Investigating different models that can be used to define the characteristics that influence customer behaviour in the on-line grocery sector

M.D.L.M. Middeweerd

Investigating different models that can be used to define the characteristics that influence customer behaviour in the online grocery sector

by

M.D.L.M. Middeweerd

Master Thesis

in partial fulfilment of the requirements for the degree of

Master of Science
in Mechanical Engineering

at the Department Maritime and Transport Technology of Faculty Mechanical, Maritime and Materials Engineering of Delft University of Technology
to be defended publicly on Thursday June 1, 2023 at 11:00 AM

Student number:	4560507
MSc Track:	Multi-Machine Engineering
Report number:	2023.MME.8804
Committee members TU Delft:	Dr. B. Atasoy
	P. Zattoni Scroccaro
Committee member ORTEC:	W. Merkx
Date:	May 25, 2023

Preface

This thesis is the final step in finishing the master's program Multi-Machine Engineering at the Delft University of Technology. Finding a topic was challenging but advantageous due to the many possibilities and interesting sectors related to this master's program. Combining my desire to understand how things work with my interest in optimisations, the grocery sector and economy were the drivers in finding an appropriate topic. At ORTEC, I had the opportunity to research a topic encompassing these aspects. The primary objective of this research is to improve the identification of customer behaviour with different customer choice models through real customer data from an online grocery store. The process of my research went in different stages. First, the study focused on the understanding of the data available. Secondly, the study revealed that the prediction performance could be improved. Based on that conclusion, the focus shifted to whether the models could be utilised to bring the simulation closer to reality. Finally, a beginning is made in analysing whether the models can be used in optimising the offer strategy. All these different aspects made the research very interesting but sometimes extremely difficult. Due to the challenges posed by the data, models and simulations, ORTEC allowed me to engage with subject matter experts throughout my research, which enabled me to acquire new knowledge and insights. The continuous support by ORTEC contributed to my technical experience in analysing in different manners. But also contributed to my programming experience in two languages, among others. As a result of all the experiences, I have acquired valuable insights that have significantly transformed my perspective on delivery services.

Upon finalising this thesis, I would like to thank my supervisors from the Delft University of Technology, Bilge Atasoy and Pedro Zattoni Scroccaro, for making time to give feedback and support throughout the process. Furthermore, I would like to thank Wouter Merkx for allowing me to conduct my research in an inspiring environment like ORTEC. His continuous support and the different views on my research were great. Additionally, I would like to thank you for all your guidance, meetings and critical notes throughout the process. Moreover, I like to thank everyone from ORTEC for the amazing time and fun discussions, as everyone was accommodating and interested in my subject. Lastly, I would thank my friends and family for their support during the moments that I, for instance, struggled with the information available.

Marloes Middelweerd
Delft, April 2023

Summary

Online groceries are becoming increasingly popular in the Netherlands, but how are the thin profit margins maintained in Attended home delivery (AHD) with the increasing success of online shopping? Grocery companies need help to be profitable because of the rising convenience request of customers who want smaller and cheaper time slots which are less cost-efficient for the company to execute. Due to the very competitive industry, companies must comply with these customer requirements to improve satisfaction and attractiveness and ensure engagement, providing a competitive advantage and driving long-term success. Demand management is introduced to manage these trade-offs for which customers' behaviour is significant information. The customers' preferences and critical market drivers can be found with customer choice modelling, which tries to identify a pattern between the offers and choices to understand how choices are made. Discrete choice models can be used to describe this pattern and explain customers' choice behaviours. This research analyses different choice models to determine if it is possible to identify customer behaviour better and optimise the offer set provided by ORTEC consequently. ORTEC is one of the world's leading providers of mathematical optimisation software and sophisticated analytics. It provides, among others, companies in AHD with their time-slotting service where supply and demand management is considered and adjusted based on the company's unique strategies. Since it is not possible for a company offering AHD services to excel in all aspects, ORTEC continuously optimises the time-slotting service to get as close as possible. Currently, ORTEC's simulations rely on simplified assumptions of customer behaviour. However, with improved identification of customer behaviour, they can examine whether it is possible to predict the effects of different strategies more accurately before implementation.

Customers' behaviours can be modelled with simple and readable parametric models of which the characteristics weights can be obtained directly. However, these parametric models have several limitations which affect the modelling performance, including lowering the prediction power. Based on the constraints and increasing data, non-parametric data-driven models (including Machine learning (ML)) are proposed nowadays to improve prediction accuracy. However, ML classifiers also have a limitation as they may be harder to understand without models providing insights.

Analysis of the F1-scores shows that both the Multinomial logit (MNL) and Neural Network (NN) models outperform the Benchmark model, derived from assumptions from the data, in accurately predicting customer behaviour. The feature importance plots show that the weight distribution between the two models varies, as expected, due to the differences in learning mechanisms and underlying architectures, which allow for different feature probability distributions. Moreover, the architecture of the models differ not only between the models but also between the three prediction steps required to determine the delivery day component. It is decided to perform the prediction in three consecutive steps to avoid significant performance degradation. The different architectures in and between the prediction process are obtained by hyperparameter tuning that optimises each step, using four solvers/activation functions suitable for multi-class problems and considering whether balanced class weights should be applied. The results are compared based on the F1-score. The NN model required additional classification decisions during the tuning process, such as determining the number of hidden layers and their width. The activation function for the output layer was consistent across all steps, using the commonly applied Softmax function for such problems.

Simulation tests are conducted since it is found that the introduced models enhance the prediction performance of customer behaviour and, will therefore be used to establish whether they enhance the simulation and bring it closer to reality compared to the Benchmark model. To determine the impact on the key performance indicators (KPIs) consisting of the number of offered time slots, the total executed routes, and the number of assigned people will be analysed. This set of KPIs needs to be analysed together as each provides valuable insights into different aspects of the outcome and provides a complete overview. Based on these KPIs, it is found that both the MNL and NN bring the simulation closer to reality compared to the Benchmark model. Therefore, it is possible to make a start to investigate whether the MNL and NN models can optimise the offer set strategy. Test scenarios are created to iden-

tify this and take both supply and demand management into account. The first test scenario relates to the execution cost and the probability of a time slot being selected. In the second scenario tested with the simulation tool focuses solely on the probability of selecting a time slot. The same choice model is used in both scenarios' strategy and time slot selection. Subsequently, all test scenario simulations are compared with the real results. After comparing, it is found that the choice models could potentially optimise the offer strategy using the first test scenario and show promising results for further opportunities. However, further research is needed to assess the exact impact of the offer strategy and can be done using different choice models. Likewise, other offer strategies can be tested based on different features, for example, the effect of price incentives, since the probability of selecting a time slot can be more accurately determined.

To conclude, this study aims to determine if customer choice behaviour can be better identified and used to optimise the provided offer set in the context of online groceries. Therefore, this study employed the parametric MNL and non-parametric NN models to evaluate customer behaviour, with both models outperforming the Benchmark model after hyperparameter tuning. The NN model performed best, but the MNL and NN models evaluated feature importance differently. After the observation that the MNL and NN improve the prediction performance, simulations are performed to determine if including the models enhances the simulations and brings the results closer to reality than the Benchmark model. By evaluating the results, it is found that including the MNL and NN models improve the performance of the simulation, allowing for a first attempt to see if the models can be utilised to optimise the offer strategy. For this reason, test scenarios are created to evaluate the influence of the strategies compared to the real model. Promising results are found based on the filtering techniques in this study. Nevertheless, only a first exploration is conducted to investigate the potential and future studies can delve much deeper into this topic. In these studies, it is recommended to utilise different features to create the strategy and look into the impact of price incentives or green labels to nudge the customers.

Contents

List of abbreviations

AHD - Attended home delivery
AUC -Area under the Curve

BAG - Bagging trees
BN - Bayesian network
BOOST - Boosting trees

DAG - Directed acyclic graph
DCM - Discrete choice model
DGP - Data generating process
DNN - Deep neural network
DT - Decision tree

ELU - Exponential Linear Unit

FN - False negative
FP - False positive

GAM - Generalised attraction model
GB - Gradient boosted trees

IIA - Independence of irrelevant alternatives

KNN - k-Nearest Neighbour
KPI - Key performance indicator

lbfgs - Limited-memory Broyden-Fletcher-Goldfarb-Shanno

ML - Machine Learning
MNL - Multinomial logit

NB - Naive Bayes
NL - Nested logit
NN- Neural network

ReLu - Rectified Linear Unit
RF - Random Forest
RFM - Recency, frequency and monetray value
ROC - Receiver operating characteristic

sag - Stochastic Average Gradient
SHAP - SHapley Additive exPlanation
SVM - Support Vector Machine

Tanh - Hyperbolic Tangent
TN - True negatives
TP - True positives

VRPTW - Vehicle routing problem with time windows

List of Figures

List of Tables

1
Introduction

The Netherlands has a growing population of more than seventeen million inhabitants and is one of the world's most advanced economies [27]. In the last decade, e-commerce revenue increased to approximately thirty billion euros in 2021. The digital environment opened new revenue streams for grocers as consumers increasingly opt for online grocery shopping. Doing online groceries was not commonplace in the Netherlands before COVID-19; nevertheless, shopping regulations, supply shortages and particularly safety regulations resulted in unexpected growth rates in e-grocery [95][12]. After the lockdown and when ordinary life was resumed, the numbers decreased again. However, the supermarket revenue generated by online grocery purchases accounts now for seven and a half per cent, which is still an increase relative to prior COVID-19 [43][22]. The total revenue of online supermarkets was over two billion euros in 2021, where the three most significant online supermarkets[1] were taking almost ninety per cent of the total market share [101][82].

Even though the digital industry has expanded, most Dutch shoppers experience a different level of satisfaction online than they have after an offline experience. Customer satisfaction is fundamental for online profitability as satisfaction drives loyalty, and loyalty results in more extensive frequent baskets. Crucial driving factors for satisfaction are simplicity, findability and especially experience. The improvement of findability is a must for light, inexperienced shoppers as the dissatisfaction rates are currently generated by missing options for comparison and pack size. Getting the satisfaction of heavy buyers is a more significant challenge as the experience strongly influences them, and they are more inclined to switch retailers [95].

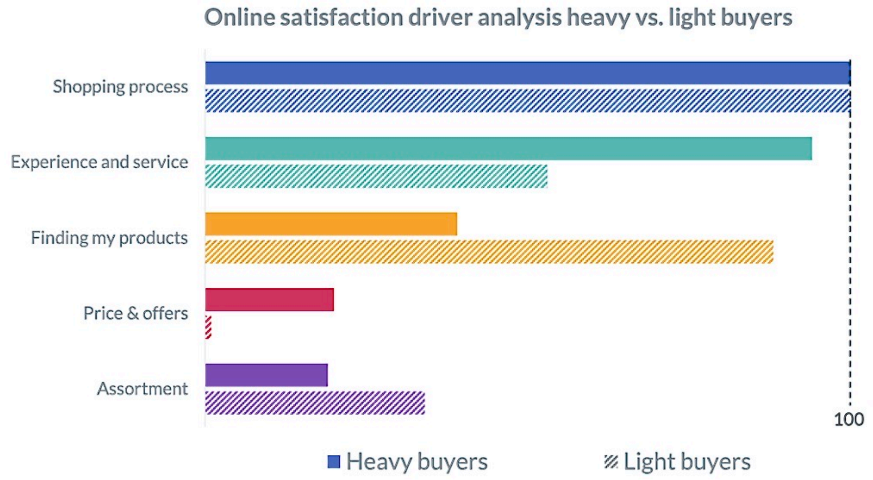


Figure 1.1: Analysis of online satisfaction drivers of heavy and light buyers [95].

[1] Albert Heijn, Jumbo and Picnic

As illustrated above, the customer experience is a considerable driver of the market success of the supermarket as the customer-to-customer interactions through social media create opportunities and, on the other hand, challenges for firms [63][58]. When customers exchange thoughts and ideas with each other, they develop their sense of relationship with a firm through identification with others which can build feelings of confidence and control [93]. However, it is proposed that customer experience is dynamic, which means that next to, for example, the influence of other customers, the current experience is affected by the previous one [98]. Nonetheless, the key reason for buying online groceries is convenience through time and effort saving with minimal physical and mental effort, which is why online retailers offer customers an online grocery shopping experience, where the groceries will be delivered to the customers' front doors or even into the kitchens. For a favourable experience, delivery must be on time, with good quality products and is the interaction between the customer and delivery service [91][113][17].

1.1. Overview of Attended Home Delivery

The convenience of online shopping is experienced through saving time and effort, which is the primary reason customers buy online [42]. When doing online groceries, people do not need to consider the store's specific opening and closing times and allow customers to do the groceries whenever and where they want. As a result, the groceries can be done online while waiting or travelling, which is a significant advantage in terms of time and effort saving as purchasing groceries is the most common and frequent shopping activity. At the same time, online grocery shopping allows customers with reduced mobility, without transport or free time to get a hold of their groceries [40].

To enhance the favourable experience, buyers expect that the delivery is on-time and contains good quality products. To achieve that, the retailer must deliver the groceries to the door at a pre-arranged time slot chosen by the customer. This type of home delivery is called Attended Home Delivery (AHD) and is needed as fresh groceries can spoil, and customers want good quality products. AHD is common for grocery deliveries and other products, such as home services and products purchased online that need security or special handling. With AHD, the delivery is direct to the customer's front door and is based on delivery time slots to make it as convenient as possible for the customer. However, this convenience will create significant logistical challenges for companies because the customer decides which time slot it selects, directly impacting the operation efficiency. Despite the inconveniences that the company may experience, there are also advantages to having customers choose their preferred time slots, as it can prevent costly delivery failures as much as possible [60]. Besides, the customer might choose the time slot, but the company determines which time slots they offer and for which fees when it uses dynamic pricing. However, when a company expands its offer set, the convenience improves, but the deliveries will be more diffuse. Diffuse deliveries mean fewer visits per vehicle in a specific zip code will be done as fewer customers are assigned to a route, resulting in higher delivery costs. The same holds for shorter time slots; it enlarges customer convenience but reduces the routing flexibility and thus increases delivery costs. Therefore, companies will use the opportunity to change the offer set and temporarily close popular time slots to overcome these problems. However, in doing so, the company must face complex trade-offs between customer preference and efficiency. The number of time slots offered is determined mainly based on capacity, cost and delivery location. Consequently, it might be the case that the provided time slots do not meet the customers' expectations, reducing the attractiveness of online groceries as the suitability to the provided time slot is low. Three conditions need to be considered to generate suitable time slots and to minimise the attractiveness reduction:

1. The potential of acceptance by the customer has the highest priority, meaning that an empty offer set should be avoided as it will lead to disappointment.
2. The collection of time slots should deviate as little as possible from the customer's preferences regarding the time of the day.
3. The length of the offered time slots is crucial as length and availability affect customer satisfaction and, thus, suitability [3] [26].

