MASTER

Causal and Interpretable Predictive Analytics in Operational Decision Making for Emergency Maintenance Order Fulfillment
a Case Study at ASML

Borghouts, Tom

Award date:
2023

Eindhoven University of Technology

Master Thesis

Department of Industrial Engineering & Innovation Sciences

Causal and Interpretable Predictive Analytics in Operational Decision Making for Emergency Maintenance Order Fulfillment: a Case Study at ASML

Master Thesis Report

Author:
Tom Borghouts
0998254

Supervisors TU/e
dr. ir. H. (Rik) Eshuis
dr. ir. Z.A. (Zaharah) Bukhsh

Company Supervisors
B. (Begoña) Alcorta

Eindhoven, 2023-01-24

Abstract

The undeniable emergence of Big Data analytics is considered as one of the enablers to support decision-makers in their complexity increasing processes. This thesis explored the use of causal and interpretable predictive analytics in operational decision-making, specifically in the context of emergency maintenance order fulfillment in the Customer Supply Chain Management department (CSCM) of ASML, a leading semiconductor company. The study aimed to explore how this approach can be used to identify the most effective sourcing solution, defined as the option with the lowest cost and highest feasibility, in order to minimize redundant checks in the current decision-making process. To answer this research question, the study compared the performance of machine learning (ML) models using causal techniques to traditional ML techniques. It also explored the use of ML Interpretability methods for model validation, improvement, and knowledge discovery, and considered the potential benefits of integrating model abstention into the best ML prediction model. The results demonstrated that the suitability of Causal ML in the problem setting described may be questionable due to the limitations of this method on currently available algorithms and the difficulty of meeting the necessary conditions for successful causal inference in a problem with limited causal data and operational decision-making. The results from the used global ML Interpretability techniques first needed to be converted to an easier interpretable format before they could properly be used for model validation and knowledge discovery with SMEs and stakeholders. Yet, the use of global ML Interpretability techniques for model improvement showed only a minor increase in predictive performance. The use of a sequentially learned ambiguity abstention model demonstrated how it could improve the predictive performance for the non-rejected cases, which could increase end-users' trust in the model and still remove redundant checks by only requiring planners to use the current workflow for the rejected cases.

Keywords:
Causal Machine Learning, Interpretable Machine Learning, Machine Learning Abstention, Operational Decision Making

Executive Summary

Introduction

Increasing complexity and demand for explainability in operational decision-making calls for the adoption of Big Data, analytics, and AI to assist decision-makers [Gartner, 2021]. The use of Machine Learning (ML) in operations management and supply chain management has been recognized as a useful tool [Baryannis et al., 2019, Topan et al., 2020]. Despite the recent advances in ML, researchers raise their concerns about the lack of robustness, inability to connect causes and effects, explainability and interpretability in "traditional" ML methods [Pearl, 2019, Schölkopf, 2022]. Yet, the latter obstacles are particularly important for an AI solution to be valuable in the domain of supply chain management [Baryannis et al., 2019]. Causal ML is the study of how to use ML techniques to infer causal relationships between variables in a certain problem, and ought to be the solution to the earlier mentioned obstacles. In response to the need for explainability and interpretability, recent research has focused on both developing interpretable models and methods for generating explanations. [Carvalho et al., 2019], and can be applied for model validation, model improvement, and knowledge discovery [Du et al., 2019]. Despite this emergence, there is still no consensus on the processes and routines for these purposes [Molnar, 2020], especially in operational decision-making in the supply chain management domain [Baryannis et al., 2019]. Also, the presence of unavoidable noise and uncertainties results in mispredictions [Chow, 1970], which eventually can lead to bad decisions and low end-user trust [Yin et al., 2019]. One approach to address these issues is through the use of abstention models, which are designed to refrain from making a prediction when the model is uncertain. These models are often used in high-stakes domains such as healthcare, but their application in operational decision-making remains limited, despite their potential benefits.

ASML, an innovative, semiconductor industry leader, aspires data-driven decision-making. In particular, ASML's customer supply chain management department aspires the use of ML to become more data-driven in its decision-making in its mission to fulfill the demands and needs of its customers. Specifically, the current decision-making process in its emergency maintenance order fulfillment process follows a static workflow designed to decrease overall high labour and activity costs. Accordingly, planners from the global operations center investigate the cheapest and most occurring solutions first until a feasible solution is found, which results in many redundant checks.

The objective of this thesis was to explore how causal and interpretable predictive analytics can support operational decision-making in an emergency maintenance order sourcing process. We conducted an exploratory case study to determine how this approach could be used to identify the most effective sourcing solution, defined as the option with the lowest cost and highest feasibility, to minimize redundant checks.

Research Questions

From the above, we formed the following main research question:

> "How can causal and interpretable predictive analytics be used to support ASML's global operations center planners in choosing the most effective sourcing solution within the emergency maintenance order sourcing process?"

In order to answer this research question, we defined the following sub-research questions.

1. *How can Causal ML be used to predict the most effective sourcing solution?*
2. *How do Causal ML models compare with traditional ML prediction techniques when applied to predicting the most effective sourcing solution?*
3. *How can ML Interpretability methods be used for model validation and improvement, and knowledge discovery?*
4. *How can the best ML prediction model be enhanced with model abstention?*

Methodology

We employed a tailored version of the CRISP-DM process model [Wirth and Hipp, 2000], a widely used and industry-agnostic approach for data mining and knowledge discovery projects, as shown in Figure 1. The iterative nature of the model allowed us to progress through the various phases, like general Data Understanding and Preparation, and address the sub-research questions.

First of all, we defined our evaluation metrics. The ML model's Precision was deemed more important for expensive solutions, as we would not want to mispredict those solutions. Whereas we do not want to miss out on cheap solutions, and thus deem Recall more important for these solutions. For the purpose of Causal ML, we used the causal policy learner Doubly Robust Random Forest Policy Learner (DRRFPL). This model first learns the causal effects and compensates for bad models of past treatments or outcome variables with good models of the opposite, which was needed as our outcome variable was sparse, i.e. we only have data on the most effective solutions. This policy learner then builds a multitude of decision trees with the objective function to treat as many samples correctly as possible. To address the issues of outcome sparsity and class imbalance, we conducted experiments using synthetic control samples to provide contrast for the model, as well as resample techniques to mitigate the impact of the class imbalance. We selected the best set-up and compared this to traditional ML. For this purpose, we used a manually developed Random Forest Classifier (RFC), and AutoML by Azure. The former method is known to attain high accuracy and still be interpretable, while the latter allows to quickly search a multitude of different ML methods regardless the extent of interpretability.

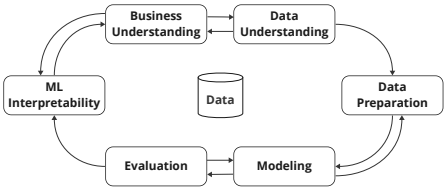


Figure 1: Research framework based on CRISP-DM [Wirth and Hipp, 2000]

For the purpose of knowledge discovery, model validation and improvement with ML Interpretability, we mainly used global ML Interpetability methods. These types of ML Interpretability methods describe the average behaviour of a ML model. We used these methods to gain insights and presented them to SMEs for model validation and knowledge discovery. Next to that, we visualized different methods to discover subgroups within the data that were mispredicted. Based on the insights, we developed a set of Model Debug Opportunities (MDOs) with the goal of enhancing the performance of the best prediction model. These MDOs were then evaluated and compared to the best ML model.

Finally, we designed an abstention model for our best prediction model that rejected predictions for which the model is uncertain (ambiguity). We learned this model sequentially to the ML model, as this would enhance interpretability, and we optimized the thresholds to reject with a genetic algorithm, for convenience and domain appropriateness. Next, we estimated the costs saved per Emergency Order (EMO) if a planner would have executed the checks for the predicted or optimal solution, allowing us to compare the different workflows. For EMOs that were not the most effective, we used the cumulative costs until the most effective was reached according to the predicted class probabilities by the ML model.

Results and Discussion

Firstly, we compared the different Causal ML set-ups. None of the solutions scored well on the set performance metrics, despite differences in performance per class and individual metrics. All the tested setups contained a preferential bias towards the cheapest solution and majority class `Unrestricted`. We presumed that this bias was coming from either data to algorithm or algorithm to user, but later concluded that it was mainly coming from the former one, which reflects on the measurement, omitted variable and representation bias. Besides, the causal assumptions were violated which could have affected the performances strongly. For example, we could not include all confounding variables because of data availability constraints. The poor predictive performances would not only lead to poor decision-making, but also hindered the ability to extract causal knowledge.

Next, we compared the best-performing Causal ML model to traditional ML methods (RFC and

AutoML) and found that the traditional ML models generally outperformed the Causal ML model. We mainly presumed that this came from bigger noise in the features that are learned to be causal compared to the features that were learned not important for Causal ML in line with the work of Fernández-Loría and Provost (2022). Besides, the objective function of the policy learner reflected on the accuracy rather than our performance metrics. Between the two traditional ML methods, we observed the same prediction behaviour and performance. Still, the models missed out on a substantial part of the minority solutions, which again underlined the bias towards the majority solution.

In the interest of ML Interpretability, we applied various interpretability methods to gain insights into the ML prediction model. These methods mainly agreed on the feature importance rankings. Further analysis revealed the relationship between the predicted probability for a particular solution and different values for each important feature. We generally observed similar patterns for the minority classes, where a few features were found to be useful for distinguishing between solutions. The results and insights from this analysis could be used to validate the model with SMEs and eventually stakeholders. While the more sophisticated interpretability methods were difficult for SMEs to understand, summarizing and highlighting particularities allowed for proper model validation with both SMEs and stakeholders.

Lastly, after choosing the design of the abstention model, we analyzed the current prediction uncertainty for the ML model's (mis)predictions per solution. This uncovered that the model was fairly sure about its mispredictions on the solution majority solution ($median = 0.76$), but less sure about its mispredictions on minority solutions ($median = 0.59$), which once again exhibited the present bias. Subsequently, we designed and optimized the abstention model. As can be seen in Table 1, using the ML prediction model would realize savings, where an abstention model would increase these savings, but both are not close to the optimal workflow (no redundant checks).

Workflow	**Costs (min)**	**Savings (min)**
Current	11,759	N/A
Without rejection	11,128	631
With rejection	10,773	986
Optimal	6,170	5,589

Table 1: Costs and savings realized on the test set

Conclusion and recommendations

With the findings of this thesis, we conclude and recommend the following:

- Based on the assumptions and conditions of causal inference, the suitability of Causal ML in our problem setting may be questionable. In a problem with limited causal data and operational decision-making, it may be difficult to meet all necessary conditions for success with this ML method. While the chosen Causal ML method was useful for multi-class classification, its black-box characteristic and limited objective function may be limiting factors as well.
- Presumably because of the above given explanations, traditional ML still remains a better option as it can rely on non-causal but more informative data which contain less noise. Besides, AutoML stood out as a method to quickly obtain a model comparable to a manually developed ML model, regardless of its current limitations. Businesses should explore this field of AutoML in order to save time on developing models and ramp up the quantity while remaining quality.
- The use of global ML Interpretability methods allowed to validate the model with business, but did not contribute significantly to diminishing the bias and improving the model. Subsequently, future research should examine the use of local ML Interpretability methods for this purpose.
- The use of a sequentially learned ambiguity abstention model showed an increase in predictive performance for non-rejected cases, which may increase user trust in the model, and also reduced redundant checks by requiring planners to only follow the current workflow for rejected cases. We showed that the workflow with abstention resulted in cost savings of 56% compared to the workflow without abstention. Though, to realize the optimal potential benefits of this approach, the base ML model performance should be improved.

Preface

To begin with, I would like to express my gratitude for your interest in this master thesis report. This thesis was conducted as part of and concluded my master program Operations Management and Logistics at the Technical University of Eindhoven.

The journey leading up to this conclusion began in 2016, when I started my bachelor's degree in Industrial Engineering with a broad interest. As time passed, my focus narrowed to the field of advanced data analytics (applied in operations management). In addition to the knowledge I gained through my studies, my extracurricular activities as a student have contributed significantly to shaping who I am today. I want to especially thank those who I hardly knew or did not know at all seven years ago, but who have played a central role in my life. In particular, I want to thank my roommates from Zjem & Confiture for the last 5.5 years, my boardmembers from UniPartners Eindhoven, Ruben and Juup, my friends from Dordrecht, and last but not least my family.

Next, I would like to express my gratitude to ASML for providing me with the opportunity to conduct my thesis. My team, Service Business Applications, made me feel welcome and helped me gain insight into, especially, the field of advanced analytics and automation. I would especially like to thank Begoña for her excellent supervision and for sharing her extensive knowledge of ASML's supply chain in combination with advanced analytics solutions.

Finally, I would like to thank my first TU/e supervisor dr. ir. H. Eshuis for his supervision. I am grateful that you challenged me to research beyond just a machine learning-based predictive analytics solution.

Contents

List of Acronyms

BPM	Business Process Management
CS	Customer Support
CSCM	Customer Supply Chain Management
CV	Cross-validation
DSP	Direct Service Parts
DRRFPL	Doubly Robust Random Forest Policy Learner
EMO	Emergency Order
GOC	Global Operations Center
ICE	Individual Conditional Expectation
MDI	Mean Decrease in Impurity
ML	Machine Learning
PDP	Partial Dependence Plot
PIMP	Permutation Importance
RFC	Random Forest Classifier
SLA	Service Level Agreement
SME	Subject Matter Expert
SO	Service Order
UND	Ultimate Need Date
XAI	eXplainable Artificial Intelligence

List of Figures

List of Tables

1 Introduction

Making optimal decisions in operational decision-making is becoming increasingly challenging. According to a recent Gartner survey, 65% of the respondents indicated that the complexity and involvement of stakeholders in decision-making processes are increasing [Gartner, 2021]. Additionally, 53% of the respondents indicated they feel greater pressure to explain or justify their decisions. With an increase in data-driven solutions in automating such processes, human decision-makers certainly should not be swapped. In fact, the emergence of (Big) Data, analytics and AI should be embraced in order to support and complement human decision-makers in their tasks. Accordingly, ASML, an innovative semiconductor industry leader, aspires data-driven decision-making. Specifically, in ASML's Customer Supply Chain Management (CSCM) department's mission to fulfill the demands and needs of their customers, they live up to their vision: "We team up with technology for a predictable, no-touch and circular supply chain network, thereby maximizing efficiency and minimizing our footprint".

The emergence of Machine Learning (ML) has not remained unnoticed in the discipline of operations management in combination with Big Data analytics [Topan et al., 2020]. Even more, ML has achieved extraordinary capabilities with the continuous theoretical developments and increasing computational power systems. Despite the recent advances in ML solutions, researchers raise their concerns on the neglectful use of these "traditional" ML methods [Pearl, 2019, Schölkopf, 2022], pertaining to the lack of explainability and interpretability, robustness, the inability to connect causes and effects. Instead, they propose Causal ML, which overcomes these obstacles with the proper embodiment of Causal Inference. In line with the first mentioned concern, AI solutions for operational decision-making in a supply chain context are ought to be predominantly valuable if they are made interpretable and justified [Baryannis et al., 2019]. Despite this awareness, Causal ML and ML Interpretability are understudied in this field of operational decision-making. In a case study conducted at ASML's CSCM department, we explore how causal and interpretable predictive analytics can be used in operational decision-making for emergency maintenance order fulfillment.

This chapter explains the reasoning behind this research. It provides an overview of the business context, prospective, scope, and problem statement, and then presents a brief literature review and discusses the business and scientific relevance of the study. Together, they led to the development of the research question and sub-research questions for this study.

1.1 Business description

This research is commissioned by Advanced Semiconductor Materials Lithography (ASML) which is one of the world's leading designers and manufacturers of the lithography machines that are an essential component in chip manufacturing. ASML's customers are companies such as Intel, who use machines in 'fabs' – microchip manufacturing plants – to create microchips that are eventually used in many electronic devices, including smartphones, laptops and much more. Within this semiconductor industry, ASML is an innovation leader and provides their customers with everything they need – hardware, software and services – to mass produce patterns on silicon, allowing them to increase the value and lower the cost of a chip. The industry is driving Moore's Law into the next decade, enabling global megatrends like 5G, AI, HPC, VR/AR, autonomous vehicles. As a result of the sky-rocketing chip demand caused by these trends, ASML needs to further develop its existing and new machines.

1.2 Problem context

With the purpose of answering the needs in machine availability of ASML's customers, after-sales service is included as a product. This enables less machine downtime because of transparent and controlled customer supply chain management. To keep the machines produced by ASML operational, they are (preventively) maintained. The materials required for this are kept in stock in various warehouses around the world which can be distinguished into two levels: local warehouses, and central warehouses. Next to these warehouses, there is also stock stored at the ASML factory, and ASML's

suppliers if the material cannot be received yet. When one or more materials are needed at a customer's site, a Service Order (SO) is created. Besides maintenance orders being a SO, also other types of events, like installations or machine upgrades, also become a SO. However, for the sake of this research, we use SO as a term for maintenance order as it is used like this within ASML. The material sourcing of the SOs is fulfilled by automation engines, but also manually when automation is not possible or a feasible (unrestricted stock available) and on time (lead time does not exceed the time to Ultimate Need Date (UND)) solution cannot be found.

The moment a customer's machine breaks down and needs corrective maintenance, a SO is created in which the necessary materials are requested for the fastest possible delivery. To distinguish the priority of a SO, this type of SO is called an Emergency Order (EMO). The Global Emergency Support Automation (GESA) system is responsible for finding a fitting solution for the EMO and tries to source the needed materials. However, if no feasible solution for sourcing a material for the EMO can be found, the sourcing challenge is forwarded to the Global Operations Center (GOC), which is located in Taiwan and operates 24/7, to find a suitable solution. A possible solution would then be to get the part from the factory in Veldhoven. By performing various checks, the GOC knows which (bundle of) activities it must perform in order to realize a suitable solution. These decisions are, just as the automatic systems, also based on business rules. Besides SOs for corrective maintenance being EMOs, a total or item(s) from initially non-emergent SOs belonging to preventive maintenance can become emergent as well. This can be the result of, for example, parts that arrived broken at the customer because of quality issues, whereafter a new demand is generated for this SO item, or stock reservations are manually overruled by planners, whereafter no feasible solution can be found anymore for a specific SO item.

The manual EMO sourcing process and its phases can be found in Figure 2. When a SO is requested on emergency and GESA cannot fulfill this order, the GOC receives this order (phase 0). Next on, GOC gives a first response with an initial action plan and preliminary Estimated Time of Arrival (ETA) to give the CS engineer at the customer's site a status update (phase 1). Subsequently, a GOC planner picks up the SO and starts performing checks to find a feasible sourcing solution (phase 2). When a feasible solution is found, it may need to be checked physically for its availability inside or outside ASML's network (phase 3). After availability confirmation, the ETA will be confirmed, and a delivery notification will be created when the material is made available (phase 4). The SO is now sourced and transported to the local warehouse, while in the meantime this is monitored by the GOC as well (phase 5).

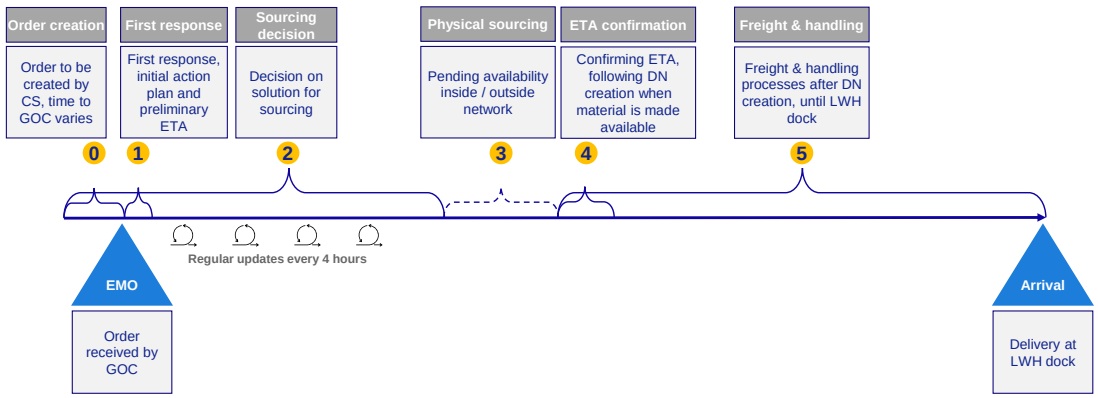


Figure 2: Manually handled EMO process phases

As explained above, the process starts when the SO is received by the GOC. A GOC planner picks up the order and performs multiple checks on the SO (phase 2). This checking phase is called *request preliminary checks*. Every set of checks can be seen as an investigation of a sourcing solution, e.g. checking if an alternative material is available. However, when a set of checks shows solution

infeasibility, the planner moves on to the next set of checks. So, in some cases, the first set of checks can already give a feasible solution after which the process ends, but in some cases, it can take a lot more time to find a feasible solution, as multiple sets of checks have to be executed. This makes the lead time and processing time uncertain. Planners follow one workflow for the process which is designed to minimize the sourcing time. The first sets take less time for the planner, are less dependent on other teams, are in general less costly, and will result in less (planning) disruptions than later sets. For example, the checks start with analyzing the stock positions beginning with regional to worldwide warehouses, but checks in a later set will analyze the repair status of a material at ASML's factory. As the case may be, planners have to wait for information on the checks because of for example contact with other teams. The average time spent by a GOC planner is 20 minutes and can vary from 10 to 80 minutes, but is strongly dependent on material availability and thus on how many sets of checks have to be performed. When planners cannot find a feasible solution after performing all the checks, the EMO will be escalated. In case of escalation, the shift lead, and in some cases the manager of the GOC, will be involved in the process. Together, they will try to find a feasible solution by contacting for example suppliers.

1.3 Prospective

ASML's CSCM department has set a dream state of the operating framework in which it wishes to operate its supply chain by 2025. This dream state is oriented towards achieving three targets: becoming 80% predictable, 90% no touch, and 100% circular supply chain. Predictability could for example be knowing upfront that an event, such as maintenance, would happen, or estimating lead times with high accuracy. No touch refers to handling its supply chain documents in an automated manner, e.g. handling a SO, without manual activities. Finally, a circular supply chain focuses for example on the remake and reuse of leftover materials. The current situation status for every pillar is not yet known, but they can still be improved up to their targets. Within ASML's CSCM department, becoming more data-driven is believed to be a critical enabler for the achievement of the set targets. In conformity with the above, its vision "We team up with technology for a predictable, no-touch and circular supply chain network, thereby maximizing efficiency and minimizing our footprint" plays a central role in its mission to deliver material to its customers in the field on time, in full, and at the right quality and cost.

Furthermore, due to the increasing demand in the chip industry, ASML is expected to grow significantly in the coming years. This increases the pressure on the GOC considerably and the current way of working would require a higher workforce to meet the requested UNDs in time. The end-to-end support lead time Service Level Agreement (SLA) for this type of request is 72 hours, which is made up of 24 hours sourcing lead time and 48 hours physical shipment. However, due to the pressure on both sourcing and physical shipment, this is currently not being achieved. Besides the increasing pressure from the customer, anecdotal evidence shows that staffing is also becoming an increasing challenge. The pool of new talent is shrinking, and ASML cannot hire new employees exponentially within the GOC. As a result of the increasing workload and the challenge in staffing, ASML will have to work differently and smarter to address this challenge and meet customers' demands.

1.4 Scope

In order to better investigate the *request preliminary checks* phase process, its problems, causes, and relationships among each other, an in-depth analysis was performed. This was done with the use of a cause and effect analysis, which is visualized in Figure 3. There were four main causes for the high workload and eventual delay in SO fulfillment. Though, *Inefficient collaboration* would be tackled by an improvement team within ASML, and *Sourcing more than needed* is out of this scope as it is a result of planning and forecasting and covering contingencies to decrease downtime. Therefore, this research was focused on the *inefficient process design* and *inefficient sourcing*.

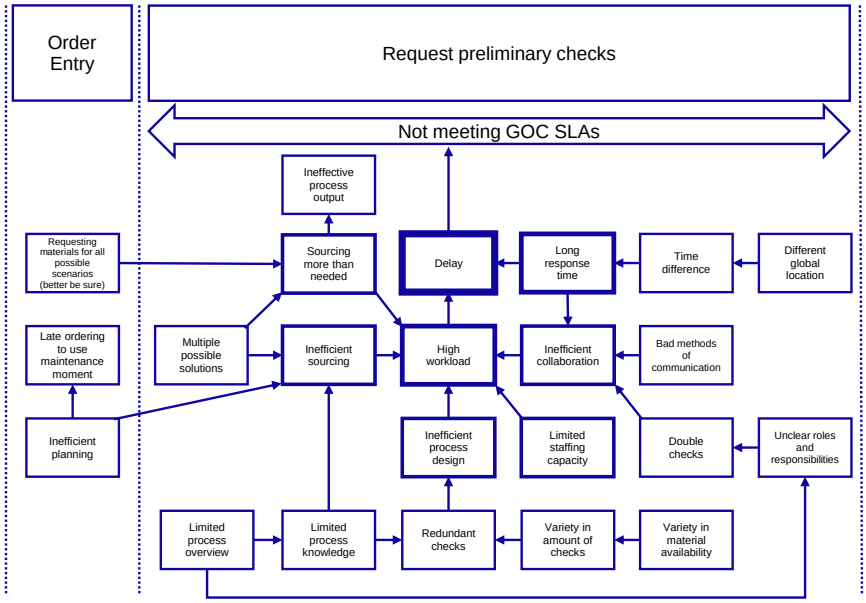


Figure 3: Cause and effect diagram manual EMO sourcing process

1.5 Business problem statement

As explained earlier, the particular set(s) of checks which will lead to a feasible solution is not known at the beginning of the process. The current decision-making process on which checks to perform is following a static workflow designed to decrease overall high labour and activity costs. This frequently results in planners performing unsuccessful checks, which makes the checks in some sense redundant. Hence, the time spent on these redundant sets of checks could be seen as inefficient use of time. As a result, the set SLAs are currently not met, and set UND of SOs are potentially delayed. Consequently, customer satisfaction can decrease, which is especially the case for machine hard-downs. Summarizing, we proposed the following business problem statement:

Business problem statement
The current manual EMO sourcing process at ASML has an inefficient process design due to the execution of redundant checks, resulting in not meeting SLAs.

1.6 Literature review

With the previously defined prospective, scope and business problem in mind, we conducted a literature review in order to gain knowledge on current (state-of-the-art) methods and techniques to solve the before mentioned business problem. Moreover, we aimed at identifying the relations and gaps in research, and how this work would contribute to current literature.

Managing and improving business processes is known in both literature and business as *Business Process Management* (BPM), and can be used to redesign processes [van der Aalst et al., 2003]. With the exponential growth of data, known as Big Data, BPM is becoming even more data driven [Wamba and Mishra, 2017]. The integration of Big Data analytics in decision-making processes has proven to enhance this process [Elgendy and Elragal, 2016]. In particular data science and predictive analytics are promising disciplines in achieving these enhancements for supply chain management [Waller and Fawcett, 2013]. Within ASML, a great amount of different (contextual) data is currently available, such as material (master) data and historical stock data. At present, this information stays underused in the decision-making for the process in our scope. Considering CSCM's ambition to become more data-driven, the focus of this literature review was from a data perspective.

Big Data in operational decision-making

With the evidently growing amount of data coming in the five V's: Velocity, Volume, Value, Variety and Veracity, Big Data is one of the drivers of Industry 4.0 [Zhou et al., 2015]. The field of service and manufacturing supply chain management has been embracing this digitization, and showed how it can enhance (operational) decision-making [Eichengreen and Gupta, 2013, Waller and Fawcett, 2013], in for example sourcing processes, [Sanders, 2016], by applying Big Data analytics. Nevertheless, literature lacks empirical research on applying Big Data analytics in supply chain and/or operational decision-making [Sanders, 2016], especially in the domain of after-sales service logistics [Topan et al., 2020]. Whereas current research focuses on proactively intervening in sourcing and replenishment processes, already ongoing processes are understudied [Topan et al., 2020]. ML is currently one of the most important and emerging techniques in the discipline of operations management in combination with Big Data analytics [Choi et al., 2018]. ML systems have been able to achieve greater predictive performance and, for most of them, increased complexity due to the exponential development in heterogeneous data collection and vast amount of computational power [Jordan and Mitchell, 2015], but still has some pitfalls and limitations, which is explained in the next sections.

Causal Machine Learning

Recent studies [Pearl, 2019, Schölkopf, 2022] raise concerns about traditional ML methodologies that are currently most used in decision-making processes with ML [Hünermund et al., 2022]. Robustness or adaptation is one of the concerns as research in ML has shown that present systems are unable to recognize or respond to new situations for which they have not been specially taught or designed. Another challenge is explainability, as a majority of the ML models remain mainly black boxes and are unable to provide the justifications for their predictions. As a result, this reduces the user's trust or a system developer's ability to diagnose biases. The inability to connect causes and effects is the third concern, which Pearl (2019) considers as "a necessary (though not sufficient) ingredient for achieving human-level intelligence". Pearl (2019) and Scholkopf (2022) argue that these obstacles can be overcome when traditional ML is enriched with causal modelling tools, which can be called Causal Machine Learning (Causal ML) [Schölkopf, 2022]. Recently, a range of studies, see for instance [Bozorgi et al., 2020], [Bozorgi et al., 2021], or [Shoush and Dumas, 2022b], have tried to support decision-making in business processes by applying uplift modelling, which is a group of techniques used to estimate the incremental effect of certain actions [Gutierrez and Gérardy, 2017]. Further, Kallus and Zhou (2018) and Athey and Wager (2021) argue that Causal ML based policy learning, evaluation, and optimization can lead to optimal data-driven decisions. These types of predictive and prescriptive analytics are based on traditional ML techniques but implicate Causal Inference. The induction of Causal Inference in ML can enable the transition from association, i.e. correlation, based reasoning to counterfactual reasoning, which is the ultimatum in causality [Pearl, 2019]. With this counterfactual reasoning, we can also answer questions about interventions and observations, and overcome the earlier mentioned obstacles of traditional ML. Even though the previously mentioned authors showed promising results, they agree that more empirical research should be conducted.

ML Interpretability

The second concern mentioned by Pearl (2019), reflected on the inability to explain or interpret the majority of the currently existing and used ML models. Specifically, a majority of them and their applications lack transparency, interpretability, verifiability, and explainability [Carvalho et al., 2019]. As a result, ML system designers are not aware of biases such as discriminatory decisions against particular people or groups, or wrong decisions in healthcare caused by this bias [Mehrabi et al., 2021, Caton and Haas, 2020]. In the context of this research, Baryannis et al. (2019) suggest that outcomes from AI solutions must be interpretable and justified in the context of supply chains if they ought to be valuable and able to be included into supply chain resource management related decision-making processes. Next to the system developer's need for ML Interpretability, it could also be needed for explaining certain predictions to the end-user in order to get trusted. Explainable Artificial Intelligence (XAI), which focuses research on ML Interpretability and seeks to make a shift toward a more transparent AI, emerged as a field of study in an effort to address this issue, as being

both relevant to industry and society. In consequence, the development and current available ML Interpretability techniques are increasing [Molnar, 2020]. These techniques can also be used for other purposes. Du et al. (2019) defined the three applications: model validation, knowledge discovery, and model debugging/improvement. Despite the increasing attention in research, there are still no well-defined processes and routines to utilize ML Interpretability techniques [Molnar, 2020].

