Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 112 (2022) 590–595

www.elsevier.com/locate/procedia

14th CIRP Conference on Intelligent Computation in Manufacturing Engineering, Gulf of Naples, Italy

Analysis of feature extraction algorithms for quality prediction using machine learning in injection molding

Dimitri Kvaktun[a]*, Alexander Hoffmann[a], Reinhard Schiffers[a]

[a]*University of Duisburg-Essen, Lotharstraße 1, Duisburg 47057, Germany*

* Corresponding author. Tel.: +49-203-379-1655; fax: +49-203-379-4379. *E-mail address:* dimitri.kvaktun@uni-due.de

Abstract

When using machine learning for quality prediction in injection molding, feature processing is an important step in data preparation to improve the quality of model prediction. The objective of this study was to evaluate the prediction performance of different feature extraction algorithms compared to more common feature selection. Two test specimens, each with two recorded quality features, were produced in six different injection molding process states, resulting in 11.720 injection cycles as the data base. Depending on the process state, R^2 of up to 0.99 could be achieved. Nevertheless, the results show that feature selection is preferable for feature processing.

Keywords: Feature processing; Industry 4.0; Aritifcal intelligence; Smart manufacturing; Quality prediction; Dimension reduction; Data preprocessing

1. Introduction

Thermoplastic injection molding is one of the most popular plastics processing methods, as it allows mass production of complex parts in a single discontinuous production step. The process consists of the following steps: First, the material, which is usually in granular form, is plasticized by the rotation of a screw and additional heat input from heating elements. Then the melt is injected into the cavity of the mold. Due to shrinkage effects caused by the cooling of the part, the screw injects further material to compensate for this effect during the holding pressure phase. Finally, the mold opens and the part is ejected. [1]

The quality of the molded parts depends on various internal and external influences, like machine settings, machine wear, material batch variations, etc. In order to compensate for these disturbing effects, various methods have been developed throughout history. [2]

These control strategies can be divided into three categories depending on the parameters they use for control: Machine parameter control, process parameter control and quality control. Machine parameters are independently controllable variables such as cylinder temperature or holding pressure. Process parameters are dependent variables and result from various machine parameters and other parameters like mold geometry. Examples are cavity pressure or melt temperature. The result of machine, process and external parameters are quality parameters like part weight and dimensions. [3]

The goal of all control strategies is to adjust the machine parameters to obtain the desired quality. Strategies that directly measure quality are often not feasible in an industrial context due to additional equipment costs, measurement time, and other constraints like space in the mold. Therefore, current research focuses on indirect prediction methods. These can be based on physical or statistical modeling, although the former is less pursued since a complete physical description of the injection molding process is not possible. [4]

Currently, machine learning has been proven as a statistical modeling method with good prediction performance. The quality prediction task is categorized as a supervised learning problem because the goal of machine learning is to find the relationship between the input data (also called "features") and the output data. In injection molding, the machine and process

10.1016/j.procir.2022.09.059

Fig. 1. Steps involved in creating a quality prediction using machine learning.

parameters are the input data and the quality variables are the output data. [5]

Before a prediction can be implemented, three steps are necessary (cf. Fig. 1), with the feature preprocessing step (marked in red in Fig. 1) being as important as the selection of a suitable machine learning method and its training. This preprocessing step has three goals: 1) reduce the amount of data to limit memory requirements and decrease computation time; 2) focus on relevant data for next use; and 3) increase the prediction performance of the machine learning algorithm. [6]

Feature processing algorithms can be divided into feature selection (FS), feature extraction (FE), and feature construction. While FE and FS reduce the dimensionality of the data (number of features), feature construction can expand the number by creating new features that attempt to increase the expressiveness of the original features [7]. Since we focus on dimensionality reduction methods (DRM) in this paper, feature construction is not discussed further.

Although feature preprocessing is an important step, it is often neglected when using machine learning in injection molding. Usually, the focus is on training the model and the prediction itself. As a result, few research papers in recent years mention this step, and even fewer address it (cf. Table 1).