As above-mentioned, a trade-off must be made between customer preferences and efficiency to manage profitably and suitability. To nudge customers to other time slots than their preferred one, incentives can be used. Incentives, including price and green labels, influence customer behaviour and try to convince customers to select other or longer time slots that are more cost-efficient when needed [3]. Price incentives consist, for example, of discounts, and green labels indicate the greenest option; however, they might impact the perceived customer service [2]. These trade-offs between supply and demand management are made to manage profitability as online grocery companies have to deal with low-profit margins and struggle to be profitable [10]. The supply side seeks the most cost-efficient fulfilment of a given demand with supply chain planning, inventory management and vehicle routing to manage this. In contrast, customer demand focuses on given supply capabilities in the best possible way. Hence, the time windows need to be short enough to be accepted but long enough to provide flexibility for creating efficient routes [60].

1.2. The scope of the report

This study will focus on customer choice behaviour regarding online grocery home delivery time slots. Based on this primary focus, different topics are researched, including which requirements are incorporated in the delivery, how the delivery process is managed to be profitable and most importantly, how customer behaviour can be modelled. Determining the behaviour of customers regarding choices can be done in different ways. In literature covering the retail sector, it can be found that customer choice models are used for various purposes, including assortment decisions, pricing and especially for profit optimisation. As profitability is considered the primary objective in many papers, the easily understandable parametric models that give insights into the characteristics that influence customer behaviour are redeemed for the more advanced non-parametric data-driven models. These models include machine learning algorithms and have higher accuracy and predictive power. However, these models are commonly used as a black box as the purpose is not to understand the motivations in customers' choice behaviour but to gain the highest prediction accuracy possible. Nevertheless, it is considered lucrative to understand the journey in general and specifically the customers' behaviour by identifying and understanding the drivers in this choosing process. From these findings, in combination with the research, the following question in this study will be addressed:

How can customer choice behaviour be better identified and used to optimise the provided offer set in the context of online groceries?

To answer this question, four subquestions are created to break the question down and to explore and clarify the different characteristics. The subquestions are:

- *What is customer choice behaviour in Attended Home Delivery, and how can it be identified?*
- *Which customer characteristics influence the behaviour the most?*
- *What is the difference between modelling with Machine Learning models and parametric models?*
- *Does the learned behaviour affect the simulation's routes and the number of offered slots?*

This study contributes to prior research in the grocery sector, which has primarily focused on the supply chain but has recently shifted its attention to demand management and modelling customer behaviour. However, current customer choice models used in retail are often simplistic, and advanced models utilised in other sectors have shown promise but are frequently employed as black boxes. Therefore, this research seeks to extend the existing literature by using advanced customer choice models to model customer behaviour in the retail sector while providing insights into the model and identifying which features are essential. The remainder of this review is set up as follows. It continues with section 2, where information from the literature is gathered to understand how the order process works and how the behaviour of customers selecting a time slot can be modelled. In addition, it also provides a more detailed examination of different parametric and non-parametric models used for modelling customer choice behaviour, offers examples of where the models are used, and identifies trends in model development. Subsequently, section 3 outlines the proposed methodology for this study, starting with a discussion of the demand management framework and how the growing amount of data can be managed in combination with strategies influencing customers to select cost-efficient time slots without sacrificing attractiveness. This section also addresses feature selection and how to compare parametric and non-parametric models of customer behaviour. A Benchmark model, derived from assumption form the data, is introduced to ensure that more advanced models add additional value, and prediction performance results are discussed. Following with section 4, where the data retrieval process is explained. Initial and comprehensive analyses are performed to determine better which features and whether time slot class weights are needed before the hyperparameter tuning occurs. In section 5, the final architecture of the advanced models is determined after discussing the results of the hyperparameter tuning. This section also analyses the models' prediction performance and feature importance and provides confusion matrices and probability density plots to improve the understanding of the models. Based on the discussed results and the improved prediction performance relative to the Benchmark model in section 6, the outline of the used simulation and test scenarios will be explained. These simulations will be used to determine the effect of the improved customer behaviour on optimising the offer set. The last section, section 7, will give a summary and a conclusion of the findings and ends with a discussion indicating relevant areas for further investigations.

2 Literature review

Before the research question and the four subquestions can be answered, relevant information from the literature will be gathered. The literature review will focus on how the ordering process in online groceries works, how customer behaviour can be modelled, and which trends in customer choice models are present. The methodology for identifying customer behaviour will be presented from this literature, after which it will also be applied in a case study. The knowledge of the order process is of great importance for the simulation, which will be carried out later to measure the influence of better-identified customer behaviour in online groceries. Key performance indicators (KPIs) need to be known and established to see if the behaviour adds value to the process. Therefore this literature review will also investigate which strategies and KPIs are available in the online grocery sector to apply the simulation.

2.1. Order process

Previous research focused on the supply chain with the consequence that the supply-oriented approaches have been studied for many years. However, demand management is now receiving more attention, resulting in more advanced technologies to understand customer behaviour better. A better understanding of customers' behaviour is already followed in companies' different service strategies and offerings. This is an expected result, as a company cannot excel in the highest quality, fastest delivery, and most excellent variety at the lowest price [110]. For example, the result is that one company proposed different delivery fees and more time windows per day of different lengths. In contrast, another company proposed one free time window of one hour for each day of the week to customers [112].

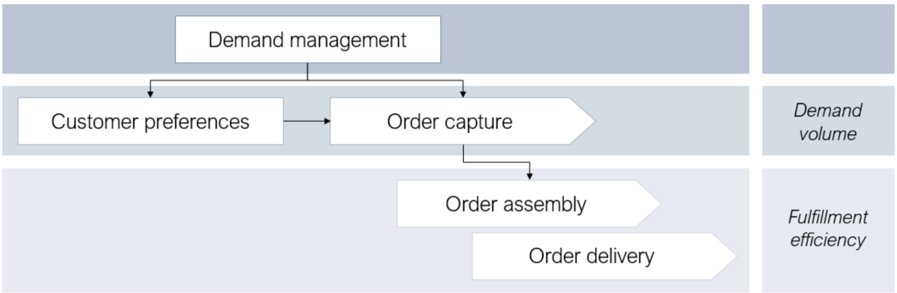


Figure 2.1: Effects of demand management [112]

In AHD, demand management can be used to maximise the overall profit, as the planning can be seen as an assortment of delivery options. Therefore it can be linked to assortment planning for physical products, which is thoroughly investigated; only AHD impacts the delivery costs [15]. The demand management of AHD intends to shape and generate customer demand to benefit the fulfilment process. In other words, it aims to manage the trade-offs between generated demand volume and fulfilment efficiency, as visualised in Figure 2.1, and thus between interrelated cost and revenue effects. In this context, the customer's order decoupling point is the most relevant part of the fulfilment process, which consists of three main steps: order capture, assembly, and delivery [25]. In the order capture, the customer and the company agree on when and where the order will be delivered. This agreement comes forward when the company shows different time slots, as mentioned above, from which the customer chooses one. This booking process needs to be smooth, so the company needs to provide the time slots in at most a few seconds, after which the customer can select the preferred one [49]. An order will only be placed when the offered time slots meet the customer's preferences and expectations. When the order is placed, the assembly will be scheduled, consisting of warehouse operations to prepare for delivery. The delivery will occur where the physical delivery is within the selected time window. An overview of a possible ordering process can be found in Figure 2.2.

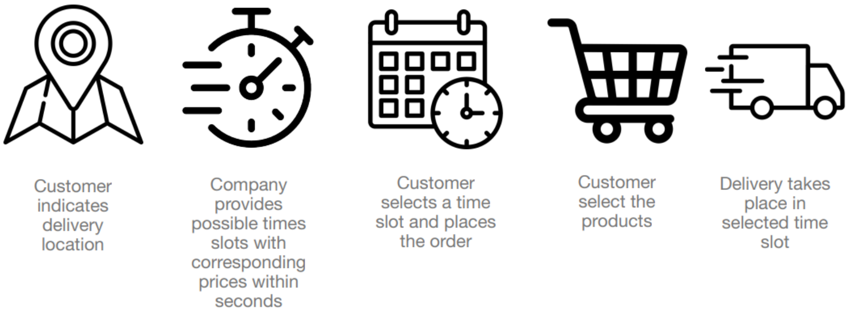

Customer indicates delivery location
Company provides possible times slots with corresponding prices within seconds
Customer selects a time slot and places the order
Customer select the products
Delivery takes place in selected time slot

Figure 2.2: Ordering process from a customer perspective.

In this figure, the first step in the ordering process is that the customer has to indicate the delivery location. Following the indication of the delivery location, the company provides feasible delivery time slots with corresponding prices, when applicable. The customer then chooses a time slot and places the time slot order. After reserving a time slot, the customer has to add products to the online basket. This step can be conducted during the selling horizon, where a finite set of products is offered to heterogeneous customers [103]. The order can be finalised whenever the customer has completed selecting the products. In this step, the company confirms the delivery, starts the assembly and prepares the delivery to be executed in the predetermined time slot [60]. However, this process can differ per company since some start with the customer selecting the products rather than selecting a time slot.

Above in Figure 2.1, the critical steps of the fulfilment process in AHD service are highlighted. How efficiently these are conducted depends on different optimisation processes in the assembly and delivery phases and the preferences of individual orders. For instance, the assembly in larger fulfilment centres can be done (semi-) automated or in waves to reduce time and labour, and the location of the centres also plays a role in the lead time. The delivery routes are planned in the delivery phase, where different optimisation choices can be made. Nevertheless, the processes are linked to customer preferences and the available service options. Small price incentives can control the demand as the customer choice behaviour will be influenced [24]. Not only will customers' behaviour be influenced, but some customers may be shown the same service options because availability is checked first as to whether a particular time slot is feasible given already accepted customers. The feasibility during the order capture is quickly checked by anticipating and rapid assessment of the delivery step of the order based on a vehicle routing problem with time windows (VRPTW). However, it is still possible that a feasible time slot may not be shown as it can be decided not to offer. This decision is based on the request's generated profit (before delivery), the expected choice behaviour and the opportunity costs of serving the customer request in a specific time slot. After all, it may be advantageous to reserve it for more attractive future customers or direct the customer to a more suitable service option.

Different costs, in combination with the company's KPIs, influence whether a time slot is offered. The company's objective is to maximise the expected total profit after delivery and the expected total profit before delivery, less the total expected delivery costs. In other words, the decision on whether it is favourable to offer a time slot is based on the opportunity costs. The opportunity costs to serve a customer within a specific time slot include, among other things, the marginal delivery cost and the cost of future orders that need to be displaced. The future order displacement costs are included as customers cannot be served due to delivery capacity and time while serving the current customer request. However, ultimately the opportunity costs depend on the realised order and the final delivery route. The final route is only known after the booking horizon ends since only the already placed orders are known before. Consequently, expectations about future delivery locations and time slot choices have to be considered. Considering that decisions regarding future offers determine the expected choices for time slots, the opportunity cost estimate should be carefully coordinated with the demand management of customers who are anticipated to make requests following the current one using the VRPTW's solution [68].

In the ordering process of AHD, demand management indicates two primary levels that influence the demand: the offered delivery time windows and the corresponding price for delivery. Both can be determined as dynamic or static; an overview of the different options can be found in Table 2.1. In this table, the dynamic approaches make decisions per customer request during the booking horizon, and the static approach makes decisions before the start of the booking horizon based on specific characteristics and is not updated during the process [68]. The specific characteristics are fed by previous forecast data and can be used by deciding on the amount, length and discount of time slots. If new data is available, the static model should update and becomes, in this way, a dynamic model instead [116].

	Time slot allocation	Time slot pricing
Static	*Differentiated slotting*	*Differentiated pricing*
Dynamic	*Dynamic slotting*	*Dynamic pricing*

Table 2.1: The classification of the four demand management concepts [5][116].

Dynamic price incentives are used to steer for a balanced demand over the week and day and are based on the time slot's popularity. When uniform pricing is preferred over dynamic, and no dynamic slotting is used, the demand typically produces imbalance as the delivery capacity is relatively inflexible, resulting in over capacity. The number of offered time slots can depend on the corresponding zip code's demand volumes, which may result in lower service quality. This means that geographic areas with a low demand receive fewer time windows than areas with a high customer demand to retain efficient delivery routes and achieve economies of scale. In other words, demand management is similar to revenue management as it maximises the revenues generated with a predetermined capacity and aims to exploit market heterogeneities [5]. Based on the heterogeneity, the market can be partitioned into segments with various sensitivities and preferences [4].

Compared to the airline industry, revenue management can be used as the segmentation is between business and leisure travellers. In this example, business travellers have a higher willingness to pay and have a different valuation regarding flexibility and cancellations than leisure travellers. Based on this segmentation, airlines can do better than simply selling based on first-come-first-served at a fixed price and uses the flexibility to adjust prices and volumes offered to different segments in real-time. However, e-grocery combines physical products and delivery services and has to consider the product dimensions as it affects revenues and capacity. Therefore, it has a significant cost impact as the customer influences the costs of the delivery based on location, time and order size and translates demand management to profit management rather than revenue management [5].

Based on demand management, the offered time slots to customers and price need to be determined. Where the geographic demand is comprehensible, as a minimum demand can be required to justify the area, determining the available capacity is less evident than initially appearing. Within the capacity also, the picking capacity in the warehouse and available driving time are included next to the physical fleet size. Resulting in that clustering orders is directly linked to transportation planning. That is only part of it, as with demand management, the capacity is not sold with a first-come-first-serve mentality, and the segments based on heterogeneity require more differentiation between orders. It may be more beneficial to reserve scarce capacity for the most profitable consumers. The segments emerge based on heterogeneity and can be partitioned based on different factors. For example, order size, as losing a large order from a frequent customer, is worse than losing a small incidental order. Delivery location, as mentioned above, or the customer's flexibility in choosing a time slot, determines the offered time slot in demand management [102].

In addition to time slotting, pricing is an even richer tool for demand management as it has a finer gradation of incentives. However, the difficulty with pricing is determining the magnitude of the discounts and premiums, as discounts can impact the margins and spoil the reference price. Besides, the prices can aim at different goals, influencing the basket and, therefore, the corresponding revenues.

Demand management may lead to time slotting and pricing benefits, such as offering smaller time windows without affecting efficiency and reducing the risk of failed deliveries. However, customers may also receive unexpected price changes as unfair, or when it follows a regular pattern, they will learn to anticipate them so that the effect will be limited. A good understanding of consumer behaviour and delivery cost dependencies is needed to obtain these benefits [5]. For example, when a multinomial logit (MNL) model to describe customer behaviour is included in revenue management, the revenues can be increased by up to five per cent comparing methods where the choice behaviour is not considered [116].