ML prediction abstention

While ML prediction models can achieve excellent predictive performance, they can perform worse for some of the regions in the sample population where it is more difficult to differentiate between classes. Because of uncertainties and noise inherent in any pattern recognition task, errors are generally unavoidable [Chow, 1970]. As a matter of fact, mispredictions can lead to bad decisions, and negatively affect end-users' trust [Yin et al., 2019]. Hence, being careful about samples for which the ML model has a higher uncertainty could be helpful. Even so, Senge et al. (2014) contend that "a trustworthy representation of uncertainty is desirable and should be considered as a key feature of a ML method". Abstaining from a prediction or prescription on samples for which there is high uncertainty seems to accomplish above described goal [Hendrickx et al., 2021, Kompa et al., 2021], and, yet, a closely related field to ML Interpretability [Brinkrolf and Hammer, 2018]. Model abstention because of uncertainty can result in improved predictive performance for the non-rejected samples, enhance a decision maker's trust, and still be beneficial in time saved as only the rejected samples do not benefit from the model [Hendrickx et al., 2021]. Despite its attention in several safety-sensitive, medical and economic domains [Hendrickx et al., 2021, Kompa et al., 2021], operational decision-making processes still remains unexplored.

Synthesis

As previously stated, the emergence of ML has not gone unnoticed in operations management. Nevertheless, there exists a lacunae in research and practice on the use of ML to support operational decision-making, particularly for ongoing processes in sourcing and replenishment processes. Though, the use of traditional ML comes along with obstacles, namely the inability to connect causes and effects, the lack of robustness, explainability and interpretability. Especially the latter is considered critical in the domain of supply chain related decision-making. Using causal inference in combination with ML is deemed to overcome these challenges, which already showed some promising results in predictive and prescriptive analytics for decision support in business processes. Complementary, the development of ML Interpretability techniques allow researchers to open the black-box of ML systems, yet there lacks empirical research on how to use such methods and techniques in practice. Finally, despite the advantages of ML abstention models to deal with ML uncertainty, it has not been investigated in the realm of operational decision-making where the stakes are lower, but the frequency of decision-making is higher.

1.7 Business and scientific relevance

With regard to the scope, prospective, and business problem statement, the business objective is actually two-sided. On the one hand, GOC planners should be supported in the decision-making process on which set of tasks to execute in order to minimize the sourcing time. Next to the development of such a recommendation system, knowledge should be gained about the process and the EMOs themselves. The gained insights could eventually be used to tackle the other causes for the order delays earlier identified in Figure 3, or for redesigning other business processes around the current process scope. This could, for example, help in proactively decreasing the number of EMOs and thereby decreasing the workload. Simultaneously, the exploration of these objectives contributes to the second aspect which is the ambition of becoming more data-driven and living up to CSCM's set dream state.

As explained in Section 1.6, business decision-making processes improvement initiatives are often done with traditional ML models. The neglectful deployment of this type of ML results in less robust models, i.e. are less reliable in new situations, because the real causes and effects are not

modeled. Since ASML and its business environment are highly dynamic and continuously changing, new situations will obviously appear. Involving Causal Inference in such ML models could increase robustness and decrease for example negative impact from wrong predictions or eventual prescriptions on decision-making with high costs in new situations. Despite promising results in earlier work, applying Causal ML for business process improvement is still in its infancy. For this reason, we want to explore how this can be used and how it compares to traditional ML. Moreover, we explained the benefits and necessity of interpretability of the ML models used for decision-making, in particular for supply chain management. Despite this awareness, the use of such ML Interpretability techniques still lacks well-defined routines and processes. In the view of ML model abstention, research showed how this model assertion can leverage the benefit of ML systems in high-stake decision-making tasks. However, the use of this approach in operational decision-making in business processes still remains unexplored, while the earlier mentioned benefits could be shared in this field.

Altogether, recent work has shown promising predictive analytics methods and techniques for accomplishing this road of becoming more data-driven and tackling the earlier stated business problem, but with the absence of complete and detailed preliminary research in this specific context, an exploratory case study ought to be needed.

1.8 Research questions

The research objective of this thesis was to explore how causal and interpretable predictive analytics can support operational decision-making in an emergency maintenance order sourcing process. Specifically, this was done with an exploratory case study conducted at ASML, where we studied how this approach can be used for finding the most effective sourcing solution, in order to reduce the number of redundant checks. This effectiveness was defined as the most desired outcome in a certain situation, e.g., if both sourcing solutions 1 and 2 give a feasible solution, solution 1 is always more desired because of the lower costs and thus most effective.

Main research question
How can causal and interpretable predictive analytics be used to support ASML's global operations center planners in choosing the most effective sourcing solution within the emergency maintenance order sourcing process?

To answer this main research question, four sub-research questions were used, which can be found below. To begin with, it should be explored how Causal ML can be used to predict the most effective sourcing solution. With the identified gap in research on the application of this approach for predictive analytics and the inherently different requirements, the development of such a prediction solution is not a trivial task. Subsequently, we aimed to compare how Causal ML models compare with traditional ML models in the same predictive task as the previous sub-research question. Yet, we liked to know if indeed Causal ML can overcome the concerns earlier mentioned in contrast to traditional ML. Next, we aimed to explore how ML Interpretability methods can be used for model validation and improvement, and knowledge discovery. Finally, after obtaining the best ML prediction model, we aimed to explore how model abstention can be used to improve this model.

1. *How can Causal ML be used to predict the most effective sourcing solution?*
2. *How do Causal ML models compare with traditional ML prediction techniques when applied to predicting the most effective sourcing solution?*
3. *How can ML Interpretability methods be used for model validation and improvement, and knowledge discovery?*
4. *How can the best ML prediction model be enhanced with model abstention?*

1.9 Thesis outline

In this chapter, we explained the business and problem context, the prospective, scope and problem statement. Subsequently, we performed a literature review with which the research objective and questions could be formed. This thesis continues with a more in-depth literature review on related work and state-of-the-art methods and techniques in Chapter 2. Thereafter, the methodology used to answer the research questions is discussed in Chapter 3, and forms the structure of the further report. Next, a general data understanding and preparation is performed in Chapter 4. Following, the four sub-research questions are discussed in Chapter 5, 6, 7, and 8 respectively. This thesis closes with a final conclusion, including recommendations, limitations and future research in Chapter 9.

2 State-of-the-art

After we formulated our research objective and questions, a more in-depth literature review was performed to acquire the needed knowledge for developing the research design and the execution of this study. This chapter starts with a related work part where the current practices in the domain of operational decision-making in supply chain management are discussed. Subsequently, current state-of-the-art methods and techniques are discussed. Finally, this chapter concludes with the methods and techniques that were chosen to be used in this research.

2.1 Related work

The three preceding industrial eras, recently referred to as Industry 1.0, 2.0, and 3.0, have led to Industry 4.0, which was a predicted continuation of those eras [Pereira and Romero, 2017]. From a technical standpoint, Industry 4.0 refers to a situation in which "increasing digitization and automation as well as enhanced connectivity is facilitated by the construction of a digital value chain" [Oesterreich and Teuteberg, 2016]. For many years, the field of service and manufacturing supply chain management has been embracing digitization and showed that this field integrated with Big Data enables the development of better decision-making processes [Eichengreen and Gupta, 2013]. Large financing efforts encourage scholars and practitioners to participate in studies that can advance the development and use of Big Data. As a result, research demonstrated that the use of Big Data analytics can be applied across the supply chain, involving sourcing processes [Sanders, 2016]. However, it still received little attention compared to other domains according to Baryannis et al. (2019). Despite the fact that businesses have an optimistic perception toward Big Data, the literature on Big Data in business is highly scattered and lacks empirical inputs according to Sanders (2016). Next to that, Frank et al. (2019) argued that the implementation of Big Data analytics in manufacturing companies is often poorly done. A deeper comprehension of how Big Data might boost the value creation of supply chain management processes is required to solve this theoretical gap and to provide guidance for practitioners in this domain.

Looking at the scope of this thesis, it can be placed in the domain of after-sales service logistics. Topan et al. (2020) reviewed current practices in operational spare parts service logistics for service control towers. Specifically, they state that simple (business) rules or manual problem-solving based on expert knowledge is currently most used for operational planning problem-solving. Next to that, they argue that the current focus in research is mainly on proactively intervening in sourcing and replenishment processes, and a literature gap exists in processes for orders that have already started. Whereas, identifying the conditions which make an intervention optimal and measuring its consequence would help decision makers to select the best intervention.

Choi et al. (2018) reviewed various existing Big Data-related analytics techniques in operations management, and asserted by stating that this discipline, which focuses on the optimal use of resources to increase operational effectiveness and efficiency, should embrace the chance to adapt to Big Data. The authors emphasize the potential of using ML as a powerful tool to achieve this purpose. Complementary, Bastani et al (2022) studied the applications of different ML methods, including supervised, unsupervised, and reinforcement learning, in various areas of operations management, with goals ranging from descriptive to prescriptive analytics. One of the future directions discussed is using Causal Inference in ML for case-based decision-making in operations management, which can be defined as Causal ML, and will be addressed in Section 2.2.1.

In the context of supply chains, Baryannis et al. (2019) argued that outcomes from AI solutions must be interpretable and justified if they ought to be valuable and able to be included in supply chain resource management related decision-making processes. The urge of this interpretability comes from three reasons given by Molnar (2020): 1) finding meaning within and gaining the knowledge captured by ML models; 2) detecting bias in models; and 3) increasing acceptance of produced solutions, which are directly relevant to supply chain resource management. In Section 2.2.2, a more extensive literature

review on ML Interpretability is given.

2.2 State-of-the-art methods and techniques

2.2.1 Causal machine learning

Due to the exponential growth in heterogeneous data gathering and enormous amount of processing power, ML systems have been able to improve predictive performance and, for a majority of them, increased complexity [Jordan and Mitchell, 2015]. Despite these advances, researchers raise concern about the wide use of these techniques [Pearl, 2019], and not since long ago [Schölkopf, 2022]. According to Peal (2019), there are basically three concerns.

1. One of the drawbacks of traditional ML, is the implicit use of correlation, as correlation does not imply causation. As a consequence, there is a lack of robustness and invariancy in these (prediction) ML models. In changing and dynamic situations, the correlations are still taken as predictors or indicators, while the real cause and effects are not discovered and modelled. This results in less reliable predictions as the model can create spurious relationships, which are eventually translated to predictions. For ASML, set in a continually changing and dynamic environment, this could mean that prediction models take into account features that do not influence a specific target, resulting in bad predictions.

2. Another barrier is explainability, or the fact that ML models are still largely black-boxes and unable to justify the assumptions that went into their predictions or recommendations, also known as ML Interpretability. This undermines end-user confidence and prevents the models from being diagnosed, repaired, or improved by ML system developers. As a result, systems could contain discriminatory biases, resulting in unfair decisions made based on gender or race [Caton and Haas, 2020].

3. The inability to connect causes and effects is the third barrier. To achieve human-level intelligence, this characteristic of human cognition is a required element of ML systems. This component should enable these systems to create a compact and modular representation of their surroundings, question that representation, alter it through imaginative activities, and then be able to correctly answer "What if?" queries. This would allow, as a user, to give interventional inquiries and help in improving decision-making on interventions or choosing from a set of actions. Currently, this is mainly done with correlation based techniques which do not distinguish between cause and effect.

Causal Inference
The process of determining and measuring the independent, true impact of a specific phenomenon that is a part of a broader system is known as Causal Inference [Holland, 1986]. One key difference between causal inference and inference of association (correlation) studies is that causal inference investigates the response of an effect variable when the cause of that effect variable is altered, while the latter studies the relationship between variables without considering the underlying causal mechanisms. Yet, there are some conditions and assumptions that should be met for proper use of Causal Inference in Causal ML [Pearl, 2009, Pearl, 2019].

- **Unconfoundedness**: Causal ML algorithms can be used under the assumption that there are no unobserved confounding factors (variables that directly influence on the target variable). In contrast, the estimates of the treatment effects in the model may be biased in the presence of unobserved confounders.

- **Exchangeability**: Another condition that should be met in order to estimate causal effects is exchangeability, also known as ignorability. It states that the decision to provide a treatment to one individual should not affect the likelihood or choice of administering a specific treatment to another individual.

- **Consistency**: The consistency assumption states that the potential outcome under a specific treatment is equal to the observed outcome if that treatment is actually received.
- **Positivity**: This condition states that a treatment assignment is not deterministic for every sample. This implies that there is a chance for either treatment to be applied to every group of interest.

Literature mainly distinguishes between two frameworks within Causal ML, namely the framework of structural causal graphs and the potential outcome framework, which is explained in the next sections.

Structural causal graph

The graphical representation of the causal assumptions (the causal relationships among the variables) in Causal Inference is called structural causal graphs. Learning and discovering the structure of such causal graphs from observational data is in literature mainly done with the use of Bayesian networks [Pourret et al., 2008]. This tool has been proven to be an effective and versatile tool and has been applied to a variety of research fields. Next to this type of causal discovery, causal graphs can be drawn with domain experts.

The combination of the structural causal graphical framework and ML prediction models is not unknown. Brunk et al. (2021) developed more comprehensible predictions for business processes that consider cause and effect relationships among an event log's variables. By telling the end-user the cause and effects within the predictions, the predictions got more comprehensible. Next to that, they showed that including context variables can increase prediction accuracy. Still, the predictions themselves were based on correlation and not on causation.

Potential outcome framework

Estimating the potential outcome of an intervention or activity with Causal ML is in literature mainly done via the Neyman-Rubin outcomes framework [Rubin, 2005]. The base of the framework is N cases indexed by i. A (process) intervention is considered as a treatment, where $Y_i(1)$ denotes a case i's outcome when it receives the treatment and $Y_i(0)$ denotes a case i's outcome when it receives no or the control treatment. The causal effect of receiving a treatment compared to the control treatment given a case is represented as π_i and calculated with the following formula:

$$\pi_i = Y_i(1) - Y_i(0) \tag{1}$$

Within the population N we can distinguish subgroups, where X_i represents this as a vector of variables. The expected causal effect of the treatment for a subgroup within the population is called the Conditional Average Treatment Effect (CATE) and can be estimated by:

$$CATE : \pi(X_i) = \mathrm{E}[Y_i(1)|X_i] - \mathrm{E}[Y_i(0)|X_i] \tag{2}$$

Conventionally, this CATE cannot be estimated because both $Y_i(1)$ and $Y_i(0)$ cannot be measured. With some assumptions, it can still be estimated indirectly. Nevertheless, in this research both $Y_i(1)$ and $Y_i(0)$ can be measured. This is explained later in Chapter 5. The CATE for a treated case i, is sometimes called the uplift as it estimates the effect of treating the case compared to not treating the case. The modelling of uplifting allows determining which action to take or treatment to apply to optimize the (business process) outcome.

Gutierrez and Gérardy (2017) distinguished three different uplift model approaches, namely: two-model (meta-learners), class-transformation, and direct uplifting. The two-model approach basically builds two predictive models, one exclusively using the treatment group data and the other exclusively using the control group data. The class-transformation uses a class variable transformation which can

transform any single (ML) classification model into an uplift model. Direct uplifting is a pure uplifting approach that allows to directly model the treatment effects.

Most of the innovations with Causal ML that are suggested in literature come from direct marketing, where uplift modelling is used to improve targeted advertising campaigns in terms of both chosen target population and campaign design [Devriendt et al., 2018]. Yet, this approach for prescriptive analytics is getting its attention in BPM initiatives, especially in prescriptive (business) process monitoring [Bozorgi et al., 2020, Bozorgi et al., 2021, Shoush and Dumas, 2022a, Shoush and Dumas, 2022b]. Bozorgi et al. studied how this potential outcome framework, specifically uplift modelling, could be used in predictive and prescriptive process monitoring. In their first research, the outcome was a case-level recommendation system for interventions in a loan application process that maximizes the return-on-investment [Bozorgi et al., 2020]. The authors used action rule mining on event logs as a tool to efficiently select only the rules for which there is high revenue and focus only on these cases. To determine the causal effect of the selected rules, the CATE for each rule was estimated with the help of an uplift model. The paper proposed a prescriptive monitoring method that uses a causal meta-learning approach named orthogonal random forests. Bozorgi et al. (2021) continued this idea of uplift modelling in BPM and showed that this could help reduce process cycle time. Shoush and Dumas (2022a) extended this idea of triggering interventions at run-time while respecting resource constraints. Finally, the authors continued this research by considering whether to trigger an intervention now or later, according to the level of uncertainty in the prediction [Shoush and Dumas, 2022b].

A complementary research field is policy evaluation and optimization [Dudík et al., 2015, Athey and Wager, 2021]. Policy evaluation is to determine the estimated causal effect of a certain policy or intervention. Policy optimization seeks to identify the policy that maximizes the expected total benefits. The goal of uplifting is comparable to learning optimal treatment and policy assignment rules, but the specific outcomes are different since these methods put more emphasis on the causal effect estimation loss rather than not or wrongly intervening loss on the total benefits. Accordingly, the problem can be seen as a causal classification task rather than a causal effect estimation task [Athey et al., 2017, Athey and Wager, 2021], of which the objective equation can be found in Equation 3. Although Cheng et al. (2022) asserted that Causal ML and traditional ML have different learning objectives and, therefore, cannot be compared on the same metrics, the adjustment to a classification problem would allow comparing Causal ML to traditional ML with traditional ML metrics like Accuracy.

$$V(\pi) = \sum_{i} \sum_{t} \pi_t(X_i)(Y_{(i,t)} - Y_{(i,0)}) \tag{3}$$

Concluding, estimating CATE for optimal treatment policies has been proven to be beneficial for predictive and prescriptive (process) analytics for business process outcome optimization. Rather than formulating such Causal ML problems as a causal effect estimation loss, transforming it into a classification task can enhance the goal of maximizing optimal treatment or intervention assignment. Such treatment would in our study be investigating a sourcing solution, while the outcome would be if this solution is the most effective one or not.

2.2.2 Machine Learning Interpretability

As just explained in the previous section, and earlier addressed in Section 1.6, many ML models still remain largely black-boxes while interpretability and explainability of these systems ought to be necessary for value creation in decision-making tasks in supply chain management. Defining the meaning of the term "Machine Learning Interpretability" is not a trivial task. Several definitions can be found in literature, but the one that is most used is "the degree to which a human can understand the cause of a decision" by Miller (2019). Literature also interchangeably uses the term "explainability",

as they are closely related. In the work of Molnar (2020), a distinguishment is made between the terms interpretability/explainability and explanation, where “explanation” is used for explanations of individual predictions. While the goal of the latter is rather justifying a specific decision, the goal of the former is more important for scientific understanding or bias detection [Doshi-Velez and Kim, 2017]. For this reason, the term "interpretability" was more in line with our goal in this research. Du et al. (2019) defined the three main application fields of ML Interpretability as follows:

1. **Model Validation**
   Learned models are often evaluated through the process of performance validation, which is defined as the process of evaluating a given performance metric on a chosen metric such as accuracy. However, Ho et al. (2020) argued that it is crucial to collect domain-relevant features for model inclusion because they point to plausible explanations for the ML model. One should work toward models that incorporate real causal predictors of the result in order to prevent overfitting or non-generalizability problems. To make sure that models do not breach ethical and legal requirements, or rely on irrelevant variables to make decisions, ML Interpretability may be used to determine whether models have used these biases.

2. **Model Debugging**
   ML Interpretability techniques can enable ML model developers in getting insights into how the model works as explained above. Some of the techniques can also be used to analyze the misbehaviour, like mispredictions, and eventually, debug the model. The improvement with debugging can be done in several ways:

   (a) **Model assertions**
   One way to improve and debug ML models is to include model assertions, like constraints, applied to the outputs given by the ML models. These ML model extensions can be exact, applied deterministic functions on model outputs, and soft applied probabilistic functions on the model [Kang et al., 2018]. However, most of the researched methods are applied to non-tabular data, but to more complex data such as video analytics [Kang et al., 2020], which makes the literature rather scattered. It can also be used to let the model abstain from certain predictions. This will be addressed later in Section 2.2.3.

   (b) **Feature refinement**
   From a feature perspective, developers could use ML Interpretability for refinement of the features [Zhang et al., 2019]. One could see if certain mispredictions are made because of specific feature values. This could help in identifying features that could avoid the model from mispredicting, or debugging features themselves.

   (c) **Data debugging**
   Th outcomes of model validation in combination with ML Interpretability can help identify data bugs on local model level for a single or a group of predictions [Pradhan et al., 2021]. This would be the subsequent of bias detection.

3. **Knowledge discovery**
   ML is one of the methods to discover those (difficult) relationships that are not or hardly possible to acquire and statistically test by humans because of for example high dimensionality [Frawley et al., 1992]. Hence, this field forms a method for knowledge discovery [Frawley et al., 1992]. However, with models increasing in complexity and becoming more often black-boxes, it is hard to just tell for humans how the total model works. By applying several ML Interpretability methods, relationships learned by the model can be discovered and used as knowledge [Molnar, 2020].

With a variety of different ML Interpretability techniques, classifying them can be done using different criteria [Doshi-Velez and Kim, 2017, Carvalho et al., 2019, Molnar, 2020]. We highlighted and explained the different criteria chosen to be important for our research in the following paragraphs.

Scope
The part of the prediction process that each interpretability tool seeks to explain can be categorized according to its scope. Algorithm Transparency is about how the algorithm creates the model and what kind of relationships the model can learn. Next to this a priori method, there exist two post hoc methods. On the one hand, there exist we have global methods that help in understanding how the complete model works and makes its predictions. This method can give a general and holistic understanding of the obtained ML model. On the other hand, local interpretability tools are more related to explainability as they aim to demonstrate how a single prediction was made.

Model Specific vs Model Agnostic
Further, we could distinguish between interpretation tools that could only be applied to a specific ML model class, e.g. the tools that can only be applied to tree-based ML models. On the other hand, model-agnostic tools work for any ML model class. These methods are not limited to a specific ML model class, and can be applied to any trained ML model, without knowing the model internals.

Results
The methods can also be distinguished by the results they produce. To begin with, this result could be a `feature summary`, which gives a statistical summary of the features and/or their relationships to each other and to the target variable. Next, the `model internals` can be obtained for intrinsically interpretable ML models. `Data points` is the third explanation result and explains the model or prediction by returning data points. Finally, `surrogate intrinsically interpretable model` are approximated simpler models that could make a complex model more interpretable.

Techniques
In Appendix A, we categorized all current off-the-shelf ML Interpretability techniques according to the above defined criteria. Local approaches' primary objective is justification on individual samples, whereas global methods are better suited for scientific understanding or bias detection. Next to this, we were mainly interested in feature summaries to get a better understanding of the total model working. Accordingly, we chose complementing `Feature Importance`, `Individual Condition Expectation` plots [Goldstein et al., 2015], `Permutation Importance` [Breiman, 2001], `Partial Dependence Plots` [Friedman, 2001], and Shapley Values [Lundberg and Lee, 2017] as ML Interpretability techniques for our research.

Goals
Interpretability of ML systems or their explanations provided by ML Interpretability methods mainly has three goals [Rüping et al., 2006]. The first goal, Accuracy, reflects on the extent to the given explanation provided by the explanation method refers to the actual connection between the prediction made by the ML model, also known as fidelity. The Understandability of an explanation refers to how easily it can be comprehended by an observer. This is an important goal because, even if an explanation is accurate, it will be of no use if it is not understandable. Efficiency refers to the amount of time it takes for a user to understand an explanation, and reflects on the comprehensibility property.

Error Analysis
For model validation and debugging, there currently exist some tools, such as PaLM [Krishnan and Wu, 2017] and Manifold [Zhang et al., 2019], which allow analyzing which features or data points cause mispredictions and biases. Unfortunately, these tools were not available to deploy in Python or were not possible to use for data security reasons. Nevertheless, their methods can be used in this error analysis. They mainly use standard visualization techniques such as scatter plots and bar plots to identify these bugs. Yet, these methods can only take into account a maximum of two or three features per analysis. By reducing the total dimensionality, while keeping all features, clustering techniques can also be used to find bugs [Zheng et al., 2006].

2.2.3 Model Abstention

In the previous chapter, we pointed out that model abstentions can enhance decision-making with ML prediction models. Such model assertion on the ML system is often also called "a reject option" [Hendrickx et al., 2021]. By measuring uncertainty in predictions, the opportunity to reject the ML system to provide a decision which has great uncertainty is enabled.

Uncertainty types
This uncertainty can come from either aleatoric and/or epistemic uncertainty [Hendrickx et al., 2021, Barandas et al., 2022]. Aleatoric uncertainty is caused by unpredictability present in the data, such as the variability and randomness as a result of the data-measurement process. On the other hand, epistemic uncertainty is caused by incorrect modelled relationships because of the lack of knowledge in the model.

Rejection types
Literature mainly distinguishes between two types of rejections that can be made [Hendrickx et al., 2021,Kompa et al., 2021,Barandas et al., 2022]: ambiguity and novelty rejection. The former rejection type reflects on the uncertainty of the model prediction for a specific sample. These types of rejections are inherently dependent on the chosen ML prediction model. The novelty rejection learns to reject samples that are too dissimilar to the used sample population for ML model training. These types of rejections are less dependent on the chosen ML prediction model because the dissimilarity measure can be done with separate ML models.

Model assertion learning
Learning such model assertion model when to abstain can be done sequentially (first training the ML prediction model, and then the model assertion) or simultaneously with the ML prediction model [Hendrickx et al., 2021]. The sequential learning approach allows to extend a previously trained ML model with this reject option. However, this also leads to sub-optimal rejection behaviour because of a lack of bi-lateral dependency during training. This disadvantage can be overcome by using simultaneous learning because the ML prediction model and rejection model assertion are jointly trained. Nevertheless, this brings the drawback for this method of not being able to reuse pre-trained ML models.

Learning objectives
Such model assertion mainly has two goals that should be balanced according to Hendrickx et al. (2021). While the rejection model should improve the ML model's predictive performance, the rejections should not lead to a decrease in the sample coverage such that the ML model becomes useless. Accordingly, metrics should be chosen to quantify and evaluate what is desired and what is not. According to Hendrickx et al. (2021) and Barandas et al. (2022), the most widely used method in classification tasks is the Accuracy-Reject Curve (ARC) [Nadeem et al., 2009]. This curve displays the Accuracy of the non-rejected samples and the sample coverage (number of samples not rejected vs rejected). However, this evaluation method only reflects on the non-rejected samples, while one could also be more interested in the classification quality and rejection quality [Condessa et al., 2017]. Accordingly, the authors defined the evaluation metric "classification quality" as "the correct decision-making of the classifier–rejector, assessing both the performance of the classifier on the set of non-rejected samples and the performance of the rejector on the set of misclassified samples". In addition, the "rejector quality" measures ability to reject misclassified samples. The introduction of both rejector quality and classification quality allows to correctly and objectively evaluate different ML models and rejector combinations. Besides Accuracy, other ML evaluation metrics like F-1 score and Area Under the ROC Curve (AUC) could be used for the same purpose [Hendrickx et al., 2021, Condessa et al., 2017, Barandas et al., 2022].

Applications
The previously mentioned reviews and surveys [Hendrickx et al., 2021, Kompa et al., 2021, Barandas et al., 2022] on the field of ML model abstention predominantly reported applications of this method in safety-sensitive, medical and economic domains. Nevertheless, [Maggi et al., 2014] studied how a sequentially learned ambiguity rejector could enhance predictive business process monitoring. Their work shows how abstaining from a prediction when the class support in the predicted Decision Tree leaf node is below a threshold can improve prediction performance. Though, Maggi et al. (2014) also noted that this could possibly decrease the usability if the threshold is set too low, as the number of cases handled by the recommendation system decreases. As well, Metzger and Föcker (2017) studied the effect of the ML model its uncertainty measures and costs of (in)correct rejections, on the total costs saved. They showed that such an abstention model is beneficial in predictive business process monitoring, albeit the effectiveness of proactive process adaptions and the relative costs of these process adjustments are key determinants for the net benefit of the system.

2.3 Conclusion

In this chapter, we conducted a more in-depth literature review on the related work in our problem context, Causal ML, ML Interpretability, and model abstention. With the acquired knowledge, we were able to decide on the proper methods to answer our research questions.

Firstly, we concluded that problem solving of operational decision-making problems (for orders that already started) is currently mainly done with simple business rules or based on expert knowledge. Similarly, this method is currently used in the manual EMO sourcing process. Nevertheless, ML is pointed out as one of the promising techniques for support in this decision-making with the emergence of Big Data. Among the current ML methods, Causal ML is underlined to be an important future research topic for case-based decision-making in operations. Furthermore, the importance of ML Interpretability of the provided ML solution is punctuated.

Among the available Causal ML innovations, uplift modelling and policy learning and optimization stood out as promising and suitable methods for our research problem. The latter focuses on optimal treatment or intervention assignment which in our case reflects on predicting the most effective sourcing solution. Next to that, the objective function of this method is formulated as a classification problem that enables comparing traditional ML to Causal ML. We, therefore, chose to use this Causal ML method to exploit in our research.

Further, we reviewed several applications and methods used for ML Interpretability. We discussed that global methods are more suitable for scientific understanding or bias detection rather than justification on individual samples, which is the main goal of local methods. For this reason, we chose to use global methods for our third sub-research question on how ML Interpretability can be used for model validation, debugging, and knowledge discovery.

Finally, the reviewed model abstention on several topics ought to be important for designing a rejection model. To complement our study on ML Interpetability, we chose to study how an ambiguity rejector could enhance the best obtained ML prediction model. Dependent on choices later the exact rejector design was specified as of its dependency on the chosen best ML prediction model.

3 Methodology

At the end of Chapter 1, we formulated our main and sub-research questions. The literature review conducted in the previous chapter, allowed us to acquire the necessary knowledge to establish the methodology used to answer the research questions. The section begins by providing an overview of the theoretical framework that serves as the basis for this study. The sub-research questions are then situated within this framework, and the methods used to address these questions are detailed.

3.1 Framework

This research aims to gain valuable insights from data through mathematical and analytical models and applications. More than twenty years after CRISP-DM's introduction [Wirth and Hipp, 2000] it is still the de-facto standard and an industry-independent process model for developing data mining and knowledge discovery projects [Schröer et al., 2021]. The process model is complemented with convenient task descriptions which can be followed as guidelines. Next to that, the life cycle can be iterated multiple times which is suitable for continuous projects and exploratory studies. However, the CRISP-DM framework is still highly generic. For this reason, we used a tailored version of this framework where the deployment phase was left out and an ML Interpretability phase was embedded. With this in mind, we used several different (partial) iterations of this framework to answer our sub-research questions. We indicated the starting phase in the framework for each of the sub-research questions with an asterisk (*). Next, an elaboration on the data analysis preliminaries is given, whereafter each sub-research question follows, and closing with a conclusion.

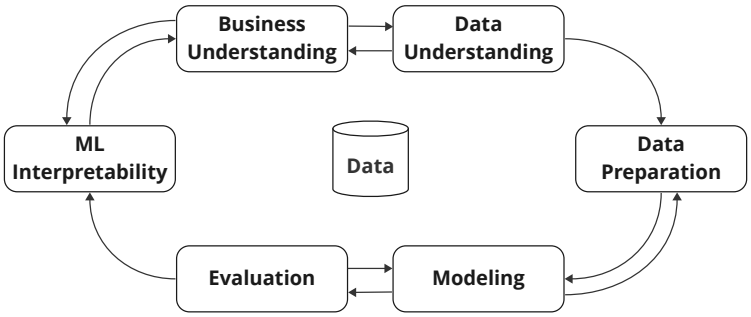


Figure 4: Research framework based on CRISP-DM [Wirth and Hipp, 2000]

3.2 Data analysis preliminaries

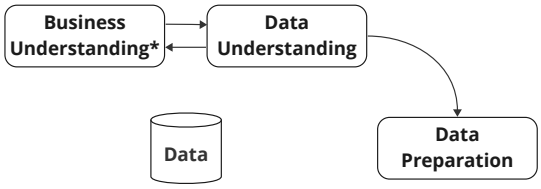


Figure 5: Data analysis preliminaries framework design

We already started this thesis with the Business Understanding phase and continued this in Chapter 4, where we gained more knowledge about the process in scope. In particular, the main objectives for this Business understanding were to identify and formulate which solutions should be predicted and which features to include in our preliminary data set. We did this by interviewing SMEs and stakeholders.

