in Public Use Data”. In: *Annals of Applied Statistics* 6 (1), pp. 229–252.
- Wang, Wei, Jiong Yang, and Richard R. Muntz (1997). “STING: A Statistical Information Grid Approach to Spatial Data Mining”. In: *VLDB'97 Proceedings of the 23rd International Conference on Very Large Data Bases*. Morgan Kaufmann, pp. 186–195.

- Wieland, Shannon C., Christopher A. Cassa, Kenneth D. Mandl, and Boonie Berger (2008). “Revealing the Spatial Distribution of a Disease While Preserving Privacy”. In: *PNAS* 105 (46), pp. 17608–17613.
- Wolf, Michael K. (1988). “Microaggregation and Disclosure Avoidance for Economic Establishment Data”. In: *1988 Proceeding of the Business and Economic Statistics Section: Papers Presented at the Annual Meeting of the American Statistical Association, New Orleans, Louisiana, August 22-25, 1988*. American Statistical Association, pp. 355–360.
- Xu, Xiaowei, Martin Ester, Hans-Peter Kriegel, and Jörg Sander (2017). “A Distribution-Based Clustering Algorithm for Mining in Large Spatial Databases”. In: *Proceedings: 14th International Conference on Data Engineering*. Los Alamitos: IEEE, pp. 324–331.
- Young, Caroline, David Martin, and Chris Skinner (2009). “Geographically Intelligent Disclosure Control for Flexible Aggregation of Census Data”. In: *International Journal of Geographical Information Science* 23 (4), pp. 457–482.
- Yuill, Robert S. (1971). “The Standard Deviational Ellipse: An Updated Tool for Spatial Description”. In: *Geografiska Annaler: Series B: Human Geography* 53 (1), pp. 28–39.
- Zandbergen, Paul A. (2014). “Ensuring Confidentiality of Geocode Health Data: Assessing Geographic Masking Strategies for Individual-Level Data”. In: *Advances in Medicine* 2014, pp. 1–14.
- Zhang, Su, Scott M. Friendschuh, Kate Lenzer, and Paul A. Zandbergen (2015). “The Location Swapping Method for Geomasking”. In: *Cartography and Geographic Information Science* 44 (1), pp. 22–34.
- Zhang, Tian, Raghu Ramakrishnan, and Miron Livny (1996). “BIRCH: An Efficient Data Clustering Method for Very Large Databases”. In: *SIGMOD’96 Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*. New York: ACM, pp. 103–114.
- Zhou, Yijie, Francesca Dominici, and Thomas A. Louis (2010). “A Smoothing Approach for Masking Spatial Data”. In: *The Annals of Applied Statistics* 4 (3), pp. 1451–1475.
- Zhu, A-Xing, Guonian Lu, Jing Lu, Cheng-Zhi Qin, and Chenghu Zhou (2018). “Spatial Prediction Based on Third Law of Geography”. In: *Annals of GIS* 24 (4), pp. 225–240.
- Zimmerman, Dale L. and Claire Pavlik (2008). “Quantifying the Effects of Mask Metadata Disclosure and Multiple Releases on the Confidentiality of Geographically Masked Health Data”. In: *Geographical Analysis* 40 (1), pp. 52–76.

## A. Code

Chapter 3 described how each masking method can be implemented. The corresponding code is found in this chapter. Note that the code, for most masking methods only shows the function which was written to apply the masking method without loading data sets or setting the parameters. Furthermore, the code was written in *R* 3.6.

### A.1. Aggregation

#### MDAV

Use of function “microaggregation” in the *sdcMicro*-Package (Templ et al., 2021) in R.

```
klumpen <- microaggregation(daten, variables=c("EAST","NORTH"),
  method='mdav',aggr=k)
klumpencenter <- klumpen$mx
colnames(klumpencenter) <- c("centerx","centery")
```

#### Official Statistics Grid (example for 100 meters grid)

```
EASTM <- round_any(EAST,100,f=floor)
NORTHM <- round_any(NORTH,100,f=floor)
LBcorner <- paste(EASTM,NORTHM,sep=",")
```

```
EASTM <- EASTM+50
NORTHM <- NORTHM+50
CEcorner <- paste(EASTM,NORTHM,sep=",")
```

#### Adaptive Areal Elimination

```
## The given example code is for state electorate-polygons
states <- readOGR("Data/STATE_ELECTORATES_SHAPEFILE.shp",verbose
  =TRUE,stringsAsFactors = FALSE)
statesasdata <- tidy(states, region = "ELECTORATE")
statesasdata$idnum <- as.numeric(as.factor(statesasdata$id))

polys <- statesasdata
names <- unique(polys$group)
names <- as.data.frame(names,stringsAsFactors = FALSE)
neu <- df_to_SpatialPolygons(polys, "group", c("long","lat"),
  CRS("+proj=longlat +ellps=GRS80 +no_defs"))
```

```

polys <- SpatialPolygonsDataFrame(neu, names, match.ID = F)
groupdata <- tidy(polys, region="names")
groupdata$ELECTORATES <- gsub("\\.\\.*", "", groupdata$id)
neu <- df_to_SpatialPolygons(groupdata, "group", c("long", "lat")
  , CRS("+proj=longlat +ellps=GRS80 +no_defs"))
names <- unique(subset(groupdata, select=c("id", "ELECTORATES")))
polys <- SpatialPolygonsDataFrame(neu, names, match.ID = F)

datadf <- read.csv("Data/DATASET.csv", sep=",", stringsAsFactors =
  FALSE)
datadf$n <- 1:nrow(datadf)
data <- read.csv("Data/DATASET.csv", sep=",", stringsAsFactors =
  FALSE)
data$n <- 1:nrow(datadf)
coordinates(data) <- c("LONGITUDE", "LATITUDE")
proj4string(data) <- CRS("+proj=longlat +ellps=GRS80 +no_defs")

neu <- over(data, states)
datadf$poly <- neu$ELECTORATE

sub <- subset(datadf, is.na(poly))
if(nrow(sub)!=0){
  coordinates(sub) <- c("LONGITUDE", "LATITUDE")
  proj4string(sub) <- CRS("+proj=longlat +ellps=GRS80 +no_defs")
  n <- length(sub)
  nearestCantons <- character(n)
  for (i in seq_along(nearestCantons)) {
    nearestCantons[i] <- states$ELECTORATE[which.min(gDistance(
      sub[i,], states, byid=TRUE))]
  }
  nearestCantons <- as.data.frame(nearestCantons,
    stringsAsFactors=FALSE)
  sub <- subset(datadf, is.na(poly))
  sub$poly <- nearestCantons$nearestCantons
  datadf$poly <- ifelse(is.na(datadf$poly),
    sub$poly[match(datadf$n, sub$n)], datadf$poly)
}

## AAE
zentroide <- as.data.frame(centroid(states))
colnames(zentroide) <- c("longcenter", "latcenter")
zentroide <- tibble::rownames_to_column(zentroide, var="centroid"
  )
zentroide$names <- states@data$ELECTORATE
datadf <- merge(datadf, zentroide, by.x="poly", by.y="names", all.x=
  TRUE, all.y=FALSE)

```

```

datadf$centroid <- NULL
colnames(datadf)[which(colnames(datadf)=="longcenter")] <-
  paste0("LONGITUDEM_",e)
colnames(datadf)[which(colnames(datadf)=="latcenter")] <- paste0
  ("LATITUDEM_",e)

## ARP
freqnames <- as.data.frame(table(datadf$poly))
names <- as.character(freqnames$Var1)

result <- NULL
results <- NULL
for (i in 1:length(names)){
  subdata <- subset(statesasdata, id==names[i],select=c("long",
    "lat"))
  polysub <- Polygon(subdata)
  n <- as.numeric(freqnames$Freq[freqnames$Var1==names[i]])
  sample <- spsample(polysub,n=n*3,"random")
  sample <- as.data.frame(sample)
  subnames <- polys@data$id[polys@data$ELECTORATES==names[i]]
  if (length(subnames)==1){
    sample <- sample_n(sample,size=n)
  } else{
    for(v in 1:length(subnames)){
      subsubdata <- subset(groupdata, id==subnames[v],select=c("
long","lat"))
      sample$check <- point.in.polygon(sample$x, sample$y,
subsubdata$long, subsubdata$lat, mode.checked=FALSE)
      colnames(sample)[which(colnames(sample)=="check")] <-
paste0("check",v)
    }
    sample$check <- rowSums(sample[3:ncol(sample)])
    sample <- subset(sample, check==1, select=c("x","y"))
    sample <- sample_n(sample, size=n)
  }
  result <- cbind(sample[1],sample[2],replicate(n,names[i]))
  colnames(result) <- c("longrandom","latrandom","poly2")
  results <- rbind(results,result)
}

datadf <- datadf[order(datadf$poly),]
datadf <- cbind(datadf,results)

```

## A.2. Adjusting Coordinates

### Affine Transformations

```

## displacement using translation
afftransDUT <- function(east,north,xc,yc){
  eastM <- east+xc;
  northM <- north+yc;
  as.data.frame(cbind(df,eastM,northM))
}

## change of scale
afftransCS <- function(east,north,xc,yc){
  eastM <- east*xc;
  northM <- north*yc;
  as.data.frame(cbind(df,eastM,northM))
}

## rotation around origin: angle
afftransRotAngle <- function(east,north,angle){
  eastM <- (east*cosd(angle))-(north*sind(angle));
  northM <- (east*sind(angle))+(north*cosd(angle));
  as.data.frame(cbind(df,eastM,northM))
}

# rotation around origin: radian
afftransRotRadian <- function(east,north,radian){
  eastM <- (east*cos(radian))-(north*sin(radian))
  northM <- (east*sin(radian))+(north*cos(radian))
  as.data.frame(cbind(df,eastM,northM))
}

# rotation around arbitrary point: angle
afftransRotAngleArbitrary <- function(east,north,angle,peast,
  pnorth){
  movedeast <- east-peast;
  movednorth <- north-pnorth;
  eastRot <- (movedeast*cosd(angle))-(movednorth*sind(angle));
  northRot <- (movedeast*sind(angle))+(movednorth*cosd(angle));
  eastM <- eastRot+peast;
  northM <- northRot+pnorth;
  as.data.frame(cbind(df,eastM,northM))
}

## rotation around arbitrary point: radian
afftransRotRadianArbitrary <- function(east,north,angle,peast,
  pnorth){

```

```

movedeast <- east-peast;
movednorth <- north-pnorth;
eastRot <- (movedeast*cos(angle))-(movednorth*sin(angle));
northRot <- (movedeast*sin(angle))+(movednorth*cos(angle));
eastM <- eastRot+peast;
northM <- northRot+pnorth;
as.data.frame(cbind(df,eastM,northM))
}

```

### Random Perturbation

```

## uniform distribution
randpertUnipopdense <- function(east,north,rmin,rmax){
  eastM <- NULL
  northM <- NULL
  for(num in 1:length(east)){
    random <- runif(1,0,10)
    if(random<=5){
      eastM[num] <- east[num]-runif(1,min=rmin[num],max=rmax[num])
    } else {
      eastM[num] <- east[num]+runif(1,min=rmin[num],max=rmax[num])
    }
    random <- runif(1,0,10)
    if(random<=5){
      northM[num] <- north[num]-runif(1,min=rmin[num],max=rmax[num])
    } else {
      northM[num] <- north[num]+runif(1,min=rmin[num],max=rmax[num])
    }
  }
  as.data.frame(cbind(east,north,eastM,northM))
}

```

```

## normal distribution
randpertNorm <- function(east,north,mean,sd){
  eastM <- NULL
  northM <- NULL
  for(num in 1:length(east)){
    xrand <- rnorm(1,mean=mean,sd=sd[num])
    yrand <- rnorm(1,mean=mean,sd=sd[num])
    while(xrand==0 & yrand==0){ # make sure that xrand and yrand
      is both not 0; at least one has to be > 0
      xrand <- rnorm(1,mean=mean,sd=sd[num])
    }
  }
}

```

```

    yrand <- rnorm(1,mean=mean,sd=sd[num])
  }
  eastM[num] <- east[num]+xrand
  northM[num] <- north[num]+yrand
}
as.data.frame(cbind(east,north,eastM,northM))
}

```

```

## within a circle
randpertCircle <- function(east,north,rmax){
  angle <- round(runif(length(east),0,360));
  Rrand <- runif(length(east),0,rmax);
  rotx <- Rrand*cosd(angle);
  roty <- Rrand*sind(angle);
  eastM <- rotx+east;
  northM <- roty+north;
  as.data.frame(cbind(east,north,eastM,northM))
}

```

### Donut Geomasking

```

donut <- function(east,north,rmin,rmax){
  angle <- round(runif(length(east),0,360));
  Rrand <- runif(length(east),rmin,rmax);
  rotx <- Rrand*cosd(angle);
  roty <- Rrand*sind(angle);
  eastM <- rotx+east;
  northM <- roty+north;
  as.data.frame(cbind(east,north,eastM,northM))
}

```

```

## finding rmin and rmax based on k-nearest-neighbor
kmaxmin <- function(longitude,latitude,reslong,reslat,maxk,mink)
{
  coords <- as.data.frame(matrix(c(longitude,latitude),ncol=2))
  colnames(coords) <- c("LONGITUDE","LATITUDE")
  coords$id <- paste0(coords$LONGITUDE,coords$LATITUDE)
  rescoords <- as.data.frame(matrix(c(reslong,reslat),ncol=2))
  colnames(rescoords) <- c("LONGITUDE","LATITUDE")
  rescoords$id <- paste0(rescoords$LONGITUDE,rescoords$LATITUDE)
  rescoords <- rescoords[!rescoords$id %in% coords$id, ]
  rescoords <- matrix(c(rescoords$LONGITUDE,rescoords$LATITUDE),
    ncol=2)
  coords <- as.data.frame(matrix(c(longitude,latitude),ncol=2))
}

```

```

maxradius <- NULL
minradius <- NULL

no_cores <- detectCores() - 2
cl<-makeCluster(no_cores)
registerDoSNOW(cl)
iterations <- length(longitude)
pb <- txtProgressBar(max=iterations, style=3)
progress <- function(n) setTxtProgressBar(pb,n)
opts <- list(progress=progress)

results = foreach(z=1:length(longitude),.combine=rbind,.
  options.snow=opts,.packages="dplyr") %dopar% {
  coord_df <- data.frame(rescoords, dist = geosphere::
    distHaversine(rescoords,c(longitude[z],latitude[z])))
  maxradius[z] <- sort(coord_df$dist,partial=maxk)[maxk]
  minradius[z] <- sort(coord_df$dist,partial=mink)[mink]
  return(c(maxradius[z],minradius[z]))
}
close(pb)
stopCluster(cl)
as.data.frame(cbind(unnname(results[,1]),unnname(results[,2])))
}

# using data set as reference
kmaxmin2 <- function(longitude,latitude,reslong,reslat,maxk,mink
){
  coords <- as.data.frame(matrix(c(longitude,latitude),ncol=2))
  colnames(coords) <- c("LONGITUDE","LATITUDE")
  coords$id <- paste0(coords$LONGITUDE,coords$LATITUDE)
  rescoords <- as.data.frame(matrix(c(reslong,reslat),ncol=2))
  colnames(rescoords) <- c("LONGITUDE","LATITUDE")
  rescoords$id <- paste0(rescoords$LONGITUDE,rescoords$LATITUDE)
  rescoords <- matrix(c(rescoords$LONGITUDE,rescoords$LATITUDE),
    ncol=2)
  coords <- as.data.frame(matrix(c(longitude,latitude),ncol=2))

  maxradius <- NULL
  minradius <- NULL

  no_cores <- detectCores() - 2
  cl<-makeCluster(no_cores)
  registerDoSNOW(cl)
  iterations <- length(longitude)
  pb <- txtProgressBar(max=iterations, style=3)
  progress <- function(n) setTxtProgressBar(pb,n)

```

```

opts <- list(progress=progress)

results = foreach(z=1:length(longitude),.combine=rbind,.
  options.snow=opts,.packages="dplyr") %dopar% {
  coord_df <- data.frame(rescoords, dist = geosphere::
    distHaversine(rescoords,c(longitude[z],latitude[z]))
  maxradius[z] <- sort(coord_df$dist,partial=maxk)[maxk]
  minradius[z] <- sort(coord_df$dist,partial=mink)[mink]
  return(c(maxradius[z],minradius[z]))
}
close(pb)
stopCluster(cl)
as.data.frame(cbind(unnamed(results[,1]),unnamed(results[,2])))
}

```

### Voronoi Masking

```

voronoi <- function(lon,lat){
  voronoi <- deldir(lon, lat)
  voronoiDF <- as.data.frame(voronoi$dirsgs)
  voronoiDF$id <- seq(dim(voronoiDF)[1]);
  rm(voronoi)

  no_cores <- detectCores() - 2
  cl<-makeCluster(no_cores)
  registerDoSNOW(cl)
  iterations <- length(lon)
  pb <- txtProgressBar(max=iterations, style=3)
  progress <- function(n) setTxtProgressBar(pb,n)
  opts <- list(progress=progress)

  LongitudeM <- NULL
  LatitudeM <- NULL

  results = foreach(i=1:length(lon),.combine=rbind,.packages=c("
    mapproj","tidyr","geosphere"),.options.snow=opts) %dopar% {
    p <- c(lon[i],lat[i])
    result <- NULL
    voronoiSub <- subset(voronoiDF, voronoiDF$ind1==i |
    voronoiDF$ind2==i)
    if(nrow(voronoiSub)==0){
      voronoiSub <- voronoiDF
    }
    for(v in 1:nrow(voronoiSub)){
      line <- matrix(c(voronoiSub$x1[v],voronoiSub$x2[v],
        voronoiSub$y1[v],voronoiSub$y2[v]),ncol=2,nrow=2)
    }
  }
}

```

```

    result <- rbind(result, dist2Line(p,line,distfun=
distHaversine))
  }
  for(u in 1:nrow(result)){
    if(result[u,1]==min(result[,1])){
      move <- result[u,2:3]
    }
  }
  if(length(move)>2){
    move <- move[1:2]
  }
  LongitudeM[i] <- move[1]
  LatitudeM[i] <- move[2]
  return(c(LongitudeM[i],LatitudeM[i]))
}
close(pb)
stopCluster(cl)
names(results)[names(results) == "V1"] <- "LongitudeM"
names(results)[names(results) == "V2"] <- "LatitudeM"
as.data.frame(results)
}

```

### Location Swapping

```

locationswapping <- function(Longitude, Latitude, radius,
  residentLongitude, residentLatitude){
  coord <- c(paste(Longitude, Latitude, sep=", "))
  res <- as.data.frame(cbind(residentLongitude, residentLatitude)
  )
  res$test <- paste(residentLongitude, residentLatitude, sep=", ")
  res <- res[!res$test %in% coord, ]
  res$test <- NULL
  colnames(res) <- c("Longitude", "Latitude")

  rad <- radius
  LongitudeM <- NULL
  LatitudeM <- NULL

  no_cores <- detectCores() - 2
  cl<-makeCluster(no_cores)
  registerDoSNOW(cl)
  iterations <- length(Longitude)
  pb <- txtProgressBar(max=iterations, style=3)
  progress <- function(n) setTxtProgressBar(pb,n)
  opts <- list(progress=progress)

```

```

results = foreach(i=1:NROW(coord),.combine=rbind,.options.snow
=opts,.packages="dplyr") %dopar% {
  coord_df <- data.frame(res,
  within_radius = geosphere::distHaversine(res, c(Longitude[i]
], Latitude[i])) < rad[i])
  subcoord <- subset(coord_df, within_radius==TRUE)
  if(NROW(subcoord)==0){
    distance <- geosphere::distHaversine(res, c(Longitude[i],
Latitude[i]))
    mindist <- min(distance)
    coord_df <- data.frame(res,
    within_radius = geosphere::distHaversine(res, c(Longitude[
i], Latitude[i])) <= mindist)
    subcoord <- subset(coord_df, within_radius==TRUE)
  }
  samp <- sample_n(subcoord,1)
  LongitudeM[i] <- samp$Longitude #coord_df$Longitude[coord_df
$id==samp]
  LatitudeM[i] <- samp$Latitude #coord_df$Latitude[coord_df$
id==samp]
  return(c(LongitudeM[i],LatitudeM[i]))
}
close(pb)
stopCluster(cl)
results <- as.data.frame(results)
names(results)[names(results) == "V1"] <- "LongitudeM"
names(results)[names(results) == "V2"] <- "LatitudeM"
as.data.frame(results)
}

## location swapping using donut
locationswappingdonut <- function(Longitude, Latitude, radius,
residentLongitude, residentLatitude){
  coord <- c(paste(Longitude, Latitude, sep=", "))
  res <- as.data.frame(cbind(residentLongitude, residentLatitude)
)
  res$test <- paste(residentLongitude, residentLatitude, sep=", ")
  res <- res[!res$test %in% coord, ]
  res$test <- NULL
  colnames(res) <- c("Longitude", "Latitude")

  rad <- radius
  LongitudeM <- NULL
  LatitudeM <- NULL

  no_cores <- detectCores() - 2

```

```

cl<-makeCluster(no_cores)

registerDoSNOW(cl)
iterations <- length(Longitude)
pb <- txtProgressBar(max=iterations, style=3)
progress <- function(n) setTxtProgressBar(pb,n)
opts <- list(progress=progress)

results = foreach(i=1:NROW(coord),.combine=rbind,.options.snow
=opts,.packages="dplyr") %dopar% {
  coord_df <- data.frame(res,
  radius = geosphere::distHaversine(res, c(Longitude[i],
Latitude[i])))
  coord_df$within_radius <- coord_df$radius<rad[i] & coord_df$
radius>(rad[i]/2)
  subcoord <- subset(coord_df, within_radius==TRUE)
  if(NROW(subcoord)==0){
    distance <- geosphere::distHaversine(res, c(Longitude[i],
Latitude[i]))
    mindist <- min(distance)
    coord_df <- data.frame(res,
  within_radius = geosphere::distHaversine(res, c(Longitude[
i], Latitude[i])) <= mindist)
    subcoord <- subset(coord_df, within_radius==TRUE)
  }
  samp <- sample_n(subcoord,1)
  LongitudeM[i] <- samp$Longitude
  LatitudeM[i] <- samp$Latitude
  return(c(LongitudeM[i],LatitudeM[i]))
}
close(pb)
stopCluster(cl)
results <- as.data.frame(results)
names(results)[names(results) == "V1"] <- "LongitudeM"
names(results)[names(results) == "V2"] <- "LatitudeM"
as.data.frame(results)
}

```

### Verified Neighbor Approach

```

verifiedneighbour <- function(Longitude, Latitude, category, radius
, residentLongitude, residentLatitude, rescategory, k){
  coord <- c(paste(Longitude, Latitude, sep=", "))
  residents <- as.data.frame(cbind(residentLongitude,
  residentLatitude))
  residents$rescategory <- rescategory

```

```

residents$test <- paste(residentLongitude, residentLatitude, sep
="," )
residents <- residents[!residents$test %in% coord, ]
residents$test <- NULL
colnames(residents) <- c("Longitude", "Latitude", "category")
res <- subset(residents, select=c(Longitude, Latitude))

rad <- radius

no_cores <- detectCores() - 2
cl<-makeCluster(no_cores)
registerDoSNOW(cl)
iterations <- length(Longitude)
pb <- txtProgressBar(max=iterations, style=3)
progress <- function(n) setTxtProgressBar(pb,n)
opts <- list(progress=progress)

LongitudeM <- NULL
LatitudeM <- NULL

results = foreach(i=1:NROW(coord), .combine=rbind, .options.snow
=opts, .packages="dplyr") %dopar% {
  cat <- category[i]
  ressub <- subset(residents, residents$category==cat, select=c
("Longitude", "Latitude"))
  coord_df <- data.frame(ressub, within_radius = geosphere::
distHaversine(ressub, c(Longitude[i], Latitude[i])) < rad[i
])
  coord_df$rescategory <- ressub$category
  subcoord <- subset(coord_df, within_radius==TRUE)
  if(NROW(subcoord)<k){
    cat <- category[i]
    ressub <- subset(residents, residents$category==cat, select
=c("Longitude", "Latitude"))
    distance <-geosphere::distHaversine(ressub, c(Longitude[i
], Latitude[i]))
    distance <- sort(distance)
    mindist <- distance[k]
    coord_df <- data.frame(ressub, within_radius = geosphere::
distHaversine(ressub, c(Longitude[i], Latitude[i])) <=
mindist)
    coord_df$rescategory <- ressub$category
    subcoord <- subset(coord_df, within_radius==TRUE)
  }
  samp <- sample_n(subcoord,1)

```

```

    LongitudeM[i] <- samp$Longitude#coord_df$Longitude[coord_df$
id==samp]
    LatitudeM[i] <- samp$Latitude#coord_df$Latitude[coord_df$id
==samp]
    return(c(LongitudeM[i],LatitudeM[i]))
}
close(pb)
stopCluster(cl)
results <- as.data.frame(results)
names(results)[names(results) == "V1"] <- "LongitudeM"
names(results)[names(results) == "V2"] <- "LatitudeM"
as.data.frame(results)
}

```

## Street Masking

### Python Code

```

import osmnx
from osmnx import graph_from_bbox
from osmnx.distance import add_edge_lengths
from osmnx.utils_graph import remove_isolated_nodes
from osmnx.io import save_graph_shapefile
G = graph_from_bbox(north=-25.9963, south=-38.06260, east
    =141.00296, west=129.00130, network_type="drive", truncate_by
    _edge=True)
G = remove_isolated_nodes(G)
G = add_edge_lengths(G)
osmnx.io.save_graph_shapefile(G, filepath=None, encoding='utf-8',
    directed=False)

```

### R Code

```

nodes <- read.dbf("StreetMask/nodes.dbf")
nodes <- subset(nodes, select=c("osmid","y","x"))
colnames(nodes) <- c("osmid","LATITUDE","LONGITUDE")

time <- NULL
depth <- 30
work <- subset(nodes, select=c("LONGITUDE","LATITUDE"))

no_cores <- detectCores() - 1
cl<-makeCluster(no_cores)
registerDoSNOW(cl)
iterations <- NROW(daten)
pb <- txtProgressBar(max=iterations, style=3)
progress <- function(n) setTxtProgressBar(pb,n)
opts <- list(progress=progress)

```

```

closest <- NULL
max <- NROW(daten)
closest = foreach(i=1:max,.combine='c',.options.snow=opts,.
  packages=c("geosphere","data.table")) %dopar% {
  coord_df <- data.frame(work, dist = geosphere::distHaversine(
    work, c(daten$LONGITUDE[i], daten$LATITUDE[i])))
  coord_df$osmid <- nodes$osmid
  x <- coord_df$osmid[coord_df$dist==min(coord_df$dist)]
  return(x)
}
close(pb)
stopCluster(cl)

daten$osmid <- closest
end1 <- Sys.time()

no_cores <- detectCores() - 1
cl<-makeCluster(no_cores)
registerDoSNOW(cl)
iterations <- NROW(daten)
pb <- txtProgressBar(max=iterations, style=3)
progress <- function(n) setTxtProgressBar(pb,n)
opts <- list(progress=progress)

LongitudeM <- NULL
LatitudeM <- NULL
result <- NULL
max <- NROW(daten)
result = foreach(i=1:max,.combine=rbind,.options.snow=opts,.
  packages=c("geosphere","dplyr")) %dopar% {
  coord_df <- data.frame(work, dist = geosphere::distHaversine(
    work, c(nodes$LONGITUDE[nodes$osmid==daten$osmid[i]], nodes$
      LATITUDE[nodes$osmid==daten$osmid[i]])))
  coord_df$osmid <- nodes$osmid

  coord_df <- coord_df[order(coord_df$dist),]
  avgdist <- mean(coord_df$dist[2:(depth+1)])

  sub <- subset(coord_df,dist<avgdist & daten$osmid[i]!=coord_df
    $osmid)
  mask <- sample_n(sub,1)
  LongitudeM[i] <- mask$LONGITUDE
  LatitudeM[i] <- mask$LATITUDE

  return(c(LongitudeM[i],LatitudeM[i]))
}

```

```

}
close(pb)
stopCluster(cl)
result <- as.data.frame(result)
names(result)[names(result) == "V1"] <- "LongitudeM"
names(result)[names(result) == "V2"] <- "LatitudeM"

daten$LONGITUDEM <- result$LongitudeM
daten$LATITUDEM <- result$LatitudeM

```

### A.3. Coordinate Replacement

#### Random Projection

```

randomprojection <- function(east,north,n){
  points <- as.data.frame(cbind(east,north))
  slope <- NULL
  intercept <- NULL
  choice <- NULL
  for(i in 1:n){
    reast <- runif(2,min=min(east),max=max(east))
    rnorth <- runif(2,min=min(north),max=max(north))

    slope[i] <- (rnorth[2]-rnorth[1])/(reast[2]-reast[1])
    intercept[i] <- -slope[i]*reast[1]+rnorth[1]
    predicted <-slope[i]*east+intercept[i]
    residuals <- north-predicted #observedy-predictedy
    choice[residuals>0] <- "1"
    choice[residuals<0] <- "0"

    points <- cbind(points,choice)
    colnames(points)[2+i] <- paste0("choice",i)
    choice <- NULL
  }
  points <- points[,3:ncol(points)]
  masked <- col_concat(points, sep = "")
  return(c(masked,slope,intercept))
}

```

#### Anonymization of Distance Matrices via Lipschitz Embedding

Use of code by Martin Kroll, provided by Prof. Dr. Rainer Schnell.

```

sa <- shapefile("Data/STATE_SHAPEFILE.shp")
daten <- fread("Data/DATASET.csv",sep=",")
coordinates(daten) <- c("LONGITUDE","LATITUDE")
proj4string(daten) <- CRS("+proj=longlat +ellps=GRS80 +no_defs")

```

```
d <- 60
k <- 20
N <- length(daten)
lipschitz.coordinates <- matrix(0, nrow = N, ncol = d)

for (i in 1:d) {
  reference.set <- spsample(sa, k, type = "random", iter = +Inf)
  temp <- spDists(daten, reference.set, longlat = TRUE)
  temp <- apply(temp, 1, min)
  lipschitz.coordinates[, i] <- temp
}

D.approx <- dist(lipschitz.coordinates, method = "maximum", diag
  = TRUE, upper = TRUE)
D.approx <- as.matrix(D.approx)
```

### **Distance Approximation using Intersecting Sets of Grid Points**

Based on the code of Schnell, Klingwort, et al. (2021), minor changes were made to calculate the distances between points and not to points of interest.