2.2. Customer choice model

As mentioned in the introduction, not only the number of available time slots determines the satisfaction and attractiveness of the delivery service but so does the length of the offered slots. The length impacts not only the attractiveness of the time slot but also the efficiency of fulfilment [6]. Discrete choice models (DCMs) are used to describe how a customer chooses an option from a set of alternatives to ensure customer satisfaction and attractiveness and the company's efficiency. Therefore, it explains customers' choice behaviours and is widely used in psychology, economics, transportation, marketing and operations studies. In recent years, customer choice models have increased sharply because of the growth in online retailing as they can identify customer behaviour and be used to make efficient pricing and revenue management decisions, among others [37].

Customer choice modelling is a scientific method used to find the critical market drivers by comparing choices among choice sets and measuring the customer's preferences [109]. This means that customers select one option from a series of multiple competing offers and that the modelling approach tries to identify a pattern between the offers and choices to understand how choices are made [87][90]. During this process, it is assumed that the customer sees all the options together at a certain time and decides based on preferences. And this process can be considered binary since the customer chooses a particular option, and each choice affects the model [48].
When broadly speaking, there are four different sorts of customer choice models, and an overview is given in Table 2.2. To achieve appealing structural qualities, the models impose some amount of independence or limit the level of dependence of the consumer's choice decision on the available options. These independencies can be relaxed a bit to develop a more generalised model that is more widely applicable. However, more parameters are often needed to specify the model in this case, and a trade-off between parametric and non-parametric models needs to be made.

In DCMs, the choice probabilities can be modelled for individuals or consumer segments. The segments are determined based on the characteristics of customers' choices in combination with clustering techniques. A segment consists of customers with similar behaviour concerning their preferred time slots, and between the segments, there are different requirements and needs [85][59]. After understanding how choices are made, predicting customers' future choices and the related costs may be

Attraction model	Assigns an attraction value to all available options, and the probability of choosing the option is proportional to this value.
Utility based	Determines the option with the highest realized consumption utility. For the selection, different criteria are possible.
Temporal model	The preference of a customer depends on the occurrence sequence of corresponding events.
Rank-based model	Customers rank all the available options and chooses the highest ranked option.

Table 2.2: Four different customer choice models [37]

possible. For example, a generalised attraction model (GAM) is used to determine customers' choices, which anticipates opportunity costs comprising marginal insertion costs and future displacement costs [68]. The earlier mentioned price incentives can influence customers' behaviour and nudge them to cost-efficient time windows. The used incentives can have different forms, for example, the size of the delivery charge depends on the time slots, including discounts or points when choosing unpopular slots, or the environmental impact is indicated. With these incentives, the impact of future orders is tried to predict with a dynamic decision model so that the profit can be maximised as the customers' choices directly impact the delivery costs. In addition to the GAM, the most commonly applied MNL model can be considered to determine customer choices. Despite MNL being the most widely used model, GAM allows for demand overestimation to circumvent customer dissatisfaction with increasingly tight time offers, which is not considered in the MNL model. These two models are parametric and incorporate utility, but alongside parametric models, non-parametric models are increasingly common in choice modelling [116].

2.3. Parametric models

The first type of model is the parametric model, embedded in random utility theory and characterised by simplifying the function to a known form. In parametric models, the data is summarised through a collection of parameters that are size fixed and are determined by assumptions about the underlying data distribution. The random utility theory assumes that customers associate a particular utility with every product as each product is a choice option and makes a decision based on maximising utility. In this study, the different time slots are the products; consequently, a DCM can be derived. The utility of a choice option consists of a deterministic component, u_j, which is the mean utility of the alternative and a random component, ϵ_j, with mean zero. The combination of these two components is expressed as a utility with Equation 2.1.

$$U_j^l = u_j^l + \epsilon_j^l \tag{2.1}$$

where j is the alternative in a set of products offered to a customer and l indicates a particular segment as a customer population is assumed to consist of $\mathcal{L} := \{1, ..., L\}$ segments.

The probability that the customer chooses product j can be determined with Equation 2.2. However, it is always possible that a customer does not choose or buy from a competitor and is indicated with U_0 [103].

$$P_j(S) = P\left(U_j = max\left\{U_{j'} : j' \in S \cup \{0\}\right\}\right) \tag{2.2}$$

where S indicates the offer set $S \subseteq \mathcal{J}(c_t)$ with c_t the available inventory at time t.

2.3.1. Multinomial logit model

The MNL model is the most widely used and assumes that the decision-maker chooses a time slot that maximises their utility as described in Equation 2.2. This model assumes that the entire segment can be described with the same parameters. Multiple customer segments can be combined with demand management that looks at differences in customer preferences and characteristics. The MNL requires multiple identified segments to provide an accurate prediction model based on the different characteristics. In this way, the model can be used separately for each segment. When it is unknown to which segment the customer belongs, the individual segment-level MNL models are linked. The probability of the customer belonging to the segments is determined. In other words, a finite mix of MNL models is called the finite-mixture logit or latent class model [103].

This model's random component, ϵ_j, is assumed to be independent and identically distributed. Following a Gumbel distribution, the deterministic part, u_j^l, is assumed to be a linear function. The linear function can be found in Equation 2.3 where β_0^l is the base utility across all options, β_j^l the utility of the time slot, and β_d^l the sensitivity of utility of the delivery charge. When no delivery time slot is chosen, and the order is lost, the utility can be indicated with $u_0 = 0$ as it is normalised to zero.

$$u_j^l = \beta_0^l + \beta_j^l + \beta_d^l d_j \tag{2.3}$$

where d_j the delivery charge is of time slot alternative j.

One of the characteristics of MNL models is the independence of irrelevant alternatives (IIA). In other words, the relative odds of choosing one option over the other are not dependent on the attributes of the other available options. This implies that proportional substitution occurs across alternatives which may lead to the overestimation of choice probabilities by customers [103]. The IIA is not a specific feature of choice models as the more complex models, for example, the nested logit model, do not have the same problem. However, the MNL in AHD can use standard maximum likelihood methods as the delivery location is known in advance [116]. Thus the β^l sensitivity parameters are estimated by maximising the log-likelihood function concerning β^l and the probability that time slot j is chosen given the offered time slots $j \subseteq S$ enclosed with the delivery charges $\vec{d}$ can be found in Equation 2.4.

$$P_j^l(\vec{d}) = \frac{exp\left(\beta_0^l + \beta_j^l + \beta_d^l d_j\right)}{\sum_{k \subseteq S} exp\left(\beta_0^l + \beta_k^l + \beta_d^l d_k\right) + 1} \tag{2.4}$$

Estimating the β^l parameters includes three sets of historical periods, indicated with h. Where $\mathcal{P}_h$ denotes the set of chosen time slots, $\bar{\mathcal{P}}_h$ indicates the set of customers that did not select a time slot, $\bar{\bar{\mathcal{P}}}_h$ the set of periods without arrivals, and F_{ht} the offered set of time slots at delivery charges $\vec{d}_{ht}$, and $j(h,t)$ the chosen time slot j in history h at time t. The log-likelihood that needs to be maximised to estimate the β^l can be found in Equation 2.5, and the global maximisation can be found with the use of standard non-linear programming solvers such as the Quasi-Newton method [116].

$$\begin{aligned}\mathcal{L}\left(\beta^l\right) = \sum_h \sum_{t \in \mathcal{P}_h} \Big[\beta_0^l + \beta_{j(h,t)}^l + \beta_d^l d_{j(h,t)} \\ - \log\Big(\sum\nolimits_{k \subseteq F_{ht}} exp\left(\beta_0^l + \beta_k^l + \beta_d^l d_k\right) + 1\Big) \\ + \sum_h \sum_{t \in \bar{\mathcal{P}}_h} \Big(-\log\Big(\sum\nolimits_{k \subseteq F_{ht}} exp\left(\beta_0^l + \beta_k^l + \beta_d^l d_k\right) + 1\Big)\Big)\end{aligned} \tag{2.5}$$

As mentioned, the MNL is a widely used model to model customer choice behaviour in different sectors such as retail, airline and the travel industry. The MNL model is then used to optimise the profit [69]. An example where the MNL model has been used to optimise profit is time slot pricing. A dynamic pricing policy is employed based on the customer delivery slot choice model [115]. In addition, the model can be used to determine customers' utility regarding service attributes and the willingness to pay [9]. Or the MNL can be used as a Benchmark model to compare the working of newly introduced models that model customer choice behaviour.

2.3.2. Nested logit model

The Nested logit (NL) model aggregates alternatives that share unobserved common attributes into nests, meaning that alternatives can be partitioned into subsets. In this way, IIA only holds within a nest but not across nests [39]. That the IIA holds within the nest is since the probability ratio of an alternative is independent of the attributes or existence of all other alternatives. On the contrary, the IIA does not hold across nests, as the probability ratio of alternatives in different nests depends on the attributes of other nests' alternatives [104][108]. The nets correspond to various categories of products, and different variants of the product category can be found within a nest [38]. After the nests are created, a customer decides which nest to purchase from, or if no purchase is made, the preference value can be indicated by v_0. An alternative within the nest is selected if a nest is chosen. The nests are indicated with the variable S_k where $k \in K$ indicates the set of the nests and $v_{kj} = exp\left(\frac{u_j}{\mu_k}\right)$ the preference value for product j for nest k is. The preference value consists of u_j, indicating the visible part of the utility and μ_k, a measure of independence in unobserved utility among the alternatives in the nest. The value of μ_k is restricted to the range between zero and one. A lower value implies more correlation among the alternatives in the nest. When the value of μ_k is equal to one for all nests, the model is equivalent to the MNL model, where there is no correlation between the alternatives [39]. In addition, the overall preference for a nest can be indicated with $V_k(S_k) = \sum_{j \in S_k} v_{kj}$ [103].

The choice probability of product j from nest k can be determined following Equation 2.6. The parameters can be estimated by simultaneous or sequential maximum likelihood. A bottom-up approach is used with the sequential form, meaning that the lower modes are estimated first. Subsequently, the estimated coefficients are included as explanatory variables in the second layer [104].

$$P_{jk} = \frac{v_{kj} V_k(S_k)^{\mu_k - 1}}{v_0 + \sum_{h \in K} V_h(S_h)^{\mu_h}} \tag{2.6}$$

The NL model is, for instance, used to get insights into customers' behaviours, and with the behaviours, the price of a time slot is determined. During this research, they use different lengths of time slots and allow overlap between a long and short time slot [60]. In addition, the model is also used to demonstrate the effects of changing utility variables, service factors, and socioeconomic and delivery activity factors on the choice probabilities [119].

2.3.3. Generalised attraction model

The GAM model allows for partial demand dependencies in estimation since the MNL model is too optimistic and overestimates probabilities as it is a particular case of the primary attraction model and operates under the simplest type of customer choice process. To overcome the overestimation limitations, the GAM model is addressed, and the attractiveness of options includes the not offered options, which results in a less optimistic but not too pessimistic model. In other words, the GAM model includes the possibility that a time slot is chosen at a different company rather than leaving without a time slot. The Expectation-Maximisation algorithm is utilised to estimate the parameters of the GAM model, and the market share needs to be known to improve the estimated parameters. Regardless, the offered time slots and the corresponding customer choices are the only necessary data to estimate. The probability that a customer chooses time slot j from the offered slots can be determined with Equation 2.7 where v_j a measurement of the attractiveness of the different choices is and w_j the external alternative attraction value. Nevertheless, the GAM model is a limiting case of the NL model, where the offerings within each nest are perfectly correlated.

$$P_j(S) = \frac{v_j}{\widetilde{v}_0 + \widetilde{V}(S)} \tag{2.7}$$

with $\widetilde{v}_j = v_j - w_j$, $\widetilde{V}(S) = \sum_{j \in S} \widetilde{v}_j$ and $\widetilde{v}_0 = v_0 + W(N)$ where $S \subset N$. Losing the order is indicated with $j = 0$, and N indicates all options, including the external alternative [38].

The GAM is, for example, used to optimise managing demand through dynamic time slot allocation. Where the dynamic allocation considers the GAM-determined customer behaviour and the approximation of the opportunity cost [68].

2.3.4. Markov chain choice model

The Markov chain choice model is a good approximation to any random utility DCM under mild assumptions. The model generalises widely used DCMs and permits computationally efficient unconstrained assortment optimisation. However, the parameters of the choice model are determined using data from an underlying model. The choice probability computed by the Markov model overlaps with the probabilities of all products and assortments given by that model. The data used are the choice probabilities of specific assortments obtained from the GAM or MNL that may be used as the underlying model. To emphasise, only the choice probabilities of specific assortments are needed for the Markov chain choice model because no additional information of the underlying model is needed; the estimation is thus data-driven [46]. To continue, the model substitutes the customers' behaviour captured by a preference list and is interpreted as a sequential transition from one option to another [16]. In the Markov chain model, the customer enters the process where the current state represents the desired option. If that option is unavailable, the customer shifts to another option according to the Markov chain probabilities. This process continues until the preferred option is available or the customer leaves the process without purchasing. The options are captured by $j \in \mathcal{J}$ with $J + 1$ states as the non-purchase option is included. The arrival probability is indicated with v_i where i indicates the option and makes a purchase if i is available. In other words, if product i is available, the customer will select that option. Otherwise, it proceeds to a different option indicated with j, and the non-purchases option is indicated with $i = 0$. As a result, v_i can be assumed to be the probability that a customer chooses alternative i. The transition probability is represented by ρ_{ij} and indicates the probability of substituting alternative i with j given the unavailability of product i [103]. The probability of the choice options can be found in Equation 2.8, which can be estimated by expectation maximisation, and the non-purchase option can be indicated with $1 - \sum_{j \in J} \rho_{ij}$ [36][97][37].

$$\rho_{ij} = \begin{cases} 1 & \text{if } i = 0, j = 0 \\ \delta_{ij} & \text{if } i, j \in S, i \neq j \\ 0 & \text{otherwise} \end{cases} \tag{2.8}$$

where v_i is indicated with $\pi(i, S)$ and $\delta_{ij} = \frac{\pi(j,S\backslash\{i\} - \pi(j,S)}{\pi(i,S)}$ indicates the increased probability of selecting j when i is not available [16].

The above-introduced Markov chain choice model is used for assortment optimisation, single resource revenue management and network revenue management and proposed tractable solutions. However, during the modelling, the model may suffer from overfitting, especially when the training data needs to be more extensive in combination with too many parameters to be estimated [36]. The Markov chain choice model is also used to estimate parameters from choice probabilities for assortments of different sizes [46][16]. In addition, the model is also used for other purposes, such as ranking web pages in the Google Search engine [47].

To summarise, the advantages of parametric models are their simplicity, readability and the fact that they can extrapolate choice predictions to new alternatives that have not already been observed in history. Based on the extrapolation, it is possible to predict how, for example, price incentives affect customers' choices [108]. Another advantage of linear parametric models such as MNL is that the estimated weights insights allied to the characteristics can be directly obtained [79]. However, the parameters of a theory-driven parametric model are only significant when it is assumed that the model and the used theory are correctly specified since the estimated parameters are interpreted as marginal utilities [107].

