

Designing an Analytical Control Chart System with ML-predicted Quality Characteristics*

Till Carlo Schelhorn^{1,*}, Jonas Gunklach¹ and Alexander Maedche¹

¹Human-centered Systems Lab (h-lab), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

Abstract

Quality management plays a vital role in manufacturing organizations to ensure effective and efficient production processes. To achieve this, organizations implement various data-driven techniques and tools to monitor and manage the quality of their production processes. One essential tool is the control chart, which tracks the performance of a specific quality characteristic over time by taking samples. However, manual sample-taking for a large number of quality characteristics can be time-consuming and costly. To address this challenge, organizations seek to enhance the efficiency of the sample-taking process while accurately detecting production process performance. Recently, machine learning (ML) models have been proposed to predict various quality characteristics, thereby reducing the need for manual measurements. However, existing control chart system designs have been found to be inadequate for integrating ML-predicted quality characteristics. To address this gap, this research aims to design an analytical control chart system with quality characteristics predicted by ML models. Our technical evaluation indicates significant improvements in the efficiency of the quality management process while feedback from a focus group demonstrates the effectiveness of our proposed solution.

Keywords

Quality Management, Statistical Process Control, Control Charts, Machine Learning

1. Introduction

Quality management (QM) in manufacturing is crucial to ensure that a production process consistently functions well [1]. In this line, organizations are implementing Business Intelligence and Analytics (BIA) systems to measure the quality of a production process and detect quality issues. BIA can be defined as "techniques, technologies, systems, practices, methods, and applications that analyze critical business and market and make timely business decisions" [2]. Therefore, organizations leverage statistical process control (SPC) tools to detect and reduce variability in the process [3]. More specifically, control charts aim to track all relevant quality characteristics of a process and indicate if the process is in-control [4]. For that, samples are taken at regular time intervals, and quality characteristics are analyzed for the sample. However, due to the large number of quality characteristics, manual sample-taking is time-consuming and cost-intensive for QM employees [5]. Therefore, organizations aim to improve the efficiency of the sample-taking process while detecting if the process is in-control or not [5].

M. Leyer, Wichmann, J. (Eds.): Proceedings of the LWDA 2023 Workshops: BIA, DB, IR, KDML and WM. Marburg, Germany, 09.-11. October 2023, published at <http://ceur-ws.org>

*Corresponding author.

✉ till.schelhorn@kit.edu (T. C. Schelhorn); jonas.gunklach@kit.edu (J. Gunklach); alexander.maedche@kit.edu (A. Maedche)



© 2023 Copyright by the paper's authors. Copying permitted only for private and academic purposes

CEUR Workshop Proceedings (CEUR-WS.org)

Research in the field of control charts has a long history. For example, Reynolds et al. [6] proposed an adaptive control chart with variable sampling intervals based on the current state of the process. Alwan and Roberts [7] relied on the correlation of features to increase sampling intervals. With the proliferation of machine learning (ML), recent literature proposed leveraging these techniques to extend existing control chart designs. For instance, Zan et al. [8] rely on convolutional neural networks for automated recognition of unnatural patterns, Ferrer [9] uses principal component analysis to reduce the dimensions of data in multivariate production settings, and Tong et al. [10] make use of clustering algorithms for adaptive control charts. However, these approaches do not improve the efficiency of manual sample-taking. Hryniewicz [11] aimed to solve this by replacing manual samples with predicted quality characteristics, but found that the current design of control charts is not suited for this approach. A central challenge relates to the performance of the underlying ML models and incorporating the inherent inaccuracies in these models' predictions into the control chart design. We therefore aim to investigate the development of a control chart system that relies on ML-predicted quality characteristics to reduce the necessity for manual sampling. To solve this, we articulate the following research question:

RQ: *How to design an analytical control chart system based on predicted quality characteristics?*

To answer this question, we conducted expert interviews with employees of a leading German manufacturing company to derive design principles for our system. Based on these principles, we implemented an analytical control chart system prototype. We used the quality data from an exemplary manufacturing process to design the control charts for the respective quality characteristics. Then we identified the characteristics that could be replaced by ML predictions. Finally, we evaluated (1) the ML model by comparing control charts based on ML-predicted quality characteristics with standard control charts and (2) the analytical control chart system using a focus group with manufacturing employees.

