H.J.A. de Krom

Supplier Disruption Prediction using Machine Learning in Production Environments

Supplier Disruption Prediction using Machine Learning in Production Environments

By

Hubertus Joannes Aimé de Krom

Master Thesis

in partial fulfilment of the requirements for the double degree of

Master of Science in Mechanical Engineering
at the Department Maritime and Transport Technology of Faculty Mechanical, Maritime and Materials Engineering of Delft University of Technology

&

Master of Science in Civil Engineering
at the Department Transport & Planning of Faculty of Civil Engineering and Geosciences of Delft University of Technology

to be defended publicly on Wednesday April 28, 2021 at 11:00 AM

Student number:	4349784	
ME track:	Transport Engineering and Logistics (Mechanical Engineering)	
CIE track:	Transport & Planning (Civil Engineering)	
Report number:	2021.MME.8512	
Thesis committee:	Prof.dr.ir. L.A. Tavasszy	TU Delft, CiTG, Chair
	Dr. B. Wiegmans	TU Delft, CiTG
	Ir. M.B. Duinkerken	TU Delft, 3mE
	Ir. M.J.J. Hutten	Philips
Date:	April 20, 2021	

An electronic version of this thesis is available at http://repository.tudelft.nl/.

Front page: Philips collection

Preface

This thesis presents my first and interesting hands-on experience with machine learning, applied in supplier disruption prediction in collaboration with Philips. This research finalises both my masters Transport Engineering & Logistics at Mechanical Engineering and Transport & Planning at Civil Engineering, concluding my academic years at Delft University of Technology.

Supplier disruption prediction is a topic in which the fields of Mechanical Engineering and Civil Engineering meet. The planning and operations of production are connected with the logistic activities involved to fulfil material demand via different transportation modes and network structures. Coordination is required to facilitate this efficiently in which prediction of upcoming disruptions in material availability can assist. From a Mechanical Engineering perspective, this research is interesting since it combines operations, coordination and production. The supply chain and interactions suggest similarities with distributed control, of which the results can be valuable from a Civil Engineering perspective. Incorporating influences of the supply chain (design) and corresponding transport network in disruption prediction, inefficiencies in operations and logistics may be identified, which can improve supply chain resilience and efficiency. Additionally, when disruptions can be addressed sooner, different (less polluting) transportation modes might be viable increasing the sustainability or allowing for a redesign of the underlying transport (network).

First, I would like to express my gratitude towards my graduation committee and Philips for their support and freedom throughout this research. Especially in the beginning during the definition phases, it has not always been easy to work individually from home. Their belief and understanding really helped me to continuously progress and move forward.

Furthermore, I want to thank my daily university supervisors Bart and Mark for their support and feedback throughout the course of this research. Especially the unprecedented involvement of Bart was unexpected and highly appreciated, which helped me develop personally as well. Additionally, I would like to explicitly thank Martin, my daily supervisor from Philips. The animated discussions, feedback and recurring reflections provided me with inspiring (personal) insights and practical knowledge which will definitely be valuable in my future career.

Finally, I would like to thank my family and friends for their support throughout my studies at the university and in particular my parents and little brother who managed to bear with me during my research at home.

Bart de Krom
Raamsdonksveer, April 2021

Summary

Many developments within supply chains (SCs) and supply chain management (SCM) have taken place in the last years leading to increasing SC size, global spread and interconnections between other SCs. This results in more complex, vulnerable and uncertain SC which in its turn could lead to undesired losses in shareholder value, sales, customer satisfaction and reputation. This increase in complexity and vulnerability urges an increase in monitoring of SC performance. Especially in production oriented SCs disruptions in the material flow could influence downstream supply chain performance and continuity significantly. SC integration and information transparency are often brought forward in this context. However, achieving such integration or transparency is limited to mainly conceptual studies and SC actors can be reluctant in achieving the desired transparency as it could require sharing sensitive company information. Therefore, this research focusses on exploring the potential of machine learning (ML) using a manufacturer's point of view and data available to the manufacturer to assist in predicting and mitigating material-oriented supplier disruptions and therewith increase SC resilience.

A novel methodology aimed to develop predicting classification models assisting on operational and tactical level is proposed and applied in a case study at Philips' production facility in Best to verify the expected contributions. The methodology consists of six steps (see Figure i.1) incorporating a binary and novel multiclass classification extension while following a bottom-up approach considering individual suppliers and custom supplier groups rather than all suppliers combined. This reduces initial complexity, increases expected performance and transparency which is expected to lead to easier verification and acceptance of obtained results by the targeted user group (buyers).

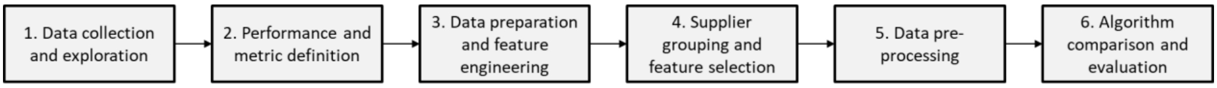
1. Data collection and exploration
2. Performance and metric definition
3. Data preparation and feature engineering
4. Supplier grouping and feature selection
5. Data pre-processing
6. Algorithm comparison and evaluation

Figure i.1: Overview of steps in proposed methodology

The steps can be summarised as follows:

1. *Data collection and exploration* incorporates system analysis with data collection and focuses on understanding the production system, its data generation (influences), limitations and prediction goal.
2. *Performance and metric definition* focusses on translating the prediction goal to suitable metrics for model performance evaluation throughout model development.
3. During *Data preparation and feature engineering,* the collected data elements are transformed to suitable formats for machine learning algorithms, incorrect data entries or noise are removed, and data characteristics (features) are added using the available raw data, domain knowledge and experience.
4. *Supplier grouping and feature selection* focusses on defining additional supplier groups using the individually considered suppliers. The dimension and complexity of the resulting subsets consisting of the individually considered suppliers and supplier groups are thereafter reduced. For each subset recursive feature elimination based on feature permutation importance is used to select the most contributing features.
5. *Data pre-processing* incorporates ML-algorithm specific transformations as data scaling or normalisation and the possibility to apply resampling on each considered subset to reduce negative impacts of data imbalance on the resulting prediction performance. No sampling, over-sampling, under-sampling and a hybrid of over- and under-sampling are applied.

6. *Algorithm comparison and evaluation* is the final step in which algorithm parameter grids for five[1] different machine learning algorithms are used to evaluate and compare prediction performances of the various algorithms for the individual suppliers and supplier groups. Additionally, post-processing by means of threshold tuning for binary classification is incorporated to shift prediction performance in the direction of the main performance metric at some cost of different less important metrics.

The methodology is applied to a selection of 21 suppliers of Philips' production facility in Best. Three years of historical purchase order (PO) data has been made available, containing order characteristics like creation-, due- and receipt date and quantity. The available data allows the focus on predicting delivery performance for PO deliveries, which in the case study translates to classifying whether a PO delivery will be delayed (binary formulation) or extremely early, early, on-time, delayed or extremely delayed (multiclass formulation).

Consultation with practitioners regarding potentially valuable features based on their experience and domain knowledge led to the definition of three feature domains: "Order", "Supplier-material" and "Dynamic 'environment'"[2]. Suggested features for those domains which could be created using the available data were added to the dataset. Besides the 21 individual suppliers, three additional supplier groups (consisting of these 21 suppliers) based on supplier location (Western Europe, Rest of Europe and Rest of World) were manually defined to potentially discover region-related influences on delivery performance. For each individual supplier and supplier group feature selection has been conducted, leading to the identification of common important features regarding considered material, provided delivery time, outstanding supplier orders or quantity, previous delivery performance and the week and day of the week when the order is due. The recurring importance of the latter was unexpected and consultation with practitioners led to the idea that this indicates inefficiencies in process-related aspects as day-offs or delayed invoicing. This illustrates that feature selection and importance can assist in identifying potential root causes for inefficiencies in operations or supplier relations, contributing towards mitigating disruptions on a tactical level.

After feature selection, prediction models for each individual supplier and supplier group are trained using a defined parameter grid for each of the selected algorithms and sampling techniques. This resulted in varying prediction performances reaching MCC[3] scores up to 0.9, accompanied with 98% accuracy, 100% precision and 83% recall in the binary problem formulation. In the multiclass case lower performances are observed (0.75 MCC, 88% accuracy, 85% macro-precision and 80% macro-recall), which can be expected since the introduction of additional prediction classes increases the complexity of the classification task and data requirements.

[1] Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, and eXtreme Gradient Boosting

[2] "Order": features which are characteristic for the PO record. Examples: creation- and due date.
"Supplier-material": features regarding the supplier-material relation. Examples: lead time and price.
"Dynamic 'environment'": features with dynamic behaviour. Examples: outstanding order quantity at the moment of ordering and preceding delivery performance.

[3] MCC (Matthew's Correlation Coefficient): correlation between prediction model and data. Accuracy: percentage of correct classifications. Precision: percentage of predictions correctly classified as delayed. Recall: percentage of actual delayed deliveries correctly identified.

The obtained results showed varying prediction performances for individual suppliers for which no specific reason in the available dataset could be identified. Given the same methodology applied for each individual model, it is expected that the differences result from random supplier behaviour or additional influences as sharing demand forecasts or communication between buyers and suppliers. In this way, differences between supplier performance can indicate random supplier behaviour or suppliers who often need additional steering. Therewith, it can assist in identifying suppliers where the SC relation might need to be re-designed, potentially leading to tactical improvements.

The high performing prediction models can be applied on an operational level assisting planners and buyers in time prioritisation, evaluating production plan feasibility, and increasing the time window in which mitigating measures can be applied. This can lead to less expensive mitigating measures and use of different transport alternatives as well.

However, the difference in individual supplier performances limits general implementation and urges additional research focussing on increasing overall prediction performance in the binary and multiclass formulation (for the specific case of Philips' production facility). Suggested is to initially focus on the differences between suppliers and their corresponding model performances by investigating internal differences regarding different ways-of-working and communication of buyers with suppliers. Thereafter, additional data collection is suggested to incorporate and account for the currently insufficiently accounted aspects.

Additionally, it is suggested to apply the methodology in different production environments to verify its generalised formulation and expected applicability.

Table of Contents

List of Abbreviations

ERP	Enterprise Resource Planning
FN	False Negative (prediction)
FP	False Positive (prediction)
IGT	Image Guided Therapy
LR	Logistic Regression
MCC	Matthew's Correlation Coefficient
ML	Machine Learning
MR	Magnetic Resonance
MRP	Material Requirements Planning
PO	Product/Purchase Order
PRC	Precision-Recall Curve
RF	Random Forest
RFE	Recursive Feature Elimination
ROC	Receiving Operator Characteristic-curve
RUS	Random Under-Sampling
SC	Supply Chain
SCM	Supply Chain Management
SMOTE	Synthetic Minority Over-sampling Technique
SMOTENC	Synthetic Minority Over-sampling Technique-Nominal Continuous
SVM	Support Vector Machine
TN	True Negative (prediction)
TP	True Positive (prediction)
XGB	eXtreme Gradient Boosting

List of Figures

List of Tables

1 Introduction

This chapter presents the purpose and general approach of the conducted research. At first, in section 1.1, a brief introduction into the context of this research and the problem statement is given. Thereafter in section 1.2, this problem is rephrased into the main research question which is divided into four sub-questions. Section 1.3 puts this research in perspective by presenting the research relevance and contributions. Section 1.4 concludes the introduction by presenting the outline of the rest of the report.

1.1 Research context and problem statement

Many developments within supply chains (SC) and supply chain management (SCM) have taken place in the last years. Examples are the increase in data collection resulting from the fourth industrial revolution (Industry 4.0), the increased demand in new emerged rural areas, changing labour demographics and the interest in adopting the concept of circular economy into SCs and SCM (Alicke & Balaji, 2013; Alicke, Rachot, & Seyfert, 2016). These developments lead to an increase in SC size, global spread and interconnections between other SCs leading to more complex, vulnerable and uncertain SCs (Zhao, Ji, & Feng, 2020), which in its turn could lead to undesired losses in shareholder value, sales, customer satisfaction and reputation (Behzadi, O'Sullivan, Olsen, Scrimgeour, & Zhang, 2017).

This increase in complexity and vulnerability of supply chains, urges an increase in monitoring of supply chain performance throughout the entire chain to maintain healthy operations and prevent supply chain (actors') degradation and losses. Especially in a production-oriented supply chain, disruptions in the material flow could influence downstream supply chain performance and continuity significantly.

Terms as SC integration and information transparency are often connected to this overarching SCM monitoring task, since by providing information transparency between supply chain actors, this monitoring and additional insights could be obtained and negative impacts resulting from disruptions mitigated (Alicke, Azcue, & Barriball, 2020). However, achieving such integration or transparency is limited to mainly conceptual studies with the exception of some automotive companies (Alicke et al., 2016). Supply chain actors can be reluctant in achieving the desired transparency as it could require sharing sensitive company information (Atallah, Elmongui, Deshpande, & Schwarz, 2003).

Therefore, one could question the direct necessity of the desired full transparency within SCs given its complications and wonder if value could be obtained by pro-actively monitoring SC components in the direct area around an SC actor, as illustrated in Figure 1.1 below. Given the increased data collection and interactions and communication with first (and higher) tier suppliers, potentially disruptions on a more local SC level can be predicted and mitigated. Especially with the increasing interest in "Big Data" and "Machine Learning (ML)" in SCM to gain additional insights in supply chain operations to improve overall performance and reduce risks (Nguyen, Zhou, Spiegler, Ieromonachou, & Lin, 2018; Ni, Xiao, & Lim, 2020; Wang, Gunasekaran, Ngai, & Papadopoulos, 2016).

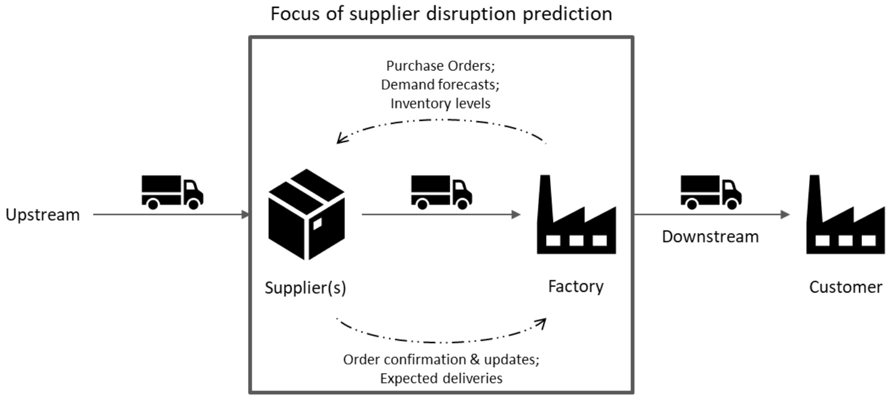

Figure 1.1: Focus area of (first tier) material-oriented supplier disruption.

This concept of predicting material-related supplier disruptions using ML on a local level has been the topic of a few recent studies. Within these studies, binary classification problems have been formulated for which ML has been applied to available historical production data to predict (1) whether an order from a supplier is expected to be delayed (Baryannis, Dani, & Antoniou, 2019; Brintrup et al., 2020) or (2) whether an item is expected to go out-of-stock (De Santis, De Aguiar, & Goliatt, 2018; Hajek & Abedin, 2020).

Since it is a new field considered only by a few studies, one could question if their findings can be generalised. In addition, the amount of research focussing on practical applications of ML in SCM is scarce and additional (decision supporting) practical research is necessary (Ni et al., 2020).

1.2 Research objective

Empirical and more advanced studies are needed to further investigate the possibilities of ML in predicting and mitigating risk resulting from supplier disruptions. Therefore, this research focusses on ML applications in material-oriented supplier disruption prediction (hereafter supplier disruption prediction), while relaxing the simplification towards binary classification. A generalised approach is proposed to explore the potential of ML in combination with commonly available data in production systems, which is applied to a case study involving Philips' Magnetic Resonance (MR) and Image Guided Therapy (IGT) factory. The research focus is translated in the following research question:

How can machine learning be applied to assist in the mitigation of material-oriented supplier disruptions in production environments?

The assistance of mitigating material disruptions will translate into developing prediction models aiming to classify expected delivery performance on an operational level, while potentially also revealing (unexpected) underlying indicators for tactical supplier performance improvements.

To structure and steer this research, the stated research question is divided into four sub-questions, which are as follows:

1. What system and characteristics are considered when researching the possibilities of machine learning-based supplier disruption prediction?
2. Which machine learning algorithms and techniques have been used in literature regarding supplier disruption prediction?
3. How could machine learning be applied in predicting supplier disruptions in a general production environment?
4. What level of prediction performance can be achieved in the specific case of Philips' production facility?

1.3 Relevance of research

The main contribution of this research is the development and application of a generalised methodology to apply machine learning in supplier disruption prediction. Additionally, the following contributions are made:

1. This research focusses on individual suppliers and supplier groups rather than the entire dataset, illustrating potential implications and limitations of preceding work regarding interpretability and applicability.
2. The implemented methodology is the first to consider and introduce multiclass classification in supplier disruption prediction.
3. The application to a case study in addition to the scarcely available applied studies regarding ML-based decision support research for material management in supply chain (risk) management.
4. Feature domains and suggestions are formulated which can serve as guidance for feature engineering in following supplier disruption prediction research.
5. Future research directions towards increasing the general performance and applicability of the methodology are formulated.

Successful creation of ML models for suppliers could increase Supply Chain Resilience by contributing in the '*anticipation'/'readiness'* and '*resistance'/'response'* dimensions of the Supply Chain Resilience Frameworks defined by Singh, Soni, and Badhotiya (2019) and Han, Chong, and Li (2020). High performing prediction models could assist to detect potential disruptions sooner (*visibility*), resulting in more time to mitigate the disruption or its negative effects (*flexibility*). Additionally, more structural causes for disruptions might be (indirectly) brought forward (*awareness*), which can reduce the possibility for disruptions to occur. The increase in *visibility* and *awareness* contributes towards the first dimension '*anticipation'/'readiness'*, whereas the increase in *flexibility* assists in the second dimension '*resistance'/'response'* dimension.

From an economic perspective, these contributions are valuable for manufacturers, since the amount of expensive last-minute mitigation measures will be reduced, and production continuity can be better maintained. Simultaneously, customer experience could improve. From an environmental perspective, less express services (as air transport) might be needed, opening the possibility to use (slower) less polluting transportation modes.

1.4 Outline

In chapter 2, a generalised production system is sketched which incorporates the scope of the indicated area in Figure 1.1. Chapter 3 reviews which ML algorithms and techniques have been used in previous research. This aids the development of a generalised ML approach for disruption prediction in chapter 4. Chapter 5 presents the application of the approach including its results in a case study regarding Philips' production facility. Chapter 6 concludes this report by stating the findings and recommendations.

2 Production environments for supplier disruption prediction

For production, its planning and fulfilment different systems and actors interact. Figure 2.1 depicts a production chain on an operational level from suppliers till order fulfilment including important interactions between actors. This depicted chain will serve as basic system for this research. Within the figure, the left side represents the material demand in the chain, where the right side represents material supply. The physical and information flows within the production chain will be discussed in section 2.1. Section 2.2 describes the storage of data from the manufacturer's perspective and dwells on its availability.

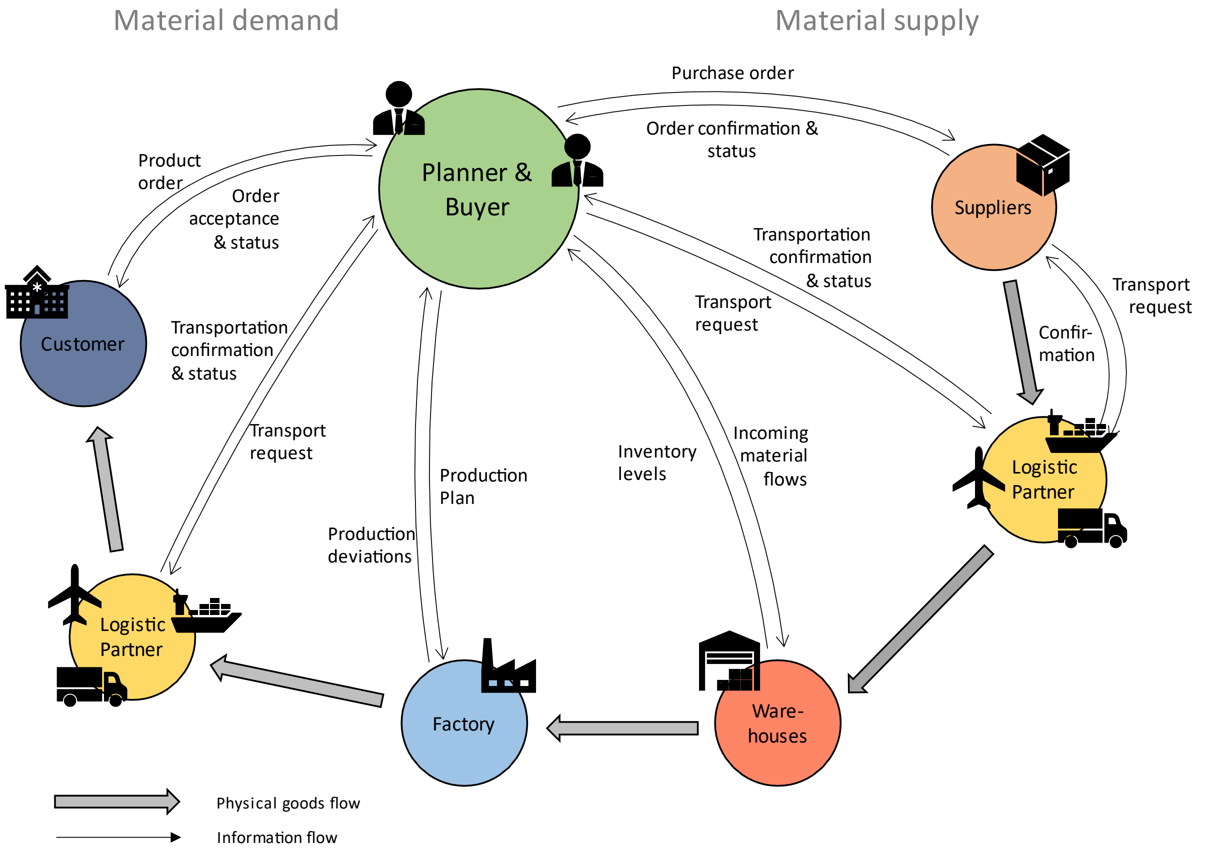

Figure 2.1: Schematic representation of physical and information flows in a production chain.

2.1 Physical and information flows

In Figure 2.1 the flow of physical goods is visualised by the large grey arrows. The physical flow originates from a supplier from which the material is transported by a logistic partner to the warehouse of the manufacturer. Depending on the requirements, different transport modalities could be selected and combined. Once delivered to the warehouse, the material is stored until it is needed on the production floor. Intra-facility transport delivers the material to the production floor and once the material is processed and the final product is manufactured, the final product is transported to the customer. Again, this transportation can be fulfilled by different or combined transport modalities.

The initiation and fulfilment of this physical flow is accompanied by various information flows, of which the most relevant are depicted in Figure 2.1 by solid lines. The customer creates a demand for a product for which an order is created and placed. This product order is processed and its fulfilment feasibility is determined based on the currently available production plan and material availability. When expected that the order can be fulfilled, the order is accepted and status updates can be communicated to the customer over time.

The acceptance of the order leads to a change in material demand, which is held against current inventory levels and incoming material flows. In case the changed demand would exceed inventory, purchase orders (POs) are created by buyers and sent to suppliers. PO acceptance or rejection is returned, such that the incoming material flows can be updated.

Depending on the supplier and its contract, the supplier or the buyer arrange transport between the supplier and the warehouse by hiring a logistic partner. Once the material is produced, the logistic partner collects and fulfils transport to the desired warehouse, whereafter the inventory levels are updated when the material is received and processed. After processing, the created PO is marked as complete.

Based on the production plan and current deviations from the plan, the completed production time can be estimated such that transport towards the customer can be arranged. By contacting a logistic partner, the products ordered by the customer can be transported once produced, closing the production chain.

2.2 Data storage and availability

In production environments, material management often uses a material requirements planning (MRP) framework to monitor and maintain material availability such that production plans can be successfully executed (Sridharan & La Forge, 2000). By projecting (expected) material demand onto current and incoming material flows using static data as lead times, shortages are identified which buyers can prevent or act upon by amongst others creating POs for suppliers and registering them in the MRP system.

Since the creation and fulfilment of material demand of suppliers is stored in an MRP system, the basic relation and behaviour of suppliers with respect to the manufacturer is present. Therefore, an MRP system serves as basis for supplier disruption prediction. Conceptually, any production environment working with a structured MRP system storing historical PO data should be eligible to explore the potential of ML in supplier disruption prediction, provided that the stored data is of sufficient quality and quantity for the specific prediction target.

Additionally, demand forecasts and inventory levels could be relevant for predicting supplier disruptions and material availability. Especially when this type of information is shared with suppliers and therefore could influence their behaviour and performance. Depending on the size and organisational structure of a manufacturer, the systems used to communicate and potentially store this information can significantly differ. These systems can range from locally and temporarily storing systems to collecting on a higher global level by means of an overarching Enterprise Resource Planning (ERP) system, which is an extension of the previously mentioned MRP framework ("Enterprise Resource Planning (ERP)," 2000). Depending on the availability, quantity and quality of these types of data in a production environment, the scope and coverage of dependencies and influences in supplier disruption prediction can be broadened or shifted.

2.3 Concluding system characteristics

When focussing on supplier disruption prediction and necessary system characteristics, production systems as depicted in Figure 2.1 in which material supply is initiated, monitored and stored in the manufacturer’s material requirements planning (MRP) system seem sufficient. Within the MRP system, historical created purchase orders (POs) can be used as a source containing supplier behaviour. Disadvantages of POs alone could be the limited visibility on internal and external influences on the supplier relation and behaviour. Therefore, incorporating additional data or databases containing for example shared information or communication with suppliers between ordering and fulfilment could reduce potential noise on supplier behaviour and improve prediction performance or broaden the scope for supplier disruption prediction.

3 Machine learning in supplier disruption prediction: model development and algorithms

This chapter presents results of the literature review focussing on applied machine learning (ML) algorithms and techniques for supplier disruption prediction in systems and environments as described in the previous chapter. It serves as a basis for choices being made in the methodology development as is described in the next chapter.

The Scopus database was consulted to explore terminology and led to the definition of the following 3 search term groups:

1. “supply chain”, “logistics”;
2. “disruption”, “deficiency”, “lead time”, “resilience”, “backorder”, “stock out”;
3. “machine learning” or “big data”.

Terms from all groups were used in different combinations to find publications. The complete search resulted in 269 results covering multiple research fields due to the disjunctive inclusion of “supply chain” AND logistics. Excluding publications not covering supply chain activities or ML applications, resulting in 64 remaining publications.

Systematically examining those illustrated the main focus of ML applications on prediction (improvements) on the demand side of production and supply chains, including incorporating different data sources or influences as weather. Non-ML applications often consider time-series or simulation for risk prediction or evaluation, as identified by M. He, Ji, Wang, Ren, and Lougee (2015) and Fagundes, Teles, Vieira de Melo, and Freires (2020). However, this research focuses on the potential of ML in material-oriented supplier disruptions, leading to a limited selection of four publications in the last three years. These four are used as basis for this chapter and are presented in Table 3.1 below. The publication dates and the small number of publications illustrates the novelty of this ML-based (applicational) research field.

Table 3.1: Selection of publications focusing on ML applications in material-oriented supplier disruptions.

Author + Reference	Title
De Santis et al. (2018)	Predicting material backorders in inventory management using machine learning
Baryannis et al. (2019)	Predicting supply chain risks using machine learning: The trade-off between performance and interpretability
Brintrup et al. (2020)	Supply chain data analytics for predicting supplier disruptions: a case study in complex asset manufacturing
Hajek and Abedin (2020)	A Profit Function-Maximizing Inventory Backorder Prediction System Using Big Data Analytics

Baryannis et al. (2019) and Brintrup et al. (2020) focus on predicting whether a supplier will deliver on time or not by formulating a binary classification problem and purely use historical POs from a manufacturer’s MRP system. De Santis et al. (2018) and Hajek and Abedin (2020) focus whether a material will go on backorder by formulating a binary classification problem as well. However, they considered the same dataset from a Kaggle’s competition (Kaggle Inc., 2021), which incorporated historical demand forecasts, sales data, inventory levels, material characteristics and whether or not the material went on backorder, which required the loosening of the detail on PO level and enabling the broadening of the scope towards inventory levels.

The remainder of the chapter is structured as presented in Figure 3.1 below, which depicts a generalised methodology structure based on observations in supplier disruption prediction literature, complemented with applied algorithms and techniques.