2.4. Non-parametric

In contrast to parametric models, where a finite set of parameter specifications based on the data distribution is required before making predictions, non-parametric models do not rely on specific parameter settings and choose a functional form based on the training data. This means that non-parametric models make few or no assumptions about the underlying distribution of the data but estimates the function that maps the independent variables to the dependent variables using data. In addition, the required specifications for parametric models are time-consuming and sensitive to errors, which may result in inaccurate models with low predictive power, biased estimates and incorrect interpretations [84]. Non-parametric modelling can be done through, for instance, generic choice models such as distribution over preference lists or Machine Learning (ML), including supervised, unsupervised, semisupervised, and reinforcement learning. When input data is utilised to map output data during a training phase, this form is called supervised learning, and models using this will be further explained. Nevertheless, for the generic choice models, data related to the product is needed meaning it can not say useful things about an unseen product.
In contrast, this is possible for ML models [34]. Furthermore, ML creates new possibilities for developing techniques for extracting behaviourally significant, statistically reliable, and computationally efficient insights from increasing datasets [107]. When modelling customer behaviour, it is of utmost importance to understand the estimated parameters, notwithstanding that the number of parameters depends on the amount of training data in ML models [11]. Accordingly, parametric models are the most commonly used modelling technique rather than ML. This is changing since ML models now can give interpretable results from the non-parametric models rather than serving as black box [96]. There are different ways to provide insights into ML models instead of using them as a black box. One of the possibilities is using the SHapley Additive exPlanation (SHAP) value that arises from the Shapley concept. The Shapley value is the average of the marginal contributions across all permutations, but more details can be found in Appendix F how the SHAP values are obtained. The SHAP value is proposed for model interpretability by clarifying the ML model with a unified approach where the collective SHAP values can indicate how much each feature contributes to the target variable. Since all observations get a SHAP value, this facilitates greater transparency [67][62].

The mentioned ML techniques have advantages over parametric models as they are used to model non-linear relationships between characteristic values and the target group, allow collinear characteristics, have more flexibility concerning modelling and create automatic customer segments. In ML, supervised learning is used for classification with the aim that after seeing a sample set, new unseen targets can be predicted given their characteristics. The targets in choice modelling are discrete and represented by the soft classification problem. The goal is to estimate the conditional probability distribution while minimising the measure of the expected error. As it involves DCM, each alternative is treated independently and predicted whether it is chosen. Different ML models are proposed to analyse customer behaviour as several factors directly or indirectly affect purchase behaviour. Besides, customers do not follow predefined rules before making a decision. Therefore, a pattern must be identified to indicate the most probable service choices [44]. Fortunately, choice modelling can be seen as a classification task since choices can be cast as mutually exclusive classes [107].

2.4.1. Random Forest

The Random Forest (RF) method is a collection of independent decision trees that are relatively quickly trained, as no hyperparameter tuning is needed to obtain good performance. Only the node size, the number of trees, and the number of feature samples must be set before training [55]. Each tree is modified and grows throughout the training, using a random subset of the training sample consisting of feature target pairs. A binary test on one variable is performed to allocate the data over child nodes at each internal node. During the training, the classifier selects the most segregated features, and each node is locally optimised with a binary test utilising the random subset of features. A tree is fully grown when there is only one sample point in a leaf, or the maximum depth of the minimum point in a leaf is achieved. At a leaf node, the empirical conditional distribution of a target is calculated given the features that lead to the leaf. After the training, the unseen data set can be used in the prediction phase. The alternative targets are propagated through all trees during the prediction and tested by binary tests. When the target belongs to a tree, the conditional probability estimate is obtained from the belonging leaf. The final conditional probability estimate is acquired from the averages across the trees. An overview of this method can be found in Figure 2.3 [64]. This figure indicates a random forest consisting of a set of trees indicated with blue circles. When a new feature is tested, each tree generates a prediction result based on the path from the root to the leaf consisting of a series of decisions. This path is indicated with green circles and is called the prediction path and contributes to the final prediction of the model. Then the RF classifies the new data point based on the average of all decision tree predictions.

The RF model trains multiple trees using bootstrapping and is used during other research, for example, for deciding the travelling mode where the RF trees use all the independent variables, enabling variance reduction between correlated trees [118]. Another instance where the RF method is utilised is in the analysis to predict the travel mode choice. During this analysis, it was noticed that the RF produced the most accurate predictions compared to other parametric and non-parametric models [50]. In addition, the RF model can simultaneously partition customers into hierarchical segments and provide a measure of feature importance as some features are more relevant than others [64].

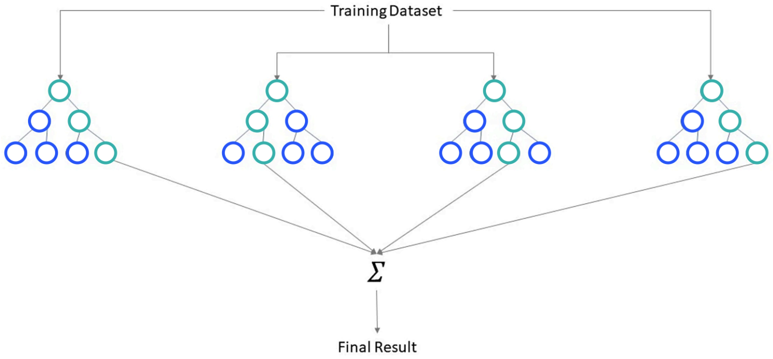


Figure 2.3: Overview of the Random Forest method [55]

2.4.2. k-Nearest Neighbour

The k-Nearest Neighbour (KNN) is a learning algorithm that compares new examples with similar examples from a training set. Based on the training set, the KNN algorithm stores an n-dimensional pattern of n attributes representing a point in the n-dimensional space. When a new example enters, the KNN compares and searches the k most similar training patterns to the unknown example, which become the k nearest neighbours of that example. The nearness can be determined using a distance metric like, for example, Manhattan or Euclidean distance. After finding the k nearest neighbours, the most commonly occurring classification for these k examples is chosen [41]. An overview of this process can be found in Figure 2.4. This approach of a supervised learning algorithm is among the simplest in ML and is widely studied in the pattern recognition field and classification projects as a predictive Benchmark. The algorithm is assumed to be simple as it just stores the labelled training pairs and is therefore indicated as a lazy learning algorithm[1]. Despite the model's simplicity, it can estimate the conditional probability that a given point pertains to a given class and the marginal probability of a feature under certain assumptions. Regardless, the KNN does not explicitly try to model the data generating process (DGP) [92][54].

[1] A lazy learning algorithm postpones the sample data processing until predictions are made.

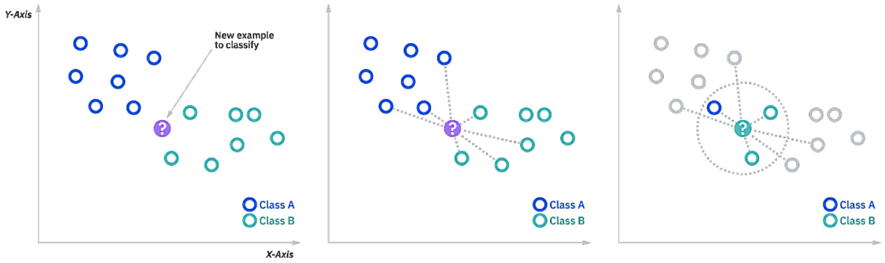


Figure 2.4: Overview of KNN classification process [54].

The KNN algorithm is, for instance, used to maximise the efficiency forecast demand in supply chain management [41]. In addition, KNN is also used to predict customers that are expected to churn based on the closeness of its features to customers in each class [94].

2.4.3. Support Vector Machine

A Support Vector Machine (SVM) can learn classification patterns with balanced accuracy and reproducibility with a segmentation function. This function is called a hyperplane and can separate (unseen) data based on features' patterns. The SVM function needs to be trained to identify a reproducible hyperplane that maximises the margin between segments. The SVM model can be linear or non-linear when it is linear; the segments can be divided with a two-dimensional hyperplane. However, if the partition can be based on a two-dimensional plane also depends on the complexity of the model's features. There are two types of maximisation of the margin, a hard margin and a soft margin. The hard margin is the simplest and the least computationally expensive, as no training errors are permitted.

In contrast, the soft margin permits the misclassification of outliers from the training data and introduces a variable to incur a penalty. Similar to other models, the SVM needs to balance two complementary aims, on the one hand optimising the accuracy and on the other hand optimising the reproducibility. To give a conceptual understanding of the SVM: the hyperplane maximises the margin between the different classes and will be identified during the training [89]. The SVM was initially designed for binary classification as the hyperplane divides two classes and is called a binary classifier. An example of a binary classifier can be found in Figure 2.5. However, when multiclass classification problems exist, one proposes combining multiple binary classifiers, whereas others consider all data in one optimisation formulation. With both approaches, either several binary classifiers must be constructed, or a larger optimisation problem is needed, but in either case, it is more computationally expensive [53]. When multiple binary classifiers are used, the multiclass problem is broken down into multiple binary classification cases. This process is called one-vs-one [73]. The number of binary classes in a multiclass problem to differentiate all possible pairs can be identified with $K(K-1)/2$, and the class with the most votes is selected for prediction. The kernel needs to be specified to let the SVM work properly, and when the kernel is non-linear, the SVM is very sensitive to overfitting [118].

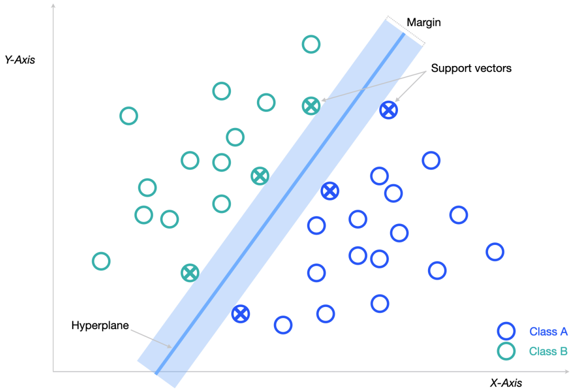


Figure 2.5: Overview of SVM

For example, the SVM is used to predict Dutch customers' travel mode and is compared with six other ML classifiers based on accuracy and sensitivity. During this analysis, the SVM model was not the best predictor, and in contrast to the other models, all the used variables had substantial importance in the modelling [50]. The model is also used for demand prediction in the retail sector and seems to be very powerful and effective in solving forecasting problems with a small sample, high dimension, and local minima that are non-linear [117].

2.4.4. Neural Networks

The biological brain inspires Neural Networks (NN), which are increasingly used for analysing customers' behaviour as it leads to improved performance and the handling of the growing volumes and diversity of available data. Before the NN model can be used, it must be trained with a sufficiently large sample set. During the training, the data is summarised and processed by neurons and weighted by the connections to produce a network output [50]. The number of neurons that are included in the NN model depends on the complexity and the number of neurons in each layer. An overview from a possible layout can be found in Figure 2.6 to understand how NN looks like. As indicated, the model consists of an input layer, one or more hidden layers consisting of neurons, and a final layer of output neurons. The input layer represents the independent variable, such as the alternatives' attributes, customers' characteristics and contextual factors, and the output layer represents the choice probabilities of all options. Between these layers, the hidden layer connects the input and output layers, and when the model consists of four or more layers, it is referred to as a deep neural network (DNN) [8][66]. Within the layers, a node can turn active or inactive.

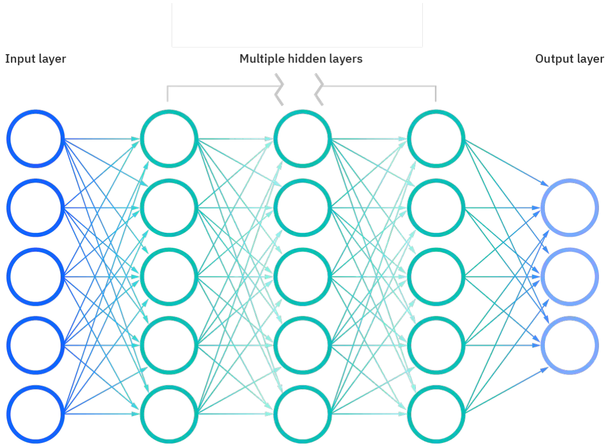


Figure 2.6: Possible overview of a NN [55].

The complexity of the model depends on the different choices made on the amount of these neurons. When the model is excessively complex compared to the underlying DGP, the NN model will fail to deliver a consistent performance after the training. Contrastingly, the model can also be too simple compared to the underlying DGP, and as a result, it will not capture the relation between the input and the observed choices. To avoid excessively complex or too simple models, the NN model is tested for various levels of complexity during the training. A training example is fed to the input layer to train the NN model, and the predicted outputs are calculated. The predicted output is then compared with the corresponding target output, after which the difference can be calculated. This difference is used for backpropagation, where the weights of the connections are fine-tuned based on the loss rate obtained during the previous iteration. Tuning the weights by iterations ensures lower loss rates, makes the model more reliable and minimises the change on misclassification [74][106].

The NN model is, for example, used to improve the parameter prediction while maintaining the interpretation relative to DCMs [96]. An NN model with multiple layers is also used for personalised content recommendations. This recommendation is highly challenging as the scale, freshness, and noise need to be considered all the time [30].

2.4.5. Bayesian Network

A Bayesian Network (BN) consists of directed acyclic graphs (DAG) that connect variables by conditional probabilities. Each node represents a random variable in the model, and the arcs display the causal relationship between the nodes. The condition is considered independent when there is no arc between the nodes. An overview of a possible DAG can be found in Figure 2.7. The construction process of a BN includes three steps: first, the variables and the range are determined. Second, the network structure and, afterwards, the local probability distribution and complete network parameter learning are determined [111]. The model's outputs are probabilities calculated with Bayes' Theorem, which gives a probability of new events depending on the information of other related events [80]. The model is advantageous for data mining and determining and explicitly displaying variables' relationships, among others. Since the outcomes of the model are generally probabilities of diverse states, the model lends well to decision-science approaches. However, as the general structure of the model is very flexible, it can be found in many new application areas. Areas where the BN model is used, are the transportation sector, where it has been applied to predicting accident situations or the cause of the accident. Thus, the BN can express the link between the influencing factors and the decision behaviour. In addition, it enables the possibility of using the BN model to understand customer choice behaviour better.

A simplified version of the BN model is the Naive Bayes (NB) model, a simple and efficient ML classifier that does not require structure learning [111]. The model is also constructed using Bayes' Theorem only has the naive assumption that all features are independent of each other [75]. Nevertheless, this assumption is in real-world assumption, sometimes not very likely, as the variables have to be completely independent. The model can, in contrast, be used as a baseline classifier for large datasets [118].