Subsequently, we started with the Data Understanding phase where we collected the initial data. We first determined which SOs fit in our problem scope (an EMO that is sourced by a GOC planner). ASML's CSCM uses a certain software for data visualization and analytics which retrieves data from their databases. The first steps of data collection were performed in this software, since pre-made datasets were online available and could be conveniently integrated, filtered, and transformed. We explored, described, and assessed the data quality to get a better understanding of the data and discussed and validated outcomes with SMEs and stakeholders.

In the Data Preparation phase, we started with a feature selection by examining the features on the inclusion criteria: relevance to the target variable, data quality, uniqueness compared to other features, and technical constraints such as limits on data volume. Next, we selected the samples to include. Missing data for which the true values cannot be retrieved can be recovered in several ways [Allison, 2001]. However, when estimating causal effects, this should only be done if the underlying causal model is sound [Pearl, 2019]. In this phase, we did not know this soundness yet, hence, we removed the samples with missing values.

3.3 Causal ML

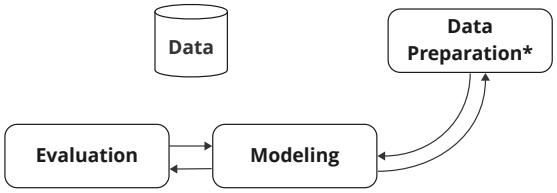


Figure 6: Causal ML framework design

While traditional ML needs predictor variables and one or multiple target variables, Causal ML needs an additional variable that reflects the outcome of a given treatment. This treatment in our problem reflects on the sourcing solution of a certain EMO. With this outcome variable, the potential effectiveness of a treatment (solution) on a certain EMO can be estimated. Since we only had data on which solution had been most effective on an EMO, our outcome variable was sparse, e.g. the same outcome variable for each historical EMO with the treated solution, and it would become impossible for the model to learn causal effects. To address this problem, we created and tested the use of a synthetic control group and synthetically created samples as it would have been an observational study. As well, we addressed the problem of the high-class imbalance present in our data. We choose to focus on resampling as a mitigation strategy to attain better input data for the ML models. As explained in Chapter 2, we decided to use a causal policy learner and optimization method. We selected a technique based on our problem characteristics, and on the availability and deployability, i.e. the technique should be available in a package that is possible to use within Python. Next, we chose a proper test design where we made a trade-off between computational cost and statistical performance. The performance metrics for model assessment were chosen in collusion with stakeholders to meet the business objective and deal with the class imbalance. In the evaluation phase, the best Causal ML model was chosen based on multiple criteria. First of all, the determined performance metrics were used. Secondly, the confusion matrices were analyzed on models' bias directions. Finally, the previous two criteria combined would lead to the model's usability in practice. A remark should be made on the fact that the evaluation method (performance metrics and confusion matrices) is not in line with current literature. As explained in Section 2.2.1 a majority of the Causal ML models focus on causal effect estimations and therefore try to optimize the accuracy of the CATE. However, policy learning and optimization algorithms focus on correctly classifying treatments or interventions to a certain instance. In the context of multi-treatments, such policy optimization would learn which treatment to give for a certain instance to optimize the total benefit of the instances. In our case, this, thus, would be which solution will be the most effective one for a certain EMO. Accordingly, this problem becomes

a causal classification problem. With that having said, we could justify for the dissimilarity to present Causal ML studies, and use performance metrics used for traditional ML. As a matter of fact, this also allowed us to compare Causal ML to traditional ML for the second sub-research question.

3.4 Traditional ML

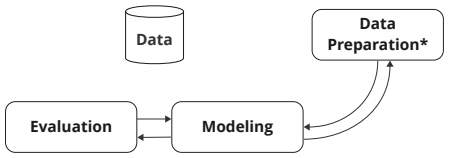


Figure 7: Traditional ML framework design

In this sub-research question, we aimed to explore how Causal ML compares to traditional ML when predicting the most effective solution. Before we selected the traditional modelling technique, we first did some more data preparation with the learned matters from the previous sub-research question. We earlier punctuated the importance of the interpretability of our ML model, and accordingly selected a method that would satisfy this need. Additionally, we created a traditional ML model with Automated ML (AutoML). The inclusion of this method, allowed to explore the use of this emerging approach and see if a better model could be developed without pleasing the importance of interpretability. For the model assessment, a proper test design was generated by trading off the extent of robustness and computational costs. The performance metrics determined in the previous research question were still used for the model assessments. In the evaluation phase, the best prediction model was chosen based on the same criteria as used in the previous research question.

3.5 ML Interpretability

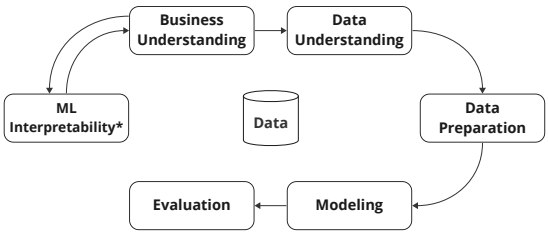


Figure 8: ML Interpretability framework design

With the obtained best ML prediction model, we used several global ML Interpretability techniques to analyze how these could be used for the three application fields defined by Du et al. (2019): Model validation, knowledge discovery, and model debugging. We first started by applying these techniques and analyzed the importance of each feature and its relationship to the target variable. In the meanwhile, we aimed to discover potential Model Debug Opportunities (MDO) which would later be used for model debugging. With the obtained insights from the ML Interpretability techniques, we validated the model working with SMEs and stakeholders. Further, we tested if the MDOs indeed could debug the models and enhance the predictive performance of the best ML prediction model. Finally, we evaluated the utilized methods on the three goals for ML Interpetability defined by Carvalho et al. (2019).

3.6 Model Abstention

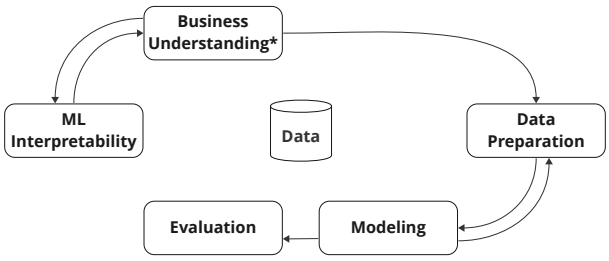


Figure 9: Model Abstention framework design

In this final sub-research question, we wanted to explore how model abstentions can be used and enhance the best ML prediction model obtained. Next to the change in predictive performance, we were also interested in the financial aspect of this abstention. Firstly, we identified the costs associated with the different checks performed on the different solutions. This was done in interviews with SMEs where estimations were made on the time spent on those checks. In the previous chapter, we decided to use an ambiguity rejector as a model abstention method. Since this method is dependent on the chosen ML model, the specific uncertainty measures could only be chosen after this choice of the best ML model. Nevertheless, we first took a step back by analyzing the ML model's uncertainty. Next, we used the earlier chosen performance metrics together with the sample coverage rates to analyze the impacts on these metrics and rates with particular rejection-thresholds. Subsequently, we optimized the thresholds according to the costs saved by the threshold settings. Finally, a comparison was made between the current workflow, the best ML prediction model with and without its model assertion, and the optimal workflow, which was defined as the workflow without redundant checks. The evaluation metrics regarded time saved by each workflow, the coverage of each workflow, and the earlier chosen performance metrics.

3.7 Conclusion

Altogether, we provided a tailored version of the CRISP-DM framework where we included ML Interpretability as a phase and excluded deployment as it does not fit in our research scope. We used this framework iteratively to answer our research questions.

4 Data Analysis Preliminaries

In this chapter, the general Data Understanding and Preparation phases relevant to all other chapters are explained. The outcomes of this chapter form a base for answering the sub-research questions in the next chapters.

4.1 Business understanding

After we defined our problem scope in Chapter 1, the exact sourcing solutions should be defined. Recalling, a SO is the request for all the materials needed to execute a specific action, and contains an item per requested material. One of these items can become an EMO, while the others do not, as a consequence of earlier mentioned issues like material availabilities. To fulfill such EMO, a GOC planner currently follows a static workflow designed to decrease overall high labour and activity costs. The first checks are performed on sourcing solutions that generally do not involve too much work from other departments, and where the lead time is generally short. When these checks show infeasibility of these solutions because there is no stock in a specific storage location, the planner performs checks for other sourcing solutions. These checks are more exceptional and require other teams to check, approve, and/or enable to source this solution. With a multitude of unique combinations of storage location types and plants, predicting the exact solution which can be directly implemented becomes fairly hard due to the low sample size per solution. Consequently, we chose to aggregate them on higher level on storage location type. Below, the eight sourcing solutions are described.

1. **Unrestricted**: The material is sourced from a storage location with no restrictions.
2. **Semi-restricted**: The material is sourced from a storage location with some restrictions.
3. **Predecessor or Successor**: The requested material has a successor which can be sourced, and is accepted by the customer.
4. **Conversion**: The material is sourced from stock on to-be-converted materials.
5. **Direct Service Parts (DSP)**: The material is sourced from incoming supply that do not need conversion at ASML's factory.
6. **Restricted**: The material is sourced from a storage location with severe restrictions.
7. **Repair**: The material is sourced from a storage location with to-be-repaired materials.
8. **Alternative**: An alternative material is used for the EMO.

However, for the solutions `Predecessor or Successor`, and `Alternative` the initial EMO is cancelled and a new EMO is created with the predecessor/successor or alternative as material instead of the original material. Since these EMOs are not directly linked to each other, connecting the cancelled EMO and new EMO becomes a hard task with a lot of uncertainties. Hence, we decided to exclude these solutions from our solution set.

4.2 Data understanding and preparation

Next, we start with the general data understanding and preparation of our problem. As we did not want to put too much emphasize on this phase, we decided to highlight the activities and the outcomes in this section.

4.2.1 Collect initial data

From the data sources in the data analytics software, the collection started with finding one or multiple data sets that contained all SOs with their items. Every EMO is part of a SO which has a sales document number, and a specific item number within that sales document. This combination of sales document number and sales document item number together is called sales document item key and

is used as unique identifier to match the data sets. The first big filter used to scope down the data was filtering on sales order keys which had the priority label 'Emergency'. Next, a pre-made filter was used to exclude EMOs which were not bucketed as GOC order. Since one of the data sources for which this data set was dependent on is only stored for one year, the data ranges from August 2021 till August 2022. At the end of this filtering, it resulted in all EMOs in GOC scope with their unique sales document item key, a material number, the time that the order was created and the time it entered GOC's scope.

Unfortunately, the sourcing solution per case was not yet determined. Though, the solution could be obtained with the use of the storage location where the material at that moment it entered the GOC scope. If for example the sourced material was located at a storage location to be converted materials at the time the GOC was sourcing, it was assumed that the GOC requested a conversion to later ship this material to the customer. Thus, this case could be labelled as sourcing solution conversion. Unfortunately, we could only directly identify the material's storage location when it was ready for shipment which generally are only `Unrestricted` and sometimes `Semi-restricted` storage locations. As a consequence almost all observed storage locations were `Unrestricted` and sometimes `Semi-restricted`. We used a unique equipment number to retrieve all the movements of this equipment throughout the full supply chain. Hence, this allowed us to see at which storage location a specific equipment was at a certain time. To determine the storage location that is sourced from, the time when the EMO was forwarded to the GOC was used. We developed an algorithm in Python to determine this solution.

On a high level, two contextual data categories can be distinguished: demand and material data. The former category contains data about the demand aspect of the order, such as the requesting customer, the maintenance activity type for which the material is needed, etc. Besides the potential predictor features, our target variable, `Solution`, is also stored in this category. The latter category describes data about the material at the moment of sourcing, such as the standard cost price, available stock, etc. The analysis of this data category may uncover discrepancies in the treatment of certain materials or identify underlying causes for the infeasibility of a solution due to a deficiency of inventory.

4.2.2 Data description and exploration

After we collected our initial data, we explored the data to uncover and analyze relationships, and verify the data quality. Next, we performed some necessary data preparation to get the data ready for the analysis in the next chapters. We summarized the activities performed in Figure 10, and highlighted the results in the following sections.

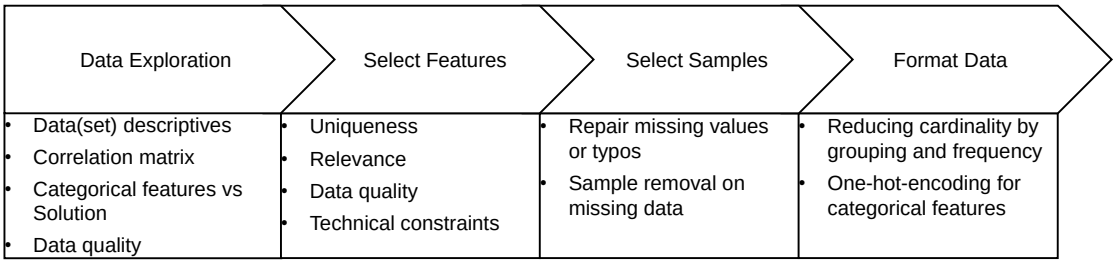


Figure 10: Summary data exploration and preparation activities

Data(set) descriptives
The initial data set contained $40,298$ EMOs, one target variable, and 52 contextual features. This included three Boolean, 12 numerical, and 36 string features. While determining the sourcing solution, every appearing origin storage location was categorized. In total, 136 different storage locations and 65 different plants were classified as equipment origin.

4.2.3 Explore data

To analyze the correlations among the features and target variable, we used a correlation matrix. We observed that most of the features did not correlate more than moderate (i.e. ± 0.49), with a few exceptions. Next, analyzed the imbalance between the sourcing solutions for which the absolute and relative size can be found in Table 2. As can be seen, there existed a high class imbalance in view of the fact that the majority-to-minority class ratios were more than 50:1 [Leevy et al., 2018]. As expected, `Unrestricted` turned out to be the solution for a big majority of the cases in our scope. Nevertheless, after discussing this observation with SMEs and stakeholders, we concluded that the data was not filtered enough. This possibly resulted in orders which are touched by a GOC planner, but sourced by the global emergency support automation engine. Hence, a bigger part of the unrestricted solution cases is probably not sourced by the GOC. Further, we observed for some of the categorical features a high cardinality, which is addressed in the data preparation phase. For the data quality verification, we analyzed the descriptives of the numerical features set, the possible values for each categorical feature, and the missing values for each feature. The results were used in the data cleaning later on.

Size	**Unrestricted**	**Semi-restricted**	**DSP**	**Conversion**	**Repair**	**Restricted**
Absolute size	32879	2524	742	2483	161	989
Relative size	0.82	0.063	0.18	0.062	0.004	0.025

Table 2: Distribution sourcing solutions

4.2.4 Select data

With the results from the previous section, we performed a data selection. Starting with the features, we defined four criteria on which the inclusion or exclusion decision is made. These criteria were: relevance to the target variable, data quality, uniqueness compared to other features, and technical constraints such as limits on data volume. We excluded for each of the criterion one feature. Regarding the samples, we excluded each EMO that did have missing values for which the true value could not be imputed. Besides, we removed duplicates caused by the data integration during the data collection which finally resulted in a population of 35501 samples.

4.2.5 Clean data

A big part of the data cleaning is already done in the previous task by removing features with quality issues. Nevertheless, we performed some other data cleaning activities. We mainly corrected typos and imputed missing data for which the true value could easily be retrieved.

4.3 Construct data

Feature construction
In our data collection, we already gathered many pre-constructed variables such as the criticality of a material in the week of sourcing. Besides, we constructed features ourselves, which can be found in Table 3.

New feature name	**Description**	**Value type**
Time_to_UND_SO	Total hours from SO creation to UND	Numerical
GOC_Entry_time_to_und	Total hours from GOC entry to UND	Numerical
GOC_Entry_weekday	Weekday of GOC entry	Categorical
Day_period_GOC_Entry	Day period GOC entry	Categorical
New_introduced	New introduced material	Boolean

Table 3: Constructed features

Encoding
Encoding features is for some ML models required as they cannot take any non-numeric values like text as input, but encoding can also help to reduce computational time and make better predictions. We used one-hot-encoding for this purpose. Furthermore, some of the categorical features with high cardinality, like `Activity_type`, can take up to 41 different values. After one-hot-encoding, this would result in a total of 41 columns for only this feature. This would increase the computational time, but also the amount of data needed for the model to identify patterns and thus for the model to generalize well outside of the training data, increases exponentially. For this reason, we used three different cardinality reduction methods: grouping on a higher level, focusing on high frequency values, or both combined. Altogether, this resulted in a decrease from 593 to 140 features.

4.4 Conclusion

In this chapter, we first described which sourcing solutions were included in our problem scope. Next, we used several methods to explore and describe the data. During data exploration, we observed a few features that had a correlation of more than 0.49. Later in this research, we address this multicollinearity and analyze its impact. Further, the expected class imbalance became visible where we could label it as a high class imbalance. Additionally, we noticed that a lot of the samples for the majority class contain samples that are not sourced but only touched by GOC planners. Unfortunately, we were not able to filter wrongly included samples out, and had a different sample representation than the defined scope. Also, features with big data quality issues are mainly removed, and typos were recovered. Some features were excluded according to the features' uniqueness, relevance, data quality, or technical constraints. A new feature was engineered which serves as information on material being newly introduced. Finally, data cardinality was reduced, and one-hot encoding was performed for categorical variables to get the data in the right format. The final feature set existed of 39 predictors and 1 target variable, which can be found in Appendix B. The next chapters will highlight additional pre-processing steps taken which are needed for the specific ML methods.

5 Causal Machine Learning

After we performed general data pre-processing tasks in the previous chapter, we first dive into some data pre-processing tasks for Causal ML specific in Section 5.1. Subsequently, the modelling phase is discussed in Section 5.2. Next, the results and findings are discussed in Section 5.3. This chapter closes with a conclusion in Section 5.4.

5.1 Data preparation

As explained in Section 3.3, we need synthetic control samples with which the model is able to choose the most effective solution for a specific EMO. In literature, this problem is solved with so called "Difference-in-differences methods", and in particular often with "synthetic control methods" [Athey and Imbens, 2017]. Their applications are on problems where some groups, such as cities or states, receive a treatment such as a change in policy but not others. These methods then create synthetic samples with statistical models, as the true causal effect of a certain synthetic sample cannot be known. However, in our case, the `Sourcing_solution` assigned to an EMO is by definition the most effective one. With this in mind, we created three methods to deal with this problem and create synthetic control samples, which are explained below.

- **Method 1 (M1)**
  In general, it can be assumed that not giving any treatment would never be the most effective one. Accordingly, all EMOs could be copied and given the solution 'None'. The outcome of samples with a treatment is 1, since it was the most effective one, and for the samples without any treatment this value is 0 since it was not the most effective one. This would, thus, serve as a synthetic control group. We assume that the model would never choose treatment 'None' since it never gives a higher treatment effect than any other solution. Before resampling, this already doubled the sample size.
- **Method 2 (M2)**
  Since we knew that each sample's solution is the most effective, we could also assume that all other solutions are not. By copying each EMO five times, assigning it a different solution per copy with the outcome 0, we tried to give the model this contrast. This made the data set six times bigger before oversampling.
- **Method 3 (M3)**
  This method was based on Method 2. However, instead of following a binary outcome variable, we could also use different values that reflect on the pain or gain when a treatment is chosen that is or is not the most effective one. The pains and gains were rough estimations on the relative time spent on a solution that is not most effective. For example, when the solution `Unrestricted` was the most effective one, but a repair was chosen, the outcome value was -3. This made the sample size six times bigger as well.

Class imbalance
As shown in Section 4.2.2, the data was highly imbalanced on the target variable. However, the distribution of the sample was altered by the synthetic control sampling process. The total samples, samples per solution, and per outcome can be found in Appendix C.1. After the different synthetic control groups were created, this imbalance disappeared for methods 2 and 3. Hence, only method 1 was tested with different resample techniques. Mainly, there exist three methods which can be used for resampling on the target variable: undersampling, oversampling, and combined methods [Japkowicz and Stephen, 2002]. For the reason of computational costs, only undersampling and a combined method were tested. As undersampling technique, Clustered Centroids [Lin et al., 2017] was used as it generally outperforms random undersampling and allowed to control the number of removed samples, which is not the case for some techniques like Edited Nearest Neighbour. This control was required to avoid the combined method to inflate the population size too much. In theory, the sample

set could become almost six times bigger, which would result in high computational costs. Clustered Centroids undersamples the majority class by replacing a cluster of majority samples, calculated with a KMeans algorithm. The majority class `Unrestricted` was undersampled to 5000 samples, which is about 1/3rd of the original class size. The combined method first undersamples the majority class with the used undersampling technique, and afterwards oversamples with Synthetic Minority Oversampling Technique (SMOTE) [Chawla et al., 2002]. This oversampling technique generates new samples by calculating the distances for the minority samples near the decision boundary. In this way, no information was lost, and the models could be better trained. It should be noted that this is done before the previous step where synthetic control samples are made. If this would be done afterwards, resampling methods would keep different samples for each group. The different Causal ML model setups can be found in Table 4.

Model	**Synthetic control sampling**	**Data (resampling strategy)**
M1-O	M1	Original
M1-U	M1	Undersampled
M1-C	M1	Combination
M2	M2	Original
M3	M3	Original

Table 4: Different Causal ML model setups based on the synthetic control sampling and the resample strategy

5.2 Modeling

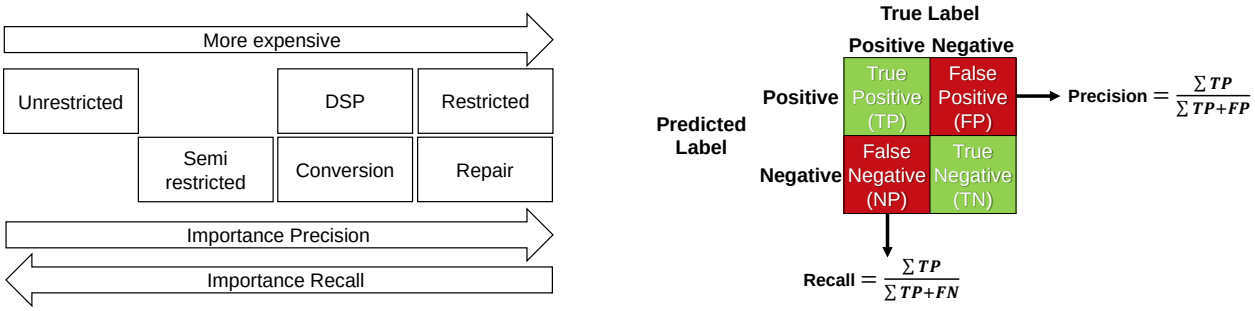
5.2.1 Select modeling technique

In the previous section, we addressed the problem of our missing control group and created three different methods based on the current research problem. To anticipate on potential deficiencies of our methods, we used a Doubly Robust policy evaluation and optimization technique as Causal ML method. This method compensates either a bad model of outcome variables with a good model of past treatments, or the opposite, a bad model of past treatments with a good model of outcome variables [Dudík et al., 2015]. This method first fits a causal effect estimator by using the doubly robust causal effect estimation technique [Funk et al., 2011]. Next, it constructs a multitude of Decision Trees that optimizes the objective function given earlier given in Equation 3, which reflects on the total gain of all training samples $V(\pi)$. As explained before, this optimization problem can be seen as a classification problem where a treatment only was assigned to a sample if it has the highest estimated causal effect. The optimal policy for a certain EMO is then retrieved from the Random Forest. With regard to our problem, this means that the effectiveness of all solutions for a certain EMO were causally estimated, whereafter the solution, which was estimated to be the most effective, would be assigned to this EMO. In particular, we used the Doubly Robust Random Forest Policy Learner (DRRFPL) from Microsoft's python package EconML [Syrgkanis et al., 2021]. Also, this package provides the opportunity to extend the analysis with ML Interpretability tools such as SHAP values.

5.2.2 Generate test design

In this section, we elaborate on the chosen evaluation metrics and the test design generated. The current EMO sourcing process is designed in such a way that the solutions which have a higher chance of feasibility and lower costs are investigated first, and other solutions with a lower chance of feasibility and higher costs are investigated later. Since the exact costs are not known yet and were investigated later in Chapter 8, we could only say, that wrongly predicting and executing a more expensive solution is more costly and missing out on solutions with low costs is not preferable. Accordingly, the lower

the costs of the solution, the more important the Recall (completeness) becomes, and the higher the costs of the solution, the more important the Precision (exactness) becomes. Below in Figure 11, this is visualized. Next to that, it is important to analyze the confusion matrices of each tested model in the evaluation phase to see the model's behaviour in terms of mispredictions. This could for example reveal any biases in the model or data, or variance between the validation and test set.



(a) Evaluation metric importances per class

(b) Confusion matrix

Figure 11: Evaluation metric visualization

After the data was prepared, the model and performance metrics were chosen, and a train-validation-test split method was used. The implementation of this method serves to mitigate the risk of overfitting in the final model, as well as to provide a more robust evaluation of the models compared to using train-test splits alone. Complementary, it allowed finding the best hyper-parameters of the models with the training and validation data, while an unseen test set is available for an objective final evaluation of their performance. The data was split with 75% of the data for training and validation, and 25% for testing, to trade-off low sample size solutions and allowing the models to have a decent training. Within the by EconML provided DRRFPL, stratified K-fold Cross Validation (CV) method is built in the algorithm and is required to deploy. However, the algorithm uses this CV for fitting the causal effect estimators. Even though CV for the Random Forests' hyper-parameter optimization would be a more robust method, this double CV would increase the computational time significantly. Therefore, the training-validation split method is the hold out method. This is again done with a 75%-25% split for training data and validation data respectively.

Since we were using resampling, we should apply this carefully on the right data. Resampling, and especially oversampling or combined techniques, should not be performed on the validation or test set, as the results will generally be too optimistic [Vandewiele et al., 2021]. Unfortunately, the CV function built in the DRRFPL forces us to train and validate the regressors with resampled data. Nevertheless, the hold out validation and test set were not resampled. A visualization of the test design process is given below in Figure 12.

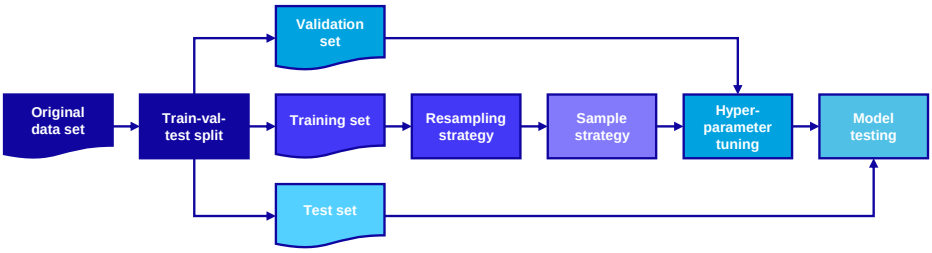


Figure 12: Test design process

5.2.3 Build model

In this section, the model building phase is described where we tried to find the optimal hyper-parameter configuration. The chosen DRRFPL actually contained three ML models: the two regressors used for the doubly robust causal effect estimations, and the RFC for the objective function. Concerning the two inner regressors, we decided to not tune these hyper-parameters as this would require an additional analysis with a significant increase in computational time as well. For the RFC, it is known that they perform well with default hyper-parameter settings [Fernández-Delgado et al., 2014]. However, the hyper-parameters which can be directly visible in the Decision Trees such as maximal depth (max_depth) and minimal sample split (min_samples_split) strongly influence the performance [Probst et al., 2019]. The number of Decision Trees within the forest can be optimized in order to improve performance. However, increasing this hyper-parameter beyond 100 can significantly increase the computational time for the causal models, while the performance gain decreases [Probst et al., 2019]. Therefore, we chose to tune the maximal depth of the trees and minimal sample split, with the possible values $\{2, 8, 16, 32\}$ and $\{10, 20, 30, 40, 50, 60\}$ respectively and keep the default values for the other hyper-parameters.

The hold out validation set was used for the evaluation of the best hyper-parameter configuration, for which the result can be found in Table 5. We compared the confusion matrices and performance metrics of the trained models on the training and test data to try to trade of between variance and bias. However, we observed that both variance and bias were rather consistent between the training and validation data and the models could not really overfit during training. A possible explanation could be, that this is more dependent on the internal causal effect estimators. Remarkably, the training time of M1 was significantly higher than M2 and M3, despite the M1 models had less samples. This could be due to the addition of the control group which serves as an extra class in this multi-class classification problem.

Model	Max_depth	Min_samples_split
M1-O	30	8
M1-U	50	8
M1-C	40	8
M2	40	8
M3	50	8

Table 5: Optimal hyper-parameter configurations Causal ML

5.2.4 Assess model

General performance

The results on the test set can be found below in Table 6. In general, none of the models performed fairly well on all metrics, nor on two of them. Next to this, some of the solutions scored really well on the one performance metric but not on the other, which makes the results look rather scattered. When comparing M1 with M2 and M3, it becomes clear that M1 had a higher Recall on average but lower Precision. When comparing the performance per different solutions, it becomes clear that the solution `Unrestricted` was best classified by the models as it on average has the highest performance in both Precision and Recall. This solution is followed at distance by `Conversion`, `Semi-restricted`, `DSP`, `Repair`, and `Restricted` respectively. In general, we could argue that M2 is performing best regarding the importance of Precision, but has an extremely low Recall. Within M1, M1-U and M1-C have about the same results compared to M1-O, as the Precision is higher for the latter and the Recall is higher for the former. This makes sense as the M1-O had more chance to overfit towards the majority sample class because of the imbalance, but had less chance to do so after resampling. Remarkably, M3-O did not predict any test samples on the classes `Restricted` and `Repair`. A deeper dive is taken into the bias of all models in the next chapter.

		Recall						Precision					
		Unrestricted	Semi restricted	Conversion	DSP	Restricted	Repair	Unrestricted	Semi restricted	Conversion	DSP	Restricted	Repair
M1	O	0.94	0.23	0.54	0.36	0.21	0.38	0.90	0.40	0.56	0.45	0.24	0.23
	U	0.64	0.67	0.68	0.57	0.28	0.26	0.94	0.17	0.32	0.25	0.17	0.22
	C	0.60	0.67	0.72	0.49	0.35	0.41	0.95	0.16	0.34	0.27	0.13	0.09
M2	O	1.00	0.03	0.27	0.11	0.03	0.06	0.86	1.00	0.91	0.70	0.86	1.00
M3	O	1.00	0.02	0.12	0.04	0.00	0.00	0.86	1.00	0.94	0.63	0.00	0.00

Table 6: The results of the different DRRFPL model configurations on the test set.
Note: colour coding indicates how well the models scored on the performance metric for a given sourcing solution. The utmost colours green and red indicate the model performed relatively well and worse respectively on this metric for a given sourcing solution, but do not reflect on overall performance.