Table 1. Overview feature selection and extraction in injection molding.

Algorithm	Approach	Reference
Different embedded methods	Feature selection	[5]
InfoGain	Feature selection	[8]
InfoGain & ReliefF	Feature selection	[9]
Correlation based feature selection & ReliefF	Feature selection	[10]
Recursive feature elimination	Feature selection	[11]
Autoencoder	Feature extraction	[12]
Principal component analysis	Feature extraction	[13]
Recursive feature elimination & Support vector wrapper	Feature selection	[14]

We do not claim completeness in this survey, and some literature not listed in the table uses manual FS, but there is a wide range of possible DRMs. However, the main difficulty is that a particular DRM may be suitable for a particular data type and application and not for others [15].

This study is a continuation of [16], in which we applied various FS algorithms to injection molding data. To our knowledge, there is no systematic analysis of FE methods on injection molding data, although this type of DRM can be as good as FS [17]. Therefore, the following questions are analyzed in this study:

- How good is the prediction of different machine learning algorithms using FE as a feature processing step?
- Which FE algorithm performs best on injection molding data?
- How important is the selection of a specific machine learning algorithm?
- How does the predictive performance of models trained using FE compare to the same models trained using FS?

The rest of this paper is organized as follows. In Chapter 2, the experimental setup for data generation is presented, as well as the FE algorithms used for comparison. Also, the overall prediction process is outlined. Next, Chapter 3 presents the results that answer the above research questions. This is followed by a discussion in Chapter 4. Lastly, the results are summarized and an outlook on future work is given.

2. Methodology

2.1. Experimental setup

All data used for the evaluation of the different FE algorithms were generated with an all-electric *KraussMaffei 120-380 PX* injection molding machine, which is part of a production cell consisting of the injection molding machine, a linear robot, a conveyor belt and 100% inline quality monitoring. Two test specimens (cf. Fig. 2) were produced, a plate specimen made of polypropylene (*Moplen HP501H, LyondellBasell Industries N.V., Rotterdam*) and a cover specimen made of polyamide 6 (*Durethan B30S, Lanxess Deutschland GmbH, Cologne*). The weight and length of the plate specimen and the weight and diameter of the cover specimen were used as quality variables. The weight was measured with a *Sartorius Entris 153I-1S* balance and the length and diameter were photographed automatically with a *Canon Eos 5D Mark III DSLR* camera with *EF 70-200 mm f/4L USM* lens. The dimensions were then extracted by an automated algorithm. Machine and process parameters were taken from the built-in machine sensors. In addition, the cavity pressure for the plate specimen was recorded. Six different experiments were used to generate the data: A start-up process, a stable process, a process with downtime, a process with regrind material, a process with regrind material and *KraussMaffei APC+* (a control strategy based on process parameters), and a Design of Experiment (DOE). These experiments were intended to reflect real situations that may occur in industry sooner or later in the production process. Each experiment had 1000 injection molding cycles, only the DOE had 860, resulting in a total of 11.720 injection molding cycles. Further information on the various experiments can be found in [18].

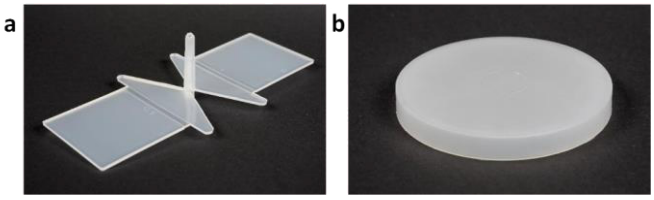

Fig. 2. (a) plate specimen; (b) cover specimen.

2.2. Feature extraction

Feature extraction is a DRM that generates a new set of features computed from the original set. While FS reduces the dimensionality of the set by selecting a list of features from the entire feature set, FE reduces the dimensionality by mapping the feature space to fewer dimensions. This is usually done by a transformation. [7]

Next, we would like to briefly introduce the algorithms we used for the study.