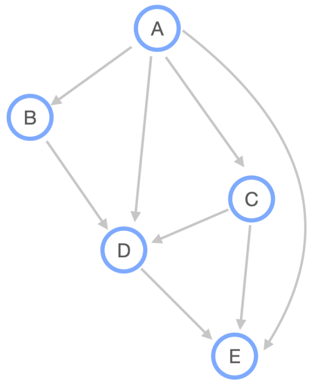


Figure 2.7: Possible overview of a DAG.

The NB model is, for example, constructed in a study that compares different parametric models with ML models by determining the travel mode [118]. To learn the behaviour of the customer choices regarding travel mode, a BN model is constructed. In this case, the BN model can capture the changes in choice behaviour if factors shifts which indicates a link between the customer behaviour and the selected features [111].

Next to the ML models, preference lists can be used to incorporate customer choice behaviour in, for instance, the optimisation of revenue by determining the availability of a particular assortment of products [34][108]. However, preference lists are not used as much as ML algorithms in customer choice behaviour in the retail sector.

To summarise, non-parametric ML models improve the model performance, specification, and estimation time compared to parametric models. However, in the first instance, they are not directly interpretable as their number of describing parameters in, for example, NN or RF model can get high. As a result, interpretation tools are developed to extract the obtained knowledge from the black box models allowing for prediction and behavioural analysis. There is a difference in how customer choice behaviour is approached to continue the comparison between the models. In parametric approaches, the models access the outcome of a choice problem as individuals select an option from a set of options to maximise their utility. On the other hand, ML models access the outcome of a choice prediction as a classification problem where the outcome is predicted based on a set of input variables [118]. ML model building does not depend on laborious and time-consuming theory. Nonetheless, numerous decisions need to be made regarding the hyperparameters settings, topology and performance function, and the model building still involves a trial-and-error approach [107].

In other literature, ML methods for classification problems are compared, and the results on binary classification tasks show that RF and NN generally perform well. In another research where high dimensional data is used, the SVM, NN, and RF methods perform properly [49]. When RF, NN and SVM methods are compared to predict purchase decisions accurately, the NN[2] model outperforms the other ML techniques when applied to the same dataset. Nevertheless, it is indicated that all techniques can support the decision-making process [28]. When almost all discussed ML methods are used to predict the same dataset, RF performs best, followed by SVM, NN[3] and NB, respectively [50]. Decision trees that are part of an RF are also found to be better interpretable than NNs [114]. In another comparison with almost all the different models, the RF performed again the best, followed by SVM, NN[3] and NB, respectively. The KNN and BN models can also be compared based on demand prediction, where it is found that the BN technique outperforms the KNN model [41].

2.5. Trend

To demonstrate the development of customer choice models, an overview of relevant literature is presented in Table 2.3. This table summarises the research conducted for this study, authors, publication year, the model type used, and the objective of each study. The objective indicates the focus of the research with, when needed, the main optimisation goal in brackets, as this can differ. It should be noted that while some papers primarily aimed to model customer behaviour, the models may have been applied for other purposes. Additionally, the area is not indicated for some models, as the technique was used for general purposes. Analysis of the publication years of the papers in Table 2.3 reveals that most papers have been publicised recently, indicating that demand management and customer behaviour in retail have gained more attention in recent years. However, a side note must be made as the obtained search was related to customer behaviour in retail and, in some cases, the transport sector. Nevertheless, looking into more detail in the found literature, it can be seen that the papers based on ML are very recent, which signals that it experienced an uptake. Furthermore, it is found that most studies initially focus on pricing and availability and later focus more on behaviour. The wide range of model types suggests no consensus exists over which ML model is best suited for choice modelling, although some studies compare different ML techniques.

The literature reveals that a significant amount of research has been conducted on how customers respond to changes in price, availability, and time slot length using primarily parametric models and, increasingly, ML techniques. When looking specifically into used models for the online grocery sector, it is found that when ML is used, simple NN models are integrated and compared with other ML models. Based on these comparisons and results from other sectors, such as e-commerce, the NN has promising results when including multiple hidden layers. However, there is limited literature on using these NNs with multiple hidden layers in customer choice models for online groceries. In addition, ML models are often employed as black boxes without offering insight into which features are crucial for predicting customer behaviour. Therefore, this study will look into the added value of a multi-layer NN in the learning process of customer behaviour and will provide insights into which customer characteristics are essential for these predictions. The most commonly used parametric, the MNL, will compare the results to indicate if the NN with multiple hidden layers adds value to the process.

[2] During this process, the NN model had multiple hidden layers.
[3] During this process, the NN model had one hidden layer.

Area	Reference	Year	Model	Objective
Retail	Mackert [68]	2019	GAM	Customer behaviour (availability)
Retail	Agatz, Fan and Stam [2]	2021	MNL	Customer behaviour (route)
Retail	Yang et al. [116]	2016	MNL	Customer behaviour (time slot)
Retail	Amorim et al. [9]	2020	MNL	Customer behaviour (utility)
Retail	Klein et al. [59]	2019	Ranking	Customer behaviour (pricing)
Retail	Mackert, Steinhardt and Klein [69]	2019	MNL	Customer behaviour (availability)
Retail	Zhu, Dou and Qiu [119]	2019	NL	Customer behaviour (choice probability)
Retail	Köhler et al. [60]	2019	NL	Customer behaviour (pricing)
Retail	Strauss, Gulpinar and Zheng [102]	2021	MNL	Customer behaviour (pricing)
Retail	Asdemir, Jacob and Krishnan [10]	2009	MNL	Customer behaviour (pricing)
Retail	Agatz, Fan and Stam [1]	2020	MNL	Customer behaviour (utility)
Retail	van Ryzin and Vulcano [108]	2015	Ranking	Learn a non-parametric choice model
Retail	Yang and Strauss [115]	2017	MNL	Customer behaviour (pricing)
Travel	Farias, Jagabathula and Shah [34]	2013	Ranking	Customer behaviour (availability)
Airline	Garrow and Koppelman [39]	2004	MNL, NL	Customer behaviour
Airline	Blanchet, Gallego and Goyal [16]	2016	Markov	Customer behaviour (availability)
	Feldman and Topaloglu [36]	2017	Markov	Customer behaviour (availability)
	Simsek and Topaloglu [97]	2018	Markov	Customer behaviour (choice probabilities)
	Gupta and Hsu [46]	2020	Markov	Customer behaviour (choice probabilities)
Retail	Chaudhuri et al. [28]	2021	NN, DT[4], RF, SVM	Customer behaviour
Retail	van der Hagen et al. [49]	2022	RF, NN, GB[5]	Time slot management in AHD
Retail	Gaur, Goel and Jain [41]	2015	KNN, BN	Demand prediction
Retail	Yue et al. [117]	2007	SVM	Demand prediction
Travel	Sifringer, Lurkin and Alahi [96]	2018	NN	Customer behaviour (improve predictability)
Travel	Hagenauer and Helbich [50]	2017	NB, SVM, NN, BOOST[6], BAG[7], RF	Travel demand
Travel	Zhao et al. [118]	2018	MNL, NB, RF, SVM, NN, BOOST, BAG	Customer behaviour (comparison)
Travel	Wang, Sun and Zhang [111]	2017	BN	Customer behaviour
Airline	Lhéritier et al. [64]	2019	RF	Customer behaviour
Entertainment	Covington, Adams and Sargin [30]	2016	NN	Customer behaviour (recommendation)
Telecom	Sabbeh [94]	2018	RF, SVM, DT, NB, KNN, LR, DT, Ada Boosting SGB	Customer behaviour (comparison)
Cloud service	Ghosh and Banerjee [44]	2020	RF	Customer behaviour (recommendation)

Table 2.3: Overview of found methods to model customer behaviour with between brackets the underlying purpose for what aim the model is used.

2.6. Summary

To encourage demand management which can maximise the overall profit as it intends to shape and generate customer demand, more advanced technologies are introduced to understand customer behaviour better. Thus, the trade-offs between generated demand volume and fulfilment efficiency are managed and hence between interrelated cost and revenue effects. In the fulfilment process, the most relevant part is the order decoupling point consisting of order capture, assembly and delivery. In order capture, the consumer and company agree on when and where the order will be delivered. This agreement arises after the company shows different time slots of which one will be selected if the customers' preferences and expectations are met. When a timeslot is selected and the order is placed, the assembly will be scheduled to prepare for delivery. The delivery of the groceries will then occur in the predetermined time window. How efficiently these processes are conducted depends on the optimisation process used and the preferences of individual orders. As efficiency relies on customer preferences, small price incentives, for example, can control the demand as they will influence customer choice behaviour.

Demand management indicates two primary levels to influence the demand: the length and corresponding price of time windows, and both can be determined as dynamic or static. With these two mechanisms, the capacity in the warehouse, physical fleet size and available driving time must also be guaranteed. Demand management aims to exploit market heterogeneity, partitioning the market into segments with various sensitivities and preferences to ensure the capacity as it is not sold with a first-come-first-server mentality since it may be more profitable to reserve scarce capacity for the most profitable customers. The segments emerge based on heterogeneity and can be partitioned based on customer characteristics. Getting the benefits of demand management requires a good understanding of both cost dependence and customer behaviour.

[4] Decision tree

[5] Gradient boosted trees

[6] Boosting trees

[7] Bagging trees

For describing the customers' behaviour of selecting an option from a set of alternatives, choice models are increasingly being employed. The models can identify the behaviour, and a company can take advantage of that to make efficient pricing and management decisions. The choice probabilities calculated with the choice models can be for individuals or customer segments. In either case, the number of parameters required needs to be determined, and a trade-off between parametric and non-parametric models must be made. Customer segments are based on customer choice characteristics and clustering techniques, as the requirements and needs of a segment are assumed to be the same. When the requirements and needs of either a segment or an individual are known, it may be possible to predict future choices and related costs. This can be done with the models above; however, the utility-based parametric MNL model is the most commonly used one.

In parametric models, all the data is summarised through a fixed-size collection of manually chosen parameters that maximise the utility as customers are assumed to associate a particular utility with every product. Each decision is based on maximising utility. Based on the model, predictions of future choices and new unobserved alternatives can be made. In addition, the advantage of parametric models is their simplicity, readability, and the estimated weights insights allied to the characteristics can be directly obtained. However, the model is only significant when the model and the used theory are correctly specified. Furthermore, the manually determined parameters are time-consuming, and some parametric models, such as MNL, encounter IIA, leading to overestimation. To dispel IIA, multiple nests can be introduced, resulting in the NL model where IIA holds within a nest and not across nests. Subsequently, the GAM model is addressed where the possibility that an option from a different company is chosen is included. Nevertheless, the GAM is a limited case of the NL model where the offerings within each nest are perfectly correlated, and the NL model is again a limited case of the MNL. The MNL and GAM may also be used as the underlying models for the Markov chain choice model that substitutes the customers' behaviour caught by a preference list and interpreted as a sequential transition. However, the other models are more commonly used to determine customer behaviour in AHD due to overfitting by more extensive training data in combination with too many parameters to be estimated.

In non-parametric models, the functional form does not rely on specific parameter settings but requires a finite set of parameters. Resulting in a more accurate model with higher predictive power, less biased estimates and more correct interpretations. However, non-parametric models like ML can be perceived as black boxes since the results are not directly interpretable. Different techniques are developed to better understand the obtained results in response to the not directly interpretable variables when utilising ML. On the other hand, ML techniques also bring other benefits, such as capturing non-linear relationships between characteristics, automatically creating customer segments and being more flexible. The RF method is an ML technique that is quickly trained and used to create customer segments and determine customer behaviour. Different classification techniques of ML are KNN and SVM. KNN is an algorithm that compares new examples with previous examples and calculates the distance to all previous data points to predict the majority label of all the closest points. However, this method is straightforward but has high prediction costs, which is critical with large data sets. The SVM technique analyses data recognises features' patterns and divides the data into categories using a hyperplane. It is possible to have multiple categories in the SVM model, but it makes the model computationally expensive. Especially when new data enters the model, the algorithm assigns it to a category. For a multiclass problem, it breaks the problem down into multiple binary classification cases, making it more computationally expensive than for a single binary problem. The NN and the BN techniques are two ML methods based on nodes. The NN is inspired by the biological brain and is increasingly used to analyse customers' behaviours. It leads to improved performances as it can handle the growing data volumes and diversity of data. However, the NN must be trained with a sufficiently large sample set. The BN uses Bayes' Theorem to calculate the probability of new data depending on past data information. The general structure of the BN model is very flexible; the model can be used in different application areas. The ML algorithms can be compared, and when done with the same data set, it is emerging that RF and the NN[8] perform well and even outperform the other models.

[8] with multiple layers

3

Proposed methodology

The above-mentioned parametric and non-parametric models can be used to describe customer choices based on their behaviour to gain more insights into the selection process of timeslots while doing online groceries. In other words, the purpose of choice models is that they can be used to determine the attractiveness and satisfaction given by a specific time slot. By doing so, demand can be managed to keep track of AHD's profitability. Once the customers' behaviour is known, demand management can influence and nudge their behaviour to cost-efficient time windows with incentives, for example. However, the market is increasing as more customers and companies enter the highly competitive industry. This growth encourages customer segmentation to ensure attractiveness by offering them the right time slots in combination with incentives to provide competitive advantages and long-term success for the company. Customer segmentation will be done carefully in this study, as it is assumed that the behaviour of a segment is homogeneous and therefore has a significant impact on the model performance. Not only is the customer demand rising likewise is the amount of accessible customer information, with the result that not all characteristics are essential to model behaviour. Therefore a selection must be identified with which features most influence customers' behaviour. However, several techniques can be used for segmentation and feature selection; also, there are different incentive types. However, for this research, only the incentives that the company uses will be taken into account. In addition to segmentation and feature selection, when looking more into detail in both model specifications, it can be found that several settings can change, affecting performance; therefore, hyperparameter tuning will be used to select the most optimal combination of settings. An example of a setting that can be used in both models is balanced class weight. Class weights can be used when the data set is not balanced, meaning that a particular class is underrepresented, which might lead to poor performance of models that assume a balanced class distribution [20]. The performance is affected as the misclassification costs are unequal, leading to ignoring the minority class and favouring the majority class [33]. The occurrence of each label can be checked, and the share can be determined to avoid favouring the majority class. However, actual data is needed to determine the class weights and will therefore be performed after the data gathering. Another specification to be decided on is which solver or activation function the models will use; for both models, multiple options are available. For the NN model, more specifications are needed, such as the number of hidden layers and how many neurons each layer must consist of for the best performance. Executing the modes can help to make these choices as the performance can be compared. All the tests are done for the same test and training data, divided into three ways using Sklearn's train test split function. Through the division of the data into three distinct sets, the impact of various options on the model's performance is evaluated using representative data from the entire dataset. Consequently, it is possible to eliminate the possibility that the model's performance is influenced by the method used to divide the dataset.