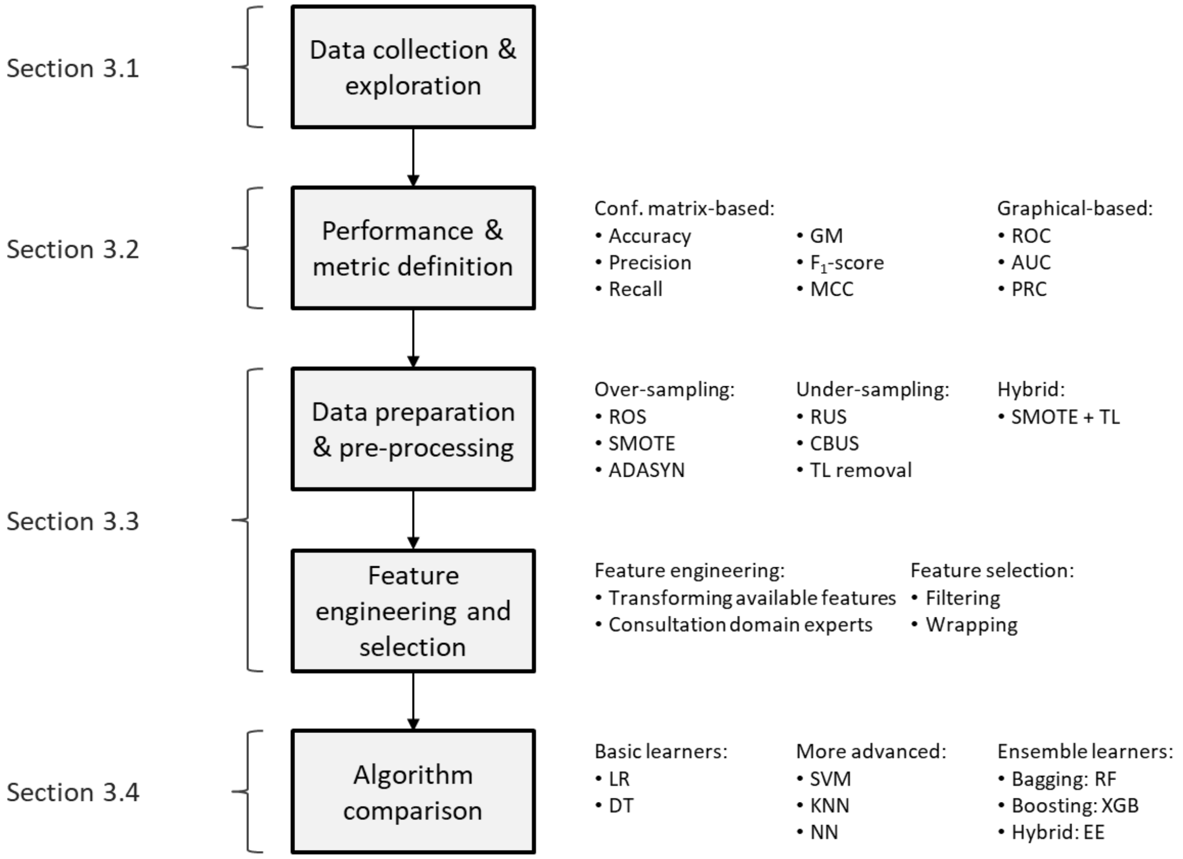

Figure 3.1: Generalised methodology components including observed techniques and algorithms in supplier disruption prediction literature.

3.1 Data collection and exploration

Data collection and exploration is often barely addressed in scientific literature, while one could argue it might be the most important phase of the entire model development. All stages of an ML model are based on the data that is being presented to it and if there is any misunderstanding or incorrect 'usage', the performance could be severely influenced. Within this phase, it is important to understand the system behind the data, therefore knowing how the data is being generated, how it could be influenced, what irregular values are and so on. Using simple visualisations and conducting regular meetings with field experts could assist in acquiring such understanding and assessing the quality of data available.

3.2 Performance and metric definition

Performance and metric definition are influenced by the goal of the model and its target values. It is necessary to define suitable performance metrics beforehand since ML algorithms use these metrics to iteratively evaluate their learning performance. Therefore, unsuitable metrics could push the ML algorithm in the wrong direction (Baryannis et al., 2019).

Commonly used classification metrics are based on elements of a confusion matrix. A confusion matrix is “a two-dimensional matrix indexed in one dimension by the true class of an object and in the other by the class that the classifier assigns.” (Ting, 2010). An example of a confusion matrix for a binary classifier is presented in Table 3.2 below. In the example table, the correctly predicted values are on the main diagonal (top left to bottom right) since the actual class and the predicted class are equal. These cells are called true positive (TP) and true negative (TN) respectively. The other diagonal presents falsely predicted classes, which are like type 1 (false positive; FP) and type 2 (false negative; FN) errors in statistics. In case of binary supplier disruption prediction, positive and negative could correspond to ‘delayed’ and ‘on-time’.

Table 3.2: Example confusion matrix for a binary classification problem.

		Predicted Class	
		P (positive)	N (negative)
Actual class	P (positive)	TP (10)	FN (3)
	N (negative)	FP (2)	TN (11)

Different numerical metrics are defined based on combinations and ratios of TP, TN, FP and FN. The most common and simple formula-based metrics generally observed and applied in literature are presented in Table 3.3 below, in which the main interpretation is presented as well.

Table 3.3: Common numerical (performance) metrics.

Metric	**Definition**	**Interpretation**
Accuracy	$\frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$	Fraction of the correctly classified values over the complete set
Precision Positive Predictive Value (PPV)	$\frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$	Fraction of the correctly classified positive values in the positive class
Negative Predictive Value (NPV)	$\frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FN}}$	Fraction of the correctly classified negative values in the negative class
Specificity True negative rate (TNR)	$\frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}$	Fraction of negative samples correctly classified
Recall Sensitivity True positive rate (TPR)	$\frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$	Fraction of positive samples correctly classified
Geometric mean of true rates (GM)	$\sqrt{\frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} * \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}}$	Geometric mean of sensitivity and specificity
F_{β}-Score	$\frac{(\beta^2 + 1) * \mathrm{PPV} * \mathrm{TPR}}{\beta^2 * \mathrm{PPV} + \mathrm{TPR}}$	Weighted (β) mean of precision and sensitivity. Special case: $\beta = 1$, since it results in the harmonic mean.
Matthews Correlation Coefficient (MCC)	$\frac{\mathrm{TP} * \mathrm{TN} - \mathrm{FP} * \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}}$	Correlation coefficient between prediction and samples

Besides numerical metrics, graphical-based metrics are often used: Area under the Receiving Operator Characteristic-curve (AUC) and Precision-Recall curves (PRC). As the name suggests, the AUC uses a Receiving Operator Characteristic-curve (ROC), which is the line representation of sensitivity as a function of (1 – specificity). The ROC visualises the probability curve of obtaining correct positive predictions (benefits) over false positive predictions (costs). The AUC metric therefore varies between 0 (complete inverse classification) and 1 (completely correct classification), with 0.5 presenting the worst performance, since no specific distinction is observed. The PRC is similar to the ROC-curve. However, PRCs show the relation between the correctness of the predicted values versus the amount of relevant positive predictions returned instead.

In presence of an imbalanced dataset, in which the different classes are not equally present, the definition and choices for metrics become more important. General metrics as accuracy become less relevant and metrics specific for the target prediction value are preferred. Combined metrics, such as the GM and F_1-score might become less suitable as well since comparison of different performances becomes more difficult due to the combined weighted definitions. Metrics as precision and recall are more suitable in the field of supplier disruption prediction since the target value is the positive (minority) class. This is observed in the selected publications, where De Santis et al. (2018), Baryannis et al. (2019) and Brintrup et al. (2020) all used precision- and recall-based metrics. De Santis et al. (2018) explicitly used precision and recall, while Baryannis et al. (2019) opted for implicit use in the F_1 score and MCC, due to the prioritisation of positive or negative predictions. Brintrup et al. (2020) combined explicit use of precision and recall with implicit use in the F_1, $F_{0.5}$ and F_2 scores since they placed the importance of false negative classification over false positive classification.

Noticeable in the work of Hajek and Abedin (2020) is the choice for following previous research and using the AUC over the Precision-Recall curve. Especially in the case of imbalanced datasets, the use of the Precision-Recall curve is more suitable since the minority (positive) class is explicitly accounted for and not weighted with the majority (negative) class. However, they used a different method to define performance and tackle class imbalance by using a cost-sensitive approach and defining a cost for each misclassification (FP and FN). Originally a misclassification cost should be defined for every material in the dataset, but for simplicity misclassification costs were extended to all the different materials. The performance of different models can thereafter also be based on the differences in final cost values.

Recent research conducted by Chicco and Jurman (2020) concluded that in general the use of Matthews Correlation Coefficient (MCC) should be preferred over the use of the F_1-score in imbalanced binary classification problems due to the intuitive- and straightforwardness. A high quality MCC score requires correct predictions in the majority of both classes (positive and negative), independent of the initial class distribution. Together with the Precision-Recall curve, one might therefore conclude that these metrics might be more suitable in general for imbalanced classification problems. However, specific applications or case-based priorities may result in different choices as is observed in the research by Brintrup et al. (2020).

3.3 Data preparation: pre-processing, feature engineering and selection

Using input of domain experts, knowledge about the considered data, the prediction goal and selected metrics, data can be pre-processed and potentially relevant features (individual measurable properties or characteristics of a phenomenon being observed (Bishop, 2006)) can be proposed ('feature engineering'). By eliminating features not significantly contributing to the prediction performance of the developed model, the complexity and data needs decrease while potentially increasing its interpretability and maintaining similar levels of performance.

In sub-section 3.3.1 observed and mentioned data (pre-)processing steps and techniques are presented. Thereafter, feature engineering and selection is addressed in sub-section 3.3.2, in which engineered features and observed selection methods are presented.

3.3.1 Data preparation and pre-processing

Datasets involving disruptions in production systems or healthy supply chains are often imbalanced (Brintrup et al., 2020). This imbalance could reduce the prediction performance which can be mitigated resampling (external), algorithm modification (internal) and cost-sensitive learning (combination). The following paragraphs focus on the resampling techniques as applied in the selected publications. A general advantage of resampling over the other techniques, is the possibility to incorporate it with any desired classifying algorithm afterwards (De Santis et al., 2018). Three categories of sampling will be covered: 'over-sampling', 'under-sampling' and 'hybrid sampling methods'. Table 3.4 presents an overview of the observed resampling techniques, including their stated positive and negative aspects. Algorithm modification and cost-sensitive learning are omitted in this review since they often result in case-specific alterations.

3.3.1.1 Over-sampling algorithms and techniques

Over-sampling categorises techniques creating (synthetic) instances of the minority class. The simplest over-sampling technique is random over-sampling (ROS). It randomly selects an instance from the minority class, which will be duplicated. However, the duplication of existing instances increases the likelihood of overfitting ("describing features that arise from noise or variance in data rather than the underlying distribution" (Webb, 2010)). Therefore, different techniques are often preferred and commonly applied.

Synthetic Minority Oversampling Technique (SMOTE) creates synthetic minority class instances, by (random) interpolating minority class instances that are closely located to each other. De Santis et al. (2018), Baryannis et al. (2019) and Hajek and Abedin (2020) all reported the use of SMOTE in their work. SMOTE uses the k-Nearest Neighbours (kNN) algorithm to find closely related instances, whereafter the set amount of over-sampling determines how many instances are synthetically created (Chawla, Bowyer, Hall, & Kegelmeyer, 2002). By creating random interpolated instances, the problem of overfitting is less present while simultaneously the decision boundaries of the minority class are spread further in the majority class space (Galar, Fernandez, Barrenechea, Bustince, & Herrera, 2012). However, the latter could potentially result in increased overlapping between classes (López, Fernández, García, Palade, & Herrera, 2013) and overgeneralisation (Hajek & Abedin, 2020; Weiss, McCarthy, & Zabar, 2007).

ADAptive SYNthetic (ADASYN) sampling, as also briefly stated and investigated by Baryannis et al. (2019), adds an additional small random component to newly created minority class instances, tending to slightly better represent real-world examples with respect to SMOTE (H. He, Bai, Garcia, & Li, 2008).

3.3.1.2 Under-sampling algorithms and techniques

Under-sampling categorises techniques reducing the number of instances in the majority class, to match the number of instances in the minority class better. The simplest form of under-sampling is random under-sampling (RUS), where random instances from the majority class are removed in the training dataset, leading to potentially discarding valuable information.

Another under-sampling approach is cluster-based under-sampling (CBUS). CBUS applies a clustering algorithm, often k-means clustering due to its linear complexity (Ofek, Rokach, Stern, & Shabtai, 2017), to identify centroids of clusters in the majority class. By selecting the centroid as a replacement instance for the entire cluster of neighbours, the amount of majority class instances will be reduced in the training set. This specific type for CBUS is also referred to as cluster centroid under-sampling, as applied in Baryannis et al. (2019). Another possibility is to select the closest instance with respect to the cluster centroid instead of the centroid itself in the majority class, keeping an actual instance instead of a synthetic one. This specific variant is applied by Hajek and Abedin (2020). They selected the CBUS algorithm to overcome the problem of the trade-off between prediction performance and complexity, based on earlier studies. The advantage of their applied CBUS variant is that actual instances remain, therewith keeping actual information instead of using synthetically created instances. An advantage of CBUS over random under-sampling is the reduced likelihood of discarding valuable data, since the clusters formed in CBUS have similar data characteristics. Therefore, removing instances from such clusters has less impact on the entire dataset's characteristics with respect to random under-sampling (Lin, Tsai, Hu, & Jhang, 2017; Ofek et al., 2017). Disadvantages of the CBUS technique have not explicitly been stated in the covered research, besides the shared notion that it still can be undesired to discard any data instances nonetheless.

Tomek link (TL) removal is also an under-sample technique. A Tomek link is a pair of instances from different classes which are their closest neighbour. In case of under-sampling using TL, the instance from the link which is part of the majority class is removed. In the application of data cleaning (full Tomek link removal), both instances from the link will be removed. By removing the majority instance and keeping the minority instance, the distinction between both classes can become more explicit, increasing the potential performance of different ML classifier models especially focussing on the minority class (López et al., 2013). However, depending on the distribution of classes in the obtained dataset, Tomek link removal can have insignificant impact in the increase of classifier performances. As an example, if minority class instances are heavily enclosed by majority class instances, the removal of some majority class instances will still result in the dominant representation of the majority class over the minority class.

3.3.1.3 Hybrid sampling algorithms and techniques

To overcome the potential insignificant impact of TL removal, it is often combined with other resample techniques, resulting in a 'hybrid' technique. A common combination is the together with SMOTE, as tried by Baryannis et al. (2019). SMOTE is used to create additional samples of the minority class, whereafter full Tomek link removal is used to reduce overfitting potential and enhance distinction between class clusters (reduce class overlapping) (Batista, Prati, & Monard, 2004; Devi, Biswas, & Purkayastha, 2019).

Table 3.4: Resampling techniques applied in supplier disruption prediction literature, complemented with positive and negative characteristics.

Type	Technique	Positive	Negative
Over-sampling	Random over-sampling	- Simple technique	- Increase likelihood of overfitting
	Synthetic Minority Oversampling TEchnique (SMOTE)	- Reduced likelihood of overfitting with respect to random over-sampling - Spread decision boundaries of minority class further in majority class space	- Potential increase of overlapping between classes
	ADAptive SYNthetic (ADASYN)	- Same positives as SMOTE - Slightly more realistic due to minor additional random component	- Potential increase of overlapping between classes
Under-sampling	Random under-sampling	- Simple technique	- Potential to discard valuable data
	Cluster-based under-sampling (CBUS)	- Less potential to discard valuable data with respect to random under-sampling	- -
	Tomek Link (TL) under-sampling	- Enhances distinction of decision boundaries between classes - Suitable to remove noise in the dataset	- Performance improvement can be negligible if minority instances are heavily enclosed by majority instances
Hybrids	SMOTE + Tomek	- Reduce potential of overfitting with respect to SMOTE alone - Enhances distinction between class clusters	- -

3.3.2 Feature engineering and selection

Feature engineering is the process of enriching available data by combining existing features or data characteristics in a suitable way given the applicational field and/or by complementing the data using different data sources. It is highly dependent on the dataset and the prediction goal and requires domain knowledge to effectively combine existing features (Baryannis et al., 2019). Therefore, it is difficult to identify or define a common approach for feature engineering.

Baryannis et al. (2019) applied feature engineering by mainly focussing on splitting dates into different categories as day, week, season and taking differences between dates. Brintrup et al. (2020) focussed more on applicational specific features after consultation with domain experts resulting in features as 'average number of orders', 'average monthly book size' and their own defined 'agility' feature, covering "the capability of suppliers to handle the highest monthly order variations as a proxy of supplier's flexibility". Their definition of 'agility' is according to equation (1) below, in which A_s is the agility score, p the set of products offered by the supplier, t the index for orders and w the order date (Brintrup et al., 2020). The final applied agility score is normalised by the maximum agility score of all suppliers A_{max}, as given in equation (2). Brintrup et al. (2020) classify the agility-feature as a 'dynamic feature', since the agility score of a supplier might change over time as its ability to handle order variations might change.

Within the research from Baryannis et al. (2019), feature engineering did not significantly improved prediction performance, while the agility feature from Brintrup et al. (2020) contributed to a performance increase.

$$A_s \cong \begin{cases} \sum_p \sum_t \left| w_{p,t+1} - w_{p,t} \right|, & \textit{if order is not delayed} \\ 0, & \textit{otherwise} \end{cases} \tag{1}$$

$$A'_s = \frac{A_s}{A_{max}} \tag{2}$$

According to H. Liu (2010), “Feature selection is the study of algorithms for reducing dimensionality of data to improve ML performance.” By applying feature selection techniques, the performance of the ML algorithm is ‘optimised’ and the outcome can become better interpretable (Kotu & Deshpande, 2015). Feature selection can also positively contribute towards tackling the curse of dimensionality and overfitting (Baryannis et al., 2019; Brintrup et al., 2020). The number of features (and therewith dimensions) of a dataset can be reduced in roughly two ways, by filtering (feature ranking) or by wrapping (subset selection).

Filtering can be partially manually done based on domain knowledge by omitting redundant features (such as order ID’s). Additionally, features can be filtered by applying techniques to rank relevance of features under consideration. In case of numerical data, Principal Component Analysis (PCA) can be applied. Chi-squared-based filtering can be applied on categorical data (Kotu & Deshpande, 2015).

Baryannis et al. (2019) stated the use of various feature selection techniques by feature ranking based on the chi-squared test and the ANOVA F-value, which are all methods easily accessible and implemented in the scikit-learn library (Pedregosa et al., 2011). For additional information regarding PCA and chi-squared-based filtering, the author refers to Kotu and Deshpande (2015). Brintrup et al. (2020) mentioned the use of feature selection but did not explicitly stated which techniques were applied. Similar to feature engineering, De Santis et al. (2018) and Hajek and Abedin (2020) did not mention feature selection, potentially due to their usage of a dataset from Kaggle’s competition “Can you Predict Product Backorders?”. Therefore, there is no presence of a manufacturing party with specific domain knowledge, indicating that domain knowledge is indeed required for meaningful feature engineering and selection.

Wrapping or subset selection is the second class of feature selection algorithms. Wrapping requires a learning algorithm, which iteratively selects or keeps attributes positively contributing the most towards the performance of the learning algorithm, expressed by the metrics selected before. Different learning algorithms could respond differently on the in- or exclusion of features and result in (slightly) different selections.

Examples of wrapping techniques are recursive feature elimination and feature selection using ensembles (e.g., Extra-Trees algorithm), which are both used by Baryannis et al. (2019). However, they did not state which classifier or specific parameters they used for the recursive feature elimination and other applied techniques.
Recursive feature elimination considers all features at first, whereafter each iteration the least contributing feature is removed, until the required number of features is obtained, or no significant improvement is achieved.
Forward feature selection is the opposite approach, where the most contributing single feature is used to start the search, whereafter each iteration the most contributing feature is added to the base set of features until the set maximum number of features is obtained or all features are selected.
Exhaustive search considers all possible combinations of features, which are all evaluated and used to select the most suitable configuration. However, this approach is very computational expensive and therefore often not applied.

3.4 Algorithm comparison

After feature engineering and selection, the selection for a suitable ML algorithm needs to take place. In general, it is not easy to select a specific algorithm which will perform best in advance. Therefore, often several ML algorithms are applied (and partially finetuned) to explore several algorithm's performances. Depending on the defined performance metrics and model goal, the best performing algorithm is selected. This indicates the need for research focussing on quicker identification of potential algorithms, such that more time can be allocated to increasing model performance rather than exploring different algorithms.

An overview of algorithms applied in supplier disruption prediction literature including positive and negative attributes is presented in Table 3.5 below. The algorithms are briefly addressed in the following paragraphs.

Table 3.5: Machine learning algorithms applied in supplier disruption prediction, complemented with positive and negative characteristics.

Algorithm	Positive	Negative
Logistic Regression (LR)	- Simple approach - Feature scaling not required	- Generally poor performance
Decision Tree (DT)	- High interpretability - Feature normalisation or scaling not necessary - 'Automatic' feature selection	- Prone for overfitting
Random Forest (RF)	- Can converge to small generalisation error with Bagging - Good performance on imbalanced datasets - Suitable for feature selection	- Sensitive to features with different values - Difficult to interpret
Support Vector Machine (SVM)	- Performs well on high dimensional data - Highly suitable for separated classes in the dataset	- Requires problem transformation to one-vs-one or one-vs-rest classifier in case of multiclass classification - Sensitive to parameter settings - Data must be scaled
k-Nearest Neighbours (kNN)	- Low complexity - Simple for (non-linear) classification - Suitable for multiclass problems	- Difficulty for successful application with (large) unbalanced datasets - Data must be scaled - Sensitive to data or objective changes
Neural Network (NN)	- Strong self-study ability - Strong ability to fit nonlinear relations - Possibility to extend to higher dimensions with additional features	- Sensitive to parameter settings (e.g., network topology) - Difficult to interpret and explain - Data must be scaled
Ensemble algorithms (ESM; e.g., XGBoost and EasyEnsemble)	- Good for assembling advantages of different methods while reducing negative characteristics	- Dependent on the underlying classifier - Can become difficult to interpret - Potentially more parameter to set (and tune) to obtain best performance

De Santis et al. (2018), Brintrup et al. (2020) and Hajek and Abedin (2020) investigated Logistic Regression (LR) as a potential method for predicting delayed supplier deliveries. Due to simplicity, LR is used to create a benchmark regarding model performance. In almost all of their conducted experiments, the different applied algorithms resulted in higher model prediction performance.

Decision Trees (DT) as classifier is applied by Baryannis et al. (2019) and Hajek and Abedin (2020). Due to the high interpretability of DTs, Baryannis et al. (2019) investigated the performance and concluded that the performance of interpretable DTs (maximum depth of 6 with maximum 13 leaf nodes) was similar to the performance of the 'less-interpretable' Support Vector Machine (SVM). The results from Hajek and Abedin (2020) showed a better performance of DTs with respect to their SVM model. De Santis et al. (2018) reported using exhaustive grid search to optimise the parameter settings of their algorithms.

Random Forest (RF) is a form of an ensemble learning algorithm that combines different DTs, trained on different subsets of the available dataset. Due to the combination of the different DTs the performance is expected to increase when the dataset could result in instable DTs. The combination reduces this instability, leading to better results with less variance. However, since the combination of trees is present, the classifier becomes less interpretable with respect to the DT-algorithm. De Santis et al. (2018), Brintrup et al. (2020) and Hajek and Abedin (2020) applied RF and concluded that in their case RF performed well, or even the best with respect to the algorithms considered.

Support Vector Machines (SVM) were applied by Baryannis et al. (2019), Brintrup et al. (2020) and Hajek and Abedin (2020), based on the SVM performance in previous research. SVM tries to identify a hyperplane, which creates a boundary between two classes. Based on this boundary new instances are classified. Due to this behaviour, applying SVM for multiclass classification would require the use of one-vs-one or one-vs-all classification. In one-vs-one classification, the multiclass problem is split in several binary problems where classification between each class pair is considered. In one-vs-all classification, the multiclass problem translates to several binary problems predicting whether the data entry belongs to the considered class or one of the rest. SVM applied in the basic articles performed in general well but was not the most suitable algorithm for the specific cases. Baryannis et al. (2019) reported the use of exhaustive grid search for parameter tuning for the SVM algorithm.

K-Nearest Neighbours (KNN) as classifier works similar with respect to the generally known KNN algorithm, in which new instances are assigned to the highest observed class within the k number of nearest neighbours. KNN is considered as classifier by Brintrup et al. (2020) and Hajek and Abedin (2020). However, the KNN prediction performance was not the best or the worst within all considered algorithms.

A (Multi-Layer) Neural Network (NN) is only considered by Hajek and Abedin (2020), potentially due to the complexity of creating and understanding NN's with respect to the previous algorithms. The NN model created by Hajek and Abedin (2020) performed worst from all considered algorithms. However, nothing has been reported regarding model parameters, indicating that hyperparameter optimisation could improve performance. They concluded that under-sampling in their case worsens the performance of the NN-based model.

Ensemble Learners (ESM) is a specific type of ML algorithms. According to Galar et al. (2012), the objective of ensemble learners is "to try to improve the performance of single classifiers by inducing several classifiers and combining them to obtain a new classifier that outperforms every one of them." Due to this combination of different classifiers, the generalisation ability of the ensemble increases. Within ESM, different classes can be identified: Bagging, Boosting and Hybrids.

Bagging (or bootstrap aggregating) creates bootstrapped replicates of the training set, which are used to train individual classifiers in parallel (Galar et al., 2012). Thereafter, the individual classifiers are combined leading to the ensemble. The main focus of bagging is to reduce the variance of the classifier. A well-known example is the RF algorithm mentioned before. Brintrup et al. (2020) used the bagging principle with all algorithms they considered.

Boosting is a class in which different individual classifiers are not created in parallel, but sequentially. Individual classifiers are iteratively trained, where information is passed on through the iterations. Therewith, boosting mainly focusses on reducing bias in the ensemble classifier (Galar et al., 2012). Like bagging, different specific methods are applied and proposed in literature. De Santis et al. (2018) applied Gradient Boosting in their research and concluded that ensemble learners improved prediction performance. A similar conclusion is stated by Hajek and Abedin (2020), who used eXtreme Gradient Boosting (XGB), which is often applied in practical applications.

Hybrid ensembles combine bagging as well as boosting techniques. An example applied by De Santis et al. (2018) and Hajek and Abedin (2020) is EasyEnsemble (EE). Both stated that this method performed well or even better than most of their investigated algorithms. Additional information regarding the EasyEnsemble algorithm can be found in the research from X. Y. Liu, Wu, and Zhou (2009).

3.5 Conclusion literature review

When focussing on applicational studies regarding ML-based supplier disruption prediction, only recently a few studies have been conducted. More researched areas focus on applying ML in the demand side of the supply or production chains or use older techniques as time-series analysis and simulation. Within the studies focussing on supplier disruption prediction, the main problem formulations are simplified to binary classification problems and addressed using similar methodologies. These methodologies are generalised in this chapter to five components: *"Data collection and exploration"*, *"Performance and metric definition"*, *"Data preparation and pre-processing"*, *"Feature engineering and selection"* and *"Algorithm comparison"*. While each component has a specific goal and contribution, different choices can be made and different algorithms and techniques can be applied, as depicted in Figure 3.1. These possibilities have been described in this chapter and summarised including positive and negative characteristics in the tables 3.3, 3.4 and 3.5 regarding (performance) metrics, sampling and machine learning algorithms respectively.

Additionally, it was found that the presence of a (manufacturing) party with specific domain knowledge is necessary for meaningful feature engineering and selection. Additional research is needed focussing on the quicker identification of potential algorithms such that more time can be allocated to increasing model performance rather than exploring different algorithms.

The generalisation of the methodologies, the positive and negative aspects and the characteristics derived in chapter 2 will be used for the development of the methodology for the application in production environments as will be defined in the next chapter.

4 Model development approach for supplier disruption prediction

This chapter describes the novel methodology proposed in this research to initiate and explore the potential of ML in material-oriented supplier disruptions in production environments. An overview of the steps in the methodology including different algorithms and techniques are depicted in Figure 4.1 below. The methodology is suitable for the unexplored extension towards multiclass classification and includes the possibility to incorporate newly developed algorithms or techniques.

In the proposed methodology the possibility to consider and compare individual suppliers and custom supplier groups is incorporated, which has, to the best of the author's knowledge, not been applied in supplier disruption prediction literature before. Expected is that a better understanding and practical use result from including supplier groups, since the complexity is initially reduced and extracted supplier specific behaviour could be verified easier and accepted by buyers. This could potentially lead to a higher acceptance and adoption of ML techniques in production environments.
Additionally, the application of threshold tuning combined with non-cost sensitive learning algorithms in supplier disruption prediction has not been observed before, which is expected to be valuable, since it incorporates another possibility to focus model performance on case specific targets without structurally altering trained models.

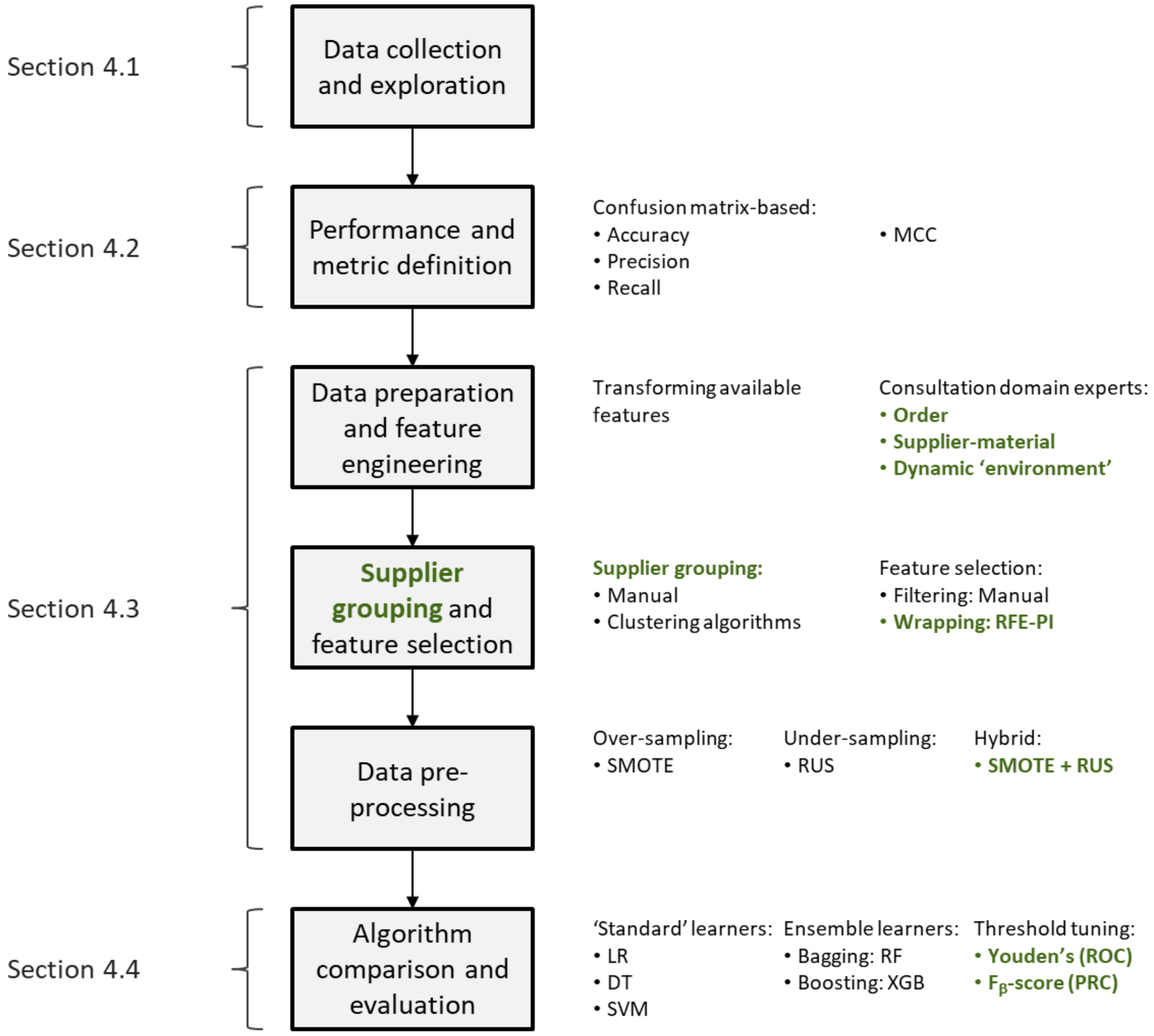

Figure 4.1: Overview of steps in proposed methodology including techniques and algorithms. Indicated in green: existing techniques newly applied in supplier disruption prediction and additions to different steps with respect to literature, besides the incorporation of multiclass classification and the formulation of the generalised methodology.