Principal component analysis (PCA) is an orthogonal linear projection method in which the old features are replaced by new, so-called principal components (PCs) by linear combination. Here, the variance in the original features represents their importance. After sorting the PCs by variance, a certain number is selected and the rest is discarded. [15]

Independent component analysis (ICA), like PCA, is a projection method. While PCA tries to find uncorrelated features, ICA looks for independent features. In this work, we use a reconstruction ICA that modifies the standard ICA to improve the performance of the FE method. [19]

Kernel PCA (kPCA) is a nonlinear DRM and an extension of conventional linear PCA using the "kernel trick", which results in lower computational power. Instead of computing the covariance matrix, the eigenvectors of the kernel matrix are computed [20]. The kernel also makes PCA suitable for nonlinear mappings [15].

Multidimensional scaling (MDS) is a manifold approach that generates new features that maintain the distances between old data points for a selected neighborhood [21]. We have used two different MDS methods. First, the classical MDS (CMDS), which uses Euclidean distance. Second, a non-classical MDS (NCMDS), which extends the classical MDS by allowing different criteria to construct the configuration, e.g., non-metric scaling.

Like MDS, *locally linear embedding* (LLE) is a nonlinear DRM that aims to preserve the local properties of the data while reducing the feature space. LLE consists of three steps. First, the k-nearest neighbors are identified. Second, a set of weights is computed for each point using a linear combination of the neighbors. Last, the points are transformed to a lower feature space while maintaining the weight matrix from step 2. [21]

The *Isomap* algorithm is an extension of the classical MDS by using geodesic distances instead of Euclidean distance. This enables the MDS to deal with non-linear data. There are furthermore two criteria for the neighborhood. First, the epsilon neighborhood, which is a threshold for a real value where only points with a Euclidean distance less than epsilon are considered neighbors. Second, the k-neighborhood, which simply considers the nearest k points. [21] In this work we only consider the k-neighborhood.

Autoencoders were not actually developed with dimensionality reduction in mind but have shown that they can be used as such. They are a neural network consisting of at least three layers, where the output layer has the same number of neurons as the input layer, and the hidden layer(s) have fewer than the input and output layers. This structure is also referred to as the "encoder-decoder architecture," where the encoder encodes the high-dimensional data into a lower dimension, while the decoder uses the low-dimensional data and attempts to reconstruct the original high-dimensional data. The goal is to reduce the reconstruction error, using only the encoder for FE. [22]

2.3. Quality prediction system

All steps to build a quality prediction model were performed in MATLAB2020b. First, the data were cleaned of features with no or constant entries. This left a total of 48 features. Then, the data were divided into 80% training data and 20% test data using the holdout method [23]. The FE algorithms were then applied. Table 2 shows the different configurations of the methods with their hyperparameters.

Table 2. Feature extraction configurations

Algorithm	Hyperparameter
PCA, ICA, CMDS, NCMDS	-
LLE	Number of neighbors [5,15,25]
Isomap	Number of neighbors [15,25]
kPCA	Kernel [Linear, Polynomial, Gaussian, Exponential, Laplacian]
Autoencoder	Transfer function [Saturating linear, Log-sigmoid]

A total of 16 FE models were initially considered, with the feature space reduced to five. This number was chosen as the threshold, since previous evaluations have shown that the optimal number of features is between five and ten [24].

In the next step, the training of the model was performed. The following seven machine learning algorithms were used: Artificial neural networks (ANN) [25], support vector machines (SVM) [26], binary decision trees (DT) [27], k-nearest-neighbors (kNN) [28], ensemble methods (EM) (LSBoost [29] & random forest [30]), Gaussian process regression (GP) [31] and multiple linear regression (MLR) [29]. The hyperparameters were optimized using Bayesian optimization with 5-fold cross validation [24]. An overview of the hyperparameters and their values can be found in [18]. The coefficient of determination (R^2) was used as the evaluation criterion for the prediction on the test dataset.