2. Foundations

SPC is a collection of methods and techniques based on statistics to establish process control and on this basis continuously improve process performance [3]. A process is out of control if the variability present in a process is produced by an assignable cause and not by natural variation [5]. Identifying the out-of-control process as quickly as possible is one of the major objectives of SPC [4]. Control charts are the main tools for detecting out-of-control processes. Over time samples are taken from the running process and all relevant quality characteristics are measured. These characteristics are all features of the manufactured product relevant to its quality and functioning. To detect if a process is out of control, the location (mean) and variability (range or variance) of the samples are tracked. For each quality characteristic, two control limits are defined: the upper control limit and the lower control limit setting the borders of the in-control process. A value exceeding one of the control limits signals the process being out of control and requires countermeasures to be taken [5]. Important for a well-functioning control chart are meaningful control limits. Therefore a control chart is divided into two phases [4]. Phase I consists of a retrospective analysis of past process data and setting control limits to determine if and when the process was in control. In Phase II these control limits are used to monitor

the process [12]. It normally requires an iterative process of setting control limits, revising and adjusting them until appropriate control limits are found [13]. Once process control is established various process capability indices (Cpk) are used to measure the actual performance of the in-control process and its capability to produce parts inside given specification limits [4]. Figure 1 illustrates the underlying components of Cpk: **process width** and **design width**.

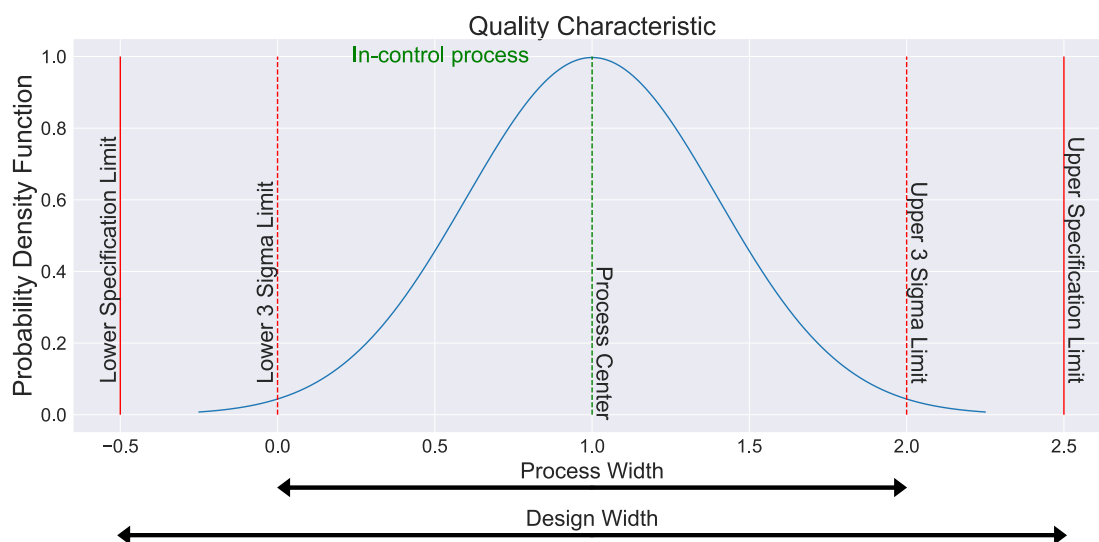


Figure 1: Visualization of process and design width (own representation based on [4])

The **design width** establishes the range within which the specific quality characteristic must fall to meet the criteria for satisfactory quality. It is restricted by the upper and lower specification limits. The **process width** defines the actual performance of the process. With the probability distribution of quality characteristics derived from historical data, the process width is determined by the 3-sigma (3σ) limits of this distribution, representing the range within which 99.7% of produced parts fall (for simplification we chose to display a normal distribution). The Cpk is determined by the ratio between the **process width** and the **design width**, reflecting the process's capacity to meet specified quality criteria. The greater the distance between the 3σ limits and the specification limits, the higher the Cpk value, indicating superior process performance. In practice, a Cpk higher than 1.33, indicating the process produces approximately 96 defective parts per million, is assumed to be sufficient [5].

3. Design Principles

To derive design principles for our system we conducted a series of semi-structured expert interviews with process experts responsible for the production lines and quality management experts [14]. We started with two expert interviews and identified the other participants using snowball sampling [15]. In total, we interviewed 15 people, twelve process experts, and three from the domain of quality management. The interviews took between 25 and 60 minutes. Each

interview was recorded and later transcribed. Based on the transcription we identified several key issues relevant to our design process.