## A.4. Overview of Masking Methods (Detailed)

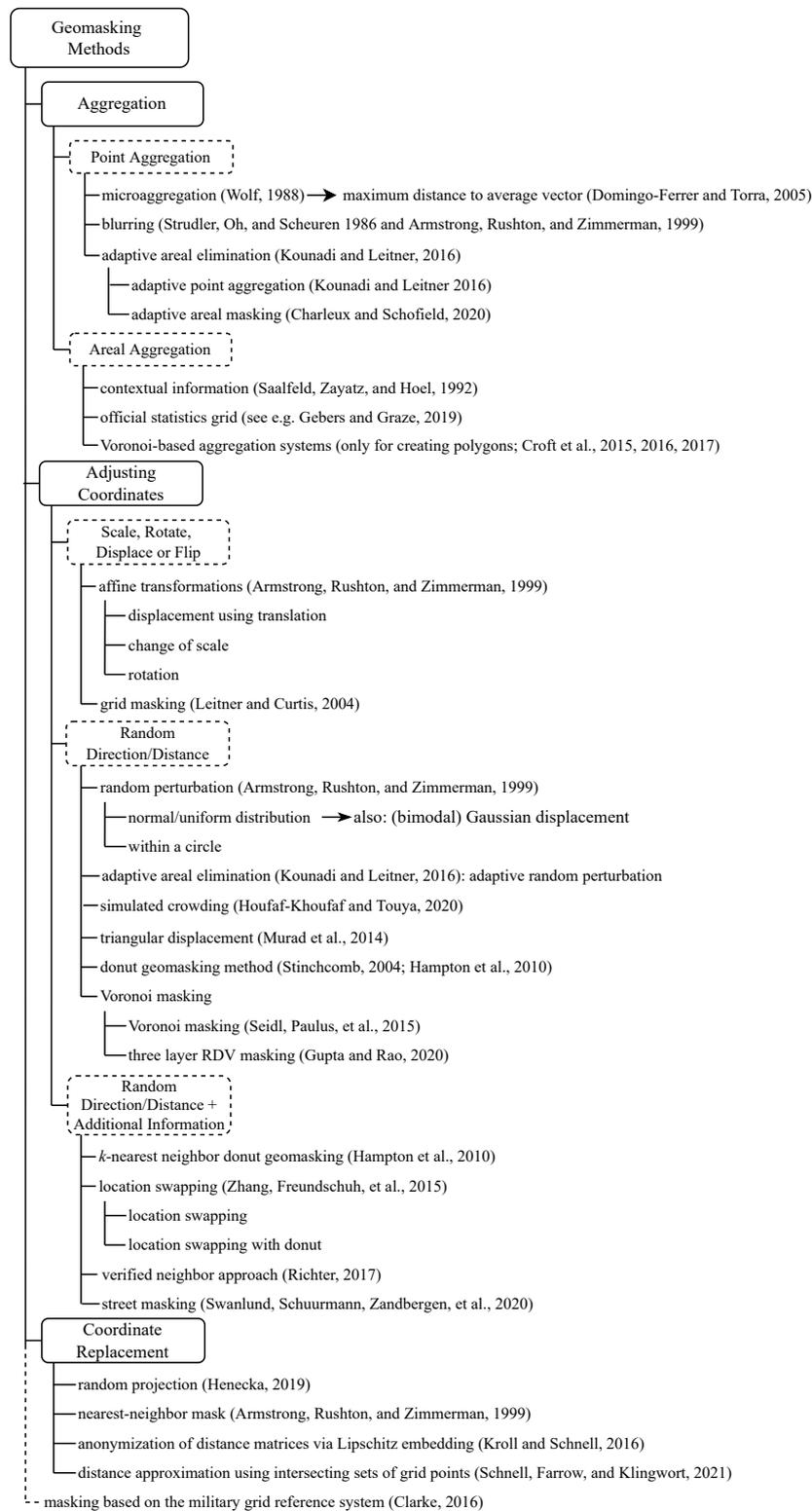


Figure A.1.: Detailed overview of masking methods.

## A.5. Abbreviations of Masking Methods and Parameter Choices

APA	adaptive point aggregation using state electorates
APA LGA	adaptive point aggregation using local government areas
ARP	adaptive random perturbation using state electorates
ARP LGA	adaptive random perturbation using local government areas
CS	change of scale (random number between 0 and 2)
DD 3	donut masking using the estimation of the average distance between people based on postcode population density multiplied by 3 (minimum radius multiplied by 2)
DD 4	donut masking using the estimation of the average distance between people based on postcode population density multiplied by 4 (minimum radius multiplied by 2)
DD 5	donut masking using the estimation of the average distance between people based on postcode population density multiplied by 5 (minimum radius multiplied by 2)
DD LGA 3	donut masking using the estimation of the average distance between people based on local government area population density multiplied by 3 (minimum radius multiplied by 2)
DD LGA 4	donut masking using the estimation of the average distance between people based on local government area population density multiplied by 4 (minimum radius multiplied by 2)
DD LGA 5	donut masking using the estimation of the average distance between people based on local government area population density multiplied by 5 (minimum radius multiplied by 2)
Dk 5	$k$ -nearest neighbor donut masking (maximum radius $k = 5$ , minimum radius $k = 2$ )
Dk 25	$k$ -nearest neighbor donut masking (maximum radius $k = 25$ , minimum radius $k = 2$ )
Dk 50	$k$ -nearest neighbor donut masking (maximum radius $k = 50$ , minimum radius $k = 5$ )
Dk 100	$k$ -nearest neighbor donut masking (maximum radius $k = 100$ , minimum radius $k = 10$ )
Dk 500	$k$ -nearest neighbor donut masking (maximum radius $k = 500$ , minimum radius $k = 50$ )
Dk 1000	$k$ -nearest neighbor donut masking (maximum radius $k = 1000$ , minimum radius $k = 100$ )
DkData 20	$k$ -nearest neighbor donut masking using the data set as reference file (maximum radius $k = 20$ , minimum radius $k = 2$ )

---

DUT	displacement using translation (random number between -10 km and 10 km)
Grid 100	100 meters official statistics grid
Grid 1000	1000 meters official statistics grid
ISGP	distance approximation using intersecting sets of grid points ( $r = 40000$ , 5000000 grid points)
Lipschitz	anonymization of distance matrices via Lipschitz embedding ( $d = 60$ , $k = 20$ )
LS 3	location swapping using the estimation of the average distance between people based on postcode population density multiplied by 3
LS 4	location swapping using the estimation of the average distance between people based on postcode population density multiplied by 4
LS 5	location swapping using the estimation of the average distance between people based on postcode population density multiplied by 5
LS LGA 3	location swapping using the estimation of the average distance between people based on local government area population density multiplied by 3
LS LGA 4	location swapping using the estimation of the average distance between people based on local government area population density multiplied by 4
LS LGA 5	location swapping using the estimation of the average distance between people based on local government area population density multiplied by 5
LSdonut 3	location swapping with donut using the estimation of the average distance between people based on postcode population density multiplied by 3 (minimum radius is half of maximum radius)
LSdonut 4	location swapping with donut using the estimation of the average distance between people based on postcode population density multiplied by 4 (minimum radius is half of maximum radius)
LSdonut 5	location swapping with donut using the estimation of the average distance between people based on postcode population density multiplied by 5 (minimum radius is half of maximum radius)
LSdonut LGA 3	location swapping with donut using the estimation of the average distance between people based on local government area population density multiplied by 3 (minimum radius is half of maximum radius)

---

LSdonut LGA 4	location swapping with donut using the estimation of the average distance between people based on local government area population density multiplied by 4 (minimum radius is half of maximum radius)
LSdonut LGA 5	location swapping with donut using the estimation of the average distance between people based on local government area population density multiplied by 5 (minimum radius is half of maximum radius)
MDAV 3	MDAV using cluster size 3
MDAV 25	MDAV using cluster size 25
MDAV 50	MDAV using cluster size 50
RandProj 100	random projection using 100 lines
RandProj 200	random projection using 200 lines
RandProj 300	random projection using 300 lines
RandProj 500	random projection using 500 lines
RandProj 1000	random projection using 1000 lines
Rot	rotation around the origin
RotArb	rotation around the spatial mean center
RPC 3	random perturbation within a circle using the estimation of the average distance between people based on postcode population density multiplied by 3 as maximum radius
RPC 4	random perturbation within a circle using the estimation of the average distance between people based on postcode population density multiplied by 4 as maximum radius
RPC 5	random perturbation within a circle using the estimation of the average distance between people based on postcode population density multiplied by 5 as maximum radius
RPC LGA 3	random perturbation within a circle using the estimation of the average distance between people based on local government area population density multiplied by 3 as maximum radius
RPC LGA 4	random perturbation within a circle using the estimation of the average distance between people based on local government area population density multiplied by 4 as maximum radius
RPC LGA 5	random perturbation within a circle using the estimation of the average distance between people based on local government area population density multiplied by 5 as maximum radius
RPN	random perturbation using normal distribution ( $\bar{x} = 0$ , sd is estimation of the average distance between people based on postcode population density)

---

RPN LGA	random perturbation using normal distribution ( $\bar{x} = 0$ , sd is estimation of the average distance between people based on local government population density)
RPU 3	random perturbation using uniform distribution using the estimation of the average distance between people based on postcode population density multiplied by 3 as maximum radius (multiplied by 2 as minimum radius)
RPU 4	random perturbation using uniform distribution using the estimation of the average distance between people based on postcode population density multiplied by 4 as maximum radius (multiplied by 2 as minimum radius)
RPU 5	random perturbation using uniform distribution using the estimation of the average distance between people based on postcode population density multiplied by 5 as maximum radius (multiplied by 2 as minimum radius)
RPU LGA 3	random perturbation using uniform distribution using the estimation of the average distance between people based on local government area population density multiplied by 3 as maximum radius (multiplied by 2 as minimum radius)
RPU LGA 4	random perturbation using uniform distribution using the estimation of the average distance between people based on local government area population density multiplied by 4 as maximum radius (multiplied by 2 as minimum radius)
RPU LGA 5	random perturbation using uniform distribution using the estimation of the average distance between people based on local government area population density multiplied by 5 as maximum radius (multiplied by 2 as minimum radius)
StreetMask 30	street masking with depth = 30
StreetMask 100	street masking with depth = 100
VNE 50 3	verified neighbor masking using the estimation of the average distance between people based on postcode population density multiplied by 3 as radius, employment status as variable and $k = 50$
VNE 50 5	verified neighbor masking using the estimation of the average distance between people based on postcode population density multiplied by 5 as radius, employment status as variable and $k = 50$
VNE 100 3	verified neighbor masking using the estimation of the average distance between people based on postcode population density multiplied by 3 as radius, employment status as variable and $k = 100$

---

VNE 100 5	verified neighbor masking using the estimation of the average distance between people based on postcode population density multiplied by 5 as radius, employment status as variable and $k = 100$
VNS 50 3	verified neighbor masking using the estimation of the average distance between people based on postcode population density multiplied by 3 as radius, sex as variable and $k = 50$
VNS 50 5	verified neighbor masking using the estimation of the average distance between people based on postcode population density multiplied by 5 as radius, sex as variable and $k = 50$
VNS 100 3	verified neighbor masking using the estimation of the average distance between people based on postcode population density multiplied by 3 as radius, sex as variable and $k = 100$
VNS 100 5	verified neighbor masking using the estimation of the average distance between people based on postcode population density multiplied by 5 as radius, sex as variable and $k = 100$
VNE LGA 50 3	verified neighbor masking using the estimation of the average distance between people based on local government area population density multiplied by 3 as radius, employment status as variable and $k = 50$
VNE LGA 50 5	verified neighbor masking using the estimation of the average distance between people based on local government area population density multiplied by 5 as radius, employment status as variable and $k = 50$
VNE LGA 100 3	verified neighbor masking using the estimation of the average distance between people based on local government area population density multiplied by 3 as radius, employment status as variable and $k = 100$
VNE LGA 100 5	verified neighbor masking using the estimation of the average distance between people based on local government area population density multiplied by 5 as radius, employment status as variable and $k = 100$
VNS LGA 50 3	verified neighbor masking using the estimation of the average distance between people based on local government area population density multiplied by 3 as radius, sex as variable and $k = 50$
VNS LGA 50 5	verified neighbor masking using the estimation of the average distance between people based on local government area population density multiplied by 5 as radius, sex as variable and $k = 50$

---

VNS LGA 100 3	verified neighbor masking using the estimation of the average distance between people based on local government area population density multiplied by 3 as radius, sex as variable and $k = 100$
VNS LGA 100 5	verified neighbor masking using the estimation of the average distance between people based on local government area population density multiplied by 5 as radius, sex as variable and $k = 100$
Voronoi	Voronoi masking



## B. DBSCAN: Parameter Choice

For *MinPts* the recommendation ( $\text{MinPts} = 4$ ) of the authors was used (Ester et al., 1996). For the radius  $\varepsilon$  the sorted  $k$ -dist graph, here 4-dist graph, was plotted as can be shown in figure B.1.

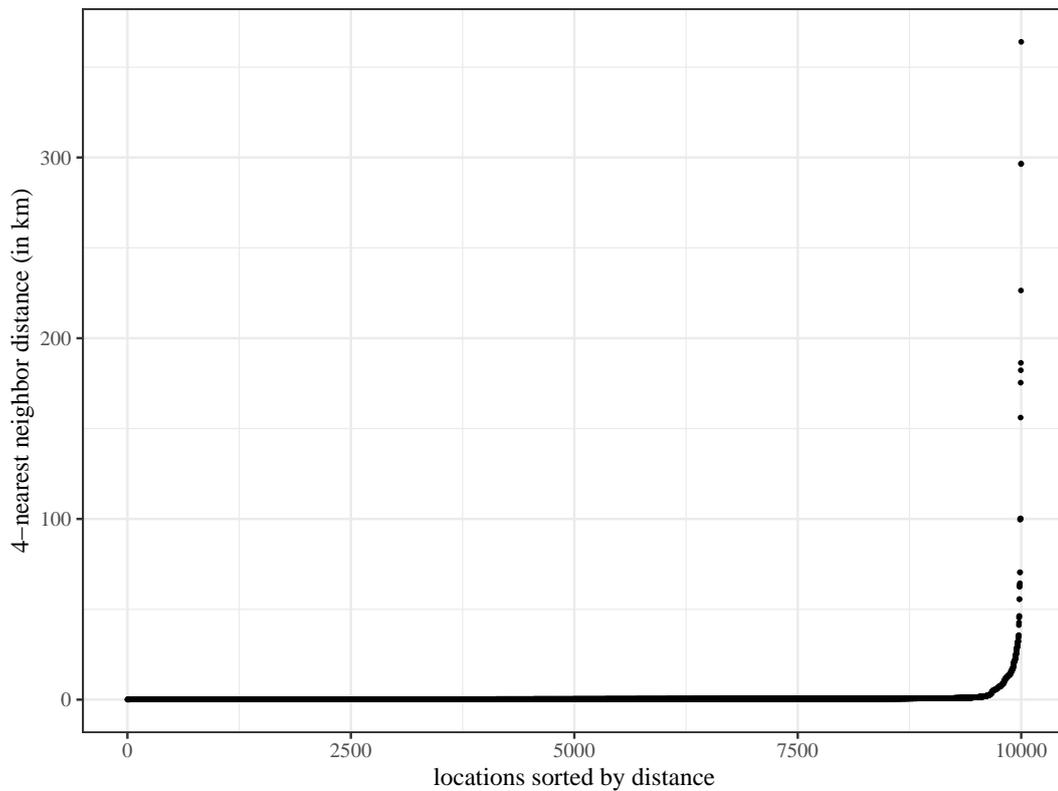


Figure B.1.:  $k$ -nearest neighbor plot for finding the optimal value of the radius  $\varepsilon$ . Distances were sorted in ascending order.

The authors state to look for the “valley” (Ester et al., 1996, p. 230). Therefore, the figure B.2 shows a closer view of the lower right quadrant. Here, there are two points where the distance suddenly increases. Therefore, the distances 3.2 km and 9.5 km were used as the value for the radius  $\varepsilon$ .

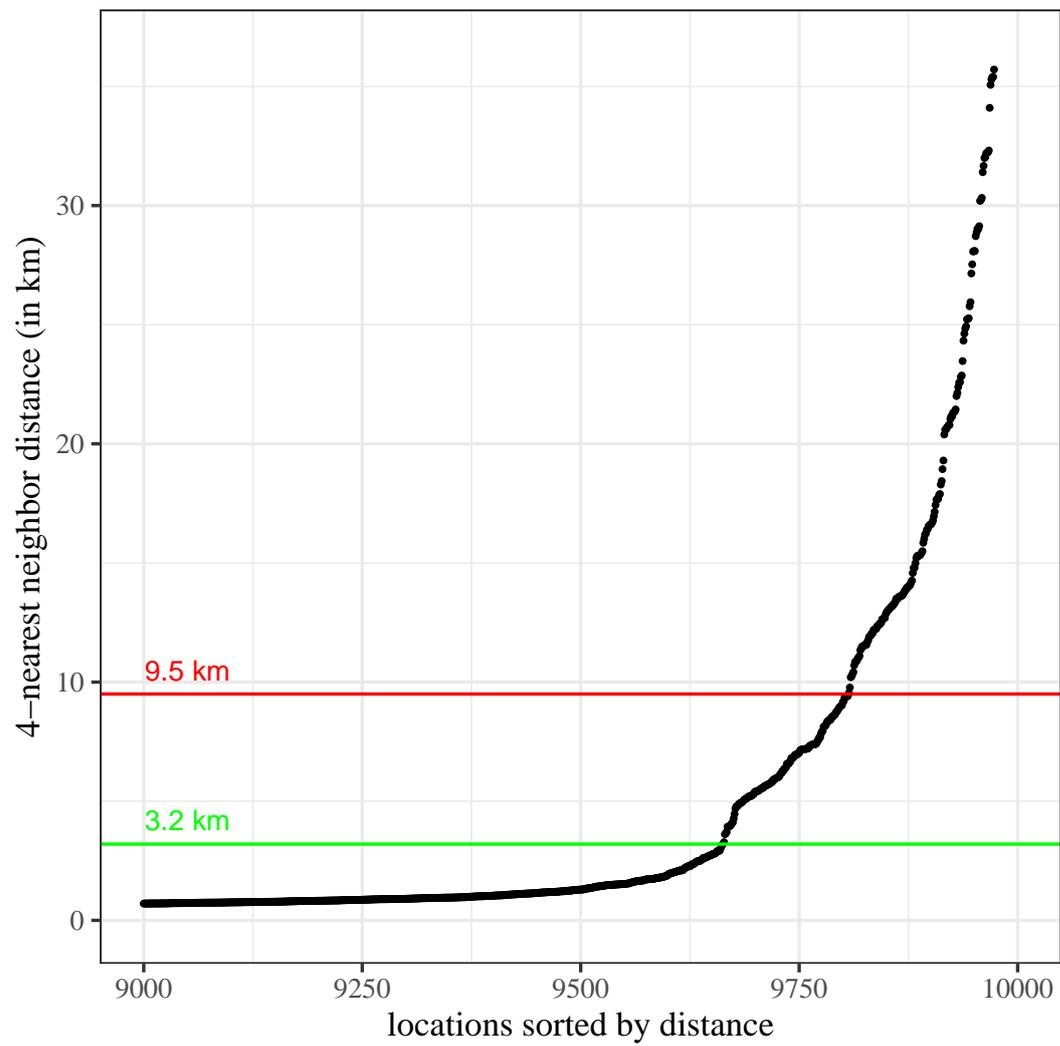


Figure B.2.:  $k$ -nearest neighbor plot for finding the optimal value of the radius  $\epsilon$ . Distances were sorted in ascending order. Zoomed in for better evaluation.

## C. Frequency Table of Overlap Between Data sets

Table C.1.: Combinations of sex, employment, and age of the overlap of the masked and identification file ( $n = 10,000$ , sorted by frequency of overlap). The fifth and sixth column contains the number of people with the respective combinations in the masked and identification file.

sex	employment	age	freq overlap	freq masked	freq ident
F	undefined	65-74	54	514	444
F	undefined	75+	49	546	550
M	Full	45-54	47	424	446
M	Full	25-34	44	418	398
M	Full	35-44	41	424	412
M	undefined	65-74	41	423	452
M	undefined	75+	41	415	371
M	Full	55-64	31	274	314
M	undefined	55-64	27	256	243
F	Full	45-54	23	257	254
F	Full	55-64	23	184	196
F	Part	25-34	23	227	221
F	Part	45-54	23	265	253
F	undefined	15-19	23	182	166
F	Part	35-44	21	243	233
F	Part	55-64	21	209	193
F	undefined	25-34	21	179	185
F	undefined	55-64	21	313	321
M	undefined	15-19	21	185	207
F	Full	25-34	19	234	238
F	undefined	35-44	19	188	175
M	undefined	35-44	18	134	139
M	undefined	45-54	18	151	142
F	undefined	45-54	16	196	214
M	Part	45-54	14	121	97
F	Full	35-44	13	190	189
F	Part	15-19	13	118	143

sex	employment	age	freq overlap	freq masked	freq ident
M	Full	65-74	13	72	70
F	Part	20-24	12	114	144
F	undefined	20-24	12	125	120
F	unemp	15-19	12	56	55
M	undefined	20-24	12	110	105
M	Part	20-24	11	102	95
M	Part	55-64	11	107	107
M	undefined	25-34	11	129	130
M	Part	25-34	10	126	107
F	afw	45-54	9	36	43
M	Part	35-44	9	95	89
F	Full	20-24	8	96	107
F	Part	65-74	8	79	70
M	Part	15-19	8	71	91
M	unemp	15-19	8	52	61
M	unemp	20-24	8	63	74
M	unemp	25-34	8	70	78
F	unemp	25-34	7	53	50
F	Full	15-19	6	33	28
M	Full	15-19	6	47	56
M	Full	20-24	6	127	121
M	unemp	35-44	6	48	52
M	unemp	45-54	6	52	54
F	afw	25-34	5	70	61
F	unemp	45-54	5	48	45
M	afw	25-34	5	38	33
M	Part	65-74	5	87	70
M	unemp	55-64	5	36	53
M	afw	35-44	4	38	37
F	afw	35-44	3	40	45
F	afw	55-64	3	40	42
F	Full	65-74	3	35	48
F	unemp	20-24	3	41	43
M	Full	75+	3	17	24
M	Part	75+	3	29	35
M	unemp	65-74	3	21	28
F	afw	15-19	2	13	17
F	Part	75+	2	24	26
F	unemp	35-44	2	45	52

---

sex	employment	age	freq overlap	freq masked	freq ident
F	unemp	55-64	2	31	34
M	afw	15-19	2	23	23
M	afw	20-24	2	24	21
F	afw	20-24	1	34	29
F	Full	75+	1	9	9
M	afw	45-54	1	35	31
M	afw	55-64	1	33	35
M	afw	65-74	1	21	22
M	afw	75+	1	5	3

---



## D. Simulation Studies for Parameter Choices of Masking Methods

### D.1. Parameter Choice for Anonymization via Lipschitz Embedding

Table D.1.: Precision and recall for simulation study for parameter choice for the anonymization via Lipschitz embedding masking method.

d	k	$\alpha = 0.1$		$\alpha = 0.5$		$\alpha = 0.9$	
		precision	recall	precision	recall	precision	recall
20	5	0.183	0.028	0.290	0.060	0.520	0.130
20	10	0.233	0.035	0.050	0.010	0.203	0.055
20	20	0.250	0.050	0.088	0.018	0.136	0.043
20	30	0.025	0.005	0.040	0.010	0.128	0.045
60	5	0.458	0.090	0.030	0.008	0.000	0.000
60	10	0.088	0.018	0.200	0.050	0.175	0.053
60	20	0.000	0.000	0.000	0.000	0.030	0.008
60	30	0.000	0.000	0.210	0.053	0.097	0.033
100	5	0.000	0.000	0.000	0.000	0.200	0.060
100	10	0.000	0.000	0.000	0.000	0.100	0.030
100	20	0.000	0.000	0.112	0.018	0.040	0.010
100	30	0.000	0.000	0.010	0.003	0.008	0.003

## D.2. Parameter Choice for Distance Approximation Using Intersecting Sets of Grid Points

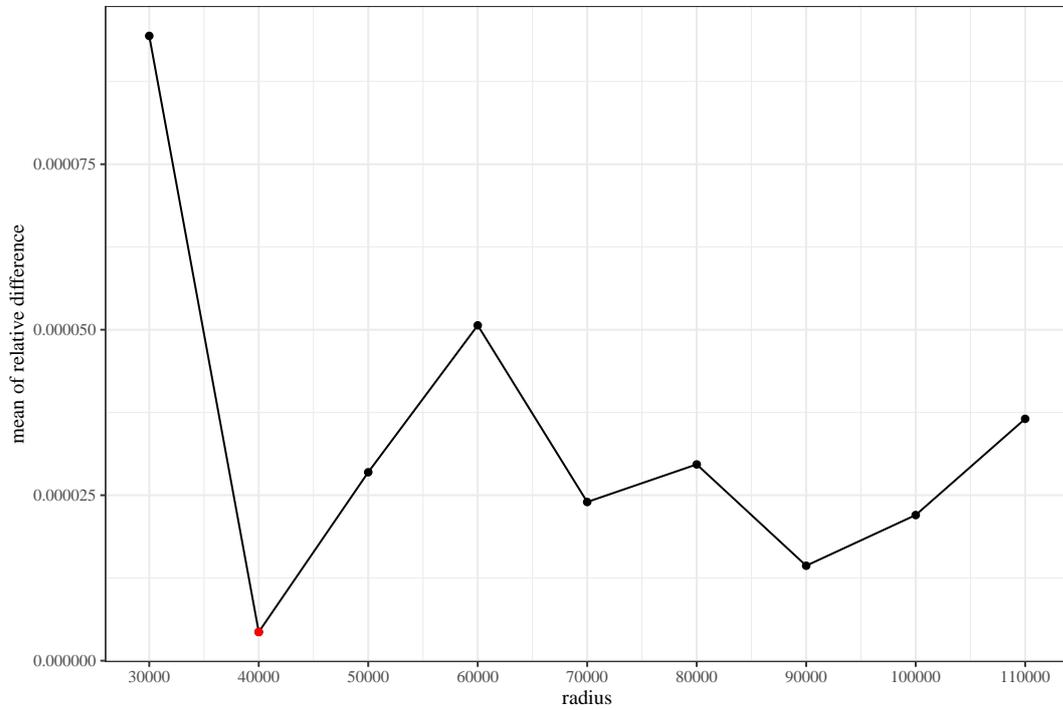


Figure D.1.: Mean of relative difference if 5,000,000 grid points are used. Red dot indicates minimum, located at  $r = 40,000$ .

## E. Execution Time Masking Methods

The average time is calculated as the arithmetic mean of calculated times over each of the 50 implementations. Not included in the calculation is loading data sets (unless additional data sets are required in the geomasking method itself) or, e.g., saving data sets, renaming of columns.