3. Results

3.1. Preselection of feature extraction configurations

In order to reduce the computation time, a pre-selection of the different configurations (cf. Table 2) was made based on the DOE dataset for the cover weight. This dataset has been shown to be suitable for model evaluation in the past [18]. A configuration was selected for further evaluation only if it could achieve an $R^2 > 0.4$ with at least one of the seven different models.

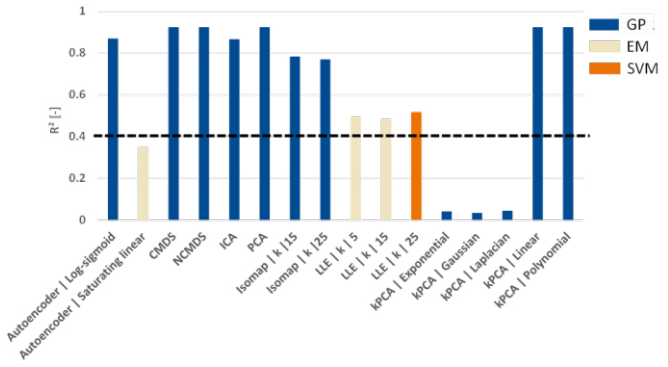

Fig. 3. R² for the best prediction performances achieved from the different FE configurations on the DOE test dataset for the cover weight.

Figure 3 shows the results of the different configurations and the model that achieved the highest performance. Of the seven possible machine learning methods, only three different ones achieved the highest coefficients of determination. Most frequently, the Gaussian process achieved the best R² (12/16), followed by the ensemble method (2/16) and the support vector machine (1/16). The highest predictive performance was obtained by NCMDS with 0.92. The lowest R² was 0.03 by kPCA with the Gaussian kernel.

With this preselection, four configurations are no longer used. In addition, the configuration LLE | k | 15 is omitted because the results are very similar to the other numbers of neighbors.

3.2. Overall prediction performances

The main research question was focused on the predictive performance when using FE. First, figure 4 shows the highest R² value for the different datasets focusing on the specimen and quality criteria. The prediction performance varies depending on the dataset. Only the DOE dataset achieved an R² above 0.7 for all modelled quality features. Overall, of the 24 datasets, ten models had an R² below 0.4, three models had an R² between 0.4 & 0.7, and eleven models had an R² above 0.7. The most common combination of FE and model

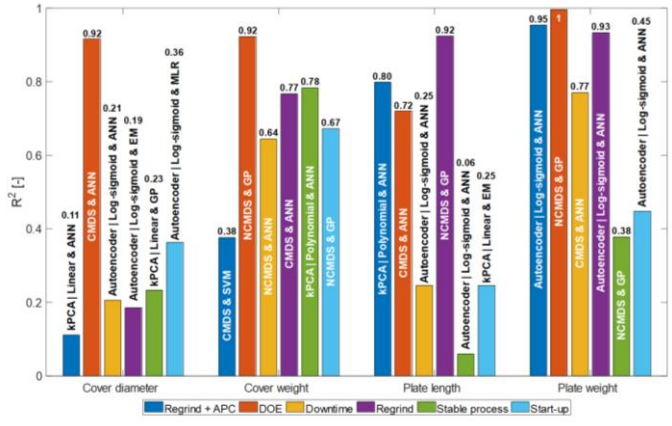

Fig. 4. Best R² for every test dataset and quality criteria.

was Autoencoder | Log-sigmoid & ANN (6), followed by NCMDS & GP (5) and CMDS & ANN (4).