3.1. Customer segmentation

To start with customer segmentation, where customers are divided among different clusters, as with the growing variety of preferences and tastes, companies cannot fully satisfy all customers. The definition of a segment is an important design decision as the model assumes that the choice behaviour of all customers within a particular segment is homogeneous [116]. This means that each customer in the segment will rank the time slots in the same order [48][69]. Nevertheless, it is assumed that between segments, the preferences of, for example, time slot alternatives are perceived differently [59]. The segmentation of customers encourages companies to identify valuable customers and retain these, as this is crucial for the company's success in the very competitive industry and is increasingly being used to understand the characteristics and needs better. When customer groups are identified, the potential profit can be identified, and knowledge of the group can be acquired for service according to specific needs and preferences. The segmentation can be based on general and product-specific variables. General variables incorporate demographics and lifestyles[1], and product-specific variables take the purchasing behaviour and intentions[2] of customers into account [88] [105].

Hence, the segments for online groceries can be created based on clustering different characteristics such as delivery day, zip code or average order value. When the segmentation is based on the delivery day, it is assumed that every customer considers all time slots on that delivery day and allows a request to be part of exactly one segment. When time preferences are included, two customers ordering simultaneously for the same zip code area and the same delivery may have different time slots offered with different prices as their estimated choice preferences are likely to deviate. As the preferences impact the price optimisation when dynamic pricing is used, this may result in price discrimination based on characterises and is regarded as unacceptable by many customers. Accordingly, companies possibly reject models that offer different prices in the same time slot for the same location and order size [116]. To implement a more refined segmentation, behavioural data concerning what customer purchases are needed, such as the type of preferred products, total expenses, ordering frequency or the reaction to sales promotions. In addition, general data is considered private property and, therefore, hard to obtain and time-consuming [105].

Since clustering is an essential and comprehensive tool used for customer segmentation, many different techniques can be divided into two major groups: hierarchical and partitional clustering. Hierarchical clustering finds nested clusters, and on the other hand, partitional clusters consist of non-overlapping clusters [35]. However, different algorithms can be used to determine the clusters. One widely used technique is the k-means algorithm. Which needs the number of clusters prior, or the segments can be determined based on, for instance, artificial intelligence algorithms, preference lists, marketing surveys or on judgement and market knowledge [18][88][108][59]. The k-means algorithm is, for instance, used to cluster AHD customers along four dimensions: basket value, basket size, the share of perishables items, and share of discount value [9]. Alternatively, a decision tree can handle the categorical variables in the dataset [28]. On the other hand, customer segmentation was done for years based on recency, frequency and monetary value (RFM) indicators. The RFM is a simple model which proved its place, and even with more sophisticated models, people continued using the RFM. However, the RFM was still too complex and time-consuming in some cases, and a simplified, more practical version was needed. As a result, the customer value matrix with five segments was introduced and is in the marketing industry commonly used. The five segments are Best, Spender, Average, Frequent and Uncertain [72][65]. Based on the customer value matrix, loyal customers' behaviour can be considered more important to satisfy than the behaviour and satisfaction of incidental customers. Moreover, customer engagement with a company provides a competitive advantage and drives it to long-term success [28]. In this study, customer segmentation is achieved through the assignment of a feature value based on demographic and historical characteristics.

[1] e.g. age, sex, income, education level etc.

[2] e.g. purchase frequency, spending, consumption etc.

3.2. Feature selection

Before the customer choice models can be applied, a selection of characteristics has to be made, especially with the increasing amount of customer data where not all variables have the same relevance. Specifying the most affecting characteristics influencing the model in parametric models is essential before it can be used. However, when the most affecting characteristics are identified, the model dimensionality can be reduced as multiple variables with low influence will be removed [94]. Nevertheless, the manual selection processes of these variables that are regarded as essential are laborious and prone to errors. Particularly with the availability of larger and larger datasets and the increasing number of possibilities, the specifications grow beyond manageable. The feature selection is based on the models' calibration on the entire dataset in combination with forecasting performance measures. These measures penalise the model performance for including too many "useless" features. ML or other data-driven models can be used for this process for more convenience as they are more flexible and can directly learn from the data [84][118]. There are different ML techniques, which can be divided into two categories: supervised and unsupervised. The supervised technique is used for labelled data, while the unsupervised technique can be used for unlabeled data. To continue grouping, the classification techniques can also be categorised:

- Wrapper methods
- Filter methods
- Embedded methods
- Hybrid methods

At wrapper methods, ML algorithms are greedy and try to fit into the given dataset, search the space of all possible subsets of features, and evaluate the subset against the evaluation criterion. The second classified technique is filtering, where the intrinsic properties are measured with univariate statistics. These methods are less computationally expensive than the previously mentioned wrapper methods. The embedded methods are acquired when combining the advantages of the wrapper and filter methods. This iterative technique carefully extracts the features that contribute most of the training but maintains a reasonable computational cost. When combining the different techniques, hybrid methods arises. These hybrid techniques are combinations of filtering and wrapping techniques [45][19]. An example of the embedded method is the RF algorithm. This model uses the mean decrease accuracy that measures the impact of the individual features on the model accuracy. Next to the RF model, the Boruta selection technique can be utilised. This technique considers all characteristics relevant to the target variable and can handle interactions between characteristics [94].

When no ML is used for the feature selection, the selection can, for instance, be made based on a survey. With the survey, the features are selected, and discretisation will occur afterwards. This process is, for example, used when residents' behaviour is estimated with a BN model. The overall features are obtained via the survey. The importance of each characteristic is determined based on the mutual information calculation between the travel mode and the variables [111]. In a customer behaviour analysis in AHD estimated with MNL, the included features were: the set of offered and selected time slots with the corresponding price, the time to deliver, the slot size, and the delivery day and time [9].

Based on the above-described feature selection methods, it is chosen to combine two classification techniques in this study. First, the filter method will determine the correlation between the features and the output variable and between the features themselves. A variable will be incorporated into the feature set when it has a high correlation with the output and a low correlation with the other variables. The correlation of all variables will be tested using a correlation matrix. This filtering step is pre-processing and will be done before executing the models; therefore, it does not depend on any ML algorithm. The advantage of using the filter method as a pre-processing step is that it removes redundant data reducing dimensions and computational time. When the filer method is performed, the MNL and NN can be used where the embedded method will be executed since this method is integrated into both the classifiers. The embedded method determines and assigns weights for all features during training to produce the best classification performance. The weights assigned to the features give an understanding of the data and help with the model interpretability allowing for more insights.

3.3. Incentives

The purpose of demand management is to influence and nudge customers to time windows that are cost-efficient for the company. This can be done when the customer behaviour is modelled, and the segmentation and features are indicated. After this, the effect of incentives can be demonstrated. As mentioned earlier, these incentives influence customers' choice of delivery windows to improve routing efficiency. Moreover, time slot allocation can be better managed, allowing more consumers to be served, maximising total profits and lowering fuel consumption and emissions. Despite the usage of price incentives, it is assumed that a customer's behaviour and likelihood of selecting a particular time slot is known as they are only used for feasible time slots with a positive chance of being selected. The needed knowledge can be obtained through historical data on customer behaviour.

There are various incentives, including price and green labels. However, it is found that providing incentives only for a few time slots is sufficient. Furthermore, there are also different pricing methods; for example, a price incentive can be given to the slot with the cheapest insertion costs, which maximises the expected profits or the one the company prefers [24]. Besides the different methods, both the insertion price and time can differ. This means it is not established at which price and at what point in the booking horizon the incentive is included [83]. If the price is unexpectedly changed, the customers might see it as unfair. However, on the other hand, when the implementation pattern is regular, it is possible that the customers' behaviour changes as they learn to anticipate resulting in a limited effect of the price incentives. The described price incentives are based on extrinsic motivation, and the mentioned green labels have intrinsic motivation by customers. Green labels effectively influence customer behaviour and steer them to more eco-friendly delivery options, especially for more eco-conscious people impact is noticeable.

When the usage of green labels is combined with price incentives, it is found that combining them for the same slot is not beneficial. Regardless, it is found that using green labels is more effective in steering the time slot choice of customers than with price incentives [1]. However, it is of significant importance that using customer behaviour for incentives does not become part of price discrimination based on characteristics. In addition, the size of the incentives must ensure not to eliminate the already thin margin as already small discounts can affect profitability.

However, in this study, the used company offers all time slots for the same price but decides which time slots to offer and which not to offer. In addition, all the time slots have the same length and do not use green labels to indicate the greenest delivery moment. Therefore, the identified customer behaviour will not include knowledge about these types of incentives. Nonetheless, to apply incentives properly, knowledge about the probability of choosing a time slot is needed, which will be obtained in this study. If advanced models prove more effective in determining customer behaviour, data from another company containing incentives will be used to assess the effect of incentives during a simulation.

3.4. Solvers and activation functions

Since solving both MNL and NN models are optimisation problems, different activation functions and solvers can be used. Unfortunately, none works best for all optimisation problems as it depends on various factors, such as the data set and the model's architecture; therefore, the model must be performed multiple times with different functions to select the best option for that particular prediction. Several solvers and activation functions are commonly used for both models to minimise the cost function. To start with, the MNL model where the commonly used solvers that handle multinomial losses are:

- newton-cg
- lbfgs
- sag
- saga.

The newton-cg is a newton method that uses an exact Hessian matrix[3]. The lbfgs solver is an abbrevi-

[3] Hessian matrix is a squared matrix of second-order partial derivatives of a function.

ation for the Limited-memory Broyden-Fletcher-Goldfarb-Shanno solver and approximates the second derivative matrix with gradient evaluations. To save memory, it only stores the last few updates. The third solver is the Stochastic Average Gradient descent, abbreviated sag solver and is a variation of gradient descent and incremental aggregated gradient approaches. For these approaches, a random sample of previous gradient values is used. The last solver, the saga, is an extension of the sag solver and allows for L1 regularisation[4] to prevent the model from overfitting by penalising the magnitude of the coefficients of the model's parameters. The L1 regularisation term is added to the model's cost function minimised by the solver, which encourages the model to have sparse parameter estimates [51][99]. The four solvers will be performed, and the best-performing one will be selected to identify customer behaviour during this study.

For NN models, different functions are used for MNL models, and the NN model differentiates activation functions between the function in hidden layers and the output layer. The output layer for multiclass classification problems has the Softmax function as the most common activation function with N output units. Softmax is used as it normalises the output of each unit to a probability distribution over N classes, where N represents the number of output categories. Since multiple activation functions exist for the hidden layers, the frequently used functions that will be compared in this study to find the most optimal configuration of the NN are:

- ReLU
- ELU
- Tanh
- Sigmoid

The Rectified Linear Unit (ReLU) is the most commonly used activation function for hidden layers due to its simplicity, computational efficiency, and as it can help to learn no-linear relationships. Secondly, ELU will be tested, which is an abbreviation of Exponential Linear Unit and avoids the problem that a ReLu function can "die". ReLu is assumed to be "dying" when the output of a large fraction of neurons becomes inactive with a zero output as a result of negative values. The ELU function approaches this differently by setting negative values with a smooth transition to zero instead of making all variables zero. The third activation function in this research is the Tanh which stands for Hyperbolic Tangent and is a sigmoidal function that returns values between minus one and one. It is a popular activation function for hidden layers in a NN due to its symmetry around zero and its ability to model non-linear relationships in the data. The last compared activation function is Sigmoid, a non-linear activation function that maps any input value to a value between zero and one. However, both Sigmoid and Tanh suffer from the problem of vanishing gradients, making the training of the NN difficult [31][14][21]. To give a complete overview of which activation function performs best, the NN models will be executed with different hidden layers and various numbers of neurons in these layers. The number of hidden layers that will be used is 2, 3, 4, and 5, and the number of neurons in the hidden layers will equal 30, 50, 70 or 90.

3.5. Performance measures

The model performance of the above-described activation functions and selected features can be evaluated with well-known criteria for testing prediction model accuracy [49]. The criteria used include accuracy, precision, recall and the F1-score and are calculated based on a confusion matrix. Where the confusion matrix consists of four terms:

- *True positives (TP)* - The number of true positives indicates the number of correctly predicted positive customers.
- *True negatives (TN)* - The number of true negatives indicates the number of correctly negative specified customers.

[4] L1 regularisation, also referred to as Lasso regularisation, is a method utilised in ML to mitigate overfitting and decrease the complexity of a model. This is achieved by including a penalty term in the model's loss function, which is directly proportional to the absolute values of the model's weights. The aim of this penalty is to force the model to reduce the coefficients of less important features to zero, resulting in a sparse model that retains only the most significant features.

- *False positives (FP)* - The number of customers that the prediction algorithm incorrectly specified as positive.
- *False negative (FN)* - The number of customers that the prediction algorithm incorrectly specified as negative.

Based on these four terms, a confusion matrix can be derived to evaluate the performance of a classification model. Nevertheless, it will not be used to compare different models but to give a general feeling of the performance. A confusion matrix is an NxN table, where N is the number of unique output labels, and the axis represents either the predicted label and the other the actual label. The confusion matrix can provide a general performance feeling since, in addition to the number of correctly classified cases, it also indicates which category it should have been when incorrectly predicted.

Another evaluation metric is accuracy, as this measures the number of correct predictions made by the model concerning the total number of predictions made. The accuracy can be determined using Equation 3.1 and is only valid when the predictions are based on a balanced data set; otherwise, the results can sound great, whereas the model performs poorly.

$$\textit{Accuracy} = \frac{\text{TP + TN}}{\text{TP + FP + TN + FN}} \tag{3.1}$$

In the case of imbalanced datasets, comparing executed models using the traditional evaluation metrics may only provide a general indication of performance. The F1-score can be used as an alternative evaluation metric in such situations. The F1-score considers class imbalance by taking the average of two standard metrics: precision and recall. The formula for calculating the F1-score is shown in Equation 3.2. Prior to determining the F1-score, precision and recall must be calculated using Equation 3.3 and Equation 3.4, respectively.

$$\textit{F1-score} = 2 \text{ x } \frac{\textit{Precision} \text{ x } \textit{Recall}}{\textit{Precision} + \textit{Recall}} \tag{3.2}$$

$$\textit{Precision} = \frac{\text{TP}}{\text{TP + FP}} \tag{3.3}$$

$$\textit{Recall} = \frac{\text{TP}}{\text{TP + FN}} \tag{3.4}$$

3.6. Benchmark model

The proposed methodology aims to estimate the time window a customer is likely to select using an MNL and NN with the best-performing architecture and feature set. To achieve this goal, three separate models will be employed to predict the delivery week, day, and part of the day to enhance the identification of customer behaviour. Three models are used because attempting to predict the delivery moment simultaneously leads to a rapid decline in prediction performance. This decrease in performance can be attributed to the fact that the number of output classes becomes too large when using classification models.

