Analysis of common prediction models for a fuzzy connected source target production based on time dependent significance

Sebastian Soller
Almanara Research GmbH
sebastian.soller@almanara-research.de

Gerold Hölzl, Tobias Greiler, Matthias Kranz
University of Passau
firstname.lastname@uni-passau.de

Abstract

In an industrial setup quality measurements are taken in multiple steps of each production chain. Often a single product is evaluated for several steps in the production, but even more often pooled tests are done, to evaluate the quality of the used material or a charge. Instead of the traditional classification of sensor measurement to quality on a single process step, the question arises, how these steps interact with each other. Is it possible to foresee the faults of later production steps, by analyzing data gathered earlier? Especially in mass production this leads to unclear and fuzzy relationships. The material quality might not be known for every work piece produced but controlled in a fixed time interval. The challenge of these processes is, to correctly connect ground truth to feature vector by their temporal connection. In this work we show multiple steps to reach a better classification and insight into the production process. We gather data from a real-world environment and as a first shot, use common machine learning methods, which are available through public libraries. Therefore, we created a simple connection between the material trace elements and quality inspection. Further data analysis suggested the influence of the exact time of the quality inspection related to the measurement of trace elements. We performed a significance test, to proof the difference of time groups to each other. The identical machine learning methods were applied to these time groups and an improvement of classification accuracy of 2% could be detected. For feature approaches we propose an automatic split system, to find time dependent groups inside the data and split the data accordingly.

Categories and Subject Descriptors

H.1.1 [**Information Systems**]: Models and principles—*General systems theory*

General Terms

Theory, Performance, Reliability

Keywords

Industry 4.0, Maintenance Prediction, Machine Learning

1 Introduction

Machine learning has been further improved to monitor quality control in the recent years [3]. For many industrial processes it is important to detect and recognize anomalies in online setups [16, 15]. For an industrial environment those methods can obtain valuable information for workers on the floor level and management alike. There are many tools, such as scikit-learn [9] library in python, the weka machine learning software [21] and many more, for a fast implementation and testing proposed methods in a production environment. Those methods excel, when a feature vector and ground truth can be provided in a 1 to 1 relationship. These methods are challenged by a fuzzy relationship between target vector and ground truth in a production chain. In a production chain it is an often-proposed problem, that the quality control and the production task are not directly linked through an exact time or identifier relation. In the process of this chain, often unclear, disordered correlations arise, so-called fuzzy correlations [7]. Analyzing these contexts with AI often presents problems. The unclear and often different structure of the data makes it difficult for AI to create feature spaces and labels, which are the basis of prediction models. Human knowledge and analysis of the data can help to recognize structures and to develop models. We propose a novel approach and formulate the following question and propose solutions in this paper:
Can fuzzy connected source target production data be categorized to support a robust prediction model?
In this paper we apply machine learning methods from previous work and open libraries to a complex production task. The performance of these methods will be evaluated. We perform analysis on the time dimension, to verify the significance of time influence of an individual measurement and its problems for the machine learning algorithms. We propose a method, to restructure time dependent data to improve the results of common machine learning approaches.

2 Related Work

Fuzzy logic was first defined by G. Klir in 1995 and describes a theory developed in pattern recognition to "precisely capture the imprecise". It involves assigning numerical values to objects based on the degree to which they belong, using a relationship function. Based on these values, the objects can be classified into fuzzy sets [7]. A similar process can be applied to data in Industry 4.0. Through the process of data mining, which discovers new, meaningful relationships, patterns, and trends by reviewing large amounts of data using pattern recognition technologies as well as statistical and mathematical methods, exciting new data is created [18].

These novel data can form unclear and complicated relationships. Viertl Reinhard [19] gives a definition in his book "Statistical Methods for Fuzzy Data" how to explain this fuzzy data. "All kinds of data which cannot be presented as precise numbers or cannot be precisely classified are called nonprecise or fuzzy. [...] Also, precision measurement results of continuous variables are not precise numbers but always more or less fuzzy" [19]. A distinction is made between one-dimensional fuzzy data, which represent a measurement result of one-dimensional continuous quantities (e.g. volume measurements) and vector-valued fuzzy data, which represent measurement results in real vectors (e.g. positions of objects in space) [19].