4.1 Data collection and exploration

The goal of this step is to obtain a comprehensive understanding of the considered production system. In different production systems similar information regarding material management is being used. However, specific aspects of operations, data creation, data storage, data influences et cetera could change approaches or usage of the data in the subsequent steps in the methodology. Therefore, it is important to keep in close contact with domain experts and practitioners in the considered system to acquire the necessary insights and understanding.

These insights can be used to determine relevant data aspects with respect to the specific prediction goal. Given the ease of access or collection of these desired data elements, a data collection project could be initiated before following the rest of the methodology. Otherwise, the methodology can be followed using the already available data (provided that historical POs are included). However, for the latter it is important to have a clear understanding of the resulting limitations and underlying assumptions.

4.2 Performance and metric definition

Supplier disruption prediction literature using ML focusses on binary classification problem definitions. A new addition in the proposed methodology is the exploration of the possibilities and value of multiclass classification in supplier disruption prediction. Multiclass classification allows for more specific mitigation or risk estimations. However, the increase in number of classes requires better distinction between classes in the dataset to prevent negative effects of class overlapping. To evaluate multiclass classifiers, multiclass performance metrics need to be used.

4.2.1 Multiclass metrics

In section 3.2 the binary 2 x 2 confusion matrix is introduced for which multiple metrics could be defined. In case of multiclass classification with n classes, the matrix expands to a n x n matrix leading to the inability to use former metric definitions based on the four different outcomes (true positive, false positive, true negative and false negative). To overcome this for multiclass classification, the metrics can be class-wise defined by expressing them in a one-vs-rest fashion. Table 4.1 shows an example confusion matrix for a three-class classifier to simply illustrate this, but the same can be applied with more classes. When computing the recall value for the 'Early' class in the example – the fraction of correctly predicted 'Early' samples over all actual 'Early' deliveries – it is calculated as $\frac{3}{3+2+1} = 0.5\ (50\%)$. Similarly, the precision of the 'Early' class – fraction of correctly predicted 'Early' over all predicted as 'Early' – is then $\frac{3}{3+1+0} = 0.75\ (75\%)$. The same calculations can be repeated for each class. However, when comparing different algorithms, the increase in number of values to compare limits the transparency and suitability. Therefore, precision and recall values can be weighted to obtain one single value.

Table 4.1: Example confusion matrix for a multiclass classification problem (number of classes = 3).

		Predicted Class		
		Early	On Time	Delayed
Actual class	Early	3	2	1
	On Time	1	7	2
	Delayed	0	3	4

Three main strategies are micro (μ), macro (M) and weighted (W) averaging. However, micro averaging, averaging over all predictions over all classes, and weighted averaging, averaging weighted by the number of samples per class, are less suitable for imbalanced problems since more importance is assigned to higher frequency classes. Macro averaging assigns equal weights to all classes, making it more suitable with respect to micro or weighted averaging for imbalanced problems (Branco, Torgo, & Ribeiro, 2016). Macro-Recall and Macro-Precision can be calculated using equations (3) and (4), in which C represents the set of all classes present.

$$\mathrm{Rec_M} = \frac{\sum_{c \in C} \mathrm{recall(c)}}{|C|} \tag{3}$$

$$\mathrm{Prec_M} = \frac{\sum_{c \in C} \mathrm{precision(c)}}{|C|} \tag{4}$$

Matthews Correlation Coefficient is originally defined for the binary case. When increasing the number of classes, the original definition does not suffice anymore. Gorodkin (2004) adapted the MCC to a more general R_K coefficient for K classes. By generalising the definition, Gorodkin (2004) defined the R_K coefficient in the following way:

$$\mathrm{R_K} = \frac{\sum_{klm} C_{kk}C_{lm} - C_{kl}C_{mk}}{\sqrt{\sum_k (\sum_l C_{kl}) \left(\sum_{\substack{l' \\ k' \neq k}} C_{k'l'} \right)} \sqrt{\sum_k (\sum_l C_{lk}) \left(\sum_{\substack{l' \\ k' \neq k}} C_{l'k'} \right)}} \tag{5}$$

In equation (5), the K stands for the number of classes and the indices for C represent the value in the K-dimensional confusion matrix. Due to its definition, the minimum value of R_K varies between -1 and 0, depending on the underlying distributions. Like the binary definition, zero represents no correlation and random performance, negative values represent negative correlation, and the maximum value will remain 1, indicating the best performance (over all classes).

4.2.2 Metric selection

Given the focus of supplier disruption prediction and the inherently linked imbalance problem, the usage of accuracy is not sufficient to express the performance of an ML model in one value. Therefore, the usage of the MCC as main performance metric is proposed. MCC has a better ability with respect to accuracy to express model performance in a single value while accounting for imbalance in the binary and multiclass case. A high score requires correct predictions in all classes independent of the initial class distribution. Having a single metric for measuring (main) performance eases selection and comparison of different algorithms and configurations in the following stages. In addition, an expression of MCC for the multiclass case is present, leading to a more uniform performance expression throughout the different problem formulations.

To support the MCC expression and be able to show prediction performance per class, precision and recall as assisting measures are proposed. Precision since it represents the fraction of correct predictions *within a predicted class* and recall since it represents the fraction of correct *predictions of an actual class*. The definition of precision and recall allows to represent the prediction performance on different classes individually. The class-wise representation can be applied in the multiclass case as well, which can be macro averaged when different algorithm configurations are compared.
Accuracy can be considered as final proposed supportive numerical metric since it is able to put precision and recall into perspective with respect to other classes and the imbalance ratios.

Looking forward to algorithm comparison, graphical metrics as ROC and PRC are useful to fine-tune the final algorithm configuration by threshold tuning, which will be elaborated in section 4.4. Therefore, ROC and PRC are considered as metrics for threshold tuning in the proposed methodology.

4.3 Data preparation: feature engineering, supplier grouping, feature selection and pre-processing

With the initial understanding of the system and the available data as obtained in the first step "Data collection and exploration", the available data can be cleaned by omitting inconsistent, incomplete or noise data entries, filling in missing data entries and standardising units. In case missing data entries cannot be obtained via different systems or consultation with practitioners, missing values can be imputed based on for example the mean or median value. However, one should split the dataset first before applying such transformations.

In ML model development often three datasets are considered: a train, validation, and test set. The train set is used to train an initial model, of which components could be fine-tuned using the validation set. Thereafter, the test set is used to evaluate the (generalised) model performance since it has not been used during the model development.
These sets can be individually obtained or created by partitioning the available dataset. However, a validation set is not always necessary or available, especially if the available data cannot be partitioned in three sets with similar characteristics.
Therefore, following common literature and practice, a 0.8-0.2 train-test partition is suggested.

4.3.1 Feature engineering

To enhance model training, the available dataset can be complemented with potentially more relevant viewpoints. Simple feature engineering transformations are representing existing features into ML-understandable formats, such as splitting dates in numerical partitions as "*day*", "*week*" and "*month*" or by combining and transforming features by taking for example the ratio between the time given for a supplier to deliver and the contracted lead time.

However, these suggestions and simple transformations do not necessarily include or represent domain knowledge and practitioners' expertise, which is needed for valuable features (Baryannis et al., 2019). Domain experts and practitioners are consulted to focus on potential valuable features for material-oriented supplier disruptions in production environments. The discussions and obtained suggestions led to the definition of three different feature domains:

- "Order" represents features which are characteristic for the historical order (PO record). Examples are the simple transformations from dates to "day", "week" and "month" as stated before, differences between available dates such as difference between order and due date, or order value based on the price and quantity of ordered items.
- "Supplier-material" focusses on features based on the supplier-material relation. Examples are the ratio between the time given for a supplier to deliver and the contracted lead time as stated before, and the ratio between ordered quantity and the standard or average quantity.
- "Dynamic 'environment'" is a broader domain, which covers features with dynamic behaviour. Preceding supplier delivery performance, outstanding number of orders and outstanding number of overdue orders at the moment of order creation are all features which change over time and could be derived from the minimum expected requirement of historical PO records.

An overview of suggested features within the domains after consultation with domain experts is presented in Table 4.2 below. The overview incorporates features based on the expected minimum required data as stated in section 2.3, and potential extensions (indicated with – instead of ●) when more information is available. These potential features are adopted in the proposed methodology and serve as guidance for feature engineering.

Table 4.2: Overview of proposed (engineered) features per defined feature domain.

Feature domain	**Features**
Order	• Creation/Due Day • Creation/Due Day of Week • Creation/Due Week • Creation/Due Month • Creation/Due Season • Creation/Due Year • Days between Creation and Due date • Material • Quantity • Supplier • Value - Days between confirmed delivery date and due date - Order changed indicator - Order involved execution of mitigating measure
Supplier-material	• Price per material • Contracted lead time • Ratio of quantity over standard quantity • Ratio of material order frequency over standard frequency • Ratio of given time for fulfilment and contracted lead time • Size of product portfolio for corresponding supplier • Considered Safety time for the supplier-material combination • Unique number of materials produced/ordered at a supplier - Default shipment method - Possible alternative (express/priority) shipping methods
Dynamic 'environment'	• Previous (confirmed) order delivery performance • Open or outstanding (confirmed) quantity (per material) • Open or outstanding (confirmed) overdue quantity (per material) - Ratio of current requested/outstanding quantity over maximum allocated production quantity/capacity in a time period - Ratio of current requested quantity over shared forecasted quantity - Performance/number of deviations of supplier confirmed orders - Inventory level at moment of ordering

4.3.2 Supplier grouping

Supplier characteristics could vary significantly between different suppliers which could influence the possibility to train a ML model. Therefore, especially when exploring possibilities of ML, it is more suitable to consider single suppliers at first, whereafter suppliers could be grouped to increase the complexity, supplier coverage and more general supplier behaviour. To the best of the author's knowledge, this build-up has not been applied and documented in literature before.

Different methods could be applied to group different suppliers, of which manual grouping would be deemed simplest. Manual grouping could be performed based on information in the available dataset such as number of orders and lead time of materials, or by incorporating practitioners' expertise and already available supplier groups. Potentially interesting could be to group suppliers based on a risk measure regarding production impact, as has been acknowledged by consulted practitioners. One could think about combinations of lead time, material value, shipping possibilities, raw material requirements and uniqueness of products for such measure. However, fully defining such measure is out of scope and therefore left for future research.

Alternatively, one could apply more advanced grouping or clustering techniques and algorithms, such as ML-based methods as hierarchical clustering or k-nearest neighbours (KNN) clustering. Based on the available data and selected characteristics, similarity is expressed in distance between data entries and used to group the different suppliers. These groups could then be used as subsets for the next steps in the proposed methodology.

The usage of manual grouping while incorporating practitioners experience is initially proposed in the methodology, since it is more transparent, easier to adjust to the specific production characteristics and therewith easier to explore possibilities with. In later exploration stages or iterations of the methodology, the application of clustering algorithms could be investigated using (to-be-) defined (risk) measures as mentioned before.

4.3.3 Feature selection

Two directions for feature selection, filtering and wrapping, have been observed in literature and described in section 3.3.2. When applying filtering, the resulting feature ranking can be biased if different features describe similar information and relations, which negatively influences the selection process. Manual filtering combined with simple correlation analysis could reduce these influences, especially when combined with a wrapping technique as recursive feature elimination (RFE) or forward feature selection (FFS).

Standard RFE or FFS use an estimator with importance attributes such as feature weights or importance (in tree-based methods) which is trained on the train dataset. Based on the importance value of a feature, it is kept, added to, or removed from the feature subset. However, complications may arise when categorical variables are present in the dataset. For most estimators to handle categorical variables, the variables must be encoded in numerical representations, potentially imposing relations between different category values. Therefore, in contrast to preceding literature, the usage of feature permutation importance is proposed.

Feature permutation importance uses the difference in model prediction performance after shuffling feature values as an expression for feature importance (Breiman, 2001). By shuffling the values of a single feature, the potential link between the feature and the prediction target is broken and changes in model prediction performance (expressed in the main metric as selected in 4.2.2) could be assigned to the absence of this link. This expression of importance can be used to create a different formulation of the standard RFE or FFS techniques. Desirably a separate validation dataset or partition is used for the feature selection since it allows for a more generalised result. However, if such dataset or partition is not available, the training set could be used instead.

RFE using feature permutation importance is preferred over FFS. FFS is in general significantly more computationally expensive since it needs to train models for each new feature combination in each iteration with respect to a single combination in each RFE iteration.

4.3.4 Data pre-processing: scaling and sampling

Depending on the algorithms considered for model training and comparison, scaling of features might be necessary to improve performance. Different scaling methods as min-max scaling, max-abs-scaling or robust scaling could be feature-wise applied. Important is to apply scaling before applying any form of sampling, since the creation or removal of data points influences the value distribution within features, which could influence the results of scaling operations. Since scaling is feature and algorithm dependent, the possibility for scaling is incorporated in the methodology instead of specifying a specific scaling technique, since specific feature characteristics determine suitable scaling techniques.

As stated in literature, class imbalance is a common problem in disruption prediction in general (Baryannis et al., 2019; Brintrup et al., 2020), due to the generally small number of disruptions occurring during healthy operations. Given the different possibilities to mitigate the negative influences of class imbalance on model performance as stated in section 3.3.1, the resample strategy is incorporated in the methodology based on its modularity, easiness to apply and results in preceding literature. Three resample strategies besides not resampling are proposed, which could be run parallel and compared in the algorithm comparison step.

The first strategy would be to apply over-sampling on the minority class by means of the SMOTENC algorithms developed by Chawla et al. (2002). SMOTENC is a small alteration of the SMOTE algorithm which is often applied in practice and literature (López et al., 2013) such that categorical variables can be over-sampled as well. To reduce overgeneralisation an over-sampling factor of two is suggested as initial limit. An over-sampling factor of two corresponds to doubling the size of the minority class. Potentially different factors might result in more desired trade-offs, which could be investigated in future research. However, expected is that such factor is highly case or environment dependent.

The second proposed strategy applies under-sampling on the majority class by means of random under-sampling (RUS). RUS is simple to apply and has successfully been applied in preceding literature, despite the possibility that valuable information is omitted (Brintrup et al., 2020). However, to prevent potentially omitting too much information, a maximum under-sampling factor of three is suggested as limit. This is slightly higher than the suggested limit for over-sampling since no synthetic information is added. Similar to the over-sampling factor, it is an initial suggestion of which its impact can be investigated in future research.

The third proposed strategy combines the preceding strategies sequentially, such that negative effects of both strategies could be reduced, and larger imbalance ratios could be addressed. In the multiclass application, proposed is to sample individual classes towards the mean number of samples over the different classes.

4.4 Algorithm comparison and evaluation

With the training set being prepared for model training, ML algorithms and parameters need to be selected. Since there is no commonly accepted best algorithm, the following five algorithms are suggested:

1. Logistic Regression (LR)
2. Decision Tree (DT)
3. Support Vector Machine (SVM)
4. Random Forest (RF)
5. eXtreme Gradient Boosting (XGB)

For each algorithm an indicative grid search is applied to explore obtainable performance rather than optimising all parameters for each considered algorithm. This is proposed to limit the size of the grids (and therewith computational cost) and the required specific knowledge for each algorithm, making it more reproducible and understandable in production environments. The algorithm configurations present in the grid will be evaluated using stratified 5-fold cross-validation to additionally reduce negative influences of class imbalance and randomness during the training process.

An overview of the parameters and considered values in the indicative grids suggested is given in Table 4.3 below. Additional information regarding the parameters is presented in Appendix A: Considered algorithm parameters. The values and ranges are based on considered ranges in literature (Baryannis et al., 2019; De Santis et al., 2018) and initial explorations. After the grid search is conducted, the different algorithms and configurations can be compared based on the metrics defined before while applying the trained models on the independent test set.

Table 4.3: Overview of suggested parameter values for the indicative grid search per considered algorithm.

Algorithm	Parameters	Range
Logistic Regression	- Regularisation – C - Class weight - Solver	- 0.1, 0.5, 1, 5, 10, 50, 100, 500, 1000 - None, balanced - Newton-cg, lbfgs
Decision Tree	- Max tree depth - Max leaf nodes - Class weight	- 5, 6, 7, 8, 9, 10 - 20, 30, 40, 50 - None, balanced
Random Forest	- Max tree depth - Max leaf nodes - Nr. of estimators - Class weight	- 5, 6, 7, 8, 9, 10 - 20, 30, 40, 50 - 10, 50, 100, 500 - None, balanced_subsample
Support Vector Machine	- Regularisation – C - Gamma – γ - Class weight	- 0.1, 1, 10, 100, 1000, 10000 - 0.01, 0.1, 1, 10, 100, 1000 - None, balanced
eXtreme Gradient Boost	- Max tree depth - Min child weight - Gamma – γ - Nr. of estimators	- 5, 6, 7, 8, 9, 10 - 1, 2, 3 - 0, 1, 10, 100 - 10, 50, 100, 500

After models have been trained, additional post-processing of model results could be applied to steer predictions towards the most interesting or valuable prediction value. A common method in binary classification, which has only been observed in a similar form in the cost-sensitive approach of Hajek and Abedin (2020), is threshold tuning. Threshold tuning involves the adjustment of the threshold value used to determine to which class a prediction belongs. By adjusting the threshold value, a custom trade-off between class-specific prediction performances can be obtained. Two common trade-offs in practice are (1) between the true positive rate and true negative rate, as visualised in the ROC and (2) between precision and recall as visualised in the PRC.

As general optimisation indicator for the first case the Youden's J statistic (J) is used, which translates to the probability of making an informed decision (Powers, 2020) and can be expressed as $J = TPR - FPR$ (Youden, 1950), or visually as the distance between the ROC and the ROC of a random guesser (diagonal from bottom left to top right). Selecting the threshold value corresponding to the largest J-value flattens the difference in prediction performance over all classes, which could improve the performance on the minority class at the expense of majority class prediction performance.

In the second case, often the F_β-Score is used since it describes a relation between precision and recall. By selecting a β corresponding to the case-specific focus, the trade-off between precision and recall can be expressed and the threshold value corresponding to the largest F_β-Score can be selected.

The formulation of threshold tuning for binary classification problems cannot be generalised to multiclass classification. In fact, it would require a reformulation of the multiclass problem into pair-wise binary classification problems, for which simultaneously all thresholds could be tuned by for example Particle Swarm Optimisation, as proposed by Cheng, Chen, Khosla, and Kim (2011).

Once threshold tuning is applied, the test set is used to evaluate and compare the final model performance and its generalisability, whereafter is decided if the model can be applied in practice.

4.5 Concluding statements

This chapter described the proposed methodology to apply ML for supplier disruption prediction in a general production environment, of which the steps and outline are depicted in Figure 4.1. The important characteristics per defined step are summarised in the following sections.

Data exploration and understanding

To acquire a sufficient understanding of the production system, its data generation (influences), limitations and prediction goal, it is important to keep in close contact with practitioners and experts. Suggestions for additional data collection can follow and be formulated and proposed as a result.

Performance and metric definition

MCC as main metric has been proposed since it is able to express performance over all classes in one value while accounting for imbalance in both problem formulations (binary and multiclass). Accuracy and class-wise precision and recall are proposed as supportive metrics with the possibility of macro-averaging when aggregating to one precision or recall value. Recall is extra valuable in disruption prediction since it represents the fraction of correctly predicted instances in a class and therewith correctly predicted disruptions. ROC and PRC are only suggested as supportive graphical methods when applying threshold tuning as post-processing technique for binary classification, since it could assist in visualising the difference in performance.

Feature engineering and selection

For feature engineering three feature domains ("Order", "Supplier-material" and "Dynamic 'environment'") are defined and proposed, with the notion of a potential fourth ("Quality") if sufficient quality data is available. The different suggestions for features are presented in Table 4.2. For feature selection, the use of feature elimination using feature permutation importance is proposed, which is different with respect to applied feature selection in preceding supplier disruption literature. Feature permutation is chosen since it allows for considering categorical data in the selection process while presenting a ranking of features for each number of features considered. Feature elimination is preferred over forward feature selection due to computational expensiveness.

Supplier grouping

The usage of manual grouping while incorporating practitioners experience is initially proposed since it is an easy to implement and understandable approach for the newly introduced step in supplier disruption prediction. Additionally, it is more transparent, easier to adjust to the specific production characteristics and therewith easier to explore possibilities with. In later (exploration) stages or iterations of the methodology, the application of clustering algorithms could be investigated using (to-be-) defined (risk) measures or feature selection results for individual suppliers.

Sampling

Three different resample techniques are proposed to potentially reduce the impact of class imbalance: SMOTENC, RUS and a hybrid of both. SMOTENC is an addition towards current applied techniques, enabling the incorporation of categorical features without the need to transform them. For SMOTENC a maximum over-sample rate of 2 is suggested to prevent significant reductions in generalisation ability and increased training time. A maximum under-sample rate of 3 is suggested to prevent potentially removing too much valuable information.

Algorithm comparison

Five different algorithms are suggested to explore and evaluate performance given their complexity, illustrated performance and fundamental definitions: LR, DT, RF, SVM and XGB. For each algorithm an initial explorative grid is suggested to observe what level of performance can be obtained by adjusting the most common parameters as depicted in Table 4.3. After the grid search, the best performing classifiers in the binary formulation can be fine-tuned by changing the classification threshold to the value where the Youden's statistic or F_β-Score is the highest, depending on the case specific focus.

5 Supplier disruption prediction in Philips' production facility

This chapter describes the application of the methodology proposed in the previous chapter in Philips' production facility. The application within this case study can show initial value and potential of ML in predicting material-related supplier disruptions in production environments using the proposed methodology.

In section 5.1, a brief description of relevant case specific information is given. Section 5.2 presents how feature engineering and supplier grouping is applied, whereafter the results of the feature selection stage are presented in section 5.3. Section 5.4 presents the results of the grid searches and post-processing, whereafter section 5.5 reflects on the results and the applied approach.

5.1 Case study: Philips' MR & IGT production facility

Within the production facility several of Philips' medical imaging products as Magnetic Resonance Imaging (MRI) scanners, X-ray scanners and computed tomography (CT) scanners are being assembled and produced. The portfolio of the facility consists of 36 different products of which more than 3000 combined are produced and shipped globally on a yearly basis. Within each product up to 7500 unique components are being used, illustrating the importance of correct and efficient material management.

5.1.1 Data availability and limitations

Different systems are being used to monitor and store data and information for the different actors present in the production chain. The main system for the material management aspect is the Enterprise Resource Planning (ERP) system. Within the ERP system, information regarding material demand and flows is maintained while also being used to monitor and create POs. Data regarding historical POs has been made available for this case study, therewith fulfilling the expected minimum data requirement as stated in section 2.3 to predict delivery performance of created orders.

In addition, to maintain and improve supplier collaboration, supplier performance scores (CLIP) are being calculated based on among others delivery performance of placed and fulfilled POs. These scores are being stored in Philips' Global Supplier Rating System (GSRS) database, which has been made available for this case study. Due to definition changes in the considered time scope (April 1st, 2017 until April 1st, 2020), the latest logic behind these scores is used to recreate historical performance scores in which individual deliveries are considered and the percentage of on-time deliveries is translated to the monthly CLIP score.

Unfortunately, historical demand forecasts, inventory levels, executed mitigation measures and communication were not available for this case study. Therefore, only historical PO records on schedule line level are extracted using a third-party tool Every Angle and GSRS performance scores will be used, imposing the following implications and limitations:

- Influences or disruptions outside Philips' view impacting supplier performance, such as transportation disruptions or production problems at the supplier, are not made available in the ERP system export and therewith not explicitly represented in the available data.
- PO changes are not extracted using Every Angle, only initial PO characteristics.
- Known potential influences as shared demand forecasts, manual communication or disruption mitigation measures are not available and cannot explicitly be accounted for.

Figure 5.1 illustrates these implications and limitations by visualising an example timeline between PO creation and obtaining the PO receipt(s). Within this timeline, several events which influence supplier behaviour (delivery performance) which cannot be explicitly extracted from or represented in the available data are indicated in grey. Events that are explicitly available in the data are indicated in black. Since the result (PO Receipt date) is available in the data, these events and influences are implicitly incorporated and accounted for but could potentially result in noise and reduced prediction performance. An example export of the available data is presented in Appendix B: Example dataset.

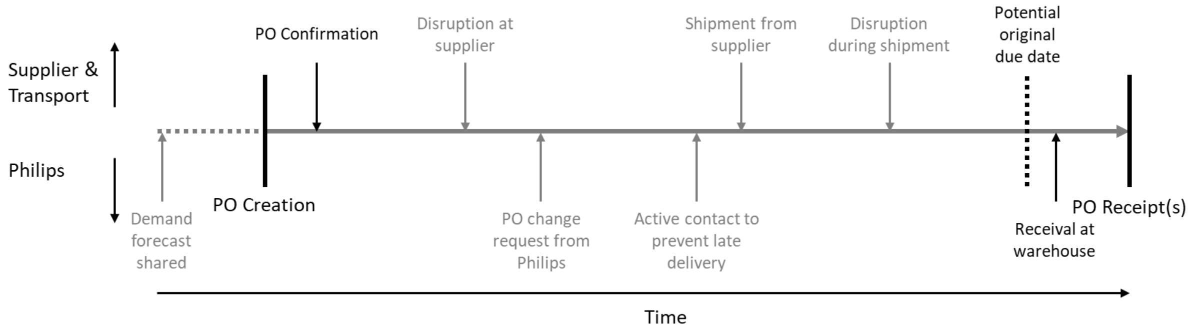

Figure 5.1: Example timeline of PO creation till fulfilment (receipt) including events that cannot be explicitly extracted from or are not explicitly represented in the available data set.

5.1.2 Case specific scope

The prediction target for this case study will be the expected delivery performance of an inbound delivery. Using Philips' definitions for delivery performance, the target classes for the binary and multiclass classification problem formulations are defined and presented in Table 5.1 below.

Table 5.1: Target classes for binary and multiclass classification.

Problem formulation	Target classes
Binary	- On-time: before or on due date - Delayed: after due date
Multiclass	- Extremely early: more than 3 days before due date - Early: between 1 and 3 days before due date - On-time: on the due date - Delayed: 1 or 2 days after due date - Extremely delayed: 3 or more days after due date

As relevant metrics for the Philips' case, the suggested metrics in section 4.2.2 are selected since the focus is predicting delayed deliveries, while limiting the number of false positives to reduce unnecessary mitigations. Therefore, the main performance metric is MCC, supported by precision, recall and accuracy.

5.1.3 Dataset characteristics and preparation

Historical order PO data between April 1st, 2017 and April 1st, 2020 from the ERP system is extracted using Every Angle. Every Angle combines different data locations in the ERP system to create a single dataset of historical deliveries which is used for this case study. An overview of features contained in the dataset is presented in Table 5.2.

Table 5.2: Overview of extracted data.

Feature	Format	Description
PO document type	Text	Category indicating type of PO
PO Schedule Line number	Text	Purchase order number + PO item number + SL number
Order date	Date	Date when the order is created
Due date	Date	Date when the order is requested to be delivered
Confirmed delivery date	Date	Confirmed delivery date by supplier (if available)
Statistical delivery date	Date	Delivery date used for supplier performance determination
Receipt date	Date	Date when the order is received and invoiced
Material	Alpha numeric	Unique code indicating material ordered
Material description	Text	Description of the material
Lead time	Integer	Contracted supplier lead time
Quantity	Float	Amount of material ordered
Material unit	Text	Unit of the order amount
Price per unit	Integer	Price of the default order unit
Safety time	Integer	Additional time used to move material requirements forward in time to cover for supplier deviations
ABC indicator	Text	Importance category of a material based on usage
Lot size type	Text	Category indicating frequency or time dependent replenishment
Minimum/fixed lot size	Integer	Minimum of fixed amount of time or quantity between orders or per order
MRP type	Text	Category indicating material planning type
Supplier	Integer	Unique code indicating supplier considered
Supplier description	Text	Supplier name
SNC relevancy	Text	Category indicating maturity level of information exchange

The raw dataset has been initially processed using Python 3.7 with the libraries NumPy (v1.19.1) (Harris et al., 2020), pandas (v1.1.1) (McKinney, 2010) and Matplotlib (v3.3.1) (Hunter, 2007) and filtered based on the following aspects:

- Removal of duplicate or incorrectly split deliveries.
- Manual imputation of missing values when present in the ERP system and otherwise removal of the specific deliveries.
- Unification of order units for materials over different orders.
- Removal of irregular deliveries consisting of inconsistent dates or quantities (because of Every Angle's export logic).
- Removal of deliveries of materials which are ordered less than five times at the supplier in the considered time scope.
- Removal of suppliers which have less than five unique orders in the considered time scope.