Benchmark models are developed to compare the prediction performance to determine whether the more advanced MNL and NN models are needed. Hence confusion matrices are constructed for these three models and can be found in Figure 3.1 and Figure 3.2, summarising predictions made based on assumptions found in the data. An assumption made is that the model selects the option that is most commonly chosen for that ordering moment. The results indicate that most orders are placed for delivery within the same week and in the morning. However, when creating the Benchmark model for predicting the delivery day, it is found that all orders are assumed to be lost based on the data set. In reality, this is not realistic, but due to the large number of orders considered lost, this option is the most common for all days. To create a more realistic Benchmark, it is assumed that orders are lost two-thirds of the time and that for the remaining one-third of the time, the order is placed for the most common delivery day when assuming that all orders are executed. An overview of the most commonly selected delivery moment for all three Benchmark models can be found in Table B.1.

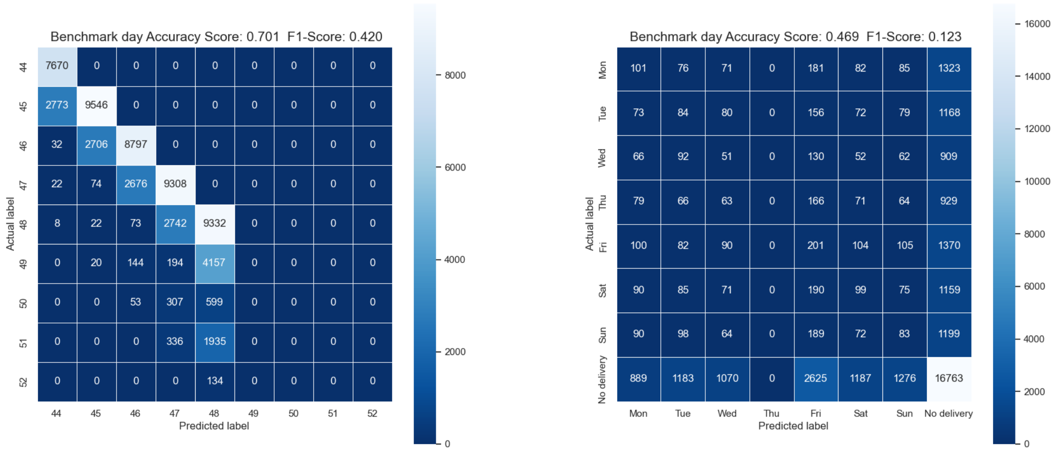


Figure 3.1: Confusion matrix of Benchmark models for predicting the week and delivery day.

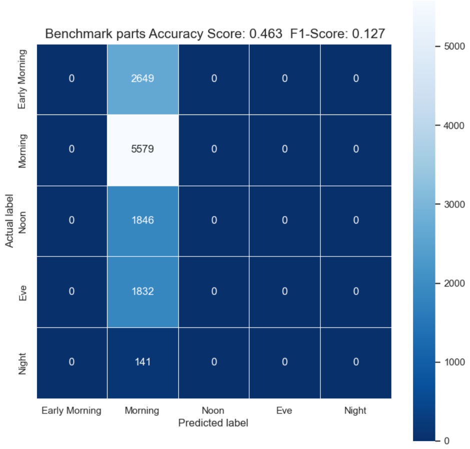


Figure 3.2: Confusion matrix of Benchmark model for predicting the part of the delivery day.

Based on the left confusion matrix in Figure 3.1, it can be found that no deliveries are predicted for weeks $49, 50, 51,$ and 52. To investigate why these weeks are not predicted in the Benchmark model, the spread of occurrence of the weeks has been looked at since it is assumed that the data set is imbalanced. In this distribution, it is found that the previous four mentioned weeks do not often appear, which makes sense why they are not the most commonly selected delivery weeks. Nevertheless, based on this data, the Benchmark model has an F1-score of $0,420$, which can be used for comparison and an accuarcy of $70,1\%$. On the right side of Figure 3.1, it can be found that the Benchmark model for predicting the day of the week, established on the assumption mentioned earlier, has an accuracy of $46,9\%$ and an F1-Score of $0,123$. However, in the results of predicting the day, it is found that there are no deliveries predicted for Thursday because more customers who requested on Wednesday selected Friday as the delivery day instead of the next day, Thursday. Therefore, when using the Benchmark model, there are no predictions for deliveries on Thursday. When using the Benchmark model for predicting the part of the delivery day, it is found that the model predicts that all customers chose a time slot in the morning. This is the most common output in the data set, so it immediately indicates consumers' preferred delivery time. Even though this is not feasible, the model's accuracy is $0,463$, meaning it correctly predicted $46,3\%$ of the delivery day parts. By which it could be assumed that the model is working. However, when checking the F1-score, it can be found that the model is not a good prediction model since the score is only $0,127$. Therefore the Benchmark model for predicting the day is a good illustration of why not to use accuracy to compare the model's performance in combination with imbalanced data.

3.7. Summary

Based on the available data and the increasing number of customers, decisions must be made about which customer characteristics help determine behaviour and, if needed, how to divide the large group of customers. The segmentation is used to cluster the growing variety of preferences and tastes, as companies cannot fully please all customers. Therefore, it might be crucial for the company's success to identify the most valuable customers. The segmentation of customers can be done based on general or product-specific variables and with various techniques. However, companies may reject models that may perform price discrimination and mostly implement more refined segmentation. Another essential part of customer choice modelling is feature selection, especially with the increasing amount of available customer data. Therefore, the most critical characterises that mainly influence customer behaviours must be specified. It is possible to make feature selection manually as in parametric models; however, this is very time-consuming and prone to errors. As a result, it can be done better and faster by using supervised or unsupervised ML learning techniques, which penalise the model performance for including unnecessary features. Based on this collected information, feature selection in this study will be made in two steps. First, a correlation matrix is created to extract features to reduce the dimensions and computation time. The used correlation matrix is part of the filtering method. Afterwards, the embedded method will be used. This method will provide a weight value to all features based on their importance and is embedded in the models.

After the feature selection, the possible architecture of both MNL and NN models are discussed as the class weights, activation function/solvers, and hyperparameters can significantly impact the models' performances. Whether class weights are needed depends on if the data set is imbalanced and is tested after the data is gathered. However, class weights can add value to the models' performance when utilising an imbalanced data set in models that assume a balanced class distribution since this may lead to ignoring the minority class due to unequal misclassification costs. And the unequal misclassification costs can result in even worse performances. If a data set is assumed to be imbalanced depends on the time each class is represented compared to the occurrence of the other classes. In addition, even if the data set is assumed to be imbalanced, the models' performance of the data with and without class weights will be compared to ensure that they add value in improving the model. Another model characteristic that can affect the model performance is the solver or activation function used in both optimisation models, as no one works best for all models. There is no common best solver or activation function for all models since various factors, such as the model architecture, impact the performance. However, some are more frequently used than others, and these more popular ones are employed to optimise performance.

In this study, the performance of four solvers used for the MNL model, each with its advantages, will be evaluated. Different activation functions will also be explored for the MNL and NN models. Specifically, the activation function used in the hidden and output layers of the NN model will be investigated. While the Softmax activation function is commonly used in the output layer of multiclass classification problems to normalise the output into a probability distribution over all classes, the performances of four activation functions in the hidden layers will also be compared. Furthermore, since the optimal activation function depends on the model's architecture, the study will test various combinations of hidden layers and neurons in combination with the solvers to identify the most optimal one.

Because there are multiple model configurations, the performance of the models can be evaluated with well-known criteria. The criteria used in this study are accuracy and the F1-score and are based on a confusion matrix consisting of TP, TN, FP and FN terms. Here the confusion matrix is predominantly used to get insights through a general feeling. Next to the insights from the confusion matrix, the accuracy can only be used to compare the performance if imbalanced data do not distort the measurement.

Nevertheless, the F1-score is not dependent on the data distribution and will be used to compare all the performances. Also, the F1-score will be used to determine whether or not to use the balanced class weights and which solver model architecture works best.

When the features and model outlines are selected based on evaluating the performance, customer behaviour can be modelled, and demand management can be applied. Demand management aims to influence and nudge customers to alternatives that make a company more efficient. When customers are influenced, allocation can be better managed, resulting in more customers being served more efficiently. Different incentives, such as prices and green labels, can be used to nudge customers. However, sudden price changes, for example, can be seen as unfair by customers, but on the other hand, when the pattern is regular, they may learn to anticipate, which limits the effect. The effectiveness of incentives on customer behaviour can also be influenced by whether it is the customer’s intrinsic or extrinsic motivation. Nevertheless, first, customer behaviour without incentives will be considered after establishing a feature set and the best configuration of the MNL and NN models. To identify if the MNL and NN add value to the process of learning customer behaviour, a Benchmark model is introduced and will be used to compare the performance in addition to only comparing the MNL and NN with each other.

4 Data

In this section, the time slot data are described. The data are acquired from an online grocery store and comprise the provided offer sets and the selected and modified time slots with estimated and actual order sizes. When the customer enters the ordering process, the company estimates the value of chilled, frozen and crates as an indication of the expected order size. The estimation is of interest as the customer selects the products after choosing the preferred time slot, as described in section 2.1. The actual order size will subsequently be used to check the availability when changes are made to the time slot or the grocery selection to check the capacity, as both are limited. In Figure 4.1, an overview of the order process can be found where the points illustrate the steps in the operation process. The points with prints indicate the steps in the process for which data are available. The red line divides the process into two parts, with on the left side the data points used in this research to predict the selected time slot and provide more insights into the customers' behaviour.

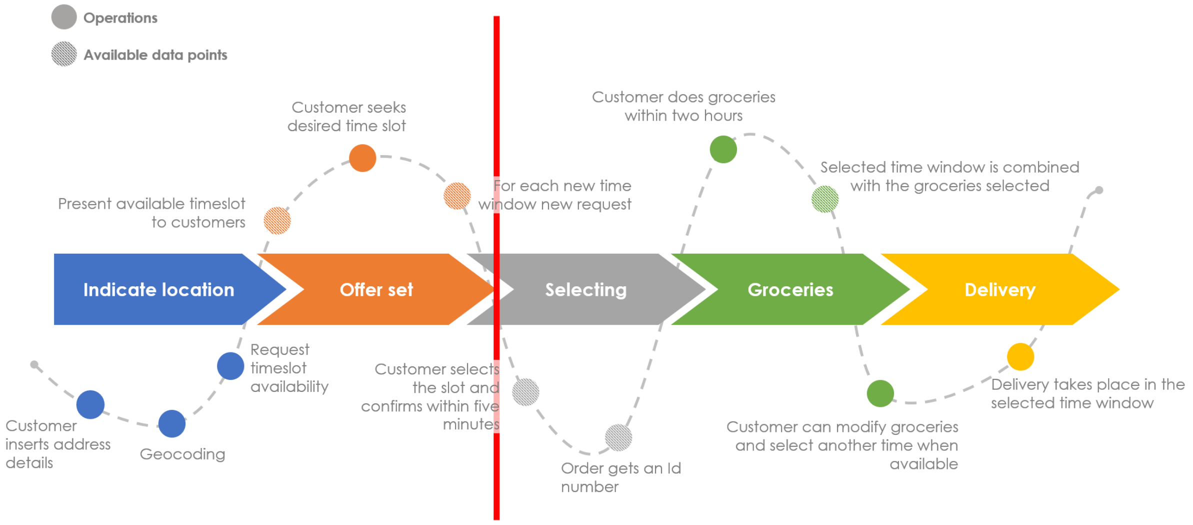

Operations
Available data points
Customer seeks desired time slot
Present available timeslot to customers
For each new time window new request
Customer does groceries within two hours
Selected time window is combined with the groceries selected
Indicate location
Offer set
Selecting
Groceries
Delivery
Request timeslot availability
Customer selects the slot and confirms within five minutes
Customer inserts address details
Geocoding
Order gets an Id number
Customer can modify groceries and select another time when available
Delivery takes place in the selected time window

Figure 4.1: Overview of the ordering process of online groceries. The red line divides the process into two parts: the left contains the offer set, the right part contains the chosen time slot and the actual order size. The left side will also be used to predict the selected time slot through a non-parametric and parametric model.

This research uses some available data from the online grocery store and focuses on one week in one division. The selection to use one week and division is based on the amount of data available as customers make many orders and changes. According to Figure 4.1, the process stores all of them separately. The predetermined delivery period starts on the twenty-eighth of November and continues till the fifth of December in 2022[1]. When looking in more detail, the data consist of two types, the

[1] Week 48

requests containing an offer set[2] and the orders containing the chosen time slot[3]. The request data include proposed offer sets but do not contain the selected time slot. The order data consist of chosen slots, including the changes in the order but do not contain the seen offer set to the customer. In addition, every order has a unique Id number, making it easy to see the customer's changes. However, offer sets do not contain this Id number, which makes it harder to merge the multiple-seen offer sets with the chosen time slot. How the two datasets are merged and which assumptions are needed for this will be explained later.

4.1. Time slot data

To obtain all request and order data for the predefined period, the requests and orders from twenty-seven days before the start of the delivery window must be considered, which means that the data will start on the first of November up until the fourth of December. It must be pulled from the data lake to use the data for analyses. This data lake stores all requests and orders for some time and allows filtering of the request data. The filtering allows for only extracting the offer sets containing a time slot in the delivery week. Therefore, this filtering technique assumes that an order is lost when the customer does not select a time slot or selects a time slot in another delivery week than the predetermined week 48. However, this filtering technique can not be used for order data extraction, making the number of points entering the analyse far more than for the requests. After extracting the order data, filtering will take place, and only the orders with a chosen time slot within the delivery window will be maintained. Figure 4.2 visualises the difference between the number of requests and orders per week made for the deliveries in week 48. Figure C.1 gives an overview of the number of weekly orders directly after extracting from the data lake.

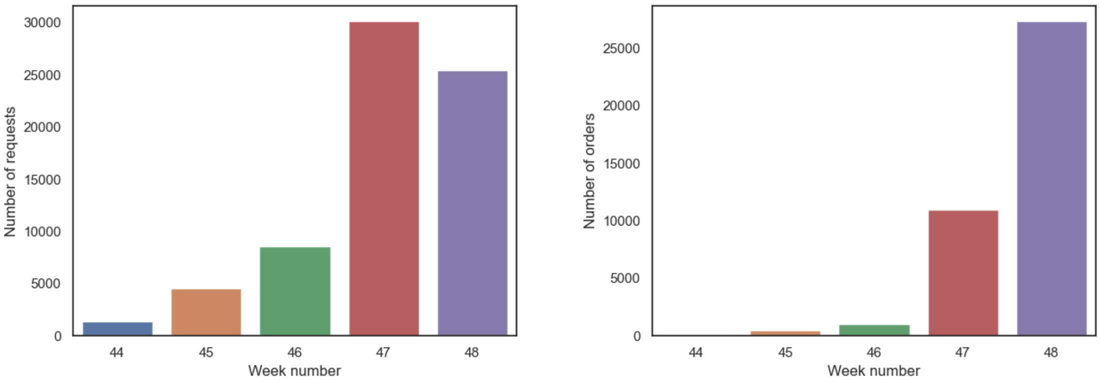


Figure 4.2: The left histogram represents the number of requests in the week by the customers, and the right histogram represents the actual orders placed or changed in the week for a delivery moment in week 48.