The process considered later represents a source target production. At the beginning - at the source - a start process is initialized, which runs through any sub-step to the end point of the production - the target [2]. There is a low amount of literature that describes more complex production chains. Especially not the dependencies between an initial resource, here called source, and its target product, called target. Through the methodology of data mining, production data from the source-target production can be recorded. Data mining describes the collection of production data relevant for objective functions, constraints, and decision variables. This recorded data is divided into dynamic and static classes [13, 11]. In this context, according to Shin et al. [13] dynamic data is used in manufacturing planning and operations, the data may include production/process plan, machine monitoring, inspection, and environmental information. For static data provided by external sources, the focus of this step is not on data logging from equipment or data collection from sensors, but mainly on insight into data characteristics and relationships between different data and identifying data sources [13].

From this arises the question: Why is the process of fault detection, prediction, and prevention relevant in Industry 4.0? This question can be answered with the two factors time and money. In state of the art industrial companies, the aspect of maintenance management has increased a lot in recent years [14]. Manual methods have proven to be not future oriented, because early failure analysis on dynamically changing failure sources and new failure types have proven to be impossible [1]. New industrial processes and their novel technological paradigms provide new opportunities to improve defect detection, prediction, and prevention. The goal of this novel method and the answer to the initial question is to reduce the probability of sudden failures in order to reduce sub optimal use of human resources [14, 1]. The resulting possibility, however, is characterized by practical limitations. The authors Angelopoulos et al. [1] clarify that: "This is a crucial and demanding process due to the autonomous and self-optimized operation of machines and the wealth of data that is collected in real-time." [1, 8] In this process, ML-based approaches must collect large amounts of data and process it in a timely manner to detect abnormal operation. These approaches can be divided into data collection, data processing for feature extraction, and finally fault classification [22]. Here, the last two steps depend heavily on the first step. These must capture theoretically extensive and high quality labeled data sets. In most cases, the collected data, are mixed with noisy data from the environment, which makes it difficult to separate the original data set from the noise. Therefore, it is important to perform data cleaning to improve the quality of the source data. This also has a positive effect on the false detection in the models. Paying attention to the expert knowledge during the initialization of the learning models can increase the quality of the learning strategy. Furthermore, problem arises in data-based modeling and fusion. The reason, according to Angelopoulos et al. [1], is that the data is vulnerable to loss, redundancy, mislabeling, class imbalance, non-stationarity, and heterogeneity of information. In addition, models are not generalizable because location-dependent limitations may occur. For this reason, real-world verification of the accuracy of learning algorithms in terms of fault prediction is essential, especially in scenarios with dynamically changing environments. Finally, the author mentions that often correlation is not carefully investigated, resulting in highly complex models that are trained with insufficient data sets and exhibit overfitting and low interpretability [1]. Time adapting models have been used in previous works [17, 6]. Different models or models of the same type applied to identified sub parts of the data have been used to improve performance. Clustering training data to built multiple prediction models instead of a single model for a use case has been done succesfully in production lines before[23]. With those clustering methods a significant improvement of performance has been reached. We aim to similarly recognize a significant separation of the data along the time axis, to improve common methods.

2.1 Hypotheses

In this work we aim to build a suitable prediction model for a fuzzy connected production process. To approach this problem we perform three steps, to show the advantages of time dependent groups. We intend to proof the necessity of modifications to common machine learning models, by applying the models to fuzzy data. The time dependency groups are analyzed for their significance and each group is used separately in for the machine learning model to improve the results.