Bias

The confusion matrices were used for the bias analysis and can be found in Appendix C.2. In general, all models had the same bias, as for almost all other solutions than `Unrestricted`, the majority of the false negatives were predicted as `Unrestricted`. However, this bias was strongest for M2 and M3. Accordingly, we observed that the Precision is high for M2 for all other solutions than `Unrestricted`, but at the same time a lower Recall of this solution. For M3, the same behaviour was visible for `Semi-restricted`, `Conversion`, and `DSP`. For the M1 models, there also existed a bias from `Restricted` as true solution to `Semi-restricted` and the other way around, which makes sense by looking at purposes in practice. Altogether, this suggests that the synthetic control samples for M2 and M3 gave a better contrast among the different solutions other than `Unrestricted`, with a high Precision as result. However, at the same time, these methods decreased the contrast between `Unrestricted` and the other solutions.

Best model selection

As previously stated, all configurations had similarities and differences in terms of behaviour and performance. M2 outperformed M1 and M3 on Precision for almost all solutions, but its Recall was extremely low for all other solutions than `Unrestricted`. While the Precision is important for these solutions, the models hardly cover any of these solutions. In practice, this means that the model could not support planners on these solutions, which degrades the model's usability. Comparing M1-O, M1-U, and M1-C, the first one had better scores respecting the importance of each performance metric per solution. Therefore, M1-O was chosen as the best performing model. In the next research question, the model is compared to traditional ML. Yet, the results are discussed more thoroughly in the next sections.

5.3 Evaluation and discussion

Bias

The bias discovered in the previous section resulted in a worse Recall for all other solutions than `Unrestricted`. This is not necessarily the worst bias, as this bias is towards the cheapest solution. This bias would cause planners to often investigate this solution first which does not take a long time to find out it is infeasible. Hence, it can be called a "preferential bias" [Rendell, 1986]. In Chapter 8, the impact of this bias in terms of costs is analyzed. Until then, we tried to obtain a better predictive performance. There are numerous different kinds of biases in ML, which can come from different sources. On a high level, Mehrabi et al. (2021) distinguished three possible bias sources: `data to algorithm`, `algorithm to user`, and `user to data`. In our case, `user to data` could not be the source of a bias according to the bias causes explained in previously mentioned work. Next, the possible causes for the biases in our models are explained.

- **Data to algorithm**
  - **Measurement Bias** is a result of the way features are measured or chosen. Our contextual data came from a weekly report and included for example `Stock` at the moment the report is created. This made the contextual factors less accurate and reliable for EMOs which are dependent on a report which is created 6 days ago. To overcome this problem, more accurate reports should be used such as on daily or hourly bases. Next to that, in Section 4.2.1, it was explained how the target variable outcome was determined for each EMO. Currently, the algorithm used to determine the solution takes the time that the EMO entered the GOC scope. It could be that, for a range of EMOs, this time was not the time it was sourced but only sent to the GOC. This could have introduced some noise in the data, which resulted in a bias.
  - **Omitted Variable Bias** arises when not all important variables are included as feature. This could explain the bias, as one of the important variables, namely the estimated time or arrival of a solution, is not modelled. This is in practice an important factor as the material should be as fast as possible at the customer. This variable could be modelled, but then it would need a separate predictive model to estimate this variable. Next to this, (less aggregated) features on stock levels should be included.
  - **Representation Bias** can be introduced when non-representative samples are included, or representative samples are not. As explained in Section 4.2.1, our data also included EMOs which were not sourced but only touched by the GOC. Unfortunately, there was no better filtering possible or available that could filter out EMOs that brought the noise. Nevertheless, this would be a next step to diminish the current bias.
- **Algorithm to user**
  - **Algorithmic Bias** occurs when the input data itself is not biased but is a result of how the algorithm is designed. This bias could have been introduced by the possible hyper-parameter settings such as split criterion, which currently is prone to class imbalance for some models. Due to computational costs, the hyper-parameter grid which was optimized only contained 2 hyper-parameters. A grid with more hyper-parameters could possibly result in a model with less bias. Though, in Section 5.2.3, we observed that the model could not really overfit and that the bias-variance trade-off is more important for the inner causal estimator regressors, which makes the likelihood of the grid extension's success smaller.
  - **Evaluation Bias** can arise from the model evaluation method. Currently, the hold out method for training and validation was used, which is not the most robust method. In the next section, a more robust method is chosen.

Respecting the current (time) scope of this research, and as for the availability and feasibility of the solutions for mitigating the `data to algorithm` bias source, we decided to only tackle the `algorithm to user` bias source in the next chapter. Nevertheless, as mentioned, we advise ASML to investigate the `data to algorithm` before bringing the final ML model to practice.

Effect Synthetic Control Group methods

As we have discussed previously, we found quite some differences in terms of model behaviour between each synthetic control group method. In M1, samples were synthetically created to create a control group. In M2 and M3, samples were synthetically created to pretend it was an observational study. The difference in bias between M1 compared to M2 and M3 could probably be explained by the fact that M2 and M3 have more contrast as a result of the synthetic control samples which reflect on each solution. The difference in terms of Recall between M2 and M3 probably can be explained by the fact that wrongly treating with a cheap solution gives less loss than if an expensive solution would be chosen. The model probably could not clearly distinguish between the solutions, and therefore prefers to choose a cheaper solution. Next to the performance, we observed that the computational

time model training was higher for M1 than M2 and M3. This actually makes sense as both the inner regressors and the RFC have one more class to utilize.

Resampling
When comparing the performance and behaviour among the M1 models, it could be seen that M1-O had a lower Recall than M1-U and M1-C, but not on the `Unrestricted` solution, which was much higher for M1-O. For both resampled samples, we first undersampled the majority class, `Unrestricted`. On the one hand, this could have resulted in information loss for the solution `Unrestricted`, but on the other hand, may have removed noise resulting in the bias towards `Unrestricted`. This could be tackled by less aggressively undersampling `Unrestricted`. Between M1-U and M1-C, there was no overall significant difference.

Correspondingly, we could argue that the resampling did not improve predictive performance because of mainly two reasons. First of all, we earlier argued that resampling should only be performed on training data to avoid over-optimistic results. While the Random Forest was built in this way, the inner causal effect estimators were fitted on fully resampled data. Secondly, the chosen undersampling method removed more than 70 per cent of the samples in the majority class. This could have resulted in too much information loss.

As explained earlier, the inclusion of the new outcome variable has resulted in a different kind of imbalance in the data set, see Appendix C.1. In traditional ML, different strategies exist and have been studied, such as resampling or cost sensitive learning [Haixiang et al., 2017]. While for this new outcome variable, no extensive research is performed on the effect of its imbalance [Devriendt et al., 2018]. In our case, the M1 models did not have any outcome imbalance, which could be the reason that they outperformed the other M2 and M3, which contained an outcome imbalance. The work of [Radcliffe and Surry, 2012] showed that producing resampling in combination with bagging is a promising solution. Firstly, a multitude of differently resampled sample populations are produced, models are built on the different samples, and bagging is used to average on the predictions. This could have been done in our case as well, but then with the majority voting for example.

Causal Inference
As explained in Chapter 2.2, the use of Causal ML comes with some assumptions and conditions. In order to see if we violated any of these, we examined each assumption on its compliance Regarding unconfoundedness and consistency, this could have been violated as explained in the omitted variable bias. The exchangeability condition was probably not violated, because by looking at the number of cases, time and number of different materials, there was a really low chance that this condition is not met (e.g., EMO 1 took 1 stock from repair, and as a result EMO 2 could not). Finally, the positivity condition could have been violated. Our feature `Stock` represents the stock on `Unrestricted` and `Semi-restricted` storage locations, and if this feature's value was zero, the solutions `Unrestricted` and `Semi-restricted` could in fact never be the most effective solution.

In line with the above, Pearl (2019) argued that all variables should be graphically modelled such that assumptions about these causal effects can be tested and made. Further research should thus identify more potential confounding variables, represent them graphically, and test them to make certain causal assumptions. Another possibility to overcome the unconfoundedness violation is by using techniques that need an auxiliary variable (often called an instrumental variable). Even though it is a common way of dealing with non-observed confounders, it needs true causal knowledge for the modelling of such variable [Guo et al., 2020]. Besides, there are currently no policy learners available which include such auxiliary variables.

Be that as it may, Fernández-Loría and Provost (2022) rightly stresses that causal decision-making is not the same as causal effect estimation. In particular, previously mentioned work argued that the causal effect estimations do not necessarily have to be accurate to have a good causal decision-making model, as the objective function in the policy learner is most important. In our case, the objective function did not directly reflect our performance metrics, which could have (partially) caused the bias

towards the majority class. In the same way as for the accuracy of the causal effect estimators, the violation of the unconfoundedness is not bad at all for causal decision-making and, in some cases, can even enhance it [Fernández-Loría and Provost, 2022]. Nevertheless, causal effect estimators should be accurate enough to be useful for causal decision-making. Further research should first analyze how accurate these causal estimators actually are.

As a final point, we observed that the RFC could not really overfit, which suggested that the inner regressors for causal effect estimation would be more important for the bias-variance trade-off. Further research should analyze the dependency with regard to this trade-off.

Interpretability
As explained in Section 2.2.1, Causal ML should allow for better explainability and causal discovery. In Section 5.2.1, we explained that the used Causal ML package could be extended with ML Interpretability techniques such as the model-agnostic technique SHAP, or the model-specific feature importance technique for Random Forests. Unfortunately, the SHAP extension was not yet supported for this specific Causal ML model. Besides, since a policy learner needs different inputs and generates different outputs compared to traditional ML, other model-agnostic techniques could not be conveniently applied. What is more, the observed poor model performances on the test set indicate that the learned relationships do not account for the real nature of the problem. As a result, generated explanations with any interpretability method for the sake of (causal) knowledge extraction would be untrustworthy [Zhao and Hastie, 2021]. This reflects on the Accuracy goal of Interpretability, where the explanation Accuracy of any ML Interpretability method with a ML model with a low predictive accuracy would be low as well [Ribeiro et al., 2016, Molnar, 2020, Carvalho et al., 2019]. Using these techniques with the current model performance would thus only be useful for debugging the model.

5.4 Conclusion

This chapter aimed to explore how Causal ML can be used to predict the most effective sourcing solution. We addressed the problem of the sparse outcome variable by creating and testing different synthetic controls. Even though creating synthetic controls as of it would have been an observational study showed to give more contrast to the DRRFPL among the minority solutions, it resulted in a bigger bias towards the majority solution `Unrestricted`. Hence, M1, which created a synthetic control group, showed results that were more in line with our performance metrics. However, the observed bias for all models was a preferential bias as it was towards the cheapest sourcing solution. In practice, this means that if a planner would investigate an infeasible solution, it would take the least time of all solutions to discover this infeasibility.

Moreover, we discussed several potential sources for the present bias, and concluded that the `Algorithm to user` bias source would be the only source to be tackled in this research respecting the current research (time) scope. Yet, before the final ML model would be applied in practice, the `Data to algorithm` bias should be investigated as well in order to diminish the bias and improve the predictive performance.

Next to this, we tested an undersampled and combined resampling technique. The results suggest that a lot of noise was removed, but also too much information was lost, which could be tackled by less aggressively undersampling. With regard to the performance metrics, the non-resampled training data with the synthetic control group (`M1-O`) was found best performing and was used in the next chapter.

Further, we discussed how the Causal Inference assumptions and conditions were partly violated, which could have negatively affected the overall performance of the models. While previous research argued that unobserved confounders are not necessarily bad for causal decision-making, we argue that further research should investigate if the causal estimators are accurate enough to be used.

In line with the above, we did not focus on the training of the inner regressors which estimate the causal effects used for policy optimization and causal decision-making. In contrast to the proposition from earlier research that causal effect estimators do not have to be very accurate for causal decision-making, we argue that further research should study the dependency in the bias-variance trade-off between the causal effect estimators and the policy optimizer. What is more, the objective of the DRRFPL function was not in line with our performance metrics, which is another possible cause for the poor performance.

Lastly, we discussed that despite the explainability benefit of using Causal ML, the current best ML model could not be used for causal discovery for this purpose as per its predictive performance. Moreover, the poor performance of this predictive model would presumably not be successful in supporting GOC planners in their decision-making because of the many mispredictions. Yet, the `M1-O` method is used in the next chapter, to compare this method to traditional ML methods.

6 Traditional Machine Learning

In the previous chapter, we explored how Causal ML can be used to predict the most effective solution. We ended up with a final Causal ML model that we wish to compare against traditional ML. Specifically, this best model is compared to two traditional ML models. As explained in Section 3.4, we included AutoML in our analysis to see if this could outperform the self-developed traditional ML model. In this chapter, we first use the modelling phase to discuss the used traditional ML models, and compare the results in Section 6.1. Thereafter, a deeper discussion on and evaluation of the models is conducted in Section 6.2. Finally, we concluded this chapter on the findings in Section 6.3.

In the previous chapter, we concluded that reducing the number of samples in the majority class to 5000 likely resulted in a loss of information for that class. In this chapter, the undersampling technique Edited Nearest Neighbour was used without controlling the number of samples removed to prevent this loss of information. Nevertheless, all data pre-processing steps were the same for these methods, with a small exception for AutoML, as is explained in the next section. Accordingly, we start this research question in the modelling phase from our research framework.

6.1 Modeling

6.1.1 Select modeling technique

In Chapter 2, we stressed the importance and need of our ML model to be interpretable, hence, it was chosen to compare Causal ML with a tree-based traditional ML technique. Traditional tree-based ML methods are inherently interpretable but can become black-box if their complexity increases. Though, they allow for better interpretation than black-box methods like Deep Neural Networks. For this purpose, it was chosen to use a Random Forest Classifier (RFC) [Breiman, 2001]. This state-of-the-art ensemble learning method for classification is commonly used because of its predictive performance and interpretability. An RFC creates a pre-set multitude of classifying Decision Trees and selects the class which is predicted most by the trees. We used scikit-learn for the deployment of the RFC in python [Pedregosa et al., 2011].

A second traditional ML model was built with Automated ML (AutoML). This method enables automation of the process of ML tasks. Some Auto ML modules already automate several pre-processing steps like feature selection and engineering, whereas most of them only stick to automation of finding the best model and its optimal hyper-parameters on a certain ML problem [Hutter et al., 2019]. The exploration of this method has mainly two advantages that leverage ASML CSCM department's ambition to realize its dream-state and our research. One potential benefit is that it can save time and resources by automating a process that would otherwise require significant manual effort. This can be especially useful for ASML when it would like to increase the number of ML applications but would be constrained by limited staffing or expertise in ML. Additionally, AutoML can help to improve the performance and accuracy of ML models by automatically searching for and selecting the best algorithms and hyper-parameters for a given task. This could help ASML, and this research, to more effectively leverage the potential of ML to solve complex problems and drive business value. In particular, the inclusion of this method enabled us to see if any black-box or other white box method would outperform Causal ML and the traditional ML method RFC, while not requiring the model to be interpretable. We used Azure as a deployment platform for AutoML to train and test this traditional ML method. The available functionalities mainly cover feature regularization of the data, model selection, and hyper-parameter tuning. The models covered are several simpler classifiers like Support Vector Machine up to more sophisticated classifiers like XGBoosting. An overview of the searched models can be found in Appendix D.1. The optimal model selection and hyper-parameter tuning can be done with a chosen test design and performance metric. Azure also provides the possibility to test the models with different featurization techniques. Accordingly, the data is, before entering the model, automatically transformed to numbers, scaled and normalized to enhance the predictive performance of certain algorithms that are sensitive to differently scaled features.

6.1.2 Generate test design

The main difference with the test design of the previous research question, is the use of stratified k-fold CV for hyper-parameter tuning instead of a holdout method. This could be done now since computational costs were less problematic because of the low sample size of the causal method. Moreover, this method generally results in less biased or less optimistic estimates [Arlot and Celisse, 2010]. A K value of 5 was chosen as a trade-off between computational time and variance from the minority classes as a result of the lower sample sizes [Arlot and Celisse, 2010]. The folds were built with stratification on the target variable `Sourcing_solution`.

As explained in 5.2.2, resampling validation and/or test data gives overoptimistic results. Therefore, only the training data is resampled for the traditional ML method RFC. However, no resampling method is tested for AutoML as the Azure platform does not provide the option for resampling the training data only. Below, the to be tested model set-ups and their names can be found in Table 7.

Model	**ML type**	**Data (resampling strategy)**
RFC-O	Traditional	Original
RFC-U	Traditional	Undersampled
RFC-C	Traditional	Combination
DRRFPL	Causal	Original
AutoML	Traditional	Original

Table 7: The to be tested model set ups

6.1.3 Build model

For both DRRFPL and RFC, an overview of the searched hyper-parameters values is given in Table 8. Compared to the previous chapter, we included the hyper-parameter `Max_features` which is the number of features considered searching for the best split. Lower values lead to more diverse and less correlated trees. Consequently, it exploits features with moderate effect on the target variable and obtains better stability when aggregating [Probst et al., 2019]. AutoML trains, validates, and tests over an extremely big grid per tested model, but limits the choice of performance metrics. In Appendix D.1, the possible performance metrics can be found. AUC weighted was chosen as a performance metric for this method as it calculates the contribution of every target class based on the relative sample size per class, hence is more robust against imbalance [Azure, 2022]. A limitation of AutoML, and this hyper-parameter optimization method and metric is that we cannot observe the results of each fold and choose the model ourselves.

Hyper-parameter	**Values**
Max_depth	{25, 30, 35, 40, 45}
Min_samples_split	{2, 8, 16}
Max_features	$\{\sqrt{n\_features}$, n_features$\}$

Table 8: Grid search for optimal hyper-parameters

In the previous chapter, we observed a strong bias towards the Majority class `Unrestricted` on both the hold out validation set and test set. The use of K-fold CV allowed to better analyze the variance and bias of the algorithms throughout the different folds. For the Causal and traditional ML, the performance metrics and confusion matrices of the training and validation data were analyzed during the CV. Regarding the variance, it was observed that the performance for the solutions

`Unrestricted` and `Conversion` were quite consistent, but the performance of the other solutions generally deviated a lot. This could be the result of two factors according to Raschka (2018):

1. The models tend to overfit on the training data. The CV results showed that the variance in performance increased with an increase in model complexity (high Max_depth and low Min_samples_split). Regarding the bias, we observed that the bias directions did not deviate significantly, but the biases became stronger with a decrease in model complexity. Conventionally, one should balance underfitting and overfitting as a bias-variance trade-off to select the best model [Hastie et al., 2009]. However, this should be considered with care [Belkin et al., 2019]. Rich models that could be considered as overfitted but evidently perform better on the validation set than underfitted models, could in fact be a better model in practice. In our case, we observed substantially higher results for more complex models.
2. The small sample sizes for the minority classes amplify the relative randomness in our training data. Among the folds, we observed a high variance for the performance per class. Particularly, for the minority classes with a low sample size, like `Repair`, the Precision could deviate between 0 and 1. This evidently suggests that this variance (partially) comes from randomness in our training data together with the low sample sizes for some minority solutions.

The performance metrics for each fold for each configuration were averaged to find the final score on these metrics per solution. As we discussed above, we should trade-off the complexity of the models as a balance between overfitting and underfitting, but also consider the substantial difference in performance if overfitted. After analyzing the averaged CV performance scores per configuration, we chose the configurations given in Table 9.

Hyper-parameter	**RFC-O**	**RFC-U**	**RFC-C**	**DRRFPL**
Max_depth	30	30	30	40
Min_samples_split	2	2	2	8
Max_features	sqrt	sqrt	sqrt	140

Table 9: Optimal hyper-parameter configurations

Concerning AutoML, Azure found its best model and hyper-parameters in a reasonable amount of time, compared to Causal ML, of 1 hour and 22 minutes. The best and worse model scored 0.89885 and 0.5 respectively on the performance metric AUC weighted. The best model chosen by AutoML is a voting ensemble which is an ensemble classifier learning model that combines the predictions from multiple other ML models, for which the predictions are summed and the class with the majority vote is used as prediction.

6.1.4 Assess model

After the hyper-parameters were tuned with and the best configurations were chosen, the models' performances on the test set could be analyzed. The results can be found in Table 10.

		Recall						Precision					
		Unrestricted	Semi restricted	Conversion	DSP	Restricted	Repair	Unrestricted	Semi restricted	Conversion	DSP	Restricted	Repair
RFC	O	0.99	0.20	0.58	0.36	0.10	0.12	0.89	0.74	0.86	0.65	0.65	0.80
	U	0.96	0.37	0.67	0.46	0.18	0.18	0.92	0.48	0.72	0.53	0.49	0.86
	C	0.94	0.42	0.68	0.47	0.18	0.18	0.92	0.40	0.67	0.50	0.40	1.00
Causal ML	O	0.94	0.21	0.54	0.35	0.22	0.38	0.90	0.39	0.57	0.47	0.23	0.22
AutoML	O	0.98	0.26	0.63	0.37	0.13	0.15	0.90	0.65	0.82	0.61	0.60	1.00

Table 10: The results of Causal ML and traditional ML on the test set.
Note: colour coding indicates how well the models scored on the performance metric for a given sourcing solution. The utmost colours green and red indicate the model performed relatively well and worse respectively on this metric for a given sourcing solution, but do not reflect on overall performance.

General performance

In general, the Recall and Precision did not deviate significantly for the `Unrestricted` solution, which was also observed in the previous chapter. For the other solutions, we see that the Recall and Precision for the other solutions deviate greatly among the tested methods.

Bias

Yet again, there existed a high bias towards the `Unrestricted` solution for all tested models, which was concluded from the confusion matrices given in Appendix D.2. This bias was less strong for the DRRFPL, but the mispredictions were now on the other solutions, resulting in a lower Precision for the other solutions. This would in practice mean that planners investigate more expensive solutions first which are not the most effective. The previously observed small bias from `Restricted` to `Semi-restricted` and the other way around in DRRFPL is partly shared with RFC and AutoML. Diversely, the `Semi-restricted` solutions were less often mispredicted on `Restricted`, especially for the resampled models. Hence, traditional ML could better distinguish between those solutions

6.2 Evaluation and discussion

Resampling

In the previous chapter, we concluded that the resampling strategies caused a loss in information of the majority class. Hence, we changed the undersampling method to a less aggressive method. However, as we could see in the performance metrics and confusion matrices, it showed about the same behaviour but less strong. For the majority class `Unrestricted` the Recall decreased and the Precision increased. For the other classes, the Recall increased, and the Precision decreased. Except the class `Repair`, where both increase when the population is resampled. A possible explanation for the repeated failure of resampling could be because resampling techniques are generally developed for binary classification problems and, as a consequence, multi-class classification problems become harder to resample [He and Garcia, 2009,Wang and Yao, 2012]. In literature, our multi-class imbalance problem is called "multiminority", as there exist multiple minority classes and one majority class [Wang and Yao, 2012]. The authors of previously mentioned work concluded the following: "Oversampling does not help the classification and causes overfitting to the minority classes with low Recall and high Precision values. Undersampling is sensitive to the number of minority classes and suffers from performance loss on majority classes.". The first statement was in our case not completely applicable, as we first undersample our minority samples. However, the observed behaviour in our study supports the second statement on undersampling. Altogether, we conclude that this strategy to deal with class imbalance did not improve the predictive performance in our problem.

AutoML

According to the model assessment, AutoML performed about the same as the RFC-O, and both methods outperformed Causal ML. As explained before in Section 6.1.3, one of AutoML's limitations

is regarding the hyper-parameter optimization. We had to choose one performance from a limited list, and we could not see the performance of each fold. As a result, the model and its hyper-parameters did not outperform the RFC on our determined performance metrics. From this point of view, we suggest that the practice of AutoML should preferably be used if the chosen performance metric matches your performance metric, and is solely one. Another, by Microsoft known, drawback of AutoML (in Azure) is the inability to inherently deal with class imbalance. Even though they state one could resample before using AutoML, this makes the possibility of using K-fold CV less appropriate. The authors of [Zöller and Huber, 2021] argue that one of the other drawbacks is interpretability. While it can choose less intrinsically interpretable models, the models and its hyper-parameters can be extracted and deployed in an integrated development environment and made interpretable there with agnostic methods like SHAP. Nevertheless, the convenience and the achieved performance of this method shows high potential. Hence, this method could be used by ASML to save time on model development when the problem is not limited by the earlier mentioned obstacles of this method.

Best model
In the previous sections, we explained how the performance deviated between the traditional and Causal ML, but in the same way how the traditional ML methods perform similarly. Next to that, the significantly better performance allows for better knowledge discovery on this problem with ML Interpretability. Further, we showed that resampling did not improve predictive performance for traditional ML, and the non-resampled data set showed the best results. Although one should pick the simpler model in case of the same performance according to the Law of Parsimony, which is also known as Occam's Razor, this should always be justified [Elder, 2018, Raschka, 2018]. In our case, we prefer RFC over the AutoML selected voting ensemble model, because of its implementability and model specific ML possibilities. Accordingly. we selected the RFC-O as the best model. Though, we would like to stress that the model is failing to classify a significant number of cases correctly, hence planners would make sub-optimal or bad decisions. Moreover, poor accuracy can erode the end-users trust [Yin et al., 2019].

Bias
As mentioned in the previous chapter, a possible source for the bias could be the `algorithm to user`. In this chapter, we increased grid size for the hyper-parameter tuning and used K-fold CV and could search if less complex models would result in less bias. Since we still obtained the same bias as we did in the previous chapter, we could argue that the bias is not coming from the `algorithm to user`, but rather from `data to algorithm`, see Section 5.3 for an explanation on both definitions. Besides resampling, research argues that feature selection is the other method on data level which can help to overcome the effect of class imbalance on the bias [Ali et al., 2013, Leevy et al., 2018]. This mainly is a result of the matter of class complexity, also known as class overlap or class separability which describes how distinguishable the classes are from one another within the data [Ali et al., 2013]. The authors of previously mentioned work argue "When overlapping patterns are present in each class for some feature space, or sometimes even in all feature space, it is quite hard to determine discriminative rules to separate the classes. The overlapping feature space caused the features to lose their intrinsic property thus making them redundant or irrelevant to help recognize good decision boundaries between classes". Consequently, a classifier's generalization capability can be negatively affected in terms of bias by this class imbalance. In the next chapter, we analyzed how this bias could be reduced with feature selection based on several model agnostic ML techniques. This stresses the importance of investigating and mitigating this bias before applying this model in practice in order to give unbiased support with fewer mispredictions to planners.

Causal ML
As explained earlier in Section 2.2.1, comparing Causal ML and traditional ML is usually not possible on performance metrics used in traditional ML. However, we justified the comparison of these traditional ML metrics as policy learning and optimization is rather a classification problem. In the previous chapter, we argued that the used objective function of this optimization method did not reflect on

our chosen performance metrics, which could have caused the bias and disappointing performance. In this Chapter, we compared Causal ML to traditional ML, and observed somewhat the same behaviour in terms of bias among all methods. Further, the variance observed between the validation sets and the test set was smaller for Causal ML. This indicates that the model is more robust and is doing a better job in generalizing, which is one of the flaws of traditional ML as explained in Section 2.2.1. Nevertheless, the results showed how traditional ML outperformed Causal ML in terms of predictive performance. Fernández-Loría and Provost (2022) argued that if the noise in the causal features is greater than in the non-causal features, traditional ML techniques may be more effective than Causal ML approaches. Reflecting on our case, this could make sense. For example, The causal feature `Stock` in our data is not accurately measured, which may lead to less accurate results when using Causal ML. On the other hand, the feature `Criticality` is more accurate and may lead to better predictions when using traditional ML. Lastly, as explained in Section 5.3, the current performance of Causal ML would make the explanations less useful, and the currently used package does not yet support model-agnostic ML Interpretability methods. By way of contrast, traditional ML has a higher performance and can be used with model-agnostic methods.

6.3 Conclusion

In this chapter, we aimed to compare a Causal ML method with traditional ML methods when applied to predicting the most effective sourcing solution. Besides, we tested two different resampling methods to overcome the negative effects of the high class imbalance. The models were trained and validated with the more robust stratified K-fold cross-validation method.

In particular, we tested two resampling techniques to deal with the class imbalance and to mitigate its degrading effect on model performance. Altogether with the previous chapter, we concluded that aggressive and less aggressive undersampling, and undersampling and oversampling combined methods did not enhance predictive performance. Even though, the minority classes' Recall increased, its Precision, which we find more important for those classes, decreased significantly. Additionally, the Recall for the majority class dropped, which is undesirable as well. Though, by undersampling less aggressively, strength of the bias from the true label `Unrestricted` to other labels decreased, compared to the undersampling in the previous chapter. Nevertheless, resampling did not improve the predictive performance of the RFC, which could be due to the fact that this strategy to mitigate class imbalance is less effective for multi-class problems.

In Section 5.3, we explained several potential sources for the present bias. By increasing our hyper-parameter grid and validating the models' performance with the more robust stratified K-fold CV, we tried to reduce this bias coming from "Algorithm to user" bias. Unfortunately, this did not result in a significant decrease. Hence, this suggests that the bias is rather coming from `data to algorithm`, which is addressed in the next chapter.

Similarly as in the previous research question, the Causal ML model achieved a low performance. Besides the possible explanations given in the previous chapter, we argue that the noise present in important features for Causal ML made these features less accurate than the important features for traditional ML. Nonetheless, we observed some similarities in the methods' behaviour, like the bias towards the majority class. This, again, indicates that the bias is coming from `data to algorithm`. From a business perspective, we discourage using this approach of Causal ML in predicting the most effective sourcing solution. This method performed rather poorly in terms of predictive performance, and for this reason becomes unreliable for knowledge extraction as well.

On the subject of AutoML, we observed that this method can obtain about the same performance as the RFC with the same bias present. The limited choice of the to-be-optimized performance metric, and the inability to use k-fold CV with resampling in Azure AutoML, could have hindered the development of a better model than the manually developed RFC. We recommend utilizing this method if the predictive modelling problem is not constrained by these factors.

7 Machine Learning Interpretability

In the previous chapter, the best predictive ML model was chosen. For sub-research question three, we aimed to explore how ML Interpretability methods can be used to validate the model, discover the relationships modelled, and improve the model with the gained insights. These three applications all contribute to a more valuable model for decision-support because the explanations and model behaviour can be justified as well as it can leverage the performance. This chapter starts by applying several ML Interpretability techniques to gain insights into the modelled relationships. In addition, we performed an error analysis to gain insights into the data regions where the model poorly performed, which can be found in Section 7.4. Eventually, the insights obtained until then are discussed and validated in Section 7.5. Throughout these analyses, multiple MDOs were identified. We trained MDO based ML models and tested them to see if they could improve the predictive performance compared to the best model from the previous chapter. The results of this analysis can be found in Section 7.6. Furthermore, we evaluated the utilized methods in 7.7. Finally, the chapter concludes in Section 7.8.

Before the methods and their outcomes are discussed, we would like to throw light on the methodology used within this chapter regarding the data. Firstly, we split our data into three sets: training, validation, and testing. We trained the ML model on the training data, applied interpretability methods on the validation data, and compared the performance of different model debug opportunities (MDO) on the test data. No cross-validation was performed at this stage. We did this for mainly two reasons: 1) It would be unfair to use the test set to look for improvements and test the model again on that test set. A fairer approach would be to identify model improvements based on a validation set and test the performance on the original test set. 2) For some of the ML Interpretability techniques, using the training data for interpreting the model could give a too optimistic measurement as it is trained on this data. Hence, it would be better to use unseen data for this purpose [Molnar, 2020].

7.1 Model-specific Interpretability

As explained in Chapter 2, model-specific interpretation methods are limited to specific model classes, which in our case is based on the inner estimators of the RFC: Decision Trees. Decision Trees are easily interpretable because their logic can be traced from the root node to the predicted leaf node. However, as the tree becomes more complex, its interpretability decreases. In the current situation, the trees are too complex to interpret as a whole system. However, the feature importance based on the split criterion in the Decision Trees can still be used to understand the global model.

