Spring 6-10-2017

APPLYING DATA SCIENCE FOR SHOP-FLOOR PERFORMANCE PREDICTION

Nikolai Stein
Julius-Maximilians-University, Würzburg, Germany, nikolai.stein@uni-wuerzburg.de

Christoph Flath
Julius-Maximilians-University, Würzburg, Germany, christoph.flath@uni-wuerzburg.de

APPLYING DATA SCIENCE FOR SHOP-FLOOR PERFORMANCE PREDICTION

Research paper

Stein, Nikolai, Julius-Maximilians-University, Würzburg, Germany, nikolai.stein@uni-wuerzburg.de

Flath, Christoph M., Julius-Maximilians-University, Würzburg, Germany, christoph.flath@uni-wuerzburg.de

Abstract

Against the backdrop of ubiquitous computing, companies from various industries are building up ever-increasing amounts of business process data. Seeking to salvage these hidden "data treasures," the need for analytical information systems is ever-growing to guide corporate decision-making. However, information systems research is still very much focused on static, explanatory modeling provided by business intelligence suites instead of embracing the opportunities offered by predictive analytics. Describing insights from a real-world manufacturing scenario, we seek to enhance the understanding of predictive modeling. In particular, we highlight that simply dumping data into "smart" algorithms is not a silver bullet. Rather, successful analytics projects require constant refinement and consolidation. To this end, we provide guidelines and best practices for modeling, feature engineering and interpretation leveraging tools from business information systems as well as machine learning.

Keywords: Predictive Analytics, Manufacturing, Process Mining

1 Introduction

Despite the increasing importance of the service sector, manufacturing still plays a key role for many leading economies around the globe. In the last decade, this sector has seen a tremendous digital transformation. Cost decreases for sensors and data storage have paved the way towards ubiquitous IT on the shop-floor instantiated by self-monitoring production equipment and networked production systems (Reddy, 2016). Consequently, manufacturing companies now possess a considerable pool of process data. Manyika et al. (2011) estimate that the manufacturing sector generated more than two exabytes of data in 2010. This data ranges from production status and utilization data to continuous tool and machinery condition monitoring.

Creating ever growing data dumps will not contribute to business value generation and therefore companies are hard-pressed to identify opportunities to benefit from their "data treasures." Leveraging this data by means of new analytics tools offers opportunities to foster data-driven decision-making to increase both efficiency and effectiveness of existing business processes. Such approaches have been discussed in both academic and practitioner literature (H. Chen, Chiang, and Storey, 2012; Sharma, Mithas, and Kankanhalli, 2014). Analytics systems are a new, evolutionary step of business intelligence applications that are expected to transform the corporate landscape. These systems seek to look beyond isolated processes and towards guiding operational or strategic decision-making (Davenport and Harris, 2007). Developing such systems will require a skilled combination of technological assets and business understanding. Hence, it should be considered a chief interest of information systems research (Agarwal and Dhar, 2014). Similarly, Gordon,

Blake, and Shankaranarayanan (2013, p.57) put forward a gap in practice-oriented research in the area of analytics and decision support.

Using a large data-set from a major manufacturing company we illustrate how a predictive analytics solution can be set up, refined and evaluated. In this process, we have to overcome two major obstacles. On the one hand, the manufacturer under consideration features a high process quality and hence very low failure rates. Such class imbalances render prediction tasks particularly challenging (Chawla, Japkowicz, and Kotcz, 2004). On the other hand, the data-set contains a vast number of variables with low information density and high redundancy. To cope with these challenges we set up a data science study in particular addressing two guiding research questions:

RQ1 What is an appropriate machine learning setup (performance metric and model) for the manufacturing failure prediction task?

RQ2 What are effective processes and methods to create meta-features from monitoring data with very low information content?

To tackle these challenges, we combine methods from machine learning and business information systems to guide the development of our predictive analytics solution. Prediction tasks in other manufacturing settings will face very similar challenges. Therefore, we are confident that these research questions and our results can be generalized and applied beyond the specific case at hand. In particular, we want to highlight that very seldom there is one silver bullet to conquer the problem and that rather any predictive analytics project will rely on incremental refinement and improvement of features and algorithms.

2 Related Work and Preliminaries

The idea of "business analytics" generating business value from data has emerged in the last decade. Following H. Chen, Chiang, and Storey (2012), the term is used to describe data science in a business context. Depending on the specific focus and impact of the application, one can distinguish descriptive, predictive and prescriptive analytics (Lustig et al., 2010). According to this classification, *descriptive analytics* encompasses business intelligence tools and processes that use data to understand and analyze past business performance. It focuses on reporting and visualization of historical data and is thus decoupled from future decisions. In contrast to this backward-oriented application, *predictive analytics* aims at uncovering explanatory structures in the data to draw inferences about future instances. The most encompassing application is *prescriptive analytics* which integrates insights from descriptive and predictive analytics to determine appropriate actions or decisions by means of optimization.

Shmueli and Koppius (2011) carve out the difference between explanatory statistical modeling and predictive modeling. They emphasize that explanatory power derived from traditional models does not imply predictive power. Consequently, predictive analytics is needed not only to create models for practical applications but also for theory building and theory testing. Manufacturing companies need to embrace business analytics in order to remain competitive in the global marketplace (Lee et al., 2013). Historically, manufacturing firms have relied on observable process outcomes through shop-floor initiatives like standardized work or continuous improvement. By incorporating advanced analytics they can also address unobservable problems like machine degradation or hidden defects. Vukšić, Bach, and Popovič (2013) report similar findings from the service industry: Here, companies widely use business process management with a focus on performance (e.g., costs) from an internal point of view while business intelligence is used as a managerial tool. The authors call for a better integration of both to improve process performance management.