Table E.1.: Execution time of masking method applications (in minutes).

methods	average time	sd	methods	average time	sd
APA <sup>a</sup>	0.14	0.00	RandProj 200 <sup>a</sup>	0.01	0.00
APA LGA <sup>a</sup>	0.37	0.00	RandProj 300 <sup>a</sup>	0.01	0.00
ARP <sup>a</sup>	0.28	0.01	RandProj 500 <sup>a</sup>	0.03	0.00
ARP LGA <sup>a</sup>	1.24	0.01	RandProj 1000 <sup>a</sup>	0.05	0.01
CS <sup>a</sup>	0.00	0.00	Rot <sup>a</sup>	0.00	0.00
DD 3 <sup>a</sup>	0.00	0.00	RotArb <sup>a</sup>	0.00	0.00
DD 4 <sup>a</sup>	0.00	0.00	RPC 3 <sup>a</sup>	0.00	0.00
DD 5 <sup>a</sup>	0.00	0.00	RPC 4 <sup>a</sup>	0.00	0.00
DD LGA 3 <sup>a</sup>	0.00	0.00	RPC 5 <sup>a</sup>	0.00	0.00
DD LGA 4 <sup>a</sup>	0.00	0.00	RPC LGA 3 <sup>a</sup>	0.00	0.00
DD LGA 5 <sup>a</sup>	0.00	0.00	RPC LGA 4 <sup>a</sup>	0.00	0.00
Dk 5 <sup>a</sup>	14.11	0.00	RPC LGA 5 <sup>a</sup>	0.00	0.00
Dk 25 <sup>a</sup>	14.18	0.00	RPN <sup>a</sup>	0.00	0.00
Dk 50 <sup>a</sup>	32.78	0.00	RPN LGA <sup>a</sup>	0.00	0.00
Dk 100 <sup>a</sup>	34.47	0.00	RPU 3 <sup>a</sup>	0.00	0.00
Dk 500 <sup>a</sup>	35.03	0.00	RPU 4 <sup>a</sup>	0.00	0.00
Dk 1000 <sup>a</sup>	34.13	0.00	RPU 5 <sup>a</sup>	0.00	0.00
DkData 20 <sup>a</sup>	0.50	0.00	RPU LGA 3 <sup>a</sup>	0.00	0.00
DUT <sup>a</sup>	0.00	0.00	RPU LGA 4 <sup>a</sup>	0.00	0.00
Grid 100 <sup>a</sup>	0.00	0.00	RPU LGA 5 <sup>a</sup>	0.00	0.00
Grid 1000 <sup>a</sup>	0.00	0.00	StreetMask 30 <sup>a</sup>	39.45	0.49
ISGP <sup>b</sup>	709.02	12.12	StreetMask 100 <sup>a</sup>	40.18	0.71
Lipschitz Embedding <sup>a</sup>	0.43	0.01	VNE 50 3 <sup>b</sup>	10.62	0.03
LS 3 <sup>b</sup>	27.70	0.11	VNE 50 5 <sup>b</sup>	10.59	0.02
LS 4 <sup>b</sup>	27.71	0.12	VNE 100 3 <sup>b</sup>	10.64	0.03
LS 5 <sup>b</sup>	27.73	0.12	VNE 100 5 <sup>b</sup>	10.62	0.02
LS LGA 3 <sup>b</sup>	27.48	0.11	VNS 50 3 <sup>a</sup>	37.63	0.09
LS LGA 4 <sup>b</sup>	27.29	0.11	VNS 50 5 <sup>a</sup>	34.84	0.07
LS LGA 5 <sup>b</sup>	27.23	0.11	VNS 100 3 <sup>a</sup>	39.33	0.12
LSdonut 3 <sup>b</sup>	28.89	0.13	VNS 100 5 <sup>a</sup>	37.58	0.08
LSdonut 4 <sup>b</sup>	29.31	0.10	VNE LGA 50 3 <sup>b</sup>	10.64	0.03

methods	average time	sd	methods	average time	sd
LSdonut 5 <sup>b</sup>	29.70	0.10	VNE LGA 50 5 <sup>b</sup>	10.59	0.03
LSdonut LGA 3 <sup>b</sup>	28.77	0.10	VNE LGA 100 3 <sup>b</sup>	10.67	0.03
LSdonut LGA 4 <sup>b</sup>	29.12	0.08	VNE LGA 100 5 <sup>b</sup>	10.64	0.02
LSdonut LGA 5 <sup>b</sup>	29.57	0.16	VNS LGA 50 3 <sup>b</sup>	17.33	0.06
MDAV 3 <sup>a</sup>	0.01	0.00	VNS LGA 50 5 <sup>b</sup>	17.22	0.08
MDAV 25 <sup>a</sup>	0.00	0.00	VNS LGA 100 3 <sup>b</sup>	17.42	0.08
MDAV 50 <sup>a</sup>	0.00	0.00	VNS LGA 100 5 <sup>b</sup>	17.35	0.11
RandProj 100 <sup>a</sup>	0.00	0.00	Voronoi <sup>a</sup>	1.69	0.38

<sup>a</sup> Windows 10 Pro, Intel(R) Core(TM) i7-7600 CPU@2.80 GHz, 2904 MHz, 2 cores, 4 threads; 16 GB RAM

<sup>b</sup> Ubuntu 18.04.4 LTS, Intel(R) Core(TM) i7-4930K CPU @ 3.406 GHz x 12

## F. Execution Times Risk Analysis

Table F.1.: Average execution times of risk methods for one replication (in minutes).

method sample	mean <sup>a</sup>			mindist <sup>a</sup>		Hungarian <sup>a</sup>		Hungarian blocking <sup>a</sup>		
	10,000	1,000	2,000	1,000	2,000	1,000	2,000	10,000		
APA	0.05	0.01	0.06	0.19	2.51	0.02	0.03	0.14		
APA LGA	0.04	0.01	0.05	0.16	1.90	0.02	0.02	0.12		
ARP	0.04	0.02	0.06	0.45	10.92	0.02	0.02	0.25		
ARP LGA	0.04	0.02	0.06	0.34	9.58	0.02	0.02	0.22		
DD 3	0.04	0.02	0.06	0.22	6.91	0.02	0.02	0.17		
DD 4	0.04	0.02	0.06	0.22	6.78	0.02	0.02	0.17		
DD 5	0.04	0.02	0.06	0.22	6.60	0.02	0.02	0.17		
DD LGA 3	0.04	0.02	0.06	0.20	5.85	0.02	0.02	0.17		
DD LGA 4	0.04	0.02	0.06	0.20	5.65	0.02	0.02	0.16		
DD LGA 5	0.04	0.02	0.06	0.20	5.92	0.02	0.02	0.17		
Dk 5	0.04	0.02	0.06	0.22	6.43	0.02	0.02	0.18		
Dk 25	0.04	0.02	0.06	0.22	8.50	0.02	0.02	0.18		
Dk 50	0.04	0.02	0.03	0.21	6.00	0.02	0.02	0.17		
Dk 100	0.04	0.02	0.03	0.22	5.71	0.02	0.02	0.17		
Dk 500	0.04	0.02	0.03	0.21	5.69	0.02	0.02	0.16		
Dk 1000	0.04	0.02	0.03	0.21	7.61	0.02	0.02	0.16		
DkData 20	0.04	0.02	0.06	0.20	5.67	0.02	0.02	0.17		
Grid 100	0.16	0.01	0.05	0.30	7.60	0.03	0.03	0.27		
Grid 1000	0.15	0.01	0.05	0.25	4.36	0.03	0.03	0.25		
LS 3	0.04	0.02	0.03	0.22	8.19	0.02	0.02	0.17		
LS 4	0.04	0.02	0.03	0.22	8.20	0.02	0.02	0.17		
LS 5	0.04	0.02	0.03	0.22	8.05	0.02	0.02	0.17		
LS LGA 3	0.04	0.02	0.03	0.20	7.61	0.02	0.02	0.17		
LS LGA 4	0.04	0.02	0.03	0.21	7.53	0.02	0.02	0.17		
LS LGA 5	0.04	0.02	0.03	0.21	7.54	0.02	0.02	0.17		
LSdonut 3	0.04	0.02	0.03	0.21	7.24	0.02	0.02	0.17		
LSdonut 4	0.04	0.02	0.03	0.21	7.33	0.02	0.02	0.17		
LSdonut 5	0.04	0.02	0.03	0.21	7.58	0.02	0.02	0.17		
LSdonut LGA 3	0.04	0.02	0.03	0.21	7.67	0.02	0.02	0.17		
LSdonut LGA 4	0.04	0.02	0.03	0.21	7.33	0.02	0.02	0.17		
LSdonut LGA 5	0.04	0.02	0.03	0.21	7.33	0.02	0.02	0.17		
MdAV 3	0.08	0.01	0.03	0.20	5.10	0.02	0.02	0.17		
MdAV 25	0.10	0.01	0.04	0.17	3.31	0.02	0.02	0.15		
MdAV 50	0.11	0.01	0.03	0.17	2.85	0.02	0.02	0.14		

method	mean	mindist		Hungarian		Hungarian blocking		
		1,000	2,000	1,000	2,000	1,000	2,000	10,000
RPC 3	0.04	0.02	0.06	0.22	7.96	0.02	0.02	0.17
RPC 4	0.04	0.02	0.03	0.22	7.98	0.02	0.02	0.17
RPC 5	0.04	0.02	0.03	0.22	7.79	0.02	0.02	0.17
RPC LGA 3	0.04	0.02	0.03	0.21	7.19	0.02	0.02	0.17
RPC LGA 4	0.04	0.02	0.03	0.20	7.34	0.02	0.02	0.17
RPC LGA 5	0.04	0.02	0.03	0.20	7.34	0.02	0.02	0.16
RPN	0.04	0.02	0.03	0.32	7.46	0.02	0.02	0.19
RPN LGA	0.04	0.02	0.03	0.21	7.63	0.02	0.02	0.18
RPU 3	0.04	0.02	0.03	0.20	7.36	0.02	0.02	0.17
RPU 4	0.04	0.02	0.03	0.22	8.22	0.02	0.02	0.18
RPU 5	0.04	0.02	0.03	0.21	7.89	0.02	0.02	0.17
RPU LGA 3	0.04	0.02	0.03	0.21	7.30	0.02	0.02	0.17
RPU LGA 4	0.04	0.02	0.03	0.21	7.40	0.02	0.02	0.17
RPU LGA 5	0.04	0.01	0.03	0.21	5.76	0.02	0.02	0.17
VNE 50 3	0.04	0.02	0.03	0.21	8.03	0.02	0.02	0.17
VNE 50 5	0.04	0.02	0.03	0.23	8.00	0.02	0.02	0.17
VNE 100 3	0.04	0.02	0.03	0.22	7.34	0.02	0.02	0.17
VNE 100 5	0.04	0.02	0.03	0.22	7.33	0.02	0.02	0.17
VNS 50 3	0.04	0.02	0.03	0.21	7.25	0.02	0.02	0.17
VNS 50 5	0.04	0.02	0.03	0.21	7.20	0.02	0.02	0.17
VNS 100 3	0.04	0.02	0.03	0.21	7.32	0.02	0.02	0.17
VNS 100 5	0.04	0.02	0.03	0.21	7.24	0.02	0.02	0.17
VNE LGA 50 3	0.04	0.02	0.03	0.26	7.67	0.02	0.02	0.17
VNE LGA 50 5	0.04	0.02	0.03	0.20	7.63	0.02	0.02	0.17
VNE LGA 100 3	0.04	0.02	0.03	0.21	7.35	0.02	0.02	0.17
VNE LGA 100 5	0.04	0.02	0.03	0.21	7.38	0.02	0.02	0.17
VNS LGA 50 3	0.04	0.02	0.03	0.21	7.23	0.02	0.02	0.17
VNS LGA 50 5	0.04	0.02	0.03	0.49	7.72	0.02	0.02	0.17
VNS LGA 100 3	0.04	0.02	0.03	0.21	7.59	0.29	0.02	0.17
VNS LGA 100 5	0.04	0.02	0.03	0.21	8.17	0.02	0.02	0.17
Voronoi	0.04	0.02	0.03	0.22	7.29	0.02	0.02	0.18

<sup>a</sup> calculated on Lenovo T470, Windows 10 Pro, Intel(R) Core(TM) i7-7600 CPU @ 2.80 GHz, 2904 MHz, 2 cores, 4 threads; 16 GB RAM

<sup>b</sup> calculated on Dell, Windows 10 Education, Intel(R) Core(TM) i7-7700 CPU @ 3.60 GHz

<sup>c</sup> *a* and *b* used

Table F.2.: Execution times of risk methods for each iteration (in minutes).

method	graph <sup>c</sup>		ppr1 <sup>b</sup>	
	sample	1,000	1,000	2,000
APA		1.63	2.27	14.34
APA LGA		2.36	1.91	13.20
ARP		1.66	1.80	11.27
ARP LGA		2.21	1.82	11.91

method	graph	pprl	
method	1,000	1,000	2,000
DD 3	4.76	1.98	13.62
DD 4	4.85	1.96	13.61
DD 5	4.91	1.94	13.69
DD LGA 3	4.77	1.96	13.61
DD LGA 4	4.89	1.98	13.53
DD LGA 5	4.89	1.91	13.62
Dk 5	5.86	1.93	13.47
Dk 25	5.86	1.94	13.49
Dk 50	6.02	1.96	13.45
Dk 100	6.12	1.94	13.48
Dk 500	6.78	1.94	13.53
Dk 1000	7.98	1.93	13.57
DkData 20	7.56	1.95	13.63
Grid 100	4.61	1.95	13.03
Grid 1000	5.42	1.96	12.72
ISGP	39.46	2.00	12.74
Lipschitz Embedding	8.59	2.47	13.77
LS 3	4.55	1.96	13.30
LS 4	4.65	1.97	13.32
LS 5	4.70	1.96	13.38
LS LGA 3	4.57	1.92	13.36
LS LGA 4	4.70	1.93	13.25
LS LGA 5	4.81	1.94	13.02
LSdonut 3	4.53	1.93	13.00
LSdonut 4	4.64	1.95	13.15
LSdonut 5	4.68	1.94	13.00
LSdonut LGA 3	4.56	1.94	13.06
LSdonut LGA 4	4.74	1.93	13.11
LSdonut LGA 5	4.83	1.92	12.96
MDAV 3	12.60	1.93	13.22
MDAV 25	6.30	1.91	13.04
MDAV 50	12.43	1.96	13.30
RandProj 100		2.57	13.92
RandProj 200		3.24	17.73
RandProj 300		2.97	16.76
RandProj 500		3.03	18.03
RandProj 1000		2.70	15.09
RPC 3	4.70	1.97	13.35
RPC 4	4.80	1.92	13.09
RPC 5	4.90	1.95	13.09
RPC LGA 3	4.78	1.94	13.12
RPC LGA 4	4.90	1.95	12.88
RPC LGA 5	5.08	1.96	12.84

method	graph	pprl	
method	1,000	1,000	2,000
RPN	4.53	1.98	13.02
RPN LGA	4.56	1.97	12.99
RPU 3	4.86	1.93	12.82
RPU 4	4.95	1.94	13.01
RPU 5	5.11	1.97	12.73
RPU LGA 3	4.88	1.98	12.96
RPU LGA 4	5.03	1.96	13.13
RPU LGA 5	5.44	1.95	13.26
VNE 50 3	4.54	1.92	12.67
VNE 50 5	4.70	1.87	12.71
VNE 100 3	4.51	1.87	12.68
VNE 100 5	4.68	1.90	12.90
VNS 50 3	4.56	1.92	12.92
VNS 50 5	4.70	1.93	12.90
VNS 100 3	4.54	1.92	13.03
VNS 100 5	4.68	1.95	13.02
VNE LGA 50 3	4.54	1.94	12.74
VNE LGA 50 5	4.76	1.94	12.77
VNE LGA 100 3	4.54	1.93	12.99
VNE LGA 100 5	4.73	1.93	12.96
VNS LGA 50 3	4.58	1.95	12.94
VNS LGA 50 5	4.74	1.96	12.94
VNS LGA 100 3	4.58	1.93	13.27
VNS LGA 100 5	4.74	3.14	13.29
Voronoi	35.93	1.95	12.89

<sup>a</sup> calculated on Lenovo T470, Windows 10 Pro, Intel(R) Core(TM) i7-7600 CPU @ 2.80 GHz, 2904 MHz, 2 cores, 4 threads; 16 GB RAM

<sup>b</sup> calculated on Dell, Windows 10 Education, Intel(R) Core(TM) i7-7700 CPU @ 3.60 GHz

<sup>c</sup> *a* and *b* used

DUT <sup>a</sup>			CS <sup>a</sup>		
1,000	2,000	10,000	1,000	2,000	10,000
0.09	0.73	3.42	0.93	1.62	6.88

Rot <sup>a</sup>			RotArb <sup>a</sup>		
1,000	2,000	10,000	1,000	2,000	10,000
0.04	0.07	0.27	0.03	0.06	0.22

<sup>a</sup> calculated on Lenovo T470, Windows 10 Pro, Intel(R) Core(TM) i7-7600 CPU @ 2.80 GHz, 2904 MHz, 2 cores, 4 threads; 16 GB RAM

# G. Detailed Results of the Risk-Utility Analysis

## G.1. Spatial Mean Center

Table G.1.: Spatial mean (mean center) comparison of masked data with original data (in meter).

masking method	nearest mean center	furthest mean center	mean distance from orig. center	sd of distances from orig. center
APA	22,795.828	22,795.828	22,795.828	0.000
APA LGA	582.431	582.431	582.431	0.000
ARP	22,208.102	25,259.830	23,481.426	655.424
ARP LGA	216.510	1088.115	746.885	184.716
CS	116,797.083	2,050,727.804	1,169,524.707	640,413.332
DD 3	1.983	44.322	18.639	10.391
DD 4	3.348	77.337	25.114	15.410
DD 5	1.942	63.828	33.774	14.983
DD LGA 3	1.765	42.950	19.963	10.616
DD LGA 4	2.236	65.320	28.769	13.327
DD LGA 5	7.151	72.586	32.867	17.380
Dk 5	0.047	0.911	0.390	0.222
Dk 25	0.035	1.753	0.640	0.421
Dk 50	2.661	29.121	14.350	7.083
Dk 100	1.154	36.202	17.967	9.299
Dk 500	14.126	132.501	55.532	27.276
Dk 1000	12.906	257.999	103.237	53.237
DkData 20	9.999	228.957	96.193	54.945
DUT	1,717.448	12,501.972	7,701.913	2,751.240
LS 3	0.909	13.149	6.367	3.032
LS 4	0.390	19.672	7.685	4.059
LS 5	10.353	45.931	25.074	8.207
LS LGA 3	11.225	31.937	20.791	5.624
LS LGA 4	15.465	36.640	26.159	5.665
LS LGA 5	7.208	47.225	25.995	7.431
LSdonut 3	13.351	28.086	18.596	2.853
LSdonut 4	18.931	38.663	29.442	4.402
LSdonut 5	83.523	119.067	101.516	9.122
LSdonut LGA 3	23.269	42.343	29.855	3.919
LSdonut LGA 4	19.998	38.789	30.408	4.130
LSdonut LGA 5	17.503	70.877	44.189	10.341
MDAV 3	0.000	0.000	0.000	0.000

masking method	nearest mean center	furthest mean center	mean distance from orig. center	sd of distances from orig. center
MDAV 25	0.000	0.000	0.000	0.000
MDAV 50	0.000	0.000	0.000	0.000
Grid 100	0.119	0.119	0.119	0.000
Grid 1000	6.125	6.125	6.125	0.000
Rot	37,408.919	4,071,434.969	2,545,551.791	1,190,579.250
RotArb	0.000	0.000	0.000	0.000
RPC 3	3.402	34.291	15.425	7.503
RPC 4	2.336	47.447	19.045	10.992
RPC 5	3.272	68.432	25.021	15.998
RPC LGA 3	0.712	35.354	16.277	8.276
RPC LGA 4	3.581	36.448	17.198	8.996
RPC LGA 5	1.226	53.774	25.849	13.508
RPN	0.878	26.460	10.520	6.143
RPN LGA	2.809	35.550	12.492	7.240
RPU 3	2.094	74.222	26.253	15.072
RPU 4	3.582	88.407	29.379	15.887
RPU 5	10.159	83.670	38.679	16.794
RPU LGA 3	3.078	65.652	26.915	15.576
RPU LGA 4	2.224	80.786	32.978	18.784
RPU LGA 5	4.106	97.108	43.154	20.727
StreetMask 30	235.781	243.777	239.513	1.696
StreetMask 100	228.739	259.224	248.477	6.048
VNE 50 3	11.221	74.938	48.103	15.888
VNE 50 5	13.730	101.695	59.876	20.132
VNE 100 3	22.382	178.939	85.341	36.551
VNE 100 5	18.678	183.063	81.275	34.293
VNS 50 3	5.739	44.926	25.672	10.922
VNS 50 5	8.049	75.905	41.062	15.139
VNS 100 3	6.020	92.559	45.597	23.629
VNS 100 5	3.538	113.827	48.664	27.956
VNE LGA 50 3	3.539	84.850	48.792	18.154
VNE LGA 50 5	15.402	106.711	61.092	19.640
VNE LGA 100 3	23.323	130.646	73.906	32.723
VNE LGA 100 5	16.424	170.946	89.589	41.203
VNS LGA 50 3	12.936	52.233	34.457	9.667
VNS LGA 50 5	10.720	67.056	39.454	14.674
VNS LGA 100 3	8.254	92.990	47.245	21.090
VNS LGA 100 5	8.749	95.327	52.455	26.425
Voronoi	26.270	26.270	26.270	0.000
Grid 100	0.1195	0.1195	0.1195	0
Grid 1000	6.1249	6.1249	6.1249	0
ISGP	0.0000	0.0000	0.0000	0.0000
Lipschitz Embedding	0.0000	0.0000	0.0000	0.0000

## G.2. Spatial Median Center

Table G.2.: Spatial median (median center) comparison of masked data with original data (in meter).

masking method	nearest median center	furthest median center	mean distance from orig. center	sd of distances from orig. center
APA	803.929	803.929	803.929	0.000
APA LGA	1,020.215	1,020.215	1,020.215	0.000
ARP	43.579	339.880	175.156	74.429
ARP LGA	719.148	1,025.093	894.212	75.601
CS	116,660.105	2,048,656.486	1,168,196.010	639,712.154
DD 3	2.272	44.681	20.245	9.465
DD 4	1.792	62.914	27.000	12.613
DD 5	1.781	40.796	21.834	9.944
DD LGA 3	5.381	42.596	21.505	9.757
DD LGA 4	1.830	39.251	20.360	9.522
DD LGA 5	6.248	49.563	25.638	10.575
Dk 5	1.647	18.912	9.023	4.341
Dk 25	1.833	29.882	12.463	6.147
Dk 50	1.174	42.390	16.265	8.659
Dk 100	4.006	40.171	18.878	8.095
Dk 500	5.267	85.359	36.951	17.805
Dk 1000	2.689	76.069	32.120	16.563
DkData 20	7.491	93.230	35.628	17.738
DUT	1,717.350	12,501.320	7,701.502	2,751.094
LS 3	4.926	26.249	12.480	4.500
LS 4	1.879	29.652	13.379	6.506
LS 5	1.991	36.115	17.231	8.338
LS LGA 3	1.256	28.034	11.346	5.846
LS LGA 4	0.982	31.107	12.674	7.078
LS LGA 5	2.844	37.463	19.846	8.159
LSdonut 3	1.949	27.793	14.555	5.259
LSdonut 4	1.494	29.611	13.156	6.478
LSdonut 5	3.114	49.482	23.735	9.457
LSdonut LGA 3	2.992	35.596	16.321	6.917
LSdonut LGA 4	1.541	37.061	15.782	8.281
LSdonut LGA 5	1.146	46.309	24.912	9.721
MDAV 3	27.110	27.110	27.110	0.000
MDAV 25	57.343	57.343	57.343	0.000
MDAV 50	132.719	132.719	132.719	0.000
Rot	37,245.877	4,066,871.157	2,542,308.904	1,189,252.760
RotArb	189.025	15,593.116	10,384.648	4,516.957
RPC 3	1.256	34.316	12.731	6.604
RPC 4	1.188	42.959	16.248	8.949
RPC 5	3.951	39.815	23.112	8.614
RPC LGA 3	1.714	33.407	15.231	8.052
RPC LGA 4	1.589	32.809	16.451	8.017
RPC LGA 5	3.802	39.982	23.086	9.488
RPN	2.568	30.753	11.591	5.920

masking method	nearest	furthest	mean distance	sd of distances
	median center	median center	from orig. center	from orig. center
RPN LGA	2.123	32.426	14.409	7.674
RPU 3	1.941	65.100	30.677	14.271
RPU 4	4.131	69.915	35.045	12.945
RPU 5	11.456	59.842	32.990	12.684
RPU LGA 3	3.666	49.369	26.290	10.790
RPU LGA 4	11.475	51.676	27.689	9.182
RPU LGA 5	4.924	60.485	26.059	11.470
StreetMask 30	4.038	61.547	32.640	13.133
StreetMask 100	8.160	62.826	32.420	12.871
VNE 50 3	3.336	46.242	20.221	11.424
VNE 50 5	2.334	56.470	21.052	11.315
VNE 100 3	2.725	60.978	24.373	15.072
VNE 100 5	2.725	68.322	27.318	14.600
VNS 50 3	0.292	38.153	17.634	8.821
VNS 50 5	3.584	33.356	17.486	7.151
VNS 100 3	2.344	40.938	20.554	8.308
VNS 100 5	2.298	47.601	22.616	11.326
VNE LGA 50 3	1.680	48.997	20.376	11.256
VNE LGA 50 5	2.672	60.389	20.281	11.805
VNE LGA 100 3	1.474	58.566	25.908	13.510
VNE LGA 100 5	3.773	64.759	28.542	14.668
VNS LGA 50 3	3.563	35.322	17.098	7.742
VNS LGA 50 5	1.266	48.250	18.008	11.045
VNS LGA 100 3	2.031	51.793	21.474	10.883
VNS LGA 100 5	3.310	62.795	25.478	13.239
Voronoi	10.117	10.117	10.117	0.000
Grid 100	50.771	50.771	50.771	0.000
Grid 1000	502.036	502.036	502.036	0.000
ISGP	4,257.098	4,271.724	4,265.419	3.454
Lipschitz Embedding	3,390.298	22,739.070	10,019.750	5,216.632

### G.3. Standard Distance

Table G.3.: Standard distance of the masked coordinates (in meter).

masking method	minimum	maximum	mean	sd
	stand. dist.	stand. dist.	stand. dist.	stand. dist.
APA	190,671.557	190,671.557	190,671.557	0.000
APA LGA	124,102.710	124,102.710	124,102.710	0.000
ARP	204,328.203	213,269.774	207,487.943	1,870.759
ARP LGA	125,362.841	128,210.962	126,784.691	661.985
CS	3,946.528	243,014.100	119,287.196	78,129.859
DD 3	123,837.842	124,237.855	124,056.581	77.120
DD 4	123,897.843	124,233.759	124,062.496	89.608

masking method	minimum stand. dist.	maximum stand. dist.	mean stand. dist.	sd stand. dist.
DD 5	123,677.234	124,292.046	124,067.908	113.884
DD LGA 3	123,846.018	124,198.936	124,038.825	86.139
DD LGA 4	123,728.956	124,283.870	124,069.303	111.631
DD LGA 5	123,840.130	124,424.224	124,067.368	125.755
Dk 5	124,025.442	124,027.802	124,026.640	0.570
Dk 25	124,025.017	124,028.318	124,026.854	0.830
Dk 50	123,872.843	124,245.083	124,028.242	76.437
Dk 100	123,872.313	124,243.201	124,052.810	92.739
Dk 500	123,566.164	124,834.522	124,174.546	303.774
Dk 1000	123,339.710	125,712.609	124,406.081	549.969
DkData 20	123,392.256	126,061.492	124,513.468	618.802
DUT	124,026.651	124,026.651	124,026.651	0.000
LS 3	123,992.408	124,066.504	124,023.742	18.489
LS 4	123,991.530	124,068.753	124,026.543	17.790
LS 5	123,849.172	124,043.316	123,945.033	42.702
LS LGA 3	123,976.764	124,039.631	124,003.019	16.191
LS LGA 4	123,953.318	124,033.356	123,987.094	15.672
LS LGA 5	123,872.424	124,005.400	123,938.148	27.117
LSdonut 3	123,976.402	124,061.442	124,020.778	19.023
LSdonut 4	123,812.650	123,904.808	123,851.688	21.418
LSdonut 5	123,494.541	123,613.032	123,543.739	32.512
LSdonut LGA 3	123,942.675	124,064.231	124,000.763	22.507
LSdonut LGA 4	123,980.603	124,032.155	124,000.779	13.323
LSdonut LGA 5	123,787.579	123,927.740	123,863.388	32.187
MDAV 3	123,895.913	123,895.913	123,895.913	0.000
MDAV 25	122,656.260	122,656.260	122,656.260	0.000
MDAV 50	121,577.623	121,577.623	121,577.623	0.000
Rot	124,026.651	124,026.651	124,026.651	0.000
RotArb	124,026.651	124,026.651	124,026.651	0.000
RPC 3	123,947.534	124,180.906	124,050.180	55.108
RPC 4	123,904.378	124,209.658	124,052.530	62.539
RPC 5	123,865.658	124,276.460	124,051.214	98.732
RPC LGA 3	123,919.861	124,181.324	124,051.721	62.903
RPC LGA 4	123,864.256	124,235.343	124,036.634	83.516
RPC LGA 5	123,836.783	124,308.447	124,067.303	104.866
RPN	123,940.460	124,157.326	124,031.110	41.538
RPN LGA	123,953.393	124,160.633	124,036.541	47.067
RPU 3	123,899.447	124,382.458	124,076.694	98.363

masking method	minimum stand. dist.	maximum stand. dist.	mean stand. dist.	sd stand. dist.
RPU 4	123,720.163	124,409.628	124,081.085	127.841
RPU 5	123,847.759	124,433.426	124,083.875	125.646
RPU LGA 3	123,830.927	124,330.754	124,071.011	115.246
RPU LGA 4	123,761.416	124,420.422	124,099.641	160.104
RPU LGA 5	123,687.231	124,471.168	124,137.545	157.265
StreetMask 30	123,326.035	123,355.885	123,342.765	6.613
StreetMask 100	123,264.723	123,422.444	123,336.231	35.111
VNE 50 3	123,342.263	123,867.745	123,595.870	122.517
VNE 50 5	123,193.149	123,991.198	123,554.878	169.289
VNE 100 3	122,761.967	123,992.185	123,389.109	271.618
VNE 100 5	122,844.539	124,062.740	123,385.223	243.746
VNS 50 3	123,757.033	124,082.589	123,919.168	87.025
VNS 50 5	123,652.102	124,030.102	123,835.379	97.234
VNS 100 3	123,357.914	124,187.222	123,748.284	182.322
VNS 100 5	123,266.300	124,204.154	123,721.714	205.253
VNE LGA 50 3	123,314.332	123,870.717	123,584.931	124.735
VNE LGA 50 5	123,209.356	123,904.173	123,510.126	149.242
VNE LGA 100 3	122,775.495	123,990.913	123,389.472	273.916
VNE LGA 100 5	122,841.972	123,975.807	123,334.606	280.642
VNS LGA 50 3	123,656.093	124,050.947	123,910.880	84.711
VNS LGA 50 5	123,628.294	124,006.510	123,836.684	105.358
VNS LGA 100 3	123,356.482	124,128.432	123,734.245	189.980
VNS LGA 100 5	123,401.079	124,140.294	123,731.069	202.747
Voronoi	123,826.136	123,826.136	123,826.136	0.000
Grid 100	124,027.072	124,027.072	124,027.072	0.000
Grid 1000	124,025.621	124,025.621	124,025.621	0.000
ISGP	23,771.795	23,772.626	23,772.155	0.163
Lipschitz Embedding	85,420.858	118,163.534	107,974.079	7,947.098