Since many past works (Section 2) come to the conclusion that applied machine learning approaches lead to a very inaccurate model, this hypothesis is tested in the first approach. Two ML approaches are trained and tested with the data from an underlying use case, and the following hypotheses were developed to evaluate the an initial approach. The ML mod-

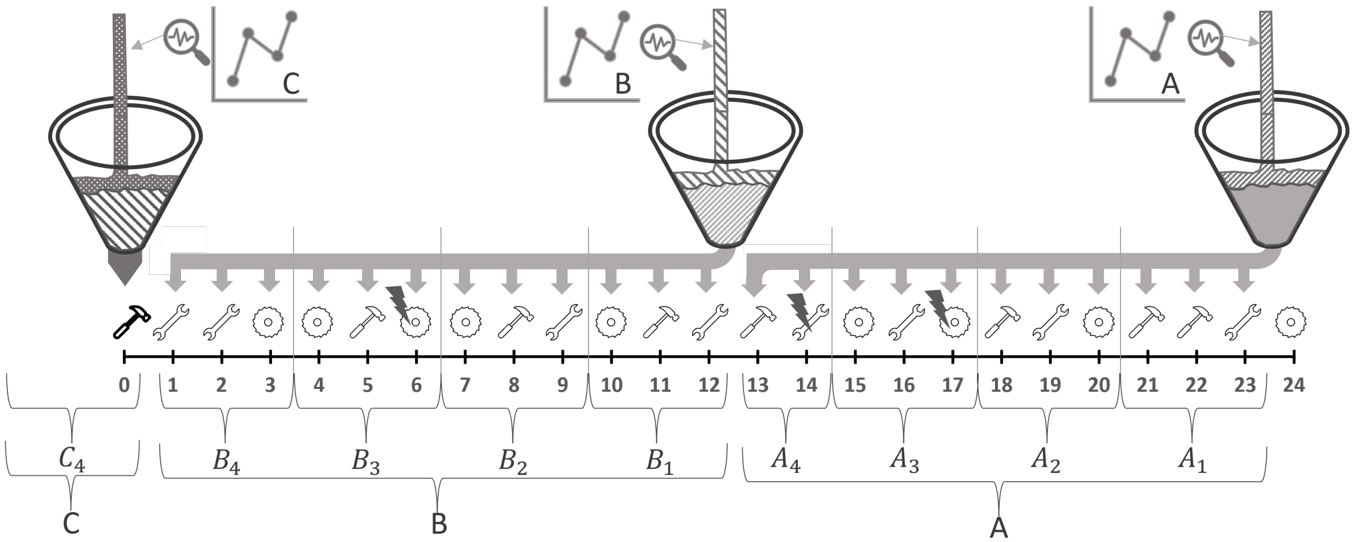

Figure 1. Time dependent process. From each filling a finite number if work pieces can be produced. In example, work pieces 1 to 12 are produced with filling B while 13 to 23 are produced with filling A.

els aim to predict the error rate of each segment, as exemplary shown in figure 1. For each measurement we get a single model and prediction related to a material measurement. These results will solve as a basis compared to the prediction of the common choice.

For the second hypothesis we test the time dependency on production processes. By comparing the different time groups between fuzzy connected measurements:

$H1 =$ The time dependency groups directly influence the results of the production process.

$H0 =$ The time dependency has no influence on the results of the production process.

From the results of our hypothesis, we will determine fitting groups to split the data into, for an improved prediction. If we can determine a significant difference between the different temporal sections, we can split the data to train separate ML models on the data segment. The average results of this method will be compared to the previous results and common choice to improve classification value.

3 Methods

3.1 Common machine learning models

As shown in Figure 1 we cannot classify each individual work piece based on an individual material measurement, as only the measurement for the whole filling is known. A single work piece could not be differentiated from the previous or following work piece of the same material measurement. Therefore we use classification, to classify the whole segments performance, i.e work pieces 1 to 12 for the sensor measurements B and 13 to 23 for the sensor measurements A 1. The classification is performed on the percentage of errors, which are sorted into the classes good (100 percent good), and faulty (less than 98% good). The SVM [10] and Decision Tree [12] approaches are used to classify faulty work pieces based on measurement data. For pre-processing, the data set is modified to get a one-to-one relation between the source and the target. By analyzing transport times and measurement times each fault percentage of the quality inspection segment is linked to the feature vector of the measurement values. To further reduce the features the Variance Threshold method was used [4]. The SVM and Decision Tree, can be executed with a variety of parameters. To determine the best parameters, there is the hyper parameter optimization approach [5]. This approach was also used in this work to generate optimal results. We use a k-Fold Cross-Validation to evaluate the results. This method allows us, to get close to real results, as in the following setting, the trained model would include all machines and workpieces of the past, to classify those that are upcoming.