Recent research regarding machine learning applications for manufacturing follows two distinct directions. On the one hand, technical solutions are used to identify relevant information from large data-sets with many variables. To predict the level of machine degradation, Mosallam, Medjaher, and Zerhouni (2016) apply unsupervised learning to select meaningful variables from a set of monitoring data. Subsequently, features are derived by fitting linear curves to the selected variables. The authors report good results in

a turbofan engine as well as a battery health setting. Sipos et al. (2014) design an information system to predict failures of medical equipment based on log data. To select the relevant variables, multiple linear classifiers are trained using sample data. Subsequently, the highest ranking variables are selected as features for the classification model. On the other hand, a part of the existing literature focuses on possibilities to aggregate the inherent process knowledge in the data. Schwegmann, Matzner, and Janiesch (2013) design a predictive analytics tool combining business intelligence and real-time process monitoring for a maintenance application scenario. By following an event-driven approach, this tool is able to reduce the lag between event observation and the decision-maker's response. Breuker et al. (2016) integrate process-mining and predictive modeling techniques to streamline operational business processes. Process-mining reveals business process models from historical transaction data. Subsequently, predictive analytics approaches facilitate the prediction of the future behavior of currently running process instances. They illustrate how this approach can be used to monitor the likelihood of negative events or detect fraudulent behaviour in real time.

3 Research Approach and Case Study Overview

Modern manufacturing results in a host of data output including status information and error codes from machine tools as well as time stamps tracing individual parts on their way across the shop-floor. Bosch, one of the world's leading manufacturing companies, hosted a data science competition on Kaggle, the leading crowd-sourcing platforms for predictive modeling (Kaggle.com, 2016). This competition features a very large data set with anonymized measurements of production jobs moving through different manufacturing lines and stations. In addition to the measurements, the result of an ex-post quality control process are provided. To generate business value from the available data, participants were challenged to predict the defectiveness of individual production jobs. The competition has spurred ample contributions by competition participants as well as academic publications (Mangal and Kumar, 2016; Maurya, 2016; Pavlyshenko, 2016). We want to address this prediction task as a data science study following the guidelines for applying big data analytics (Müller et al., 2016). Correspondingly, we structure our anlysis along the proposed three phases:

Data collection The data-set was collected in a manufacturing environment. It comprises a total of roughly 2.4 million manufacturing jobs. Each job has a unique id and 4,264 anonymized features. These features can be split into 968 numeric, 2,140 categorical and 1,156 time variables that are measured along 52 stations on 4 different manufacturing lines. Due to data anonymization, no information on the meaning of the numeric and categorical variables is available and only the manufacturing line and the station of the feature recording can be retrieved from the variable name. The time variables indicate when each measurement was taken. To ensure the generalizability of the predictive algorithms, the data-set is split equally into a training and a test data-set. Jobs in the training set are labeled with $Response = 1$ for products failing quality control and 0 otherwise. In the validation set no information on product quality is provided as the response variable is to be predicted. Process quality is very high: Failures only occur in 0.58% of the cases while 99.42% of the observed jobs pass quality control.

Data analysis We develop a predictive model for failure detection in a stepwise fashion. Starting with a naïve approach training a gradient boosting machine on the raw data, we increase the performance by identifying and removing features with low predictive power. Subsequently, we apply process mining to retrieve the underlying process structure from the anonymized data-set. We use this structure to identify valuable features based on failure-rates and manufacturing batches. Finally, anomaly detection is used to further increase the predictive power of the model.

Result interpretation The system is evaluated in Section 6. On the one hand, the detection of defective parts can be significantly improved. On the other hand, weak spots in the process can be identified by exploring the final model parameters. The approach can readily be transferred and implemented for other use cases as the underlying data was anonymized and additional information was unavailable.

4 Model Setup

Prior to any modeling activities, a suitable evaluation metric has to be chosen. This metric has to account for the specific properties of the given scenario, e.g., skewed classes and misclassification cost distributions (Flach, 2003). Therefore, understanding the importance of the evaluation metric is fundamental for the success—or failure—of every data science project (Davis et al., 2007). We then perform an exploratory data analysis to identify simple linear relationships between the input variables and the outcome. This is used to choose an initial model which is then improved by means of feature reduction.

4.1 Evaluation metric

Prior to training a predictive model, a suited evaluation criterion has to be selected. Standard evaluation metrics for classification problems like accuracy fail in settings with high class imbalances (i.e., a simple model predicting "non-defective" for all parts achieves an accuracy of 99.42% in the problem at hand). Following Powers (2011), the Matthews correlation coefficient (MCC) is considered to be robust against class imbalances. It measures the correlation coefficient between the observed and predicted binary classification and returns values between -1 and +1. A value of +1 indicates a perfect prediction while a value of 0 is a random prediction and a value of -1 indicates a total disagreement between prediction and observation. Thereby, the MCC takes into account the true and false positives and negatives:

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{1}$$

Taking a closer look at this metric, we see that MCC optimization is about maximizing the numerator which internalizes the trade-off between maximizing true and minimizing false prediction. For a given model with fixed predictive power the MCC can be optimized by minimizing the product of false positives and false negatives. Assuming that a given model yields n wrong classifications ($FP + FN = n$), the term will be smaller if either the number of false positives or the number of false negatives is very high compared to a case with a balanced error distribution. This means that the determination of the optimal threshold obtains from the maximization of quadratic function as illustrated in Figure 1.
An MCC-optimized model will either have a low false positive and high false negative rate or vice versa. The optimal trade-off is determined by the respective cost for the errors. In settings with high costs for non-detected defects (e.g., product recalls) the number of false negatives will be minimized, in settings with high costs for wrong alerts (e.g., complex quality control) the number of false positives will be minimized. The ability to adopt to the cost structure of the underlying problem is an additional benefit of the MCC metric in the manufacturing context.

4.2 Modeling approach

Having established a suitable evaluation metric, an appropriate prediction model has to be selected. In general, it is recommended to start a prediction project with white-box models such as logistic regression or decision trees as these models offer greater transparency with respect to the underlying model and rules which are used to generate predictions (Kuhn and Johnson, 2013). Yet, in the problem at hand these approaches fail to achieve meaningful results. This is because simple predictive models are typically unable to pick up non-linearities or complex higher-level interactions between variables. There are hardly any linear correlations between the response variable and the numerical variables present in the data-set.