This resulted in a dataset of 68807 different deliveries corresponding to 26512 unique orders for 2899 unique materials at 180 suppliers. Of those deliveries 12321 (17.9%) were delayed and 21132 (30.7%) were delivered early. The average delivery moment is around two days before the due date. Table 5.3 presents the number of deliveries per delivery performance class.

Table 5.3: Number of deliveries per delivery performance class in the considered dataset.

Extremely Early	Early	On-time	Delayed	Extremely Delayed
12357	8775	35354	6642	5679

5.2 Feature engineering and supplier grouping

To enrich the dataset by feature engineering, feasible suggested features in the three domains presented in Table 4.2, literature and additional suggestions of Philips were engineered. Removing irrelevant features regarding delivery performance like 'PO document type' and 'Material description' from Table 5.2 while adding the engineered features resulted in the list presented in Appendix C: List of (engineered) features.

Suppliers with a minimum amount of 300 PO records and a product portfolio different than (software) licenses are selected, to explore the potential of considering individual suppliers and supplier groups. This resulted in a sub-selection of 21 suppliers, which are further categorised based on their region, to potentially explore regional effects. Three regions are defined: 'Western Europe', 'Rest of Europe' and 'Rest of World'. The classification towards 'Western Europe' or 'Rest of Europe' follows the United Nations standard country and area codes (United Nations Statistics Division, 2021). This resulted in a 'Western Europe' group consisting of 11 suppliers, a 'Rest of Europe' group of 6 suppliers and a 'Rest of World' group of 4 suppliers as presented in Table 5.4. The supplier imbalance ratios (number of on-time deliveries per delayed delivery) and number of data points are presented as well. The delivery performance distributions of the selected suppliers and groups are presented in Appendix D: Performance distributions of the selected suppliers.

After the supplier grouping the datasets per supplier are split using a 0.8-0.2 train-test split following common practice and preceding literature.

*Table 5.4: Basic (data) characteristics of the selected suppliers and groups. Imbalance ratio: number of on-time deliveries per delayed delivery. *excluded from supplier group since the performance score (CLIP) was not applicable to a large amount of the orders.*

Supplier	1	2	3	4	5	6	7	8	9	10	11	12
Region	RoW	WE	RoE	RoW	RoE	WE	RoE	WE	WE	WE	WE	RoW
Imbalance ratio	3.92	0.19	1.55	2.7	2.42	2.02	2.98	25.6	10.53	6.45	1.89	2.81
# data points	1804	633	301	529	277	1860	688	19550	680	685	689	1064

Supplier	13	14	15	16*	17	18	19	20	21	WE	RoE	RoW
Region	WE	WE	WE	RoW	RoE	WE	RoE	WE	RoE	WE	RoE	RoW
Imbalance ratio	6.39	4.99	3.77	8.56	0.9	1.8	4.09	12.3	6.71	4.73	2.89	3.31
# data points	1493	923	1501	1157	744	10909	326	399	1788	39322	4124	3397

5.3 Feature selection

This section describes the implementation and results of the recursive feature elimination (RFE) using feature permutation importance applied on the selected suppliers and supplier groups. Two suppliers (11 and 18) are selected to elucidate the results in more detail.

5.3.1 Implementation

Before applying recursive feature elimination using feature permutation importance (RFE-FPI), features irrelevant for the considered supplier (group) or highly correlated features (based on Pearson's correlation coefficient) with similar meaning are removed from the (supplier) specific subset. Irrelevant features for individual suppliers are for example 'Unique materials' since it has the same value for all data entries or 'Sup. outstanding POs (SA)' if no schedule line agreements are used besides standard POs. Observed correlated features are 'Created/Due Month' and 'Created/Due Season', which leads to omitting the season-related features. Another example is 'Outstanding PO items' and 'Outstanding quantity', which could be highly correlated if material is often ordered in the same quantity. This would result in omitting 'Outstanding PO items', since it is expected that quantity influences the delivery performance more than PO items.

A Random Forest (RF) estimator with a maximum tree depth of 6, a subset sample of 80% and a balanced subsample class weight is used for all feature selection steps in the binary and multiclass cases. A maximum tree depth of 6 is selected to reduce the potential of overfitting in the initial and especially the later stages of the RFE process when fewer features are present and is based on the minimum amount of data entries and shown performance of decision trees with a maximum depth of 6 in the work of Baryannis et al. (2019). Different maximum depths have been explored, but no significant differences in ranking were observed.

In each elimination step, a model is trained and scored using stratified 5-fold cross-validation, whereafter each feature is individually considered and permuted 30 times. The difference in model prediction performance after permutation determines the permutation importance of the permuted feature. The feature with the lowest mean permutation importance (based on the MCC) is removed and a new iteration follows until the prescribed number of resulting features is obtained.

It is expected that material is an important feature. Therefore, the RFE is run with and without the imposition of material as feature throughout the elimination process.

5.3.2 Feature selection example: supplier 11 (binary)

After the RFE is completed, model prediction performance over the number of features is visualised as depicted in Figure 5.2, which is the output of the RFE-FPI for the binary problem formulation without material as feature imposed for supplier 11. In the figure, the four selected performance metrics are expressed over the number of features selected. From the figure, it can be observed that the maximum MCC score can be obtained when the top seven features are considered, as indicated with the dashed box. In addition, the resulting accuracy, precision and recall scores for the top 7 features are (one of the) highest observed as well. Therefore, for this supplier in the binary problem formulation the top 7 features consisting of 'Material', 'Order/Lead time ratio', 'Due Week day', 'Price', 'CLIP score', 'Sup. outstanding POs (SA)' and 'Quantity' are selected, of which their permutation importance is depicted in Figure 5.3. The imposition of the 'Material' feature had no effect on this selection, since 'Material' is in the top 7 features, nevertheless.

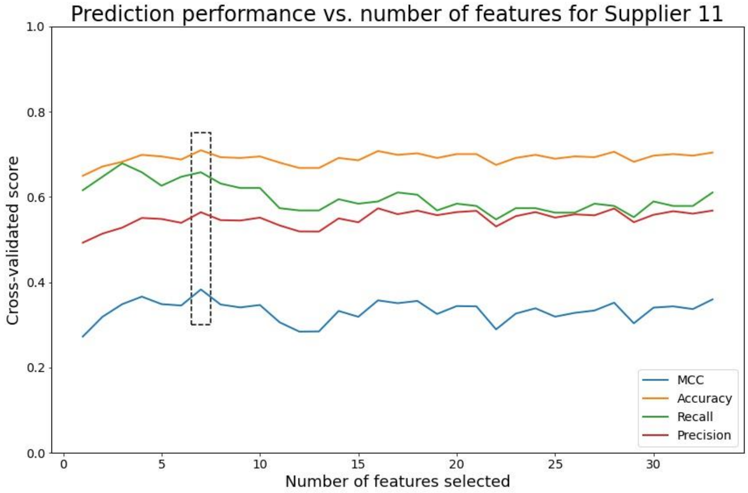

Figure 5.2: Prediction performance over number of features without material imposed as feature for supplier 11 in the binary problem formulation. Dashed box gives best performance obtained. MCC: Matthew's Correlation Coefficient, Accuracy: fraction of correct predictions, Recall: fraction of correctly classified actual delayed deliveries, Precision: fraction of correct predictions classified as delayed (see Table 3.3).

From Figure 5.3 it can be observed that the feature representing 'Material' has the most significant link with the binary prediction target, since differences in MCC performance could reach values up to 0.25. The importance of the feature 'Material' can be expected when looking at the specific supplier in more detail, since its product portfolio for Philips consists of a variety of products ranging from bulk materials as bolts and screws to specific custom-made products for Philips.

To some extent, this customisation behaviour can also be incorporated in the 'Price' feature, since custom-made products are often more expensive than off-the-shelf products, which can explain the importance of 'Price' as shown in the figure. The importance of 'Order/Lead time ratio' can be expected as well in this instance, since the time given for specific (custom made) products with respect to the contracted lead time could influence the fulfilment feasibility and therewith delivery performance. Similar holds for 'Quantity' if suddenly more Philips specific products are ordered.

Outstanding POs (and schedule line agreements) together with an expression of the CLIP score can be expected in general since it indicates the demand on the supplier and the recent performance.

The importance and selection of 'Due Week day' is an interesting result since initially this was not expected or considered as important by domain experts. However, after consultation agreed was that this could illustrate inefficiencies in process-related aspects such as day-offs or delayed invoicing.

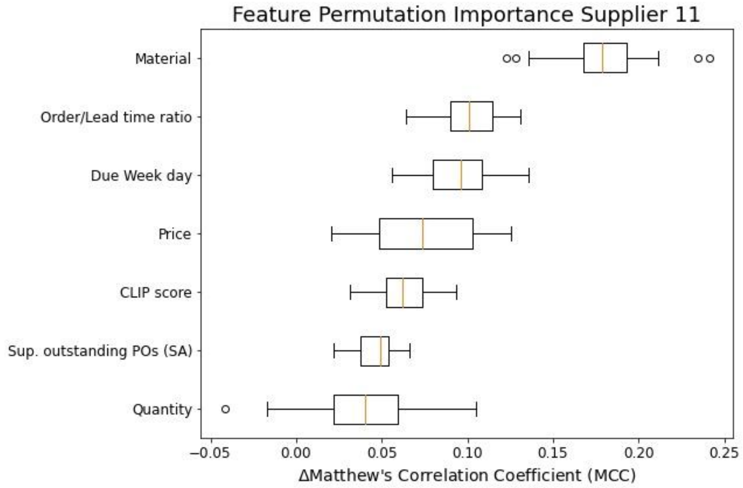

Figure 5.3: Feature permutation importance box plot of the selected features for supplier 11 in the binary problem formulation.

5.3.3 Feature selection example: supplier 18 (binary)

An example of a supplier in which the imposition of 'Material' during feature elimination makes a (negative) difference is supplier 18. Figure 5.4 shows the different performance scores for the number of features considered for this supplier. 'Material' is the seventh most important feature and after excluding it from the subset, a noticeable increase in prediction performance is observed.

This supplier produces material which has in general short lead times and uses similar raw materials for the different products in their portfolio, which could explain this sudden increase. In addition, from a more algorithmic perspective, this supplier has almost 840 unique material numbers which are considered as categorical variables, increasing the complexity to train a model. Especially if the tree that can be trained is constrained to be relatively small (because of setting the max depth to 6) for the resulting number of features, the reduction in complexity after removing the feature could result in increased prediction performance.

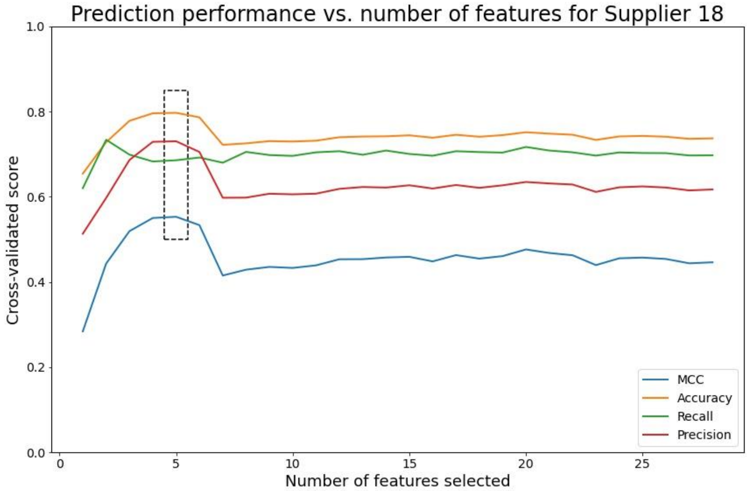

Figure 5.4: Prediction performance over number of features without material imposed as feature for supplier 18 in the binary problem formulation. Dashed box gives best performance obtained. MCC: Matthew's Correlation Coefficient.

To verify that this drop in complexity and increase in performance is not the result of overfitting, training performance is investigated. The mean and standard deviation of the performance scores of the different folds of the trained model are presented in Table 5.5. Similar performance is ranges are observed on all metrics, illustrating the generalisability of the trained model rather than the occurrence of overfitting.

Table 5.5: Mean and standard deviation of prediction performances of the stratified 5-fold feature selection model for the top 5 features for supplier 18 in the binary problem formulation. MCC: Matthew's Correlation Coefficient, RF: Random Forest.

Supplier 18	Test fold	Train fold
MCC	0.5530 ± 0.0108	0.5731 ± 0.0056
Accuracy	0.7971 ± 0.0058	0.8065 ± 0.0020
Precision	0.7303 ± 0.0174	0.7452 ± 0.0053
Recall	0.6857 ± 0.0199	0.6960 ± 0.0141
Algorithm settings: RF		
Max depth: 6	Class weight: balanced subsample	
Max samples: 0.8		

The selected features and their permutation importance are depicted in Figure 5.5. The selection of 'Due Week day' as well as 'Due Week' could suggest that specific periods in history were characteristic for delivery (mis-)performance, which could question the possibility to generalise historical behaviour towards future predictions. Visualising the delivery performance over the 'Due Week day' and 'Due Week' as shown in Figures 5.6 and 5.7 does not explicitly show such periods. Therefore, it is expected that the increase in performance can be accounted for by the decrease in data complexity and supplier specific characteristics.

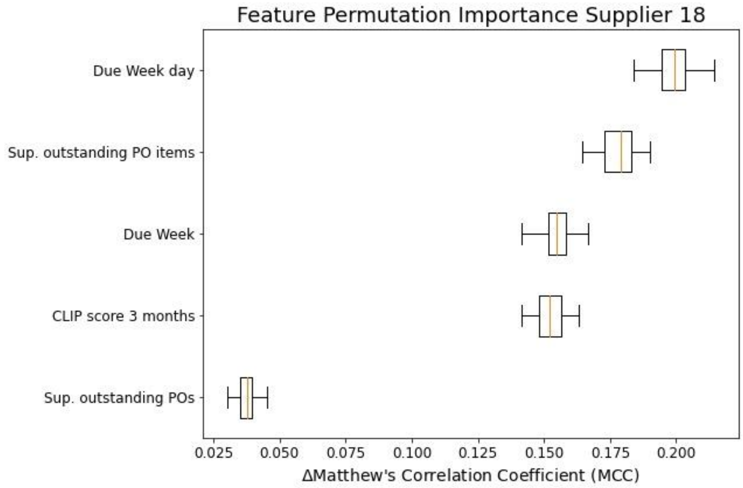

Figure 5.5: Feature permutation importance box plot of the selected features for supplier 18 in the binary problem formulation.

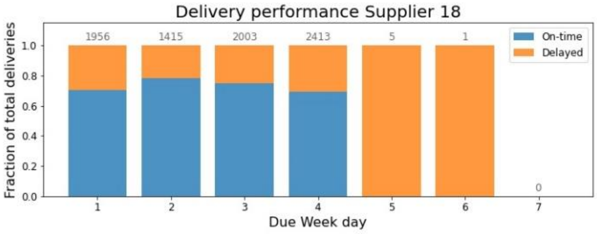

Figure 5.6: Delivery performance of supplier 18 over 'Due Week day'. Total numbers of data entries corresponding to the 'Due Week day' are presented in dark grey above the bars.

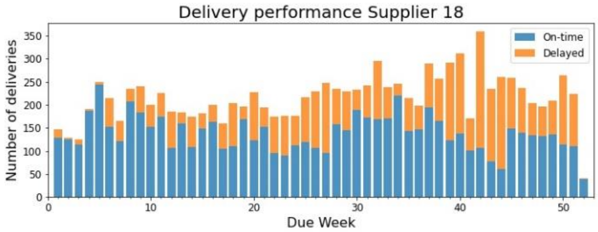

Figure 5.7: Delivery performance of supplier 18 over 'Due Week'.

5.3.4 Feature selection results for all considered suppliers and groups

The same approach and exploration are conducted for each supplier (group) for the binary and multiclass problem formulation, resulting in the selected number of features per supplier (group) as presented in Tables 5.6 and 5.7. The specific features selected per supplier (group) are presented in Appendix E: Selected features per supplier (group). In 3 out of 48 situations, the explicit imposition of 'Material' was valuable for prediction performance, which for supplier 5 could be expected due to the material-specific production technique (moulding) which is used.

In Tables 5.6 and 5.7 the selected number of features per supplier in both problem formulations are presented. Differences between both problem formulations can be expected since the increased complexity of multiclass classification can require additional data or brings out different relations between features and the added classes.

Differences between suppliers can result from unique supplier behaviour and characteristics leading to different important features, or the ability to extract (and generalise) this behaviour from the available dataset.

Table 5.6: Selected number of features for the selected suppliers and groups in the binary problem formulation.
**excluded from supplier group since the performance score (CLIP) was not applicable to large amount of supplier's orders.*

Supplier	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16*	17	18	19	20	21	WE	RoE	RoW
# of features	10	11	11	9	11	11	11	14	4	13	7	13	6	12	13	6	6	5	8	3	17	9	14	11
Mat. imposed	-	-	-	-	x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Region	RoW	WE	RoE	RoW	RoE	WE	RoE	WE	WE	WE	WE	RoW	WE	WE	WE	RoW	RoE	WE	RoE	WE	RoE	WE	RoE	RoW

Table 5.7: Selected number of features for the selected suppliers and groups in the multiclass problem formulation.
**excluded from supplier group since the performance score (CLIP) was not applicable to large amount of supplier's orders.*

Supplier	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16*	17	18	19	20	21	WE	RoE	RoW
# of features	10	6	7	5	9	11	7	17	8	8	13	15	17	12	15	12	13	14	9	-	9	6	8	16
Mat. imposed	-	x	-	-	x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Region	RoW	WE	RoE	RoW	RoE	WE	RoE	WE	WE	WE	WE	RoW	WE	WE	WE	RoW	RoE	WE	RoE	WE	RoE	WE	RoE	RoW

To identify commonly shared important features, the selected features for the individual suppliers and the regional supplier groups are combined into region-based bar charts using the top 5 features of each individual supplier, which are depicted in Figures 5.8-5.13. From the figures, expected important features as 'Material', 'Order/Lead time ratio' and 'Sup. outstanding quantity' are often observed throughout the different regions for individual suppliers and the supplier groups. However, interesting is the recurring importance of 'Due Week day' in both problem formulations. As stated in the example of supplier 11 before (section 5.3.2), this might be the result of inefficiencies in process-related aspects, which becomes more plausible since it is observed for multiple individual suppliers.

For completeness, the region-based bar charts consisting of all selected features are visualised in Appendix F: Overviews of selected features per region.

5.3.5 Feature selection: additional value

The unexpected identification of 'Due Week day' is an example of how feature importance and selection can assist in identifying potential directions and causes of non-optimal supplier relations or behaviour for individual suppliers or groups. These directions can be used for internal investigations or performance improvement programmes to potentially discover inefficiencies which can lead to more efficient and reliable supplier relations. Therewith, feature importance and selection can assist in mitigating disruptions on a tactical level.

It is expected that feature importance in the binary case is initially more viable, since it directly reflects the importance of the distinction between on-time and delayed deliveries.

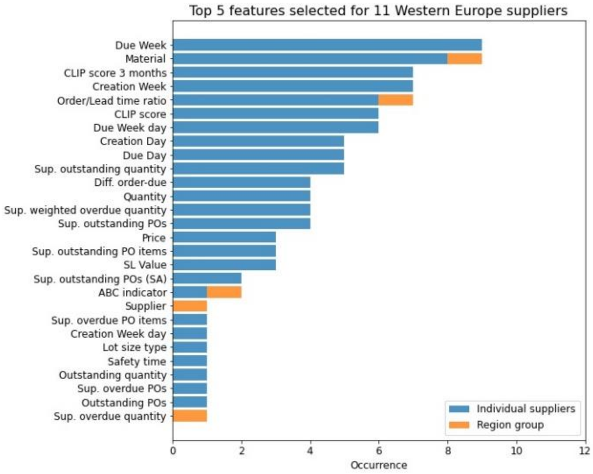

Figure 5.8: Overview of features present in top 5 for each individual supplier (group) in Western Europe for the binary problem formulation.

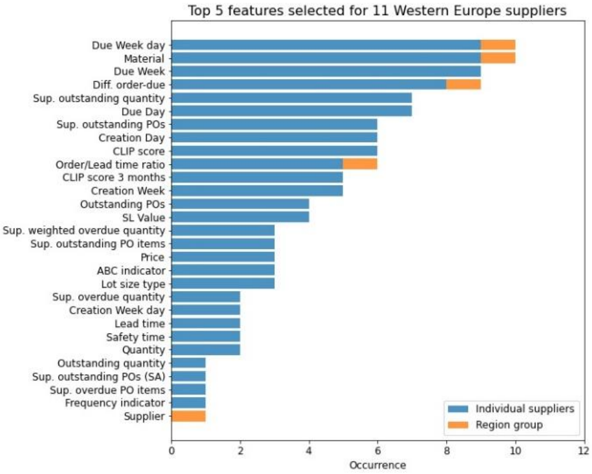

Figure 5.9: Overview of features present in top 5 for each individual supplier (group) in Western Europe for the multiclass problem formulation.

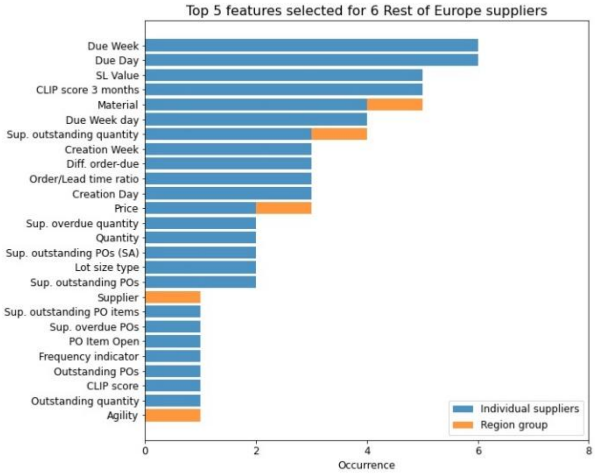

Figure 5.10: Overview of features present in top 5 for each individual supplier (group) in the Rest of Europe for the binary problem formulation.

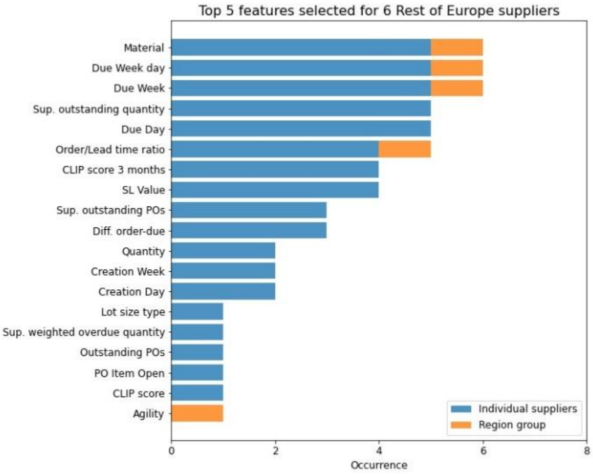

Figure 5.11: Overview of features present in top 5 for each individual supplier (group) in the Rest of Europe for the multiclass problem formulation.

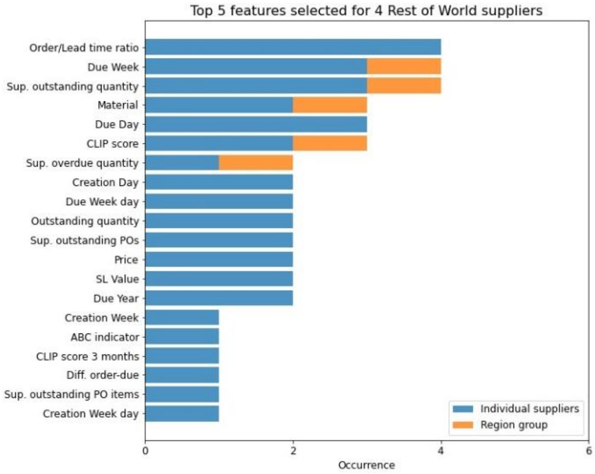

Figure 5.12: Overview of features present in top 5 for each individual supplier (group) in the Rest of World for the binary problem formulation.

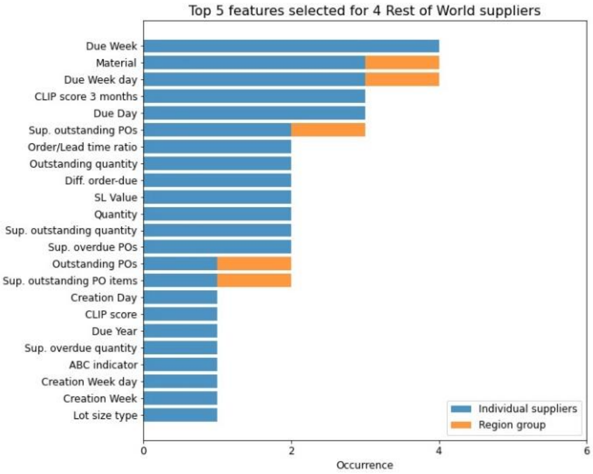

Figure 5.13: Overview of features present in top 5 for each individual supplier (group) in the Rest of World for the multiclass problem formulation.

5.4 Algorithm evaluation

After feature selection described in the previous section, different model configurations are created, trained and compared. Using python 3.7 with the libraries NumPy (v1.19.1) (Harris et al., 2020), pandas (v1.1.1) (McKinney, 2010), Matplotlib (v3.3.1) (Hunter, 2007), scikit-learn (v0.24.0) (Pedregosa et al., 2011), imbalanced-learn (v0.7.0) (Lemaître, Nogueira, & Aridas, 2017) and xgboost (v1.3.1) (Chen & Guestrin, 2016) these different model configurations are implemented and tested and results visualised. The models are trained on a computer with an Intel Core I5-8365 CPU and 16 GB RAM.

For each supplier (group) the parameter grids presented in Tables 5.8 are considered in the binary and multiclass problem formulations. The grids follow the suggested grid proposed in the methodology, with the addition to add scaling of features for LR and SVM and a minor change in the gamma range for SVM based on initial explorations. In the multiclass case, the LR and SVM implementations are omitted due to the significant increase in computational time and limited observed performance. Default algorithm implementations from the scikit-learn and xgboost libraries were used as basis in which the grid parameters were implemented. For the RF and XGB algorithms a subsample size of 0.8 is additionally set to stimulate variation in internal estimators.

Table 5.8: Applied parameter values in the grid searches per considered algorithm in both problem formulation.

Algorithm	Parameters	Range (binary)	Range (multiclass)
Logistic Regression (LR)	- Regularisation – C - Class weight - Solver - Sampling - Scaling	- 0.1, 0.5, 1, 5, 10, 50, 100, 500, 1000 - None, balanced - Newton-cg, lbfgs - None, SMOTENC, RUS, Hybrid - Yes, No	
Decision Tree (DT)	- Max tree depth - Max leaf nodes - Class weight - Sampling	- 5, 6, 7, 8, 9, 10 - 20, 30, 40, 50 - None, balanced - None, SMOTENC, RUS, Hybrid	- 5, 10, 15, 20 - 100, 200, 400, 800 - None, balanced - None, SMOTENC, RUS, Hybrid
Random Forest (RF)	- Max tree depth - Max leaf nodes - Nr. of estimators - Class weight - Sampling	- 5, 6, 7, 8, 9, 10 - 20, 30, 40, 50 - 10, 50, 100, 500 - None, balanced_subsample - None, SMOTENC, RUS, Hybrid	- 5, 10, 15, 20 - 100, 200, 400, 800 - 10, 50, 100, 500 - None, balanced_subsample - None, SMOTENC, RUS, Hybrid
Support Vector Machine (SVM)	- Regularisation – C - Gamma – γ - Class weight - Sampling	- 0.1, 1, 10, 100, 1000, 10000 - 0.001, 0.01, 0.1, 1, 10 - None, balanced - None, SMOTENC, RUS, Hybrid	
eXtreme Gradient Boosting (XGB)	- Max tree depth - Min child weight - Gamma – γ - Nr. of estimators - Sampling	- 5, 6, 7, 8, 9, 10 - 1, 2, 3 - 0, 1, 10, 100 - 10, 50, 100, 500 - None, SMOTENC, RUS, Hybrid	- 5, 10, 15, 20 - 0, 1 - 0, 1, 10 - 10, 50, 100 - None, SMOTENC, RUS, Hybrid

After completing the grid searches and analysing the results of the best performing parameter combinations many cases of overfitting (and poor generalisation performance) were observed, indicated by noticeable differences between train and test performance during model training (Figure 5.14). Figure 5.14 presents the MCC performances of the best performing models (in terms of MCC performance) on the test set, test and train folds during model training for the suppliers and groups.
An example of such an overfitted model is the best model for supplier 4, of which the train and test results are presented in Table 5.9. The performance scores obtained from the independent test set are presented in the first column 'Score on test set'. The mean training scores and their standard deviation resulting from the cross-validation during training are presented in the second and third column.