Table 1. Listing of the average and minimum as well as maximum values of the distance to the last/next measuring point.

	Distance to last measuring point	Distance to next measuring point
Average	164.96	140.89
Standard deviation	113.24	88.54
Min	3.85	0.0
Max	2849.08	3730.96

3.2 Proofing time dependency

We initially define the time dependency. Two similar temporal values can be used in this scenario. One of them can be the elapsed time since the last relevant measurement for the relevant work pieces. The starting point is the time *t3* when the part is produced. From the measuring point *m1* to the finished part, the following time sequences are run through: From time point *m1* to time point *t1* the material is still in the furnace; From time point *t1* to time point *t2* the alloy is transported to the machine; From time point *t2* to time point $t3$ the alloy is in the machine and remains there until the production of the investigated product (see Figure 2). The sum of all listed times results in the distance to the last measuring point. Table 1 shows the average values for this distance. Another value can be the time to the next measuring point. The approach aims at the temporal split of the past time up to the last measuring point. The time up to the last measuring point is determined for all data points (see table 1). This time is divided into 60-minute intervals for the exemplary case shown in this work. This results in the following segments: [0-60];[60-120];[120-180];[180-240];[240-300];[300-open]. We will treat each of these intervals of as a group for the Chi-squared test [20]. The comparison for each group is based on the number of errors and non-errors for each group. A comparison of the observed and expected values will proof or disprove the hypotheses.

3.3 Predicting Split Data

If we can proof a significant difference between the time groups, we can split the data for training and testing into these groups. Separate ML models will be used for the classification of each upcoming work piece. For the example case in Figure 1 this would result in a Model, which is trained with segment A1 and used to classify B1, trained with A2 to classify B2 and so forth. The SVM [10] and Decision Tree [12] approaches will be used with identical preprocessing setups, to achieve comparable results.

4 Study and Data

Melting and taking the measured data

The first step operates at the level of furnaces. In the industrial setup there are 9 furnaces in total, which are identified with unique IDs in the interval [1-9]. Different types of parts with different alloys are cast in the system, so not all furnaces contain the same alloy. During the process, at a certain time t_1, a sample of the alloy m_1 is taken and evaluated. From the sample 25 measuring points are extracted, which represent the individual trace elements of the metals in the alloy. The quantity of trace elements is given as a percentage, which includes only positive values in the range [0-100]. This results in a starting measurement at the source, which is relevant for the further course of the process until a new measurement is carried out. However, there is an exception that in a fraction of the processes, based on the results of the extracted measuring points, a later re-alloying takes place. This process adds the first fuzzy component to the observed system, as it cannot clearly be identified, if the inspection measurements are still identical at a given time between t_1 and the next measurement t_6.

Removal of the alloy and transport to the machine

The next step maps the logistics. For this purpose, at a certain time t_2 a part of the alloy in the furnace is taken out and transported to a suitable machine (depending on the alloy). One or two transports can be realized from such an extraction of the alloy. Due to the local conditions the transport task can take different time. For example, transport $trans_1$ arrives at *Machine 1* at time t_3, while transport $trans_2$ arrives at *Machine 2* at a later time t_4.

Pouring the product

In the next step, different parts are produced on different machines, which are uniquely identified by an ID. The parts can be distinguished by a group of parts. The parts are produced one by one, so the process takes several minutes. After the production of a product, at time t_5, it is checked whether the part is defective, in which case the defect code is stored. Since each group of parts has different complexity in the casting, the processing time may vary. As the machines must guarantee a running production, a new alloy arrives before the old alloy is used up. Thus, a fluent ratio needs to be assumed. This means that some percentage of the alloy stems from the previous transport and thus belongs to the previous measured data and some percentage belongs to the new transport and the new measured data. In an ideal timing this relation should be around 45-55 but can vary due to transport limitations.

Record the measurement data

At time t_6 a new measurement m_2 is performed, in which again the 25 trace elements are measured in percentage. This can be used as a reference measurement to determine changes in the alloy.