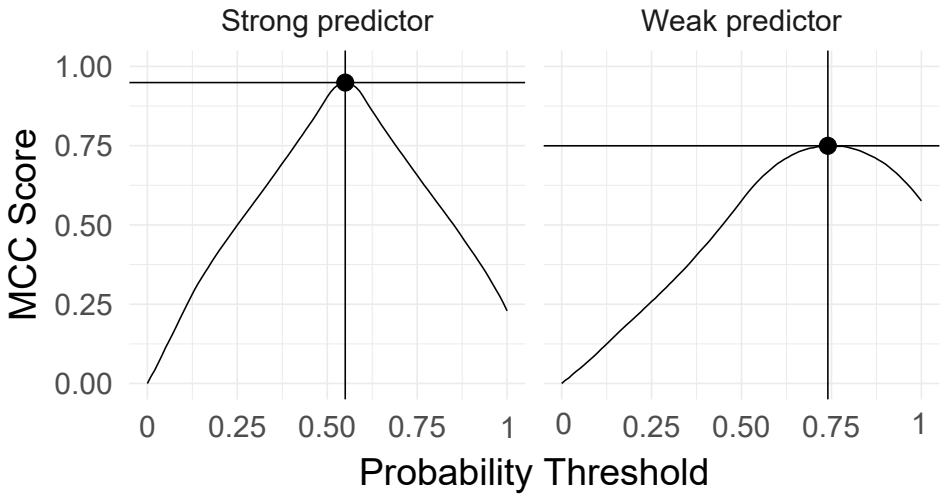

Figure 1: Illustration of MCC optimization for two predictors of different power.

Additionally, the features with the highest correlation coefficients are missing for many observations. These findings are illustrated in Figure 2. Furthermore, non-regularized regression and decision tree models are prone to over-fitting to very large data-sets. Consequently, conventional algorithms for failure detection cannot be applied.

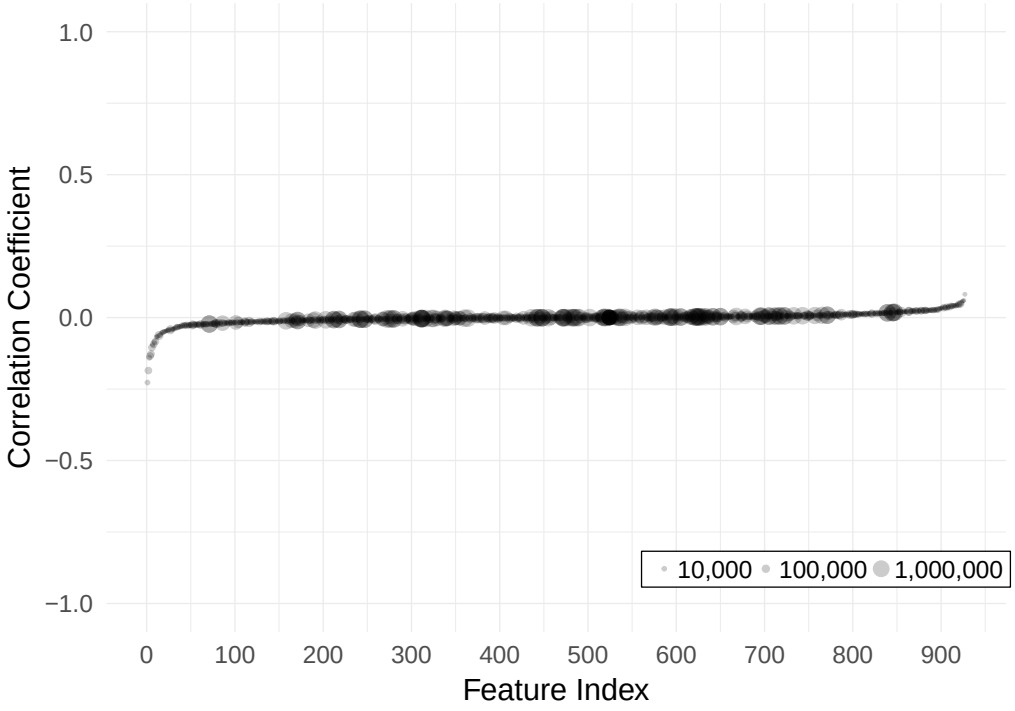

Figure 2: Ordered linear correlation coefficients between response variable and numerical variables (*Node size indicates the number of observations with this feature*).

Given the large number of features and observations, we selected gradient boosting machines (GBM) as an appropriate model class. Gradient boosting is a machine learning algorithm developed in the late ‘90s that produces state-of-the-art results for many commercial and academic applications. Gradient boosting

increases the weight of the samples misclassified by the first model and decreases the weight of the samples that are classified correctly to train another decision tree. This step is repeated n times where n is the number of boosting iterations. In this way, the algorithm always trains models using data samples that are difficult to learn in the previous round, which results an ensemble of models that are good at learning different parts of the training data (Friedman, 2002).

Our initial prediction model follows a naïve approach. As there are no obvious relationships or patterns between the variables and the outcome of the quality control we train a first model on all available raw data and forgo any pre-processing and feature engineering. To this end, we load the complete data-set into the big data machine learning framework H_2O (The H2O.ai team, 2015). This setup is able to perform machine learning tasks even on the entire data-set. While training a gradient boosting machine on the framework used is possible, it becomes a time consuming task. On a 12-core virtual machine with 50 gigabytes of memory the training with 100 boosting iterations and a tree depth of 10 takes about 72 hours. The naïve model yields an MCC score of 0.22 which leaves room for improvement. Some improvements could be achieved by means of hyper-parameter tuning on the algorithmic level. However, given the multi-day length of training runs, the computational costs of this approach would be prohibitive. Therefore, we first seek to reduce the data-set by identifying and removing non-informative variables.

4.3 Feature reduction

To identify non-informative variables, we train separate boosting models using either the numerical or the categorical features. To reduce the computational load and speed up the process, the training is performed on samples of 200,000 rows. Subsequently, the importance of the features is determined by calculating the Kullback-Leibler divergence, also referred to as *information gain* in the machine learning context (Friedman, Hastie, and Tibshirani, 2001). The sum of the information gains for all features always equals one. Therefore, this metric evaluates relative variable predictiveness as opposed to offering an absolute value. Figure 3 summarizes the cumulative information gains for the numeric and the categorical data. It becomes obvious that a relatively small set of features carries the bulk of the relevant information while the biggest part can be considered noise. In case of the numeric variables most information is captured by a subset of only 150 of the 968 features with about 80% captured by the first 50 features. Even more dramatically, out of the 2,140 categorical variables all information gain is captured by only 27 variables with about 80% being condensed in only one variable. We removed all variables without information gain and reduce the number of variables to 150 numerical and 27 categorical features.