To track the relevant quality characteristics like length, diameter, or the surface finish of the products samples are taken in regular intervals, so-called interval inspections. According to the experts, these are required for assessing the current state of the process. However, they represent a time-consuming and therefore costly activity. More efficient planning with fewer inspections or a smaller sample size was requested by the experts. On the other hand, these interval inspections are necessary to ensure process control by identifying outliers. The experts feared that reducing the number of interval inspections may lead to a loss of control over the process. They conclude that a decent trade-off between an increased efficiency of the interval inspections versus a decrease in the effectiveness of control charts is necessary to guarantee sufficient process control. We formulate our first design principle:

DP1 (Efficiency vs. Effectiveness trade-off): *The system should increase the efficiency of interval inspections, however, needs to ensure sufficient process control*

According to the experts, they often feel uncomfortable adjusting intervals or sample sizes of interval inspections because they lack objective metrics for evaluating possible risks. They stated that only the process capability index is used to determine the current state of the process. The experts conclude that metrics for assessing the potential risks of not detecting an out-of-control process would be useful when adjusting interval inspections. Therefore, we formulate our second design principle:

DP2 (Risk evaluation): *The system should provide metrics for risk assessment*

The experts stated that most decisions regarding the processes are made subjectively based on their know-how and process knowledge. This expert knowledge includes past experiences with the process, quality incidents, corrective actions, and general process performance. According to the experts, this knowledge should be incorporated into our solution and we formulate the following design principle:

DP3 (Expert knowledge): *The system should include the experts domain knowledge*

To reduce the workload of the expert they asked for automation of a possible solution. Analyzing data manually and making changes in the software for inspection planning are tedious tasks. This contradicts the expert's desire to control all decisions made in a possible solution. Also, the incorporation of expert knowledge (DP3) implies some manual tasks. Therefore a trade-off between automation of the solution and the expert controlling the whole process is required. Our last design principle reads:

DP4 (Automation): *The system should be automated as much as possible while still giving the user control over all decisions made*

4. System Design

Based on these design principles we implemented our prototype. The system consisted of two main components: A front-end application served as a tool for designing control charts and supporting the decision of choosing the quality characteristics to replace with ML prediction and a back-end algorithm for computing all possible combinations of characteristics that could be replaced by these predictions.

The front-end application was implemented using the open-source framework Dash [16]. It consists of four different views (see Figures 2-5). The first two views are meant to assist in the design of control charts for a given process. The purpose of views three and four is to support the decision to choose the characteristics to replace by ML predictions.

The design of a control chart is conducted in a retrospective analysis (Phase I) of process data to define the in-control process (see section 2). Based on the historic data of the in-control process the control limits for the control charts are defined. Therefore our application first enables the user to identify the in-control process by setting suitable limits (see Figure 2). The user is presented with historical data of interval inspections to support this process. The drop-down menu at the top left (1) allows the user to select the quality characteristic to inspect. Next to the drop-down menu (2) he can switch between showing the mean, range, or all values of the individual samples of the inspections. Also, the period to consider can be changed (3). With simple input fields (4) the user is able to adjust the limits for the in-control process. All changes are shown in real time in the central graph (5). Specification limits of the quality characteristics are shown in red, and limits of the in-control process are in orange. In-control data points are colored blue, and points indicating an out-of-control process are red.



Figure 2: View 1: Defining the in-control process

When satisfied by the definition of the in-control process the user continues to the detailed view for configuring the control chart for that specific quality characteristic (see Figure 3). The graphs on the left (1) show the distribution of the mean and range values grouped by the in-control (green) and out-of-control (red) classification from the previous view. For each group, a combination of a probability density function (PDF) and a histogram is shown. We used the formula proposed by Doane [17] to determine the number of bins of the histogram and used the open source package SciPy [18] to fit the best matching distribution function to the data. The respective control limits for the control chart are displayed as orange vertical lines in the

graphs. The user can adjust them via the input fields at the top right (2). Below (3), the α - and β -errors of the control charts are shown. These give the user an estimation of the performance of the control chart. The graphs and the error metrics are updated in real time when adjusting one of the control limits. In addition, the Cpk value for different time frames (last month, last 3 months, year to date, and all time) is calculated based on the in-control data and displayed at the bottom right (4). The user can use these values to determine if the in-control process was well defined.