4.1 Data

Data collection was done over a two-year interval, from the 14th of august 2019 to the 13th of august 2021. Data was sampled according to the processes and not on an equidistant sampling frequency. For the measurement of product quality, we got 7236905 evaluations between 660 product types which were produced on 49 different machines. The quality assessment resulted in 7074034 and 162871 erroneous products, which were divided into 8 error categories. The material analysis was done with 45083 measurements over the two-year time frame from 7 supply machines. The material was transported from the supply machines to the production machines in 288200 transport processes. This results in a data set of 288200 measurement vectors, which are used to predict the error rate of upcoming work pieces.

5 Result

5.1 Common machine learning approaches

We split the data into a test and training set. As 90 percent was used for training, the rest of the data set was used to test the classification, in the previously described k-Fold Cross-Validation. The test set includes the IO class 44448, the other classes 22782 samples. The selection of the most common class would therefore result in an accuracy of 66.11%. The classification results for the Decision Tree are 70.84% and for the Support Vector Machine 71.10%. Classification results did not improve, when using the machine and the work-piece type to the feature vector.

The data from the common approach was split into the time classes. For each class a classification was performed the classification results for the Decision Tree range from are 72.69% to 73.24 % and for the Support Vector Machine 73.04% to 74.85%.

5.2 Significance of time dependency

The table 2 shows the distribution of number of defects in relation to the total production quantity in a certain time

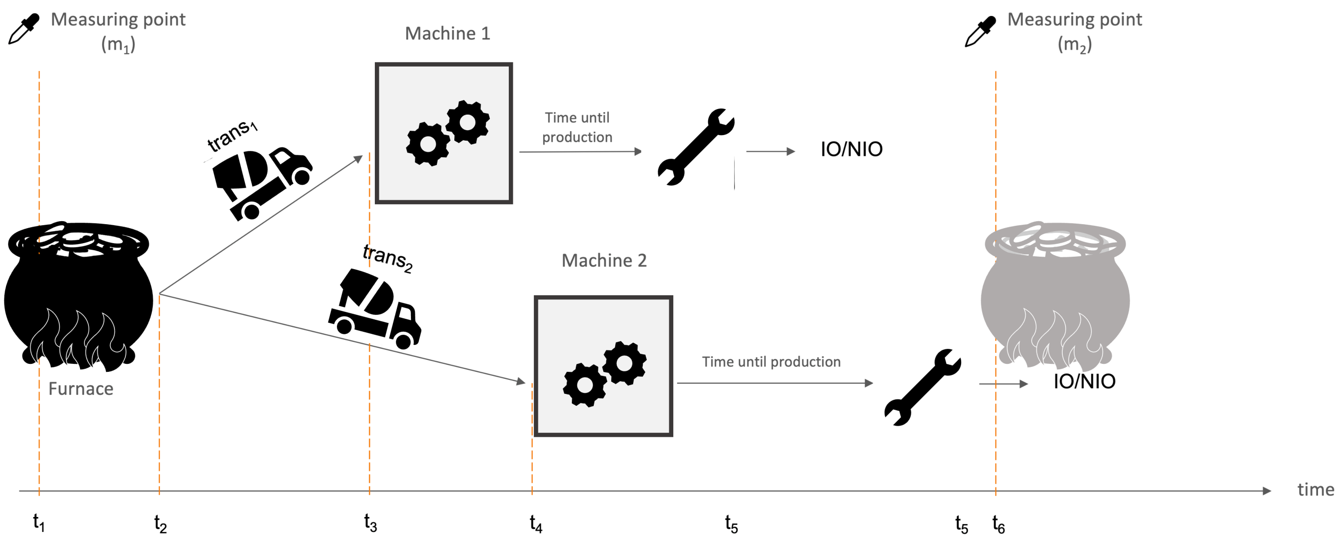

Figure 2. Representation of the considered melt casting process. Each measuring point can be attributed to multiple fillings of machines. The time of transportation and time until production, are also used for analysis.

Table 2. Illustration of the different time intervals from Challenge 2 and the frequency of errors as well as the total number of parts.