MDI Feature importance

The intrinsic feature importance measure in Random Forests is based on the Mean Decrease Impurity (MDI) for the chosen splitting criterion [Breiman, 2001], which in our case is the Gini coefficient. This technique computes the normalized total reduction of the Gini coefficient brought by a particular feature. The 20 features with the highest MDI feature importance can be found in Figure 13.

5!The feature importances were generally low, indicating that no single feature is a strong predictor for the solution. This may suggest the presence of multicollinearity among the features. Of the top 13 and 14 out of 20 displayed features, most were numerical, indicating that categorical features were generally less important. Additionally, several correlated features had similar feature importances.

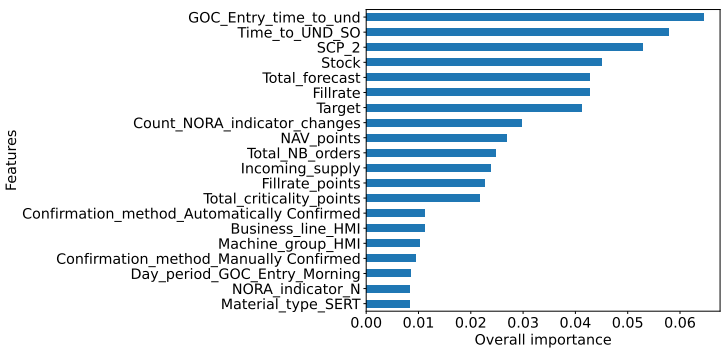
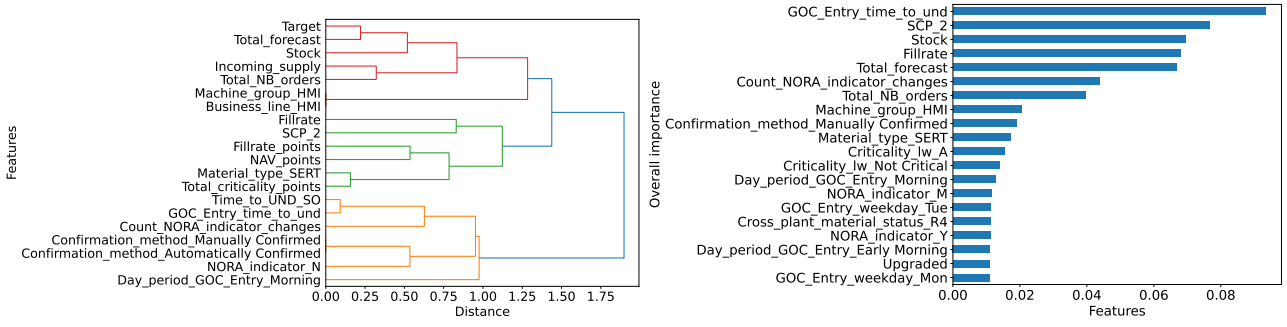


Figure 13: MDI feature importance

The three observations are in line with the drawbacks when evaluating feature importance with this model-specific technique. Research argues that this technique suffers from mainly two biases [Strobl et al., 2006, Wei et al., 2015]. Firstly, this technique tends to inflate the feature importance of features with a high cardinality. These features have a higher chance of appearing more than once in an individual tree and could result in an increase in their importance. Consequently, our one-hot-encoded categorical features seemed to be less important. The authors of the previously mentioned work suggest that Permutation Importance (PIMP) would be a better technique for feature importance identification. Secondly, in the case of multicollinearity (correlated predictor features), the feature importance of one or more correlated features could be shown as weak as it can select one of the features and neglect the importance of the other one. In the next Section, this problem is addressed by removing correlated features with hierarchical clustering on Spearman rank-order correlations.

7.2 Hierarchical clustering correlations

To overcome the pitfall caused by multicollinearity, a feature selection was performed on the predictor features' correlations. This was done by hierarchical clustering the features on their Spearman rank-order correlations [Dormann et al., 2013, Azpiroz et al., 2021]. Subsequently, a threshold was chosen for which only a single feature from each cluster was kept if their cluster distance was below this threshold. The main advantage over pairwise correlations is the fact that we can observe clusters of correlated features instead of only two correlated features. A visualization of the 20 most important features hierarchically clustered on their correlations is shown in Figure 14a. The choice of threshold was based on visual inspection of the graph, where a trade-off was made between loss in information and collinearity. A value of 0.5 was chosen which allowed to exclude highly correlated features for which the removed feature did not add too much value compared to the kept feature. This threshold was not only used for the 20 features with the highest importance, but also for the total feature set. The feature reduction resulted in a decrease in the feature set from 140 to 99. Consequently, this also formed the first potential MDO and is called `MDO-1`. The 20 features with the highest MDI feature importance after the feature selection are shown in Figure 14b. As can be seen, this removed the feature `Time_to_UND_SO` which was correlated with `GOC_Entry_time_to_und`. Consequently, the feature importance of the latter increased. Yet, there were no features that have high importance, but the 14 highest features from Figure 14b show to have substantial importance compared to the other features. We, therefore, selected these features for the second MDO: `MDO-2`.



(a) Collinearity hierarchical clustering dendrogram **(b)** MDI feature importance

Figure 14: Feature importance analysis after feature exclusion

The removal of some features also caused a loss of information. To recapture this lost information, one additional feature was engineered. This feature was the difference between the features `GOC_Entry_time_to_und` and the previously removed `Time_to_UND_SO`, and was called `Time_to_UND_diff`. Consequently, we could capture for example if an EMO instantly got sent to the GOC or if the SO already existed for a longer time. This new feature was added to the reduced feature set and raised the third MDO: `MDO-3`.

7.3 Model agnostic methods

Model-agnostic techniques are applicable to any ML model and are applied after the model has been trained (post hoc). These methods can give different insights and overcome the bias of the MDI technique towards features with high cardinality. In the next sections, three different methods are discussed.

Permutation Importance

Permutation Importance (PIMP) is an agnostic feature importance technique that measures the difference in prediction performance after a feature's value is permuted [Breiman, 2001]. This is done for all features, whereafter they are ranked on the negative impact they have on the performance. One of the drawbacks of this technique in our case was that the importance is based on one performance metric. Since we did not have one single performance metric, balanced_accuracy was chosen, which is calculated as the average of the individual solution accuracies, with each solution being weighted equally. A higher balanced accuracy can be seen as a sign of the model's robustness and reliability, and can help the planners make better decisions based on the model's predictions. The results are shown in Figure 15.

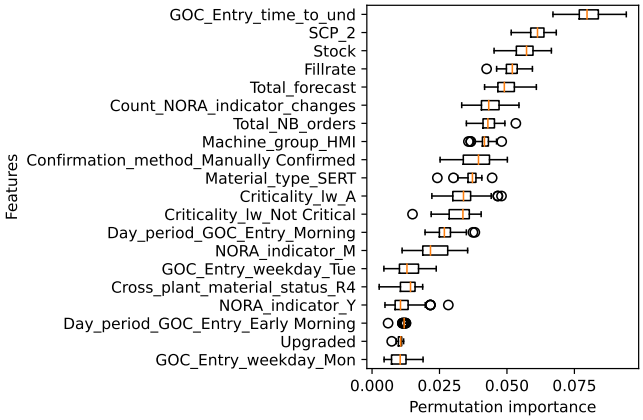


Figure 15: Permutation importances

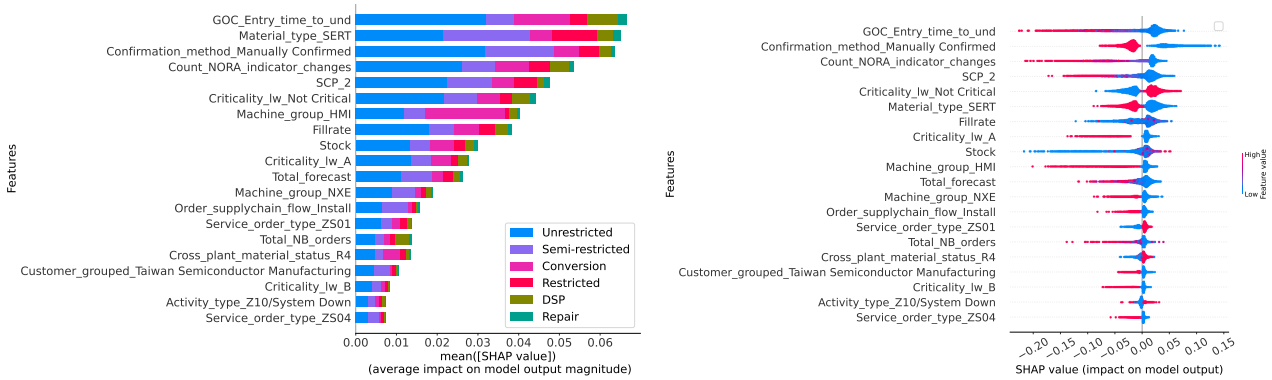
When analyzing the results, we could again see that the (permutation) feature importances were not that high and none of the features was particularly strong predictors of the target variable. Furthermore, the standard deviations of the PIMPs were overall the same and neither high nor small. This suggests that the feature informative levels do not explain most of the variance (changing a feature's value can influence the feature's importance). Accordingly, this means that the model could be overfitted, because of the moderate variance. As fourth feature selection, we chose to select the features with substantial feature importance raised by this model agnostic technique for the fourth MDO: `MDO-4`. This accounted for the 14 features with the highest feature importances.

Shapley

By calculating the contribution of each feature to the prediction, SHapley Additive exPlanation (SHAP) seeks to explain the prediction of an instance [Lundberg and Lee, 2017]. Shapley values are calculated using the SHAP explanation approach which uses coalitional game theory. This technique starts by locally computing the SHAP values but can be used globally by summarizing on class and problem scope.

In Figure 16a, the SHAP feature importances per solution can be found. In line with the previously used methods, there were no features that predominantly determined the solution. Next to that, about the same features and their importance order appeared in the 20 highest ranked features. Moreover, the feature importances per solution can be analyzed. As can be seen, the features are generally most important for the majority class `Unrestricted`. Among the minority classes, features seem not so important for `Restricted` and `Repair`. This was not a surprise as the model did not have a high predictive performance for these features, and is not able to model their relationships properly. In order to see how the value of each feature per solution impacts the model output, beeswarm plots were made, which can be found in Appendix E.2. As illustration, the beeswarm summary plot of `Unrestricted` can be found in Figure 16b. We used these plots later during model validation. The

18 features with the highest importance were chosen to be included as feature selection for the fifth MDO: `MDO-5`, as they showed to have substantial feature importance compared to the others.



(a) SHAP feature importance summary plot **(b)** SHAP beeswarm plot `Unrestricted`

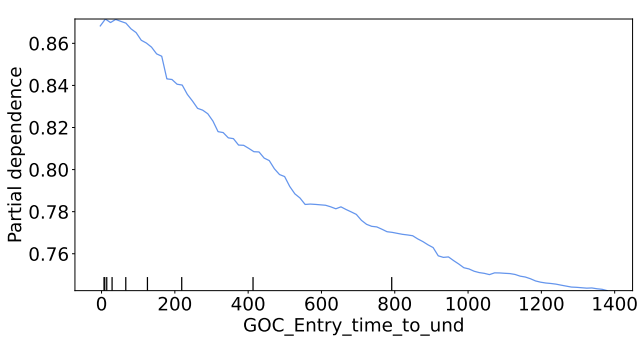
Figure 16: SHAP analysis

Feature importance overlap

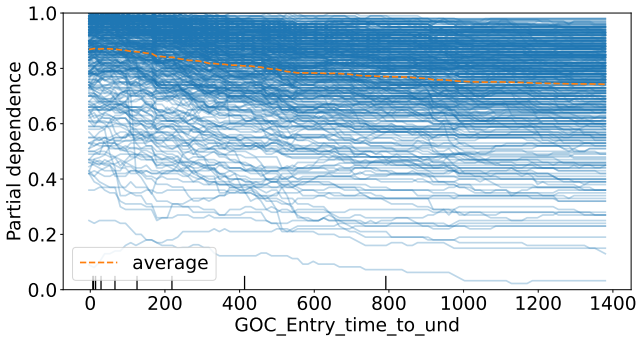
In order to compare the obtained results, we summarized the top 10 ranked features per feature importance method, which can be found in Appendix E.1. We observed that the different methods do not completely agree on the exact feature importance rankings, but show a decent overlap. The fact that different feature importance measures result in different rankings is not unknown and is even likely to happen [Rajbahadur et al., 2022]. Rajbahadur et al (2022) reported that this is especially the case when comparing model agnostic and specific methods, because of feature interactions. In order to mitigate this effect, these feature interactions should be reduced. The authors hypothesize that when this is done, SHAP and PIMP would strongly overlap. In our case, we reduced the feature interactions by hierarchical clustering on correlations, but the model agnostic methods did not overlap more than with the model-specific MDI. Particularly, MDI and PIMP had a much greater overlap compared to SHAP with these techniques. This contradiction could possibly be explained because the feature interactions were still too strong as a result of choosing a too low threshold for removal.

Partial Dependence Plot

Up until now, we obtained the feature importances (for each solution), and analyzed the impact of the features' values on the model output. Supplementary, Partial Dependence Plots (PDPs) were used to describe the relationship between the target variable and the features. This tool enables us to comprehend how various values of a specific attribute affect the predictions made by the model. Thereby, it demonstrates if the predicted class probability of a solution and a feature have a linear, monotonic, or more complex relationship. One drawback of this method is the moment a feature could be really important according to other methods such as PIMP, but the PDP could be flat since the feature affects the prediction mainly through interactions with other features. To overcome this drawback, Individual Conditional Expectation (ICE) plots can be added [Goldstein et al., 2015,Molnar et al., 2020]. This local agnostic technique shows the partial dependence of an individual sample when the value of a selected feature changes, and can show how individuals differ from the averaged partial dependence line. However, both PDPs and ICE plots become less reliable for values for which fewer data points were available, as such that these regions should be interpreted with care. In the plots, the relative distribution of the data on the features can be found as lines on the x-axis. We made both PDP and ICE plots for the 10 features with the highest importance according to the SHAP feature importance summary plot in Figure 16a. The results can be found in Appendix E.2, and an illustration of both PDP and ICE plot can be found Figure 17.



(a) PDP GOC_Entry_time_to_und for `Unrestricted`



(b) ICE plot GOC_Entry_time_to_und for `Unrestricted`

Figure 17: PDP and ICe analysis

We mainly observed monotonic relationships between the features and the predicted class probability of the solutions being the most effective ones. This indicates that there were hardly any complex relationships for features to predict the solution modelled. The ICE plots allowed to see if there was a high variance in terms of partial dependence per sample. For some features, there was some variance visible, but none of the features' PDP was strongly affected by this as such that the partial dependence seemed smaller. Hence, there was not a lot of variance in the partial dependence per feature value on the samples. This suggests that the features influence the prediction in the same way.

7.4 Error analysis

As explained in Section 2.2.2, error analysis can be used to debug ML models. We used well known visualization techniques utilized in debug tools to discover subgroups with significantly more mispredictions. The used techniques and the findings are discussed below, and an illustration of each technique can be found in Appendix E.3.

1. **Box plots**
   We first made box plots for the count of mispredictions for each feature on all true solutions together. Unfortunately, this did not show any significant difference between subgroups, as the distributions of correct and mispredicted per feature were generally the same.

2. **Box plots per true solution**
   As the previous box plots showed an aggregated view and did not give any insights, we used the same technique for the solutions separately. Yet, this gave no useful insights as well. Some differences were observed among the solutions, but these were not significant due to the small sample size for some feature solution combinations.

3. **Pair plots on features**
   In order to see if the relationship between two features could show any subgroups, we made pair plots for the 20 most important features. Unfortunately, this did not provide any useful insights as well.

4. **Principal Component Analysis 3d plot**
   Lastly, we reduced the feature set to three dimensions with a Principal Component Analysis (PCA) [Tipping and Bishop, 1999, Zheng et al., 2006]. After analyzing the errors on this reduced feature set, we could retrieve the features correlated to the errors. A 3d plot was made, but this unfortunately again did not give any useful insights as well.

In this section, we tried to identify subgroups for which significantly more mispredictions were made. This analysis on a global level did, unfortunately, not result in any insightful information. This increases the likelihood that the bias is coming from the data to algorithm and specifically the

measurement and representation bias, as it seems that these errors are the result of rather noisy data. A next step in debugging could be analyzing the mispredictions on local level by using local ML Interpretability methods [Amershi et al., 2015, Pradhan et al., 2021]. This could then be used to search for the cause of each error. According to Amershi et al (2015), these causes could then belong to one of the three common error sources: mislabeled data, feature deficiencies, and insufficient data.

7.5 Validation and knowledge discovery

With the previous ML Interpretability methods, we gained an understanding on the model's working and the modelled relationships among the features. In general, the same features were found important, but none of them predominantly determines the solution. As explained earlier, this could be due to overfitting, the curse of class complexity, and the possible presence of multicollinearity. Next to this, we defined several MDOs which serve as feature selection in order to debug the model. In order to analyze the modelled relationships, we used SHAP beeswarmplots, PDPs and ICE plots. For the majority of the features, the different techniques agreed on the feature's relationship to the target variable. Also, for most of the features, the relationships were the same for all other solutions than `Unrestricted`. This could indicate that many features determine the distinguishment of `Unrestricted` vs the rest. This is actually not strange as the other solutions could be seen as an exceptional solution. Nevertheless, we also observed differences in relationships between the other solutions than `Unrestricted`. For example, the predicted solution probability increases if the machine group is HMI for the solutions `Conversion` and `Restricted`, but decreases for the other solutions. This indicates that this feature on the machine group being HMI is a predictor variable used to distinguish between these minority classes.

For the purpose of model validation and knowledge discovery we used the insights gained from the previously discussed techniques. For each of the most important features per solution, we assessed the PDPs and SHAP beeswarm plots on their shown relationships and importance. Next, we presented and discussed the relationships in an interview with SMEs. During this interview, it became apparent that the comprehensibility of some of the result visualizations was low for the SMEs as they were not familiar with these ML Interpretability techniques and visualizations. Instead, we proposed a summary table where the feature importances and effect towards the solutions were displayed. This summary table was based on the SHAP and PDP plots, where for each solution the 10 most important features were analyzed on their importance and response. We ranked the features per solution on the impact they have on the predicted class probability. We ranked a feature high for a certain solution if there is a relatively high increase in predicted class probability with an increase for this feature. For features that did not have a high impact, they were ranked in the middle. Finally, for a feature solution combination where there was a relatively high decrease in predicted class probability, this feature was ranked low. Subsequently, we used directional coloured arrows to point out the direction and magnitude of the impact. This allowed us to represent the results as a simplified version of the outcomes of the model. This visualization can be found in Appendix E.4. However, this was only possible for the feature solution combinations which had a monotonic relationship, as the impact could be very low for more complex relationships. The SMEs perceived this summary as more comprehensible. Though, since this method simplifies the outcomes of the techniques, the preliminary results of the techniques were analyzed first to identify particularities that were not captured by the summary. For example, the rate of growth on predicted class probability for certain data regions is not displayed. As a consequence, some important particularities could stay undiscovered. Eventually, we used this summary together while mentioning the particularities that were not covered by the summary to validate the results with the SMEs and stakeholders.

The outcomes of this interview can be found in Appendix E.4. Generally, the results were consistent with expectations. The features present among the most important features were as expected more important than the features ranked lower. For instance, the results showed that the feature `Customer` did not have high feature importance, which indicates customers are treated the same, as expected.

Also, there were no surprises regarding the relationships to the predicted probabilities of the solutions, and their (separate) importance on this solution. Though, some features like `Stock` were expected to be more important. As earlier explained in Section 5.3, this could be due to the measurement bias. We used a weekly report on the stock values, which becomes less accurate for EMOs that were created later in the week. Next, this feature described the available `Unrestricted` and `Semi-restricted` stock, which is an aggregation and makes the feature less informative, as well as there are no stock features for the other solutions which could have helped.

7.6 Model debugging

In this section, a summary of the identified MDOs is given, whereafter we tested if the identified MDOs can increase the predictive performance. In essence, we performed a feature selection for all identified MDOs as we selected subsets of relevant features from a larger set of features to use in building a ML model. Research mainly categorized the different feature selection techniques into three categories [Chandrashekar and Sahin, 2014, Miao and Niu, 2016]. The first category is known as the filter techniques which is applied as data preprocessing technique in order to eliminate features on their data characteristics, like variance. Secondly, there are wrapper techniques which are based on the intended ML model itself to evaluate. Thirdly, a combination of the previous two is called embedded feature selection techniques which iteratively evaluate each iteration of the model training process and select those features which contribute the most. Essentially, we used filter and wrapper techniques to obtain our MDOs. Firstly, we reduced the multicollinearity based on hierarchical clustering on correlations, which is a filter technique. Whereafter, we used techniques that are based on the model performance (except from MDI). The filter technique reduced the feature set to some extent, but still resulted in a big feature set. The wrapper method was used for the model agnostic ML Interpretability techniques, as they ranked their feature importances based on the impact on model performance. Finally, by nature, our RFC is an embedded method itself as it selects the best features, according to the chosen split criterion itself. However, this still provides the opportunity to overfit, which we tried to reduce with the identified MDOs. In summary, we have the following MDOs:

1. **MDO-1**: This feature set was obtained by removing the feature interactions by hierarchical clustering on correlations. This feature selection removed 41 correlated features, which resulted in a feature set size of 99.
2. **MDO-2:** This feature set was the result of interpreting the MDI feature importance after the correlated features were removed. Hence, the 14 most important features according to this model-specific method were chosen.
3. **MDO-3:** This feature set contained an additional engineered feature to recapture the lost information by removing the correlated feature `Time_to_UND_SO`. The addition of this feature resulted in a feature set of 100 features.
4. **MDO-4:** This feature set was identified based on the model agnostic technique PIMP. The 13 highest ranked features were chosen for this MDO.
5. **MDO-5:** This feature set contained the 18 highest ranked features determined by the SHAP feature importance.

We used the same method for hyper-parameter tuning for the MDOs as used in the previous chapter. Subsequently, we tested the MDOs on the test set and compared them to the best model from the previous chapter, for which the results can be found in Table 11. Overall, the results did not deviate as much as seen before in the previous chapters. `MDO-2`, `MDO-4`, `MDO-5`, showed a significant lower Precision for all other solutions than `Unrestricted` compared to the other MDOs and full feature set. These MDOs also do not compensate for their loss in Precision on the Recall scores. After inspecting the confusion matrices, nothing could be concluded about the MDOs except for having a slightly higher bias. The bias direction and magnitudes differed per MDO but were not worthwhile

to mention. According to the performance metrics chosen in Chapter 5, the full, `MDO-1`, and `MDO-3` performed best. Following the importance of each metric on the respective solutions, these MDOs outperformed the full feature set. Even though the scores were almost equal, MDO-3 performed slightly better. This suggests that removing the feature interactions and the addition of the engineered feature evidently improved the model's predictive performance. Given these points, the ML Interpretability feature selection techniques that could be classified as wrapper technique did not enhance predictive performance, but decreased it. This could be the result of a too severe feature selection whereafter the models were forced to generalize too much.

	Recall						Precision					
	Unrestricted	Semi-restricted	Conversion	DSP	Restricted	Repair	Unrestricted	Semi-restricted	Conversion	DSP	Restricted	Repair
Full	0.99	0.20	0.58	0.36	0.10	0.12	0.89	0.74	0.86	0.65	0.65	0.80
MDO-1	0.99	0.15	0.51	0.27	0.07	0.12	0.88	0.87	0.87	0.73	0.76	1.00
MDO-2	0.99	0.15	0.52	0.29	0.08	0.15	0.89	0.64	0.79	0.62	0.46	1.00
MDO-3	0.99	0.15	0.52	0.26	0.08	0.12	0.88	0.87	0.88	0.73	0.79	1.00
MDO-4	0.99	0.16	0.52	0.30	0.09	0.15	0.89	0.60	0.79	0.61	0.45	1.00
MDO-5	0.98	0.18	0.51	0.32	0.10	0.15	0.89	0.62	0.79	0.60	0.52	1.00

Table 11: The results of the MDOs on the test set.
Note: colour coding indicates how well the models scored on the performance metric for a given sourcing solution. The utmost colours green and red indicate the model performed relatively well and worse respectively on this metric for a given sourcing solution, but do not reflect on overall performance.

7.7 Evaluation

With the aim of addressing our third sub-research question concerning the use of ML Interpretability methods for model validation, improvement, and knowledge discovery, we demonstrated the application of several post hoc methods for these purposes. There are various methods in the literature for evaluating the effectiveness of the methods employed in this study [Carvalho et al., 2019]. Despite the existence of properties on which ML Interpretability can be evaluated, it is not clear for all of them how to evaluate and measure them [Robnik-Sikonja and Bohanec, 2018, Carvalho et al., 2019]. Nevertheless, we could assess the models on the three goals in ML Interpretability earlier discussed in 2.2.2, namely Accuracy, Understandability, and Efficiency [Rüping et al., 2006]. Still, these goals are partially connected to the properties outlined by Robnik-Sikonja and Bohanec (2018), such as the concept of fidelity, which is analogous to Accuracy, as defined by the aforementioned authors. Accordingly, we decided to evaluate the Interpretability of the utilized methods on the three goals for which the results can be found in Appendix E.5. This showed the advantages and disadvantages of each method, albeit it demonstrated there is no "best" or one-size-fits-all method. They all differ in their objective, and what they wish to display, which makes the one method more comprehensible because of a simple objective, but the other is less comprehensible but more insightful and has a higher Accuracy. Therefore, we suggest that these different methods should be utilized in a complementary manner, depending on the specific insights or relationships that need to be captured. Nevertheless, we argue that SHAP beeswarm plots, PDP and ICE plots convey similar information and therefore do not need to be used simultaneously, unless one wishes to conform its findings.

7.8 Conclusion

In this chapter, we used several ML Interpretability techniques to visualize, analyze, and interpret the modelled relationships by the best predictive model from Chapter 6. We demonstrated that reducing the multicollinearity by hierarchical clustering on correlations gives a more generalized and less biased view of the feature importances and gives a less biased. Overall low feature importances were found, which indicates that the model is overfitted, is prone to class complexity, and/or high multicollinearity is present. Especially, the features were found less important for the minority solutions. In order to overcome these potential causes for the low model performance, we used the feature importances to create MDOs, which served as feature selections. Among the different techniques used for the retrieval

of the feature importances, the top 10 were mainly the same, but the ranking differed. As explained, these differences in ranking could be due to still present multicollinearity. We used the insights for the purpose of model validation and knowledge discovery, with mainly no unexpected relationships. Though, we noted that SMEs which are not familiar with these techniques find methods with higher Accuracy, like PDP or ICE plots, harder to comprehend. A summarized version was made which was perceived as better understandable, though it was supported with information on particularities not covered by this summary. The use of several methods was on the one hand convenient as they could complement each other, but on the other hand, some had a big overlap which makes using all of them valuable. Subsequently, we tried to identify subgroups of mispredictions that could possibly explain the bias present in the model by using data visualization techniques used in modern debug tools. Unfortunately, none of the visualizations revealed any subgroups for mispredictions. This indicates that the bias present is rather coming from the `data to algorithm`. Future work could use local ML Interpretability methods, to identify if these mispredictions come from mislabeled data, feature deficiencies, and/or insufficient data. Finally, we compared the performance of each of the identified MDOs. We concluded that the feature selection based on the feature importance measures resulted in the models generalizing too much, while removing feature interactions and refinement of features can enhance predictive performance. As has been noted, there still exists a high bias towards the majority solution. In the next chapter, the best ML prediction model is used to see if model abstention can enhance decision-making in our problem context, and mitigate this just mentioned bias.

8 Model Abstention

During the last three chapters, we developed several predictive ML models with the aim to support in decision-making on the most effective sourcing solution. The obtained best model still has a reasonable amount of misclassifications, which could lead to poor decisions and a low end-user's trust. With this in mind, we explained in Sections 1.6 and 2.2.3 how model abstentions could mitigate those effects by abstaining from a prediction based on its prediction uncertainty. In this chapter, we aimed to explore how model abstention could be used for this purpose. We first investigated the costs of performing a solution in Section 8.1. Next, we analyzed and developed a rejector that learned to abstain from a prediction with high uncertainty in Section 8.2. Thereafter, we compared the potential savings of the ML model with and without model abstention to the current and optimal workflow in Section 8.3. A final conclusion on this chapter is drawn in Section 8.4.

8.1 Cost estimations

The business motivation of this study was mainly to decrease the number of redundant checks in the manual EMO sourcing process. Throughout this thesis, multiple models were trained and tested, and eventually, an RFC with reduced multicollinearity and additional engineered feature was considered as best performing. To see if the best model can realize savings in terms of time spent on EMOs, the costs of performing the checks for the solutions were estimated in interviews with SMEs and based on experienced (>2 years) planners. For that reason, the obtained absolute savings could in practice be more optimistic as less experienced planners take more time per check. As a matter of fact, we only knew what the most effective solution on a historical EMO had been, and not if other solutions would have been feasible or not. Together with the stakeholders, we assumed for the potential saving calculation, that the most effective solution is the only feasible solution. Consequently, if the solution with the highest predicted class probability is historically seen not the most effective one, the planner would continue with the solution with the second highest predicted class probability, and so on.

An overview of the time spent for each check for each solution is given in Appendix F.1. Accordingly, the time spent on an EMO with the current workflow was calculated by adding each redundant and non-redundant checks of a certain solution. On the contrary, the time spent on an EMO following the ML model was calculated by adding each performed non-redundant check until the most effective solution was found to the activity set and then calculating the time spent. The time spent for the optimal workflow is calculated by only adding the non-redundant checks for the most effective solution. We would like to remark that the prediction model cannot save time on the solution `Unrestricted` because it contains only necessary checks for this solution.

8.2 Model abstention development

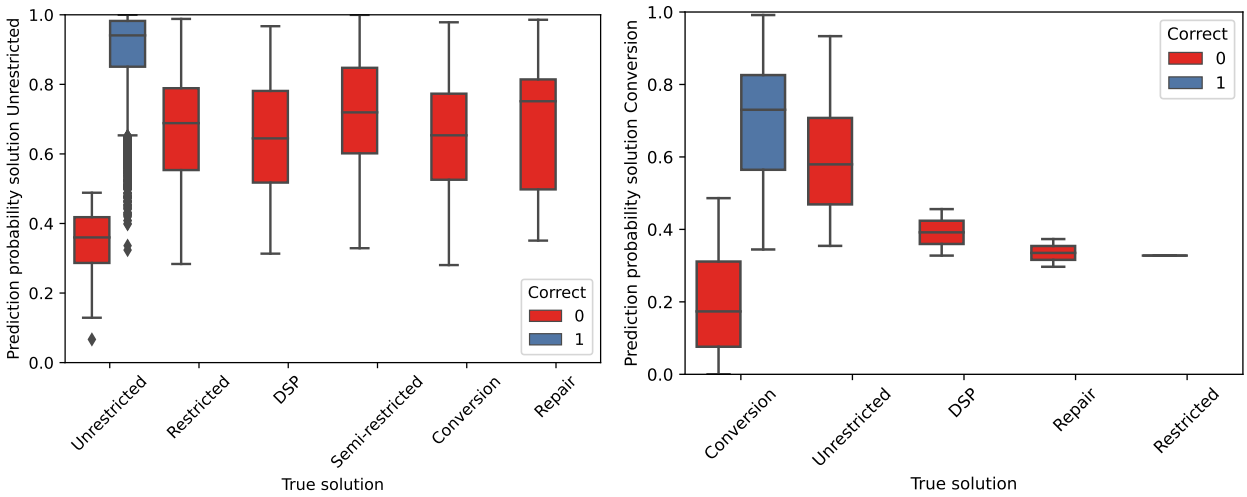
As explained in Section 2.2.3, there are several decisions to be made when developing a model abstention. Specifically, we considered what kind of rejector type would be in line with our objective, how we would learn the rejector, and what learning objectives should be optimized. In this section, we first elaborate on these choices, the analysis of the current uncertainty in the ML prediction model, and the learning of the abstention model when it is beneficial to reject a prediction.