Next, figures 5 & 6 show a comparison between the different FE algorithms for the plate and cover specimen to answer the research question of which algorithm performs best on injection molding data. The highest R² and the mean value from the six experiments are shown. The plate weight could be predicted with an R² above 0.9 with each FE method. This is also true for length, except for the LLE | k | 5 configuration, which only achieved 0.71. The prediction performance for the cover varied more and was worse than for the plate. Here, the highest R² for cover weight was 0.92 (NCMDS) and the lowest 0.49 (LLE | k |5). From the mean values it can be seen that no FE algorithm had performed significantly better than others. Only LLE and Isomap showed inferior values on both cover and plate. The mean value of the Autoencoder | Saturating linear was also lower for the cover dataset than that from the PCA, ICA, kPCA, and MDS variants. Only NCMDS and Autoencoder | Log-Sigmoid achieved an R² above 0.7 with the highest being 0.71, and that only for the plate weight (cf. Fig. 5). To look at the prediction accuracy in more detail, figure 7 shows the R² on the start-up process test dataset for the cover specimen. This dataset was used because the overall prediction performance here was typically lower than for other datasets (cf. Fig. 4). Again, LLE and Isomap showed lower values (R² of ~0.1) than the other algorithms. Considering a threshold of 0.7, none of the models on this dataset were able to exceed it.

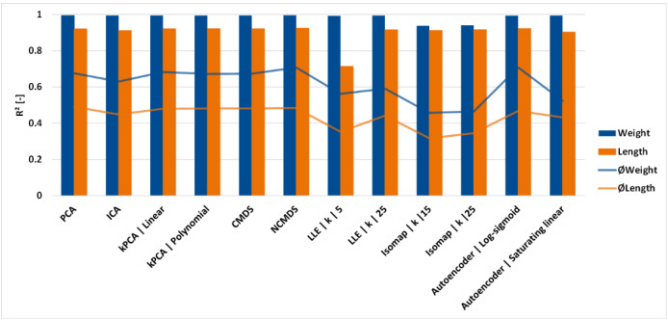

Fig. 5. Highest and mean R² for the FE algorithms on the plate specimen test dataset.

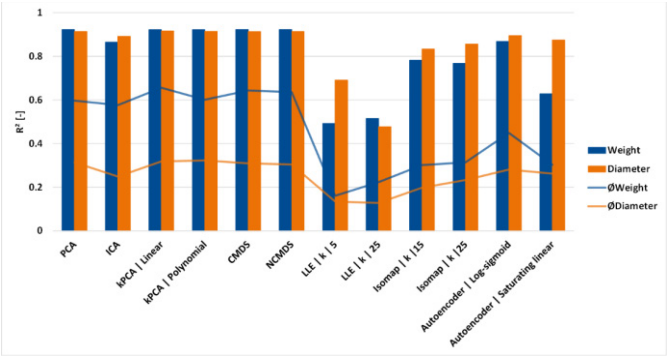

Fig 6. Highest and mean R² for the FE algorithms on the cover specimen test dataset.

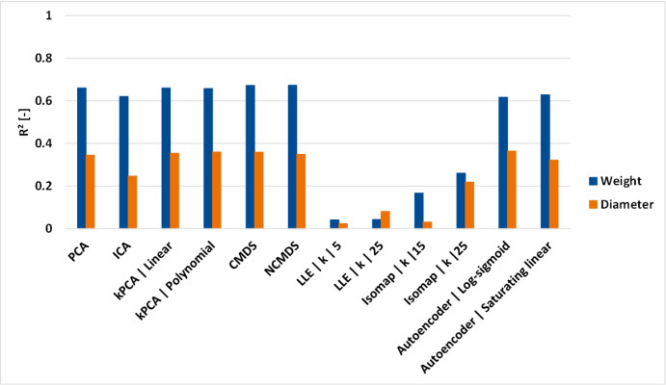

Fig 7. R² for the cover specimen on the start-up process test dataset.

3.3. Comparison of different machine learning algorithms

To answer the question how important the right choice of machine learning algorithm is, figure 8 shows the comparison of the different used algorithms. GP and ANN most frequently achieved the highest R² (cf. Fig. 8a) with a total share of 75%. However, if we look at the relative share of the sum of R² over all datasets (cf. Fig. 8b), we see an even distribution, with only MLR and kNN have a smaller share.