Category	Number of errors (% of total errors)	Ratio error / no error (Ratio in %)
[0-60]	11042 (6,9%)	11042:424788 (2,5%:97,5%)
(60-120]	42589 (26,7%)	42589:1774196 (2,3%:97,7%)
(120-180]	50810 (31,9%)	50810:2370524 (2,0%:98,0%)
(180-240]	32042 (20,1%)	32042:1455497 (2,1%:97,9%)
(240-300]	12929 (8,1%)	12929:561827 (2,2%:97,8%)
>300	9550 (6,0%)	9550:357876 (2,5%:97,5%)

interval. The first column of the table lists the time intervals defined in Challenge 2. The second column shows the errors that occurred per interval. The percentage refers to the proportion in relation to the total number of errors that occurred. The next column shows the ratio *faults occurred* to *parts produced correctly*. In the column *Number of errors* one can see an increase of errors with increasing time distance from the measurement. A variance in the individual time classes and the percentages of error can be seen. This significant difference is detected with a Chi-Square test. Here the column *Number of errors* from the table 2 is taken as *observed values* for the test. The *expected values* are obtained from the $(rowsum * columnsum)/samplesize$. The observed and expected values for the chi square method can be found in Table3 and 4. The Chi-Square test results in a p-value of 0.000029. The p-value is less than the significance level of $\alpha = 0.05$, thus the result is statistically significant. This means that the time classes and the error proportions differ significantly.

6 Discussion

The Decision Tree and Support Vector machine outperform the Common Choice for the direct application of the models from the libraries. This shows, those common li-

Table 3. Observed Values of time dependency classes and Error, No Error distribution.

		Error	No Error	
Observed values	0-60	11.042	424.788	**435.830**
	60-120	42.589	1.774.196	**1.816.785**
	120-180	50.810	2.370.524	**2.421.334**
	180-240	32.042	1.455.497	**1.487.539**
	240-300	12.929	561.827	**574.756**
	>300	9.550	357.876	**367.426**
		158.962	**6.944.708**	**7.103.670**

Table 4. Expected Values of time dependency classes and Error, No Error distribution.

		Error	No Error	
Expected values	0-60	9.753	426.077	**435.830**
	60-120	40.655	1.776.130	**1.816.785**
	120-180	54.183	2.367.151	**2.421.334**
	180-240	33.287	1.454.252	**1.487.539**
	240-300	12.862	561.894	**574.756**
	>300	8.222	359.204	**367.426**
		158.962	**6.944.708**	**7.103.670**

braries can often be applied with their proposed pipelines and yield improvement for a supervised system. With low time effort, those libraries provide an initial look, of how good the data is suited from preprocessing to evaluation.
The splitting and testing of time intervals as groups in a Chi-squared test resulted in a significant difference between each class. Therefore, each class was analyzed and classified individually. This shows the improvement between a non-time dependent analysis and the machine learning success after splitting the data. The test for significance can be used for different groupings for the data. We decided to group by hour in this example case for a proof of concept.
The Decision Tree and Support Vector Machine both improved by above by around 2% in average for each segment. The strong significance of differentiation between the segments also showed in the classification results, and the improved ML models perform better. This demonstration of significance provides additional information, about unknown temporal dependencies inside a process, where those dependencies were not known previously.

7 Conclusions and Future Work

Through splitting the data according to its time dependency, the classification results were improved. Overall both approaches, deliver around 30% wrong results and would need to be further improved to contribute valuable information to improve the process. The approach did yield improvement and therefore can further be tested on additional setups. For this we started to further gather data of climate dependent measurement and evaluation systems, to proof the approach in a different field.
To further modify our split of the data, beyond a testing setup, where the data is split into pre given groups, such as one hour in the proposed approach, we plan on implementing an automatic splitting. Therefore, a time stream data can be split into categories iterative, to find the most different groups, by repeatedly testing for their significance until the best split groups are found.