Rejector Design
In Section 2.2.3, we distinguished two types of rejectors: novelty and ambiguity based. We chose the ambiguity type because of three reasons. Firstly, this complements our ML Interpretability study as we would analyze the model's behaviour on a new aspect: prediction uncertainty. Next to this, some ambiguity rejectors do not need another ML model to be developed, whereas novelty rejectors would always need this. Finally, the effect of the bias learned by the model could be mitigated. In terms of learning strategy, we chose to focus on sequential learning. In this way, we would not have to change our ML prediction model and would be allowed to better understand its current behaviour. The used RFC from scikit-learn has the possibility to retrieve the predicted class probability of an

EMO on a solution. These probabilities could reflect on how confident or uncertain the model is about its prediction is, e.g. if the highest predicted probability would be 0.4, the model is less sure about this prediction. On that account, we could set a threshold for which the model should abstain from its rejection. This threshold would thus reflect how confident a model should be about its prediction when we want to give this prediction to the decision-maker. Previous work showed how putting this threshold on the predicted class probability higher than the average could significantly increase the accuracy [Maggi et al., 2014].

Uncertainty

Before we chose the reject threshold(s), we first explored the model's behaviour in terms of uncertainty for mispredictions. In order to do so, we made box plots of the predicted class probabilities for each predicted solution per True solution. Such plots visualize the (un)certainty, which is the predicted probability of the predicted solution in our case, of the model when it (mis)predicts a given solution. From these plots, we could analyze how certain the model was when predicting a given solution when it was correct or incorrect. The results can be found in Appendix F.2, but the plots for `Unrestricted` and `Conversion` are shown in Figure 18 as illustration. We chose these solutions as they represent the general contrast between the majority and minority classes well. As can be seen in Figure 18a, the predicted probability for wrongly predicted EMOs from all other solutions than `Unrestricted` were quite high for this solution, which means the model was fairly sure about its prediction, despite its incorrectness. This was not unexpected because of the earlier observation of the model bias towards this solution. Comparing this with Figure 18b, the model was less sure about its mispredictions on the solution `Conversion`. In particular, we observed that the model was quite sure for the samples which were mispredicted on solution `Unrestricted` ($median = 0.76$), and less sure on the mispredictions on other solutions ($median = 0.59$).



(a) (Un)Certainty of the ML model when (mis)predicting on the solution `Unrestricted` per True solution

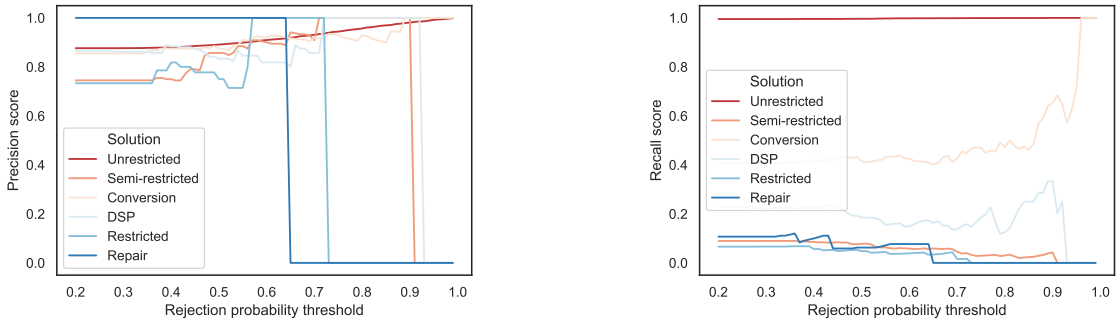
(b) (Un)Certainty of the ML model when (mis)predicting on the solution `Conversion` per True solution

Figure 18: Box plots on predicted class probability per True solution when (mis)predicted on solution `Unrestricted` and `Conversion`.

Threshold Optimization

In order for the decision-making process to benefit from such model abstention, the thresholds of when to abstain should be determined. As explained in Section 2.2.3, this is in research mainly done based on the trade-off between model accuracy and sample coverage. Attention should be paid to how this predictive performance is measured, as the rejection of predictions splits up this performance in non-rejected prediction performance, classification quality and rejection quality [Barandas et al., 2022]. In the end, we were only interested in how the rejector could enhance the earlier obtained ML

model, as it cannot change the model's prediction and we do not compare different ML systems and/or different rejectors. For this reason, we were only interested in the classification of the non-rejected samples, which assesses the classifier's ability in classifying non-rejected samples. Hence, this allows analyzing the performance metrics, Precision and Recall, earlier used on the prediction problem. We visualized the rejection probability thresholds of these performance metrics to analyze their behaviour. The plots can be found in Figure 19. In these visualizations, only the impact on the true solution is shown. As can be seen, there is no optimal threshold for which all performance metrics improved for all solutions. Regarding the Recall score, the solutions `Unrestricted` and `Conversion` are the only solutions with a monotonically increasing relationship with an increase in the threshold. This means that the model is fairly sure about these solutions. Regarding the Precision, all solutions could benefit from rejecting predictions below a threshold of 0.55, as the Precision score is higher than the original score for all solutions.



(a) Precision score and rejection probability threshold per solution

(b) Recall score and rejection probability threshold per solution

Figure 19: Abstainity analysis

We could attempt to find optimal values for each threshold according to the performance metrics, but in the end, the costs saved on the redundant checks was of the highest interest. Therefore, we compared the time spent on an EMO with the current workflow, as the prediction being rejected, and the time spent on an EMO with the workflow determined by the prediction model, as the prediction not being rejected. For this purpose, we first calculated the time spent for each EMO with the prediction model, and the time spent with the current workflow. Subsequently, optimal threshold values could be determined to maximize the realized savings by the system. We chose an evolutionary optimization technique for this purpose because it is an efficient meta-heuristic for solving operations management optimization problems [Katoch et al., 2021] and easy to implement in Python. We first trained our best ML prediction model on the training to validate set, whereafter we predicted on the validation set. The predictions on this validation set were used to determine the best thresholds. The default settings for the genetic algorithm were used with a constraint on the maximum number of iterations without improvement being 10. This resulted in convergence after 50 iterations, as can be seen in Figure 20. The increase in costs saved from the initial population in the genetic algorithm compared to the best solution was rather small. This indicates that the initial population already contained a good candidate solution and/or the net gain with an model abstention cannot be very high in our case.

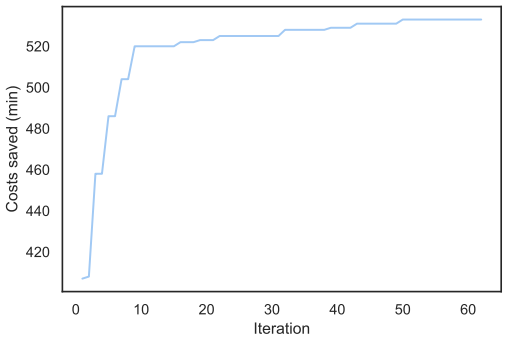


Figure 20: Convergence plot of the genetic algorithm

Solution	Threshold
Unrestricted	0.87
Semi-restricted	0.37
Conversion	0.38
DSP	0.27
Restricted	0.36
Repair	0.33

Table 12: Thresholds per solution

The found thresholds are shown in Table 12. As can be seen, the thresholds for all other solutions than `Unrestricted` were fairly low. This means that for these other solutions, the model assertion recommends to only reject if the model is really unsure. Furthermore, abstaining from predictions with a predicted class probability below these thresholds would save 533 minutes compared to the current workflow. Though, these thresholds are not guaranteed to be optimal as a genetic algorithm can converge to local optima. An exact optimization method could possibly obtain other thresholds resulting in a better performance. In the next section, we used these thresholds on the test set to determine the potential savings of the ML model with and without the abstentions.

8.3 Results and potential savings

In this section, we first applied the prediction model with the abstention model on the test set to analyze the impact on the performance metrics. Thereafter, we analyzed the potential savings of the ML prediction model with and without the abstention model.

Results

We retrained the ML prediction model on the full training set and predicted the most effective sourcing solutions for each EMO in the test set. Thereafter, we could determine which EMOs would have been rejected, and thus would have followed the current workflow. As can be seen in Table 13, as expected, a lot of the True Positive (FP) EMOs of `Unrestricted` were rejected, but at the same time also a lot of the False Positive (FP). For the other solutions, the False Negative (FN) got predominantly rejected, whereas only a few TP and FP were rejected. This indicates that the bias towards the majority solution `Unrestricted` was diminished by the abstention model. In fact, the rejector tackled the epistemic uncertainty, which is the uncertainty caused by incorrect modelled relationships because of the lack of knowledge in the model. Specifically, the results indicate that this was the bias-epistemic uncertainty which was removed.

Solution	Rejected			ΔPrecision	ΔRecall	Coverage
	TP	FP	FN			
Unrestricted	2184	849	3	0.09	-0.01	0.71
Semi-restricted	5	1	395	0.01	0.35	0.28
Conversion	3	2	183	0.01	0.46	0.52
DSP	0	0	76	-	0.37	0.43
Restricted	2	1	169	0.01	0.21	0.26
Repair	0	0	27	-	0.45	0.21

Table 13: Effect abstention model

The impact on the performance metrics for the non rejected EMOs can be found in Table 13 as well. The results show that the Precision slightly increased for some solutions, and, overall, there was a significant increase in Recall. Nevertheless, as earlier explained, it is important to analyze the coverage of the total system as well. High performance on the non-rejected samples can be misleading, as the usability of the system drops with the number of rejected samples. As can be seen in Table 13, the coverage was comparable to the earlier seen Recall of the results without the abstention model, which once again implies that the abstention model diminished the bias towards `Unrestricted`.

As earlier mentioned, both the stated and observed accuracy of a ML system affect the end-users trust [Yin et al., 2019]. With this abstention model, both accuracies improved, which would be beneficial for the confidence an end-user would have in the system. However, the low coverage for some solutions may still lead to a loss of user confidence in the system. This relates back to the birth of abstaining from a prediction [Chow, 1970], where the optimum should be found between rejected proportion and the accuracy of the system.

Potential savings
In Section 8.1, we explained how the potential savings could be calculated. Subsequently, we created a model assertion that learned when it is beneficial to abstain from a prediction based on the model's uncertainty. In this section, we compared the costs of four theoretically possible workflows that we identified during this research.

1. **Current workflow:** This workflow was the workflow that is currently used in the manual EMO sourcing process, which is explained at the beginning of this research in Chapter 1.
2. **ML prediction model workflow:** This workflow was based on the best obtained ML prediction model. The solutions are sorted in descending order on their predicted class probability, where the planner starts with the solution with the highest probability and continues to the next if it turns out to be not feasible.
3. **ML prediction model + abstention model workflow:** This workflow was based on the previous workflow, but if the highest predicted class probability was below the optimized threshold, the current workflow was followed.
4. **Optimal workflow:** This workflow represented the workflow in an ideal world where the solution would be known upfront. Hence, only the costs of the most effective solution without the redundant checks were taken into account.

As earlier this chapter explained, there are no redundant checks for the solution `Unrestricted` in the current workflow. Therefore, we did not include the costs for the EMOs with this solution being the most effective in this analysis. Though, we included the costs for EMOs that were mispredicted on this solution. In Table 14 the total costs and savings of each workflow on the test set are given. As can be seen, both ML with and without the abstention model realized cost savings on the test set. What is more, the rejector saves 56% more minutes compared to the model without these abstentions. Nevertheless, the ML model with and without the abstentions did not come close to the optimal workflow. This was as expected for the workflow of ML without the rejector, as we earlier observed the moderate predictive performance. For the workflow with the reject option, we were able to increase the savings to some extent, but not close to optimal. Nevertheless, the effectiveness of the abstention model is highly dependent on the relative costs savings of these predictions [Metzger and Föcker, 2017]. A small deviation in the used costs per solution could already manifest the change in saved costs.

Workflow	Costs (min)	Savings (min)
Current	11,759	N/A
ML	11,128	631
ML + abstention model	10,773	986
Optimal	6,170	5,589

Table 14: Costs and savings realized on the test set

8.4 Conclusion

In this chapter question, we aimed to explore how the best prediction ML model could be enhanced with an abstention model. Firstly, we estimated the costs associated with performing checks on the solutions. Subsequently, we designed the abstention model based on the properties of the used RFC, and our desideratum for ML Interpretability. We chose an ambiguity rejector which was sequentially learned when to abstain based on the highest predicted class probability. We optimized the rejection thresholds to abstain based on the saved costs with a genetic algorithm. These thresholds were used on the test set, whereafter we could compare the costs associated with the current workflow, the ML model with and without model assertion, and the optimal workflow. The results showed that the abstention model enhanced the ML prediction model in mainly two ways. Firstly, the abstention model could defer from the ML prediction model to the current workflow such that there was no time lost compared to this current workflow, which increased the saved costs by 56%. Next to this, the abstention model increased the ulterior predictive performance of the ML prediction model. This increase would be in both observed and stated predictive performance which increments the end-users' trust. Utilizing this abstention model would require planners to more often use the current workflow, but if the prediction is not rejected, it is significantly more often correct. Not to mention, the results showed that the bias-epistemic uncertainty resulting in the bias towards the `Unrestricted` can be reduced significantly with the use of this abstention model. On the other hand, the coverage for the minority solutions dropped significantly, albeit these rejected predictions would mainly have resulted in excessive costs. Still, we stress that the total net benefit of such an abstention model is highly dependent on the estimated costs and the effectiveness of the ML prediction model.

9 Conclusion

In this final chapter, we present the conclusions of this thesis. The conclusions on the main and sub-research questions are discussed in Section 9.1. Then, in Section 9.2, we elaborate on the scientific and business contribution, and recommendations for ASML. Finally, in Section 9.3, we outline the limitations and suggest future research for this thesis.

9.1 Conclusions

In Section 1.8, we established the main research question for this thesis, which was divided into four sub-research questions. In this section, we first address these sub-research questions and then provide an answer to the main research question.

> *Sub-research question (1): How can Causal ML be used to predict the most effective sourcing solution?*

Conventional Causal ML models typically aim to estimate causal effects, but we justified transforming this into a causal classification problem using policy learning and optimization. However, this transformation encountered difficulties specific to Causal ML. Specifically in our case, this method required synthetic control samples and raised a new class imbalance regarding the outcome variable. Tests of various solutions for these problems showed predominantly poor predictive performance. Besides potential bias sources that can come along with every predictive modelling task, some algorithm and Causal Inference specific requirements and constraints could be the reason for this poor performance. We suppose: 1) the unconfoundedness condition is critical for this method, 2) causal effect estimators should be modelled and evaluated separately and in control of the developer before the policy optimizer is learned, 3) the objective function of this policy optimizer should be tailored to the research problem. This poor performance also degraded the appropriateness of using this Causal ML for knowledge and causal discovery which ought to be one of the advantages of this ML method.

> *Sub-research question (2): How do Causal ML models compare with traditional ML prediction techniques when applied to predicting the most effective sourcing solution?*

In this sub-research question, we compared the performance of Causal ML to that of traditional ML models (RFC and AutoML), where the former was outperformed by the latter on the defined performance metrics. We attribute this to the less accurate features identified as important by Causal ML, and the ability of traditional ML to rely on associations in cases of unconfoundedness. The poor performance together with the disengagement of using Causal ML for (causal) knowledge extraction, results in our suggestion that Causal ML without instrumental variables should only be used if the conditions on Causal Inference are strictly met. The decent performance of the traditional ML methods also allowed us to use ML Interpretability practices for knowledge discovery.

> *Sub-research question (3): How can ML Interpretability methods be used for model validation and improvement, and knowledge discovery?*

To investigate the use of ML Interpretability in enabling valuable and deployable ML models for operational decision-making in supply chain management, we aimed to explore its potential in this domain. Next to this, such methods ought to be useful for model debugging to improve predictive performance. We demonstrated how global ML Interpretability methods can be used to analyze and interpret the modelled relationships such that they could be used for model validation and knowledge discovery. In particular, SHAP, PDP, and ICE plots were found useful as they allow for a more granular view and insightfulness. Accordingly, these methods uncovered that most of the features are used to distinguish between the majority and minority solutions, and only a few among the minority solutions. The use of global agnostic methods for model debugging resulted in a minor increase in performance.

Sub-research question (4): How can the best ML prediction model be enhanced with model abstention?

With the belief that model abstentions can enhance the ML predictions model in terms of predictive performance, we explored in this sub-research question how such abstention could be used in our case. The results demonstrated a significant reduction in bias-epistemic uncertainty. Accordingly, the use of an abstention model resulted in improved performance on the chosen metrics and a reduction in time spent on the workflow with the base ML model, with a particularly notable improvement in Recall. This increase in performance could enhance the planners' trust in the system. Altogether, we argue that this is a convenient way of dealing with a ML model's deficiencies.

Main research question: How can causal and interpretable predictive analytics be used to support ASML's global operations center planners in choosing the most effective sourcing solution within the emergency maintenance order sourcing process?

Even though in theory Causal ML would be the ultimate solution for predictive and prescriptive analytics, the adaption and development of such a method should be done with care. The satisfaction of the conditions could in practice be the first obstacle. For example, the unconfoundedness assumption, which states that all confounding variables are observed, is in practice difficult to satisfy. For example, we were limited to the data available on the stock and lead time per solution which in practice are confounders. Moreover, practitioners should consider other aspects of the data, such as representation and measurement accuracy. Accordingly, some variables that were expected to be important (e.g., stock) were found to have low feature importance and predictive power due to deficiencies in measurement, leading to bias and epistemic uncertainty in the model. Next to this, the transformation of the business classification problem to a causal classification problem needed some data preparation on the outcome sparsity which in our case was explored without a solid theoretical foundation. Additionally, the choice or design of algorithms should be carefully considered. Literature argued that the accuracy of causal effect estimators is not crucial for causal decision-making. However, using pre-made functions limits one's control, which in our case resulted in a lack of knowledge if our causal effect estimators were accurate *enough*. Next to this, the objective of a policy optimizer should be specific to the problem at hand. In our case, we prioritized the overall accuracy of the model over our defined performance metrics, which may have resulted in a bias toward the majority solution.

Comparing Causal ML to traditional ML, the latter was performing significantly better on the set performance metrics. As explained, this could be due to the predictive power of some non-causal features such as criticality. The inclusion of AutoML in our study showed how it can compete with manually developed ML models. With the limitations in mind, like scarce performance metrics, this method is very convenient for developing models for the purpose of proof of concepts. Furthermore, the predictive performance of traditional ML was more appropriate for knowledge discovery.

Regarding ML Interpretability, we observed that the utilized methods produced similar insights, and their complementary use facilitated the summary and interpretation of the results before presenting and verifying them with SMEs and stakeholders. Though, the more sophisticated methods were too complex for non-ML practitioners to understand and a summarized version was needed for this understanding. By way of contrast, the used global methods did not reveal any biases that could be tackled for the purpose of model debugging. In fact, the main takeaway of the results from the MDOs was that one should analyze the multicollinearity to improve your model's generalizability and proper model interpretation.

Lastly, concerning abstention models, we believe that such ML model assertions can definitely enhance ML models. When the goal of predictive analytics is decision support rather than automation, the sample coverage is not one of the main goals that should be considered. Though, regarding the obtained EMO coverage for some solutions, one could argue that the total system does not meet the usability requirements. However, the results showed that there were hardly any rejections that would have resulted in cost savings.

9.2 Scientific contribution and recommendations

The introduction of this thesis outlined the scientific and business significance of the study. In this section, we detail the contributions of the research to both scientific and business fields and present specific recommendations for ASML.

9.2.1 Scientific contribution

In general, we demonstrated how Big Data analytics can create value in supply chain management processes, which contributes to the lacunae in empirical studies in this field [Frank et al., 2019]. In the realm of operational decision-making in after-sales service logistics, and especially for orders that have already started, we have demonstrated how using predictive analytics can outperform the current solutions based on (simple) business rules and expert knowledge reported by Topan et al. (2020). Specifically on predictive analytics and its obstacles mentioned by Pearl (2019), we showed that utilizing Causal ML methods is not a trivial task in practice. Some deterrents for this matter are the complexity of the problem, data availability and quality. Moreover, the transformation from conventional Causal ML to a classification problem is hardly studied. We came across several obstacles and limitations of the currently available methods for this purpose, such as the objective function misalignment with the problem objective. Next to this, research argues that there is still a long road ahead for an effective design of AutoML solutions, where exploratory studies are needed for this purpose to reveal the design and technical challenges and flaws [Elshawi et al., 2019, Karmaker et al., 2021]. In this thesis, we demonstrated how this method can be used, and its limitations regarding the to be optimized performance metrics and dealing with (high) class imbalance. Earlier, we explained the lack of standard routines and processes on ML Interpretability [Molnar, 2020], and the lacunae in knowledge on the utilization of such methods in operational decision-making in our supply chain domain [Baryannis et al., 2019]. In this thesis, we demonstrated how global ML Interpretability methods can be used complementarily for model validation and knowledge extraction. Besides we contributed to the lacunae of ML abstention where we demonstrated how uncertainty thresholds could be optimized with a genetic algorithm in a multi-class setting with dynamic costs, in a low-stake, high-frequency decision-making problem.

Lessons learned:
In this thesis, we used an exploratory case study to contribute to the current lacunae in research and business in the field of causal and interpretable predictive analytics. On the journey towards our end results, there were some important lessons learned:

1. First of all, formulating, aggregating and labelling targets was not a trivial task. It required a deeper understanding of the problem, but despite the valuable contributions of the SMEs, estimations on the occurrence of each solution were rather hard to make. As a result, there existed a high class imbalance and low sample size which could be avoided by different aggregation levels and still maintaining the business value.

2. Regarding the chosen performance metrics, the combination of Precision and Recall with each a different, at the time, not quantifiable importance per solution, made it challenging to optimize hyper-parameters and compare different models. A better approach would be to quantify them upfront, allowing for the creation of a scoring solution that can be optimized and interpreted more easily.

3. As mentioned before, Causal ML should be utilized with care. As of yet, research still contains a big lacuna in this discipline. The assumptions and conditions to be made and satisfied, made this approach less convenient than expected. Besides, there is still a lack of frameworks, workflows, pipelines, and best practices for the use of Causal ML, which made the study and deployment of these algorithms in a classification study harder than expected.

9.2.2 Business relevance and recommendations

From a business perspective, this thesis contributed mainly to two aspects. Firstly, we showed how predictive analytics solutions can be used to support ASML's GOC planners in choosing the most effective sourcing solution within the emergency maintenance order sourcing process. We demonstrated that this solution could remove redundant checks from the workflow and increase the performance on the set SLAs to some moderate extent, which responds to our earlier defined problem statement. We did this by using Causal and traditional ML methods, for which the latter showed to remove a fair number of redundant checks. Besides, we used ML Interpretability methods to extract knowledge on important determinants in this process. Looking forward, we demonstrated how abstention models could be used to reject wrong predictions and enhance the planners' trust. The second aspect regards the set dream state, where we explored and demonstrated how ASML's CSCM department could become more data-driven, and accomplish this dream state.

Next to the contributions, we would like to recommend some actions on the prolongation of this study.

1. The used data model and modelling skeleton could also be used for other problems. To start with, the scope of the problem could be adjusted. For instance, the SOs with other priority labels, which form about 60% of the SOs sourced by the GOC, could easily be included as well. Next to this, different target labels like globally or regionally sourced could easily be used to generate more insights. Together, this could increase the total benefit of the developed solution.
2. We also recommend using some of the insights gained with the ML Interpretability methods to present and include in the onboarding of new planners. For instance, the knowledge of the observed increase in the probability of the solution `Conversion` to be the most effective solution on an EMO for the machine group HMI, could help planners to get faster to the most effective solution.
3. Despite the presence of many unrepresentative `Unrestricted` solutions in our data, the majority of EMOs are sourced from locations where the automated fulfillment algorithm should be used. We recommend studying the root causes of the EMO not being sourced by these algorithms, in an effort to reduce these EMOs being forwarded to the GOC.
4. As noted in Section 8.3, the observed potential savings are not close to optimal or substantial. In order to increase these savings and make the exact solution worth deploying, the `data to algorithm` bias should be tackled on the causes discussed in Section 5.3.

9.3 Limitations and future research

Firstly, the likelihood of unobserved confounders is one risk to the internal validity of our findings. To assess the validity, a sensitivity analysis to identify the strength of the impacts of a confounder required for our model to fundamentally change could be used. Presuming that is the case, extra features, like stock for each solution, should be included in the data. Besides, the conclusions drawn on Causal ML could be a second threat to internal validity as we only tested one method. Yet, causal feature selection methods could be used to exclude non-causal features present in the current feature set [Yu et al., 2021]. On the other hand, future research should examine the use of instrumental variables, which mimic the behaviour of omitted confounding variables, in policy learning to overcome the problem of unconfoundedness.

A second threat to the internal validity is the result of the chosen granularity of our solutions, which resulted in small sample sizes for some solutions. Combined with the high class complexity, the ML models found it difficult to distinguish between the solutions. Furthermore, due to these small sample sizes, the results could in fact deviate from differently chosen random states of the model. Similarly, the abstention model was trained on a relatively small validation set that was prone to instability, as previously noted. Despite that the model's behaviour was not that different compared

to the test set, future research could study the effect of using more robust methods like CV for the purpose of rejector learning. Next to this, the used data did not represent the real problem in scope which make the results less reliable for the model performance in practice. With this in mind, better filtering should be applied to diminish this limitation.

In this thesis, we primarily focused on the development of a ML prediction model. However, it is important to also consider the implementation of this data analytics tool, which is often poorly done by manufacturing companies [Frank et al., 2019]. Some aspects to investigate before implementing are on the concept of model deployment like concept drift, and cross-cutting aspects like end users' trust [Paleyes et al., 2022]. With the continuous changes in ASML's supply chain, the learned models are prone to concept drift, which is a phenomenon where the data underlying the model has changed significantly resulting in degradation of model performance [Tsymbal, 2004]. Regarding the latter, local ML Interpretability methods can be used complementary to global methods to enhance this trust by explaining individual predictions [Carvalho et al., 2019]. It is worth noting that if such a decision support tool were to be implemented and retrained after use to address issues such as concept drift, the data could have become dirty due to the inclusion of solutions that were wrongly classified as the most effective but used by the planner. This would result in contaminating the data with incorrect information.

On the note of theoretical generalizability, known as transferability, we argue that this is two-sided. On the one hand, the internal (quantitative) findings, like the specific insights on feature importances are hardly generalizable, as they are too problem specific. However, on a more qualitative note, the used methodologies and their results could definitely be used for other components in the order fulfillment process of ASML, as also explained in the second recommendation. In a broader context, the methodology could be used for studies on predictive and prescriptive process monitoring without event logs, where the focus is rather on contextual features and less on the process. Yet, the adjustment of the Causal ML application in our problem makes it less transferable to problems where no outcome sparsity exists. The approach for model abstention that was used can be readily applied to other domains and problems where the ML model provides inherent uncertainty measures. The main drivers for developing an abstention model in these cases are cost savings, improved predictive performance, and increased end-user trust.

A limitation of the used ML Interpretability methodology is the fact that the discovered relationships cannot directly be interpreted as causal. Instead, Zhao and Hastie (2021) propose that ML Interpretability methods like PDP can be used for causal interpretations if they are complemented by a structural causal model and based on a ML prediction model with a good predictive performance. Hence, this could be done in further research by ASML if they wish to interpret the modelled relationships as causal. Another limitation and at the same time future research is on the use of ML Interpretability methods to obtain the influence of two features together on the model. This would allow us to discover more complicated relationships and uncover potential bias sources.

Due to the time constraint on this thesis, we focused on global ML Interpretability methods. We presumed that the bias is mainly coming from the `data to algorithm`, which actually can be debugged with local ML Interpretability methods [Krishnan and Wu, 2017, Zhang et al., 2019]. As for abstention models, future research could study how novelty rejectors compare against our designed ambiguity rejector. The last threat to internal validity is the high dependency of the costs savings on the cost estimations of the checks.

As a final note, it is worth mentioning that the focus of this exploratory case study was relatively broad, encompassing multiple topics. Although this allowed us to explore several fields and gain knowledge on know-hows, it also resulted in less research depth on these topics. For example, we only used one Causal ML model, which decreases the validity of the statements on Causal ML.

References

[Ali et al., 2013] Ali, A., Shamsuddin, S. M., and Ralescu, A. L. (2013). Classification with class imbalance problem. *Int. J. Advance Soft Compu. Appl*, 5(3).

[Allison, 2001] Allison, P. D. (2001). *Missing data*. Sage publications.

[Amershi et al., 2015] Amershi, S., Chickering, M., Drucker, S. M., Lee, B., Simard, P., and Suh, J. (2015). Modeltracker: Redesigning performance analysis tools for machine learning. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 337–346.

[Arlot and Celisse, 2010] Arlot, S. and Celisse, A. (2010). A survey of cross-validation procedures for model selection. *Statistics surveys*, 4:40–79.

[Athey and Imbens, 2017] Athey, S. and Imbens, G. W. (2017). The state of applied econometrics: Causality and policy evaluation. *Journal of Economic perspectives*, 31(2):3–32.

[Athey and Wager, 2021] Athey, S. and Wager, S. (2021). Policy learning with observational data. *Econometrica*, 89(1):133–161.

[Athey et al., 2017] Athey, S., Wager, S., et al. (2017). Efficient policy learning. Technical report.

[Azpiroz et al., 2021] Azpiroz, I., Oses, N., Quartulli, M., Olaizola, I. G., Guidotti, D., and Marchi, S. (2021). Comparison of climate reanalysis and remote-sensing data for predicting olive phenology through machine-learning methods. *Remote Sensing*, 13(6).

[Azure, 2022] Azure, M. (2022). Prevent overfitting and imbalanced data with automated machine learning. `https://learn.microsoft.com/en-us/azure/machine-learning/concept-automated-ml`. Accessed: 2022-11-30.

[Barandas et al., 2022] Barandas, M., Folgado, D., Santos, R., Simão, R., and Gamboa, H. (2022). Uncertainty-based rejection in machine learning: Implications for model development and interpretability. *Electronics*, 11(3):396.

[Baryannis et al., 2019] Baryannis, G., Dani, S., and Antoniou, G. (2019). Predicting supply chain risks using machine learning: The trade-off between performance and interpretability. *Future Generation Computer Systems*, 101:993–1004.

[Belkin et al., 2019] Belkin, M., Hsu, D., Ma, S., and Mandal, S. (2019). Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854.

[Bozorgi et al., 2020] Bozorgi, Z. D., Teinemaa, I., Dumas, M., La Rosa, M., and Polyvyanyy, A. (2020). Process mining meets causal machine learning: Discovering causal rules from event logs. In *2020 2nd International Conference on Process Mining (ICPM)*, pages 129–136. IEEE.

[Bozorgi et al., 2021] Bozorgi, Z. D., Teinemaa, I., Dumas, M., La Rosa, M., and Polyvyanyy, A. (2021). Prescriptive process monitoring for cost-aware cycle time reduction. In *2021 3rd International Conference on Process Mining (ICPM)*, pages 96–103. IEEE.

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

[Brinkrolf and Hammer, 2018] Brinkrolf, J. and Hammer, B. (2018). Interpretable machine learning with reject option. *at-Automatisierungstechnik*, 66(4):283–290.

[Carvalho et al., 2019] Carvalho, D. V., Pereira, E. M., and Cardoso, J. S. (2019). Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8).