Identifying duplicates is the next step to reduce the number of features. However, column-wise comparisons are computationally expensive and not feasible due to the size of the data-set. Hence, we use digest hashing for data de-duplication. To this end, a 32-bit hash is calculated for each column. Subsequently, duplicate features can be identified and removed by a fast pairwise comparison of the hashes. We find that the time stamp variables are recorded for some of the features on a station at the same time. Hence, 1,030 of the time stamps are redundant and can be removed.

The reduced data size allows us to replace the H_2O deployment by a lightweight R implementation. This allows us to leverage extreme gradient boosting (XGB) developed by T. Chen and Guestrin (2016). XGB is a state-of-the-art gradient boosting implementation offering superior speed by eploiting sparsity of feature matrices. This more efficient implementation allows training of models with thousands of boosting iterations within less than a day facilitating efficient hyper-parameter optimization. Furthermore, XGB facilitates direct integration of custom evaluation metrics instead of standard metrics. Consequently, we directly modeled the MCC score as the base for the machine learning algorithm. In combination with the removal of duplicated features, the new model realized an MCC score of 0.24 corresponding to a 9%

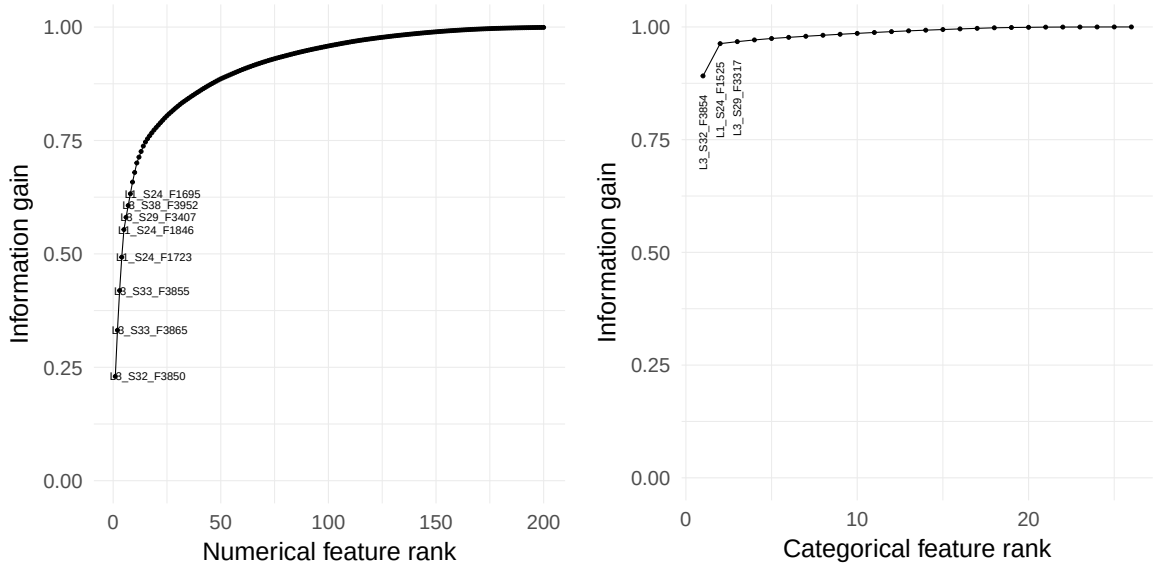

Figure 3: Relative information gain of numerical and categorical features.

increase in predictive power.

5 Feature Engineering

So far our analysis only relied on features provided in the raw data set. To further increase the quality of the model, new features have to be developed. Feature engineering describes the process of aggregating information in the data or adding even new information by using domain knowledge. Following Domingos (2012), this phase is critical to ensure the success of any data mining project. Going beyond basic raw features requires a significant portion of business and process understanding as well as creativity and luck. Going forward, we first retrieve the process structure from the anonymized data-set. This information is used to iteratively refine the predictive model. To this end, we aggregate the existing raw variables to more powerful features by modeling system failure rates and approximating individual production lots.

5.1 Process structures

To obtain a deeper understanding of the underlying processes, we apply process mining to identify relevant patterns. Following Van Der Aalst et al. (2011), this approach can help reveal a process model without any a-priori information. This is especially valuable for the anonymized data-set at hand. To proceed, we filter the individual job data for non-empty features to identify the stations that each job passes through. Subsequently, the stations are ordered by ascending time to create a network representation of the jobs. Figure 4 shows the production network from different perspectives.[1]

First, the complete graph with all occurring edges is visualized. Most parts follow a sequential path through two of the four production lines before they are classified as defective or non-defective. Next, the paths that only occur sporadically are removed by filtering for edges with a frequency exceeding the first quartile. In view of the base failure-rate of 0.58%, the remaining main paths through the manufacturing network all lead to a non-defect classification. We see that some of the stations perform parallel operations (e.g., S14→S18) while others have to be visited in a sequential order (e.g., S12→S13). The last graph visualizes the main process paths resulting in defective products.

[1] Our code to identify the process patterns is publicly available at `www.kaggle.com/gingerman/bosch-production-line-performance/shopfloor-visualization-2-0`