#### G.4. Standard Deviational Ellipse

Table G.4.: Angle of rotation of the standard deviational ellipses of the masked coordinates (in degree).

masking method	minimum angle	maximum angle	mean angle	sd angle
APA	122.516	122.516	122.516	0.000

masking method	minimum angle	maximum angle	mean angle	sd angle
APA LGA	120.927	120.927	120.927	0.000
ARP	122.799	125.949	124.184	0.700
ARP LGA	119.706	122.138	120.740	0.583
CS	121.032	121.032	121.032	0.000
DD 3	120.935	121.142	121.023	0.048
DD 4	120.875	121.213	121.024	0.069
DD 5	120.796	121.227	121.036	0.087
DD LGA 3	120.906	121.161	121.036	0.053
DD LGA 4	120.867	121.180	121.042	0.071
DD LGA 5	120.875	121.208	121.028	0.085
Dk 5	121.032	121.033	121.032	0.000
Dk 25	121.031	121.034	121.032	0.001
Dk 50	120.914	121.178	121.026	0.061
Dk 100	120.919	121.204	121.046	0.065
Dk 500	120.632	121.700	121.037	0.208
Dk 1000	120.280	122.049	121.063	0.384
DkData 20	119.833	121.994	121.007	0.413
DUT	121.032	121.032	121.032	0.000
LS 3	121.025	121.072	121.048	0.012
LS 4	121.028	121.074	121.046	0.011
LS 5	120.988	121.096	121.050	0.022
LS LGA 3	121.064	121.119	121.093	0.014
LS LGA 4	121.076	121.130	121.099	0.013
LS LGA 5	121.059	121.127	121.091	0.018
LSdonut 3	120.893	120.947	120.927	0.012
LSdonut 4	120.938	120.981	120.960	0.012
LSdonut 5	120.938	121.010	120.970	0.021
LSdonut LGA 3	121.088	121.142	121.120	0.012
LSdonut LGA 4	121.071	121.114	121.093	0.009
LSdonut LGA 5	121.096	121.165	121.129	0.018
MDAV 3	121.072	121.072	121.072	0.000
MDAV 25	120.944	120.944	120.944	0.000
MDAV 50	121.438	121.438	121.438	0.000
Rot	0.032	178.032	81.192	55.055
RotArb	1.032	178.032	83.552	48.894
RPC 3	120.967	121.118	121.034	0.036
RPC 4	120.931	121.158	121.030	0.049
RPC 5	120.913	121.218	121.034	0.066

masking method	minimum angle	maximum angle	mean angle	sd angle
RPC LGA 3	120.949	121.116	121.032	0.037
RPC LGA 4	120.943	121.118	121.025	0.035
RPC LGA 5	120.906	121.206	121.038	0.064
RPN	120.971	121.094	121.028	0.029
RPN LGA	120.918	121.168	121.026	0.039
RPU 3	120.872	121.251	121.048	0.074
RPU 4	120.850	121.228	121.035	0.076
RPU 5	120.812	121.315	121.052	0.105
RPU LGA 3	120.812	121.212	121.033	0.087
RPU LGA 4	120.764	121.246	121.031	0.089
RPU LGA 5	120.805	121.284	121.034	0.114
StreetMask 30	120.700	120.715	120.709	0.003
StreetMask 100	120.663	120.756	120.701	0.018
VNE 50 3	120.597	120.969	120.799	0.083
VNE 50 5	120.610	121.127	120.797	0.118
VNE 100 3	120.437	121.095	120.750	0.153
VNE 100 5	120.513	121.082	120.739	0.130
VNS 50 3	120.884	121.071	120.989	0.046
VNS 50 5	120.838	121.082	120.992	0.059
VNS 100 3	120.774	121.101	120.909	0.085
VNS 100 5	120.721	121.132	120.924	0.094
VNE LGA 50 3	120.659	121.091	120.844	0.100
VNE LGA 50 5	120.611	121.090	120.823	0.108
VNE LGA 100 3	120.530	121.160	120.797	0.138
VNE LGA 100 5	120.542	121.043	120.800	0.117
VNS LGA 50 3	120.903	121.120	121.043	0.047
VNS LGA 50 5	120.878	121.117	121.021	0.055
VNS LGA 100 3	120.765	121.140	120.975	0.088
VNS LGA 100 5	120.866	121.136	120.993	0.076
Voronoi	120.999	120.999	120.999	0.000
Grid 100	121.033	121.033	121.033	0.000
Grid 1000	121.030	121.030	121.030	0.000
ISGP	148.631	148.635	148.633	0.001
Lipschitz Embedding	100.166	131.733	118.021	5.080

Table G.5.: Difference in the length of the major axis (x-axis) of the standard deviational ellipse of the original coordinates and of the masked coordinates (in meter).

masking method	minimum diff. major axis	maximum diff. major axis	mean diff. major axis	sd diff. major axis
APA	60,274.403	60,274.403	60,274.403	0.000
APA LGA	586.212	586.212	586.212	0.000
ARP	70,850.126	79,468.210	73,870.999	1,953.226
ARP LGA	705.671	3,495.888	1,996.337	631.975
CS	-110,745.115	109,737.384	-4,371.011	72,056.057
DD 3	-203.192	203.383	26.533	80.027
DD 4	-148.550	191.954	22.572	87.508
DD 5	-353.055	260.920	20.180	122.790
DD LGA 3	-174.859	187.554	2.030	89.268
DD LGA 4	-289.298	267.492	29.782	121.088
DD LGA 5	-268.511	460.982	27.529	138.955
Dk 5	-1.090	1.123	0.007	0.539
Dk 25	-1.455	1.861	0.231	0.844
Dk 50	-180.341	218.027	0.087	76.918
Dk 100	-172.043	223.963	13.956	95.101
Dk 500	-656.749	906.906	89.007	338.287
Dk 1000	-1,090.851	1,448.335	149.683	575.350
DkData 20	-1,015.728	1,918.322	296.062	649.775
DUT	0.000	0.000	0.000	0.000
LS 3	-29.243	43.704	1.898	18.007
LS 4	-28.425	44.321	4.721	17.218
LS 5	-171.333	15.483	-72.778	43.028
LS LGA 3	-24.141	32.260	4.024	13.036
LS LGA 4	-37.868	47.773	0.896	17.719
LS LGA 5	-98.442	23.731	-37.855	24.035
LSdonut 3	-29.153	42.960	8.340	16.032
LSdonut 4	-167.634	-97.755	-141.262	17.244
LSdonut 5	-515.727	-380.640	-455.261	35.900
LSdonut LGA 3	-30.924	64.760	14.065	18.084
LSdonut LGA 4	9.792	54.417	30.186	11.806
LSdonut LGA 5	-144.739	-39.704	-87.698	27.484
MDAV 3	-66.388	-66.388	-66.388	0.000
MDAV 25	-1,066.064	-1,066.064	-1,066.064	0.000
MDAV 50	-1,738.364	-1,738.364	-1,738.364	0.000
Rot	-0.000	0.000	0.000	0.000
RotArb	0.000	0.000	0.000	0.000
RPC 3	-76.501	178.268	17.610	56.459
RPC 4	-119.484	195.334	19.632	63.893
RPC 5	-153.370	242.938	18.284	99.934
RPC LGA 3	-136.698	186.355	16.225	67.059
RPC LGA 4	-189.650	214.319	0.686	86.766

masking method	minimum diff. major axis	maximum diff. major axis	mean diff. major axis	sd diff. major axis
RPC LGA 5	-165.138	247.685	16.030	103.130
RPN	-85.799	127.731	1.688	43.815
RPN LGA	-90.631	140.983	5.341	48.075
RPU 3	-198.379	392.553	27.225	111.840
RPU 4	-231.103	346.997	28.086	122.228
RPU 5	-197.224	278.373	36.845	129.619
RPU LGA 3	-208.138	263.083	26.882	116.729
RPU LGA 4	-315.908	489.376	48.026	171.828
RPU LGA 5	-405.101	411.896	65.061	163.323
StreetMask 30	-504.829	-478.202	-489.614	5.976
StreetMask 100	-548.173	-379.907	-484.567	35.566
VNE 50 3	-628.577	-77.499	-374.794	124.228
VNE 50 5	-733.707	9.569	-411.159	162.663
VNE 100 3	-1,160.606	76.352	-551.240	284.652
VNE 100 5	-1,120.902	227.455	-545.278	265.936
VNS 50 3	-258.860	51.740	-84.838	97.645
VNS 50 5	-360.817	37.952	-182.213	109.726
VNS 100 3	-657.809	160.753	-265.628	180.710
VNS 100 5	-713.609	226.942	-254.429	211.546
VNE LGA 50 3	-650.805	-76.903	-369.564	130.273
VNE LGA 50 5	-712.336	-39.309	-432.489	145.885
VNE LGA 100 3	-1,286.776	46.567	-533.525	294.650
VNE LGA 100 5	-1,121.482	95.894	-582.860	302.116
VNS LGA 50 3	-326.992	48.283	-81.105	89.471
VNS LGA 50 5	-355.581	27.912	-143.950	105.253
VNS LGA 100 3	-597.849	204.887	-234.369	210.015
VNS LGA 100 5	-569.553	203.553	-234.119	221.879
Voronoi	-150.678	-150.678	-150.678	0.000
Grid 100	0.491	0.491	0.491	0.000
Grid 1000	0.866	0.866	0.866	0.000
ISGP	-95,119.630	-95,118.983	-95,119.277	0.151
Lipschitz Embedding	-42,233.419	-3,707.823	-15,699.021	9,121.023

Table G.6.: Difference in the length of the minor axis (y-axis) of the standard deviational ellipse of the original coordinates and of the masked coordinates (in meter).

masking method	minimum diff. minor axis	maximum diff. minor axis	mean diff. minor axis	sd diff. minor axis
APA	28,538.975	28,538.975	28,538.975	0.000
APA LGA	-1,220.872	-1,220.872	-1,220.872	0.000
ARP	36,532.093	42,012.548	39,287.544	1,091.627
ARP LGA	807.053	4,230.679	2,348.260	646.828
CS	-46,419.345	45,996.950	-1,832.130	30,202.641

masking method	minimum diff.	maximum diff.	mean diff.	sd diff.
	minor axis	minor axis	minor axis	minor axis
DD 3	-158.485	146.609	14.081	58.859
DD 4	-118.359	224.379	38.805	80.587
DD 5	-177.058	245.848	58.467	82.421
DD LGA 3	-167.902	183.752	26.568	82.410
DD LGA 4	-162.879	233.437	39.176	87.153
DD LGA 5	-134.971	317.759	39.498	103.094
Dk 5	-1.059	0.931	-0.046	0.536
Dk 25	-2.239	2.421	-0.027	1.057
Dk 50	-95.346	195.470	3.839	80.200
Dk 100	-95.322	202.862	34.301	77.587
Dk 500	-244.000	562.387	169.396	211.283
Dk 1000	-82.027	1,673.851	620.153	347.474
DkData 20	-562.392	1,353.074	549.441	390.851
DUT	-0.000	0.000	0.000	0.000
LS 3	-26.558	3.391	-12.055	8.309
LS 4	-34.123	8.528	-11.545	9.867
LS 5	-93.094	26.232	-37.516	27.925
LS LGA 3	-107.727	-31.628	-70.785	18.833
LS LGA 4	-145.562	-68.482	-104.569	18.530
LS LGA 5	-188.553	-71.640	-138.771	25.014
LSdonut 3	-60.456	-12.502	-35.106	11.956
LSdonut 4	-155.058	-81.986	-115.618	16.236
LSdonut 5	-220.446	-130.066	-163.095	20.541
LSdonut LGA 3	-156.167	-57.352	-100.627	16.873
LSdonut LGA 4	-168.822	-94.603	-139.150	17.334
LSdonut LGA 5	-288.457	-144.314	-213.397	31.050
MDAV 3	-180.022	-180.022	-180.022	0.000
MDAV 25	-1,004.427	-1,004.427	-1,004.427	0.000
MDAV 50	-2,207.772	-2,207.772	-2,207.772	0.000
Rot	-0.000	0.000	0.000	0.000
RotArb	-0.000	0.000	0.000	0.000
RPC 3	-99.433	131.109	18.824	49.960
RPC 4	-112.490	111.994	20.071	56.329
RPC 5	-187.361	167.489	19.868	68.427
RPC LGA 3	-80.445	134.787	26.110	48.853
RPC LGA 4	-130.464	217.785	24.117	75.185
RPC LGA 5	-122.579	277.884	66.807	88.464
RPN	-77.303	57.604	7.494	32.327
RPN LGA	-94.656	78.844	12.821	40.907
RPU 3	-129.394	274.349	64.366	88.960
RPU 4	-241.672	337.446	73.620	128.166
RPU 5	-249.181	387.487	59.913	135.904
RPU LGA 3	-189.310	322.306	50.459	117.843

masking method	minimum diff. minor axis	maximum diff. minor axis	mean diff. minor axis	sd diff. minor axis
RPU LGA 4	-227.129	306.137	74.003	128.461
RPU LGA 5	-171.497	527.532	131.249	166.290
StreetMask 30	-620.518	-588.873	-602.426	6.931
StreetMask 100	-694.640	-548.299	-631.607	29.974
VNE 50 3	-370.260	-97.981	-220.281	62.280
VNE 50 5	-405.794	-54.481	-239.561	74.693
VNE 100 3	-553.075	-71.416	-334.401	110.634
VNE 100 5	-514.732	-51.472	-358.745	95.489
VNS 50 3	-126.742	53.439	-40.387	35.387
VNS 50 5	-228.353	14.750	-74.422	43.882
VNS 100 3	-405.855	-12.582	-136.582	73.585
VNS 100 5	-313.242	56.480	-159.863	78.044
VNE LGA 50 3	-397.627	-99.088	-261.137	64.438
VNE LGA 50 5	-466.441	-146.988	-304.550	77.714
VNE LGA 100 3	-663.129	-167.867	-375.890	111.923
VNE LGA 100 5	-596.205	-234.936	-400.072	79.975
VNS LGA 50 3	-192.269	-10.992	-106.061	43.897
VNS LGA 50 5	-245.788	-50.826	-148.072	40.505
VNS LGA 100 3	-353.810	-71.197	-197.484	68.507
VNS LGA 100 5	-371.137	-46.113	-206.323	74.959
Voronoi	-159.306	-159.306	-159.306	0.000
Grid 100	-0.082	-0.082	-0.082	0.000
Grid 1000	-4.733	-4.733	-4.733	0.000
ISGP	-34,018.559	-34,017.840	-34,018.238	0.187
Lipschitz Embedding	-9,065.041	2,566.923	-4,440.208	2,355.421

## G.5. Distance Between Coordinates of Data Set

Table G.7.: Mean and median distance between the coordinates (arithmetic mean of replications; standard deviation in parenthesis). Third and fifth column show the difference to the original mean (positive number indicate that the masked distance is larger; negative numbers that the masked distance is smaller).

masking method	mean distance	difference to original	median distance	difference to original
APA	152,116.668 (0.000)	47,589.447	37,422.776 (0.000)	1,196.800
APA LGA	104,862.170 (0.000)	334.949	37,639.723 (0.000)	1,413.747
ARP	157,094.392 (1,130.611)	52,567.172	39,594.845 (119.590)	3,368.869

masking method	mean distance	difference to original	median distance	difference to original
ARP LGA	106,862.165 (266.966)	2,334.945	38,968.381 (180.893)	2,742.405
CS	99,463.248 (65,012.022)	-5,063.972	34,521.489 (22,607.779)	-1,704.487
DD 3	104,537.237 (28.660)	10.017	36,225.721 (3.50)3	-0.255
DD 4	104,539.107 (35.996)	11.886	36,225.373 (5.170)	-0.603
DD 5	104,542.265 (44.754)	15.044	36,224.962 (6.693)	-1.014
DD LGA 3	104,530.380 (32.776)	3.159	36,225.363 (4.230)	-0.613
DD LGA 4	104,541.503 (45.046)	14.283	36,226.455 (5.659)	0.479
DD LGA 5	104,542.205 (44.121)	14.984	36,225.259 (5.584)	-0.716
Dk 5	104,527.233 (0.556)	0.013	36,225.868 (0.770)	-0.107
Dk 25	104,527.361 (0.893)	0.140	36,225.914 (1.420)	-0.062
Dk 50	104,525.976 (21.332)	-1.244	36,226.107 (2.668)	0.131
Dk 100	104,534.926 (28.734)	7.706	36,226.084 (5.608)	0.108
Dk 500	104,566.499 (76.587)	39.279	36,225.341 (16.545)	-0.634
Dk 1000	104,628.877 (134.116)	101.656	36,219.640 (28.032)	-6.336
DkData 20	104,636.439 (154.246)	109.219	36,213.932 (29.072)	-12.043
DUT	104,526.923 (4.762)	-0.298	36,225.859 (1.851)	-0.117
ISGP	44,326.354 (0.195)	-60,200.866	36,100.104 (0.630)	-125.872
Lipschitz Embedding	93,126.272 (2043.369)	-11,400.948	36,103.432 (19.069)	-122.544
LS 3	104,522.593 (8.944)	-4.627	36,223.692 (2.413)	-2.284

masking method	mean distance	difference to original	median distance	difference to original
LS 4	104,525.224 (8.983)	-1.997	36,223.531 (4.110)	-2.445
LS 5	104,480.056 (22.274)	-47.164	36,222.830 (5.141)	-3.146
LS LGA 3	104,514.269 (9.538)	-12.952	36,225.186 (3.407)	-0.790
LS LGA 4	104,499.478 (9.197)	-27.742	36,223.209 (3.893)	-2.767
LS LGA 5	104,477.761 (16.057)	-49.460	36,221.540 (4.529)	-4.436
LSdonut 3	104,507.540 (10.109)	-19.681	36,221.786 (3.318)	-4.189
LSdonut 4	104,451.387 (11.695)	-75.833	36,219.966 (4.310)	-6.010
LSdonut 5	104,301.089 (15.986)	-226.132	36,215.408 (3.858)	-10.568
LSdonut LGA 3	104,512.809 (12.618)	-14.411	36,222.591 (2.678)	-3.385
LSdonut LGA 4	104,504.729 (9.485)	-22.491	36,218.830 (4.269)	-7.146
LSdonut LGA 5	104,440.111 (19.835)	-87.109	36,215.782 (4.304)	-10.193
MDAV 3	104,481.110 (0.000)	-46.111	36,232.000 (0.000)	6.024
MDAV 25	104,160.902 (0.000)	-366.318	36,315.326 (0.000)	89.350
MDAV 50	103,608.224 (0.000)	-918.996	36,123.796 (0.000)	-102.179
Rot	97,182.004 (7,697.145)	-7,345.217	33,674.447 (2,670.105)	-2,551.529
RotArb	104,456.138 (76.186)	-71.082	36,181.557 (33.875)	-44.419
RPC 3	104,535.536 (23.084)	8.315	36,225.632 (2.783)	-0.344
RPC 4	104,535.636 (25.621)	8.416	36,224.621 (3.942)	-1.355
RPC 5	104,535.426 (40.138)	8.206	36,225.866 (5.318)	-0.110

masking method	mean distance	difference to original	median distance	difference to original
RPC LGA 3	104,535.113 (25.985)	7.892	36,226.218 (2.729)	0.242
RPC LGA 4	104,528.229 (31.225)	1.009	36,226.668 (4.026)	0.693
RPC LGA 5	104,542.679 (42.009)	15.459	36,226.998 (5.255)	1.022
RPN	104,527.059 (15.315)	-0.161	36,225.488 (2.354)	-0.488
RPN LGA	104,530.977 (19.000)	3.757	36,226.030 (2.423)	0.054
RPU 3	104,544.090 (37.956)	16.869	36,223.627 (6.607)	-2.349
RPU 4	104,541.463 (48.431)	14.242	36,223.184 (7.425)	-2.792
RPU 5	104,546.729 (52.384)	19.508	36,222.441 (10.571)	-3.534
RPU LGA 3	104,540.523 (41.017)	13.302	36,225.636 (7.130)	-0.340
RPU LGA 4	104,554.481 (52.069)	27.260	36,226.372 (7.769)	0.396
RPU LGA 5	104,572.590 (65.699)	45.370	36,227.914 (9.304)	1.938
StreetMask 30	104,134.227 (4.323)	-392.993	36,221.633 (3.524)	-4.342
StreetMask 100	104,121.009 (14.621)	-406.211	36,213.709 (8.183)	-12.267
VNE 50 3	104,388.659 (35.738)	-138.561	36,191.309 (10.443)	-34.667
VNE 50 5	104,370.248 (49.364)	-156.972	36,188.931 (12.594)	-37.044
VNE 100 3	104,297.415 (76.320)	-229.805	36,172.432 (12.265)	-53.543
VNE 100 5	104,293.176 (74.627)	-234.044	36,169.352 (12.073)	-56.624
VNS 50 3	104,503.835 (28.317)	-23.386	36,214.337 (4.709)	-11.639
VNS 50 5	104,449.437 (34.693)	-77.784	36,212.000 (6.137)	-13.976

masking method	mean distance	difference to original	median distance	difference to original
VNS 100 3	104,430.332 (50.591)	-96.889	36,207.523 (4.824)	-18.453
VNS 100 5	104,416.846 (61.124)	-110.374	36,205.160 (6.447)	-20.816
VNE LGA 50 3	104,381.892 (34.831)	-145.329	36,192.681 (10.074)	-33.295
VNE LGA 50 5	104,350.011 (44.944)	-177.210	36,194.049 (12.599)	-31.927
VNE LGA 100 3	104,299.077 (83.387)	-228.143	36,175.746 (12.984)	-50.230
VNE LGA 100 5	104,261.107 (86.558)	-266.114	36,171.323 (15.224)	-54.653
VNS LGA 50 3	104,489.438 (27.956)	-37.783	36,215.482 (4.731)	-10.494
VNS LGA 50 5	104,453.553 (34.442)	-73.667	36,212.335 (6.349)	-13.641
VNS LGA 100 3	104,422.176 (54.555)	-105.044	36,208.377 (4.591)	-17.599
VNS LGA 100 5	104,400.460 (56.391)	-126.760	36,205.329 (6.452)	-20.647
Voronoi	104,436.201 (0.000)	-91.019	36,216.860 (0.000)	-9.116
Grid 100	104,224.284 (0.000)	-302.936	36,100.554 (0.000)	-125.422
Grid 1000	104,227.456 (0.000)	-299.765	36,124.784 (0.000)	-101.192

## G.6. Distance Between Original and Masked Coordinates

Table G.8.: Detailed results for mean and median distances points are moved.

method	mean	sd	min	max	
APA	39,677.527	0.000	39,677.527	39,677.527	mean
	3,017.620	0.000	3,017.620	3,017.620	median
APA LGA	8,489.559	0.000	8,489.559	8,489.559	mean
	4,289.409	0.000	4,289.409	4,289.409	median
ARP	46,417.115	612.853	45,280.561	48,453.054	mean
	4,085.771	32.554	4,015.227	4,186.988	median

method	mean	sd	min	max	
ARP LGA	12,563.146	134.603	12,277.552	12,920.925	mean
	6,120.041	67.724	5,990.164	6,250.059	median
CS	1,170,831.218	641,103.719	116,930.112	2,052,668.194	mean
	1,167,795.251	639,504.323	116,617.397	2,048,074.850	median
DD 3	521.218	2.796	516.854	526.712	mean
	69.899	0.203	69.421	70.306	median
DD 4	625.125	5.171	614.097	637.504	mean
	85.269	0.347	84.438	85.912	median
DD 5	729.069	9.188	705.521	757.598	mean
	102.168	0.362	101.278	102.867	median
DD LGA 3	523.985	2.728	517.239	529.954	mean
	86.931	0.364	86.133	87.800	median
DD LGA 4	629.781	5.402	615.305	644.024	mean
	107.125	0.516	105.646	107.911	median
DD LGA 5	731.632	8.765	711.228	746.442	mean
	126.207	0.526	124.994	127.690	median
Dk 5	33.914	0.072	33.773	34.115	mean
	30.511	0.097	30.291	30.747	median
Dk 25	63.246	0.295	62.204	63.725	mean
	55.175	0.360	53.975	55.764	median
Dk 50	132.515	8.617	113.747	147.311	mean
	83.838	0.425	82.957	84.672	median
Dk 100	237.752	12.839	213.776	262.545	mean
	121.425	0.650	120.023	123.269	median
Dk 500	983.814	24.475	929.194	1029.254	mean
	295.637	1.453	291.769	299.233	median
Dk 1000	1,808.572	43.613	1,711.946	1,905.683	mean
	436.099	2.483	430.733	441.240	median
DkData 20	2,156.495	47.875	2,024.220	2,263.255	mean
	489.265	2.846	482.796	495.590	median
DUT	7,700.957	2,750.898	1,717.235	12,500.416	mean
	7,701.500	2,751.094	1,717.350	12,501.319	median
LS 3	197.754	4.409	185.918	207.869	mean
	59.722	0.192	59.355	60.376	median
LS 4	242.450	4.959	232.408	254.605	mean
	79.503	0.295	78.958	80.266	median
LS 5	312.697	13.017	285.785	339.191	mean
	98.908	0.346	98.072	100.000	median
LS LGA 3	235.875	5.629	222.468	245.679	mean
	71.868	0.301	71.127	72.646	median
LS LGA 4	305.204	5.977	292.780	318.299	mean
	95.196	0.454	94.305	96.327	median
LS LGA 5	364.269	8.555	346.608	380.983	mean
	117.970	0.630	116.491	119.280	median

method	mean	sd	min	max	
LSdonut 3	321.569	1.310	318.893	324.522	mean
	63.056	0.257	62.485	63.581	median
LSdonut 4	370.794	1.798	367.031	374.941	mean
	83.750	0.223	83.294	84.547	median
LSdonut 5	528.500	4.528	520.332	537.367	mean
	103.837	0.267	103.183	104.286	median
LSdonut LGA 3	329.496	1.555	326.377	332.467	mean
	78.975	0.361	78.216	79.674	median
LSdonut LGA 4	432.697	1.922	429.407	438.841	mean
	103.524	0.469	102.507	104.582	median
LSdonut LGA 5	524.446	3.052	517.861	533.250	mean
	127.298	0.642	126.150	128.563	median
MDAV 3	800.946	0.000	800.946	800.946	mean
	152.951	0.000	152.951	152.951	median
MDAV 25	4601.068	0.000	4601.068	4601.068	mean
	675.859	0.000	675.859	675.859	median
MDAV 50	6,875.113	0.000	6,875.113	6,875.113	mean
	1,015.118	0.000	1,015.118	1,015.118	median
Rot	2,548,618.339	1,192,021.539	37,451.769	4,076,247.818	mean
	2,541,755.210	1,188,948.905	37,353.458	4,066,122.559	median
RotArb	83,589.937	37,813.949	1,084.523	124,309.739	mean
	23,263.834	10,525.217	301.718	34,613.940	median
RPC 3	315.073	7.340	298.163	329.903	mean
	49.671	0.327	48.804	50.468	median
RPC 4	417.625	9.795	400.363	437.366	mean
	66.087	0.513	65.002	67.427	median
RPC 5	521.780	13.974	482.366	558.207	mean
	82.608	0.697	80.682	83.820	median
RPC LGA 3	312.112	7.427	293.964	327.747	mean
	57.638	0.517	56.546	58.826	median
RPC LGA 4	418.558	14.245	390.942	450.104	mean
	77.014	0.625	75.905	79.154	median
RPC LGA 5	523.862	12.468	499.058	551.769	mean
	96.029	0.746	94.707	98.181	median
RPN	260.640	6.649	248.581	276.507	mean
	38.914	0.312	38.180	39.597	median
RPN LGA	261.505	6.822	248.638	279.833	mean
	46.659	0.434	45.553	48.014	median
RPU 3	739.581	2.480	733.067	744.130	mean
	99.106	0.263	98.595	99.634	median
RPU 4	892.583	5.694	878.326	906.208	mean
	120.078	0.379	119.396	120.763	median
RPU 5	1,046.432	7.705	1,030.418	1,063.215	mean
	142.368	0.553	141.252	143.779	median

method	mean	sd	min	max	
RPU LGA 3	742.987	2.978	736.292	750.765	mean
	121.966	0.287	121.161	122.421	median
RPU LGA 4	895.624	5.906	878.231	905.670	mean
	149.291	0.585	147.666	150.560	median
RPU LGA 5	1,056.161	7.163	1,040.849	1,069.586	mean
	177.701	0.715	176.155	179.269	median
StreetMask 30	523.805	1.509	521.085	526.599	mean
	170.479	0.746	168.673	172.442	median
StreetMask 100	730.434	5.818	718.009	745.242	mean
	296.348	1.384	293.598	299.328	median
VNE 50 3	548.340	24.038	494.405	589.879	mean
	183.920	0.973	182.154	185.602	median
VNE 50 5	611.894	27.260	551.830	675.411	mean
	190.094	0.945	188.342	192.481	median
VNE 100 3	895.482	39.311	814.618	990.532	mean
	259.682	1.019	257.442	261.761	median
VNE 100 5	954.415	37.029	873.741	1,015.025	mean
	266.163	1.736	262.733	271.539	median
VNS 50 3	315.240	16.738	283.850	347.256	mean
	135.538	0.696	133.992	137.143	median
VNS 50 5	406.597	23.554	358.138	459.228	mean
	141.053	0.637	139.524	142.735	median
VNS 100 3	498.890	25.721	438.475	563.482	mean
	191.525	0.895	189.517	193.803	median
VNS 100 5	568.034	26.764	507.863	608.450	mean
	196.553	1.048	193.905	198.566	median
VNE LGA 50 3	566.707	20.842	522.246	625.678	mean
	185.166	0.892	182.383	187.201	median
VNE LGA 50 5	648.732	26.415	576.391	699.868	mean
	196.586	1.111	194.010	198.819	median
VNE LGA 100 3	925.865	36.313	818.133	1,002.923	mean
	262.345	1.468	259.570	266.793	median
VNE LGA 100 5	992.028	34.875	908.814	1,077.137	mean
	269.110	1.542	266.430	272.340	median
VNS LGA 50 3	345.646	16.032	313.118	380.643	mean
	137.102	0.737	135.344	138.629	median
VNS LGA 50 5	442.582	21.350	397.411	479.590	mean
	147.952	0.783	146.591	149.943	median
VNS LGA 100 3	526.332	24.447	461.097	575.307	mean
	193.029	1.146	190.762	196.299	median
VNS LGA 100 5	603.983	29.709	535.204	678.383	mean
	200.267	0.984	197.915	202.226	median
Voronoi	177.872	0.000	177.872	177.872	mean
	65.214	0.000	65.214	65.214	median

method	mean	sd	min	max	
Grid 100	38.252	0.000	38.252	38.252	mean
	39.894	0.000	39.894	39.894	median
Grid 1000	382.676	0.000	382.676	382.676	mean
	397.903	0.000	397.903	397.903	median
ISGP	55,383.179	0.082	55,382.957	55,383.360	mean
	10,366.291	1.812	10,361.156	10,369.383	median
Lipschitz	23,726.107	10,447.891	11,533.774	50,603.377	mean
	11,771.115	6,588.386	4,524.874	28,764.392	median