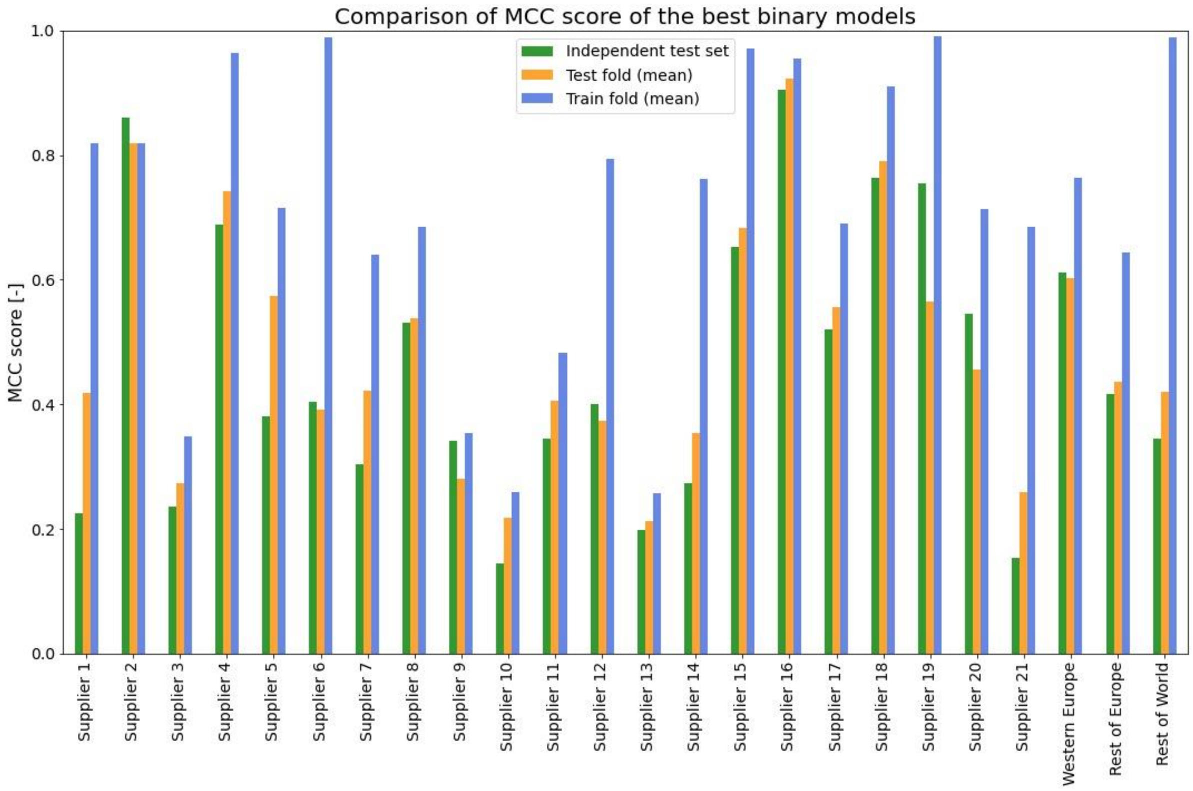

Figure 5.14: Matthew's Correlation Coefficient (MCC) scores for the best model configuration in the considered binary problem formulation. 'Best' is the algorithm configuration resulting in the highest MCC score on the test set after selecting the highest performing configurations in the grid per algorithm based on the test fold MCC score.

Table 5.9: Prediction performance of overfit eXtreme Gradient Boosting (XGB) model for supplier 4 in the binary problem formulation. Test set: performance on independent test set. Test fold: mean performance ± standard deviation on test fold during training. Train fold: mean performance ± standard deviation on train fold. MCC: Matthew's Correlation Coefficient.

Supplier 4	Test set	Test fold	Train fold
MCC	0.6883	0.7426 ± 0.0868	0.9639 ± 0.0031
Accuracy	0.8774	0.9006 ± 0.0324	0.9858 ± 0.0012
Precision	0.7857	0.8489 ± 0.0605	0.9911 ± 0.0083
Recall	0.7586	0.7711 ± 0.1029	0.9561 ± 0.0099
Algorithm settings: XGB			
Gamma: 0	Max depth: 8	Min child weight: 1	
# estimators: 100	Sampling: None	Max subsample: 0.8	

The high performance scores on the train fold with respect to the noticeable lower performance scores on the test fold illustrate this overfitting behaviour, which limits the generalisability and therewith applicational value of the specific model. Therefore, to prevent comparing overfitted model results and drawing incorrect conclusions, all models with a delta mean MCC score of 0.1 between the train and test folds during model training are omitted from the results. This results in lower obtained MCC performance, but higher generalisability, as is illustrated in Figure 5.15 by similar MCC performance over the independent test set, test and train folds.

Taking supplier 4 again as example, the results of the new best XGB model are presented in Table 5.10[4]. The results show similar performance over the test set, test and train folds and noticeable are the differences in parameter settings. In this case gamma and minimum child weight are changed from 0 to 1 and 1 to 3 respectively, which were expected to reduce the potential of overfitting, as is illustrated.

[4] The new best model is trained using DT. However, the results of the new best XGB model are used for comparison and maintaining consistency between examples.

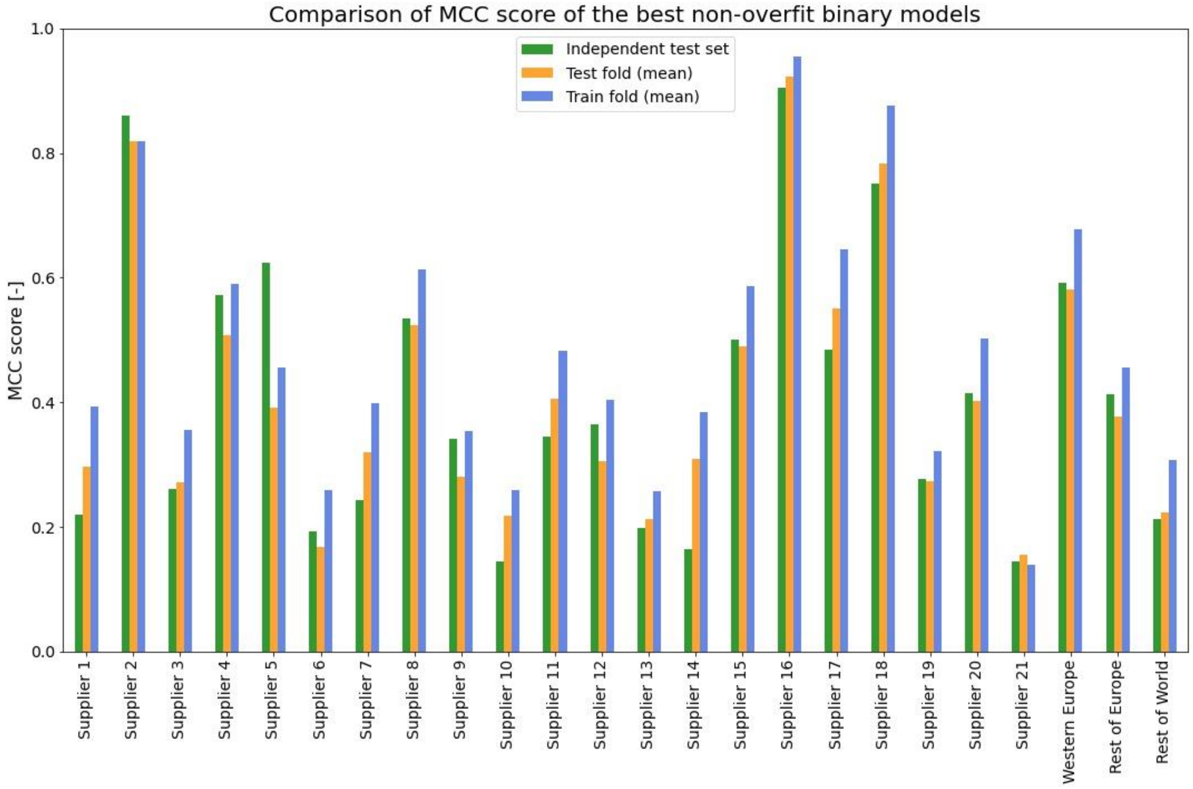

Figure 5.15: Matthew's Correlation Coefficient (MCC) scores for the best non-overfit model configuration in the binary problem formulation. 'Best non-overfit' is the algorithm configuration resulting in the highest MCC score on the test set after selecting the highest performing configurations in the grid per algorithm based on the test fold MCC score with $|MCC_{test\ fold} - MCC_{train\ fold}| \leq 0.1$.

Table 5.10: Prediction performance of best non-overfit eXtreme Gradient Boosting (XGB) model for supplier 4 in the binary problem formulation. Test set: performance on independent test set. Test fold: mean performance ± *standard deviation on test fold. Train fold: mean performance* ± *standard deviation on train fold. MCC: Matthew's Correlation Coefficient.*

Supplier 4	Test set	Test fold	Train fold
MCC	0.5545	0.5458 ± 0.1445	0.6358 ± 0.0343
Accuracy	0.8302	0.8367 ± 0.0446	0.8629 ± 0.0117
Precision	1.0000	0.8489 ± 0.0882	0.9395 ± 0.0293
Recall	0.3793	0.4652 ± 0.1478	0.5262 ± 0.0499

Algorithm settings: XGB		
Gamma: 1	Max depth: 9	Min child weight: 3
# estimators: 100	Sampling: None	Max subsample: 0.8

From an operational perspective, the results in Table 5.10 show value given the MCC and accuracy score and the 100% score on the deliveries classified as too late (precision). However, only 38% of the actual delayed deliveries have been correctly identified using this model, reducing this applicational value. Additional post-processing without altering the trained model could assist in steering this behaviour towards the class and metric of most interest. In the Philips' case with the aim for production continuity, this would translate to aiming for higher recall values at some cost of precision and general accuracy. Applying threshold tuning on Table 5.10's model as proposed in the methodology by maximising the Youden's statistic or F_β-Score leads to the results presented in Table 5.11. From the table it can be observed that in both cases the correct identification of delayed deliveries is increased at the cost of correctly identifying on-time deliveries (reduced accuracy and reduced precision), therewith illustrating the trade-off that can be made and increasing the potential applicational value on an operational level.

Table 5.11: Comparison of eXtreme Gradient Boosting (XGB) model prediction performance for supplier 4 in the binary problem formulation when applying threshold tuning. ROC: tuning towards optimal Youden's statistic score. PRC: tuning towards optimal F_1-score. MCC: Matthew's Correlation Coefficient.

Supplier 4	No tuning	ROC	PRC
MCC	0.5545	0.5138	0.4547
Accuracy	0.8302	0.7642	0.7453
Precision	1.0000	0.5455	0.5238
Recall	0.3793	0.8276	0.7586
Algorithm settings: XGB			
Gamma: 1	Max depth: 9	Min child weight: 3	
# estimators: 100	Sampling: None	Max subsample: 0.8	

Comparing the sampling strategies of the models presented in Figure 5.15 and their overfit counterparts, the sampling strategies presented in Table 5.12 are observed. For the overfit models over-sampling or no sampling are more applied which can be expected. Especially since over-sampling could decrease generalisation performance and lead to overfitting (Weiss et al., 2007) and no sampling could indicate too much freedom during model training.

The use of under-sampling (combined with over-sampling) seems to be valuable to reduce overfitting, increase generalisation and reduce negative impacts of class imbalance, which were reasons for Brintrup et al. (2020) to not even consider applying over-sampling in their research.

Table 5.12: Applied sampling strategies (columns) in best model performances illustrated in figures 5.14 (overfitting) and 5.15 (non-overfitting) with and without overfit mitigation in the binary problem formulation.

	Over-sampling	**Under-sampling**	**Hybrid sampling**	**No sampling**
Overfitting	5	2	5	12
Non-overfitting	0	9	9	6

Tables 5.13 and 5.14 present the model performance scores on the independent test sets for the best performing supplier-algorithm combinations (without post-processing) for the binary and multiclass formulations, respectively. In Table 5.14 the precision and recall scores are macro-averaged and for supplier 20 no parameter combination in the applied grid led to a similar (non-overfit) MCC score in the train and test folds.

*Table 5.13: Overview of best prediction performance scores on the test set and corresponding algorithm per supplier (group) for the binary problem formulation. *excluded from supplier group since the performance score (CLIP) was not applicable to a large amount of the orders. MCC: Matthew's Correlation Coefficient, LR: Logistic Regression, DT: Decision Tree, RF: Random Forest, SVM: Support Vector Machine, XGB: eXtreme Gradient Boosting.*

Region	**Supplier**	**Algorithm**	**Sampling**	**MCC**	**Accuracy**	**Precision**	**Recall**
Western Europe (WE)	**2**	**XGB**	**Hybrid**	**0.8604**	**0.9606**	**0.9810**	**0.9717**
	6	XGB	Under	0.1926	0.5860	0.4188	0.6504
	8	*XGB*	*Hybrid*	0.5345	0.9716	0.7000	0.4286
	9	SVC	Under	0.3247	0.6691	0.2000	0.9167
	10	LR	Hybrid	0.1451	0.7080	0.2105	0.4444
	11	LR	Under	0.3453	0.6667	0.5147	0.7292
	13	SVC	Hybrid	0.1993	0.7391	0.2500	0.4750
	14	SVC	None	0.1643	0.8378	1.0000	0.0323
	15	*XGB*	*None*	0.4999	0.8505	0.7143	0.4762
	18	**XGB**	**None**	**0.7510**	**0.8863**	**0.8517**	**0.8254**
	20	*XGB*	*Hybrid*	0.4157	0.9125	0.4286	0.5000
	WE	*XGB*	*None*	0.5915	0.8945	0.8175	0.5091

Table 5.13 (continued).

Region	Supplier	Algorithm	Sampling	MCC	Accuracy	Precision	Recall
Rest of Europe (RoE)	3	XGB	Under	0.2618	0.6066	0.5000	0.7500
	5	*SVC*	*Under*	0.6244	0.8214	0.6364	0.8750
	7	XGB	None	0.2435	0.7681	0.8000	0.1143
	17	*RF*	*Under*	0.4842	0.7383	0.7910	0.6795
	19	XGB	Under	0.2769	0.7576	0.4000	0.4615
	21	LR	Hybrid	0.1442	0.6341	0.1898	0.5652
	RoE	DT	None	0.4128	0.7964	0.6594	0.4292
Rest of World (RoW)	1	DT	Hybrid	0.2202	0.6233	0.3019	0.6575
	4	*DT*	*Hybrid*	0.5728	0.7830	0.5652	0.8966
	12	XGB	Under	0.3652	0.6854	0.4421	0.7500
	16*	**SVC**	**Hybrid**	**0.9042**	**0.9828**	**1.0000**	**0.8333**
	RoW	DT	Under	0.2121	0.6044	0.3271	0.6646

*Table 5.14: Overview of best prediction performance scores on the test set and corresponding algorithm per supplier (group) for the multiclass problem formulation. Precision and recall values are macro-averaged. *excluded from supplier group since the performance score (CLIP) was not applicable to a large amount of the orders. MCC: Matthew's Correlation Coefficient, DT: Decision Tree, XGB: eXtreme Gradient Boosting.*

Region	Supplier	Algorithm	Sampling	MCC	Accuracy	Precision	Recall
Western Europe (WE)	*2*	*XGB*	*Hybrid*	0.5877	0.8583	0.4743	0.4790
	6	XGB	Hybrid	0.2621	0.3898	0.4129	0.4325
	8	XGB	Hybrid	0.4749	0.9494	0.5137	0.5447
	9	XGB	Hybrid	0.4919	0.6985	0.3184	0.3712
	10	XGB	Hybrid	0.3697	0.5401	0.3844	0.4038
	11	XGB	Hybrid	0.2000	0.3478	0.3577	0.3544
	13	DT	Under	0.1204	0.2274	0.3182	0.3534
	14	XGB	Hybrid	0.1523	0.3405	0.3186	0.3688
	15	XGB	None	0.3479	0.5781	0.5690	0.3384
	18	*XGB*	*Hybrid*	0.6691	0.7438	0.7439	0.7453
	20	-	-	–	–	–	–
	WE	*XGB*	*Under*	0.6047	0.7612	0.6136	0.6739
Rest of Europe (RoE)	3	XGB	Hybrid	0.1928	0.3770	0.5028	0.3349
	5	XGB	None	0.3796	0.5179	0.3738	0.3706
	7	XGB	Hybrid	0.3947	0.5580	0.4727	0.4335
	17	XGB	Hybrid	0.3733	0.5638	0.3536	0.4243
	19	XGB	Hybrid	0.0000	0.4091	0.0818	0.2000
	21	XGB	None	0.2659	0.5168	0.5535	0.3212
	RoE	XGB	Hybrid	0.3822	0.5176	0.4989	0.5015
Rest of World (RoW)	1	XGB	Hybrid	0.3937	0.5706	0.4865	0.4505
	4	XGB	Hybrid	0.4971	0.6509	0.5652	0.5389
	12	XGB	None	0.4881	0.6244	0.5776	0.4505
	16*	**XGB**	**Under**	**0.7569**	**0.8836**	**0.8513**	**0.8023**
	RoW	XGB	Hybrid	0.3588	0.5206	0.5015	0.4502

When comparing the MCC and accuracy scores of the different suppliers, it can be observed that very different levels of performance are achieved on the same grid of model parameters, with a shared difficulty to create high performing models in the multiclass formulation. The resulting supplier models range from almost random predictions (supplier 19, multiclass) to predicting with high accuracy (supplier 16, binary).

Suppliers indicated in bold in Tables 5.13 and 5.14 correspond to supplier prediction models which seem suitable for direct implementation as decision support using a threshold of 0.8 for recall and accuracy. Suppliers in italic are expected to lead to applicable prediction models when applying threshold tuning, considering a finer grid, or performing additional individual parameter tuning. Nevertheless, for slightly more than half of the considered suppliers no valuable prediction model could be created, illustrating the need for further (follow-up) research.

From the results it can also be observed that over-sampling alone has not led to best performing models in both problem formulations. In the binary case no particular algorithm-sampling method combination consistently led to the best results, although (combined) under-sampling is observed in 75% of the cases. In the multiclass formulation the XGB-Hybrid sampling combination often leads to the best results. However, the multiclass performance cannot equal the obtained performances in the binary formulation.

To get insights and potential explanations for the observable difference in performance or success factors, data characteristics of the different suppliers were compared. However, no shared characteristics visible in the dataset were observed for the successful or potential suppliers that were noticeably different to the unsuccessful suppliers. Examples are imbalance ratios ranging from 0.19 to 25.6, dataset sizes ranging from 277 to 39322 entries and the selected number of features ranging from 3 to 14.

Since the applied methodology and considered parameter grids are consistent throughout the different suppliers and the same data source is used, it is expected that the reason for the different performance follows from individual supplier behaviour itself, insufficient available data (for the encountered scenarios), Philips' interactions with its suppliers or additional external influences. Especially since some of these aspects are known to be present and not explicitly represented in the available dataset as identified and stated before (see section 5.1.1).

5.5 Discussion and reflection

The obtained results show that high prediction performances can be obtained after application of the proposed methodology on the available data set. In the binary case MCC scores up to 0.9 are achieved, accompanied with 98% accuracy, 100% precision and 83% recall on independent test sets. In the multiclass formulation MCC scores up to 0.75 are achieved, accompanied with 88% accuracy, 85% macro-precision and 80% macro-recall. However, performance in the multiclass case is in general lower than its binary counterpart. This can be expected since the introduction of additional prediction classes increases the complexity of the classification task and data requirements, since distinctions between all classes need to be represented in the dataset as well. Therefore, multiclass classification is proposed to be a follow-up and extension of binary classification once valuable performances are obtained in the binary problem formulations.

For the best obtained results, under-sampling (combined with over-sampling) has been applied in 75% and 83% of the cases in the binary and multiclass problem definition, respectively. This illustrates the value of sampling techniques, and in particular under-sampling, to reduce negative effects of class imbalance and increase generalisability and performance of prediction models in supplier disruption prediction.

Focussing on the obtained results, it is difficult to compare the applied region-based grouping with its individual components. The combination of different suppliers with different sizes influences the results to such extent that it is difficult to state if a region-based supplier group can cover general supplier behaviour better than its individual components purely based on the performance scores. In the 'Western Europe' group for example, suppliers 8 and 18 account for 77% of the total number of orders, expecting that the performance of the grouped model is highly in line with their individual models. However, when comparing the selected features (see Appendix E: Selected features per supplier (group)), the main features of supplier 18 are not selected in the supplier group while high performance[5] is obtained. This indicates that different, more generalised behaviour is captured and therewith value of the applied grouping.

In contrast, in the 'Rest of World' (RoW) group each individual supplier model performs better than the group model, illustrating that the applied grouping is too limited. Therefore, different methods or logic for grouping should be applied in future research.

Reflecting on the obtained results from the binary formulation presented in Table 5.13, one could question the general applicability of the proposed methodology and the value of ML in this case. However, since suppliers are present for which high performing or promising prediction models can be trained, it is difficult to believe the methodology is the limiting factor for the unsuccessful suppliers. The same holds for the data usage since order data are created and managed in a similar way. Therefore, it is expected that influences on delivery behaviour, such as communication between buyers and suppliers and shared forecast information (as suggested in Figure 5.1), affect the captured behaviour in the data to such extent that operational behaviour cannot be extracted using the available data in combination with the proposed methodology. This hypothesis can be supported and potentially verified by consulting responsible buyers for each of the considered suppliers and compare their answers regarding the amount of communication with suppliers, frequency of executing mitigating measures and general experience with the supplier. Unfortunately, this could not be included in this research, and is urged to carry out for the initiated follow-up projects.

Nevertheless, because of these uncaptured influences, the inability to successfully extract supplier behaviour could assist in identifying operations which are (heavily) manually influenced, or suppliers which perform unpredictable themselves. In both cases this translates to re-evaluating the current supply chain design and interactions, which could lead to a better performing, more predictable supplier relation. Therewith, assisting the mitigation of disruptions on a more structural (tactical) level.

[5] An MCC score of 0.59 with an accuracy of 90% and 82% precision

6 Conclusion and recommendations

This chapter presents the main findings of this research and recommendations for future research directions and Philips. In section 6.1 the main findings regarding the defined sub-questions are presented whereafter the main research question is answered. Section 6.2 presents recommendations for future research, whereafter section 6.3 concludes this report by presenting recommendations for Philips.

6.1 Main findings

Chapter 2 focussed on the first sub-question by formulating a generalised presentation of a production environment focusing on the interaction between production and its suppliers. Common physical and information flows are presented, illustrating the distinction between the material demand and supply side, in which procurement (planners and buyers) takes a central managing role. Material Requirement Planning (MRP) systems are commonly used in production systems, storing large amounts of data regarding the material supply-related flows in created purchase orders (POs). Therefore, production environments in which procurement manages material supply and stores historical POs are considered, potentially accompanied by data sources covering additional information flows as shared demand forecasts or executed mitigating measures.

The second sub-question is addressed in chapter 3, resulting in a literature review focussing on applied machine learning (ML) techniques and algorithms for supplier disruption prediction. The review explained the observed methodologies in literature and showed a limited number of studies, illustrating the novelty of this empirical field. Only the application of binary classification for predicting delayed deliveries or stock-out occurrences is observed, indicating the unexplored field of multiclass classification and therewith a gap in current literature. Additionally, different performance metrics are considered in the different studies, reducing the comparability of obtained performance results. It was found that the presence of a (manufacturing) party with specific domain knowledge is necessary for meaningful feature engineering and selection. Sampling techniques as Synthetic Minority Over-sampling (SMOTE) and random under-sampling (RUS) to reduce negative influences of class imbalance combined with Logistic Regression (LR), Decision Tree (DT), Random Forest (RF) and Support Vector Machine (SVM) algorithms are mainly applied. The common approach to explore different algorithms indicate the need for research focussing on faster identification of potential algorithms such that more time can be allocated to increasing model performance rather than exploring different algorithms.

Chapter 4 focuses on the developed methodology as answer to the third sub-question. The proposed methodology consists of six steps incorporating binary and the novel extension to multiclass classification. Additionally, the methodology considers individual suppliers and custom supplier groups rather than all suppliers at once, which is different from preceding work. Considering individual suppliers and custom groups reduces initial complexity. Additionally, extracted supplier specific behaviour can be easier verified and accepted by buyers. The six steps of the methodology are:

1. *Data collection and exploration*: system analysis incorporated with data collection and focuses on understanding the production system, its data generation (influences), limitations and prediction goal.
2. *Performance and metric definition*: translating the prediction goal to suitable metrics for model performance evaluation throughout model development. Matthew’s Correlation Coefficient (MCC) is selected as primary performance metric supported by accuracy, precision and recall.

3. *Data preparation and feature engineering*: transformation to suitable ML formats of the collected data elements, removal of incorrect data entries or noise and addition of data characteristics (features) using available raw data, domain knowledge and experience. For feature engineering 3 main feature domains ("Order", "Supplier-material" and "Dynamic 'environment'") including feature suggestions are defined and proposed.
4. *Supplier grouping and feature selection*: defining additional supplier groups using the individually considered suppliers and reducing the dimension and complexity of the resulting sub-problems by feature selection. Recursive feature elimination using feature permutation importance is considered rather than standard feature importance to support categorical features as well as giving a clearer result regarding relevant characteristics related to observed supplier (group) behaviour.
5. *Data pre-processing*: application of ML-algorithm specific transformations as data scaling or normalisation and the possibility to apply resampling on each considered subset. No sampling, SMOTENC, random under-sampling and a hybrid formulation of both are incorporated to incorporate support for categorical features while extending the currently applied techniques.
6. *Algorithm comparison and evaluation*: evaluation of model performances for the individual suppliers and supplier groups resulting from algorithm parameter grids for five[6] different ML algorithms. Additionally, post-processing by means of threshold tuning for binary classification is incorporated.

Chapter 5 presents the results of the application of the methodology on 21 suppliers in a case study at Philip's production facility to explore its potential value and answer the fourth sub-question. Historical PO data has been made available to predict if placed POs will be delivered delayed in the binary case, or if they will be delivered extremely early, early, on-time, delayed or extremely delayed in the multiclass case. Results from the feature selection step indicate an unexpected importance of the feature 'Due Week day', corresponding to the day of the week when an order was due. Discussion with practitioners led to the hypothesis that inefficiencies occur in process-related aspects like day-offs of buyers or delayed invoicing, which further (internal) research needs to verify. In the binary problem formulation, MCC scores up to 0.9 are obtained, accompanied with 98% accuracy, 100% precision and 83% recall on independent test sets for individual suppliers. In the multiclass case lower performances are observed (0.75 MCC, 88% accuracy, 85% macro-precision and 80% macro-recall), illustrating the possibility to develop multiclass prediction models with the proposed methodology. However, the generally lower performance with respect to binary classification leads to the conclusion that multiclass classification for now must be seen as an extension of binary classification when valuable performance in binary classification is (already) obtained. Additionally, threshold tuning showed the possibility to improve identification of actual delayed deliveries at the cost of increasing the number of false positives.

The results also illustrate the importance to consider individual suppliers or subsets and be aware of limitations when focussing on the entire dataset as whole as it can translate to a 'supplier imbalance problem', negatively influencing the results and practical use. The value of applying resample techniques (under-sampling, potentially combined with over-sampling) to reduce negative effects of class imbalance and overfitting is illustrated as well.

[6] Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, and eXtreme Gradient Boosting

The differences in prediction performance between individual suppliers show that historical PO data is insufficient in a general context. Literature obtained sufficient prediction performances resulting from using historical PO data, but this research illustrated that additional factors influence prediction performance and feasibility for individual suppliers, emphasizing the importance of the first step of the methodology *"Data collection and exploration"*.
Additionally, preceding literature does not mention overfitting or generalisation ability of prediction models and does not state training performance, making it difficult to put their results into perspective. As figures 5.14 and 5.15 illustrate, higher prediction scores can be obtained when a model is overfit and recreates the dataset rather than the covered relations and behaviour. Especially in the research from Brintrup et al. (2020), it would have been valuable to present the train results, since the limited amount of considered features combined with the available datapoints with respect to the used maximum tree depth could result in overfitting. Therefore, preceding results cannot be generalised and similar methodologies and reporting are needed, in which the proposed methodology aims to assist.

The possibility of the methodology to identify important characteristics relevant for observed delayed deliveries can assist root cause analysis to improve (internal) operations or supplier relations. Additionally, differences between suppliers of successful and less successful supplier models can be investigated which can lead to redesigns of supply chain components. Therewith, both can assist in mitigating disruptions on a tactical level. The high-performing prediction models can be applied on an operational level assisting planners and buyers in time prioritisation, evaluating production plan feasibility, and increasing the time window in which mitigating measures can be applied. This can lead to less expensive mitigating measures and use of different transport alternatives as well.

Therefore, the research question *"How can machine learning be applied to assist in the mitigation of material-oriented supplier disruptions in production environments?"* can be answered by concluding that machine learning can be applied to assist in mitigating material-oriented supplier disruptions on an operational and tactical level. This includes assisting in evaluating the feasibility of (upcoming) orders and production plans, increasing the time window for executing mitigating measures and identifying possible root causes for inefficient processes or supplier relations. In the Supply Chain Resilience Framework, this translates to the expected contributions towards an increase in supply chain resilience by increasing awareness, flexibility and visibility. However, additional research is needed to increase overall prediction performance (in the specific case study) and therewith direct applicability in operations.

6.2 General recommendations

Multiple future research directions can be defined to increase the possibilities of the proposed methodology and improve the application value. Grouping of suppliers has been manually done based on region to potentially discover regional-related effects. However, the application of manual grouping on region might be too limited and different grouping methods or characteristics to group by need to be investigated. Examples are ML-based clustering or grouping based on different characteristics or to-be-defined risk measures with respect to potential impact on production continuity. These measures could for example depend on lead times, manufacturing method, material specificness and potential of quality issues. Additionally, applying supplier grouping after feature selection on the individual suppliers can be investigated, since suppliers with similar important features might show similar behaviour.

The imposition of 'Material' during feature selection showed minor (positive) changes that could arise after imposing specific features. Since feature selection impacts the following development steps significantly, it can be valuable to compare different selection methods combined with the feature permutation score. Suggestions are the computationally expensive full factorial search, or the adaptation in which performance of different subsets of a fixed number of features is evaluated.
Additionally, the application of different estimators for the future permutation importance can be explored to investigate the impact on feature ranking and bias towards increased performance for similar estimators in the algorithm comparison step.

The results show the valuable addition of resampling during model training. The applied sampling factors for over-sampling and under-sampling are determined based on consultation with ML practitioners. However, different sampling factors could be more valuable without reducing the generalisation ability significantly. Additional research could focus on the effect of these factors and potentially define thresholds after comparing prediction performances over different (publicly available) datasets.