[Caton and Haas, 2020] Caton, S. and Haas, C. (2020). Fairness in machine learning: A survey. *arXiv preprint arXiv:2010.04053*.

[Chandrashekar and Sahin, 2014] Chandrashekar, G. and Sahin, F. (2014). A survey on feature selection methods. *Computers and Electrical Engineering*, 40(1):16–28. 40th-year commemorative issue.

[Chawla et al., 2002] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

[Choi et al., 2018] Choi, T.-M., Wallace, S. W., and Wang, Y. (2018). Big data analytics in operations management. *Production and Operations Management*, 27(10):1868–1883.

[Chow, 1970] Chow, C. (1970). On optimum recognition error and reject tradeoff. *IEEE Transactions on information theory*, 16(1):41–46.

[Condessa et al., 2017] Condessa, F., Bioucas-Dias, J., and Kovačević, J. (2017). Performance measures for classification systems with rejection. *Pattern Recognition*, 63:437–450.

[Devriendt et al., 2018] Devriendt, F., Moldovan, D., and Verbeke, W. (2018). A literature survey and experimental evaluation of the state-of-the-art in uplift modeling: A stepping stone toward the development of prescriptive analytics. *Big Data*, 6:13–41.

[Dormann et al., 2013] Dormann, C. F., Elith, J., Bacher, S., Buchmann, C., Carl, G., Carré, G., Marquéz, J. R. G., Gruber, B., Lafourcade, B., Leitão, P. J., et al. (2013). Collinearity: a review of methods to deal with it and a simulation study evaluating their performance. *Ecography*, 36(1):27–46.

[Doshi-Velez and Kim, 2017] Doshi-Velez, F. and Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

[Du et al., 2019] Du, M., Liu, N., and Hu, X. (2019). Techniques for interpretable machine learning. *Communications of the ACM*, 63(1):68–77.

[Dudík et al., 2015] Dudík, M., Erhan, D., Langford, J., and Li, L. (2015). Doubly robust policy evaluation and optimization. *ArXiv*, abs/1503.02834.

[Eichengreen and Gupta, 2013] Eichengreen, B. and Gupta, P. (2013). The two waves of service-sector growth. *Oxford Economic Papers*, 65(1):96–123.

[Elder, 2018] Elder, J. (2018). Chapter 16 - the apparent paradox of complexity in ensemble modeling*. In Nisbet, R., Miner, G., and Yale, K., editors, *Handbook of Statistical Analysis and Data Mining Applications (Second Edition)*, pages 705–718. Academic Press, Boston, second edition edition.

[Elgendy and Elragal, 2016] Elgendy, N. and Elragal, A. (2016). Big data analytics in support of the decision making process. *Procedia Computer Science*, 100:1071–1084. International Conference on ENTERprise Information Systems/International Conference on Project MANagement/International Conference on Health and Social Care Information Systems and Technologies, CENTERIS/ProjMAN / HCist 2016.

[Elshawi et al., 2019] Elshawi, R., Maher, M., and Sakr, S. (2019). Automated machine learning: State-of-the-art and open challenges. *arXiv preprint arXiv:1906.02287*.

[Fernández-Delgado et al., 2014] Fernández-Delgado, M., Cernadas, E., Barro, S., and Amorim, D. G. (2014). Do we need hundreds of classifiers to solve real world classification problems. *Journal of Machine Learning Research*, 15:3133–3181.

[Fernández-Loría and Provost, 2022] Fernández-Loría, C. and Provost, F. (2022). Causal decision making and causal effect estimation are not the same... and why it matters. *INFORMS Journal on Data Science*.

[Frank et al., 2019] Frank, A. G., Dalenogare, L. S., and Ayala, N. F. (2019). Industry 4.0 technologies: Implementation patterns in manufacturing companies. *International Journal of Production Economics*, 210:15–26.

[Frawley et al., 1992] Frawley, W. J., Piatetsky-Shapiro, G., and Matheus, C. J. (1992). Knowledge discovery in databases: An overview. *AI magazine*, 13(3):57–57.

[Friedman, 2001] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

[Funk et al., 2011] Funk, M. J., Westreich, D., Wiesen, C., Stürmer, T., Brookhart, M. A., and Davidian, M. (2011). Doubly robust estimation of causal effects. *American journal of epidemiology*, 173(7):761–767.

[Gartner, 2021] Gartner, I. (2021). Effective decision making must be connected, contextual and continuous.

[Goldstein et al., 2015] Goldstein, A., Kapelner, A., Bleich, J., and Pitkin, E. (2015). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *journal of Computational and Graphical Statistics*, 24(1):44–65.

[Guo et al., 2020] Guo, R., Cheng, L., Li, J., Hahn, P. R., and Liu, H. (2020). A survey of learning causality with data: Problems and methods. *ACM Comput. Surv.*, 53(4).

[Gutierrez and Gérardy, 2017] Gutierrez, P. and Gérardy, J.-Y. (2017). Causal inference and uplift modelling: A review of the literature. In *International conference on predictive applications and APIs*, pages 1–13. PMLR.

[Haixiang et al., 2017] Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., and Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, 73:220–239.

[Hastie et al., 2009] Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

[He and Garcia, 2009] He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284.

[Hendrickx et al., 2021] Hendrickx, K., Perini, L., Van der Plas, D., Meert, W., and Davis, J. (2021). Machine learning with a reject option: A survey. *arXiv preprint arXiv:2107.11277*.

[Holland, 1986] Holland, P. W. (1986). Statistics and causal inference. *Journal of the American Statistical Association*, 81(396):945–960.

[Hünermund et al., 2022] Hünermund, P., Kaminski, J., and Schmitt, C. (2022). Causal machine learning and business decision making. *Available at SSRN 3867326*.

[Hutter et al., 2019] Hutter, F., Kotthoff, L., and Vanschoren, J. (2019). *Automated machine learning: methods, systems, challenges*. Springer Nature.

[Japkowicz and Stephen, 2002] Japkowicz, N. and Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent data analysis*, 6(5):429–449.

[Jordan and Mitchell, 2015] Jordan, M. I. and Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260.

[Kang et al., 2018] Kang, D., Raghavan, D., Bailis, P., and Zaharia, M. (2018). Model assertions for debugging machine learning. In *NeurIPS MLSys Workshop*, volume 3, page 10.

[Kang et al., 2020] Kang, D., Raghavan, D., Bailis, P., and Zaharia, M. (2020). Model assertions for monitoring and improving ml models. In Dhillon, I., Papailiopoulos, D., and Sze, V., editors, *Proceedings of Machine Learning and Systems*, volume 2, pages 481–496.

[Karmaker et al., 2021] Karmaker, S. K., Hassan, M. M., Smith, M. J., Xu, L., Zhai, C., and Veeramachaneni, K. (2021). Automl to date and beyond: Challenges and opportunities. *ACM Computing Surveys (CSUR)*, 54(8):1–36.

[Katoch et al., 2021] Katoch, S., Chauhan, S. S., and Kumar, V. (2021). A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, 80(5):8091–8126.

[Kompa et al., 2021] Kompa, B., Snoek, J., and Beam, A. L. (2021). Second opinion needed: communicating uncertainty in medical machine learning. *NPJ Digital Medicine*, 4(1):1–6.

[Krishnan and Wu, 2017] Krishnan, S. and Wu, E. (2017). Palm: Machine learning explanations for iterative debugging. In *Proceedings of the 2nd Workshop on Human-In-the-Loop Data Analytics*, HILDA'17, New York, NY, USA. Association for Computing Machinery.

[Leevy et al., 2018] Leevy, J. L., Khoshgoftaar, T. M., Bauder, R. A., and Seliya, N. (2018). A survey on addressing high-class imbalance in big data. *Journal of Big Data*, 5(1):1–30.

[Lin et al., 2017] Lin, W.-C., Tsai, C.-F., Hu, Y.-H., and Jhang, J.-S. (2017). Clustering-based undersampling in class-imbalanced data. *Information Sciences*, 409:17–26.

[Lundberg and Lee, 2017] Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *NIPS*.

[Maggi et al., 2014] Maggi, F. M., Di Francescomarino, C., Dumas, M., and Ghidini, C. (2014). Predictive monitoring of business processes. In Jarke, M., Mylopoulos, J., Quix, C., Rolland, C., Manolopoulos, Y., Mouratidis, H., and Horkoff, J., editors, *Advanced Information Systems Engineering*, pages 457–472, Cham. Springer International Publishing.

[Mehrabi et al., 2021] Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., and Galstyan, A. (2021). A survey on bias and fairness in machine learning. *ACM Comput. Surv.*, 54(6).

[Metzger and Föcker, 2017] Metzger, A. and Föcker, F. (2017). Predictive business process monitoring considering reliability estimates. In *International Conference on Advanced Information Systems Engineering*, pages 445–460. Springer.

[Miao and Niu, 2016] Miao, J. and Niu, L. (2016). A survey on feature selection. *Procedia Computer Science*, 91:919–926. Promoting Business Analytics and Quantitative Management of Technology: 4th International Conference on Information Technology and Quantitative Management (ITQM 2016).

[Molnar, 2020] Molnar, C. (2020). *Interpretable machine learning*. Lulu. com.

[Molnar et al., 2020] Molnar, C., König, G., Bischl, B., and Casalicchio, G. (2020). Model-agnostic feature importance and effects with dependent features–a conditional subgroup approach. *arXiv preprint arXiv:2006.04628*.

[Nadeem et al., 2009] Nadeem, M. S. A., Zucker, J.-D., and Hanczar, B. (2009). Accuracy-rejection curves (arcs) for comparing classification methods with a reject option. In *Machine Learning in Systems Biology*, pages 65–81. PMLR.

[Oesterreich and Teuteberg, 2016] Oesterreich, T. D. and Teuteberg, F. (2016). Understanding the implications of digitisation and automation in the context of industry 4.0: A triangulation approach and elements of a research agenda for the construction industry. *Computers in industry*, 83:121–139.

[Paleyes et al., 2022] Paleyes, A., Urma, R.-G., and Lawrence, N. D. (2022). Challenges in deploying machine learning: A survey of case studies. *ACM Comput. Surv.* Just Accepted.

[Pearl, 2009] Pearl, J. (2009). *Causality*. Cambridge university press.

[Pearl, 2019] Pearl, J. (2019). The seven tools of causal inference, with reflections on machine learning. *Communications of the ACM*, 62(3):54–60.

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

[Pereira and Romero, 2017] Pereira, A. C. and Romero, F. (2017). A review of the meanings and the implications of the industry 4.0 concept. *Procedia Manufacturing*, 13:1206–1214.

[Pourret et al., 2008] Pourret, O., Na, P., Marcot, B., et al. (2008). *Bayesian networks: a practical guide to applications*. John Wiley & Sons.

[Pradhan et al., 2021] Pradhan, R., Zhu, J., Glavic, B., and Salimi, B. (2021). Interpretable data-based explanations for fairness debugging. *arXiv preprint arXiv:2112.09745*.

[Probst et al., 2019] Probst, P., Wright, M. N., and Boulesteix, A.-L. (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery*, 9(3):e1301.

[Radcliffe and Surry, 2012] Radcliffe, N. J. and Surry, P. D. (2012). Real-world uplift modelling with significance-based uplift trees.

[Rajbahadur et al., 2022] Rajbahadur, G. K., Wang, S., Oliva, G. A., Kamei, Y., and Hassan, A. E. (2022). The impact of feature importance methods on the interpretation of defect classifiers. *IEEE Transactions on Software Engineering*, 48(7):2245–2261.

[Raschka, 2018] Raschka, S. (2018). Model evaluation, model selection, and algorithm selection in machine learning. *arXiv preprint arXiv:1811.12808*.

[Rendell, 1986] Rendell, L. (1986). A general framework for induction and a study of selective induction. *Machine learning*, 1(2):177–226.

[Ribeiro et al., 2016] Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "why should i trust you?": Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[Robnik-Sikonja and Bohanec, 2018] Robnik-Sikonja, M. and Bohanec, M. (2018). Perturbation-based explanations of prediction models. In *Human and Machine Learning*.

[Rubin, 2005] Rubin, D. B. (2005). Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331.

[Rüping et al., 2006] Rüping, S. et al. (2006). Learning interpretable models.

[Sanders, 2016] Sanders, N. R. (2016). How to use big data to drive your supply chain. *California Management Review*, 58:26 – 48.

[Schölkopf, 2022] Schölkopf, B. (2022). Causality for machine learning. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, pages 765–804.

[Schröer et al., 2021] Schröer, C., Kruse, F., and Gómez, J. M. (2021). A systematic literature review on applying crisp-dm process model. *Procedia Computer Science*, 181:526–534.

[Shoush and Dumas, 2022a] Shoush, M. and Dumas, M. (2022a). Prescriptive process monitoring under resource constraints: A causal inference approach. In Munoz-Gama, J. and Lu, X., editors, *Process Mining Workshops*, pages 180–193, Cham. Springer International Publishing.

[Shoush and Dumas, 2022b] Shoush, M. and Dumas, M. (2022b). When to intervene? prescriptive process monitoring under uncertainty and resource constraints. *Lecture Notes in Business Information Processing*, 458 LNBIP:207 – 223. Cited by: 0; All Open Access, Green Open Access, Hybrid Gold Open Access.

[Strobl et al., 2006] Strobl, C., Boulesteix, A., Zeileis, A., and Hothorn, T. (2006). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics*, 8:25 – 25.

[Syrgkanis et al., 2021] Syrgkanis, V., Lewis, G., Oprescu, M., Hei, M., Battocchi, K., Dillon, E., Pan, J., Wu, Y., Lo, P., Chen, H., Harinen, T., and Lee, J.-Y. (2021). Causal inference and machine learning in practice with econml and causalml: Industrial use cases at microsoft, tripadvisor, uber. In *2021 Knowledge Discovery and Data Mining*.

[Tipping and Bishop, 1999] Tipping, M. E. and Bishop, C. M. (1999). Mixtures of probabilistic principal component analyzers. *Neural computation*, 11(2):443–482.

[Topan et al., 2020] Topan, E., Eruguz, A., Ma, W., van der Heijden, M., and Dekker, R. (2020). A review of operational spare parts service logistics in service control towers. *European Journal of Operational Research*, 282(2):401–414.

[Tsymbal, 2004] Tsymbal, A. (2004). The problem of concept drift: definitions and related work. *Computer Science Department, Trinity College Dublin*, 106(2):58.

[van der Aalst et al., 2003] van der Aalst, W. M., ter Hofstede, A. H. M., and Weske, M. (2003). Business process management: A survey. In *Business Process Management*.

[Vandewiele et al., 2021] Vandewiele, G., Dehaene, I., Kovács, G., Sterckx, L., Janssens, O., Ongenae, F., De Backere, F., De Turck, F., Roelens, K., Decruyenaere, J., Van Hoecke, S., and Demeester, T. (2021). Overly optimistic prediction results on imbalanced data: a case study of flaws and benefits when applying over-sampling. *Artificial Intelligence in Medicine*, 111:101987.

[Waller and Fawcett, 2013] Waller, M. A. and Fawcett, S. E. (2013). Data science, predictive analytics, and big data: a revolution that will transform supply chain design and management.

[Wamba and Mishra, 2017] Wamba, S. F. and Mishra, D. (2017). Big data integration with business processes: a literature review. *Business Process Management Journal*.

[Wang and Yao, 2012] Wang, S. and Yao, X. (2012). Multiclass imbalance problems: Analysis and potential solutions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(4):1119–1130.

[Wei et al., 2015] Wei, P., Lu, Z., and Song, J. (2015). Variable importance analysis: A comprehensive review. *Reliability Engineering and System Safety*, 142:399–432.

[Wirth and Hipp, 2000] Wirth, R. and Hipp, J. (2000). Crisp-dm: Towards a standard process model for data mining. In *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*, volume 1, pages 29–40. Manchester.

[Yin et al., 2019] Yin, M., Wortman Vaughan, J., and Wallach, H. (2019). Understanding the effect of accuracy on trust in machine learning models. In *Proceedings of the 2019 chi conference on human factors in computing systems*, pages 1–12.

[Yu et al., 2021] Yu, K., Liu, L., and Li, J. (2021). A unified view of causal and non-causal feature selection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(4):1–46.

[Zhang et al., 2019] Zhang, J., Wang, Y., Molino, P., Li, L., and Ebert, D. S. (2019). Manifold: A model-agnostic framework for interpretation and diagnosis of machine learning models. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):364–373.

[Zhao and Hastie, 2021] Zhao, Q. and Hastie, T. (2021). Causal interpretations of black-box models. *Journal of Business & Economic Statistics*, 39(1):272–281.

[Zheng et al., 2006] Zheng, A. X., Jordan, M. I., Liblit, B., Naik, M., and Aiken, A. (2006). Statistical debugging: Simultaneous identification of multiple bugs. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 1105–1112, New York, NY, USA. Association for Computing Machinery.

[Zhou et al., 2015] Zhou, K., Liu, T., and Zhou, L. (2015). Industry 4.0: Towards future industrial opportunities and challenges. In *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pages 2147–2152.

[Zöller and Huber, 2021] Zöller, M.-A. and Huber, M. F. (2021). Benchmark and survey of automated machine learning frameworks. *Journal of artificial intelligence research*, 70:409–472.

Appendices

A Literature Review

Method	Global vs Local	Agnostic vs Specific	Result
Local Surrogate Model (LIME)	Local	Model Agnostic	Feature summary
BreakDown	Local	Model Agnostic	Feature summary
Scoped Rules (Anchors)	Local	Model Agnostic	Feature summary
Counterfactual Explanations	Local	Model Agnostic	New Data point
Partial Dependence Plot (PDP)	Global	Model Agnostic	Feature summary
Accumulated Local Effects (ALE) Plot	Global	Model Agnostic	Feature summary
Feature Interaction	Global	Model Agnostic	Feature summary
Functional Decomposition	Global	Model Agnostic	Feature summary
Permutation Feature Importance	Global	Model Agnostic	Feature summary
Global Surrogate	Global	Model Agnostic	Feature summary
Influence Functions	Both	Model Agnostic	Existent Data point
Prototypes and Criticisms	Both	Model Agnostic	Existent Data point
Individual Condition Expectation (ICE)	Both	Model Agnostic	Feature summary
Feature Importance	Both	Model Specific	Feature summary
Shapley Values (SHAP)	Both	Model Agnostic	Feature summary

Table 15: ML interpretability techniques

B Data analysis preliminaries

Variable	Nobs	Mean	Min	Max	Median	Mode	Variance	Skewness	Kurtosis
Count_NORA_indicator_changes	11,573	2.12	1.00	39.00	2.00	1.00	4.23	7.01	94.63
Upgraded	40,298	0.15	0.00	1.00	0.00	0.00	0.13	1.94	1.76
Time_to_UND_SO	40,298	-446.10	-9144.93	1,152.55	-99.96	-751.23	688,618.98	-3.50	16.52
GOC_Entry_time_to_und	40,298	-307.05	-11,397.76	7,047.08	-73.57	-247.42	406,517.60	-4.63	34.34
SCP	40,277	9,269.21	0.00	5,290,023.61	1,243.26	0.11	4,402,814,124	39.32	2,229.48
Target	40,298	80.89	0.00	1,354.00	38.00	0.00	13,435.82	3.10	13.87
Stock	40,099	41.95	-1.00	941.00	17.00	1.00	5,329.21	5.10	40.91
Incoming_supply	25,972	11.60	-0.15	506.00	2.00	0.00	1,008.77	7.91	84.29
Fillrate	40,298	1.79	0.00	162.00	0.85	0.00	11.41	6.90	153.91
Total_forecast	40,298	19.77	0.00	1,254.36	4.64	0.00	1,767.51	7.00	110.56
NAV_points	40,287	5.22	0.00	30.00	3.00	3.00	63.70	2.11	3.48
Fillrate_points	40,298	2.11	0.00	12.00	1.00	0.00	8.97	1.66	2.19
Total_NB_orders	30,393	21.37	0.00	1,193.00	2.00	0.00	4,488.47	7.43	73.76
Total_criticality_points	23,601	15.06	0.00	58.00	9.00	3.00	250.82	1.32	0.71

Table 16: Descriptive statistics of numerical variables

Variable name	Description	Example	Type
Activity_type	Maintenance activity type	'Z86/Billable'	Categorical
Business_line	Business line of machine	'HMI'	Categorical
Confirmation_method	Confirtmation method of service order	'Semi Automated'	Categorical
Count_NORA_indicator_changes	Number of changes in NORA indicator	3	Numerical
Criticality	Criticality this week	'L3'	Categorical
Criticality_lw	Criticality last week	'L3'	Categorical
Cross_plant_material_status	Cross plant material status	'R2'	Categorical
Customer_name	Name of customer	'Taiwan Semiconductor Manufacturing'	Categorical
Day_period_GOC_Entry	Day period the EMO entered the GOC scope	'Early morning'	Categorical
Fillrate	Fillrate	0.45	Numerical
Fillrate_points	Fillrate points from criticality	3	Numerical
Gesa_scope	If the order is in the GESA scope	'In Scope'	Categorical
GOC_Entry_weekday	Weekday the EMO entered the GOC scope	'Monday	Categorical
Goodwill_reason	Goodwill reason for EMO creation	'PRIORITY LEAD-TIME'	Categorical
Highest_maintenance_activity_type	Highest maintenance activity type	'Z86/Billable'	Categorical
Hold_in_local_warehouse	If material is in stock in local warehouse	'No'	Boolean
Incoming_supply	Amount of incoming supply	12	Numerical
Machine Down?	If machine is down or not	'No'	Boolean
Machine_type_text	Machine type	'NXE:3400C-S3PLUSMV'	Categorical
Manual_vs_automated	If service order is confirmed manually or automatically	'Manual'	Categorical
Material_type	Type of material	'SERP'	Categorical
NAV_points	Not available points from criticality	1	Numerical
New_introduced	If the material was newly introduced	1	Numerical
NORA_indicator	If in NORA scope or not	'Y'	Categorical
Order_supplychain_flow	Type of supply chain flow	'After-sales'	Categorical
Planning_level	Planning level of service order	LPA'	Categorical
Profit_center	Profit center of machine	'EUV'	Categorical
Requesting_Region	Requesting region	'EMEA'	Categorical
SCP_2	Standard cost price	245.12	Numerical
Service_order_type	Service order type	'ZS01'	Categorical
Solution	Sourcing solution	'Conversion'	Categorical
Special_parts_planner	If it is a special planned pat	'GOC special planned'	Categorical
Stock	Unrestricted stock for material	3	Numerical
Target	WW stock target of material	6	Numerical
Time_to_UND_SO	Time to UND at SO creation in hours	45	Numerical
Total_criticality_points	Total number of criticality points	22	Numerical
Total_forecast	Total WW forecast of material	12	Numerical
Total_NB_orders	Total outstanding new buy orders	34	Numerical
Upgraded	If the SO is upgraded with higher priority	1	Numerical

Table 17: Features in final data set

C Causal Machine Learning

C.1 Sample distribution

	Total Samples	Samples per solution							Samples per outcome				
		0	**1**	**2**	**3**	**5**	**4**	**6**	**-3**	**-2**	**-1**	**0**	**1**
M1-O	53250	26625	22658	1693	1041	665	455	113	-	-	-	26625	26625
M1-U	21268	10634	6667	1693	1041	665	455	113	-	-	-	10634	10634
M1-C	80004	40002	6667	6667	6667	6667	6667	6667	-	-	-	40002	40002
M2	159750	-	26625	26625	26625	26625	26625	26625	-	-	-	133125	26625
M3	159750	-	26625	26625	26625	26625	26625	26625	52472	51754	28899	-	26625

Table 18: Samples per solution and outcome variable

C.2 Confusion matrices Causal ML

		True label					
		Unrestricted	**Semi restricted**	**Conversion**	**DSP**	**Restricted**	**Repair**
Predicted Label	**Unrestricted**	7055	377	146	63	157	19
	Semi restricted	167	127	6	2	18	0
	Conversion	161	1	208	0	3	0
	DSP	44	4	7	48	2	2
	Restricted	92	46	9	6	49	0
	Repair	20	1	8	14	1	13

Table 19: Confusion matrix M1-O

		True label					
		Unrestricted	**Semi restricted**	**Conversion**	**DSP**	**Restricted**	**Repair**
Predicted Label	**Unrestricted**	4821	126	55	38	57	9
	Semi restricted	1753	373	19	7	95	1
	Conversion	536	3	261	1	7	1
	DSP	173	9	23	76	5	14
	Restricted	233	44	22	8	65	0
	Repair	23	1	4	3	1	9

Table 20: Confusion matrix M1-U

		True label					
		Unrestricted	**Semi restricted**	**Conversion**	**DSP**	**Restricted**	**Repair**
Predicted Label	**Unrestricted**	4509	115	46	37	49	11
	Semi restricted	1795	374	20	12	89	1
	Conversion	504	16	276	2	6	3
	DSP	151	5	11	65	3	5
	Restricted	470	43	22	4	80	0
	Repair	110	3	9	13	3	14

Table 21: Confusion matrix M1-C

		True label					
		Unrestricted	**Semi restricted**	**Conversion**	**DSP**	**Restricted**	**Repair**
Predicted Label	**Unrestricted**	7527	541	281	119	222	29
	Semi restricted	0	15	0	0	0	0
	Conversion	8	0	103	0	2	0
	DSP	3	0	0	14	0	3
	Restricted	1	0	0	0	6	0
	Repair	0	0	0	0	0	2

Table 22: Confusion matrix M2

		True label					
		Unrestricted	**Semi restricted**	**Conversion**	**DSP**	**Restricted**	**Repair**
Predicted Label	**Unrestricted**	7533	547	338	128	230	34
	Semi restricted	0	9	0	0	0	0
	Conversion	3	0	46	0	0	0
	DSP	3	0	0	5	0	0
	Restricted	0	0	0	0	0	0
	Repair	0	0	0	0	0	0

Table 23: Confusion matrix M3

D Traditional Machine Learning

D.1 Hyper-parameters RQ2

Metric
Accuracy
AUC_weighted
Average_precision_score_weighted
Norm_macro_recall
Precision_score_weighted

Table 24: Performance metrics Azure AutoML

Azure AutoML algorithms
Logistic Regression
Light GBM
Gradient Boosting
Decision Tree
K Nearest Neighbors
Linear SVC
Support Vector Classification (SVC)
Random Forest
Extremely Randomized Trees
Xgboost
Naïve Bayes
Stochastic Gradient Search

Table 25: Azure AutoML algorithms

D.2 Results Traditional ML

		True label					
		Unrestricted	**Semi restricted**	**Conversion**	**DSP**	**Restricted**	**Repair**
Predicted Label	**Unrestricted**	7458	440	158	82	193	25
	Semi restricted	23	111	0	2	12	1
	Conversion	32	2	222	0	2	0
	DSP	16	1	4	48	1	4
	Restricted	9	2	0	1	22	0
	Repair	1	0	0	0	0	4

Table 26: Confusion matrix RFC-O

		True label					
		Unrestricted	**Semi restricted**	**Conversion**	**DSP**	**Restricted**	**Repair**
Predicted Label	**Unrestricted**	7220	335	111	60	141	19
	Semi restricted	170	205	8	5	40	0
	Conversion	87	4	257	3	5	1
	DSP	34	3	6	61	3	8
	Restricted	28	9	1	4	41	0
	Repair	0	0	1	0	0	6

Table 27: Confusion matrix RFC-U

		True label					
		Unrestricted	**Semi restricted**	**Conversion**	**DSP**	**Restricted**	**Repair**
Predicted Label	**Unrestricted**	7052	303	99	54	137	19
	Semi restricted	289	235	12	9	44	0
	Conversion	115	3	262	3	5	1
	DSP	42	4	6	63	3	8
	Restricted	41	11	5	4	41	0
	Repair	0	0	0	0	0	6

Table 28: Confusion matrix RFC-C

		True label					
		Unrestricted	**Semi restricted**	**Conversion**	**DSP**	**Restricted**	**Repair**
Predicted Label	**Unrestricted**	7080	380	149	62	153	19
	Semi restricted	154	115	3	2	20	0
	Conversion	154	1	206	0	3	0
	DSP	39	3	6	47	2	2
	Restricted	91	56	13	7	51	0
	Repair	21	1	7	15	1	13

Table 29: Confusion matrix DRRFPL

		True label					
		Unrestricted	**Semi restricted**	**Conversion**	**DSP**	**Restricted**	**Repair**
Predicted Label	**Unrestricted**	7411	405	137	77	177	21
	Semi restricted	54	143	2	3	15	2
	Conversion	43	2	240	1	5	1
	DSP	19	1	4	49	2	5
	Restricted	12	5	1	3	31	0
	Repair	0	0	0	0	0	5

Table 30: Confusion matrix AutoML

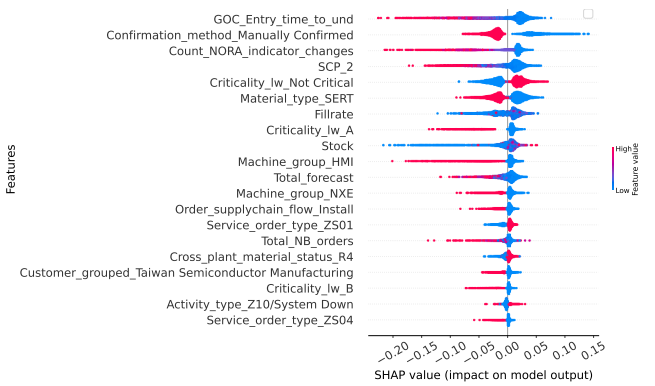
E Machine Learning Interpretability

E.1 Feature importance summary

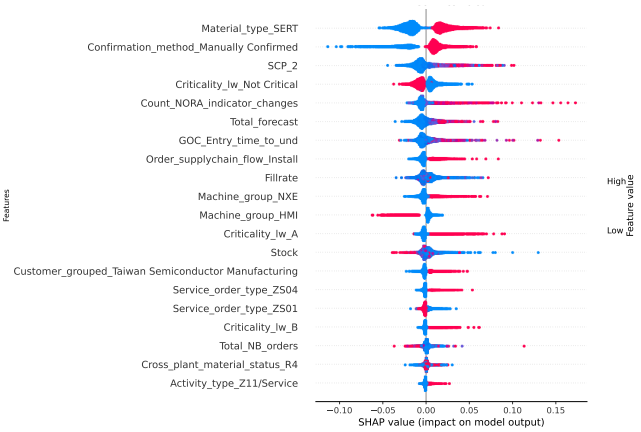
Feature	**MDI**	**PIMP**	**SHAP**
Confirmation_method_Manually Confirmed	2	2	8
Count_NORA_indicator_changes	4	5	6
Criticality_lw_A			1
Criticality_lw_Not Critical			4
Fillrate	6	7	3
GOC_Entry_time_to_und	10	10	10
Machine_group_HMI	3	3	7
Material_type_SERT	1	1	9
SCP_2	9	9	5
Stock	8	8	2
Total Forecast	5	6	
Total NB orders	7	4	

Table 31: Summary feature importances. Note: "10" is the highest feature importance of that method, "1" the lowest, and empty not present, colors are used only for visual interpretation ease.