Table G.9.: Minimum, maximum, mean and median standard deviations of the distances over all iterations.

masking method	minimum sd	maximum sd	mean sd	median sd
APA	0.000	0.000	0.000	0.000
APA LGA	0.000	0.000	0.000	0.000
ARP	483.789	310,559.554	22,743.650	1,835.458
ARP LGA	395.651	247,684.838	5,975.914	3,411.335
CS	588,094.919	803,157.966	641,107.463	639,501.640
DD 3	4.277	8,379.806	59.774	8.028
DD 4	8.826	19,134.937	120.085	16.074
DD 5	12.621	26,266.135	179.675	24.119
DD LGA 3	4.480	11,884.340	60.305	9.880
DD LGA 4	8.786	24,038.525	120.677	19.791
DD LGA 5	12.976	29,752.924	179.854	29.626
Dk 5	0.026	108.204	5.573	5.134
Dk 25	1.669	519.254	22.482	19.126
Dk 50	2.432	76,499.120	51.138	25.143
Dk 100	4.756	78,250.837	103.602	37.209
Dk 500	28.871	85,119.292	440.730	89.940
Dk 1000	58.039	148,556.014	805.963	132.588
DkData 20	51.997	189,315.425	1,036.878	186.640
DUT	2,738.175	2,755.564	2,750.898	2,751.093
LS 3	0.000	31,466.575	95.556	18.964
LS 4	0.000	30,603.090	118.807	25.914
LS 5	0.000	62,598.194	178.765	32.976
LSdonut 3	0.000	4,571.381	40.783	11.121
LSdonut 4	0.000	4,648.041	53.630	14.764
LSdonut 5	0.000	32,764.993	71.289	18.309
LSdonut LGA 3	0.000	4,535.388	47.133	13.885
LSdonut LGA 4	0.000	9,376.356	57.684	18.210
LSdonut LGA 5	0.000	12,658.268	71.217	22.716
LS LGA 3	0.000	13,575.140	113.953	23.405

masking method	minimum sd	maximum sd	mean sd	median sd
LS LGA 4	0.000	24,185.887	145.854	31.774
LS LGA 5	0.000	27,434.203	173.692	40.175
MDAV 3	0.000	0.000	0.000	0.000
MDAV 25	0.000	0.000	0.000	0.000
MDAV 50	0.000	0.000	0.000	0.000
Rot	1,094,411.852	1,506,127.747	1,192,032.459	1,188,927.920
RotArb	540.600	798,019.268	37,814.116	10,529.417
RPC 3	12.785	28,830.619	180.689	24.002
RPC 4	16.985	32,231.909	239.594	32.157
RPC 5	20.948	43,740.252	300.455	40.174
RPC LGA 3	12.760	34,395.506	180.978	29.576
RPC LGA 4	18.166	53,791.626	241.624	39.429
RPC LGA 5	22.543	59,228.641	300.412	49.227
RPN	8.683	21,171.389	134.755	18.219
RPN LGA	9.264	24,531.282	134.922	22.395
RPU 3	3.878	8,223.129	59.749	8.057
RPU 4	7.787	17,638.524	119.707	16.088
RPU 5	11.486	29,108.090	179.974	24.129
RPU LGA 3	3.750	12,154.434	60.686	9.864
RPU LGA 4	8.422	23,158.928	120.870	19.782
RPU LGA 5	10.307	38,793.047	179.024	29.609
StreetMask 30	6.526	5,807.748	73.963	54.169
StreetMask 100	27.518	27,954.917	194.273	99.155
VNE 50 3	11.595	117,537.368	309.815	64.970
VNE 50 5	14.687	128,121.537	380.851	67.894
VNE 100 3	21.420	119,518.698	549.663	94.061
VNE 100 5	22.520	123,835.353	594.740	95.462
VNS 50 3	10.016	128,114.163	188.379	45.557
VNS 50 5	13.071	125,683.725	261.192	46.946
VNS 100 3	11.581	110,051.781	312.274	65.193
VNS 100 5	21.353	115,235.933	368.238	66.781
VNE LGA 50 3	10.770	124,480.131	327.503	65.785
VNE LGA 50 5	16.215	127,283.922	362.855	69.842
VNE LGA 100 3	16.593	133,126.500	572.159	95.643
VNE LGA 100 5	20.544	124,606.042	603.851	96.009
VNS LGA 50 3	11.339	128,223.779	205.623	45.533
VNS LGA 50 5	15.210	128,243.327	249.002	51.557
VNS LGA 100 3	12.370	108,505.869	329.533	65.674
VNS LGA 100 5	14.121	105,277.414	361.768	68.188
Voronoi	0.000	0.000	0.000	0.000
Grid 100	0.000	0.000	0.000	0.000
Grid 1000	0.000	0.000	0.000	0.000
ISGP	0.215	37.129	6.821	7.451

masking method	minimum sd	maximum sd	mean sd	median sd
Lipschitz Embedding	5,046.659	113,564.576	13,537.316	6,828.730

## G.7. Spatial Autocorrelation

Table G.10.: Results of Moran's I values for proportion of single households. Expected value always at -0.0001. Moran's I always significant ( $p < 0.01$ ).

method	mean	sd	min	max
APA	0.013	0.000	0.013	0.013
APA LGA	0.007	0.000	0.007	0.007
ARP	0.040	0.000	0.039	0.041
ARP LGA	0.038	0.000	0.037	0.040
CS	0.129	0.000	0.129	0.129
DD 3	0.104	0.001	0.103	0.106
DD 4	0.102	0.001	0.101	0.104
DD 5	0.099	0.001	0.098	0.101
DD LGA 3	0.103	0.001	0.101	0.104
DD LGA 4	0.100	0.001	0.099	0.101
DD LGA 5	0.097	0.001	0.096	0.099
Dk 5	0.128	0.001	0.127	0.130
Dk 25	0.127	0.001	0.126	0.128
Dk 50	0.124	0.001	0.123	0.126
Dk 100	0.120	0.001	0.119	0.122
Dk 500	0.104	0.001	0.102	0.105
Dk 1000	0.092	0.001	0.090	0.093
DkData 20	0.088	0.001	0.086	0.090
DUT	0.129	0.000	0.129	0.129
LS 3	0.122	0.001	0.120	0.124
LS 4	0.120	0.001	0.118	0.124
LS 5	0.118	0.001	0.117	0.121
LS LGA 3	0.117	0.001	0.115	0.120
LS LGA 4	0.115	0.001	0.113	0.118
LS LGA 5	0.114	0.001	0.111	0.116
LSdonut 3	0.120	0.001	0.118	0.122
LSdonut 4	0.122	0.001	0.120	0.124
LSdonut 5	0.121	0.001	0.119	0.123
LSdonut LGA 3	0.118	0.001	0.117	0.120
LSdonut LGA 4	0.118	0.001	0.116	0.121

method	mean	sd	min	max
LSdonut LGA 5	0.116	0.001	0.114	0.118
MDAV 3	0.090	0.000	0.090	0.090
MDAV 25	0.047	0.000	0.047	0.047
MDAV 50	0.035	0.000	0.035	0.035
ISGP	0.106	0.000	0.105	0.107
Lipschitz Embedding	0.125	0.000	0.124	0.127
Rot	0.129	0.000	0.129	0.129
RotArb	0.129	0.000	0.129	0.129
RPC 3	0.109	0.001	0.107	0.111
RPC 4	0.106	0.001	0.104	0.107
RPC 5	0.103	0.001	0.101	0.104
RPC LGA 3	0.107	0.001	0.106	0.109
RPC LGA 4	0.104	0.001	0.102	0.105
RPC LGA 5	0.101	0.001	0.100	0.103
RPN	0.111	0.001	0.110	0.113
RPN LGA	0.110	0.001	0.108	0.111
RPU 3	0.102	0.001	0.101	0.104
RPU 4	0.099	0.001	0.097	0.100
RPU 5	0.096	0.001	0.094	0.097
RPU LGA 3	0.100	0.001	0.099	0.102
RPU LGA 4	0.097	0.001	0.095	0.098
RPU LGA 5	0.094	0.001	0.092	0.095
StreetMask 30	0.124	0.001	0.122	0.127
StreetMask 100	0.117	0.001	0.115	0.120
VNE 50 3	0.118	0.001	0.115	0.121
VNE 50 5	0.115	0.001	0.112	0.117
VNE 100 3	0.113	0.001	0.110	0.115
VNE 100 5	0.110	0.001	0.108	0.112
VNS 50 3	0.121	0.001	0.118	0.123
VNS 50 5	0.118	0.001	0.115	0.119
VNS 100 3	0.119	0.001	0.117	0.121
VNS 100 5	0.116	0.001	0.114	0.119
VNE LGA 50 3	0.115	0.001	0.113	0.117
VNE LGA 50 5	0.111	0.001	0.109	0.115
VNE LGA 100 3	0.110	0.001	0.108	0.112
VNE LGA 100 5	0.108	0.001	0.105	0.110
VNS LGA 50 3	0.117	0.001	0.115	0.119
VNS LGA 50 5	0.113	0.001	0.111	0.116
VNS LGA 100 3	0.115	0.001	0.113	0.117

method	mean	sd	min	max
VNS LGA 100 5	0.112	0.001	0.109	0.114
Voronoi	0.248	0.000	0.248	0.248
Grid 100	0.118	0.000	0.118	0.118
Grid 1000	0.064	0.000	0.064	0.064

Table G.11.: Results of Moran's I values for proportion of full-time working people. Expected value always at -0.0001. Moran's I always significant.

method	mean	sd	min	max
APA	0.005	0.000	0.005	0.005
APA LGA	0.010	0.000	0.010	0.010
ARP	0.035	0.001	0.034	0.036
ARP LGA	0.039	0.001	0.037	0.040
CS	0.170	0.000	0.170	0.170
DD 3	0.129	0.001	0.127	0.132
DD 4	0.124	0.001	0.122	0.126
DD 5	0.120	0.001	0.118	0.123
DD LGA 3	0.125	0.001	0.124	0.128
DD LGA 4	0.121	0.001	0.119	0.123
DD LGA 5	0.117	0.001	0.115	0.120
Dk 5	0.169	0.001	0.166	0.171
Dk 25	0.167	0.001	0.164	0.170
Dk 50	0.162	0.001	0.160	0.166
Dk 100	0.156	0.001	0.152	0.159
Dk 500	0.130	0.001	0.128	0.134
Dk 1000	0.116	0.001	0.113	0.119
DkData 20	0.109	0.001	0.106	0.112
DUT	0.170	0.000	0.170	0.170
LS 3	0.160	0.001	0.158	0.164
LS 4	0.158	0.002	0.154	0.163
LS 5	0.156	0.002	0.151	0.160
LS LGA 3	0.158	0.002	0.155	0.163
LS LGA 4	0.156	0.002	0.151	0.161
LS LGA 5	0.154	0.002	0.150	0.158
LSdonut 3	0.163	0.002	0.160	0.170
LSdonut 4	0.162	0.002	0.158	0.166
LSdonut 5	0.159	0.002	0.155	0.163
LSdonut LGA 3	0.160	0.002	0.157	0.164

method	mean	sd	min	max
LSdonut LGA 4	0.159	0.002	0.157	0.164
LSdonut LGA 5	0.160	0.002	0.156	0.164
MDAV 3	0.114	0.000	0.114	0.114
MDAV 25	0.052	0.000	0.052	0.052
MDAV 50	0.035	0.000	0.035	0.035
ISGP	0.128	0.000	0.128	0.129
Lipschitz Embedding	0.163	0.001	0.160	0.167
Rot	0.170	0.000	0.169	0.170
RotArb	0.170	0.000	0.169	0.170
RPC 3	0.136	0.001	0.133	0.139
RPC 4	0.130	0.001	0.128	0.132
RPC 5	0.126	0.001	0.123	0.128
RPC LGA 3	0.133	0.001	0.130	0.135
RPC LGA 4	0.127	0.001	0.125	0.130
RPC LGA 5	0.123	0.001	0.121	0.126
RPN	0.139	0.001	0.136	0.141
RPN LGA	0.136	0.001	0.134	0.140
RPU 3	0.125	0.001	0.123	0.128
RPU 4	0.119	0.001	0.116	0.121
RPU 5	0.114	0.001	0.113	0.117
RPU LGA 3	0.122	0.001	0.120	0.126
RPU LGA 4	0.116	0.001	0.114	0.118
RPU LGA 5	0.111	0.001	0.109	0.113
StreetMask 30	0.170	0.002	0.166	0.175
StreetMask 100	0.157	0.002	0.153	0.165
VNE 50 3	0.156	0.002	0.151	0.161
VNE 50 5	0.151	0.002	0.145	0.155
VNE 100 3	0.152	0.002	0.148	0.158
VNE 100 5	0.148	0.003	0.144	0.155
VNS 50 3	0.160	0.002	0.155	0.164
VNS 50 5	0.155	0.002	0.152	0.161
VNS 100 3	0.158	0.002	0.153	0.162
VNS 100 5	0.154	0.002	0.149	0.161
VNE LGA 50 3	0.153	0.002	0.149	0.159
VNE LGA 50 5	0.150	0.002	0.146	0.157
VNE LGA 100 3	0.150	0.002	0.145	0.156
VNE LGA 100 5	0.147	0.002	0.143	0.153
VNS LGA 50 3	0.157	0.002	0.153	0.160
VNS LGA 50 5	0.153	0.002	0.149	0.159

method	mean	sd	min	max
VNS LGA 100 3	0.155	0.002	0.150	0.163
VNS LGA 100 5	0.152	0.002	0.148	0.157
Voronoi	0.303	0.000	0.303	0.303
Grid 100	0.157	0.000	0.157	0.157
Grid 1000	0.086	0.000	0.086	0.086

## G.8. Clustering

Table G.12.: Results of DBSCAN clustering algorithm ( $\epsilon = 3200$ ).

method	numclus	larger10	larger20	larger30	single	stat
Orig	130	60	30	17	248	
APA	45.00	45.00	45.00	45.00	0.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	45.00	45.00	45.00	45.00	0.00	min
	45.00	45.00	45.00	45.00	0.00	max
APA LGA	67.00	64.00	53.00	44.00	4.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	67.00	64.00	53.00	44.00	4.00	min
	67.00	64.00	53.00	44.00	4.00	max
ARP	64.02	10.46	3.80	2.64	1991.26	mean
	5.25	2.38	1.01	0.66	19.01	sd
	53.00	6.00	2.00	1.00	1951.00	min
	83.00	18.00	7.00	4.00	2041.00	max
ARP LGA	74.00	22.04	12.04	7.50	1285.60	mean
	6.04	3.01	1.51	1.27	21.58	sd
	61.00	16.00	10.00	4.00	1233.00	min
	88.00	28.00	16.00	10.00	1335.00	max
CS	111.16	53.24	27.74	16.02	203.38	mean
	36.34	14.22	6.87	3.72	85.17	sd
	5.00	4.00	3.00	2.00	7.00	min
	142.00	64.00	34.00	20.00	278.00	max
DD 3	109.08	54.22	28.46	16.68	399.16	mean
	2.67	1.04	0.99	0.51	12.39	sd
	104.00	52.00	27.00	16.00	369.00	min
	115.00	57.00	31.00	18.00	423.00	max

method	numclus	larger10	larger20	larger30	single	stat
Orig	130	60	30	17	248	
DD 4	103.64	52.70	27.64	16.82	434.26	mean
	2.97	1.49	1.10	0.66	9.63	sd
	98.00	50.00	26.00	16.00	411.00	min
	110.00	55.00	30.00	18.00	452.00	max
DD 5	99.36	51.02	27.58	16.88	474.18	mean
	3.82	2.08	1.16	0.59	15.77	sd
	89.00	46.00	25.00	16.00	432.00	min
	108.00	55.00	30.00	18.00	523.00	max
DD LGA 3	111.02	56.86	30.12	16.80	358.32	mean
	2.99	1.20	1.21	0.61	9.85	sd
	104.00	55.00	27.00	16.00	336.00	min
	118.00	60.00	32.00	19.00	377.00	max
DD LGA 4	105.34	54.38	29.34	16.80	405.68	mean
	3.71	1.76	1.15	0.70	13.76	sd
	100.00	50.00	26.00	16.00	379.00	min
	115.00	58.00	32.00	18.00	439.00	max
DD LGA 5	101.68	51.02	28.44	16.74	447.98	mean
	3.09	2.14	0.93	0.60	14.41	sd
	94.00	48.00	26.00	16.00	420.00	min
	108.00	56.00	31.00	18.00	484.00	max
Dk 5	129.54	59.84	30.14	17.00	247.94	mean
	0.58	0.37	0.67	0.00	0.24	sd
	128.00	59.00	29.00	17.00	247.00	min
	130.00	60.00	32.00	17.00	248.00	max
Dk 25	128.66	59.66	30.66	16.92	247.32	mean
	0.98	0.56	0.89	0.40	0.91	sd
	127.00	58.00	29.00	16.00	245.00	min
	130.00	61.00	33.00	18.00	249.00	max
Dk 50	128.42	59.76	31.06	16.90	246.44	mean
	1.36	0.98	1.13	0.68	2.46	sd
	126.00	58.00	29.00	16.00	241.00	min
	132.00	62.00	34.00	19.00	250.00	max
Dk 100	128.84	59.78	31.10	17.00	243.72	mean
	1.52	0.91	1.04	0.73	3.07	sd
	126.00	58.00	29.00	16.00	235.00	min
	133.00	61.00	34.00	19.00	250.00	max
Dk 500	78.52	57.58	29.26	16.58	487.62	mean
	2.67	1.40	1.19	0.91	12.29	sd

method	numclus	larger10	larger20	larger30	single	stat
Orig	130	60	30	17	248	
	74.00	55.00	26.00	15.00	459.00	min
	84.00	62.00	32.00	19.00	510.00	max
Dk 1000	52.74	37.24	27.60	15.74	747.26	mean
	2.75	1.55	1.31	0.85	13.19	sd
	48.00	34.00	25.00	14.00	717.00	min
	58.00	41.00	30.00	18.00	776.00	max
DkData 20	43.88	28.86	26.10	15.56	850.44	mean
	3.09	1.36	1.18	0.81	14.24	sd
	37.00	26.00	24.00	14.00	815.00	min
	49.00	32.00	29.00	18.00	879.00	max
DUT	130.00	60.00	30.00	17.00	248.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	130.00	60.00	30.00	17.00	248.00	min
	130.00	60.00	30.00	17.00	248.00	max
ISGP	130.00	60.00	30.00	17.00	248.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	130.00	60.00	30.00	17.00	248.00	min
	130.00	60.00	30.00	17.00	248.00	max
Lipschitz Embedding	130.00	60.00	30.00	17.00	247.96	mean
	0.00	0.00	0.00	0.00	0.20	sd
	130.00	60.00	30.00	17.00	247.00	min
	130.00	60.00	30.00	17.00	248.00	max
LS 3	129.20	59.28	30.52	16.84	247.28	mean
	1.23	0.70	0.68	0.37	3.39	sd
	127.00	57.00	29.00	16.00	241.00	min
	131.00	60.00	32.00	17.00	255.00	max
LS 4	128.90	59.20	30.34	16.80	247.92	mean
	1.37	0.90	0.69	0.53	4.43	sd
	126.00	57.00	29.00	16.00	240.00	min
	132.00	61.00	31.00	18.00	259.00	max
LS 5	127.86	59.80	30.32	16.98	251.72	mean
	1.48	1.11	0.98	0.68	5.33	sd
	125.00	58.00	29.00	16.00	242.00	min
	131.00	62.00	33.00	19.00	265.00	max
LS LGA 3	127.96	58.72	30.72	18.28	246.90	mean
	1.41	0.70	0.54	0.67	3.27	sd
	125.00	57.00	30.00	17.00	238.00	min

method	numclus	larger10	larger20	larger30	single	stat
Orig	130	60	30	17	248	
	132.00	60.00	32.00	19.00	253.00	max
LS LGA 4	128.20	58.98	30.94	18.22	242.88	mean
	1.29	0.87	0.89	0.74	2.91	sd
	126.00	56.00	28.00	17.00	236.00	min
	131.00	61.00	33.00	20.00	248.00	max
LS LGA 5	128.12	59.12	30.52	18.12	243.24	mean
	1.60	0.69	0.76	0.66	4.05	sd
	125.00	58.00	29.00	17.00	235.00	min
	132.00	61.00	32.00	20.00	258.00	max
LSdonut 3	128.20	59.60	29.78	16.84	248.64	mean
	0.95	0.64	0.42	0.37	2.95	sd
	126.00	58.00	29.00	16.00	240.00	min
	131.00	60.00	30.00	17.00	257.00	max
LSdonut 4	127.54	58.14	30.22	16.88	256.42	mean
	1.45	0.70	0.79	0.52	4.09	sd
	123.00	57.00	29.00	16.00	248.00	min
	130.00	59.00	32.00	18.00	264.00	max
LSdonut 5	127.68	62.04	29.64	17.18	242.84	mean
	1.50	1.09	0.88	0.69	2.98	sd
	125.00	59.00	28.00	16.00	237.00	min
	131.00	64.00	31.00	18.00	249.00	max
LSdonut LGA 3	127.88	59.94	31.08	19.00	246.76	mean
	1.06	0.55	0.49	0.57	2.99	sd
	126.00	59.00	30.00	18.00	240.00	min
	130.00	61.00	32.00	20.00	252.00	max
LSdonut LGA 4	126.02	58.94	30.52	18.66	254.70	mean
	1.42	1.00	0.91	0.85	4.19	sd
	124.00	56.00	29.00	17.00	244.00	min
	130.00	61.00	32.00	20.00	266.00	max
LSdonut LGA 5	130.20	60.56	30.28	17.76	236.32	mean
	1.67	1.01	1.09	0.92	4.01	sd
	127.00	59.00	28.00	16.00	227.00	min
	135.00	63.00	32.00	19.00	245.00	max
MdAV 3	89.00	50.00	30.00	17.00	531.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	89.00	50.00	30.00	17.00	531.00	min
	89.00	50.00	30.00	17.00	531.00	max

method	numclus	larger10	larger20	larger30	single	stat
Orig	130	60	30	17	248	
MDAV 25	94.00	94.00	94.00	13.00	0.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	94.00	94.00	94.00	13.00	0.00	min
	94.00	94.00	94.00	13.00	0.00	max
MDAV 50	65.00	65.00	65.00	65.00	0.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	65.00	65.00	65.00	65.00	0.00	min
	65.00	65.00	65.00	65.00	0.00	max
Rot	127.08	58.76	29.90	16.46	246.48	mean
	2.87	1.22	0.61	0.50	1.69	sd
	122.00	57.00	29.00	16.00	244.00	min
	130.00	60.00	32.00	17.00	248.00	max
RotArb	130.00	60.00	30.00	17.00	248.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	130.00	60.00	30.00	17.00	248.00	min
	130.00	60.00	30.00	17.00	248.00	max
RPC 3	116.78	56.64	28.84	16.84	340.24	mean
	2.23	1.12	0.65	0.37	8.20	sd
	111.00	54.00	28.00	16.00	324.00	min
	121.00	59.00	30.00	17.00	361.00	max
RPC 4	111.46	55.12	28.42	16.84	375.98	mean
	2.52	1.35	1.13	0.47	9.12	sd
	106.00	53.00	26.00	16.00	350.00	min
	117.00	58.00	30.00	18.00	392.00	max
RPC 5	105.00	53.54	27.76	16.72	416.52	mean
	2.52	1.37	1.17	0.57	10.88	sd
	100.00	51.00	25.00	16.00	388.00	min
	111.00	57.00	30.00	18.00	453.00	max
RPC LGA 3	117.98	58.78	30.86	16.90	300.52	mean
	2.08	0.95	0.86	0.36	7.92	sd
	114.00	56.00	29.00	16.00	286.00	min
	123.00	61.00	32.00	18.00	320.00	max
RPC LGA 4	113.24	57.88	30.18	16.98	335.66	mean
	2.36	1.45	1.21	0.65	10.97	sd
	107.00	55.00	28.00	16.00	314.00	min
	118.00	61.00	33.00	18.00	368.00	max
RPC LGA 5	107.52	55.92	29.14	16.70	378.14	mean
	3.18	1.88	1.14	0.76	13.70	sd

method	numclus	larger10	larger20	larger30	single	stat
Orig	130	60	30	17	248	
	102.00	50.00	27.00	15.00	342.00	min
	115.00	60.00	32.00	18.00	409.00	max
RPN	120.88	57.40	28.96	16.86	317.38	mean
	2.10	1.11	0.73	0.35	8.44	sd
	116.00	55.00	28.00	16.00	299.00	min
	126.00	60.00	30.00	17.00	338.00	max
RPN LGA	120.22	58.88	31.04	16.88	287.20	mean
	1.92	0.92	0.92	0.33	7.22	sd
	115.00	57.00	29.00	16.00	269.00	min
	124.00	60.00	32.00	17.00	301.00	max
RPU 3	106.44	48.12	27.22	16.60	469.68	mean
	4.17	1.75	0.95	0.49	16.86	sd
	98.00	45.00	25.00	16.00	420.00	min
	116.00	51.00	29.00	17.00	511.00	max
RPU 4	98.50	45.10	26.62	16.62	530.96	mean
	4.40	1.75	1.18	0.49	16.43	sd
	89.00	41.00	24.00	16.00	498.00	min
	107.00	49.00	29.00	17.00	567.00	max
RPU 5	93.08	42.22	25.68	16.44	576.26	mean
	3.59	2.09	1.33	0.67	15.16	sd
	84.00	37.00	23.00	15.00	543.00	min
	100.00	46.00	29.00	18.00	619.00	max
RPU LGA 3	107.92	46.22	27.08	16.74	491.46	mean
	3.78	2.39	1.18	0.60	13.97	sd
	98.00	41.00	25.00	16.00	458.00	min
	116.00	51.00	29.00	18.00	519.00	max
RPU LGA 4	106.38	40.04	25.42	16.62	549.70	mean
	4.13	2.32	1.57	0.75	16.74	sd
	96.00	36.00	22.00	15.00	521.00	min
	114.00	45.00	30.00	19.00	603.00	max
RPU LGA 5	103.66	35.52	23.36	16.16	584.36	mean
	5.24	2.01	1.22	0.82	17.62	sd
	92.00	30.00	20.00	14.00	548.00	min
	115.00	39.00	26.00	18.00	623.00	max
StreetMask 30	127.96	58.50	32.16	18.04	244.92	mean
	1.18	1.04	1.27	0.75	3.14	sd
	125.00	56.00	29.00	17.00	239.00	min

method	numclus	larger10	larger20	larger30	single	stat
Orig	130	60	30	17	248	
	130.00	61.00	35.00	20.00	253.00	max
StreetMask 100	120.26	59.66	32.12	18.44	270.90	mean
	2.48	1.24	1.17	0.70	7.16	sd
	114.00	56.00	30.00	17.00	256.00	min
	125.00	62.00	34.00	20.00	287.00	max
VNE 50 3	132.20	62.80	32.22	18.66	213.08	mean
	2.42	1.50	1.07	1.02	8.87	sd
	126.00	59.00	30.00	16.00	188.00	min
	137.00	66.00	34.00	20.00	235.00	max
VNE 50 5	130.70	62.42	32.38	18.74	218.30	mean
	2.89	1.86	1.14	0.75	9.20	sd
	125.00	58.00	30.00	17.00	198.00	min
	137.00	66.00	35.00	20.00	238.00	max
VNE 100 3	127.56	63.98	33.16	20.54	200.88	mean
	3.62	1.81	1.15	0.95	12.87	sd
	120.00	60.00	30.00	18.00	175.00	min
	134.00	68.00	36.00	23.00	231.00	max
VNE 100 5	126.02	64.08	32.80	20.62	206.18	mean
	2.85	2.13	1.28	1.18	11.66	sd
	121.00	60.00	30.00	18.00	182.00	min
	133.00	70.00	36.00	23.00	235.00	max
VNS 50 3	133.46	60.92	31.34	17.96	227.16	mean
	1.89	1.37	1.30	0.99	5.90	sd
	130.00	58.00	28.00	16.00	215.00	min
	137.00	64.00	35.00	20.00	239.00	max
VNS 50 5	130.94	61.94	31.50	17.84	232.12	mean
	2.24	1.58	1.28	0.93	7.13	sd
	127.00	59.00	28.00	16.00	215.00	min
	136.00	65.00	35.00	19.00	250.00	max
VNS 100 3	135.18	65.24	31.92	19.06	194.40	mean
	2.32	1.60	1.21	1.06	8.26	sd
	130.00	62.00	29.00	17.00	175.00	min
	141.00	69.00	34.00	21.00	214.00	max
VNS 100 5	134.16	65.82	31.76	19.16	198.40	mean
	2.14	2.02	1.20	1.13	8.11	sd
	130.00	62.00	28.00	17.00	180.00	min
	140.00	70.00	34.00	22.00	217.00	max

method	numclus	larger10	larger20	larger30	single	stat
Orig	130	60	30	17	248	
VNE LGA 50 3	131.24	61.56	32.50	20.10	213.90	mean
	2.66	1.98	1.16	1.07	8.54	sd
	125.00	56.00	30.00	18.00	198.00	min
	138.00	66.00	35.00	22.00	234.00	max
VNE LGA 50 5	131.48	62.12	32.38	19.90	209.92	mean
	2.40	1.92	1.01	0.81	8.83	sd
	126.00	58.00	30.00	18.00	191.00	min
	138.00	66.00	35.00	22.00	233.00	max
VNE LGA 100 3	126.04	63.46	32.86	21.50	202.90	mean
	3.78	2.32	1.46	1.18	12.93	sd
	118.00	57.00	29.00	19.00	171.00	min
	134.00	68.00	35.00	24.00	230.00	max
VNE LGA 100 5	125.52	62.72	33.06	21.46	204.18	mean
	3.09	2.32	1.06	0.89	10.27	sd
	118.00	57.00	30.00	20.00	185.00	min
	132.00	68.00	36.00	23.00	223.00	max
VNS LGA 50 3	132.12	60.52	31.28	19.04	226.78	mean
	1.66	1.33	1.16	0.86	5.29	sd
	127.00	57.00	29.00	18.00	213.00	min
	135.00	64.00	34.00	22.00	242.00	max
VNS LGA 50 5	131.08	60.66	31.34	18.84	225.70	mean
	2.17	1.33	1.21	1.06	6.91	sd
	125.00	58.00	28.00	17.00	212.00	min
	136.00	64.00	33.00	21.00	240.00	max
VNS LGA 100 3	133.68	64.40	31.94	20.32	195.14	mean
	2.83	2.10	1.02	1.13	8.83	sd
	127.00	61.00	30.00	17.00	178.00	min
	140.00	69.00	34.00	23.00	215.00	max
VNS LGA 100 5	133.82	65.06	32.16	20.36	194.22	mean
	2.24	1.70	1.15	1.03	9.20	sd
	129.00	61.00	29.00	19.00	177.00	min
	139.00	69.00	34.00	22.00	212.00	max
Voronoi	135.00	61.00	32.00	19.00	235.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	135.00	61.00	32.00	19.00	235.00	min
	135.00	61.00	32.00	19.00	235.00	max
Grid 100	130.00	60.00	30.00	17.00	248.00	mean
	0.00	0.00	0.00	0.00	0.00	sd

method	numclus	larger10	larger20	larger30	single	stat
Orig	130	60	30	17	248	
	130.00	60.00	30.00	17.00	248.00	min
	130.00	60.00	30.00	17.00	248.00	max
Grid 1000	126.00	59.00	31.00	17.00	246.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	126.00	59.00	31.00	17.00	246.00	min
	126.00	59.00	31.00	17.00	246.00	max