Additionally, the proposed parameter grid can be re-evaluated since different combinations resulted in overfitting. Considering a finer grid could result in higher performance of individual models as well, although it would increase computational time.

Future research directions regarding extensions of the proposed methodology can focus on the possibility to update or re-train developed models over time in the applicational context, to account for changes in supplier behaviour or ways of working. An example could be to include the newly acquired data in the previous dataset, while reducing weights (and therewith impact) of the older data. However, its potential and value within production environments needs to be assessed.

Finally, incorporating different data sources and applying the methodology on different production environments could improve the general prediction performance while showing its general applicability to introduce and explore ML for material-oriented disruption prediction in production.

6.3 Recommendations for Philips

For Philips, different suggestions and explorative directions are proposed, focussing more on the results to increase the feasibility and applicability for the considered and other production facilities. The results have shown the possibility of the methodology to extract supplier behaviour and develop successful prediction models. However, this has not been achieved for all the selected suppliers. The available data was not sufficient to indicate specific attributes which contribute positively or negatively towards the prediction performances. Since the methodology for the different supplier models is the same, it is expected that the prediction performance is influenced by external and internal factors such as communication between buyers and suppliers, shared demand forecasts or executed mitigating measures. Therefore, additional research is needed to focus on these influences and differences between supplier prediction performances.

Similarly, it is recommended to investigate and verify the idea that the importance of 'Due Week day' reflects inefficiencies in internal processes since it could limit overall performance.

The available data did not cover intermediate PO or shipment updates and changes which are necessary to incorporate prediction updates in the resulting models. Therefore, it is recommended to collect and incorporate more data related to order updates throughout the active period of a PO such that these prediction updates can be incorporated and provided. Examples of such order updates are shipment updates or updates received from the supplier.

It can also be interesting to shift the scope from manually created orders to orders following schedule line agreements (SA) or which are results of supplier managed inventory (VMI) policies. Expected is that less influences from Philips' buyers are present, which can result in stronger expressions of supplier behaviour leading to higher applicability of ML in disruption prediction.

The methodology can be applied to different production facilities as well. Differences in way-of-working and suppliers will result in different performances, which can lead to better insights regarding the general applicability and feasibility in Philips' production facilities.

Regarding data extraction, recommended is to extract the required data elements from the ERP system itself without the use of a third-party extraction/reporting tool as Every Angle. The initial use of such tools can be viable for quick extraction and analysis in the early stages. However, for deeper analysis, the implied logic and limitations of such tools can negatively influence results or impose additional work to reverse the undesired transformations or logic.

Bibliography

Alicke, K., Azcue, X., & Barriball, E. (2020). Supply-chain recovery in coronavirus times - plan for now and the future. Retrieved from https://www.mckinsey.com/business-functions/operations/our-insights/supply-chain-recovery-in-coronavirus-times-plan-for-now-and-the-future

Alicke, K., & Balaji, I. (2013). *Next generation supply chain: Supply chain 2020* - 2013:

Alicke, K., Rachot, J., & Seyfert, A. (2016). *Supply Chain 4.0 - the next-generation digital supply chain* - 2016:

Atallah, M. J., Elmongui, H. G., Deshpande, V., & Schwarz, L. B. (2003, 24-27 June 2003). *Secure supply-chain protocols.* Paper presented at the EEE International Conference on E-Commerce, 2003. CEC 2003.

Baryannis, G., Dani, S., & Antoniou, G. (2019). Predicting supply chain risks using machine learning: The trade-off between performance and interpretability. *Future Generation Computer Systems, 101*, 993-1004. doi:10.1016/j.future.2019.07.059

Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explorations, 6*(1), 20-29.

Behzadi, G., O'Sullivan, M. J., Olsen, T. L., Scrimgeour, F., & Zhang, A. (2017). Robust and resilient strategies for managing supply disruptions in an agribusiness supply chain. *International Journal of Production Economics, 191*, 207-220. doi:10.1016/j.ijpe.2017.06.018

Bishop, C. M. (2006). *Pattern recognition and machine learning*: Springer.

Branco, P., Torgo, L., & Ribeiro, R. P. (2016). A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys, 49*(2). doi:10.1145/2907070

Breiman, L. (2001). Random Forests. *Machine Learning, 45*(1), 5-32. doi:10.1023/A:1010933404324

Brintrup, A., Pak, J., Ratiney, D., Pearce, T., Wichmann, P., Woodall, P., & McFarlane, D. (2020). Supply chain data analytics for predicting supplier disruptions: a case study in complex asset manufacturing. *International Journal of Production Research, 58*(11), 3330-3341. doi:10.1080/00207543.2019.1685705

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321-357.

Chen, T., & Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System*. Paper presented at the Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, California, USA. https://doi.org/10.1145/2939672.2939785

Cheng, S. Y., Chen, Y., Khosla, D., & Kim, K. (2011) Optimal multiclass classifier threshold estimation with particle swarm optimization for visual object recognition. In & C. V. L. Unr, I. Desert Research, L. Berkeley, Nasa, & Intel (Vol. Ed.)*: Vol. 6939 LNCS. 7th International Symposium on Visual Computing, ISVC 2011* (pp. 536-544). Las Vegas, NV.

Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics, 21*(1). doi:10.1186/s12864-019-6413-7

De Santis, R. B., De Aguiar, E. P., & Goliatt, L. (2018). *Predicting material backorders in inventory management using machine learning.* Paper presented at the 2017 IEEE Latin American Conference on Computational Intelligence, LA-CCI 2017.

Devi, D., Biswas, S. K., & Purkayastha, B. (2019). Learning in presence of class imbalance and class overlapping by using one-class SVM and undersampling technique. *Connection Science, 31*(2), 105-142. doi:10.1080/09540091.2018.1560394

. Enterprise Resource Planning (ERP). (2000). In P. M. Swamidass (Ed.), *Encyclopedia of Production and Manufacturing Management* (pp. 184-187). Boston, MA: Springer US.

Fagundes, M. V. C., Teles, E. O., Vieira de Melo, S. A. B., & Freires, F. G. M. (2020). Decision-making models and support systems for supply chain risk: literature mapping and future research agenda. *European Research on Management and Business Economics, 26*(2), 63-70. doi:10.1016/j.iedeen.2020.02.001

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2012). A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews, 42*(4), 463-484. doi:10.1109/TSMCC.2011.2161285

Gorodkin, J. (2004). Comparing two K-category assignments by a K-category correlation coefficient. *Computational Biology and Chemistry, 28*(5-6), 367-374. doi:10.1016/j.compbiolchem.2004.09.006

Hajek, P., & Abedin, M. Z. (2020). A Profit Function-Maximizing Inventory Backorder Prediction System Using Big Data Analytics. *IEEE Access, 8*, 58982-58994. doi:10.1109/ACCESS.2020.2983118

Han, Y., Chong, W. K., & Li, D. (2020). A systematic literature review of the capabilities and performance metrics of supply chain resilience. *International Journal of Production Research*, 1-26. doi:10.1080/00207543.2020.1785034

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., . . . Oliphant, T. E. (2020). Array programming with NumPy. *Nature, 585*(7825), 357-362. doi:10.1038/s41586-020-2649-2

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). *ADASYN: Adaptive synthetic sampling approach for imbalanced learning.* Paper presented at the 2008 International Joint Conference on Neural Networks, IJCNN 2008, Hong Kong.

He, M., Ji, H., Wang, Q., Ren, C., & Lougee, R. (2015). *Big data fueled process management of supply risks: Sensing, prediction, evaluation and mitigation.* Paper presented at the 2014 Winter Simulation Conference, WSC 2014.

Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering, 9*(3), 90-95. doi:10.1109/MCSE.2007.55

Kaggle Inc. (2021). Kaggle Competitions. Retrieved from https://www.kaggle.com/competitions

Kotu, V., & Deshpande, B. (2015). Chapter 12 - Feature Selection. In V. Kotu & B. Deshpande (Eds.), *Predictive Analytics and Data Mining* (pp. 347-370). Boston: Morgan Kaufmann.

Lemaître, G., Nogueira, F., & Aridas, C. K. (2017). Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research, 18*, 1-5.

Lin, W. C., Tsai, C. F., Hu, Y. H., & Jhang, J. S. (2017). Clustering-based undersampling in class-imbalanced data. *Information Sciences, 409-410*, 17-26. doi:10.1016/j.ins.2017.05.008

Liu, H. (2010). Feature Selection. In C. Sammut & G. I. Webb (Eds.), *Encyclopedia of Machine Learning* (pp. 402-406). Boston, MA: Springer US.

Liu, X. Y., Wu, J., & Zhou, Z. H. (2009). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 39*(2), 539-550. doi:10.1109/TSMCB.2008.2007853

López, V., Fernández, A., García, S., Palade, V., & Herrera, F. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences, 250*, 113-141. doi:10.1016/j.ins.2013.07.007

McKinney, W. (2010). *Data Structures for Statistical Computing in Python.* Paper presented at the Proceedings of the 9th Python in Science Conference.

Nguyen, T., Zhou, L., Spiegler, V., Ieromonachou, P., & Lin, Y. (2018). Big data analytics in supply chain management: A state-of-the-art literature review. *Computers and Operations Research, 98*, 254-264. doi:10.1016/j.cor.2017.07.004

Ni, D., Xiao, Z., & Lim, M. K. (2020). A systematic review of the research trends of machine learning in supply chain management. *International Journal of Machine Learning and Cybernetics, 11*(7), 1463-1482. doi:10.1007/s13042-019-01050-0

Ofek, N., Rokach, L., Stern, R., & Shabtai, A. (2017). Fast-CBUS: A fast clustering-based undersampling method for addressing the class imbalance problem. *Neurocomputing, 243*, 88-102. doi:10.1016/j.neucom.2017.03.011

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12*, 2825-2830.

Powers, D. M. (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*.

Scikit-Learn. (2021). RBF SVM parameters - scikit-learn 0.24.1 documentation. Retrieved from https://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html

Singh, C. S., Soni, G., & Badhotiya, G. K. (2019). Performance indicators for supply chain resilience: review and conceptual framework. *Journal of Industrial Engineering International, 15*, 105-117. doi:10.1007/s40092-019-00322-2

Sridharan, V., & La Forge, R. L. (2000). Materials Requirements Planning (MRP). In P. M. Swamidass (Ed.), *Encyclopedia of Production and Manufacturing Management* (pp. 466-470). Boston, MA: Springer US.

Ting, K. M. (2010). Confusion Matrix. In C. Sammut & G. I. Webb (Eds.), *Encyclopedia of Machine Learning* (pp. 209-209). Boston, MA: Springer US.

United Nations Statistics Division. (2021). UNSD - Methodology: Standard country or area codes for statistical use (M49). Retrieved from https://unstats.un.org/unsd/methodology/m49/

Wang, G., Gunasekaran, A., Ngai, E. W. T., & Papadopoulos, T. (2016). Big data analytics in logistics and supply chain management: Certain investigations for research and applications. *International Journal of Production Economics, 176*, 98-110. doi:10.1016/j.ijpe.2016.03.014

Webb, G. I. (2010). Overfitting. In C. Sammut & G. I. Webb (Eds.), *Encyclopedia of Machine Learning* (pp. 744-744). Boston, MA: Springer US.

Weiss, G., McCarthy, K., & Zabar, B. (2007). *Cost-Sensitive Learning vs. Sampling: Which is Best for Handling Unbalanced Classes with Unequal Error Costs?* Paper presented at the DMIN.

XGBoost. (2021). XGBoost Parameters - xgboost 1.4.0-SNAPSHOT documentation. Retrieved from https://xgboost.readthedocs.io/en/latest/parameter.html#parameters-for-tree-booster

Youden, W. J. (1950). Index for rating diagnostic tests. *Cancer, 3*(1), 32-35. doi:https://doi.org/10.1002/1097-0142(1950)3:1<32::AID-CNCR2820030106>3.0.CO;2-3

Zhao, J., Ji, M., & Feng, B. (2020). Smarter supply chain: a literature review and practices. *Journal of Data, Information and Management, 2*(2), 95-110. doi:10.1007/s42488-020-00025-z

Appendix A: Considered algorithm parameters

The parameters selected for the proposed grid searches are the following:

- *Regularisation* - C:
    - Logistic Regression: the value of C represents the amount of regularisation (variance reduction) applied in the calculation of the error function. The smaller the value for C, the more regularisation is applied and the simpler the decision function gets, reducing the potential for overfitting.
    - Support Vector Machine: "trade-off of correct classification of training examples against maximization of the decision function's margin" (Scikit-Learn, 2021). It serves as a regularisation parameter, in which smaller values for C aim for a larger margin leading to a simpler decision function with less potential for overfitting.
- *Gamma* - γ:
    - Support Vector Machine: the value of γ represents the effect of a single sample on the final model. The γ-value could be described as the inverse radius of influence for samples, with smaller values meaning a larger radius.
    - eXtreme Gradient Boost: the value of γ describes the minimum loss reduction in order to split a current node in the decision tree (XGBoost, 2021). Increasing the value for γ results in a more conservative model, reducing the potential for overfitting.
- *Class-weight*: a possibility to adjust the weight of a sample given the number of occurrences of the specific class. This parameter enables a (simple) implementation of cost-sensitive learning since different misclassification costs can be introduced and used.
- *Maximum tree depth*: the set value represents the maximum depth of the decision tree being created. Smaller values lead to smaller more conservative trees reducing the potential of overfitting.
- *Maximum leaf nodes*: the value describes the maximum number of final (leaf) nodes present in the created decision tree. Smaller values lead to a smaller tree with less potential for overfitting.
- *Minimum child weight*:
    - Decision Tree and Random Forest: the value represents the minimum number required samples to form a tree node. Higher values reduce the number of nodes resulting in more conservative trees, while simultaneously reducing the potential of overfitting.
    - eXtreme Gradient Boost: the value represents the "minimum sum of instance weight (hessian) needed in a child" (XGBoost, 2021). Therefore, it does not set the minimum required samples, but the minimum sum, which results in a similar behaviour: higher values reduce the number of nodes resulting in more conservative trees, while simultaneously reducing the potential of overfitting.
- *Number of estimators*: the value represents the number of estimators considered in the bagging or boosting process.

Appendix B: Example dataset slice

Table B.1: Example entries and format of obtained dataset

Doc. Type	Order number	Created On	Stat.-Rel. Del. Date	Confirmed Delivery Date	Order due date	Date of goods receipt	Material	Material Description	Lead time
F\NB	1234567890/1/1	Apr/01/2018	Jul/01/2018		Jul/01/2018	Jul/07/2018	450000000012	Ex. Material 1	95 days
F\NB	1234567891/1/1	Apr/04/2018	May/01/2018	May/01/2018	May/01/2018	May/01/2018	450000002011	Ex. Material 2	21 days
F\NB	1234567891/1/2	Apr/04/2018	Jun/01/2018	May/28/2018	Jun/01/2018	May/28/2018	450000002011	Ex. Material 2	21 days
F\NB	1234567891/2/1	Apr/04/2018	Sep/01/2018		Sep/01/2018	Sep/01/2018	450000000012	Ex. Material 1	95 days
F\NB	1234567892/1/1	May/06/2018	Jun/01/2018	Jun/01/2018	Jun/01/2018	Jun/07/2018	450000100012	Ex. Material 3	30 days
…	…	…	…	…	…	…	…	…	…

Order Unit	Quantity	Unit Price	Safety time	ABC indicator	Lot size	Min Lot Size	Max Lot size	Fixed lot size	Supplier	Supplier Description	Supplier Country	SNC Relevancy	MRP Type
PC	2	1000	10	G	4W	1	0	0	1234567	Ex. Supplier 1	CN (China)	Not SNC relevant	PD
PC	100	50	0	A	2W	10	0	0	1220034	Ex. Supplier 2	NL (Netherlands)	Not SNC relevant	Z7
PC	100	50	0	A	2W	10	0	0	1220034	Ex. Supplier 2	NL (Netherlands)	Not SNC relevant	Z7
PC	50	200	10	G	4W	1	0	0	1234567	Ex. Supplier 1	CN (China)	Not SNC relevant	Z7
PC	10	1500	10	B	4W	0	0	10	1220034	Ex. Supplier 2	NL (Netherlands)	Not SNC relevant	PD
…	…	…	…	…	…	…	…	…	…	…	…	…	…

Appendix C: List of (engineered) features

Table C.1: Overview of all considered (engineered) features

Feature		Format	Description
Order			
	Day	Integer	Day of the month of the creation or due date
	Week day	Integer	Week day of the creation or due date
Creation/	Week	Integer	Week of the creation or due date
Due	Month	Integer	Month of the creation or due date
	Season	Integer	Season of the creation or due date
	Year	Integer	Year of the creation or due date
Supplier		Category	Supplier number as categorical variable
Material		Category	Material number as categorical variable
Quantity		Float	Quantity of ordered material
Schedule Line value		Float	Monetary value of the specific schedule line
Diff. order-due		Integer	Difference in days between order and due date
Order/Lead time ratio		Float	Ratio between time between order and due date normalised by the material's lead time
Frequency indicator		Binary	Indicator whether the order is placed sooner with respect to lot size or last quarter's median order frequency
Quantity indicator		Binary	Indicator whether the order quantity is higher than the fixed lot size or last quarter's median order quantity
Supplier – material			
Lead time		Integer	Contracted lead time in number of days
Price		Float	Price of the ordered material per standardised unit
Safety time		Integer	Number of workdays the material order is moved forward
ABC indicator		Category	Material class according to ABC classification (quantity-spend relation)
SNC relevancy		Text	Category indicating maturity level of information exchange
Lot size type		Text	Category indicating frequency or time dependent replenishment
Unique materials		Integer	Number of unique materials supplied by supplier
Agility (Brintrup et al., 2020)		Integer	Proxy for ability of supplier to handle order variance

Dynamic 'environment'

CLIP score	Float	Supplier performance score of the last preceding recorded month, as calculated within Philips' GSRS
CLIP score 3 months	Float	Average of CLIP scores from the last three months
Outstanding POs	Integer	Number of outstanding standard orders for the material
Outstanding PO items	Integer	Number of outstanding PO items for the material
Overdue POs	Integer	Number of overdue standard orders for the material
Overdue PO items	Integer	Number of overdue PO items for the material
Outstanding quantity	Float	Outstanding quantity for the material
Overdue quantity	Float	Overdue quantity for the material
Weighted overdue quantity	Float	Quantity overdue weighted by the amount of days overdue for the material
Sup. outstanding POs	Integer	Number of outstanding standard orders on supplier level
Sup. outstanding POs (SA)	Integer	Number of outstanding schedule line agreements for the material
Sup. outstanding PO items	Integer	Number of outstanding PO items on supplier level
Sup. overdue POs	Integer	Number of overdue standard orders on supplier level
Sup. overdue PO items	Integer	Number of overdue PO items on supplier level
Sup. outstanding quantity	Float	Quantity of outstanding material on supplier level
Sup. overdue quantity	Float	Quantity of overdue material on supplier level
Sup. weighted overdue quantity	Float	Quantity overdue weighted by the amount of days overdue on supplier level