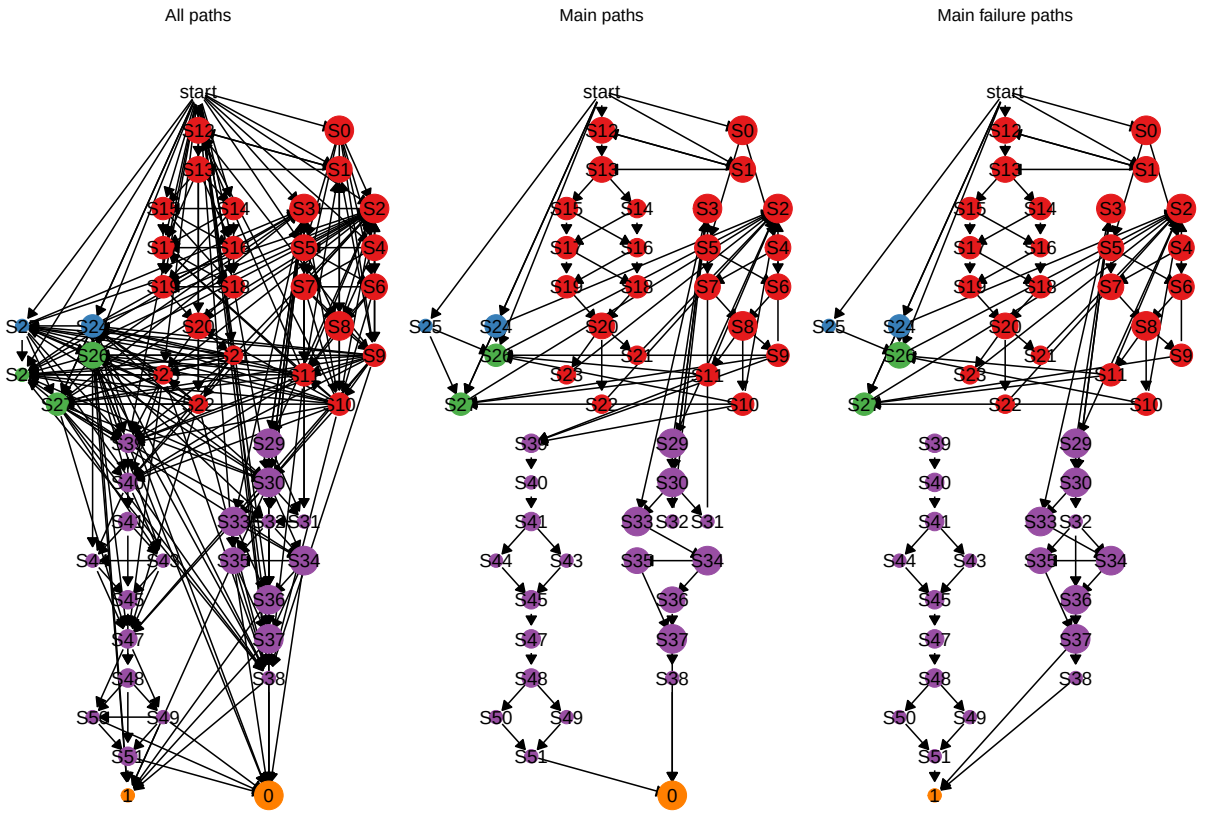

Figure 4: Shopfloor process visualization and illustration of predictive process patterns (*node colors indicate the line, node size indicates usage frequency of the given station*).

5.2 Failure rate features

We can leverage the manufacturing process flows to develop stronger features for the machine learning approach. In particular, we are interested in combining multiple individually weak features into strong combined features.

While individual categorical features are fairly non-predictive (Figure 2), we can condense their joint information content by means of aggregating approaches akin to Hauser et al. (2015). To this end, we determine failure-rates for any given realization of the different categorical variables (including the frequent absence of a variable signified by an "NA" coding). We find that defect-rates are significantly increased for some (possibly seldom) categorical variable values. For example, jobs featuring the value "2" for feature "F3854" have a failure-rate of 16.13% compared to the base rate of 0.58%. Using path-wise aggregation along the process flow we can derive meta-features from the individual defect-rates FR_i. We apply three different aggregation schemes, namely the maximum failure-rate $\max_i FR_i$, the mean failure rate $\frac{1}{|i|}\sum_i FR_i$ as well as the compound rate $\Pi_i(1-FR_i)$. This aggregation approach is illustrated in the top panel of Figure 5. The table in the top-right illustrates that the meta-features exhibit a much higher correlation with the target label than the original set of unprocessed categorical features. This suggests that the meta-features succeed in distilling the information content from the raw feature realizations.

We apply an analogue procedure to capture temporal and station-level failure behavior (lower panel of Figure 5). Item fail rates vary depending on the time they went through a given station. This may be due to machine wear, operator fatigue, material problems or other external influences. Direct encoding of station-time stamp pairs would again yield an enormous number of weak features. By determining station-level fail rates with subsequent path-wise aggregation we can again create condensed and predictive

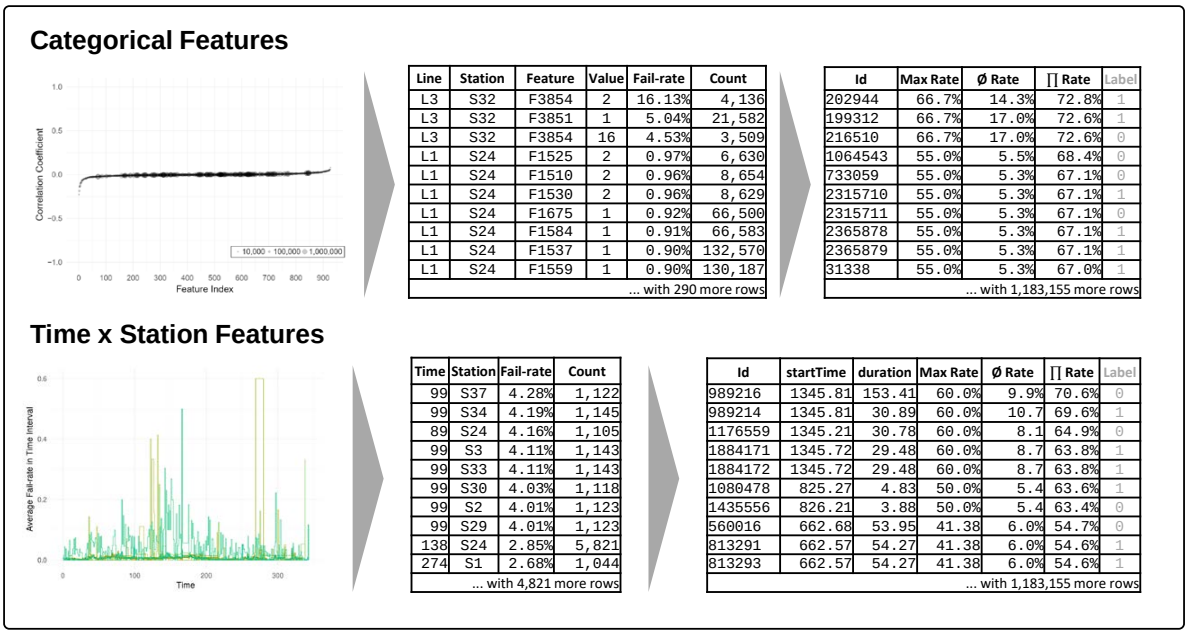

Figure 5: Failure rate feature generation through aggregation of categorical and temporal features.

meta features. These features are complementary to the categorical fail rates and combined with the base model boost predictive power to an MCC value of 0.28.