3.4. Comparison with feature selection

The last question was whether feature extraction or feature selection is better as a step of feature processing in injection molding. Here we used a forward search with correlation-based feature selection. This combination was found to provide good results [18]. Figure 9 shows the highest R² achieved by both methods. Of the 24 datasets, FE was superior to FS only once, with a difference of 0.03 for the regrind dataset using the cover weight. The largest overall difference was 0.51 for the regrind & APC+ dataset for the cover weight as well. On average, the FS had an R² 0.12 higher than FE.

4. Discussion

Relating the results to the steps of generating a quality prediction (cf. Fig. 1), it can be seen that the largest influence on the prediction performance came from the data source then

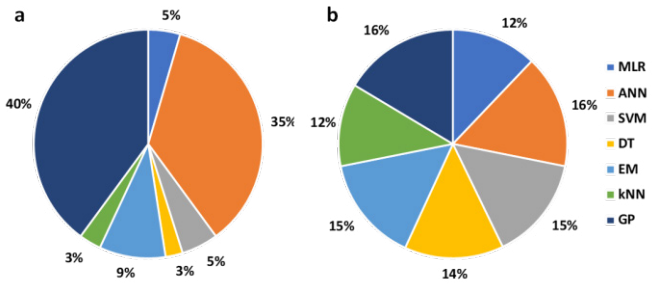

Fig. 8. Comparison of machine learning algorithms according to best R² (a) and summed R² (b).

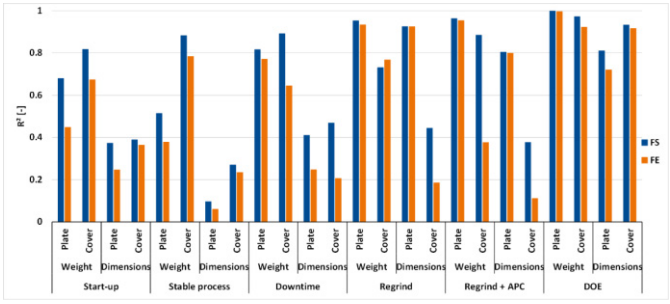

Fig. 9. Comparison feature selection and extraction.

followed by the choice of the machine learning algorithm (cf. Fig. 8a) and lastly the choice of the feature processing algorithm (cf. Figs. 5 & 6). This order is also consistent with the results from [18,24]. Comparing the R² from the weight and the dimensions, it can be seen that the weight was easier to predict than the length or the diameter. One reason for this could be the accuracy of the measurement methodology. While the measurement of weight has few disturbing factors, the measurement of dimensions is affected by various disturbing factors such as ambient light, shape tracking algorithms, position on the conveyor belt, etc. In addition, only the DOE dataset was able to provide high predictive values across all specimens and quality criteria, which may be one reason for its popularity as a data source for predicting injection molding quality. However, for use in industry, the use of a stable process should be aimed for, as performing a DOE takes time and resources.

In principle, a high prediction quality could be achieved with FE, so that this type of DRM can be used for quality prediction. Only LLE and Isomap stood out here with worse R². An explanation for the results of Isomap could be that the geodesic distances represent the present feature space worse. This would also be confirmed by the higher R² of CMDS, which uses Euclidean distance. An explanation for LLE could be the lack of optimization of the number of neighbors.

The comparison of the different machine learning methods shows that the best prediction performance could be achieved with GP or ANN (cf. Fig. 8a). Nevertheless, it should be noted that a simple multiple linear regression did not provide significantly worse results (cf. Fig. 8b). This is also consistent with the findings in [18]. Compared to the other methods (except kNN), MLR has the advantage that no long training phase is required. As a result, the quality prediction application is ready for use much faster.

The comparison between feature extraction and selection has shown that FS is to be preferred as a step of feature processing. It should be noted that FE is particularly useful for high-dimensional data (HDD). HDD is data where the number of features is greater than the number of samples (here cycles) [32]. This was not the case in this study. In addition, the results can be interpreted after feature selection because the features remain unchanged. This is advantageous in terms of quality control because the information can be traced directly to the machine and process variables.