The user can switch back and forth between the two views we just described (Figure 2 and 3). This supports the iterative process of Phase I of designing a control chart where control limits are evaluated and the in-control definition of the process can be revised (see Section 2). The whole process requires extensive knowledge about the process from the user to identify the in-control process correctly and find suitable control limits for the resulting control charts. Using our application the process expert can incorporate his know-how (DP3).



Figure 3: View 2: Configuring the control chart

After finalizing this process the available data and parameters of the newly configured control charts were fed into our back-end algorithm. Its purpose was to identify all possible combinations of quality characteristics that are valid candidates to be replaced by ML predictions. For each characteristic, a new ML model was constructed and trained. We used the Open Source framework LightGBM as a regression model (see [19] for documentation). Gradient Boosting Decision Trees (GBDT) became more and more popular over the last years and proved to be accurate models in many data science competitions [20]. LightGBM provides a GBDT framework with high efficiency and fast training without compromising the prediction

accuracy [21]. For hyper-parameter tuning of the models, we used the Optuna framework, a state-of-the-art optimizer for hyper-parameters also proven to be one of the most efficient available [22]. Our train and validation data set contained historical data for 7 months (Sep 2021 - Mar 2022). A hold-out test set contained almost 2.5 months (Apr 2022 - mid-June 2022), 25% of the total data available. The hold-out set was reserved for testing our results in our technical evaluation. After preprocessing the train and validation data consisted of 359 individual inspections each containing the measurements of the 15 quality characteristics. The inspections were taken in almost consistent intervals of 8 hours with only a few gaps in between (e.g. due to public holidays). Each combination of quality characteristics was evaluated if it was a reasonable candidate to be replaced by the machine learning predictions. The models were trained individually. The target values were all quality characteristics that should be predicted. As features, we used the data of all quality characteristics not included in the prediction. We validated the prediction results with a 10-fold cross-validation approach [23]. Due to the limitation of our hardware, only 30 trials of the Optuna algorithm were performed for parameter optimization of each of the models. Each run was evaluated with the root mean squared error (RMSE) metric with the aim of minimizing it [24].

Based on the predicted values we tested the previously designed control charts and compared the results with the control charts using the real interval inspections. We used classification error metrics for our evaluation. Of interest were the false negative rate (false alarms) and the true negative rate (correctly identified out-of-control instances). Also, the Cpk of the process could be calculated using the data labeled as in-control by our ML model. This was done for each validation set of the cross-validation. The overall mean of the Cpk was used to identify if the ML-based control charts could guarantee sufficient process control. A Cpk lower than 1.33 disqualified a candidate.

All valid combinations of quality characteristics that could be replaced by ML predictions are presented to the user in view three of our front-end application (Figure 4). The error metrics, overall out-of-control instances recognized ($1-\beta$ -error), and the number of false alarms (α -error) are displayed. These give the user an estimation of the overall performance of the prediction models (DP2). For evaluating the efficiency increase the percentage savings of physical inspections when using the ML-predicted quality characteristics is presented. Based on this number the average monthly saved inspections are calculated and displayed.

Clicking on one of the combinations opens a detailed view showing information on the individual quality characteristics (Figure 5). For each characteristic, the average Cpk and the standard deviation from our cross-validation runs are displayed. Also, the percentage of out-of-control instances recognized as well as the false alarms and the saved inspections are shown for each quality characteristic individually. Note that the characteristics discussed here are sensitive proprietary information of our industry partner and therefore concealed. They relate to various features of the manufactured product, such as length, diameter, and surface finish.

With this data at hand, the user can select a combination of quality characteristics to be replaced by ML predictions. He can choose according to his personal risk affinity and weigh the saved inspections and the possible risks of missed out-of-control instances against each other. By only including combinations with a Cpk higher than 1.33 sufficient process control is still guaranteed (DP1). The user can incorporate his knowledge about the process into his decision (DP3). The process of identifying the in-control process, designing the control charts, and finally