8 References

[1] A. Angelopoulos, E. T. Michailidis, N. Nomikos, P. Trakadas, A. Hatziefremidis, S. Voliotis, and T. Zahariadis. Tackling faults in the industry 4.0 era—a survey of machine-learning solutions and key aspects. *Sensors*, 20(1):109, 2020.

[2] O. Ben-Ammar, B. Bettayeb, and A. Dolgui. Integrated production planning and quality control for linear production systems under uncertainties of cycle time and finished product quality. *International Journal of Production Research*, 58(4):1144–1160, 2020.

[3] C. A. Escobar and R. Morales-Menendez. Machine learning techniques for quality control in high conformance manufacturing environment. *Advances in Mechanical Engineering*, 10(2):1687814018755519, 2018.

[4] F. J. Ferri, P. Pudil, M. Hatef, and J. Kittler. Comparative study of techniques for large-scale feature selection. In *Machine Intelligence and Pattern Recognition*, volume 16, pages 403–413. Elsevier, 1994.

[5] M. Feurer and F. Hutter. Hyperparameter optimization. In *Automated machine learning*, pages 3–33. Springer, Cham, 2019.

[6] G. Hoelzl, S. Soller, and M. Kranz. Detecting seasonal dependencies in production lines for forecast optimization. *Big Data Research*, 2022.

[7] G. Klir and B. Yuan. *Fuzzy sets and fuzzy logic*, volume 4. Prentice hall New Jersey, 1995.

[8] A. Maier, S. Schriegel, and O. Niggemann. Big data and machine learning for the smart factory—solutions for condition monitoring, diagnosis and optimization. In *Industrial Internet of Things*, pages 473–485. Springer, 2017.

[9] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[10] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[11] N. V. Queipo, S. Pintos, N. Rincón, N. Contreras, and J. Colmenares. Surrogate modeling-based optimization for the integration of static and dynamic data into a reservoir description. *Journal of Petroleum Science and Engineering*, 35(3-4):167–181, 2002.

[12] L. Rokach and O. Maimon. Top-down induction of decision trees classifiers-a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35(4):476–487, 2005.

[13] S.-J. Shin, D. B. Kim, G. Shao, A. Brodsky, and D. Lechevalier. Developing a decision support system for improving sustainability performance of manufacturing processes. *Journal of Intelligent Manufacturing*, 28(6):1421–1440, 2017.

[14] H. J. Smadi and A. K. Kamrani. Product quality-based methodology for machine failure analysis and prediction. *International Journal of Industrial Engineering*, 18(11), 2011.

[15] S. Soller, B. Fleischmann, M. Kranz, and G. Hölzl. Evaluation and adaption of maintenance prediction methods in mixed production line setups based on anomaly detection. In *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, pages 520–525. IEEE, 2021.

[16] S. Soller, G. Hölzl, and M. Kranz. Predicting machine errors based on adaptive sensor data drifts in a real world industrial setup. In *2020 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 1–9. IEEE, 2020.

[17] S. Soller, M. Kranz, and G. Hoelzl. Adaptive error prediction for production lines with unknown dependencies. In *Proceedings of the 10th International Conference on Web Intelligence, Mining and Semantics*, pages 227–234, 2020.

[18] T. Velmurugan. Performance based analysis between k-means and fuzzy c-means clustering algorithms for connection oriented telecommunication data. *Applied Soft Computing*, 19:134–146, 2014.

[19] R. Viertl. *Statistical methods for fuzzy data*. John Wiley & Sons, 2011.

[20] K. F. Wallis. Chi-squared tests of interval and density forecasts, and the bank of england's fan charts. *International Journal of Forecasting*, 19(2):165–175, 2003.

[21] I. H. Witten, E. Frank, M. A. Hall, C. J. Pal, and M. DATA. Practical machine learning tools and techniques. In *DATA MINING*, volume 2, page 4, 2005.

[22] K. Worden, W. J. Staszewski, and J. J. Hensman. Natural computing for mechanical systems research: A tutorial overview. *Mechanical Systems and Signal Processing*, 25(1):4–111, 2011.

[23] D. Xu, Z. Zhang, and J. Shi. Training data selection by categorical variables for better rare event prediction in multiple products production line. *Electronics*, 11(7):1056, 2022.