5.3 Manufacturing batch features

Sequence dependencies are commonplace in manufacturing settings due to grouping of production jobs into batches or production lots. Consequently, failure information on individual items from a lot may be relevant for failure detection with respect to other lot members.

To approximate batches in the data-set we follow two avenues: First we use the time stamps to approximate individual production lots in the data. To this end, the data-set is ordered by the start time and the time difference between two subsequent parts is calculated. Small differences suggest that two parts are part of the same lot while bigger differences indicate different lots. A more focused approach relies on the assumption of the data-set Id column not being random but actually revealing information on the underlying process. By filtering the data to only feature pairs of subsequent Ids (approximately half of the data) we can analyze this hypothesis. Table 1 presents the results of this sequence-level analysis.[2]

Current label	Probability of subsequent "1"	Count
"0"	0.55% $\approx$ base-rate	6,479
"1"	5.79% $\gg$ base-rate	398

(a) Jobs sorted by start time.

Current label	Probability of subsequent "1"	Count
"0"	0.52% $\approx$ base-rate	3,055
"1"	10.08% $\ggg$ base-rate	345

(b) Subsequent Ids sorted by Id.

Table 1: Failure rates dependent on previous observation.

Both approaches reveal greatly increased fail-rates of the subsequent job if the current job is labeled defective—a 10-fold increase in the coarse start time approach and a 20-fold increase with the Id-approach.

[2] Our code to identify the failure probabilities of batches is publicly available at `www.kaggle.com/gingerman/bosch-production-line-performance/errors-in-sequence`

Incorporating this additional information in the form of lagged feature variables greatly improves the model's predictive performance: A minimal model with raw variables and the sequence features yields an MCC score of 0.36, the combination of sequence features and previously developed features achieves an MCC of 0.44.
We also tried to incorporate more distant pairings besides direct sequences. However, these additional sequence feature did not improve model performance but rather deteriorated predictive power.

5.4 Data anomaly features

Going beyond the more natural information sources offered by process and measurement data, a more untypical source of predictive features are data anomalies. In the data-set a hand, an initial screening had highlighted the presence of duplicate entries exhibiting identical numerical feature values despite having different Ids. Such row duplicates can arise in manufacturing systems in the context of communication crashes. SCADA systems will usually repeat the last seen value, so the measurements associated to a given sequence of part numbers correspond to the last correctly received, until the communication is recovered. If such communication failures are triggered by external events (such as power outages) they may also affect the quality of currently manufactured parts. To explore this hypothesis we created row-wise hashes across all numerical features to efficiently detect duplicate rows in the large data-set (Elmagarmid, Ipeirotis, and Verykios, 2007).[3] This thorough search for duplicates confirmed the initial observation of anomalous rows. In total there were 90,000 duplicate rows present in the data. Even more surprising, 3,293 of all 6,879 defective jobs originated from the duplicate data. In turn, the non-duplicate data subset has a corrected fail-rate of 0.33% while the duplicate subset has an eleven times higher fail-rate of 3.63%:

Duplicate row	Fail-rate	Count	Number of Failures
FALSE	0.33%	1,092,975	3,586
TRUE	3.63%	90,772	3,293

Table 2: Fail-rate of duplicated vs. non-duplicated data rows.

The inclusion of the duplicate feature enhanced the predictive performance of our model to reach an MCC score of 0.47.

6 Result Discussion and Interpretation

In the previous sections we developed a failure-detection system for a manufacturing process. The iterative model improvement is summarized in Figure 6. We created a naïve first classifier by training a gradient boosting machine on the raw monitoring data. Subsequently, we improved prediction quality by removing non-essential features and model parameter tuning. Having exhausted the potential of algorithmic model improvement, we extracted information on the manufacturing process through process mining. This enabled us to determine failure rates on a station and time level and to identify manufacturing batches. Meta-features derived from these failure rates and the batches further increase the predictive power of our model significantly. In the last step, data anomalies occurring in manufacturing systems are identified. Utilizing this feature boosts the system's MCC to 0.47. Figure 6 summarizes the performance progression over the course of the competition. Recognizing the incremental nature of the individual improvement steps it becomes evident that successful predictive modeling is not a one-shot endeavour but rather necessitates diligent and persistent development.

[3] The hash creation for the full data set took about 40 minutes.

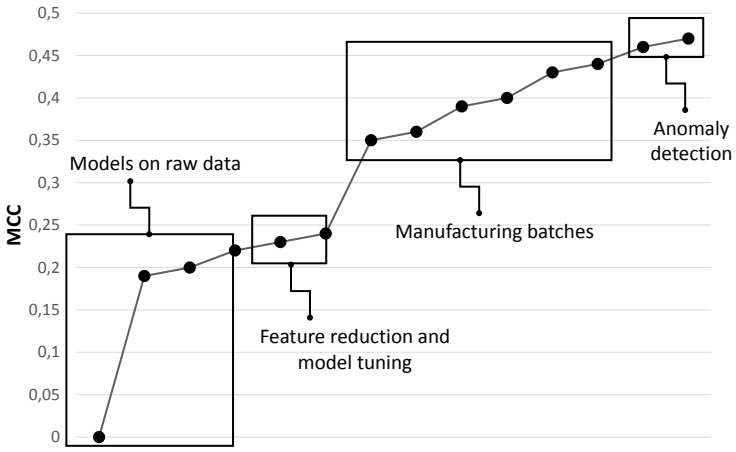

Figure 6: Performance development of the predictive model.

Competing models created by other participants rely on algorithmic approaches to extract meaningful features. Even though Pavlyshenko (2016) and Mangal and Kumar (2016) follow different approaches, both models report similar MCC scores of roughly 0.41. This confirms the suitability of our approach of combining predictive modelling with business process analysis. Indeed, the winning team reached an MCC of 0.52 applying similar features. However, they combined 165 different model instantiations for their final prediction. This highlights the potentials of algorithmic tuning to maximize the power of a given feature set.