Appendix D: Performance distributions of the selected suppliers

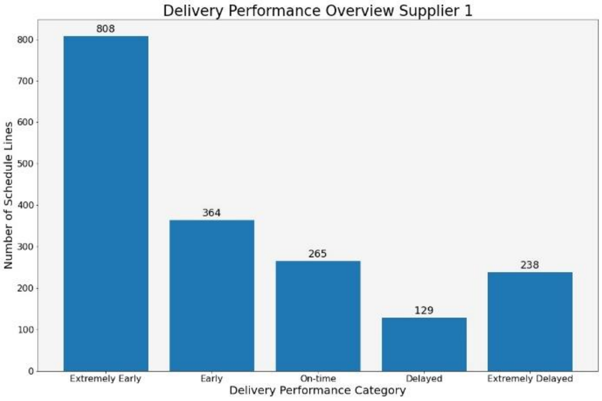

Figure D.1: Delivery performance supplier 1

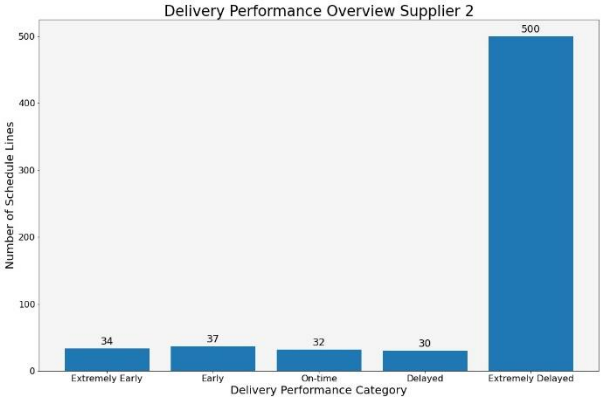

Figure D.2: Delivery performance supplier 2

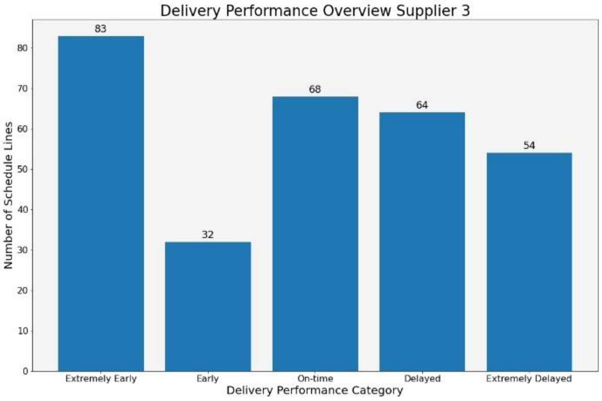

Figure D.3: Delivery performance supplier 3

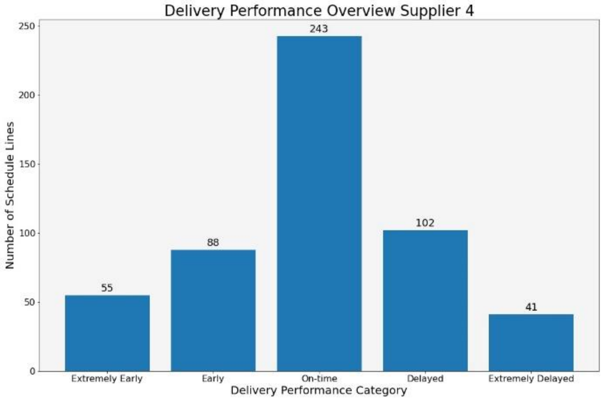

Figure D.4: Delivery performance supplier 4

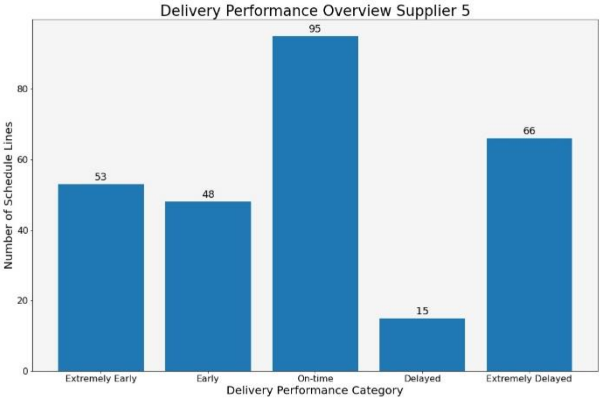

Figure D.5: Delivery performance supplier 5

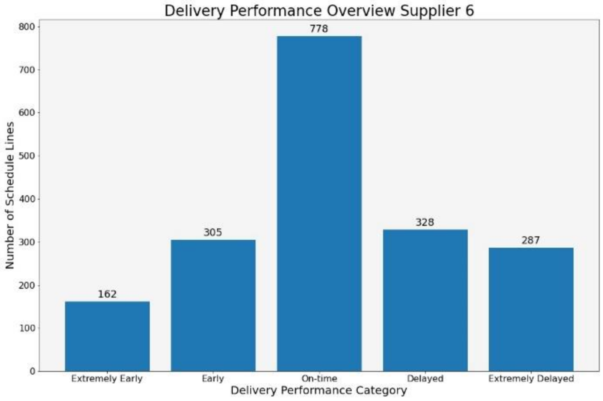

Figure D.6: Delivery performance supplier 6

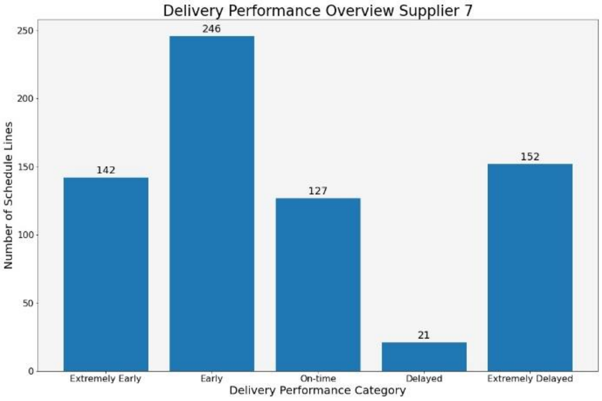

Figure D.7: Delivery performance supplier 7

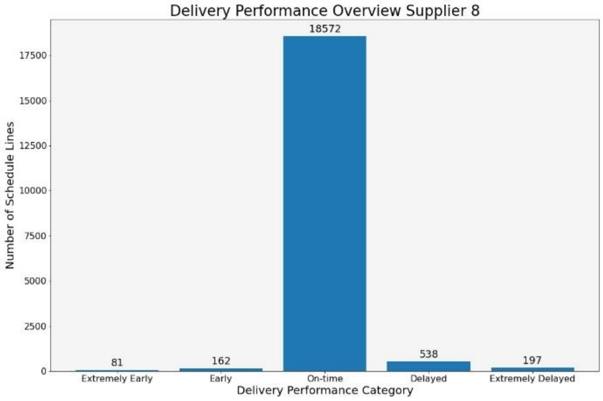

Figure D.8: Delivery performance supplier 8

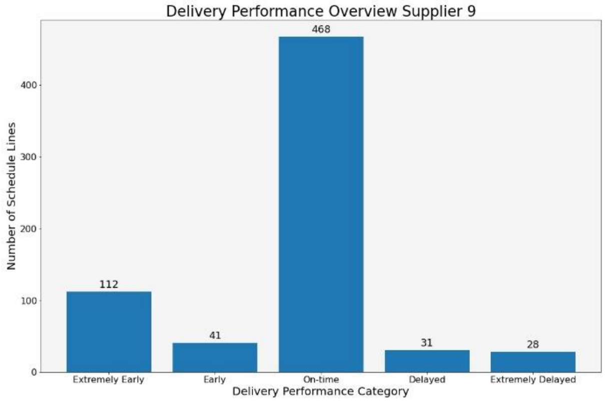

Figure D.9: Delivery performance supplier 9

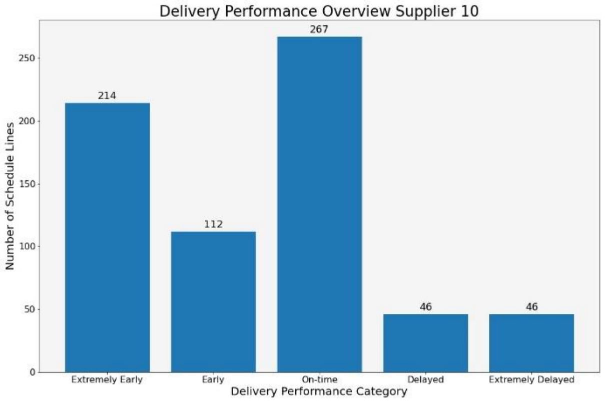

Figure D.10: Delivery performance supplier 10

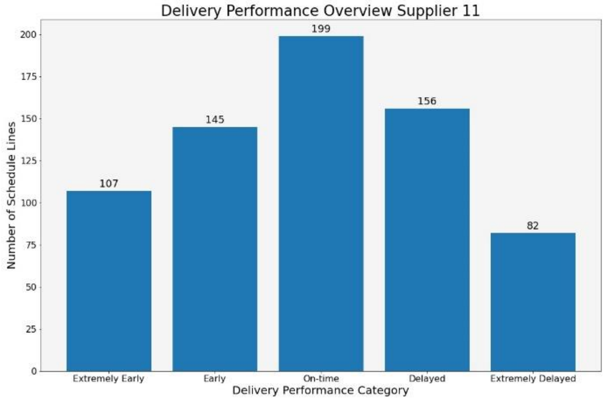

Figure D.11: Delivery performance supplier 11

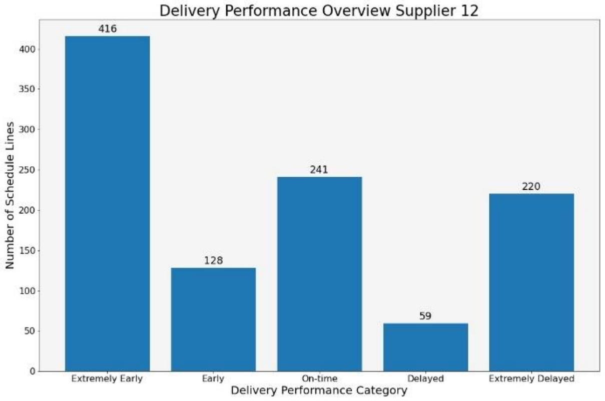

Figure D.12: Delivery performance supplier 12

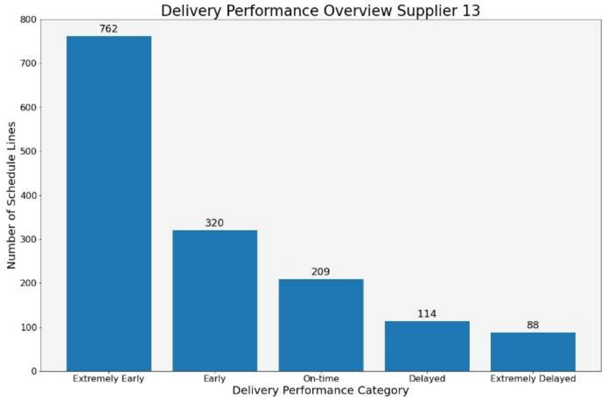

Figure D.13: Delivery performance supplier 13

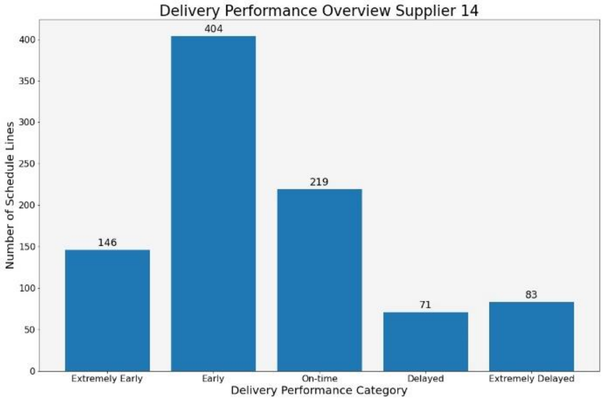

Figure D.14: Delivery performance supplier 14

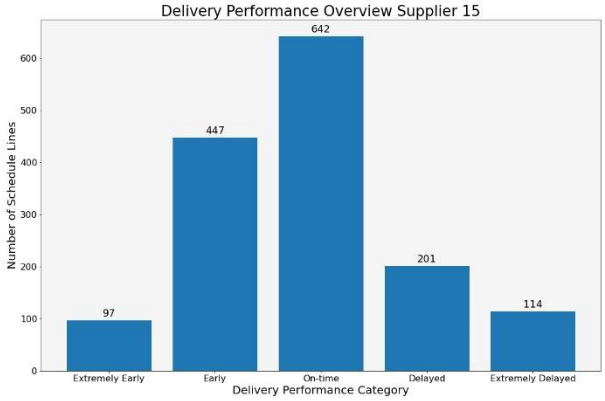

Figure D.15: Delivery performance supplier 15

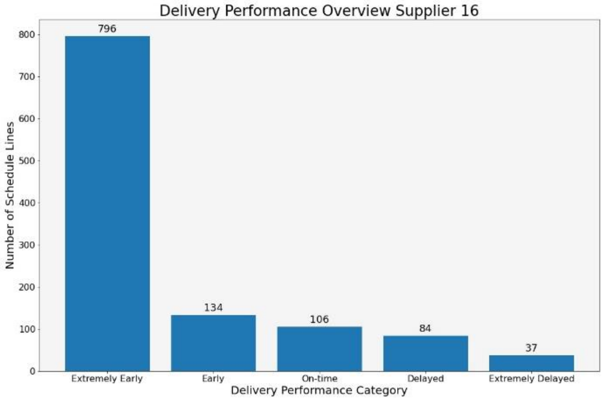

Figure D.16: Delivery performance supplier 16

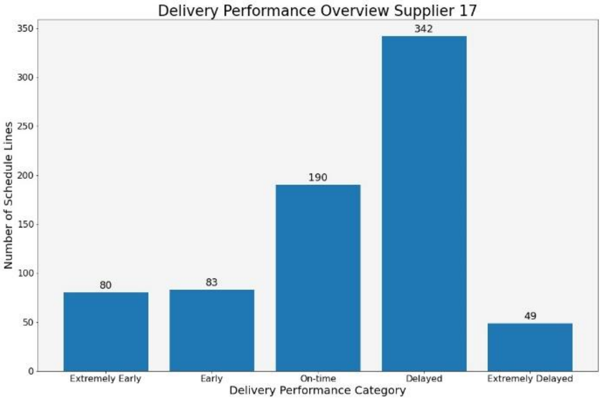

Figure D.17: Delivery performance supplier 17

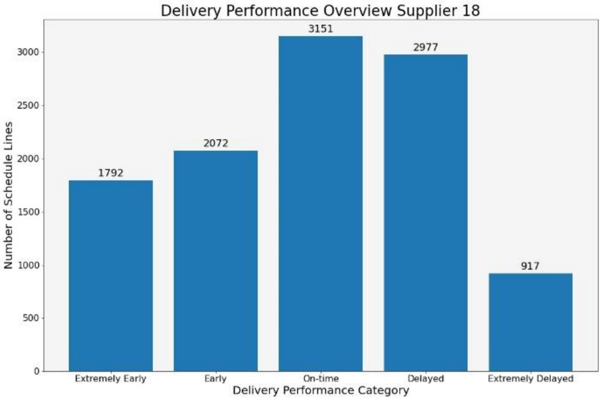

Figure D.18: Delivery performance supplier 18

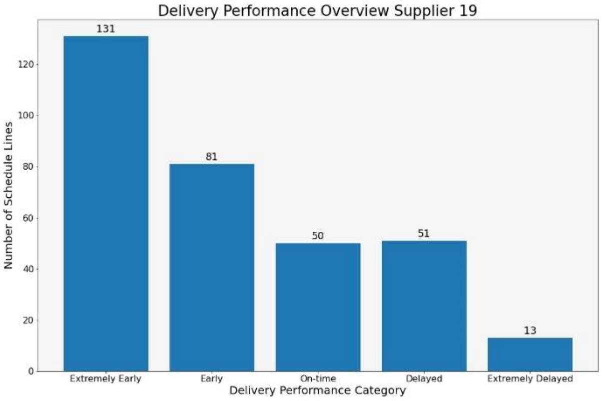

Figure D.19: Delivery performance supplier 19

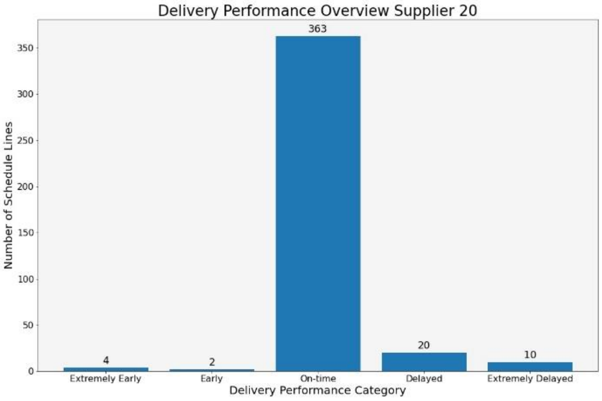

Figure D.20: Delivery performance supplier 20

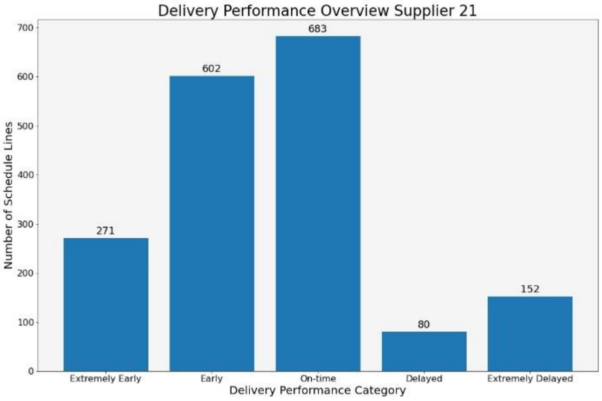

Figure D.21: Delivery performance supplier 21

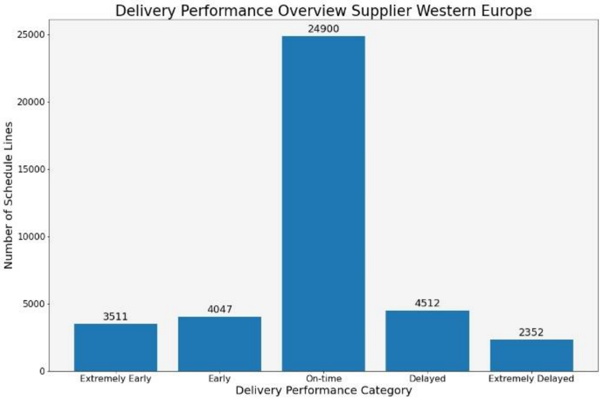

Figure D.22: Delivery performance suppliers Western Europe

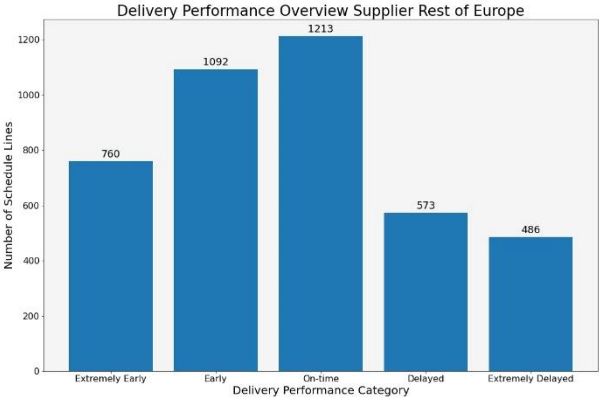

Figure D.23: Delivery performance suppliers Rest of Europe

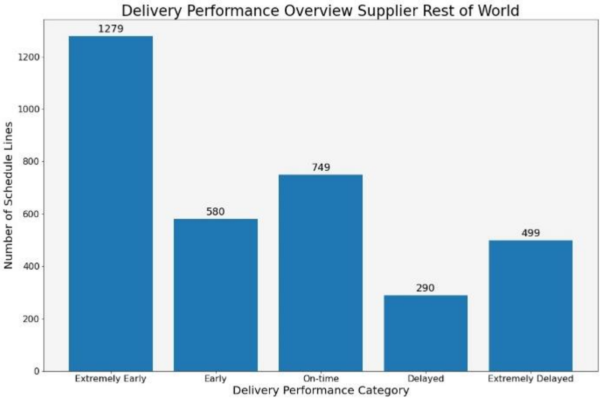

Figure D.24: Delivery performance suppliers Rest of World

Appendix E: Selected features per supplier (group)

Table E.1: Selected features and their ranking for each supplier (group) in the binary problem formulation.

Supplier	Selected features + ranking	Supplier	Selected features + ranking
1	- Due Year - Sup. overdue quantity - Material - Schedule Line value - Price - Order/Lead time ratio - Sup. outstanding POs - Due Week - CLIP score - Outstanding quantity	2	- Diff. order-due - Sup. outstanding quantity - Sup. weighted overdue quantity - Due Week - Outstanding quantity - Sup. overdue POs - CLIP score 3 months - Outstanding POs - CLIP score - Creation Day - Creation Week
3	- Due Week - Diff. order-due - Material - Sup. outstanding PO Items - Schedule Line value - Sup. outstanding quantity - Quantity - Due Day - CLIP score 3 months - Order/Lead time ratio - Due Week day	4	- Due Week - Due Day - Order/Lead time ratio - Due Week day - Sup. outstanding quantity - ABC indicator - CLIP score 3 months - CLIP score - Sup. outstanding POs
5	- Material - Sup. overdue POs - Order/Lead time ratio - Due Week - Outstanding PO items - Schedule Line value - Creation Day - Quantity - Price - Due Week day - Due Day	6	- Order/Lead time ratio - Material - Creation Day - Due Day - Due Week - CLIP score 3 months - CLIP score - Quantity - Sup. outstanding quantity - Sup. weighted overdue quantity - Price
7	- Sup. overdue quantity - Material - Creation Week - Schedule Line value - Sup. outstanding POs (SA) - Due Week - Creation Day - Diff. order-due - Lot size type - Due Day - CLIP score 3 months	8	- Due Day - Sup. weighted overdue quantity - CLIP score 3 months - Material - Sup. outstanding PO Items - Sup. outstanding POs (SA) - Diff. order-due - Due Week - Sup. overdue PO items - Creation Day - Creation Week Day - Creation Week - Due Week day - Lot size type

Supplier	Selected features + ranking	Supplier	Selected features + ranking
9	- Safety time - Material - Creation Week - Schedule Line value	10	- Due Week day - Material - Sup. outstanding quantity - Order/Lead time ratio - Price - Creation Week - Due Day - Quantity - Due Week - Sup. outstanding POs - Schedule Line value - CLIP score - CLIP score 3 months
11	- Price - Material - Due Week day - Order/Lead time ratio - CLIP score - Sup. outstanding POs (SA) - Quantity	12	- Due Year - Material - Due Day - Price - Sup. outstanding quantity - Diff. order-due - Due Week - Creation Day - Schedule Line value - Outstanding quantity - Order/Lead time ratio - Sup. outstanding PO Items - Due Week day
13	- Creation Week - Material - Due Week day - Order/Lead time ratio - Due Week - Sup. outstanding quantity	14	- Order/Lead time ratio - Material - Creation Day - Due Day - Due Week - CLIP score 3 months - CLIP score - Quantity - Sup. outstanding quantity - Sup. weighted overdue quantity - Price
15	- Due Week - Material - Diff. order-due - Schedule Line value - CLIP score 3 months - Creation Day - Due Day - ABC indicator - Order/Lead time ratio - Sup. outstanding POs - Creation Week - CLIP score	16	- Order/Lead time ratio - Due Day - Sup. outstanding quantity - Creation Week - Creation Week Day - Creation Day
17	- Due Week day - CLIP score 3 months - Due Week - Sup. outstanding POs - Material - Due Day	18	- Due Week - CLIP score 3 months - Due Week day - Sup. outstanding PO Items - Sup. outstanding POs

Supplier	Selected features + ranking	Supplier	Selected features + ranking
19	- Due Day - Due Week - Sup. outstanding quantity - Schedule Line value - Due Week day - Creation Week - Creation Day - CLIP score 3 months	20	- Due Week - Due Week day - Creation Day
21	- Material - Sup. outstanding POs - Due Day - Outstanding POs - Lot size type - Due Week - Price - Order/Lead time ratio - Creation Week - CLIP score - Sup. outstanding quantity - Schedule Line value - Sup. outstanding POs (SA) - CLIP score 3 months - Sup. overdue quantity - Diff. order-due - Outstanding quantity	Western Europe (group)	- Supplier - Material - Sup. overdue quantity - ABC indicator - Order/Lead time ratio - Sup. overdue PO items - Due Week day - Due Year - Sup. overdue POs
Rest of Europe (group)	- Supplier - Material - Agility - Sup. outstanding quantity - Price - CLIP score - Sup. weighted overdue quantity - Due Week - Creation Week - Due Week day - Schedule Line value	Rest of World (group)	- Sup. outstanding quantity - CLIP score - Material - Due Week - Sup. overdue quantity - Schedule Line value - Creation Week - Diff. order-due - Due Day - Agility - Outstanding quantity - Order/Lead time ratio - CLIP score 3 months - Safety time

Table E.2: Selected features and their ranking for each supplier (group) in the multiclass problem formulation.

Supplier	Selected features + ranking	Supplier	Selected features + ranking
1	- Material - Outstanding quantity - CLIP score - Diff. order-due - Due Week day - Due Week - CLIP score 3 months - Outstanding POs - SL Value - Quantity	2	- Material - Diff. order-due - Due Day - Sup. weighted overdue quantity - Due Week day - Due Week
3	- Order/Lead time ratio - Material - Sup. outstanding quantity - Due Week day - CLIP score 3 months - Quantity - SL Value	4	- Due Week - Due Week day - Due Day - CLIP score 3 months - Sup. outstanding POs
5	- Material - Due Week - Due Week day - Sup. weighted overdue quantity - Order/Lead time ratio - SL Value - Frequency indicator - Creation Week - Outstanding POs	6	- Due Week day - Diff. order-due - Material - Sup. outstanding POs - Sup. outstanding quantity - Due Week - Sup. overdue quantity - CLIP score 3 months - SL Value - Due Day - Order/Lead time ratio
7	- Due Week day - Material - CLIP score 3 months - Due Day - Due Week - Sup. outstanding POs - Order/Lead time ratio	8	- Sup. weighted overdue quantity - Creation Week day - Due Day - Due Week - Outstanding POs - Sup. outstanding POs (SA) - Sup. outstanding PO items - Material - Lead time - Creation Week - Creation Day - Sup. outstanding POs - Price - Sup. overdue PO items - ABC indicator - Frequency indicator - CLIP score

Supplier	Selected features + ranking	Supplier	Selected features + ranking
9	- Safety time - Material - Diff. order-due - Due Week day - Lot size type - Sup. outstanding quantity	10	- Due Week day - Material - Sup. outstanding quantity - Due Week - Diff. order-due - Sup. outstanding POs - SL Value - CLIP score 3 months
11	- Material - Due Week day - CLIP score - Price - Due Week - Sup. outstanding POs - Sup. outstanding quantity - SL Value - Creation Week - Sup. overdue quantity - Diff. order-due - Creation Week day - Creation Day	12	- Due Year - Due Week day - Material - Sup. outstanding POs - Due Day - Due Week - SL Value - Order/Lead time ratio - CLIP score 3 months - Sup. outstanding quantity - Diff. order-due - Sup. overdue quantity - Sup. overdue POs - ABC indicator - Outstanding quantity
13	- Material - Due Week day - Due Week - Sup. outstanding quantity - Quantity - Order/Lead time ratio - Due Day - Creation Week - Creation Day - SL Value - Price - Diff. order-due - Outstanding quantity - CLIP score - Lot size type - Outstanding POs - Safety time	14	- Due Day - Material - Due Week day - Diff. order-due - CLIP score - Creation Week - Creation Day - ABC indicator - CLIP score 3 months - Due Week - Order/Lead time ratio - Outstanding POs
15	- Sup. outstanding PO items - Material - Sup. outstanding quantity - Lot size type - CLIP score 3 months - Due Day - Creation Week - Creation Day - Order/Lead time ratio - CLIP score - Diff. order-due - Due Week day - Due Week - Sup. weighted overdue quantity - Sup. outstanding POs	16	- Order/Lead time ratio - Creation Week day - Due Day - Creation Week - Sup. outstanding quantity - Material - Due Week - Sup. outstanding PO items - Quantity - Creation Day - Lot size type - Sup. overdue POs

Supplier	Selected features + ranking	Supplier	Selected features + ranking
17	- Due Week day - CLIP score 3 months - Material - Due Week - Sup. outstanding quantity - SL Value - Due Day - Sup. outstanding POs - CLIP score - Diff. order-due - Creation Week - Quantity - Creation Day	18	- Due Week day - Order/Lead time ratio - Sup. outstanding POs - Material - Due Week - CLIP score 3 months - ABC indicator - Creation Day - Sup. outstanding quantity - CLIP score - Due Day - Sup. outstanding PO items - Lead time - Quantity
19	- Creation Week - Sup. outstanding quantity - Material - Due Day - Due Week - Diff. order-due - Creation Day - CLIP score 3 months - SL Value	20	-
21	- Due Week day - Material - Due Week - Sup. outstanding POs - Diff. order-due - Order/Lead time ratio - Lot size type - Due Day - Sup. outstanding quantity	Western Europe (group)	- Supplier - Material - Due Week day - Diff. order-due - Order/Lead time ratio - Due Week
Rest of Europe (group)	- Due Week day - Material - Agility - Due Week - Order/Lead time ratio - CLIP score - Diff. order-due - Creation Week	Rest of World (group)	- Material - Due Week day - Sup. outstanding POs - Outstanding POs - Sup. outstanding PO items - Diff. order-due - Agility - Sup. outstanding quantity - Due Week - Due Day - CLIP score 3 months - SL Value - Creation Week - Outstanding quantity - CLIP score - Order/Lead time ratio

Appendix F: Overviews of selected features per region

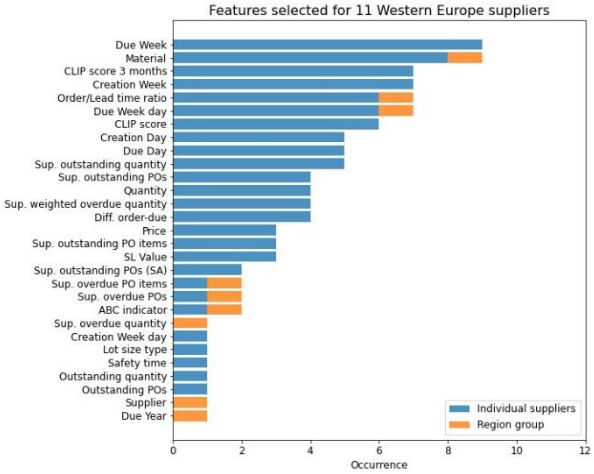

Figure F.1: Overview of features selected in Western Europe for the binary problem formulation

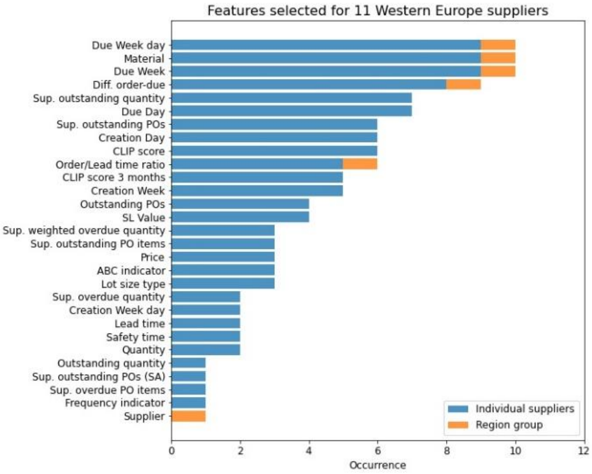

Figure F.2: Overview of features selected in Western Europe for the multiclass problem formulation

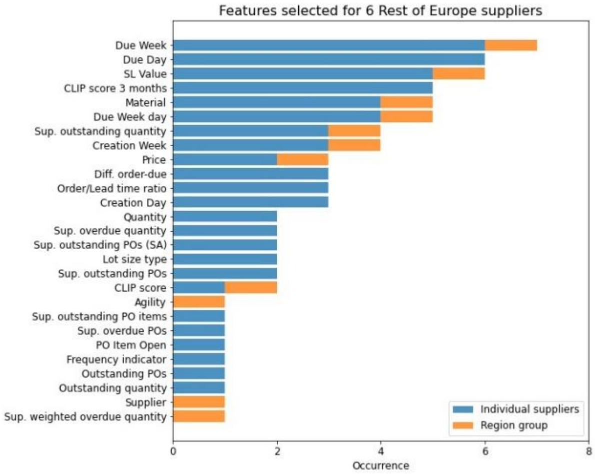

Figure F.3: Overview of features selected in Rest of Europe for the binary problem formulation

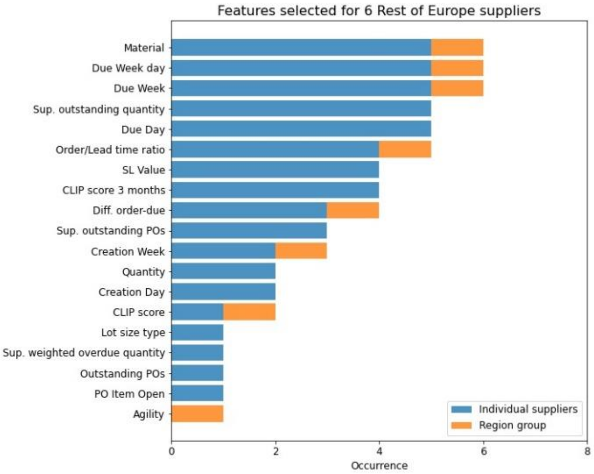

Figure F.4: Overview of features selected in Rest of Europe for the multiclass problem formulation

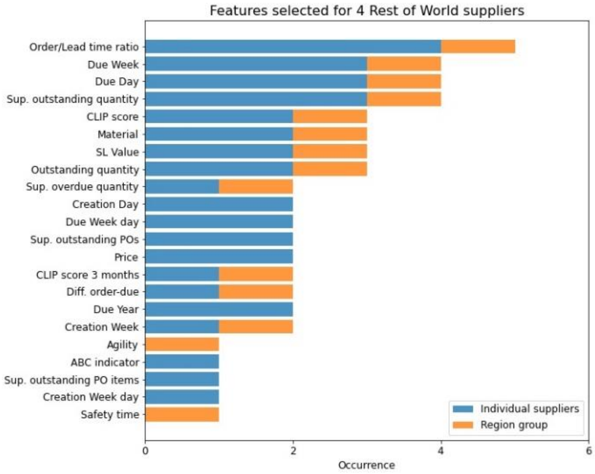

Figure F.5: Overview of features selected in Rest of World for the binary problem formulation

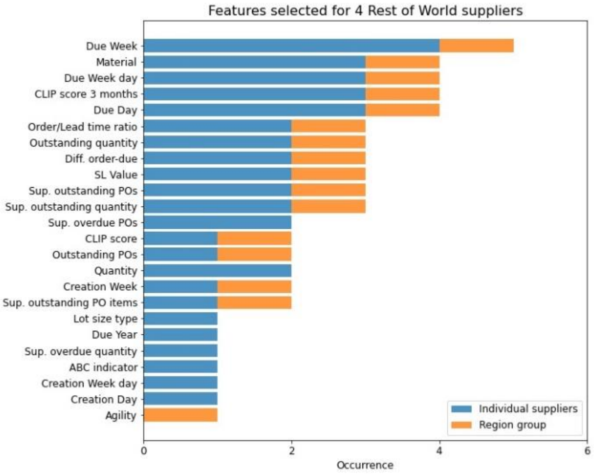

Figure F.6: Overview of features selected in Rest of World for the multiclass problem formulation

Appendix G: Research paper

Supplier Disruption Prediction using Machine Learning in Production Environments

H. J. A. de Krom, B. Wiegmans, M. B. Duinkerken, M. J. J. Hutten and L. A. Tavasszy

***Abstract*— Recent developments in supply chains and supply chain management (SCM) lead to increased complexity and vulnerability of supply chain operations. Therefore, it is needed to better anticipate and prepare for or prevent disruptions from occurring. Increasing interest for the application of machine learning in supply chain (risk) management has been observed, but applicational studies are lacking. Therefore, in this paper a generalised methodology is proposed to introduce and apply machine learning to provide insights in supply chain operations and predictive analytics for supply chain (risk) management. The methodology is applied to a case study from a high-tech medical imaging manufacturer with the focus on predicting delivery performance of supplier deliveries by means of classification. Experiments show that the application of the methodology led to successful model development resulting in binary and multiclass classification models obtaining Matthew's Correlation Coefficient (MCC) scores of 0.75 accompanied with 90% accuracy, 87% precision and 84% recall and MCC scores of 0.66 accompanied with 74% accuracy, 74% macro-precision and 74% macro-recall, respectively.**

***Index Terms*—machine learning, risk prediction, manufacturing, material management, supplier disruptions, supply chain management**

I. Introduction

Developments such as the increase in data collection resulting from the fourth industrial revolution (Industry 4.0), the increased demand in new emerged rural areas, changing labour demographics and the interest in adopting the concept of circular economy into supply chains and Supply Chain Management (SCM) [1, 2] have taken place within supply chains and SCM in the last years. These developments lead to an increase in supply chain size, global spread and interconnections between other supply chains, leading to more complex, vulnerable and uncertain supply chains [3], which in its turn could lead to undesired losses in shareholder value, sales, customer satisfaction and reputation [4].

This increase in complexity and vulnerability urges an increase in monitoring of supply chain performance to better anticipate and prepare for or even prevent disruptions from occurring [5]. Especially in a production-oriented supply chain, disruptions in the material flow could influence downstream supply chain performance and continuity significantly.

Considering the increasing interest in 'Big Data' and 'Machine Learning', combined with the expected value of predictive analytics in supply chain (risk) management [6-8], the question arises to what extent machine learning can assist in reducing supply chain risks and mitigating its negative effects. Besides conceptual studies and the expressed expected value, only limited applicational studies applying machine learning in a SCM context are observed. Most of those studies focus on demand forecasts or predicting lead time [7], although recently interest for predicting material-related supplier disruptions has been observed.

To contribute towards this interest and lack of applicational studies, this research investigated the possibility of applying machine learning in production environments to provide additional insights in supply chain operations and predictive analytics for supply chain (risk) management. A generalised methodology is proposed focussing on a systematic introduction and model development of machine learning-based prediction models whereafter it is applied in a case study of a high-tech medical imaging manufacturer. In this case study, risk is expressed as the occurrence of delayed supplier deliveries and its prediction is achieved through classification. Various machine learning algorithms and sampling techniques are considered in a binary and multiclass formulation to predict whether future deliveries will be delayed. The contributions of this paper are as follows:

- A generalised methodology that focusses on the systematic introduction and development of machine learning-based prediction models in production environments.
- An extension towards multiclass classification to explore potential additional applicational value and insights.

H. J. A. de Krom is with the Department of Maritime and Transport Technology, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands, and also with the Department of Transport & Planning, Delft University of Technology, Stevinweg 1, PO Box 5048, 2600 GA Delft, The Netherlands the (e-mail: h.j.a.dekrom@student.tudelft.nl).

M. B. Duinkerken is with the Department of Maritime and Transport Technology, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands (e-mail: m.b.duinkerken@tudelft.nl).

B. Wiegmans and L. A. Tavasszy are with the Department of Transport & Planning, Delft University of Technology, Stevinweg 1, PO Box 5048, 2600 GA Delft, The Netherlands (e-mail: b.wiegmans@tudelft.nl, l.a.tavasszy@tudelft.nl).

M. J. J. Hutten is with Philips Electronics Nederland B.V., High Tech Campus 52, 5656 AG, Eindhoven, The Netherlands, (e-mail: m.j.j.hutten@philips.com).

- An implementation of the proposed methodology that illustrates the potential and applicability in predicting delayed deliveries within a real-world production environment and supply chain.

The remainder of this paper is structured as follows. In Section II a concise overview of research efforts focussing on predicting supplier disruptions is given. The proposed methodology is presented in Section III of which the results of the application in the case study are presented in Section IV. Section V concludes and future research directions are presented.

II. Literature

To the best of the authors' knowledge, only four articles in scientific literature have applied ML in the context of identifying risk by means of predicting supplier-related disruptions in SCM. In these articles, two directions based on the prediction target are identified, (1) prediction of delayed deliveries and (2) prediction of stock-outs.