#	Virtual sensors	Out-of-control recognized	False alarms	% saved	# saved monthly
6	Characteristic 4, Characteristic 5, Characteristic 7, Characteristic 8, Characteristic 12, Characteristic 15	68.75%	11.45%	31.56%	355
6	Characteristic 2, Characteristic 5, Characteristic 7, Characteristic 9, Characteristic 12, Characteristic 15	78.12%	13.36%	30.88%	347
6	Characteristic 2, Characteristic 4, Characteristic 5, Characteristic 7, Characteristic 12, Characteristic 15	58.33%	8.53%	32.11%	361
6	Characteristic 2, Characteristic 5, Characteristic 7, Characteristic 8, Characteristic 12, Characteristic 15	64.29%	10.90%	32.24%	363
5	Characteristic 4, Characteristic 5, Characteristic 7, Characteristic 12, Characteristic 15	75.86%	9.26%	26.67%	300
5	Characteristic 5, Characteristic 7, Characteristic 8, Characteristic 12, Characteristic 15	71.43%	12.50%	26.67%	300
5	Characteristic 2, Characteristic 5, Characteristic 7, Characteristic 12, Characteristic 15	72.00%	8.64%	27.35%	308
4	Characteristic 5, Characteristic 7, Characteristic 12, Characteristic 15	77.78%	8.43%	22.18%	249
3	Characteristic 2, Characteristic 4, Characteristic 8	57.14%	11.90%	15.10%	170

Figure 4: View 3: Result overview

choosing the characteristics to replace by ML predictions is semi-automated. While all decisions are made by the user the calculations regarding the process data and the displayed metrics are automatically performed in the background (DP4). This being only a prototype there is no integration in any productive systems. The automated replacement of interval inspections by ML-predicted characteristics highly depends on the productive implementation of our solution and is not part of the scope of this work.

5. Evaluation

To evaluate our prototype from a technical performance point of view, we used the hold-out test set containing 2.5 months of historical process data. In addition, we also conducted a confirmatory focus group [25] consisting of different process and quality management experts to evaluate our proposed solution.

5.1. Technical Evaluation

Together with the responsible process expert of our exemplary production line (that also supplied the data for our application) we chose one of the combinations of quality characteristics (see Figure 4) for our technical evaluation. According to the expert, the decision was mainly made by comparing the number of saved inspections and out-of-control instances recognized. The Cpk was higher than 1.33 for each quality characteristic. Therefore, it played only a minor role in the decision process. However, the expert stressed that a Cpk lower than 1.33 would have been an absolute obstacle. Therefore, he commented on the Cpk to be a good fit as a restriction

Detail view (Characteristic 2, Characteristic 5, Characteristic 7, Characteristic 9, Characteristic 12, Characteristic 15)

Characteristic	Cpk	Out-of-control recognized	False alarms	Saved inspections
Characteristic 2	2.05 (0.28)	57.14%	4.76%	87.76%
Characteristic 5	5.74 (1.0)	0.00%	0.00%	100.00%
Characteristic 7	1.37 (0.15)	90.00%	41.03%	48.98%
Characteristic 9	1.35 (0.17)	85.71%	38.10%	55.10%
Characteristic 12	1.77 (0.2)	100.00%	2.33%	85.71%
Characteristic 15	4.08 (0.81)	0.00%	0.00%	100.00%

Schließen

Figure 5: View 4: Detailed view of results

when choosing the combinations. According to the expert, it is a metric he is quite familiar with due to his position and can be used to justify his decision to implement the control chart based on ML predictions.

We then evaluated the underlying machine learning models of the chosen combination of quality characteristics with our hold-out data set. The hold-out data contained all data not used to train and evaluate our models (see section 4) The goal was to test the effectiveness and efficiency of the control charts based on ML predictions. We evaluated the results with similar metrics as displayed in our application (see Figure 4). In addition, we compared the Cpk to the Cpk when using a standard control chart. The results of our evaluation are displayed in table 1.

Table 1

Results of the technical evaluation evaluating control charts based on ML predicted values

Characteristic	Baseline Cpk	ML prediction Cpk	α -error in %	β -error in %	Inspections saved in %
Characteristic 2	2.459	1.356	10.6	50.0	80.8
Characteristic 5	7.263	5.369	0.0	100.0	100.0
Characteristic 7	1.437	1.345	30.7	26.3	62.5
Characteristic 9	1.504	1.341	30.1	37.0	62.5
Characteristic 12	1.936	1.634	5.71	20.0	85.0
Characteristic 15	4.833	4.833	0.00	0.00	100.0

The second and third columns show the Cpk calculated based on the data labeled as in-control by the baseline control charts and the control charts based on ML predictions. The amount

of miss-classified instances is displayed in the fourth and fifth columns of the table as α and β -error. In the last column, we can see the potentially saved inspections when replacing all physical inspections of the given quality characteristics with ML predictions.