5. Conclusion and outlook

Feature extraction is a common feature processing method in machine learning applications. In this study, we evaluated several algorithms on injection molding data with the goal of using machine learning for quality prediction. After a pre-selection, we compared the 12 most promising configurations on 24 datasets consisting of six different experiments, each with two specimens and two quality criteria. Seven machine learning methods were used for prediction. A total of 2016 R^2 were analyzed. The highest R^2 was 0.995.

Measured by the achievable model quality, the feature extraction algorithms perform almost as well as the feature selection. However, acceptable models could not be generated for all process states using FE. Only ten models achieved an R^2 above 0.7, which we consider a threshold for use in quality prediction. When datasets can be modelled well, the choice of FE algorithm is secondary.

Dimensionality reduction for data preprocessing will have an important role as the digitization of industrial plastics manufacturing progresses with more and more sensor information available, in order to optimally process the available data. However, it is even more important to increase the quality of the available data to improve the prediction performance, especially with stable process data, which is the most common situation in industry. Since no general statement can be made about which combination of methods should be chosen in the respective phases of model building, dynamic model building can provide better results. We have already shown a first approach here [33].

References

[1] Johannaber F. Injection Molding Machines: A User's Guide. 4th ed. München: Carl-Hanser; 2016.

[2] Fernandes C, Pontes AJ, Viana JC, Gaspar-Cunha A. Modeling and Optimization of the Injection-Molding Process: A Review. Adv. Polym. Technol. 2018;37(2):429–449.

[3] Chen Z, Turng L-S. A review of current developments in process and quality control for injection molding. Adv. Polym. Technol. 2005;24(3):165–182.

[4] Ogorodnyk O, Martinsen K. Monitoring and Control for Thermoplastics Injection Molding A Review. Procedia CIRP 2018;67:380–385.

[5] Jung H, Jeon J, Choi D, Park J-Y. Application of Machine Learning Techniques in Injection Molding Quality Prediction: Implications on Sustainable Manufacturing Industry. Sustainability 2021;13(8):4120.

[6] Guyon I, Elisseeff A. An Introduction to Feature Extraction. In: Kacprzyk J, Gunn S, Guyon I, Nikravesh M, Zadeh LA, editors. Feature extraction: Foundations and applications. Berlin, Heidelberg: Springer Berlin Heidelberg; 2006, p. 1–25.

[7] Liu H, Motoda H. Feature Extraction, Construction and Selection: A Data Mining Perspective. Boston, MA: Springer; 1998.

[8] Ogorodnyk O, Lyngstad OV, Larsen M, Wang K, Martinsen K. Application of Machine Learning Methods for Prediction of Parts Quality in Thermoplastics Injection Molding. In: Wang K, Wang Y, Strandhagen JO, Yu T, editors. Advanced Manufacturing and Automation VIII. Singapore: Springer Singapore; Imprint: Springer; 2019, p. 237–244.

[9] Ogorodnyk O, Lyngstad OV, Larsen M, Martinsen K. Application of feature selection methods for defining critical parameters in thermoplastics injection molding. Procedia CIRP 2019;81:110–114.

[10] Ogorodnyk O, Lyngstad OV, Larsen M, Martinsen K. Prediction of Width and Thickness of Injection Molded Parts Using Machine Learning Methods. In: Kishita Y, Matsumoto M, Inoue M, Fukushige S, editors. EcoDesign and sustainability I. Singapore: Springer; 2021, p. 455–469.

[11] Nagorny P, Pillet M, Pairel E, Le Goff R, Loureaux J, Wali M et al. Quality prediction in injection molding. In: CIVEMSA: 2017 IEEE International Conference on Computational Intelligence & Virtual Environments for Measurement Systems and Applications Annecy, France, June 26-28 2017 conference proceedings. Piscataway, NJ: IEEE; 2017, p. 141–146.

[12] Mao T, Zhang Y, Ruan Y, Gao H, Zhou H, Li D. Feature learning and process monitoring of injection molding using convolution-deconvolution auto encoders. Computers & Chemical Engineering 2018;118:77–90.