Table G.13.: Results of DBSCAN clustering algorithm ( $\varepsilon = 9500$ ).

method	numclus	larger10	larger20	larger30	single	stat
Orig	97	53	29	16	134	
APA	19.00	19.00	19.00	19.00	0.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	19.00	19.00	19.00	19.00	0.00	min
	19.00	19.00	19.00	19.00	0.00	max
APA LGA	54.00	51.00	40.00	32.00	4.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	54.00	51.00	40.00	32.00	4.00	min
	54.00	51.00	40.00	32.00	4.00	max
ARP	54.42	14.18	6.76	4.32	858.82	mean
	5.51	1.93	1.19	0.91	18.15	sd
	42.00	10.00	4.00	3.00	815.00	min
	71.00	19.00	10.00	6.00	909.00	max
ARP LGA	41.80	15.82	11.16	8.20	365.74	mean
	4.54	1.89	1.25	1.14	18.15	sd
	34.00	12.00	8.00	5.00	335.00	min
	54.00	21.00	14.00	10.00	411.00	max
CS	74.40	38.68	20.66	12.48	118.26	mean
	42.17	20.31	9.74	5.19	74.66	sd
	1.00	1.00	1.00	1.00	0.00	min
	116.00	55.00	30.00	17.00	222.00	max
DD 3	91.64	52.80	27.32	17.28	143.94	mean
	2.21	1.28	0.89	0.57	7.05	sd
	87.00	50.00	25.00	16.00	127.00	min
	97.00	55.00	30.00	19.00	159.00	max
DD 4	89.90	51.80	27.06	17.50	153.08	mean
	2.15	1.39	1.04	0.61	7.68	sd
	86.00	49.00	24.00	16.00	136.00	min

method	numclus	larger10	larger20	larger30	single	stat
Orig	97	53	29	16	134	
	93.00	55.00	29.00	19.00	173.00	max
DD 5	86.74	50.60	26.78	17.38	163.32	mean
	2.55	1.68	1.13	0.75	9.39	sd
	81.00	47.00	24.00	15.00	140.00	min
	91.00	54.00	30.00	19.00	190.00	max
DD LGA 3	87.02	51.74	29.18	18.08	139.14	mean
	1.50	0.88	0.60	0.75	5.57	sd
	83.00	50.00	28.00	17.00	127.00	min
	90.00	54.00	30.00	19.00	150.00	max
DD LGA 4	84.76	51.22	28.86	17.96	144.02	mean
	1.79	1.18	1.09	0.81	7.40	sd
	81.00	49.00	27.00	16.00	129.00	min
	89.00	54.00	32.00	20.00	160.00	max
DD LGA 5	81.96	50.32	28.34	17.68	147.24	mean
	1.69	1.46	1.17	0.74	6.81	sd
	79.00	47.00	25.00	16.00	133.00	min
	86.00	53.00	30.00	19.00	161.00	max
Dk 5	96.22	52.82	29.08	16.08	134.10	mean
	0.68	0.75	0.27	0.27	0.93	sd
	95.00	52.00	29.00	16.00	133.00	min
	97.00	55.00	30.00	17.00	137.00	max
Dk 25	96.18	52.78	29.18	16.18	134.30	mean
	0.75	0.84	0.39	0.39	1.42	sd
	95.00	52.00	29.00	16.00	133.00	min
	98.00	55.00	30.00	17.00	137.00	max
Dk 50	96.14	52.36	29.04	16.04	130.86	mean
	0.97	1.10	0.53	0.53	3.63	sd
	95.00	50.00	28.00	15.00	123.00	min
	98.00	55.00	30.00	17.00	139.00	max
Dk 100	96.92	52.48	29.16	16.12	126.28	mean
	1.35	1.23	0.74	0.56	4.63	sd
	93.00	49.00	27.00	15.00	117.00	min
	100.00	55.00	31.00	17.00	135.00	max
Dk 500	63.92	47.68	28.12	16.92	208.84	mean
	2.95	1.94	1.61	1.10	8.48	sd
	58.00	44.00	25.00	14.00	187.00	min
	72.00	54.00	31.00	20.00	228.00	max

method	numclus	larger10	larger20	larger30	single	stat
Orig	97	53	29	16	134	
Dk 1000	49.36	34.76	25.42	16.48	287.20	mean
	3.95	1.68	1.43	1.39	12.95	sd
	42.00	31.00	22.00	14.00	257.00	min
	57.00	38.00	29.00	20.00	312.00	max
DkData 20	46.16	28.28	24.64	16.64	330.70	mean
	3.18	1.91	1.59	1.31	15.67	sd
	40.00	24.00	20.00	15.00	291.00	min
	54.00	33.00	27.00	19.00	365.00	max
DUT	97.00	53.00	29.00	16.00	134.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	97.00	53.00	29.00	16.00	134.00	min
	97.00	53.00	29.00	16.00	134.00	max
ISGP	96.36	52.36	29.00	16.00	134.00	mean
	0.49	0.49	0.00	0.00	0.00	sd
	96.00	52.00	29.00	16.00	134.00	min
	97.00	53.00	29.00	16.00	134.00	max
Lipschitz Embedding	96.68	52.68	29.00	16.00	133.92	mean
	0.47	0.47	0.00	0.00	0.40	sd
	96.00	52.00	29.00	16.00	132.00	min
	97.00	53.00	29.00	16.00	134.00	max
LS 3	96.76	53.16	29.28	16.26	135.66	mean
	0.85	0.84	0.45	0.44	3.02	sd
	95.00	51.00	29.00	16.00	131.00	min
	99.00	55.00	30.00	17.00	142.00	max
LS 4	96.48	53.30	29.34	16.12	135.88	mean
	1.28	0.76	0.52	0.33	3.27	sd
	94.00	52.00	29.00	16.00	129.00	min
	100.00	55.00	31.00	17.00	144.00	max
LS 5	95.86	53.60	29.42	16.18	136.02	mean
	0.97	0.78	0.67	0.39	4.49	sd
	93.00	52.00	28.00	16.00	127.00	min
	98.00	55.00	31.00	17.00	147.00	max
LS LGA 3	96.18	52.24	29.10	17.58	136.82	mean
	0.96	1.10	0.54	0.61	3.01	sd
	95.00	50.00	28.00	17.00	129.00	min
	99.00	55.00	31.00	19.00	143.00	max
LS LGA 4	95.90	52.02	29.40	17.54	130.76	mean
	1.20	1.08	0.76	0.58	3.43	sd

method	numclus	larger10	larger20	larger30	single	stat
Orig	97	53	29	16	134	
	93.00	50.00	28.00	17.00	122.00	min
	98.00	54.00	31.00	19.00	140.00	max
LS LGA 5	95.58	52.16	29.46	17.50	132.44	mean
	1.28	0.93	0.81	0.71	3.29	sd
	92.00	50.00	28.00	16.00	127.00	min
	99.00	54.00	31.00	19.00	139.00	max
LSdonut 3	96.24	53.56	28.50	16.24	135.42	mean
	1.12	0.84	0.79	0.43	2.40	sd
	94.00	52.00	27.00	16.00	131.00	min
	98.00	55.00	30.00	17.00	141.00	max
LSdonut 4	94.38	54.24	29.48	16.14	140.18	mean
	1.28	0.69	0.74	0.35	2.62	sd
	92.00	53.00	28.00	16.00	136.00	min
	98.00	56.00	31.00	17.00	145.00	max
LSdonut 5	96.16	54.82	29.18	16.18	128.34	mean
	0.89	0.75	0.39	0.39	2.17	sd
	95.00	54.00	29.00	16.00	125.00	min
	98.00	57.00	30.00	17.00	132.00	max
LSdonut LGA 3	93.70	53.36	29.32	18.10	135.00	mean
	1.16	0.75	0.84	0.46	1.94	sd
	91.00	52.00	28.00	17.00	130.00	min
	97.00	55.00	31.00	19.00	139.00	max
LSdonut LGA 4	93.30	53.38	27.88	17.92	136.34	mean
	1.13	1.03	0.80	0.67	2.62	sd
	91.00	51.00	27.00	17.00	129.00	min
	96.00	55.00	30.00	19.00	142.00	max
LSdonut LGA 5	95.02	52.98	28.64	17.66	126.08	mean
	0.94	0.74	0.75	0.77	2.41	sd
	92.00	50.00	27.00	16.00	121.00	min
	97.00	54.00	30.00	19.00	130.00	max
MDAV 3	74.00	47.00	24.00	15.00	255.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	74.00	47.00	24.00	15.00	255.00	min
	74.00	47.00	24.00	15.00	255.00	max
MDAV 25	68.00	68.00	68.00	11.00	0.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	68.00	68.00	68.00	11.00	0.00	min

method	numclus	larger10	larger20	larger30	single	stat
Orig	97	53	29	16	134	
	68.00	68.00	68.00	11.00	0.00	max
MDAV 50	42.00	42.00	42.00	42.00	0.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	42.00	42.00	42.00	42.00	0.00	min
	42.00	42.00	42.00	42.00	0.00	max
Rot	93.92	52.14	28.60	16.22	126.84	mean
	3.24	1.31	0.76	0.42	10.39	sd
	88.00	49.00	27.00	16.00	99.00	min
	97.00	53.00	29.00	17.00	134.00	max
RotArb	96.90	52.90	29.00	16.00	134.00	mean
	0.30	0.30	0.00	0.00	0.00	sd
	96.00	52.00	29.00	16.00	134.00	min
	97.00	53.00	29.00	16.00	134.00	max
RPC 3	91.46	53.20	28.52	16.92	140.70	mean
	2.01	1.01	0.74	0.72	5.41	sd
	87.00	51.00	27.00	16.00	127.00	min
	96.00	55.00	31.00	19.00	150.00	max
RPC 4	90.00	52.80	27.50	17.16	146.32	mean
	1.77	1.03	0.79	0.71	6.80	sd
	86.00	50.00	26.00	15.00	133.00	min
	94.00	55.00	30.00	18.00	161.00	max
RPC 5	88.66	52.20	26.84	17.18	154.60	mean
	2.30	1.62	1.08	0.80	7.26	sd
	84.00	49.00	25.00	16.00	140.00	min
	94.00	55.00	29.00	19.00	173.00	max
RPC LGA 3	90.18	52.08	29.22	17.36	143.90	mean
	1.70	0.94	0.55	0.72	5.15	sd
	87.00	50.00	28.00	16.00	133.00	min
	94.00	54.00	30.00	19.00	155.00	max
RPC LGA 4	87.82	52.14	29.08	17.58	140.10	mean
	1.72	1.07	0.70	0.67	5.44	sd
	84.00	50.00	28.00	16.00	129.00	min
	92.00	55.00	31.00	19.00	152.00	max
RPC LGA 5	85.20	51.28	28.66	17.80	141.08	mean
	1.97	1.21	0.85	0.67	7.48	sd
	82.00	49.00	27.00	16.00	126.00	min
	90.00	54.00	30.00	19.00	156.00	max

method	numclus	larger10	larger20	larger30	single	stat
Orig	97	53	29	16	134	
RPN	92.06	53.04	29.04	16.52	137.86	mean
	1.36	0.73	0.60	0.58	4.56	sd
	89.00	51.00	28.00	16.00	126.00	min
	95.00	54.00	31.00	18.00	148.00	max
RPN LGA	90.80	52.16	29.02	17.20	145.58	mean
	1.21	0.74	0.47	0.70	5.03	sd
	89.00	50.00	28.00	16.00	131.00	min
	94.00	54.00	31.00	19.00	158.00	max
RPU 3	89.12	51.02	26.66	17.32	150.08	mean
	2.23	1.53	0.96	0.62	6.62	sd
	85.00	48.00	25.00	16.00	137.00	min
	94.00	56.00	28.00	18.00	166.00	max
RPU 4	84.90	48.88	26.32	16.98	159.96	mean
	2.45	1.92	1.49	0.96	9.60	sd
	80.00	44.00	23.00	15.00	137.00	min
	91.00	52.00	30.00	19.00	176.00	max
RPU 5	79.02	46.54	26.22	16.76	181.46	mean
	2.80	2.00	1.42	1.00	11.20	sd
	74.00	42.00	23.00	15.00	152.00	min
	85.00	51.00	29.00	19.00	210.00	max
RPU LGA 3	84.32	50.74	28.60	17.74	142.64	mean
	2.09	1.50	1.09	0.83	7.91	sd
	81.00	48.00	26.00	16.00	126.00	min
	89.00	54.00	31.00	19.00	160.00	max
RPU LGA 4	77.30	48.74	27.22	17.58	163.44	mean
	2.31	1.47	1.33	0.81	9.22	sd
	73.00	46.00	25.00	16.00	146.00	min
	83.00	53.00	30.00	20.00	187.00	max
RPU LGA 5	73.04	46.50	26.34	17.74	170.68	mean
	3.19	1.81	1.44	0.80	8.39	sd
	67.00	43.00	24.00	16.00	153.00	min
	81.00	50.00	29.00	19.00	193.00	max
StreetMask 30	94.78	53.44	28.08	18.38	128.50	mean
	1.15	0.86	0.67	0.53	2.50	sd
	93.00	52.00	27.00	17.00	124.00	min
	98.00	55.00	29.00	19.00	135.00	max
StreetMask 100	91.98	52.82	27.14	18.36	126.64	mean
	1.70	1.53	0.97	0.66	7.89	sd

method	numclus	larger10	larger20	larger30	single	stat
Orig	97	53	29	16	134	
	89.00	50.00	25.00	17.00	111.00	min
	96.00	56.00	29.00	20.00	143.00	max
VNE 50 3	99.52	53.54	29.48	16.18	102.20	mean
	2.32	1.64	0.97	0.44	7.08	sd
	94.00	51.00	27.00	15.00	86.00	min
	104.00	57.00	31.00	17.00	116.00	max
VNE 50 5	97.78	53.16	30.16	16.22	106.52	mean
	2.13	1.53	0.89	0.55	7.16	sd
	94.00	50.00	29.00	15.00	90.00	min
	102.00	56.00	32.00	18.00	121.00	max
VNE 100 3	95.50	53.68	28.96	17.16	100.32	mean
	2.66	1.35	1.12	1.11	9.79	sd
	89.00	50.00	26.00	15.00	82.00	min
	103.00	57.00	31.00	19.00	120.00	max
VNE 100 5	94.16	53.86	29.10	17.44	103.88	mean
	2.31	1.64	1.28	1.11	7.99	sd
	90.00	51.00	27.00	15.00	87.00	min
	100.00	57.00	32.00	20.00	122.00	max
VNS 50 3	98.28	53.34	29.04	16.14	118.98	mean
	1.16	0.94	0.67	0.35	5.00	sd
	96.00	52.00	28.00	16.00	109.00	min
	100.00	55.00	31.00	17.00	129.00	max
VNS 50 5	97.72	53.64	29.10	16.12	121.50	mean
	1.55	0.90	0.91	0.39	6.06	sd
	94.00	52.00	27.00	15.00	109.00	min
	101.00	56.00	31.00	17.00	135.00	max
VNS 100 3	99.48	54.86	28.60	16.58	95.48	mean
	1.57	0.99	0.81	0.67	6.55	sd
	96.00	53.00	27.00	16.00	82.00	min
	102.00	57.00	30.00	18.00	111.00	max
VNS 100 5	97.80	54.68	28.96	16.52	99.06	mean
	2.01	1.30	0.88	0.68	5.71	sd
	95.00	53.00	27.00	15.00	86.00	min
	103.00	58.00	31.00	18.00	112.00	max
VNE LGA 50 3	98.16	52.34	29.82	17.54	104.88	mean
	1.78	1.59	0.92	0.86	7.08	sd
	94.00	49.00	27.00	16.00	88.00	min

method	numclus	larger10	larger20	larger30	single	stat
Orig	97	53	29	16	134	
	102.00	57.00	32.00	19.00	122.00	max
VNE LGA 50 5	98.32	52.38	29.86	17.40	102.38	mean
	1.99	1.61	0.90	0.73	7.07	sd
	95.00	48.00	28.00	16.00	89.00	min
	103.00	55.00	32.00	19.00	115.00	max
VNE LGA 100 3	94.72	52.62	29.04	17.94	100.96	mean
	2.60	1.58	1.23	1.24	8.25	sd
	91.00	49.00	27.00	15.00	81.00	min
	101.00	56.00	31.00	21.00	120.00	max
VNE LGA 100 5	94.44	51.86	29.16	18.08	101.84	mean
	2.49	1.71	1.17	0.97	10.14	sd
	86.00	48.00	27.00	16.00	77.00	min
	99.00	57.00	31.00	20.00	120.00	max
VNS LGA 50 3	97.96	52.24	29.08	17.50	120.22	mean
	1.26	1.00	0.80	0.71	3.72	sd
	95.00	50.00	27.00	17.00	113.00	min
	101.00	54.00	31.00	19.00	129.00	max
VNS LGA 50 5	97.00	52.20	29.18	17.32	118.66	mean
	1.23	0.95	0.80	0.55	5.39	sd
	94.00	51.00	27.00	17.00	110.00	min
	100.00	55.00	31.00	19.00	132.00	max
VNS LGA 100 3	98.50	53.70	28.18	17.60	97.36	mean
	1.90	1.20	0.83	0.76	6.98	sd
	93.00	51.00	26.00	16.00	79.00	min
	103.00	56.00	30.00	19.00	115.00	max
VNS LGA 100 5	97.92	53.90	28.92	17.70	96.92	mean
	2.37	1.20	0.90	0.89	7.81	sd
	93.00	52.00	27.00	16.00	78.00	min
	103.00	57.00	31.00	20.00	116.00	max
Voronoi	102.00	54.00	30.00	18.00	116.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	102.00	54.00	30.00	18.00	116.00	min
	102.00	54.00	30.00	18.00	116.00	max
Grid 100	96.00	52.00	29.00	16.00	134.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	96.00	52.00	29.00	16.00	134.00	min
	96.00	52.00	29.00	16.00	134.00	max

method	numclus	larger10	larger20	larger30	single	stat
Orig	97	53	29	16	134	
Grid 1000	95.00	53.00	29.00	16.00	134.00	mean
	0.00	0.00	0.00	0.00	0.00	sd
	95.00	53.00	29.00	16.00	134.00	min
	95.00	53.00	29.00	16.00	134.00	max

Table G.14.: Number of points changing from clustered to non-clustered and vice versa.

masking methods	$\varepsilon = 3,200$		$\varepsilon = 9,500$	
	clus. in orig.	not in orig.	clus. in orig.	not in orig.
	not in mask	clus. in mask	not in mask	clus. in mask
APA	0.000	248.000	0.000	134.000
APA LGA	0.000	244.000	0.000	130.000
ARP	1,789.760	46.500	784.080	59.260
ARP LGA	1,092.420	54.820	289.500	57.760
CS	8.080	52.700	24.860	40.600
DUT	0.000	0.000	0.000	0.000
DD 3	157.100	5.940	28.860	18.920
DD 4	193.920	7.660	41.240	22.160
DD 5	236.980	10.800	55.940	26.620
DD LGA 3	116.820	6.500	23.360	18.220
DD LGA 4	166.820	9.140	29.860	19.840
DD LGA 5	211.700	11.720	37.560	24.320
Dk 5	0.000	0.060	0.240	0.140
Dk 25	0.060	0.740	0.800	0.500
Dk 50	0.680	2.240	2.380	5.520
Dk 100	1.620	5.900	5.460	13.180
Dk 500	255.880	16.260	104.660	29.820
Dk 1000	521.560	22.300	190.420	37.220
DkData 20	621.820	19.380	236.600	39.900
ISGP	0.000	0.000	0.000	0.000
Lipschitz Embedding	0.000	0.040	0.000	0.080
LS 3	4.840	5.560	4.240	2.580
LS 4	6.560	6.640	5.180	3.300
LS 5	12.800	9.080	7.700	5.680
LS LGA 3	4.960	6.060	6.640	3.820
LS LGA 4	5.720	10.840	5.580	8.820
LS LGA 5	8.000	12.760	7.740	9.300
LSdonut 3	6.460	5.820	4.840	3.420
LSdonut 4	16.780	8.360	8.320	2.140
LSdonut 5	12.720	17.880	3.300	8.960
LSdonut LGA 3	4.760	6.000	3.980	2.980
LSdonut LGA 4	17.200	10.500	9.480	7.140

masking methods	$\varepsilon = 3, 200$		$\varepsilon = 3, 200$	
	clus. in orig.	not in orig.	clus. in orig	not in orig
	not in mask	clus. in mask	not in mask	clus. in mask
LSdonut LGA 5	5.660	17.340	3.980	11.900
MDAV 3	309.000	26.000	141.000	20.000
MDAV 25	0.000	248.000	0.000	134.000
MDAV 50	0.000	248.000	0.000	134.000
Rot	0.000	1.520	0.000	7.160
RotArb	0.000	0.000	0.000	0.000
RPC 3	96.320	4.080	17.800	11.100
RPC 4	133.320	5.340	29.020	16.700
RPC 5	175.480	6.960	42.680	22.080
RPC LGA 3	56.540	4.020	19.680	9.780
RPC LGA 4	93.580	5.920	22.200	16.100
RPC LGA 5	138.720	8.580	27.100	20.020
RPN	72.060	2.680	13.380	9.520
RPN LGA	42.680	3.480	19.240	7.660
RPU 3	232.000	10.320	39.940	23.860
RPU 4	295.800	12.840	52.080	26.120
RPU 5	343.420	15.160	75.920	28.460
RPU LGA 3	253.960	10.500	33.440	24.800
RPU LGA 4	313.300	11.600	55.560	26.120
RPU LGA 5	348.680	12.320	66.380	29.700
StreetMask 30	7.360	10.440	1.540	7.040
StreetMask 100	42.540	19.640	8.880	16.240
VNE 50 3	23.640	58.560	8.060	39.860
VNE 50 5	30.460	60.160	12.660	40.140
VNE 100 3	52.680	99.800	25.620	59.300
VNE 100 5	59.620	101.440	30.400	60.520
VNS 50 3	5.440	26.280	3.880	18.900
VNS 50 5	13.440	29.320	9.540	22.040
VNS 100 3	11.220	64.820	6.980	45.500
VNS 100 5	17.360	66.960	11.040	45.980
VNE LGA 50 3	22.580	56.680	8.780	37.900
VNE LGA 50 5	24.660	62.740	10.400	42.020
VNE LGA 100 3	53.760	98.860	26.460	59.500
VNE LGA 100 5	55.200	99.020	28.880	61.040
VNS LGA 50 3	5.260	26.480	6.080	19.860
VNS LGA 50 5	7.800	30.100	7.940	23.280
VNS LGA 100 3	12.140	65.000	8.640	45.280
VNS LGA 100 5	13.060	66.840	10.100	47.180
Voronoi	0.000	13.000	8.000	26.000
Grid 100	0.000	0.000	0.000	0.000
Grid 1000	0.000	2.000	0.000	0.000

## G.9. Minimum Distance Risk Method

Table G.15.: Average precision and recall for minimum distance.

masking methods	$n = 1,000$		$n = 2,000$	
	precision	recall	precision	recall
APA	0.00	0.00	0.00	0.00
APA LGA	0.20	0.01	0.00	0.00
ARP	0.00	0.01	0.00	0.00
ARP LGA	0.01	0.02	0.00	0.01
DD 3	0.12	0.37	0.12	0.37
DD 4	0.12	0.36	0.11	0.34
DD 5	0.11	0.34	0.10	0.32
DD LGA 3	0.11	0.35	0.11	0.34
DD LGA 4	0.11	0.34	0.10	0.33
DD LGA 5	0.11	0.34	0.10	0.31
Dk 5	0.12	0.39	0.16	0.48
Dk 25	0.12	0.39	0.15	0.46
Dk 50	0.12	0.39	0.14	0.43
Dk 100	0.12	0.39	0.12	0.38
Dk 500	0.10	0.31	0.07	0.23
Dk 1000	0.08	0.25	0.05	0.14
DkData 20	0.07	0.22	0.04	0.13
Grid 100	0.13	0.40	0.15	0.46
Grid 1000	0.09	0.21	0.10	0.15
LS 3	0.12	0.35	0.12	0.38
LS 4	0.11	0.35	0.11	0.35
LS 5	0.11	0.34	0.10	0.32
LS LGA 3	0.11	0.35	0.12	0.37
LS LGA 4	0.11	0.34	0.11	0.33
LS LGA 5	0.11	0.34	0.10	0.30
LSdonut 3	0.12	0.36	0.11	0.35
LSdonut 4	0.11	0.35	0.10	0.31
LSdonut 5	0.11	0.34	0.10	0.31
LSdonut LGA 3	0.11	0.35	0.11	0.34
LSdonut LGA 4	0.11	0.33	0.10	0.32
LSdonut LGA 5	0.10	0.32	0.09	0.28
MDAV 3	0.11	0.34	0.09	0.26
MDAV 25	0.04	0.03	0.00	0.00
MDAV 50	0.00	0.00	0.00	0.00
RPC 3	0.12	0.37	0.13	0.41

masking methods	$n = 1,000$		$n = 2,000$	
	precision	recall	precision	recall
RPC 4	0.12	0.37	0.12	0.38
RPC 5	0.12	0.36	0.12	0.37
RPC LGA 3	0.11	0.36	0.13	0.40
RPC LGA 4	0.11	0.36	0.12	0.38
RPC LGA 5	0.11	0.35	0.11	0.35
RPN	0.12	0.37	0.14	0.42
RPN LGA	0.11	0.36	0.13	0.41
RPU 3	0.12	0.36	0.10	0.33
RPU 4	0.11	0.35	0.10	0.30
RPU 5	0.10	0.33	0.09	0.29
RPU LGA 3	0.11	0.34	0.10	0.31
RPU LGA 4	0.11	0.33	0.09	0.28
RPU LGA 5	0.10	0.33	0.08	0.26
StreetMask 30	0.12	0.36	0.11	0.34
StreetMask 100	0.10	0.30	0.07	0.22
VNE 50 3	0.11	0.34	0.09	0.28
VNE 50 5	0.11	0.32	0.08	0.25
VNE 100 3	0.10	0.32	0.08	0.23
VNE 100 5	0.10	0.29	0.07	0.22
VNS 50 3	0.11	0.34	0.10	0.32
VNS 50 5	0.11	0.34	0.09	0.28
VNS 100 3	0.11	0.33	0.08	0.26
VNS 100 5	0.11	0.32	0.08	0.25
VNE LGA 50 3	0.11	0.34	0.09	0.27
VNE LGA 50 5	0.10	0.32	0.08	0.24
VNE LGA 100 3	0.10	0.30	0.07	0.23
VNE LGA 100 5	0.09	0.29	0.07	0.21
VNS LGA 50 3	0.11	0.35	0.10	0.30
VNS LGA 50 5	0.11	0.33	0.09	0.27
VNS LGA 100 3	0.11	0.32	0.08	0.26
VNS LGA 100 5	0.10	0.31	0.08	0.24
Voronoi	0.12	0.38	0.15	0.44