When comparing these results to the results of our cross-validation displayed in figure 5 we can see that the overall prediction accuracy was similar. Only characteristic 7 shows a significant difference in its classification error of out-of-control instances. This indicates overfitting of the underlying ML model which could be solved by providing more training data.

Regarding the Cpk of the individual quality characteristic, all values lie above the limit of 1.33. This means sufficient process control is provided by the control charts. It should be highlighted that there were no out-of-control instances present for characteristic 15 and only two for characteristic 5 which were both not recognized. The absence of out-of-control instances makes the evaluation of these characteristics difficult but the high Cpk indicates very stable processes with little risk of ever exceeding specification limits. Therefore, these characteristics are promising candidates to be replaced by ML predictions.

A large decrease of the Cpk for characteristic 2 can be detected. It is significantly worse than the baseline control chart and the results from our validation set. This was mainly due to the high β -error. While inspecting the prediction results we found some extreme outliers that were not recognized and led to a wider spread of the distribution of the characteristics values, hence a lower Cpk. Even though the Cpk dramatically decreased when using ML predictions it still did not fall below the limit of 1.33. We argue that increasing the amount of training data would probably lead to better prediction results by including instances of extreme outliers in the future.

Regarding the efficiency of the control charts based on ML-predicted quality characteristics, we can see that they require between 62.5% and 100% less interval inspections. Even when not all but every second interval inspection is replaced by the predicted quality characteristic this could have an enormous impact. With an inspection conducted on average every 8 hours, this leads to around 90 inspections per month per characteristic. With every inspection taking around 5 minutes, based on our observations, this would result in around 18 hours of work saved each month for one production line.

With our technical evaluation, we validate the results of our implemented ML models. A slight decrease in performance can be seen which indicates some overfitting to the test and validation set. We also showed the potential impact on interval inspections of our solution. To assess the long-term impact of our solution some field testing over a longer period is required.

5.2. Focus Group

To evaluate the effectiveness of our solution we conducted a confirmatory focus group [25]. The focus group consisted of six different process and quality management experts. We presented the experts with our prototype and guided them through the different views.

While adjusting the limits for defining the in-control process the experts highlighted the simplicity of the solution. With only three limits the process instances could be classified as in- or out-of-control. Easily switching between graphs of mean, range, and all values was acknowledged to support defining the limits for the in-control process. In the next view (see Figure 3) the visual representation of the probability density function of in-control and out-of-

control data appeared to be a useful aid for the user when defining the control limits of the control chart. In addition, the experts highlighted that the two error metrics (α and β error) would help to assess the risk potential of the control limits.

We also presented the results of our technical evaluation to the experts. The potential reduction of interval inspections was a great inducement for using our solution. Multiple focus group participants offered their participation in a future field study. The features implemented were confirmed to be useful and an overall high utility of the prototype was attested for. None of the members of the focus group felt overwhelmed by the application but highlighted the simplicity of adjusting the in-control process and the control chart limits.

To better understand the underlying ML models the experts requested some insights into their functioning. They suggested that this could generate profound knowledge about the process which could prove useful. We conclude that the effectiveness of our solution is confirmed by the focus group. Furthermore, the results of our technical evaluation were well received. In the future, some methods for explaining the underlying machine learning and its feature importance could be implemented to further support the process experts.

6. Discussion

The optimization of interval inspections with adaptive control charts has been a well-established topic in the literature [26] [27]. There has also been significant research into economic approaches for designing control charts, considering the costs associated with quality management measurements and quality-related issues [28]. However, the utilization of predicted quality characteristics as a substitute for manual inspections has received limited attention. Existing approaches had limited success in achieving a satisfactory outcome in the design of control charts that rely on predicted quality characteristics [11]. Our research fills this gap by providing valuable insights for researchers and practitioners interested in designing a control chart system that leverages ML-predicted quality characteristics.