In the age of big data, researchers as well as practitioners can no longer rely exclusively on standard statistical methods (e.g., linear regression) to generate business insights from large data sets. Rather, the use of machine learning becomes inevitable as these approaches are better suited to handle thousands of variables or work with unstructured data. Breiman (2001) and Shmueli (2010) show that these approaches are of special importance in studies aiming at prediction instead of description. The main advantage of state-of-the-art machine learning algorithms is that they make less statistical assumptions and are able to work with data-sets of very high dimensionality. Additionally, these methods are able to not only capture non-linear relationships but also pick up higher-order interaction effects between variables. On the downside, these black-box algorithms (e.g., gradient boosting machines) typically generate incomprehensible models and rules. Yet, the interpretability of the rules used by the algorithms is important if subsequent actions based on the predictions are to be taken by human decision-makers (Diakopoulos, 2014; Martens and Provost, 2014).

Answering the need for comprehensible prediction models as identified by Breuker et al. (2016), we recombine the trees determined by our gradient boosting machine to one aggregated tree. To this end, we make use of the fact that all 7,000 binary trees of the final model have the same depth and therefore the same number of nodes. Consequently, each node has 7,000 representations. We can determine the importance of a feature by counting how often it appears on a certain node. Figure 7 visualizes the aggregated tree with the three most frequent features at each node. As in standard decision trees, variables occurring earlier in the tree are more important than variables appearing at the end. To this end, the value of the engineered features becomes evident. The defect-rates on a machine level as well as the production lot approximation emerge as highly predictive while the raw features show up deeper in the tree. Furthermore, it becomes evident that black-box machine learning models and process mining approaches can work in unison. For instance, the boosted trees identify station 33 as a possible weak point with the feature recorded on this station ("L3-S33-F3857"). Looking at the shop-floor process structure (Figure 4) confirms this station's central role in the manufacturing process.

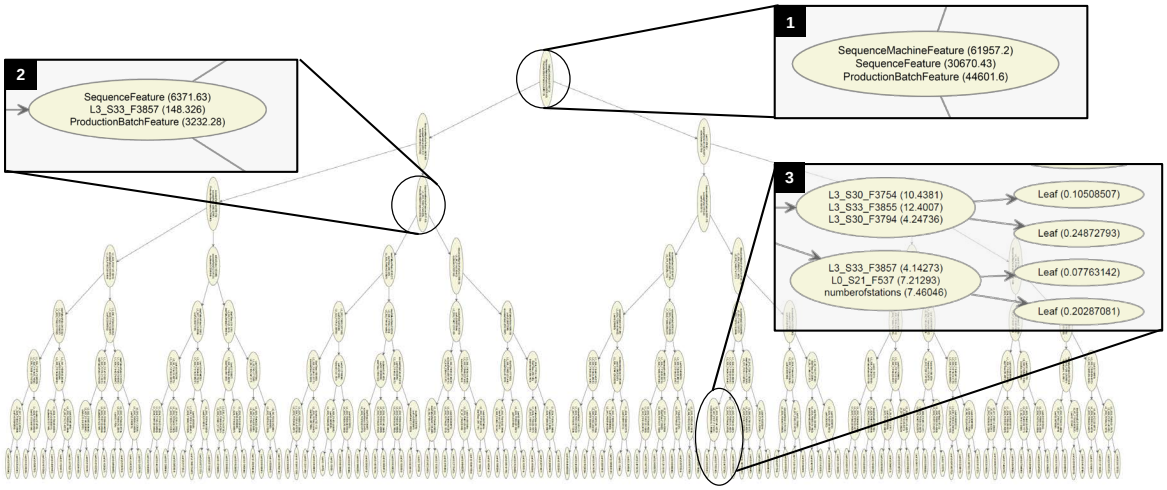

Figure 7: Retreiving system insights from black box models.

7 Conclusion and Outlook

Using a large data-set from a major manufacturing company we showcased the development, refinement and evaluation of a predictive analytics system. After identifying an appropriate metric as well as a suited machine learning algorithm we recognize that mere algorithmic tuning is not sufficient for a satisfactory model. Hence, we leveraged process-mining to derive powerful features based on the manufacturing process structure. Thereby, we illustrate that predictive modeling and business process analysis are highly complementary. Although we showcased the implementation with a data-set from a specific manufacturing process, we are confident that the approach is generic and straight-forward to transfer and implement for other use cases as the data was anonymized and no additional information about the underlying process was available.

Our findings regarding the application of big data analytics are twofold. It becomes evident that simply plunging a huge amount of data into “smart” algorithms is not the silver bullet a lot of researchers and practitioners expect it to be. Rather, we show that constant improvement, feature engineering and consolidation will complementary improve the predictive power of a business analytics system. In order to further increase the predictive power higher level modeling approaches could be applied. A first step would be the training of two distinct models for the duplicates and non-duplicates identified during the anomaly detection. Going further a set of different black-box models should be trained and combined to generate predictions from stacked predictors. Such an ensemble would come at the cost of interpretability and necessitate new methods to answer the need for comprehensibility. The increasing complexity in the data and the successful combination of process mining and machine learning emphasize the need for analytic skills as well as business understanding and showcases the comparative advantage of information systems research as a cross-disciplinary field (Agarwal and Dhar, 2014). Going forward, more practice-oriented and case-based research in different domains will be necessary to demonstrate the merit of embedding analytics solutions in corporate decision-making (Gordon, Blake, and Shankaranarayanan, 2013). Recent contributions have pointed in this direction with applications in supply chain management (Trkman et al., 2010), infrastructure management (Gust et al., 2016; Laubis, Simko, and Schuller, 2016) or transportation (Wagner, Brandt, and Neumann, 2016).

References

Agarwal, R. and V. Dhar (2014). "Editorial — Big Data, Data Science, and Analytics: The Opportunity and Challenge for IS Research." *Information Systems Research* 25 (3), 443–448.

Breiman, L. (2001). "Statistical Modeling: The Two Cultures." *Statistical Science* 16 (3), 199–215.

Breuker, D., M. Matzner, P. Delfmann, and J. Becker (2016). "Comprehensible Predictive Models for Business Processes." *MIS Quarterly* 40 (4).

Chawla, N. V., N. Japkowicz, and A. Kotcz (2004). "Editorial: special issue on learning from imbalanced data sets." *ACM Sigkdd Explorations Newsletter* 6 (1), 1–6.

Chen, H., R. H. Chiang, and V. C. Storey (2012). "Business Intelligence and Analytics: From Big Data to Big Impact." *MIS Quarterly* 36 (4), 1165–1188.

Chen, T. and C. Guestrin (2016). "XGBoost: A Scalable Tree Boosting System." In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16, pp. 785–794.