## G.10. Hungarian Algorithm

Table G.16.: Average precision and recall of the Hungarian Algorithm.

masking methods	$n = 1,000$		$n = 2,000$	
	precision	recall	precision	recall
APA	0.00	0.00	0.00	0.00
APA LGA	0.00	0.00	0.00	0.00
ARP	0.00	0.00	0.00	0.00
ARP LGA	0.00	0.00	0.00	0.00
DD 3	0.53	0.53	0.38	0.38
DD 4	0.50	0.50	0.36	0.36
DD 5	0.47	0.47	0.32	0.32
DD LGA 3	0.50	0.50	0.34	0.34
DD LGA 4	0.45	0.45	0.32	0.32
DD LGA 5	0.42	0.42	0.28	0.28
Dk 5	0.84	0.84	0.72	0.72
Dk 25	0.72	0.72	0.57	0.57
Dk 50	0.60	0.60	0.45	0.45
Dk 100	0.47	0.47	0.32	0.32
Dk 500	0.15	0.15	0.07	0.07
Dk 1000	0.07	0.07	0.02	0.02
DkData 20	0.09	0.09	0.04	0.04
Grid 100	0.80	0.80	0.69	0.69
Grid 1000	0.11	0.11	0.07	0.07
LS 3	0.58	0.58	0.42	0.42
LS 4	0.56	0.56	0.37	0.37
LS 5	0.50	0.50	0.31	0.31
LS LGA 3	0.54	0.54	0.39	0.39
LS LGA 4	0.49	0.49	0.32	0.32
LS LGA 5	0.43	0.43	0.27	0.27
LSdonut 3	0.58	0.58	0.37	0.37
LSdonut 4	0.54	0.54	0.33	0.33
LSdonut 5	0.48	0.48	0.29	0.29
LSdonut LGA 3	0.53	0.53	0.35	0.35
LSdonut LGA 4	0.45	0.45	0.30	0.30
LSdonut LGA 5	0.40	0.40	0.25	0.25
MDAV 3	0.42	0.42	0.19	0.19
MDAV 25	0.05	0.05	0.03	0.03
MDAV 50	0.02	0.02	0.00	0.00
RPC 3	0.63	0.63	0.49	0.49

masking methods	$n = 1,000$		$n = 2,000$	
	precision	recall	precision	recall
RPC 4	0.58	0.58	0.44	0.44
RPC 5	0.56	0.56	0.40	0.40
RPC LGA 3	0.59	0.59	0.46	0.46
RPC LGA 4	0.54	0.54	0.42	0.42
RPC LGA 5	0.50	0.50	0.37	0.37
RPN	0.63	0.63	0.50	0.50
RPN LGA	0.60	0.60	0.48	0.48
RPU 3	0.49	0.49	0.32	0.32
RPU 4	0.44	0.44	0.27	0.27
RPU 5	0.40	0.40	0.23	0.23
RPU LGA 3	0.40	0.40	0.27	0.27
RPU LGA 4	0.36	0.36	0.24	0.24
RPU LGA 5	0.32	0.32	0.20	0.20
StreetMask 30	0.37	0.37	0.19	0.19
StreetMask 100	0.17	0.17	0.07	0.07
VNE 50 3	0.30	0.30	0.18	0.18
VNE 50 5	0.30	0.30	0.16	0.16
VNE 100 3	0.18	0.18	0.10	0.10
VNE 100 5	0.19	0.19	0.10	0.10
VNS 50 3	0.42	0.42	0.24	0.24
VNS 50 5	0.39	0.39	0.22	0.22
VNS 100 3	0.28	0.28	0.15	0.15
VNS 100 5	0.28	0.28	0.14	0.14
VNE LGA 50 3	0.30	0.30	0.18	0.18
VNE LGA 50 5	0.29	0.29	0.16	0.16
VNE LGA 100 3	0.19	0.19	0.10	0.10
VNE LGA 100 5	0.19	0.19	0.10	0.10
VNS LGA 50 3	0.40	0.40	0.24	0.24
VNS LGA 50 5	0.36	0.36	0.21	0.21
VNS LGA 100 3	0.28	0.28	0.15	0.15
VNS LGA 100 5	0.27	0.27	0.14	0.14
Voronoi	0.58	0.58	0.50	0.50

## G.11. Hungarian Algorithm Using Additional Variables

Table G.17.: Average precision and recall of the Hungarian Algorithm with blocking by sex, age, and employment status

masking methods	$n = 1,000$		$n = 2,000$		$n = 10,000$	
	precision	recall	precision	recall	precision	recall
APA	0.18	0.18	0.13	0.13	0.02	0.02
APA LGA	0.18	0.18	0.04	0.04	0.02	0.02
ARP	0.16	0.16	0.07	0.07	0.02	0.02
ARP LGA	0.12	0.12	0.05	0.05	0.01	0.01
DD 3	0.82	0.82	0.70	0.70	0.61	0.61
DD 4	0.80	0.80	0.67	0.67	0.59	0.59
DD 5	0.78	0.78	0.66	0.66	0.57	0.57
DD LGA 3	0.78	0.78	0.68	0.68	0.58	0.58
DD LGA 4	0.75	0.75	0.66	0.66	0.56	0.56
DD LGA 5	0.73	0.73	0.65	0.65	0.55	0.55
Dk 5	0.96	0.96	0.93	0.93	0.88	0.88
Dk 25	0.95	0.95	0.91	0.91	0.83	0.83
Dk 50	0.93	0.93	0.88	0.88	0.77	0.77
Dk 100	0.89	0.89	0.83	0.83	0.70	0.70
Dk 500	0.78	0.78	0.67	0.67	0.42	0.42
Dk 1000	0.72	0.72	0.56	0.56	0.28	0.28
DkData 20	0.68	0.68	0.51	0.51	0.26	0.26
Grid 100	0.94	0.94	0.94	0.94	0.88	0.88
Grid 1000	0.85	0.85	0.64	0.64	0.34	0.34
LS 3	0.87	0.87	0.76	0.76	0.66	0.66
LS 4	0.84	0.84	0.74	0.74	0.63	0.63
LS 5	0.83	0.83	0.71	0.71	0.60	0.60
LS LGA 3	0.83	0.83	0.75	0.75	0.64	0.64
LS LGA 4	0.81	0.81	0.72	0.72	0.61	0.61
LS LGA 5	0.79	0.79	0.70	0.70	0.58	0.58
LSdonut 3	0.85	0.85	0.71	0.71	0.64	0.64
LSdonut 4	0.83	0.83	0.69	0.69	0.61	0.61
LSdonut 5	0.81	0.81	0.68	0.68	0.58	0.58
LSdonut LGA 3	0.80	0.80	0.69	0.69	0.61	0.61
LSdonut LGA 4	0.75	0.75	0.68	0.68	0.58	0.58
LSdonut LGA 5	0.75	0.75	0.67	0.67	0.56	0.56
MDAV 3	0.83	0.83	0.76	0.76	0.58	0.58
MDAV 25	0.57	0.57	0.42	0.42	0.17	0.17
MDAV 50	0.51	0.51	0.28	0.28	0.10	0.10

masking methods	$n = 1,000$		$n = 2,000$		$n = 10,000$	
	precision	recall	precision	recall	precision	recall
RPC 3	0.87	0.87	0.78	0.78	0.70	0.70
RPC 4	0.84	0.84	0.75	0.75	0.67	0.67
RPC 5	0.82	0.82	0.73	0.73	0.64	0.64
RPC LGA 3	0.82	0.82	0.77	0.77	0.68	0.68
RPC LGA 4	0.80	0.80	0.74	0.74	0.65	0.65
RPC LGA 5	0.78	0.78	0.72	0.72	0.62	0.62
RPN	0.88	0.88	0.80	0.80	0.71	0.71
RPN LGA	0.83	0.83	0.78	0.78	0.69	0.69
RPU 3	0.78	0.78	0.65	0.65	0.57	0.57
RPU 4	0.75	0.75	0.63	0.63	0.54	0.54
RPU 5	0.73	0.73	0.62	0.62	0.51	0.51
RPU LGA 3	0.72	0.72	0.63	0.63	0.54	0.54
RPU LGA 4	0.70	0.70	0.62	0.62	0.52	0.52
RPU LGA 5	0.68	0.68	0.61	0.61	0.49	0.49
StreetMask 30	0.87	0.87	0.79	0.79	0.61	0.61
StreetMask 100	0.82	0.82	0.68	0.68	0.41	0.41
VNE 50 3	0.82	0.82	0.71	0.71	0.51	0.51
VNE 50 5	0.80	0.80	0.67	0.67	0.49	0.49
VNE 100 3	0.79	0.79	0.66	0.66	0.42	0.42
VNE 100 5	0.78	0.78	0.65	0.65	0.41	0.41
VNS 50 3	0.84	0.84	0.73	0.73	0.60	0.60
VNS 50 5	0.81	0.81	0.69	0.69	0.57	0.57
VNS 100 3	0.82	0.82	0.71	0.71	0.54	0.54
VNS 100 5	0.79	0.79	0.67	0.67	0.52	0.52
VNE LGA 50 3	0.80	0.80	0.70	0.70	0.50	0.50
VNE LGA 50 5	0.77	0.77	0.66	0.66	0.48	0.48
VNE LGA 100 3	0.78	0.78	0.66	0.66	0.41	0.41
VNE LGA 100 5	0.75	0.75	0.63	0.63	0.40	0.40
VNS LGA 50 3	0.81	0.81	0.72	0.72	0.59	0.59
VNS LGA 50 5	0.79	0.79	0.68	0.68	0.56	0.56
VNS LGA 100 3	0.79	0.79	0.70	0.70	0.53	0.53
VNS LGA 100 5	0.77	0.77	0.67	0.67	0.51	0.51
Voronoi	0.92	0.92	0.88	0.88	0.78	0.78

## G.12. Graph Theoretic Linkage Attack

Table G.18.: Average precision and recall of the graph attack. For Lipschitz embedding best result for different values of  $\alpha$  was taken.

masking methods	$n = 1,000$		masking methods	$n = 1,000$	
	precision	recall		precision	recall
APA	0.24	0.69	MDAV 50	0.38	0.39
APA LGA	0.14	0.17	RPC 3	0.96	1.00
ARP	0.24	0.69	RPC 4	0.94	0.99
ARP LGA	0.14	0.18	RPC 5	0.93	0.99
DD 3	0.95	0.99	RPC LGA 3	0.94	0.98
DD 4	0.94	0.99	RPC LGA 4	0.93	0.98
DD 5	0.93	0.99	RPC LGA 5	0.90	0.96
DD LGA 3	0.95	0.97	RPN	0.98	0.60
DD LGA 4	0.94	0.97	RPN LGA	0.99	0.67
DD LGA 5	0.91	0.96	RPU 3	0.94	0.98
Dk 5	1.00	0.88	RPU 4	0.92	0.99
Dk 25	1.00	0.98	RPU 5	0.90	0.98
Dk 50	0.98	0.97	RPU LGA 3	0.93	0.97
Dk 100	0.97	0.97	RPU LGA 4	0.91	0.96
Dk 500	0.85	0.94	RPU LGA 5	0.87	0.95
Dk 1000	0.74	0.91	StreetMask 30	0.96	0.86
DkData 20	0.66	0.90	StreetMask 100	0.87	0.91
Grid 100	1.00	0.92	VNE 50 3	0.89	0.46
Grid 1000	0.81	0.87	VNE 50 5	0.89	0.66
ISGP	0.06	0.00	VNE 100 3	0.86	0.35
Lipschitz	0.12	0.01	VNE 100 5	0.85	0.49
LS 3	0.96	1.00	VNS 50 3	0.92	0.58
LS 4	0.94	0.99	VNS 50 5	0.91	0.80
LS 5	0.93	0.99	VNS 100 3	0.89	0.44
LS LGA 3	0.95	0.98	VNS 100 5	0.89	0.63
LS LGA 4	0.94	0.97	VNE LGA 50 3	0.89	0.54
LS LGA 5	0.91	0.96	VNE LGA 50 5	0.87	0.72
LSdonut 3	0.96	0.98	VNE LGA 100 3	0.85	0.40
LSdonut 4	0.95	0.98	VNE LGA 100 5	0.82	0.55
LSdonut 5	0.95	0.98	VNS LGA 50 3	0.92	0.67
LSdonut LGA 3	0.95	0.98	VNS LGA 50 5	0.90	0.85
LSdonut LGA 4	0.93	0.96	VNS LGA 100 3	0.89	0.51
LSdonut LGA 5	0.91	0.96	VNS LGA 100 5	0.87	0.70
MDAV 3	0.82	0.83	Voronoi	1.00	0.72
MDAV 25	0.58	0.63			

### G.13. Graph Matching Attack on Privacy-Preserving Record Linkage

Table G.19.: Average precision and recall of the ppl attack using SMM

masking methods	$n = 1,000$		$n = 1,000$ top 100		$n = 2,000$		$n = 2,000$ top 200	
	precision	recall	precision	recall	precision	recall	precision	recall
APA	0.01	0.06	0.01	0.01	0.00	0.03	0.00	0.00
APA LGA	0.01	0.07	0.02	0.02	0.00	0.01	0.00	0.00
ARP	0.02	0.10	0.02	0.02	0.01	0.05	0.01	0.01
ARP LGA	0.01	0.07	0.01	0.01	0.00	0.03	0.00	0.00
DD 3	0.06	0.40	0.17	0.17	0.04	0.33	0.07	0.07
DD 4	0.06	0.39	0.17	0.17	0.04	0.33	0.06	0.06
DD 5	0.05	0.37	0.17	0.17	0.04	0.33	0.07	0.07
DD LGA 3	0.06	0.39	0.17	0.17	0.04	0.35	0.07	0.07
DD LGA 4	0.06	0.39	0.17	0.17	0.04	0.35	0.06	0.06
DD LGA 5	0.06	0.39	0.17	0.17	0.04	0.34	0.06	0.06
Dk 5	0.06	0.41	0.18	0.18	0.04	0.35	0.05	0.05
Dk 25	0.06	0.39	0.15	0.15	0.04	0.29	0.04	0.04
Dk 50	0.05	0.36	0.15	0.15	0.04	0.29	0.06	0.06
Dk 100	0.06	0.39	0.15	0.15	0.03	0.28	0.03	0.03
Dk 500	0.02	0.13	0.05	0.05	0.01	0.09	0.01	0.01
Dk 1000	0.04	0.26	0.07	0.07	0.02	0.14	0.02	0.02
DkData 20	0.05	0.34	0.12	0.12	0.04	0.28	0.05	0.05
Grid 100	0.06	0.41	0.18	0.18	0.04	0.32	0.04	0.04
Grid 1000	0.05	0.32	0.12	0.12	0.04	0.36	0.08	0.08
ISGP	0.00	0.01	0.01	0.01	0.00	0.01	0.01	0.01
Lipschitz	0.00	0.03	0.00	0.00	0.00	0.01	0.00	0.00
LS 3	0.06	0.39	0.18	0.18	0.04	0.34	0.06	0.06
LS 4	0.06	0.39	0.18	0.18	0.04	0.34	0.06	0.06
LS 5	0.06	0.38	0.18	0.18	0.04	0.34	0.06	0.06
LS LGA 3	0.06	0.39	0.16	0.16	0.04	0.34	0.06	0.06
LS LGA 4	0.06	0.38	0.17	0.17	0.04	0.34	0.06	0.06
LS LGA 5	0.06	0.38	0.17	0.17	0.04	0.33	0.05	0.05
LSdonut 3	0.06	0.38	0.18	0.18	0.04	0.34	0.08	0.08
LSdonut 4	0.06	0.38	0.17	0.17	0.04	0.34	0.06	0.06
LSdonut 5	0.06	0.38	0.18	0.18	0.04	0.34	0.06	0.06
LSdonut LGA 3	0.06	0.39	0.16	0.16	0.04	0.35	0.07	0.07
LSdonut LGA 4	0.06	0.38	0.17	0.17	0.04	0.34	0.05	0.05
LSdonut LGA 5	0.06	0.38	0.17	0.17	0.04	0.33	0.06	0.06
MDAV 3	0.03	0.18	0.08	0.08	0.03	0.20	0.04	0.04
MDAV 25	0.04	0.27	0.11	0.11	0.04	0.29	0.05	0.05
MDAV 50	0.04	0.28	0.06	0.06	0.01	0.10	0.02	0.02
RandProj 100	0.01	0.05	0.00	0.00	0.00	0.02	0.00	0.00
RandProj 200	0.01	0.09	0.02	0.02	0.00	0.04	0.01	0.01
RandProj 300	0.01	0.10	0.02	0.02	0.01	0.05	0.01	0.01
RandProj 500	0.01	0.09	0.02	0.02	0.00	0.04	0.01	0.01
RandProj 1000	0.02	0.16	0.03	0.03	0.01	0.08	0.01	0.01
RPC 3	0.06	0.39	0.18	0.18	0.04	0.35	0.06	0.06

masking methods	$n = 1,000$		$n = 1,000$ top 100		$n = 2,000$		$n = 2,000$ top 200	
	precision	recall	precision	recall	precision	recall	precision	recall
RPC 4	0.06	0.39	0.18	0.18	0.04	0.34	0.07	0.07
RPC 5	0.06	0.39	0.17	0.17	0.04	0.35	0.07	0.07
RPC LGA 3	0.06	0.41	0.18	0.18	0.04	0.35	0.06	0.06
RPC LGA 4	0.06	0.39	0.17	0.17	0.04	0.35	0.07	0.07
RPC LGA 5	0.06	0.39	0.17	0.17	0.04	0.34	0.07	0.07
RPN	0.06	0.41	0.18	0.18	0.04	0.34	0.06	0.06
RPN LGA	0.06	0.40	0.17	0.17	0.04	0.35	0.06	0.06
RPU 3	0.06	0.38	0.17	0.17	0.04	0.32	0.06	0.06
RPU 4	0.06	0.39	0.16	0.16	0.04	0.32	0.07	0.07
RPU 5	0.05	0.37	0.16	0.16	0.04	0.32	0.06	0.06
RPU LGA 3	0.06	0.39	0.17	0.17	0.04	0.34	0.06	0.06
RPU LGA 4	0.05	0.37	0.17	0.17	0.04	0.33	0.06	0.06
RPU LGA 5	0.06	0.38	0.17	0.17	0.04	0.32	0.06	0.06
StreetMask 30	0.06	0.38	0.17	0.17	0.05	0.36	0.07	0.07
StreetMask 100	0.06	0.38	0.16	0.16	0.04	0.35	0.07	0.07
VNE 50 3	0.05	0.36	0.14	0.14	0.04	0.29	0.06	0.06
VNE 50 5	0.06	0.39	0.17	0.17	0.04	0.33	0.06	0.06
VNE 100 3	0.02	0.14	0.05	0.05	0.01	0.09	0.01	0.01
VNE 100 5	0.04	0.29	0.10	0.10	0.02	0.16	0.02	0.02
VNS 50 3	0.05	0.36	0.13	0.13	0.04	0.29	0.07	0.07
VNS 50 5	0.05	0.35	0.12	0.12	0.03	0.22	0.04	0.04
VNS 100 3	0.06	0.38	0.18	0.18	0.05	0.36	0.07	0.07
VNS 100 5	0.06	0.38	0.17	0.17	0.04	0.32	0.04	0.04
VNE LGA 50 3	0.02	0.14	0.03	0.03	0.01	0.09	0.01	0.01
VNE LGA 50 5	0.06	0.38	0.16	0.16	0.04	0.31	0.04	0.04
VNE LGA 100 3	0.05	0.37	0.14	0.14	0.03	0.24	0.02	0.02
VNE LGA 100 5	0.02	0.13	0.05	0.05	0.01	0.10	0.02	0.02
VNS LGA 50 3	0.05	0.34	0.13	0.13	0.02	0.17	0.01	0.01
VNS LGA 50 5	0.05	0.37	0.16	0.16	0.04	0.33	0.06	0.06
VNS LGA 100 3	0.05	0.37	0.17	0.17	0.04	0.35	0.06	0.06
VNS LGA 100 5	0.02	0.12	0.02	0.02	0.01	0.08	0.01	0.01
Voronoi	0.06	0.40	0.16	0.16	0.04	0.32	0.06	0.06

Table G.20.: Average precision and recall of the ppri attack using SHM

masking methods	$n = 1,000$		$n = 1,000$ top 100		$n = 2,000$		$n = 2,000$ top 200	
	precision	recall	precision	recall	precision	recall	precision	recall
APA	0.01	0.01	0.01	0.01	0.00	0.00	0.00	0.00
APA LGA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
ARP	0.01	0.01	0.01	0.01	0.01	0.00	0.01	0.00
ARP LGA	0.01	0.01	0.01	0.01	0.01	0.00	0.01	0.00
DD 3	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
DD 4	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
DD 5	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
DD LGA 3	0.04	0.03	0.04	0.03	0.04	0.01	0.04	0.01
DD LGA 4	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
DD LGA 5	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01

masking methods	$n = 1,000$		$n = 1,000$ top 100		$n = 2,000$		$n = 2,000$ top 200	
	precision	recall	precision	recall	precision	recall	precision	recall
Dk 5	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
Dk 25	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
Dk 50	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
Dk 100	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
Dk 500	0.02	0.02	0.02	0.02	0.03	0.01	0.03	0.01
Dk 1000	0.02	0.02	0.02	0.02	0.02	0.01	0.02	0.01
DkData 20	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
Grid 100	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
Grid 1000	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
ISGP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Lipschitz	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
LS 3	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
LS 4	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
LS 5	0.04	0.03	0.04	0.03	0.04	0.01	0.04	0.01
LS LGA 3	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
LS LGA 4	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
LS LGA 5	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
LSdonut 3	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
LSdonut 4	0.03	0.03	0.03	0.03	0.04	0.01	0.04	0.01
LSdonut 5	0.04	0.03	0.04	0.03	0.05	0.02	0.05	0.02
LSdonut LGA 3	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
LSdonut LGA 4	0.03	0.03	0.03	0.03	0.05	0.02	0.05	0.02
LSdonut LGA 5	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
MDAV 3	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
MDAV 25	0.01	0.01	0.01	0.01	0.07	0.02	0.07	0.02
MDAV 50	0.01	0.01	0.01	0.01	0.01	0.00	0.01	0.00
RandProj 100	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RandProj 200	0.01	0.00	0.01	0.00	0.01	0.01	0.01	0.01
RandProj 300	0.01	0.01	0.01	0.01	0.03	0.01	0.03	0.01
RandProj 500	0.00	0.00	0.00	0.00	0.02	0.01	0.02	0.01
RandProj 1000	0.01	0.01	0.01	0.01	0.03	0.01	0.03	0.01
RPC 3	0.03	0.03	0.03	0.03	0.04	0.01	0.04	0.01
RPC 4	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
RPC 5	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
RPC LGA 3	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
RPC LGA 4	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
RPC LGA 5	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
RPN	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
RPN LGA	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
RPU 3	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
RPU 4	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
RPU 5	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
RPU LGA 3	0.04	0.03	0.04	0.03	0.04	0.01	0.04	0.01
RPU LGA 4	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
RPU LGA 5	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
StreetMask 30	0.03	0.02	0.03	0.02	0.03	0.01	0.03	0.01
StreetMask 100	0.03	0.03	0.03	0.03	0.05	0.02	0.05	0.02
VNE 50 3	0.04	0.03	0.04	0.03	0.04	0.01	0.04	0.01

masking methods	$n = 1,000$		$n = 1,000$ top 100		$n = 2,000$		$n = 2,000$ top 200	
	precision	recall	precision	recall	precision	recall	precision	recall
VNE 50 5	0.03	0.03	0.03	0.03	0.04	0.01	0.04	0.01
VNE 100 3	0.02	0.02	0.02	0.02	0.03	0.01	0.03	0.01
VNE 100 5	0.04	0.03	0.04	0.03	0.02	0.01	0.02	0.01
VNS 50 3	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01
VNS 50 5	0.03	0.02	0.03	0.02	0.04	0.02	0.04	0.02
VNS 100 3	0.03	0.03	0.03	0.03	0.04	0.01	0.04	0.01
VNS 100 5	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
VNE LGA 50 3	0.02	0.01	0.02	0.01	0.03	0.01	0.03	0.01
VNE LGA 50 5	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
VNE LGA 100 3	0.03	0.02	0.03	0.02	0.05	0.02	0.05	0.02
VNE LGA 100 5	0.02	0.02	0.02	0.02	0.02	0.01	0.02	0.01
VNS LGA 50 3	0.02	0.02	0.02	0.02	0.03	0.01	0.03	0.01
VNS LGA 50 5	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
VNS LGA 100 3	0.03	0.02	0.03	0.02	0.04	0.01	0.04	0.01
VNS LGA 100 5	0.02	0.01	0.02	0.01	0.03	0.01	0.03	0.01
Voronoi	0.04	0.03	0.04	0.03	0.03	0.01	0.03	0.01

## H. Explanation for Summarizing GDi and LDi

All parts of the GDi and LDi are calculated for every masked data set. The results are gradually averaged to yield one value of GDi, Mdi, Odi, MAdi, LDi, Clus, SpatAutCorr for each masking method.

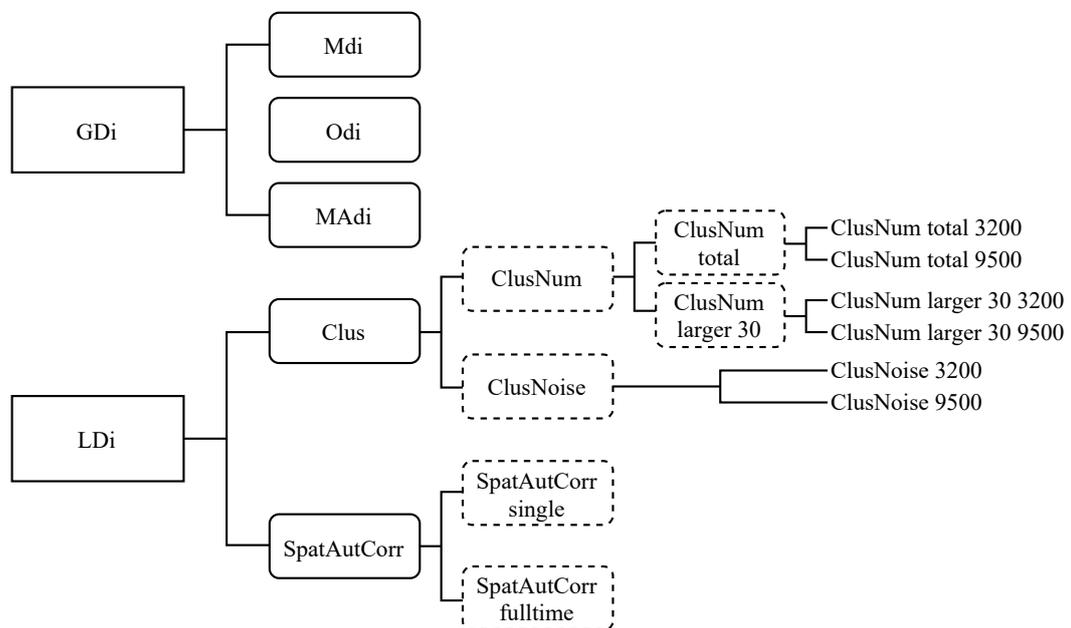


Figure H.1.: Explanation for summarizing the components of GDi and LDi (Kounadi and Leitner, 2015) for every masked data set.