A. Prediction of delayed deliveries

Baryannis et al. [9] conducted a case study in a real-world multi-tier aerospace manufacturing supply chain focussing on the relation between second and first tier suppliers. They focus on the prediction of supplier-related risk by employing two binary classification algorithms, Decision Trees (DT) and Support Vector Machines (SVM), to predict whether future deliveries of suppliers will be delayed or not using historical product delivery data. To reduce negative influences of class imbalance, they applied 5-fold cross validation and explored various resample techniques with limited to insignificant effect in their experiments. Additionally, they expressed model performance in terms of F_1-score, Average precision and Matthew's Correlation Coefficient (MCC) to reduce bias in performance expression resulting from the class imbalance. Their experiments showed good performance, but no discussion regarding training results and the potential of overfitting was presented, limiting conclusions regarding generalisability and applicability of their results.

Brintrup et al. [10] have a similar focus as Baryannis et al. [9] as they investigate the possibility to apply ML to predict delayed supplier deliveries in an Original Equipment Manufacturer case study. Again, binary classification is considered for which the application of a Random Forest (RF) classifier is selected for their experiments. Historical supplier delivery performance data extracted from the manufacturer's Enterprise Resource Planning (ERP) system is considered. They reported the use of under-sampling to reduce negative effects of class imbalance, but no discussion regarding the implementation is presented. $F_{0.5}$, F_1, F_2, precision and recall are used to express model performance. They conclude that their conducted experiments are promising as they present significant improvement in the prediction of disruptions with limited information available. However, like Baryannis et al., no discussion of potential overfitting and training results are presented.

B. Prediction of stock-outs

De Santis et al. [11] consider a different aspect of SCM risks, focussing on inventory control by developing binary classification models predicting whether an item goes on backorder. A real-world imbalanced dataset made available by Kaggle's competition *Can You Predict Product Backorders?* is used, consisting of inventory, forecasted sales and supplier delivery performance data. To reduce negative impact of the highly imbalanced dataset, SMOTE and random under-sampling (RUS) have been used in combination with Logistic Regression (LR), Decision Trees, Random Forest and Gradient Boosting algorithms. Performance is solely expressed in Area under the ROC Curve (AUC), leading to more difficult interpretation of the results as it is sensitive to class imbalance as well. This is illustrated by their presented precision-recall curve (PRC), which present low performance on the important class while the AUC illustrate high performance.

Hajek and Abedin [12] consider the same dataset and focus as De Santis et al. [11], but use a different approach to predict the occurrence of backorders. Instead of using resampling strategies alone as has been observed in the other articles, Hajek and Abedin incorporate cost-sensitive learning, an approach in which costs are associated with predictions. They defined a cost function which they embedded together with cluster-based under-sampling (CBUS) in various ML algorithms, of which the combination with a RF classifier yielded the best results in terms of AUC. Interesting to observe is that the AUC value obtained by De Santis et al. [11] is higher than their reported AUC, but training performance is not discussed by Hajek and Abedin [12] as well, making comparison difficult.

III. Methodology

The generalised methodology proposed and applied in this paper for binary and multiclass classification consists of six steps as indicated in Fig. 1, which will be elaborated in the following sections.

A. Data collection and exploration

The basis of each ML model is the data used to train the model on combined with the desired focus and prediction target. Therefore, it is important to have a comprehensive understanding of the system in which the ML model is expected to operate, whilst being aware of the influences of the specific environment on and limitations of the data generation, processing and storage. Keeping in close contact with domain experts and practitioners is necessary to acquire these insights and address the correct question or problem.

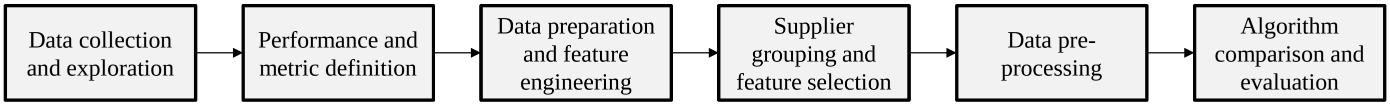
Data collection and exploration
Performance and metric definition
Data preparation and feature engineering
Supplier grouping and feature selection
Data pre-processing
Algorithm comparison and evaluation

Fig. 1: Overview of steps in the proposed methodology

Additionally, the obtained dataset is explored by simple visualisations and descriptive statistics to relate the acquired practical insights to the considered data.

B. Performance and metric definition

The selection of correct prediction performance metrics is crucial for valuable model development. Especially in a risk prediction setting where it represents the direct link to the desired outcome of the prediction process [9]. In classification, confusion matrices are commonly used to define metrics reflecting the desired focus while accounting for potential class imbalance. For binary classification, the confusion matrix is a two-by-two matrix consisting of the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) as illustrated in Table I. For multiclass classification, the confusion matrix can be easily extended to a n-by-n matrix when considering n different classes.

Given the general focus when predicting disruptions and the inherently linked imbalance problem, the use of Matthew's Correlation Coefficient (MCC) is recommended since it is able to give an intuitive and straightforward definition of performance in a single value independent of the initial class distribution [13]. The original definition for MCC in binary classification and its extension towards multiclass classification as defined by Gorodkin [14] are presented in the following two equations, respectively:

$$\text{Binary:} \frac{\mathrm{TP} * \mathrm{TN} - \mathrm{FP} * \mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}} \tag{1}$$

$$\text{Multi:} \frac{\sum_{\mathrm{klm}} \mathrm{C_{kk}C_{lm}} - \mathrm{C_{kl}C_{mk}}}{\sqrt{\sum_{\mathrm{k}}\left(\sum_{\mathrm{l}} \mathrm{C_{kl}}\right)\left(\sum_{\substack{\mathrm{l'} \\ \mathrm{k'\neq k}}} \mathrm{C_{k'l'}}\right)}\sqrt{\sum_{\mathrm{k}}\left(\sum_{\mathrm{l}} \mathrm{C_{lk}}\right)\left(\sum_{\substack{\mathrm{l'} \\ \mathrm{k'\neq k}}} \mathrm{C_{l'k'}}\right)}} \tag{2}$$

In (2), $\mathrm{C_{ij}}$ represents the number of predictions corresponding to classifying a data point to class i while it actually belonged to class j with j ≠ i.

Additionally, to support the MCC score and to be able to show prediction performance per class, precision and recall are selected. Precision represents the fraction of correct predictions *within a predicted class* and recall represents the fraction of correct *predictions of an actual class*. In the binary case, the precision and recall of the positive class are expressed as follows:

$$\text{Precision} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{3}$$

$$\text{Recall} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{4}$$

For multiclass classification, precision and recall values are class-wise computed, after which averaging can be applied to obtain a single value. Macro (M) averaging is recommended as it assigns equal weights to all classes, making it more suitable than micro or weighted averaging for imbalanced problems. Macro-precision and macro-recall are calculated using the following two formulas, respectively:

$$\text{Precision}_{\mathrm{M}} = \frac{\sum_{\mathrm{c}\in\mathrm{C}} \text{precision}(\mathrm{c})}{|\mathrm{C}|} \tag{5}$$

$$\text{Recall}_{\mathrm{M}} = \frac{\sum_{\mathrm{c}\in\mathrm{C}} \text{recall}(\mathrm{c})}{|\mathrm{C}|} \tag{6}$$

in which C represents the set of all classes considered.

Finally, accuracy can be considered as supportive metric since it is able to put precision and recall into perspective with respect to other classes and class imbalance.

TABLE I
EXAMPLE CONFUSION MATRIX

		Predicted Class	
		Positive	Negative
Actual class	Positive	True Positive	False Negative
	Negative	False Positive	True Negative

C. Data preparation and feature engineering

The available raw dataset needs to be cleaned by omitting inconsistent or other data points reflecting noise, imputing missing values and standardising units. However, before imputing missing values and standardising units, the dataset needs to be split in a train and test set to prevent data leakage.

Feature engineering is the step in which additional information is used to augment the prepared dataset. This involves transformation of existing data characteristics in the dataset and incorporation of domain knowledge and experience. Consultation with practitioners and field experts led to the definition of three feature domains relevant for supplier disruption prediction, of which suggestions are presented in Table V in the appendix:

- *Order*, representing features focussing on order characteristics as creation- and due date.
- *Supplier-material*, focussing on features regarding the considered supplier-material relation like contracted lead time and price.
- *Dynamic 'environment'*, encompassing features covering the time dimension and dynamic behaviour of which outstanding order quantity at the moment of ordering and preceding delivery performance are examples.

D. Supplier grouping and feature selection

The methodology initially considers individual suppliers since individual suppliers can vary significantly and therewith bias behaviour and performance when considering the entire dataset. Additionally, it is expected that considering individual supplier results in a better understanding and practical use. Complexity is initially reduced and extracted supplier specific behaviour could be easier verified and accepted by buyers, leading to higher acceptance and adoption of ML techniques. However, when purely focussing on individual suppliers, common behaviour might be excluded, for which the possibility of supplier grouping is introduced.

After the creation of additional supplier groups, feature selection is applied on the individually considered suppliers and defined supplier groups. By means of feature selection, the complexity can be reduced while simultaneously extracting important characteristics relevant to the prediction target which can increase understanding and assist in identifying root causes for disruptions. Recursive feature elimination (RFE) using feature permutation importance is recommended since feature permutation importance allows the presence of categorical features in the dataset. Feature permutation is expressed as the difference in model prediction performance after shuffling feature values. The selection of a performance metric able to express model performance in a single value while accounting for class imbalance is therefore needed, which illustrates the value of the recommendation for MCC as main metric.

E. Data pre-processing

To reduce negative influences of class imbalance, resampling can be applied on the resulting data subsets after feature selection. An exploration of no sampling, over-sampling, under-sampling and a hybrid form is recommended since different datasets are suitable for different resampling techniques. For over-sampling, the SMOTENC technique is recommended, which is a minor adaption of the commonly applied SMOTE technique allowing for datasets containing categorical features. For under-sampling, random under-sampling (RUS) is initially suggested given its simple implementation and shown value in preceding literature. The hybrid is a combination of SMOTENC and RUS, in which the majority class and minority class(es) are under- and over-sampled, respectively. Initially, a maximum over-sampling ratio of 2 and under-sampling ratio of 3 are suggested to reduce potential overgeneralisation and information loss.

Additionally, some ML algorithms require that the dataset is scaled and/or normalised or perform better after scaling and normalisation. Therefore, scaling and normalisation is applied in this step when needed.

F. Algorithm comparison and evaluation

Different algorithms and configurations need to be considered, since no generally best algorithm is available and the data provided to the algorithms heavily influence resulting performance. Therefore, a selection of algorithms to consider including parameter grids are needed to explore potential performance while preventing the occurrence of overfitting and reduction of generalisation performance. Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM) and eXtreme Gradient Boosting (XGB) are suggested algorithms to consider given their increasing complexity, different underlying definition and shown performance in literature.

Additionally, post-processing for binary classification models by means of threshold tuning can be applied. This can steer the prediction performance of the resulting prediction models more towards the metric most related to the desired focus at some cost of performance on different metrics.

IV. CASE STUDY

To illustrate the possibilities and value of the proposed methodology, it has been applied to a case study at a high-tech medical imaging manufacturer. A dataset containing of three years of historical deliveries has been obtained, which contained 68807 deliveries corresponding to 26512 unique orders for 2899 unique materials ordered at 180 unique suppliers after data cleaning. Of those deliveries 17.9% were delayed and 30.7% were delivered early. The average delivery moment is around two days before the due date, illustrating the problem of data imbalance. The data available for each delivery are the following:

- Order: purchase order and item number, order and due date and order quantity
- Supplier: supplier id, name, location and information exchange maturity level
- Material: material id and description, contracted lead time, safety time, price, ABC category, lot size type and material planning type
- Delivery: receipt date

The focus in the case study lies on predicting risk resulting from supplier disruptions, translating to predicting the occurrence of delayed supplier deliveries. A binary and multiclass classification problem are formulated using definitions obtained from the manufacturer (Table II).

TABLE II
TARGET CLASSES FOR BINARY AND MULTICLASS CLASSIFICATION

Problem	Target classes
Binary	• On-time: before or on due date • Delayed: after due date
Multiclass	• Extremely early: more than 3 days before due date • Early: between 1 and 3 days before due date • On-time: on the due date • Delayed: 1 or 2 days after due date • Extremely delayed: 3 or more days after due date

The following sections present the feature selection and performance results for a single supplier consisting of 10909 deliveries with a delivery performance as visualised in Fig. 2. The train set contained 8727 entries (80%) and the resulting 2182 entries (20%) form the independent test set, in which the delivery performance distribution is maintained. During feature selection and model training, stratified 5-fold cross validation is applied to ensure the same class distribution.

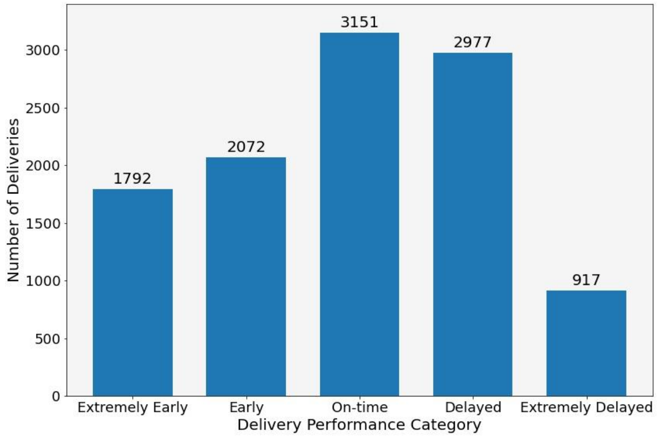

Fig. 2: Delivery performance

For the implementation of most ML algorithms, scikit-learn v0.24.0 [15] and xgboost v1.3.1 [16] are used, where we used imbalanced-learn v0.7.0 [17] for resampling. The experiments were conducted on computer with an Intel® Core™ I5-8365 CPU and 16 GB RAM.

A. Feature engineering

Irrelevant features regarding delivery performance were initially manually removed. This led to the exclusion of purchase order and item number, supplier id, name and location (since only one supplier is selected) and material description. To prevent data leakage, receipt date has been omitted as well, since this is unknown at the time of prediction. The difference between due and receipt date is taken as target variable, using the definitions presented in Table II.

Additionally, feature engineering is applied to enrich the dataset using experience and knowledge from practitioners and domain experts. Feature suggestions corresponding to the three defined feature domains have been added to the dataset while excluding highly correlated features, leading to the 28 features presented in Table VI in the appendix.

B. Feature selection

Recursive feature elimination using feature permutation importance has been conducted using a RF classifier with a max depth of 6, a max subsample of 0.8 and a balanced subsample class weight and MCC as metric.

Fig. 3 shows the results of the elimination process for the binary formulation, indicating a significant performance increase after removing the seventh most important feature, ‘Material’. In case of the specific supplier this can be expected, since the supplier has almost 840 unique material numbers associated and omitting this categorical variable reduces the dimension, and therewith complexity, significantly. The resulting subset contains five features consisting of ‘Due Week day’, ‘Sup. outstanding PO items’, ‘Due Week’, ‘Performance score 3 months’ and ‘Sup. outstanding POs’.

For the multiclass formulation, the results of the elimination process are shown in Fig. 4. The subset consisting of 14 features is selected since similar performance with respect to larger subsets is obtained and a noticeable decrease in performance is observed after removing the fourteenth feature. In this subset 8 additional features with respect to the binary formulation are selected, consisting of ‘Order/Lead time ratio’, ‘Material’, ‘ABC indicator’, ‘Creation Day’, ‘Performance score’ and ‘Due Day’.

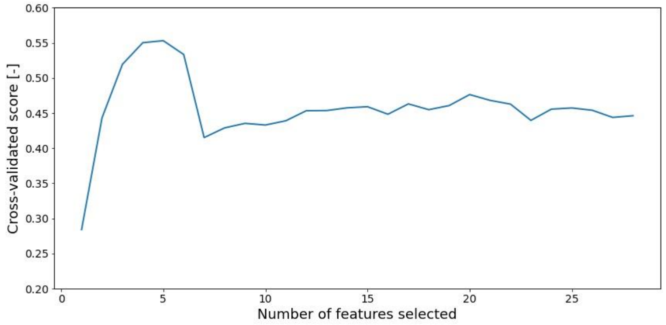

Fig. 3: Cross-validated model performance (MCC) during RFE (binary)

The observed importance of ‘Due Week day’ was not expected. Discussion with practitioners led to the hypothesis that it suggests inefficiencies in process-related aspects like day-offs of buyers or delayed invoicing. Therewith, feature importance and selection can assist in increasing model understanding, but also in potential root cause identification for inefficiencies in operations or supplier relations.

C. Algorithm comparison

Experiments using the five suggested algorithms (LR, DT, RF, SVM and XGB) have been conducted. For each algorithm, an indicative parameter grid has been defined to explore potential performance without the initial need to manually optimise parameters individually. LR and SVM have been omitted in the multiclass grids since significant increases in computational time occurred. Additionally, data pre-processing steps are included in the grid, to investigate the impact of sampling on the resulting performance. The considered grids are presented in Table VII in the appendix.

Analysis of the obtained results indicated the presence of overfitting. Therefore, parameter combinations leading to MCC differences larger than 0.1 between the test and train folds during cross-validation are omitted. The performance scores of the resulting algorithm-parameter combination with the highest MCC score on the test fold are presented in Table III and Table IV for the binary and multiclass formulation, respectively.

The results show high prediction performance in the binary formulation, reaching 88% accuracy with 85% of the predictions of late deliveries being correct and a correct prediction of 82% of the actual delayed deliveries. Similar performance is obtained on the test fold during model training as well, indicating a good generalisation ability of the trained model.

TABLE III
BEST PREDICTION SCORES FOR THE BINARY FORMULATION

Metric	Test set	Test fold	Train fold
MCC	0.7510	0.7842 ± 0.0143	0.8758 ± 0.0022
Accuracy	0.8863	0.9007 ± 0.0066	0.9432 ± 0.0010
Precision	0.8517	0.8741 ± 0.0177	0.9302 ± 0.0020
Recall	0.8254	0.8437 ± 0.0108	0.9091 ± 0.0027
Algorithm settings: XGB			
Gamma: 1	Max depth: 9	Min child weight: 3	
# estimators: 100	Sampling: None	Max subsample: 0.8	

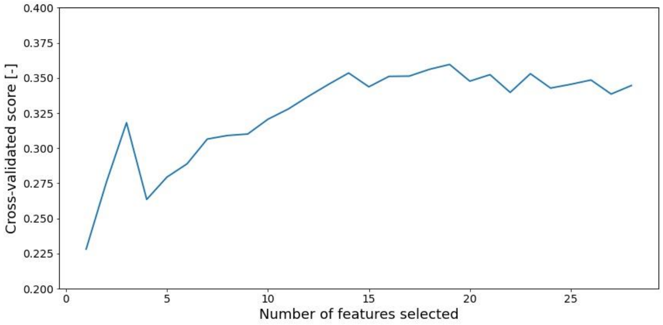

Fig. 4: Cross-validated model performance (MCC) during RFE (multi)

TABLE IV
BEST PREDICTION SCORES FOR THE MULTICLASS FORMULATION

Metric	Test set	Test fold	Train fold
MCC	0.6691	0.6599 ± 0.0099	0.7451 ± 0.0052
Accuracy	0.7438	0.7351 ± 0.0080	0.8016 ± 0.0042
Precision*	0.7439	0.7266 ± 0.0088	0.7930 ± 0.0049
Recall*	0.7453	0.7443 ± 0.0062	0.8099 ± 0.0037
Algorithm settings: XGB			
Gamma: 10	Max depth: 20	Min child weight: 0	
# estimators: 100	Sampling: Over	Max subsample: 0.8	

*Precision and recall are macro-averaged over the considered classes.

In the multiclass case, obtained performance is slightly lower, which can be expected since the introduction of additional prediction classes increases the complexity of the classification task and data requirements. Nevertheless, 74% accuracy is achieved, with a 74% macro-precision and macro-recall. Again, similar performance is obtained on the independent test set and the test fold during model training, indicating good generalisation ability.

V. CONCLUSIONS AND FUTURE RESEARCH

This paper presented a generalised methodology focussing on a systematic introduction and model development of ML-based prediction models with the aim to provide additional insights in SC operations and prediction analytics for supply chain (risk) management. The methodology initially considers individual supplier to create higher transparency and reduced complexity, whereafter supplier grouping can take place to cover more general relations. The methodology is applied in a case study of a high-tech medical imaging manufacturer. Binary and multiclass classification have been employed, of which the obtained results illustrate the possibility of the methodology to obtain good performance and additional insights. The utilisation of feature importance can assist in identifying root causes for inefficient operations and supplier relations, while high performing prediction models can assist in assessing and (timely) mitigating risk resulting from supplier disruptions.

Future research is needed to increase prediction performance in the binary and multiclass formulations by focussing on more specific parameter grids or additional parameter tuning. This includes investigating different algorithms and techniques and in particular the impact of sampling ratios on the resulting performance. Additionally, different data can be added to improve performance or incorporate the possibility to provide prediction updates once additional data (e.g., order configurations or transport updates) become available.

APPENDIX

TABLE V
OVERVIEW OF SUGGESTED FEATURES FOR THE DIFFERENT FEATURE DOMAINS

Domain	Feature
Order	• Creation/Due Day • Creation/Due Day of Week • Creation/Due Week • Creation/Due Month • Creation/Due Season • Creation/Due Year • Days between Creation and Due date • Material • Quantity • Supplier • Value • Days between confirmed delivery date and due date • Order changed indicator • Order involved execution of mitigating measure
Supplier-material	• Price per material • Contracted lead time • Ratio of quantity over standard quantity • Ratio of material order frequency over standard frequency • Ratio of given time for fulfilment and contracted lead time • Size of product portfolio for corresponding supplier • Considered Safety time • Unique number of materials produced/ordered at a supplier • Default shipment method • Possible alternative (express/priority) shipping methods
Dynamic 'environ ment'	• Previous (confirmed) order delivery performance • Open or outstanding (confirmed) quantity (per material) • Open or outstanding (confirmed) overdue quantity (per material) • Ratio of current requested/outstanding quantity over maximum allocated production quantity/capacity in a time period • Ratio of current requested quantity over shared forecasted quantity • Performance/number of deviations of supplier confirmed orders • Inventory level at moment of ordering

Table VI
Overview of considered features before recursive feature elimination

Feature	Format	Definition
Order		
Creation Day	Integer	Day of the month of the creation date
Creation Week day	Integer	Week day of the creation date
Due Day	Integer	Day of the month of the due date
Due Week	Integer	Week of the due date
Due Week day	Integer	Week day of the creation date
Material	Category	Material number as categorical variable
Quantity	Float	Quantity of ordered material
Delivery value	Float	Monetary value of the specific delivery
Order/Lead time ratio	Float	Ratio between time between order and due date normalised by the material's lead time
Late order indicator	Binary	Indicator whether the order is placed with a due date sooner than the contracted lead time
Frequency indicator	Binary	Indicator whether the order is placed sooner with respect to lot size or last quarter's median order frequency
Quantity indicator	Binary	Indicator whether the order quantity is higher than the fixed lot size or last quarter's median order quantity
Supplier – material		
Lead time	Integer	Contracted lead time in number of days
Safety time	Integer	Number of workdays the material order is moved forward
ABC indicator	Category	Material class according to ABC classification (quantity-spend relation)
SNC relevancy	Text	Category indicating maturity level of information exchange
Lot size type	Text	Category indicating frequency or time dependent replenishment
Dynamic 'environment'		
Performance score	Float	Percentage of on-time deliveries of the most recent month in which deliveries were expected
Performance score 3 months	Float	Average of Performance scores from the last three months
Outstanding POs	Integer	Number of outstanding standard orders for the material
Overdue POs	Integer	Number of overdue standard orders for the material
Outstanding quantity	Float	Outstanding quantity for the material
Weighted overdue quantity	Float	Quantity overdue weighted by the amount of days overdue for the material
Sup. outstanding POs	Integer	Number of outstanding standard orders on supplier level
Sup. outstanding PO items	Integer	Number of outstanding PO items on supplier level
Sup. overdue POs	Integer	Number of overdue standard orders on supplier level
Sup. outstanding quantity	Float	Quantity of outstanding material on supplier level
Sup. weighted overdue quantity	Float	Quantity overdue weighted by the amount of days overdue on supplier level

Table VII
Overview of applied parameter grids

Algorithm	Parameters	Range (binary)	Range (multiclass)	
Logistic Regression (LR)	- Regularisation – C - Class weight - Solver - Sampling	- 0.1, 0.5, 1, 5, 10, 50, 100, 500, 1000 - None, balanced - Newton-cg, lbfgs - None, SMOTENC, RUS, Hybrid		
Decision Tree (DT)	- Max tree depth - Max leaf nodes - Class weight - Sampling	- 5, 6, 7, 8, 9, 10 - 20, 30, 40, 50 - None, balanced - None, SMOTENC, RUS, Hybrid	- 5, 10, 15, 20 - 100, 200, 400, 800 - None, balanced - None, SMOTENC, RUS, Hybrid	-
Random Forest (RF)	- Max tree depth - Max leaf nodes - Nr. of estimators - Class weight - Sampling -	- 5, 6, 7, 8, 9, 10 - 20, 30, 40, 50 - 10, 50, 100, 500 - None, balanced_subsample - None, SMOTENC, RUS, Hybrid -	- 5, 10, 15, 20 - 100, 200, 400, 800 - 10, 50, 100, 500 - None, balanced_subsample - None, SMOTENC, RUS, Hybrid	-
Support Vector Machine (SVM)	- Regularisation – C - Gamma – γ - Class weight - Sampling	- 0.1, 1, 10, 100, 1000, 10000 - 0.001, 0.01, 0.1, 1, 10 - None, balanced - None, SMOTENC, RUS, Hybrid		
eXtreme Gradient Boosting (XGB)	- Max tree depth - Min child weight - Gamma – γ - Nr. of estimators - Sampling	- 5, 6, 7, 8, 9, 10 - 1, 2, 3 - 0, 1, 10, 100 - 10, 50, 100, 500 - None, SMOTENC, RUS, Hybrid	- 5, 10, 15, 20 - 0, 1 - 0, 1, 10 - 10, 50, 100 - None, SMOTENC, RUS, Hybrid	-

REFERENCES

[1] K. Alicke and I. Balaji, "Next generation supply chain: Supply chain 2020," McKinsey & Company, July 2013 2013.

[2] K. Alicke, J. Rachot, and A. Seyfert, "Supply Chain 4.0 - the next-generation digital supply chain," McKinsey & Company, June 2016 2016.

[3] J. Zhao, M. Ji, and B. Feng, "Smarter supply chain: a literature review and practices," *Journal of Data, Information and Management,* vol. 2, no. 2, pp. 95-110, 2020/06/01 2020, doi: 10.1007/s42488-020-00025-z.

[4] G. Behzadi, M. J. O'Sullivan, T. L. Olsen, F. Scrimgeour, and A. Zhang, "Robust and resilient strategies for managing supply disruptions in an agribusiness supply chain," (in English), *Int J Prod Econ,* Article vol. 191, pp. 207-220, 2017, doi: 10.1016/j.ijpe.2017.06.018.

[5] Y. Sheffi, *The resilient enterprise: overcoming vulnerability for competitive advantage*. Zone Books, 2007.

[6] T. Nguyen, L. Zhou, V. Spiegler, P. Ieromonachou, and Y. Lin, "Big data analytics in supply chain management: A state-of-the-art literature review," (in English), *Comp. Oper. Res.,* Article vol. 98, pp. 254-264, 2018, doi: 10.1016/j.cor.2017.07.004.

[7] D. Ni, Z. Xiao, and M. K. Lim, "A systematic review of the research trends of machine learning in supply chain management," (in English), *Intl. J. Mach. Learn. Cybern.,* Article vol. 11, no. 7, pp. 1463-1482, 2020, doi: 10.1007/s13042-019-01050-0.

[8] G. Wang, A. Gunasekaran, E. W. T. Ngai, and T. Papadopoulos, "Big data analytics in logistics and supply chain management: Certain investigations for research and applications," (in English), *Int J Prod Econ,* Review vol. 176, pp. 98-110, 2016, doi: 10.1016/j.ijpe.2016.03.014.

[9] G. Baryannis, S. Dani, and G. Antoniou, "Predicting supply chain risks using machine learning: The trade-off between performance and interpretability," (in English), *Future Gener Comput Syst,* Article vol. 101, pp. 993-1004, 2019, doi: 10.1016/j.future.2019.07.059.

[10] A. Brintrup and A. Ledwoch, "Supply network science: Emergence of a new perspective on a classical field," (in English), *Chaos,* Article vol. 28, no. 3, 2018, Art no. 033120, doi: 10.1063/1.5010766.

[11] R. B. De Santis, E. P. De Aguiar, and L. Goliatt, "Predicting material backorders in inventory management using machine learning," in *2017 IEEE Latin American Conference on Computational Intelligence, LA-CCI 2017*, 2018, vol. 2017-November: Institute of Electrical and Electronics Engineers Inc., pp. 1-6, doi: 10.1109/LA-CCI.2017.8285684.

[12] P. Hajek and M. Z. Abedin, "A Profit Function-Maximizing Inventory Backorder Prediction System Using Big Data Analytics," (in English), *IEEE Access,* Article vol. 8, pp. 58982-58994, 2020, Art no. 9046037, doi: 10.1109/ACCESS.2020.2983118.

[13] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," (in English), *BMC Genomics,* Article vol. 21, no. 1, 2020, Art no. 6, doi: 10.1186/s12864-019-6413-7.

[14] J. Gorodkin, "Comparing two K-category assignments by a K-category correlation coefficient," (in English), *Comput. Biol. Chem.,* Article vol. 28, no. 5-6, pp. 367-374, 2004, doi: 10.1016/j.compbiolchem.2004.09.006.

[15] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," (in English), *J. Mach. Learn. Res.,* Article vol. 12, pp. 2825-2830, 2011.

[16] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," presented at the Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, California, USA, 2016. [Online]. Available: https://doi.org/10.1145/2939672.2939785.

[17] G. Lemaître, F. Nogueira, and C. K. Aridas, "Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning," (in English), *J. Mach. Learn. Res.,* Article vol. 18, pp. 1-5, 2017.