We contribute to the literature on the utilization of ML techniques in the field of SPC and control charts by providing four design principles for constructing such a system. These principles can be applied across different problem scenarios, demonstrating the broad applicability of our approach. Our implementation offers practical guidance on how to address these principles in various real-world scenarios. While ML-predictions decrease the necessity for manual interval inspections we integrated the Cpk metric to ensure sufficient process control (DP1). Given the widespread utilization of Cpk as a performance indicator in SPC, this design aspect holds relevance across diverse manufacturing settings. To assess the potential risks associated with ML-predicted quality characteristics, we presented classification error metrics (α - and β -errors) of the underlying control charts (DP2). These were validated using our dedicated test and validation dataset. Through the manual identification of the out-of-control process, configuration of control charts, and selection of quality characteristics to replace by predictions, the process expert could leverage their process-specific knowledge in guiding their actions and decisions (DP3). It is important to note that full automation of the process is not feasible due to certain manual steps involved (DP4). While automated calculations were executed in the background, the seamless integration of ML-predicted quality attributes into existing SPC solutions, which

is beyond the scope of this work, remains an unresolved aspect.

7. Conclusion

The goal of this work was to design an analytical control chart system for increasing the efficiency of interval inspections. We therefore dealt with the question of how to design a system that enables the use of predicted quality characteristics in control charts. To solve this, we conducted a series of semi-structured interviews to derive design principles for such a solution. Based on these principles we implemented a prototype to support the design of control charts based on quality characteristics predicted by machine learning algorithms. We then technically evaluated the system to measure the possible impact on the efficiency of interval inspections and conducted a confirmatory focus group to evaluate the usefulness of our solution. Our results show that substantial time savings could be achieved by replacing interval inspections with ML-predicted quality characteristics. Furthermore, the focus group confirmed the effectiveness of our solution. In the future, we aim to evaluate the system over a longer period in a field experiment. Additionally, we plan to incorporate features for explaining the underlying machine learning models. This could further support the user in his decision and generate useful process insights.

Acknowledgments

We thank Max Schemmer and Ulrich Gnewuch for their useful feedback, and Stefan Heitz, for supporting our work at Robert Bosch GmbH.

References

- [1] M. Rungtusanatham, The quality and motivational effects of statistical process control, *Journal of Quality Management* 4 (1999) 243–264. URL: <https://www.sciencedirect.com/science/article/pii/S1084856899000152>. doi:10.1016/S1084-8568(99)00015-2.
- [2] H. Chen, R. H. L. Chiang, V. C. Storey, Business intelligence and analytics: From big data to big impact, *MIS Quarterly* 36 (2012) 1165–1188. URL: <https://www.jstor.org/stable/41703503>. doi:10.2307/41703503, publisher: Management Information Systems Research Center, University of Minnesota.
- [3] M. Owen, *SPC and Continuous Improvement*, Springer Berlin / Heidelberg, Berlin, Heidelberg, 1989. URL: <https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=6557055>.
- [4] D. C. Montgomery, *Introduction to statistical quality control*, 5. ed. ed., Wiley, Hoboken, NJ, 2005. URL: <http://www.loc.gov/catdir/enhancements/fy0621/2004556782-b.html>.
- [5] J. S. Oakland, *Statistical process control*, 6. ed. ed., Routledge, London and New York, 2011. URL: <http://www.sciencedirect.com/science/book/9780750669627>.
- [6] M. R. Reynolds, R. W. Amin, J. C. Arnold, J. A. Nachlas, \bar{X} charts with variable sampling intervals, *Technometrics* 30 (1988) 181. doi:10.2307/1270164.
- [7] L. C. Alwan, H. V. Roberts, Time-series modeling for statistical process control, *Journal of Business & Economic Statistics* 6 (1988) 87. doi:10.2307/1391421.