# I. R-U-Map of Larger Samples

## I.1. MPR

Table I.1.: MPR for different parameter choices of masking methods

method	$n = 1,000$	$n = 2,000$	$n = 10,000$
APA	0.46	0.13	0.02
APA LGA	0.18	0.04	0.02
ARP	0.46	0.07	0.02
ARP LGA	0.16	0.05	0.01
CS	1.00	1.00	1.00
DD 3	0.97	0.70	0.61
DD 4	0.97	0.67	0.59
DD 5	0.96	0.66	0.57
DD LGA 3	0.96	0.68	0.58
DD LGA 4	0.95	0.66	0.56
DD LGA 5	0.94	0.65	0.55
Dk 5	0.96	0.93	0.88
Dk 25	0.99	0.91	0.83
Dk 50	0.98	0.88	0.77
Dk 100	0.97	0.83	0.70
Dk 500	0.89	0.67	0.42
Dk 1000	0.82	0.56	0.28
DkData 20	0.78	0.51	0.26
DUT	1.00	1.00	1.00
Grid 100	0.96	0.94	0.88
Grid 1000	0.85	0.64	0.34
ISGP	0.03	0.01	
Lipschitz Embedding	0.07	0.01	
LS 3	0.98	0.76	0.66
LS 4	0.97	0.74	0.63
LS 5	0.96	0.71	0.60
LS LGA 3	0.96	0.75	0.64
LS LGA 4	0.95	0.72	0.61
LS LGA 5	0.93	0.70	0.58

method	$n = 1,000$	$n = 2,000$	$n = 10,000$
LSdonut 3	0.97	0.71	0.64
LSdonut 4	0.97	0.69	0.61
LSdonut 5	0.96	0.68	0.58
LSdonut LGA 3	0.96	0.69	0.61
LSdonut LGA 4	0.95	0.68	0.58
LSdonut LGA 5	0.93	0.67	0.56
MDAV 3	0.83	0.76	0.58
MDAV 25	0.60	0.42	0.17
MDAV 50	0.51	0.28	0.10
RandProj 100	0.03	0.01	
RandProj 200	0.05	0.02	
RandProj 300	0.06	0.03	
RandProj 500	0.05	0.02	
RandProj 1000	0.09	0.04	
Rot	1.00	1.00	1.00
RotArb	1.00	1.00	1.00
RPC 3	0.98	0.78	0.70
RPC 4	0.97	0.75	0.67
RPC 5	0.96	0.73	0.64
RPC LGA 3	0.96	0.77	0.68
RPC LGA 4	0.95	0.74	0.65
RPC LGA 5	0.93	0.72	0.62
RPN	0.88	0.80	0.71
RPN LGA	0.83	0.78	0.69
RPU 3	0.96	0.65	0.57
RPU 4	0.96	0.63	0.54
RPU 5	0.94	0.62	0.51
RPU LGA 3	0.95	0.63	0.54
RPU LGA 4	0.93	0.62	0.52
RPU LGA 5	0.91	0.61	0.49
StreetMask 30	0.91	0.79	0.61
StreetMask 100	0.89	0.68	0.41
VNE 50 3	0.82	0.71	0.51
VNE 50 5	0.80	0.67	0.49
VNE 100 3	0.79	0.66	0.42
VNE 100 5	0.78	0.65	0.41
VNS 50 3	0.84	0.73	0.60
VNS 50 5	0.85	0.69	0.57
VNS 100 3	0.82	0.71	0.54

method	$n = 1,000$	$n = 2,000$	$n = 10,000$
VNS 100 5	0.79	0.67	0.52
VNE LGA 50 3	0.80	0.70	0.50
VNE LGA 50 5	0.79	0.66	0.48
VNE LGA 100 3	0.78	0.66	0.41
VNE LGA 100 5	0.75	0.63	0.40
VNS LGA 50 3	0.81	0.72	0.59
VNS LGA 50 5	0.87	0.68	0.56
VNS LGA 100 3	0.79	0.70	0.53
VNS LGA 100 5	0.78	0.67	0.51
Voronoi	0.92	0.88	0.78

## I.2. GDi and LDi

Table I.2.: Results of the GDi, LDi and its average for different parameter choices of masking methods

method	Mdi	Odi	MAdi	GDi	Cluster	Moran	LDi	avg.
APA	1.70	0.82	7.94	3.49	3.03	12.17	7.60	5.55
APA LGA	0.04	0.06	0.08	0.06	3.19	12.23	7.71	3.89
ARP	1.75	1.75	9.74	4.41	8.30	9.69	8.99	6.70
ARP LGA	0.06	0.31	0.26	0.21	5.00	9.61	7.31	3.76
CS	87.02	0.00	8.27	31.76	0.89	0.01	0.45	16.10
DD 3	0.00	0.02	0.01	0.01	0.73	2.84	1.79	0.90
DD 4	0.00	0.03	0.01	0.01	0.92	3.15	2.03	1.02
DD 5	0.00	0.04	0.01	0.02	1.11	3.42	2.27	1.14
DD LGA 3	0.00	0.02	0.01	0.01	0.66	3.06	1.86	0.94
DD LGA 4	0.00	0.03	0.01	0.02	0.86	3.36	2.11	1.06
DD LGA 5	0.00	0.04	0.01	0.02	1.02	3.66	2.34	1.18
Dk 5	0.00	0.00	0.00	0.00	0.01	0.10	0.05	0.03
Dk 25	0.00	0.00	0.00	0.00	0.03	0.23	0.13	0.07
Dk 50	0.00	0.03	0.01	0.01	0.07	0.52	0.29	0.15
Dk 100	0.00	0.03	0.01	0.01	0.11	0.97	0.54	0.28
Dk 500	0.00	0.09	0.04	0.04	1.54	2.80	2.17	1.11
Dk 1000	0.01	0.17	0.06	0.08	2.68	3.98	3.33	1.70
DkData 20	0.01	0.18	0.07	0.09	3.11	4.41	3.76	1.93
DUT	0.57	0.00	0.00	0.19	0.00	0.00	0.00	0.10
Grid 100	0.00	0.00	0.00	0.00	0.01	1.01	0.51	0.25
Grid 1000	0.00	0.00	0.00	0.00	0.04	6.46	3.25	1.62
ISGP	0.00	15.33	12.54	9.29	0.00	2.80	1.40	5.35
Lipschitz Embedding	0.00	2.22	2.07	1.43	0.00	0.43	0.22	0.82
LS 3	0.00	0.01	0.00	0.00	0.07	0.73	0.40	0.20

method	Mdi	Odi	MAdi	GDdi	Cluster	Moran	LDi	avg.
LS 4	0.00	0.01	0.00	0.00	0.09	0.88	0.48	0.24
LS 5	0.00	0.01	0.01	0.01	0.13	1.05	0.59	0.30
LS LGA 3	0.00	0.03	0.00	0.01	0.18	0.99	0.59	0.30
LS LGA 4	0.00	0.04	0.00	0.01	0.20	1.18	0.69	0.35
LS LGA 5	0.00	0.03	0.01	0.01	0.22	1.35	0.79	0.40
LSdonut 3	0.00	0.06	0.00	0.02	0.08	0.69	0.38	0.20
LSdonut 4	0.00	0.04	0.02	0.02	0.13	0.64	0.38	0.20
LSdonut 5	0.01	0.03	0.06	0.03	0.15	0.83	0.49	0.26
LSdonut LGA 3	0.00	0.05	0.00	0.02	0.23	0.88	0.56	0.29
LSdonut LGA 4	0.00	0.03	0.00	0.01	0.29	0.94	0.61	0.31
LSdonut LGA 5	0.00	0.05	0.01	0.02	0.22	0.98	0.60	0.31
MdAV 3	0.00	0.02	0.01	0.01	1.62	4.09	2.85	1.43
MdAV 25	0.00	0.05	0.14	0.06	1.65	8.69	5.17	2.62
MdAV 50	0.00	0.23	0.23	0.15	4.51	9.89	7.20	3.68
Rot	189.40	31.09	0.00	73.50	0.08	0.01	0.05	36.77
RotArb	0.00	28.62	0.00	9.54	0.00	0.01	0.00	4.77
RPC 3	0.00	0.02	0.01	0.01	0.46	2.34	1.40	0.71
RPC 4	0.00	0.02	0.01	0.01	0.65	2.72	1.69	0.85
RPC 5	0.00	0.03	0.01	0.01	0.86	3.04	1.95	0.98
RPC LGA 3	0.00	0.02	0.01	0.01	0.38	2.55	1.47	0.74
RPC LGA 4	0.00	0.02	0.01	0.01	0.56	2.93	1.75	0.88
RPC LGA 5	0.00	0.03	0.01	0.01	0.76	3.24	2.00	1.01
RPN	0.00	0.01	0.00	0.01	0.34	2.09	1.22	0.61
RPN LGA	0.00	0.02	0.00	0.01	0.32	2.28	1.30	0.65
RPU 3	0.00	0.03	0.01	0.01	1.00	3.08	2.04	1.03
RPU 4	0.00	0.03	0.01	0.02	1.25	3.51	2.38	1.20
RPU 5	0.00	0.05	0.01	0.02	1.51	3.85	2.68	1.35
RPU LGA 3	0.00	0.04	0.01	0.02	1.07	3.30	2.18	1.10
RPU LGA 4	0.00	0.04	0.02	0.02	1.33	3.72	2.53	1.27
RPU LGA 5	0.00	0.05	0.02	0.02	1.51	4.07	2.79	1.41
StreetMask 30	0.02	0.18	0.06	0.09	0.22	0.28	0.25	0.17
StreetMask 100	0.02	0.18	0.06	0.09	0.45	1.10	0.77	0.43
VNE 50 3	0.00	0.13	0.05	0.06	0.43	1.08	0.76	0.41
VNE 50 5	0.00	0.13	0.05	0.06	0.46	1.42	0.94	0.50
VNE 100 3	0.01	0.16	0.07	0.08	0.82	1.46	1.14	0.61
VNE 100 5	0.01	0.17	0.07	0.08	0.87	1.74	1.30	0.69
VNS 50 3	0.00	0.03	0.01	0.01	0.21	0.80	0.50	0.26
VNS 50 5	0.00	0.03	0.02	0.02	0.25	1.12	0.68	0.35
VNS 100 3	0.00	0.07	0.04	0.04	0.47	0.93	0.70	0.37
VNS 100 5	0.00	0.07	0.04	0.04	0.49	1.25	0.87	0.45
VNE LGA 50 3	0.00	0.11	0.05	0.05	0.52	1.33	0.93	0.49
VNE LGA 50 5	0.00	0.12	0.06	0.06	0.54	1.62	1.08	0.57
VNE LGA 100 3	0.01	0.13	0.07	0.07	0.89	1.68	1.28	0.68
VNE LGA 100 5	0.01	0.13	0.08	0.07	0.91	1.91	1.41	0.74

method	Mdi	Odi	MAdi	GDi	Cluster	Moran	LDi	avg.
VNS LGA 50 3	0.00	0.02	0.01	0.01	0.30	1.07	0.68	0.35
VNS LGA 50 5	0.00	0.02	0.02	0.02	0.31	1.40	0.85	0.43
VNS LGA 100 3	0.00	0.05	0.04	0.03	0.55	1.23	0.89	0.46
VNS LGA 100 5	0.00	0.04	0.04	0.03	0.57	1.51	1.04	0.53
Voronoi	0.00	0.02	0.02	0.01	0.33	10.99	5.66	2.84

### I.3. MSE

Table I.3.: Results of the MSE for different parameter choices of masking methods

method	MSE	method	MSE
APA	2,264,755,509.382	MDAV 50	844,553.693
APA LGA	112,191.097	Rot	113,198,247.705
ARP	2,764,585,860.404	RotArb	10,856.967
ARP LGA	5,523,238.437	RPC 3	601.997
CS	4,252,206,780.727	RPC 4	727.284
DD 3	921.718	RPC 5	1,678.389
DD 4	1,437.000	RPC LGA 3	737.494
DD 5	2,229.293	RPC LGA 4	976.020
DD LGA 3	1,084.258	RPC LGA 5	2,003.724
DD LGA 4	2,233.163	RPN	234.575
DD LGA 5	2,171.166	RPN LGA	375.115
Dk 5	0.310	RPU 3	1,725.227
Dk 25	0.817	RPU 4	2,548.437
Dk 50	456.619	RPU 5	3,124.657
Dk 100	885.016	RPU LGA 3	1,859.368
Dk 500	7,408.399	RPU LGA 4	3,454.354
Dk 1000	28,321.035	RPU LGA 5	6,374.773
DkData 20	35,720.647	StreetMask 30	154,462.188
DUT	22.767	StreetMask 100	165,221.173
Grid 100	91,770.448	VNE 50 3	20,476.348
Grid 1000	89,858.825	VNE 50 5	27,077.070
ISGP	3,624,144,345.490	VNE 100 3	58,635.054
Lipschitz Embedding	134,156,982.256	VNE 100 5	60,346.072
LS 3	101.410	VNS 50 3	1,348.737
LS 4	84.674	VNS 50 5	7,253.898
LS 5	2,720.610	VNS 100 3	11,946.843
LS LGA 3	258.708	VNS 100 5	15,918.568
LS LGA 4	854.195	VNE LGA 50 3	22,333.676

method	MSE	method	MSE
LS LGA 5	2,704.078	VNE LGA 50 5	33,423.223
LSdonut 3	489.526	VNE LGA 100 3	59,002.529
LSdonut 4	5,887.401	VNE LGA 100 5	78,308.873
LSdonut 5	51,391.007	VNS LGA 50 3	2,209.069
LSdonut LGA 3	366.897	VNS LGA 50 5	6,613.105
LSdonut LGA 4	595.821	VNS LGA 100 3	14,010.607
LSdonut LGA 5	7,981.494	VNS LGA 100 5	19,248.025
MDAV 3	2,126.188	Voronoi	8,284.462
MDAV 25	134,188.931		

#### I.4. Risk-Utility Maps

Figure I.1 shows the risk-utility map if the combined GD<sub>i</sub> and LD<sub>i</sub> is used as the utility measure and only the MPR for the larger subsample ( $n = 2,000$ ) are considered. As can be seen, for the affine transformation, the results are maintained. For the other methods, there is a decrease in the MPR. However, for most geomasking methods, the MPR is still above 0.5.

For the full sample ( $n = 10,000$ ), no MPR is available for the distance approximation using ISGP and the anonymization of distance matrices via Lipschitz embedding (see figure I.2). This is caused by the fact that only the Hungarian algorithm using additional variables can handle data sets of such large sizes. But even with the full sample, the majority of the methods reach an MPR of over 0.25.

Figure I.3 shows the risk-utility map if the MSE is used as the utility measure and the MPR of the larger subsample ( $n = 2,000$ ) is used. The risk-utility map for the full sample is shown in figure I.4. Comparing the MSE and the GD<sub>i</sub> and LD<sub>i</sub> as utility measures, some masking methods are evaluated differently. For example, street masking remains relatively close to the original data set according to the GD<sub>i</sub> and LD<sub>i</sub>, but not according to the MSE. Contrary, Voronoi masking shows a lower utility than other masking methods when using the GD<sub>i</sub> and LD<sub>i</sub>, but not when using the MSE.

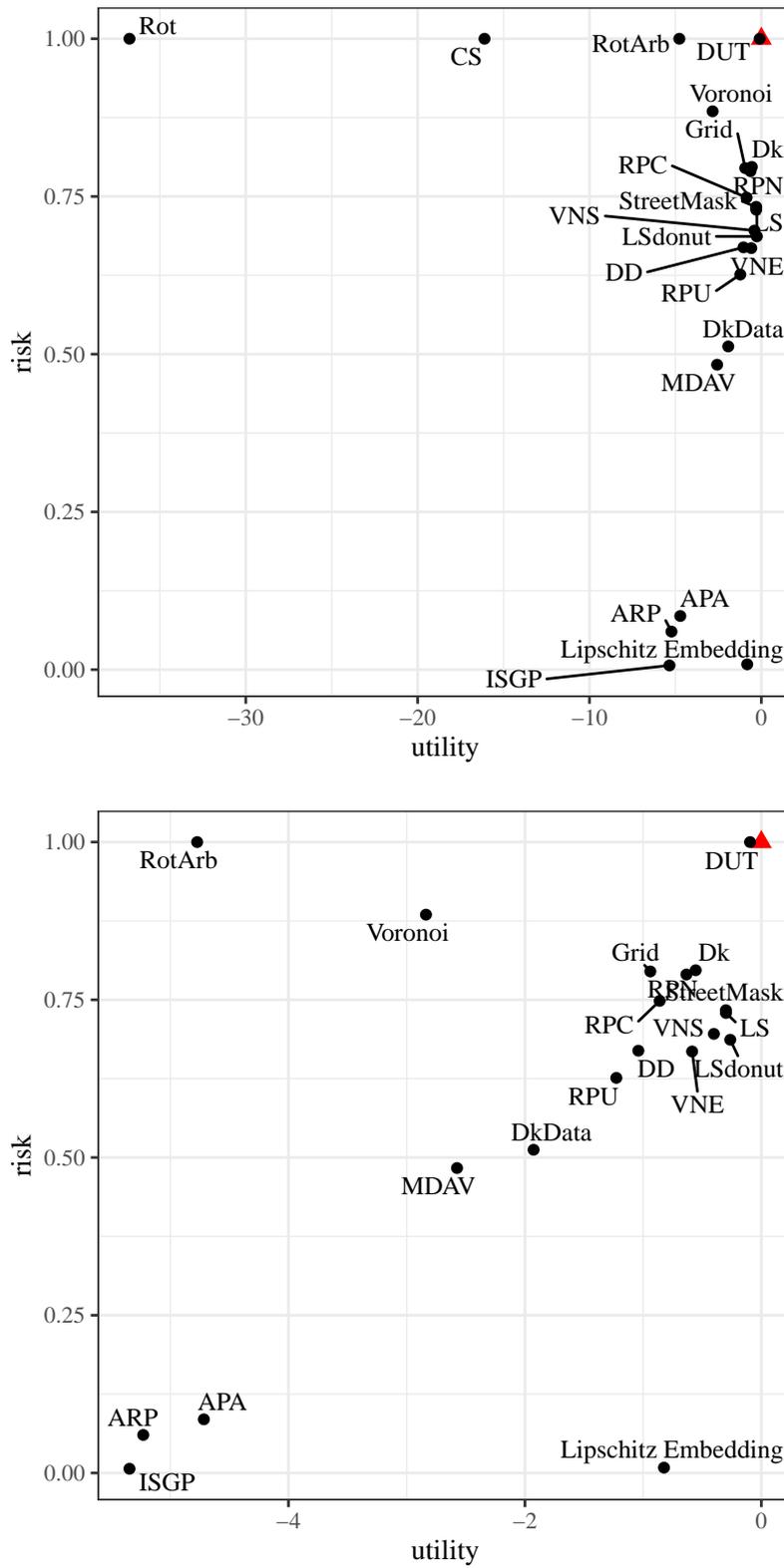


Figure I.1.: Risk-Utility Map of the larger subsample ( $n = 2,000$ ). GD<sub>i</sub> and LD<sub>i</sub> as utility measure. Red triangle shows original data. Bottom figure shows map without outliers (rotation and change of scale).

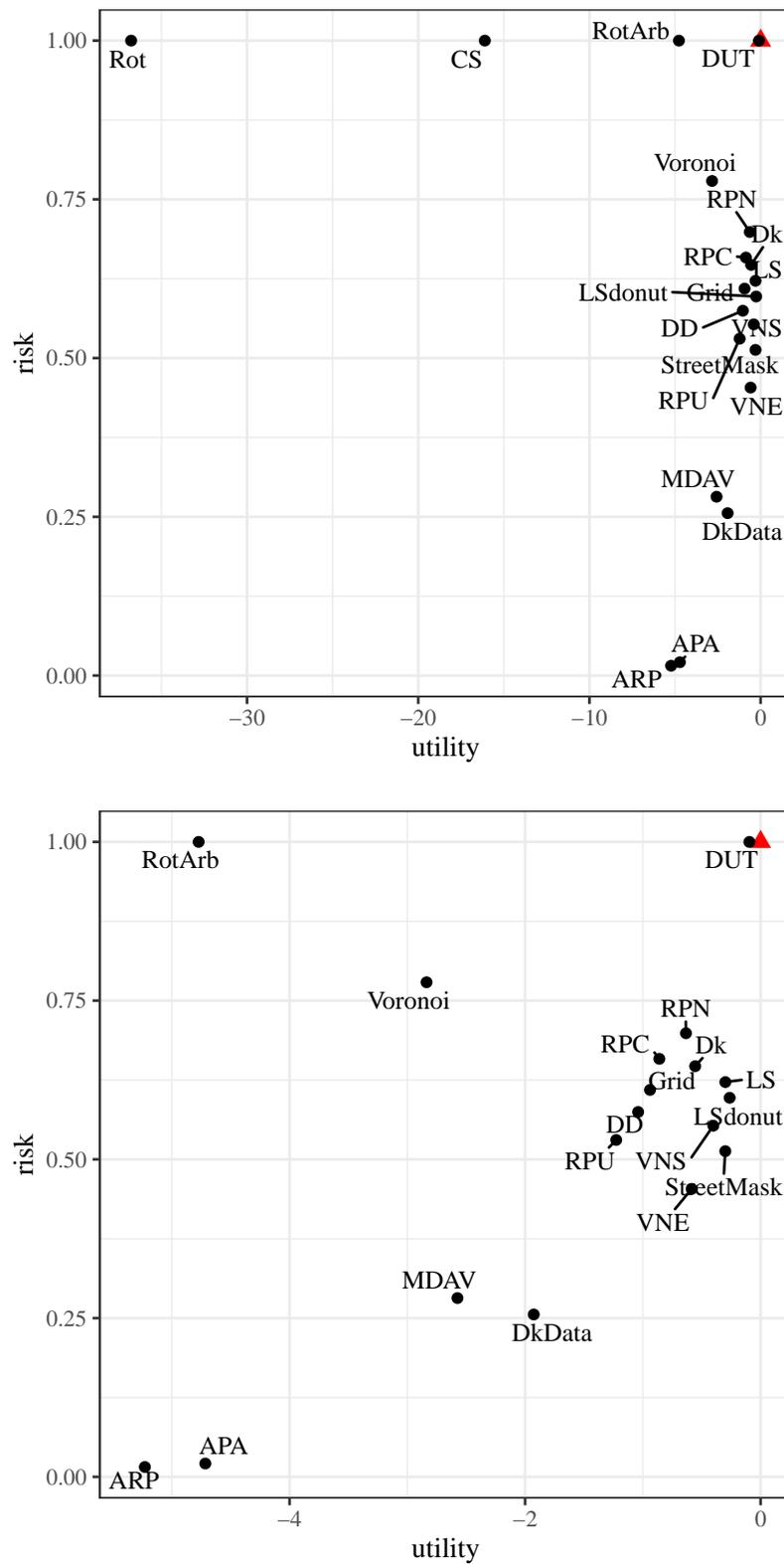


Figure I.2.: Risk-Utility Map of the full sample ( $n = 10,000$ ). GD<sub>i</sub> and LD<sub>i</sub> as utility measure. Red triangle shows original data. Bottom figure shows map without outliers (rotation and change of scale).

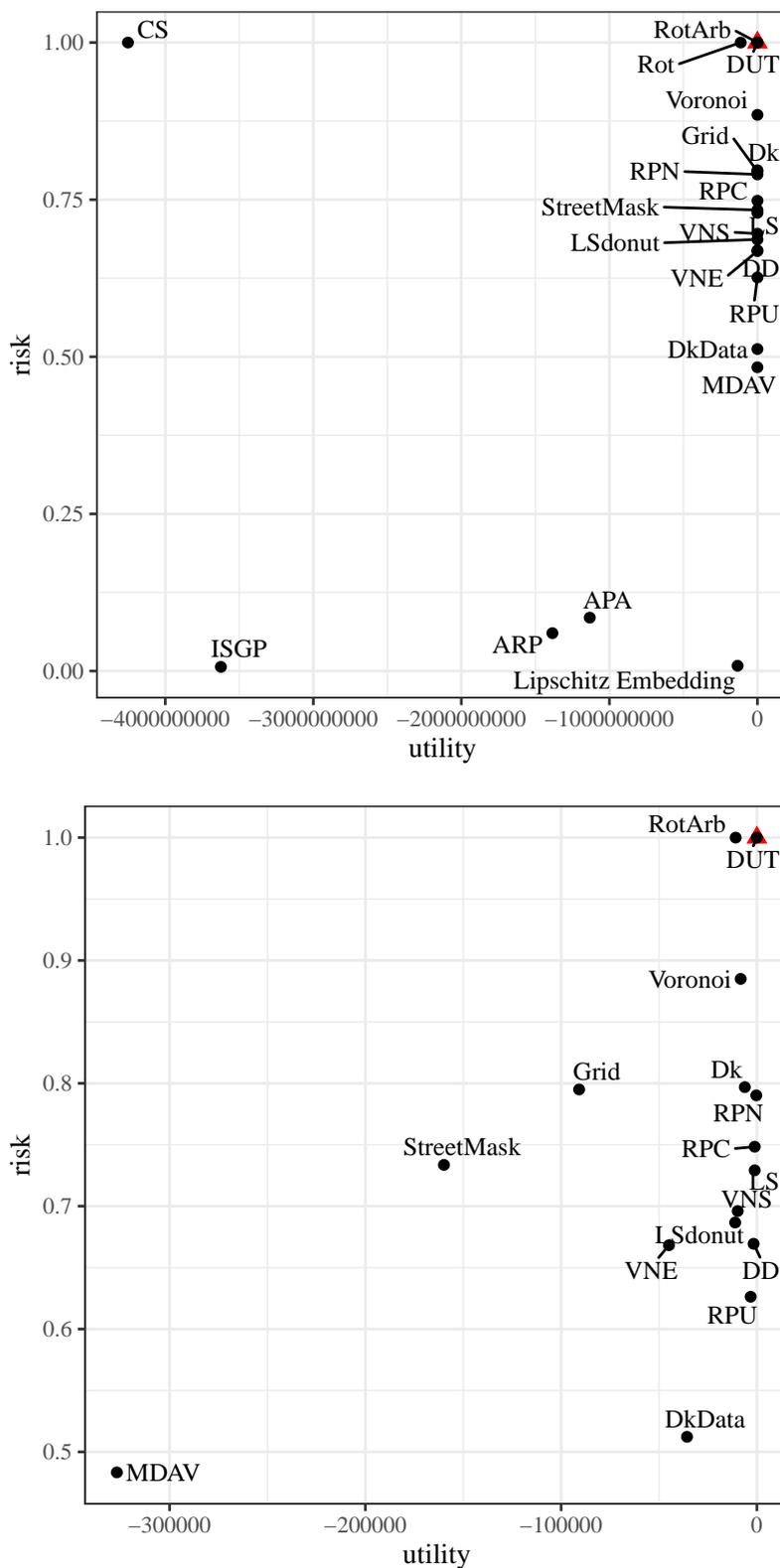


Figure I.3.: Risk-Utility Map of the larger subsample ( $n = 2,000$ ). MSE as utility measure. Red triangle shows original data. Bottom figure shows map without outliers (rotation, change of scale, distance approximation using ISGP, APA, ARP, and anonymization of distance matrices via Lipschitz embedding).

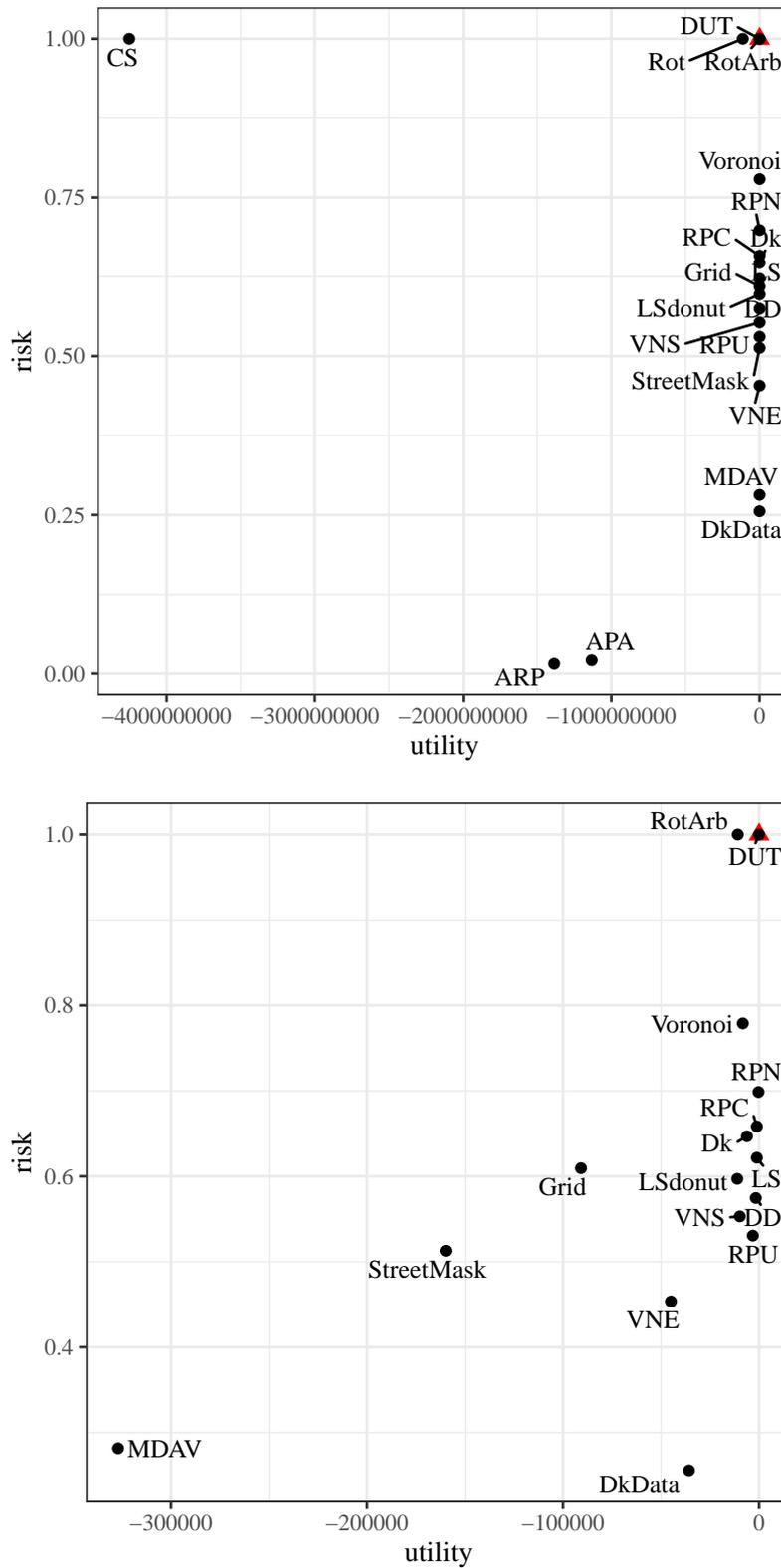


Figure I.4.: Risk-Utility Map of the full sample ( $n = 10,000$ ). MSE as utility measure. Red triangle shows original data. Bottom figure shows map without outliers (rotation, change of scale, APA, and ARP).

# DuEPublico

Duisburg-Essen Publications online

UNIVERSITÄT  
DUISBURG  
ESSEN

*Offen im Denken*

ub | universitäts  
bibliothek

Diese Dissertation wird via DuEPublico, dem Dokumenten- und Publikationsserver der Universität Duisburg-Essen, zur Verfügung gestellt und liegt auch als Print-Version vor.

**DOI:** 10.17185/duepublico/76045

**URN:** urn:nbn:de:hbz:465-20220603-134330-6

Alle Rechte vorbehalten.