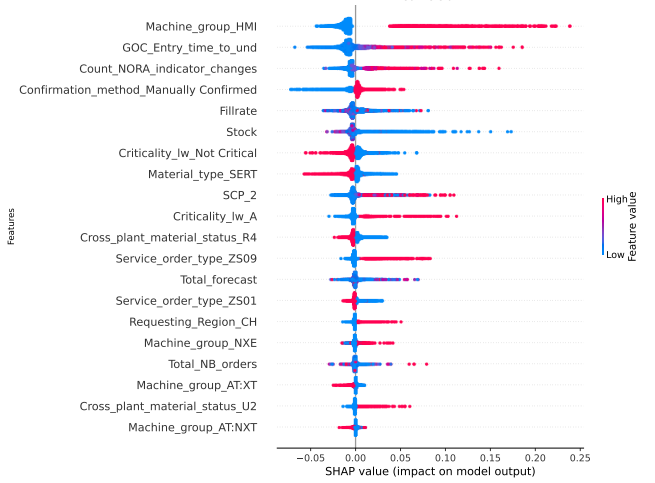
E.2 ML interpretability plots



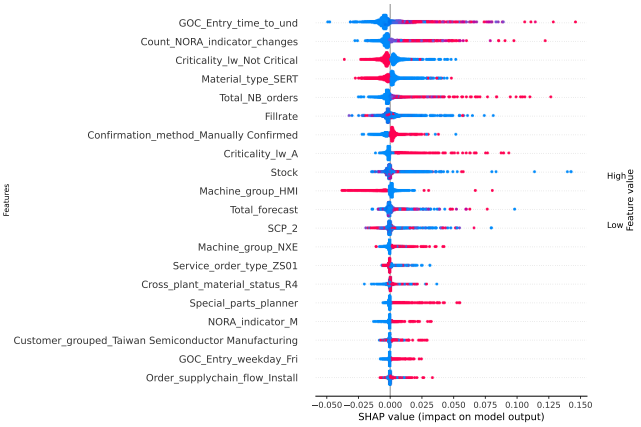
(a) Unrestricted



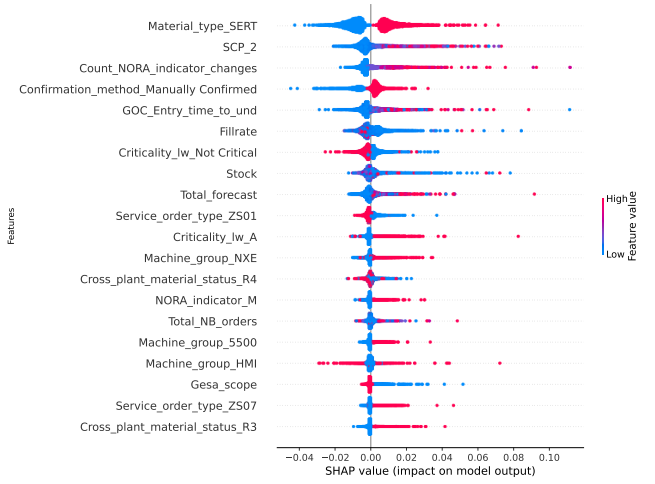
(b) Semi-restricted



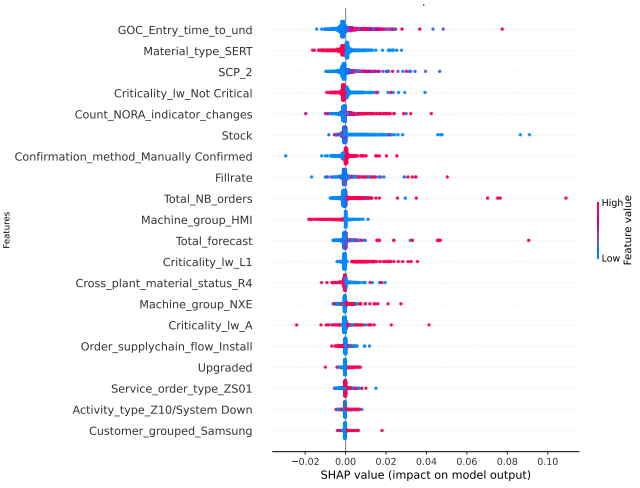
(c) Conversion



(d) DSP

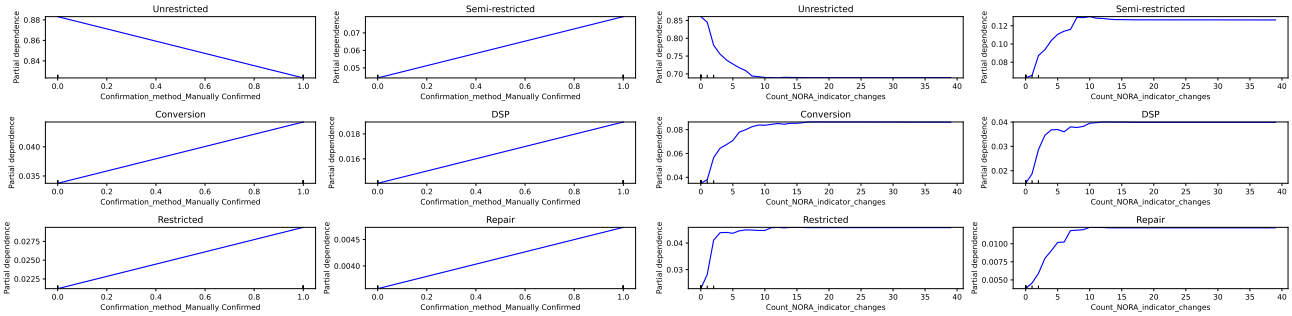


(e) Restricted



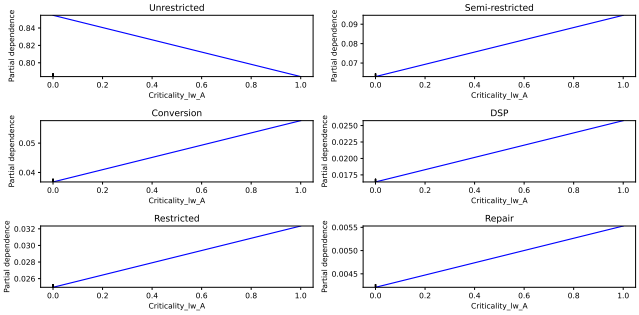
(f) Repair

Figure 21: SHAP beeswarm summary plots

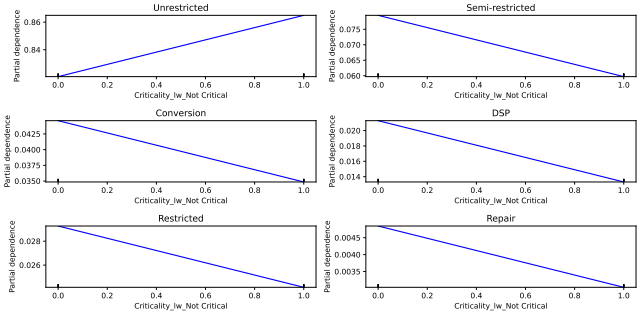


(a) Confirmation_method_Manually Confirmed

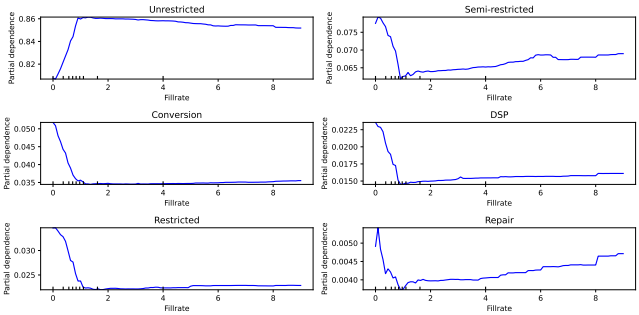
(b) Count_NORA_indicator_changes



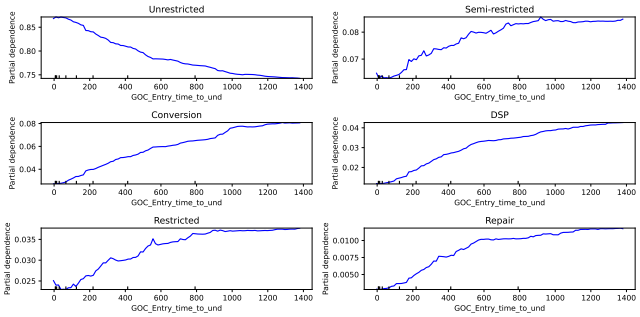
(c) Criticality_lw_A



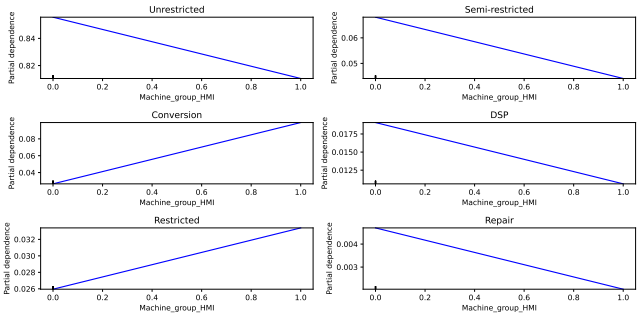
(d) Criticality_lw_Not Critical



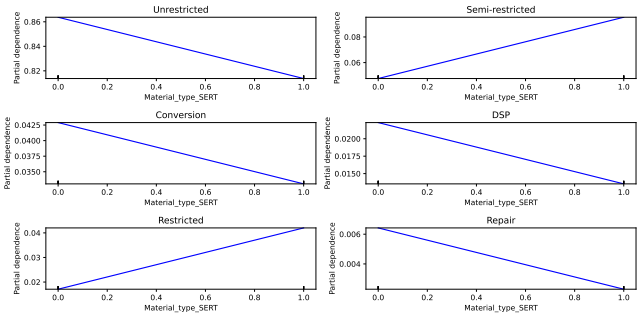
(e) Fillrate



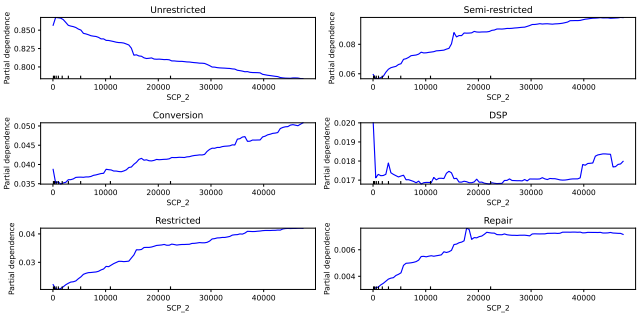
(f) GOC_Entry_time_to_und



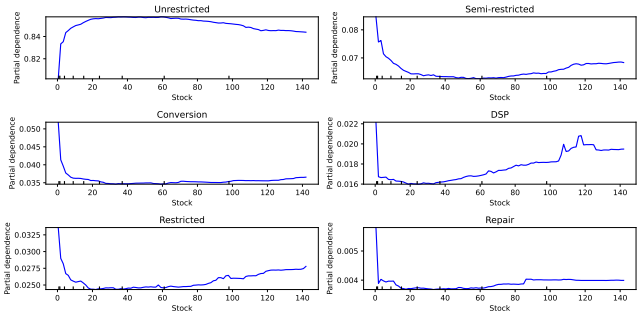
(g) Machine_group_HMI



(h) Material_type_SERT

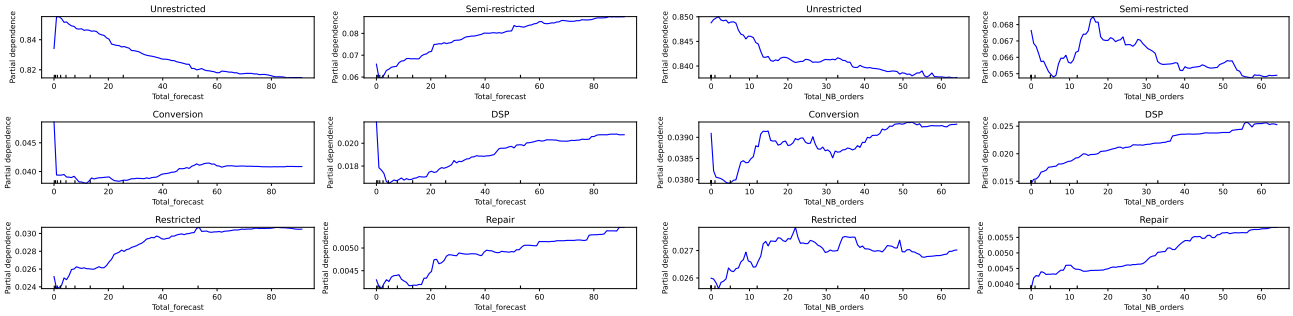


(i) SCP_2



(j) Stock

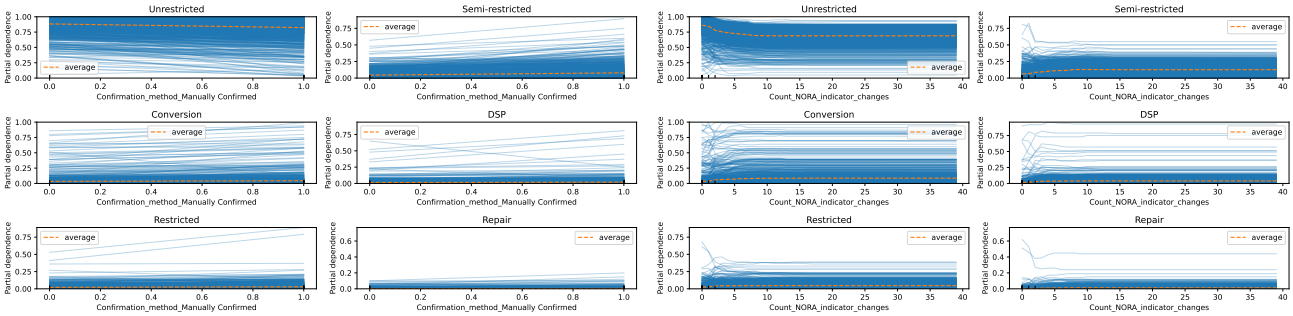
Figure 22: PDP plots: part 1



(a) Total_forecast

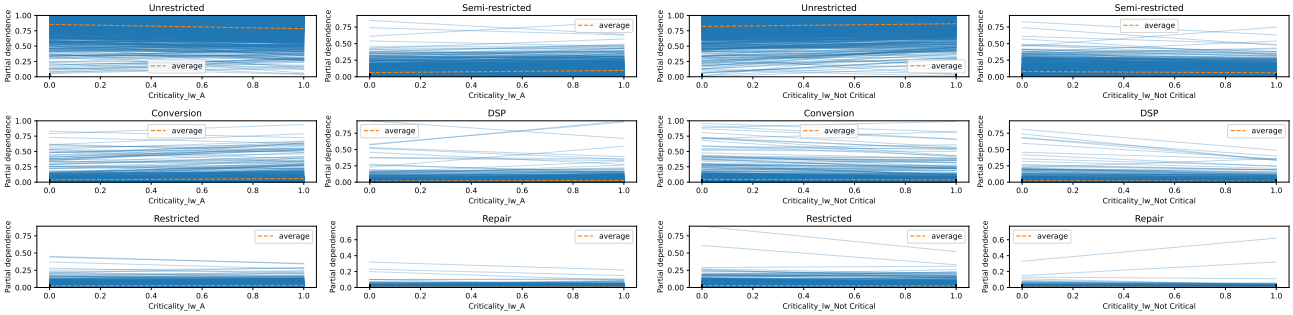
(b) Total_NB_orders

Figure 23: PDP plots: part 2



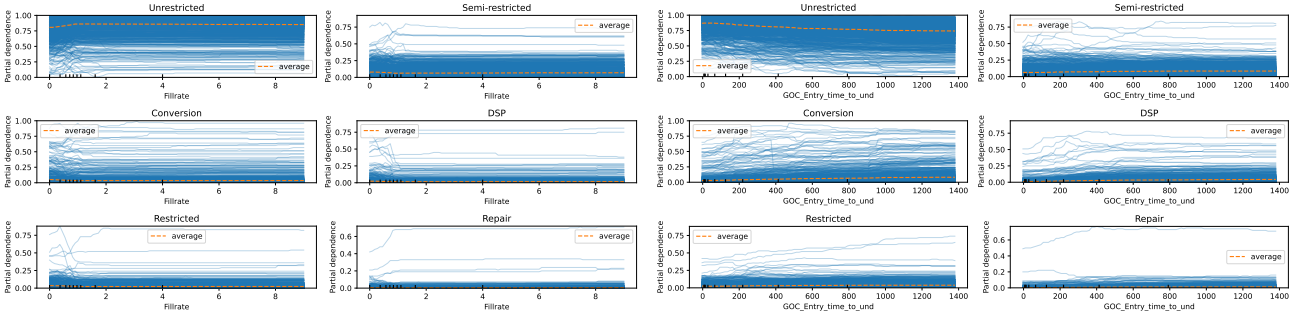
(a) Confirmation_method_Manually Confirmed

(b) Repair



(c) Criticality_lw_A

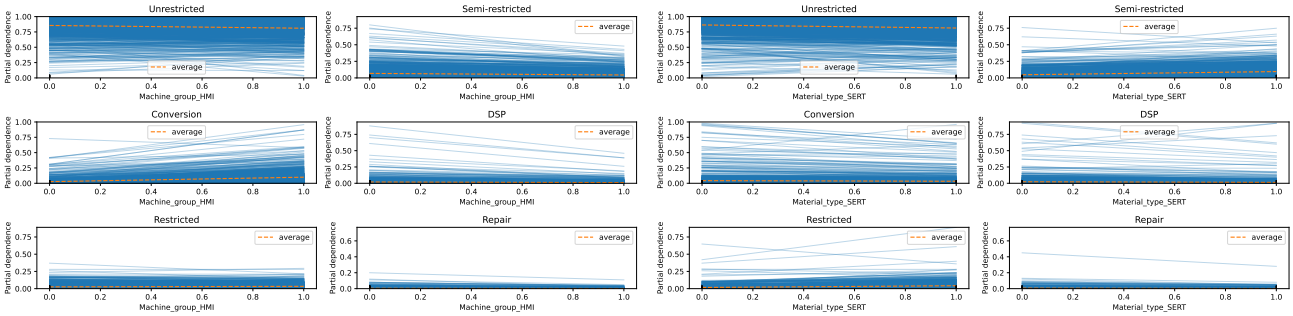
(d) Criticality_lw_Not Critical



(e) Fillrate

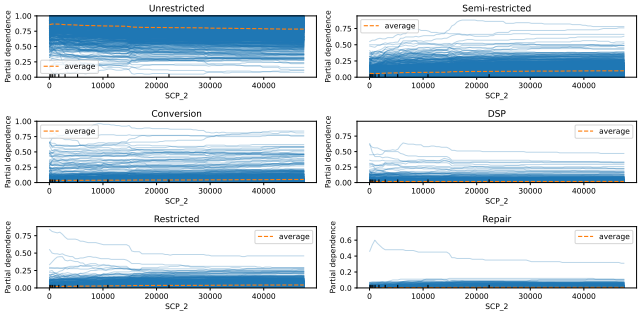
(f) GOC_Entry_time_to_und

Figure 24: ICE plots: part 1

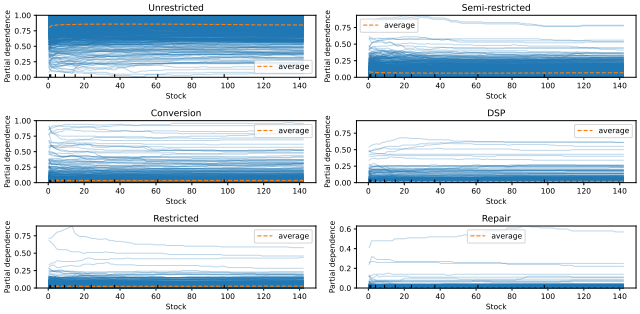


(a) Machine_group_HMI

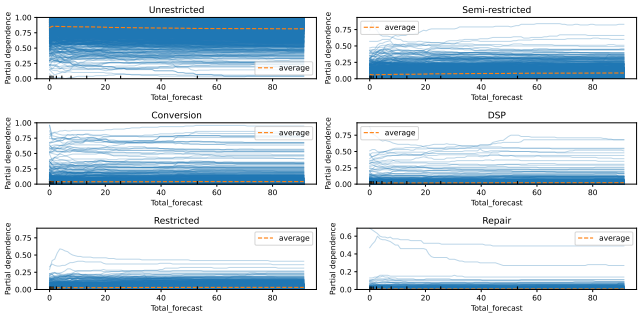
(b) Material_type_SERT



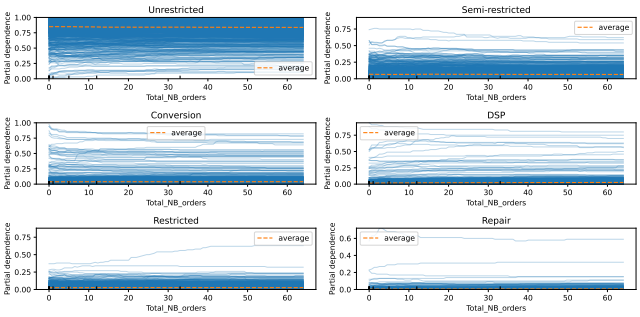
(c) SCP_2



(d) Stock



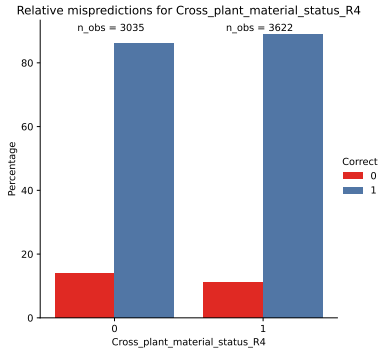
(e) Total_forecast



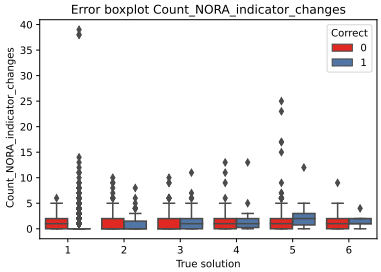
(f) Total_NB_orders

Figure 25: ICE plots: part 2

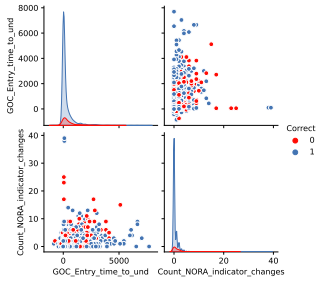
E.3 Error analysis



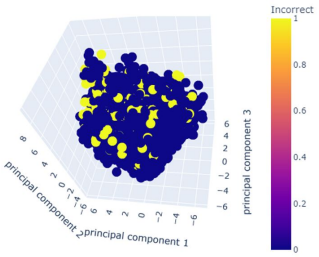
(a) Error box plot Cross_plant_material_status_R4



(b) Error box plot Count_NORA_indicator_changes



(c) Error pair plot



(d) Error PCA 3d plot

Figure 26: Error analysis examples

E.4 Model validation and knowledge discovery interviews

Feature	Min-max value	Unrestricted	Semi-restricted	Conversion	DSP	Restricted	Repair
GOC_ENTRY_time to_und	0-1000	↓	→	↑	↑	↗	↑
Machine_group_HMI	False-True	→	↓	↑	↓	→	↓
Count_NORA_indicator_changes	0-10	↓	↑	↗	↑	↑	↑
SCP_2	0-30000	→	→	↘	↓	↑	→
Fillrate	0-1.5	↑	↘	↓	↓	↓	↓
Confirmation_method_manually	False-True	→	→	↘	↘	→	↘
Stock	0-100	↑	↓	↓	↓	↓	↓
Criticality lw not critical	False-True	↑	↓	↓	↓	↘	↓
Criticality lw = A	False-True	→	↗	→	→	→	↘
Total NB orders	0-50	↗			→	↘	↘
Material_type_SERT	False-True	→	↑	↓	↓	↑	↓
Total_forecast	0-60	→	→			→	↘

Figure 27: Feature importance summary designed with and for SMEs. An arrow up, sideways, and down means relatively high positive impact, no/small impact, high negative impact on the predicted probability of the solution

Feature	Review
GOC_Entry_ time_to_UND	The difference between the PDPs showed that all the probability of an EMO being sourced from a solution other than Unrestricted increases, and from Unrestricted decreases. This makes sense as unrestricted stock can be sourced immediately, while a repair would take more time and thus is only possible if there is more time left to the UND.
Machine_group _HMI	The solution Conversion showed to have an increased probability of being the most effective solution if the EMO belongs to the machine group HMI. This makes sense since materials from this group are less often stored on *Unrestricted* stock locations.
SCP_2	This feature seemed to be mainly important for materials with a very high price. For *Semi-restricted*, *Conversion*, *Restricted*, and *Repair*, an increase of probability can be seen, while for *Unrestricted* this decreases, and for *DSP* it is unstable. Except from *DSP*, this makes sense as exceptional expensive materials are scarce and are often not stored on Unrestricted storage locations from which can be easily sourced.
Stock	As expected, the probability of *Unrestricted* being the solution increases when there is more stock available, as this feature mainly represents the storage locations belonging to this solution. Comparing the other solutions, it can be observed that they all follow a really steep slope except from *Semi-restricted*. The other solutions become redundant if there is normal stock available, but the *Semi-restricted* stock can some times be more effective if the lead time is better.
Fillrate	When the fillrate goes up, the probability of the solution *Unrestricted* goes up as well. This is in line with the feature stock which is one of the factors that determine the fillrate. Hence, no unexpected behavior was observed
Confirmation_ method _Manually_confirmed	By looking at the PDPs and SHAP beeswarm plots, it becomes clear that the probability of all other solutions than Unrestricted being the most effective solution increase and for *Unrestricted* itself decreases. This makes sense because the moment a SO got manually confirmed at the order creation, it will automatically be sent to manual planners. This often happens when there are is no Unrestricted stock available.
Criticality	Among the different criticality levels, the highest level "A" and the lowest level "Not critical", showed to be important. Where the former decreases the probability of the solution *Unrestricted*, it increases the probability for the other solutions. This makes sense since criticality increases if it is expected that materials are not directly sourceable from *Unrestricted* locations.
Material_type _SERT	The PDPs of this feature showed that the probability of the solutions *Semi-restricted* and *Restricted* increased if the material is a SERT, while the opposite goes for the other solutions. This makes sense as SERT materials more often are stored on (*Semi-*)*Restricted* storage locations where they are for example held by a local, cleaned, or inspected.
Count_NORA_ indicator_changes	As expected, the probability of *Unrestricted* being the solution decreases when there are more NORA indicator changes, while the opposite holds for the other solutions. A high number in changes shows that the order is sent back and forth to and by systems and planners, which indicates it is hard to find a effective solution.
Total_NB_orders	This features showed some instabilities as well which are probably caused by the distribution of the feature's values. This feature showed to be very important for the solution *DSP*, and has an increased probability of being the most effective solution when this value is high. This makes sense because when there are more outstanding New Buy orders at the suppliers, there are more options to source from *DSP*, which could at the end be a most effective solution.

Table 32: ML interpretability interview results

E.5 Evaluation of ML Interpretability methods

Method	Accuracy	Understandability	Efficiency
MDI	Moderate	Very high	Very high
PIMP	Moderate	Moderate-high	Moderate-high
SHAP summary plot	Moderate	Very high	Moderate
SHAP beeswarm plot	Moderate-high	Moderate-high	Moderate
PDP	Moderate-high	Moderate-high	Moderate
ICE	Moderate-high	Moderate-high	Moderate

Table 33: Evaluation of the ML interpretability methods on the goals defined by [Carvalho et al., 2019]

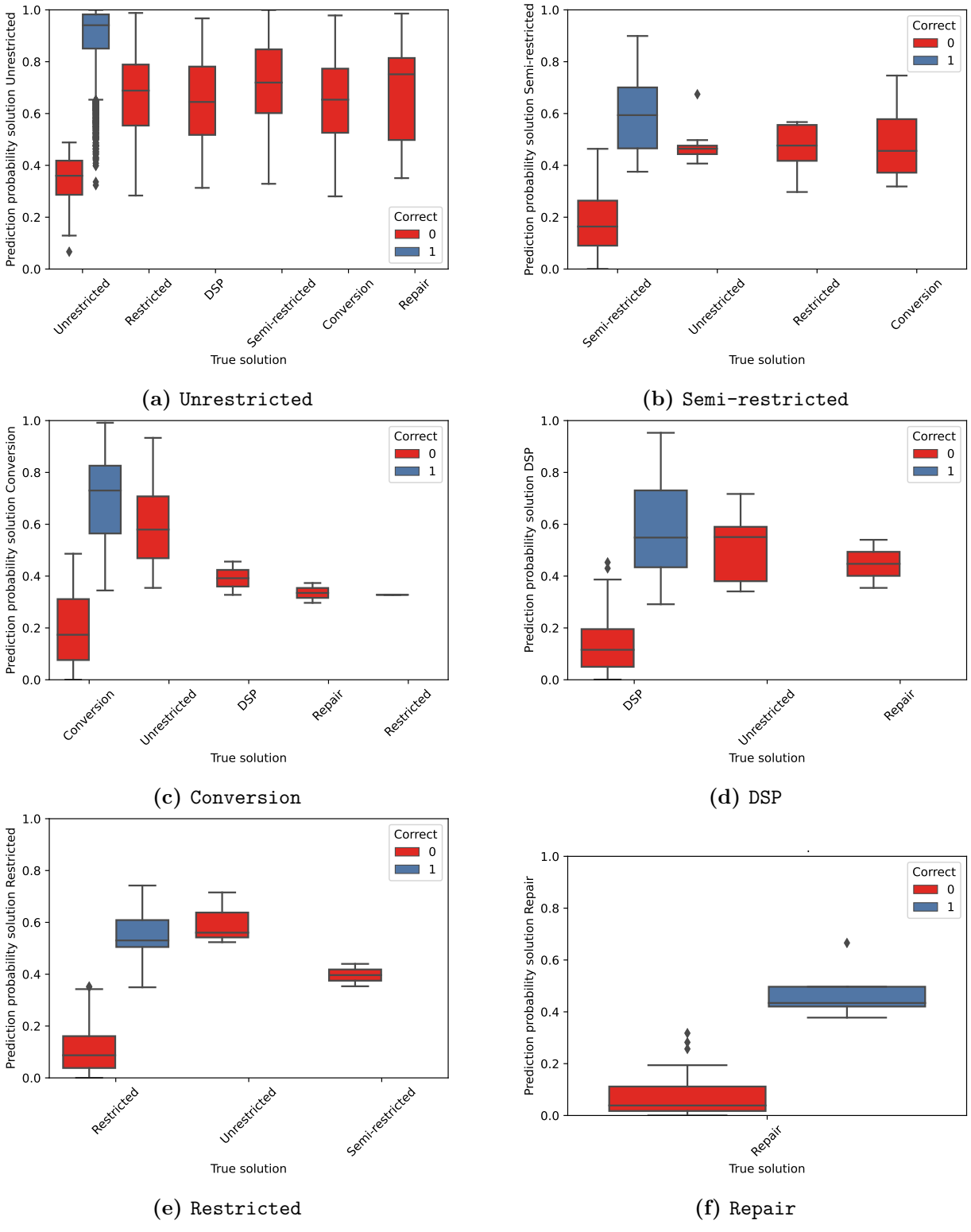
F Model Abstention

F.1 Cost estimations

	Unrestricted	Semi-restricted	Conversion	DSP	Repair	Restricted
Check 1	5	5	5	5	5	5
Check 2	3	3	3	3	3	3
Check 3	2					
Check 4		4	4	4	4	4
Check 5		2				
Check 6					2	2
Check 7						6

Table 34: Cost estimations in minutes per check performed per solution. The redundant checks are marked blue.

F.2 Uncertainty analysis



(a) `Unrestricted` **(b)** `Semi-restricted`

(c) `Conversion` **(d)** `DSP`

(e) `Restricted` **(f)** `Repair`

Figure 28: (Un)Certainty of the ML model when (mis)predicting on the predicted solution per True solution