- [8] T. Zan, Z. Liu, H. Wang, M. Wang, X. Gao, Control chart pattern recognition using the convolutional neural network, *Journal of Intelligent Manufacturing* 31 (2020) 703–716. doi:10.1007/s10845-019-01473-0.
- [9] A. Ferrer, Multivariate statistical process control based on principal component analysis (MSPC-PCA): Some reflections and a case study in an autobody assembly process, *Quality Engineering* 19 (2007) 311–325. URL: <https://doi.org/10.1080/08982110701621304>. doi:10.1080/08982110701621304, publisher: Taylor & Francis eprint: <https://doi.org/10.1080/08982110701621304>.
- [10] L. Tong, H. Lee, C. Huang, C. Lin, C. Yang, Constructing control process for wafer defects using data mining technique, *ICEB 2004 Proceedings (Beijing, China) (2004)*. URL: <https://aisel.aisnet.org/iceb2004/208>.
- [11] O. Hryniewicz, Spc of processes with predicted data: Application of the data mining methodology, in: *Frontiers in Statistical Quality Control 11*, Springer, Cham, 2015, pp. 219–235. URL: https://link.springer.com/chapter/10.1007/978-3-319-12355-4_14. doi:10.1007/978-3-319-12355-4_14.
- [12] S. Chakraborti, S. W. Human, M. A. Graham, Phase i statistical process control charts: An overview and some results, *Quality Engineering* 21 (2008) 52–62. doi:10.1080/08982110802445561.
- [13] W. H. Woodall, Controversies and contradictions in statistical process control, *Journal of Quality Technology* 32 (2000) 341–350. doi:10.1080/00224065.2000.11980013.
- [14] L. S. Whiting, Semi-structured interviews: guidance for novice researchers, *Nursing standard (Royal College of Nursing (Great Britain) : 1987)* 22 (2008) 35–40. URL: <https://pubmed.ncbi.nlm.nih.gov/18323051/>. doi:10.7748/ns2008.02.22.23.35.c6420.
- [15] L. A. Palinkas, S. M. Horwitz, C. A. Green, J. P. Wisdom, N. Duan, K. Hoagwood, Purposeful sampling for qualitative data collection and analysis in mixed method implementation research, *Administration and policy in mental health* 42 (2015) 533–544. doi:10.1007/s10488-013-0528-y.
- [16] Plotly, Dash, 2022. URL: <https://dash.plotly.com/>.
- [17] D. P. Doane, Aesthetic frequency classifications, *The American Statistician* 30 (1976) 181. doi:10.2307/2683757.
- [18] SciPy, Scipy, 2022. URL: <https://docs.scipy.org/doc/>.
- [19] Microsoft Corporation, Lightgbm 3.3.2.99 documentation, 08.07.2022. URL: <https://lightgbm.readthedocs.io/en/latest/>.
- [20] A. Natekin, A. Knoll, Gradient boosting machines, a tutorial, *Frontiers in Neurorobotics* 7 (2013) 21. URL: <https://www.frontiersin.org/articles/10.3389/fnbot.2013.00021/full>. doi:10.3389/fnbot.2013.00021.
- [21] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, Lightgbm: A highly efficient gradient boosting decision tree, in: I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, volume 30, Curran Associates, Inc, 2017. URL: <https://proceedings.neurips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf>.
- [22] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, Optuna: A next-generation hyperparameter optimization framework, in: A. Teredesai (Ed.), *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ACM Digital Library,

- Association for Computing Machinery, New York, NY, United States, 2019, pp. 2623–2631. doi:10.1145/3292500.3330701.
- [23] P. Refaeilzadeh, L. Tang, H. Liu, Cross-validation, in: *Encyclopedia of Database Systems*, Springer US, Boston, MA, 2009, pp. 532–538. doi:10.1007/978-0-387-39940-9_565.
- [24] T. Chai, R. R. Draxler, Root mean square error (rmse) or mean absolute error (mae)? – arguments against avoiding rmse in the literature, *Geoscientific Model Development* 7 (2014) 1247–1250. URL: <https://gmd.copernicus.org/articles/7/1247/2014/>. doi:10.5194/gmd-7-1247-2014.
- [25] M. C. Tremblay, A. R. Hevner, D. J. Berndt, The use of focus groups in design science research, in: A. R. Hevner, S. Chatterjee, P. Gray, C. Y. Baldwin (Eds.), *Design research in information systems*, volume 22 of *Integrated Series in Information Systems*, Springer, New York, NY and Dordrecht and Heidelberg and London, 2010, pp. 121–143. doi:10.1007/978-1-4419-5653-8_10.
- [26] T. Perdakis, S. Psarakis, A survey on multivariate adaptive control charts: Recent developments and extensions, *Quality and Reliability Engineering International* 35 (2019) 1342–1362. doi:10.1002/qre.2521.
- [27] S. Psarakis, Adaptive control charts: Recent developments and extensions, *Quality and Reliability Engineering International* 31 (2015) 1265–1280. doi:10.1002/qre.1850.
- [28] G. Celano, On the constrained economic design of control charts: a literature review, *Production* 21 (2011) 223–234. doi:10.1590/s0103-65132011005000014.