Davenport, T. H. and J. G. Harris (2007). *Competing on analytics: The new science of winning*. Harvard Business Press.

Davis, J. V., B. Kulis, P. Jain, S. Sra, and I. S. Dhillon (2007). "Information-Theoretic Metric Learning." In: *Proceedings of the 24th international conference on Machine learning*. ACM, pp. 209–216.

Diakopoulos, N. (2014). "Algorithmic Accountability Reporting: On the Investigation of Black Boxes." *Tow Center for Digital Journalism*.

Domingos, P. (2012). "A Few Useful Things to Know about Machine Learning." *Communications of the ACM* 55 (10), 78–87.

Elmagarmid, A. K., P. G. Ipeirotis, and V. S. Verykios (2007). "Duplicate Record Detection: A Survey." *IEEE Transactions on knowledge and data engineering* 19 (1), 1–16.

Flach, P. A. (2003). "The Geometry of ROC Space: Understanding Machine Learning Metrics through ROC Isometrics." In: *Proceedings of the 20th International Conference on Machine Learning*, pp. 194–201.

Friedman, J. H. (2002). "Stochastic Gradient Boosting." *Computational Statistics & Data Analysis* 38 (4), 367–378.

Friedman, J. H., T. Hastie, and R. Tibshirani (2001). *The Elements of Statistical Learning*. Vol. 1. Springer.

Gordon, S., R. Blake, and G. Shankaranarayanan (2013). "Case-Based Research in Information Systems: Gaps and Trends." *JITTA: Journal of Information Technology Theory and Application* 14 (2), 47.

Gust, G., C. Flath, T. Brandt, P. Ströhle, and D. Neumann (2016). "Bringing Analytics into Practice: Evidence from the Power Sector." In: *Proceedings of the 37th International Conference on Information Systems*.

Hauser, M., D. Zügner, C. Flath, and F. Thiesse (2015). "Pushing the limits of RFID: Empowering RFID-based Electronic Article Surveillance with Data Analytics Techniques." In: *Proceedings of the 36th International Conference on Information Systems*.

Kaggle.com (2016). *Bosch Production Line Performance*. URL: `www.kaggle.com/c/bosch-production-line-performance`.

Kuhn, M. and K. Johnson (2013). *Applied Predictive Modeling*. Springer.

Laubis, K., V. Simko, and A. Schuller (2016). "Road Condition Measurement and Assessment: A Crowd Based Sensing Approach." In: *Proceedings of the 37th International Conference on Information Systems*.

Lee, J., E. Lapira, B. Bagheri, and H.-a. Kao (2013). "Recent Advances and Trends in Predictive Manufacturing Systems in Big Data Environment." *Manufacturing Letters* 1 (1), 38–41.

Lustig, I., B. Dietrich, C. Johnson, and C. Dziekan (2010). "The Analytics Journey." *Analytics Magazine*, 11–13.

Mangal, A. and N. Kumar (2016). "Using Big Data to Enhance the Bosch Production Line Performance: A Kaggle Challenge." *arXiv preprint arXiv:1701.00705*.

Manyika, J., M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. H. Byers (2011). *Big Data: The Next Frontier for Innovation, Competition, and Productivity*. Tech. rep. McKinsey Global Institute, pp. 1–156.

Martens, D. and F. Provost (2014). "Explaining Data-Driven Document Classifications." *MIS Quarterly* 38 (1), 73–99.

Maurya, A. (2016). "Bayesian Optimization for Predicting Rare Internal Failures in Manufacturing Processes." In: *Big Data (Big Data), 2016 IEEE International Conference on*. IEEE.

Mosallam, A., K. Medjaher, and N. Zerhouni (2016). "Data-driven prognostic method based on Bayesian approaches for direct remaining useful life prediction." *Journal of Intelligent Manufacturing* 27 (5), 1037–1048.

Müller, O., I. Junglas, J. v. Brocke, and S. Debortoli (2016). "Utilizing Big Data Analytics for Information Systems Research: Challenges, Promises and Guidelines." *European Journal of Information Systems* 25 (4).

Pavlyshenko, B. (2016). "Machine Learning, Linear and Bayesian Models for Logistic Regression in Failure Detection Problems." *arXiv preprint arXiv:1612.05740*.

Powers, D. M. (2011). "Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation." *Journal of Machine Learning Technologies*.

Reddy, A. S. (2016). "Why IoT Analytics Are a Manufacturer's Most Important Tool." *Harvard Business Review*.

Schwegmann, B., M. Matzner, and C. Janiesch (2013). "A Method and Tool for Predictive Event-Driven Process Analytics." In: *11. Internationale Tagung Wirtschaftsinformatik (WI)*. Merkur, pp. 721–735.

Sharma, R., S. Mithas, and A. Kankanhalli (2014). "Transforming Decision-Making Processes: A Research Agenda for Understanding the Impact of Business Analytics on Organisations." *European Journal of Information Systems* 23 (4), 433–441.

Shmueli, G. et al. (2010). "To Explain or to Predict?" *Statistical Science* 25 (3), 289–310.

Shmueli, G. and O. R. Koppius (2011). "Predictive Analytics in Information Systems Research." *MIS Quarterly* 35 (3).

Sipos, R., D. Fradkin, F. Moerchen, and Z. Wang (2014). "Log-based Predictive Maintenance." In: *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1867–1876.

The H2O.ai team (2015). *h2o: R Interface for H2O*. R package version 3.1.0.99999.

Trkman, P., K. McCormack, M. P. V. De Oliveira, and M. B. Ladeira (2010). "The Impact of Business Analytics on Supply Chain Performance." *Decision Support Systems* 49 (3), 318–327.

Van Der Aalst, W., A. Adriansyah, A. K. A. De Medeiros, F. Arcieri, T. Baier, T. Blickle, J. C. Bose, P. van den Brand, R. Brandtjen, J. Buijs, et al. (2011). "Process Mining Manifesto." In: *International Conference on Business Process Management*. Springer, pp. 169–194.

Vukšić, V. B., M. P. Bach, and A. Popovič (2013). "Supporting Performance Management with Business Process Management and Business Intelligence: A Case Analysis of Integration and Orchestration." *International Journal of Information Management* 33 (4), 613–619.

Wagner, S., T. Brandt, and D. Neumann (2016). "In free float: Developing Business Analytics support for carsharing providers." *Omega* 59, 4–14.