[13] Zhang S, Dubay R, Charest M. A principal component analysis model-based predictive controller for controlling part warpage in plastic injection molding. Expert Systems with Applications 2015;42(6):2919–2927.

[14] Tripathi S, Mittermayr C, Muhr D, Jodlbauer H. Large scale predictability analysis of process variables from injection molding machines. Procedia Computer Science 2021;180:545–560.

[15] Ayesha S, Hanif MK, Talib R. Overview and comparative study of dimensionality reduction techniques for high dimensional data. Information Fusion 2020;59:44–58.

[16] Schulze Struchtrup A, Kvaktun D, Schiffers R. Comparison of feature selection methods for machine learning based injection molding quality prediction. AIP Conference Proceedings 2020;2289(1):20052.

[17] Khalid S, Khalil T, Nasreen S. A survey of feature selection and feature extraction techniques in machine learning. In: 2014 Science and Information Conference (SAI 2014): London, United Kingdom, 27 - 29 August 2014. Piscataway, NJ: IEEE; 2014, p. 372–378.

[18] Schulze Struchtrup A. Ganzheitliche Formteil-Qualitätsprognose für das Spritzgießen thermoplastischer Kunststoffe auf der Basis maschineller Lernverfahren: University of Duisburg-Essen, Germany; 2021.

[19] V. Le Q, Karpenko A, Ngiam J, Ng A. ICA with Reconstruction Cost for Efficient Overcomplete Feature Learningn. Advances in Neural Information Processing Systems; 2011(24). p. 1017-1025.

[20] Schölkopf B, Smola A, Müller K-R. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. Neural Computation 1998;10(5):1299–1319.

[21] Ma Y, Fu Y. Manifold learning theory and applications. Boca Raton, Fla.: CRC Press; 2012.

[22] Wang Y, Yao H, Zhao S. Auto-encoder based dimensionality reduction. Neurocomputing 2016;184:232–242.

[23] Arlot S, Celisse A. A survey of cross-validation procedures for model selection. Statistics Surveys 2010;4:40–79.

[24] Schulze Struchtrup A, Kvaktun D, Schiffers R. A Holistic Approach to Part Quality Prediction in Injection Molding Based on Machine Learning. In: Hopmann C, Dahlmann R, editors. Advances in Polymer Processing 2020: Proceedings of the International Symposium on Plastics Technology, 1st ed. Berlin, Heidelberg: Springer Berlin Heidelberg; 2020, p. 137–149.

[25] Hagan MT, Demuth HB, Beale, MH. Neural network design. 2nd ed. Boston: PWS Publ. Co.; 2014.

[26] Smola AJ, Schölkopf B. A tutorial on support vector regression. Statistics and Computing 2004;14(3):199–222.

[27] Breiman L. Classification and regression trees. Boca Raton: Chapman & Hall; 1998.

[28] Biau G, Devroye L, Dujmović V, Krzyżak A. An affine invariant k-nearest neighbor regression estimate. Journal of Multivariate Analysis. 2012(112); p.24-34.

[29] Hastie T, Tibshirani R, Friedman JH. The elements of statistical learning: Data mining, inference, and prediction. 12th ed. New York, NY: Springer; 2017.

[30] Breiman L. Random Forests. Machine Learning 2001;45(1):5–32.

[31] Rasmussen CE, Williams CKI. Gaussian processes for machine learning. 3rd ed. Cambridge, Mass.: MIT Press; 2008.

[32] Amaratunga D, Cabrer, J. High-dimensional data. Journal of the National Science Foundation of Sri Lanka. 2016;44(1). p. 3-9.

[33] Schulze Struchtrup A, Kvaktun D, Schiffers R. Adaptive quality prediction in injection molding based on ensemble learning. Procedia CIRP; 2021(99). p. 301-306.

DOI: 10.1016/j.procir.2022.09.059
URN: urn:nbn:de:hbz:465-20221125-150830-3