Figure 4.2 shows a difference between the number of requests and orders. One possible explanation for the higher number of requests in the first four weeks is that the customer asks for time slots in multiple weeks resulting in numerous requests as the system makes a new request for each change in days offered. Meaning that when a customer orders, for example, in week 46, the first request is for the next couple of delivery days. A new request is made if the customer wants to continue looking for other time slots and changes the time window. When the consumer returns to the previous offer set again, a new request is made since there is a possibility that a time slot is already booked and not available anymore. Therefore, a customer may make multiple requests for one order in a different and in the same week. Also, the performed filtering went along with the assumption that the order is lost when a customer makes several requests but ends up not ordering or ordering in another week. In other words, the customer makes one or multiple requests in the gathered weeks but does not select a time slot in the predetermined delivery week resulting in no additional order. When comparing the number

[2] Indicated with orange in Figure 4.1

[3] Indicated with grey and green in Figure 4.1

of requests and orders for week 48, the customer created more orders than requests, according to the histograms. This might be because a new order is made every time a customer changes the groceries by adding or removing products, which is done more frequently when the delivery window approaches. After looking into the distribution over the weeks, the subsequent step is looking into the distribution within the weeks. Figure 4.3 visualises the more detailed requests' distribution of the weeks divided into days with a histogram and the average distribution of the peak hours during the day with a kernel density estimate.

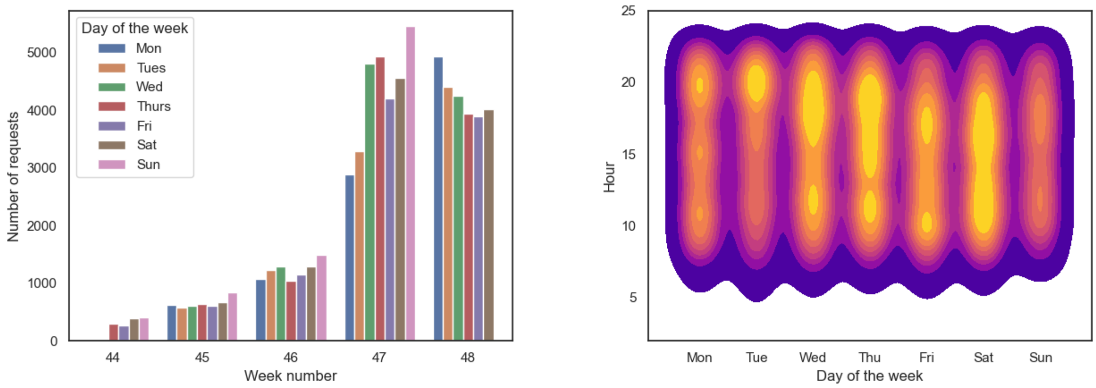


Figure 4.3: On the left, the requests' distribution within the weeks is given using histograms. The average peak hours within the days are indicated on the right with a kernel density estimate plot.

The histogram plot on the left side of Figure 4.3 confirms the increasing amount of made requests as the delivery week approaches. However, with a more detailed overview, the number of requests is decreasing in the delivery week, which is reasonable as the number of executable days and, as a consequence, time slots are reducing over the days. On the right side of the figure, the kernel density estimate plot indicates the total number of requests done during the week per hour, where the orange parts indicate more activity than the purple parts. When combining the interpretations of both plots, it becomes clear that most activity of requests is at the weekend during the day and on weekdays in the evenings, and only a few requests are made during the night. The finding that customers make fewer requests during the night can also be seen in Figure C.2, indicating the total activity per hour. When making the same set of plots for the order data, the expected result is that the histograms are higher for days closer to the delivery week. The kernel density estimate plot that indicates the activity of the order data may differ from the request data activity because orders can be made without a time slot request when the customer changes the groceries. Figure 4.4 provides an overview of the activity from made or changed orders. This figure can confirm the expectations about the order data as the histogram indicates a peak in the last days before the delivery week. Also, the kernel density estimate differs, suggesting that most changes and orders are made during the evening or Sunday.

One possible explanation for the decreasing number of requests over time is, as mentioned above, the number of executable time slots and, therefore, the decreasing size of the offer set. In Figure 4.5 on the left side, the number of offered time slots can be found. Based on this overview, it can be found that the amount of time slots offered in week 48 is decreasing compared to week 47 and are less evenly distributed, meaning that there is a higher percentage of smaller offer sets. In addition to the number of time slots offered, the requests' performance may substantiate this explanation, especially since the performance indicates the time it took to assemble the offer set with executable time slots. When it takes longer to create the offer set, it is harder to fit the new or changed order in the already existing routes; as a result, the performance deteriorates. Figure 4.5 shows on the right side how the performance changes over time, as it indicates that making the offer set in week 44 does, on average, not take longer than 0,248 seconds and in the delivery week takes, on average, 0,316 seconds which is an increase of 27,4%.

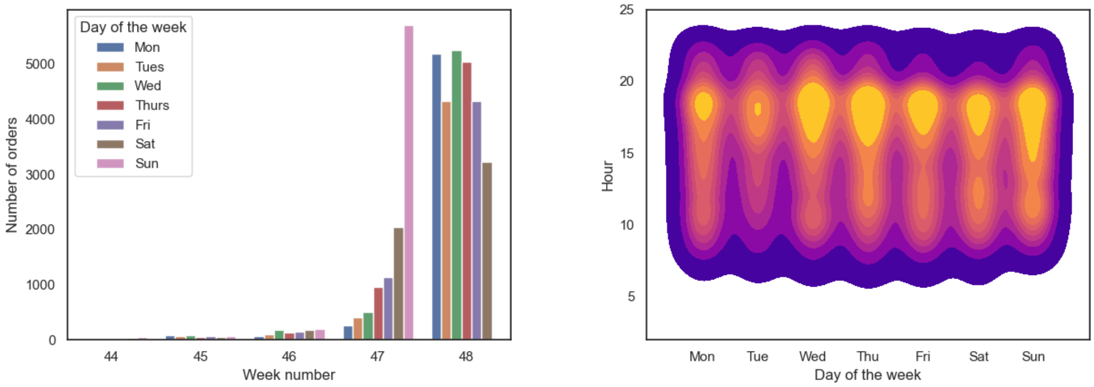


Figure 4.4: On the left, the orders' distribution within the weeks is given using histograms. The average peak hours within the days are indicated on the right with a kernel density estimate plot.

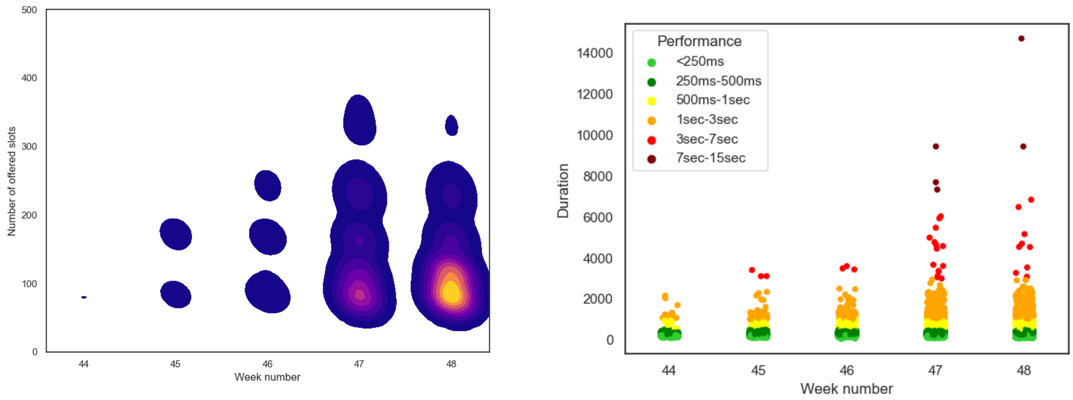


Figure 4.5: Overview of the number of slots in week 48 provided in the offer set and performance, indicated by time, for creating the offer set over the weeks.

As above-mentioned, it is possible that the customer enters the website and retrieves an offer set but that the preferred time slot is not available anymore and, as a result, leaves without ordering. Hence, for identifying customers' behaviour, it is necessary to include the possibility that a customer does not select a time slot in week 48 and decides to proceed with the order for another week or leaves the process. Both situations are identified as lost orders in this study as it is only focusing on orders placed for week 48. To create the data set for lost orders, the lost orders in week 47 and 48 are taken into account. Because as shown in Figure 4.2 in these two weeks, most of the orders are placed. In the weeks before, it is assumed that customers look without the thought of wanting to order and that they have a more exploratory view of the possibilities. Beyond that, it is also assumed that it is likely that most time slots are still available based on the number of placed orders in weeks 44, 45 and 46. The number and distribution of the lost order requests can be found in Figure 4.6.

Based on the histograms on the left side in Figure 4.6, it can be found that the most orders are lost in week 47. The kernel density estimate plot has approximately the same distribution as the kernel density estimate plot in Figure 4.3, which was expected as the customers from the lost orders also made an offer set request belonging to the number of made requests. The activity on Sunday is the main difference in the kernel density estimate plot in Figure 4.6 and is also explicable since the data set only contains one Sunday since on Sunday in week 48, no new orders can be made for week 48 and therefore there will be no lost orders on that day.

Both the order and request data, including the lost orders, will be used to estimate if the customer

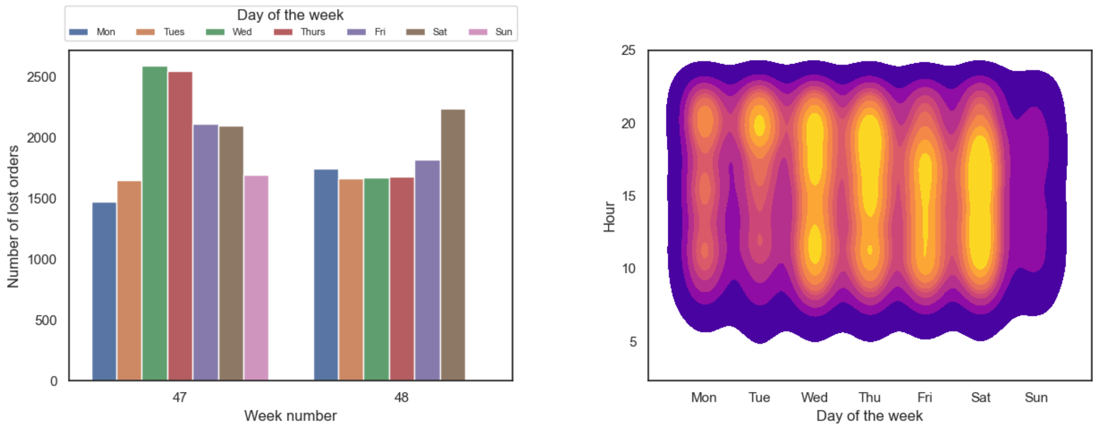


Figure 4.6: On the left, the lost orders' distribution within the two weeks is given using histograms. This plot consists of two weeks based on the made assumption about the data. The average peak hours within the days are indicated on the right with a kernel density estimate plot. It is found in this plot that the average activity is lower on Sunday compared to the other days because only one Sunday is considered. This is also the case in the histogram plot.

selected a time window for delivery and, if so, which time slot. Since the deliveries in week 48 are already executed, it is possible to look into how they were distributed throughout the week in Figure 4.7. Figure 4.7 shows that the morning time slots for delivery were the most popular, and there is only a slight difference in the number of deliveries per day. When comparing the deliveries executed in week 48 with the weeks 44, 45, 46, and 47, the same pattern can be found, meaning that the most popular delivery moment is the morning and that there is only a slight difference between the delivery days. An overview of the distribution of the history of deliveries can be found in Figure C.4.

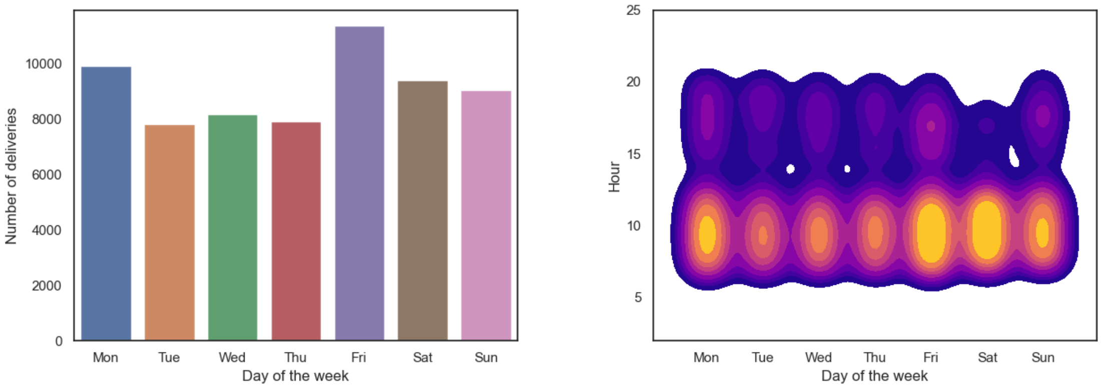


Figure 4.7: On the left side, the deliveries' distribution in week 48 per day can be found. On the right side, the activity of the deliveries is visualised using a kernel density estimate plot. Also, the kernel density estimate plot shows that all deliveries are between five in the morning and eleven in the evening.

When all data are gathered, the offer set needs to be combined with the selected time slot orders. Since this combination of data is not yet available. The process of combining the two data sets and including the lost requests are performed in different steps; an overview of this can be found in Figure 4.8. The first step is to create a list of all unique orders based on Id number and location. Based on this list of unique orders, the moment when the last time slot was selected is extracted. With the order placement time and the location known, the offer set request corresponding to the order can be found. For this process, a few assumptions are made. The first assumption is that the time of the order is at least five minutes later or one minute earlier than the time of the request offer set[4]. This assumption is made since it is possible that the order time can be one minute before the request time because of a possible

[4] This assumption is made in coordination with an expert in this process.

delay in the system. The second assumption is that orders whose location changed last minute will not be included as the request set contains the old location making it too complicated to match the right sets. And the last assumption is that only the last selected time slot is taken into account, as it is assumed that customers only then know when they want to have their groceries. Based on these assumptions, combining 98% of the unique orders from week 48 with their matching offer sets is possible. After combining the orders with their request sets, it is possible to indicate which requests are lost and are combined to make the seen offer sets that are lost. As a result, it is found that from all the requests made, two-thirds of the requests set are assumed to be lost for week 48. The real number of lost orders is assumed to be lower as this research only focuses on week 48, making the number of orders considered lost higher.

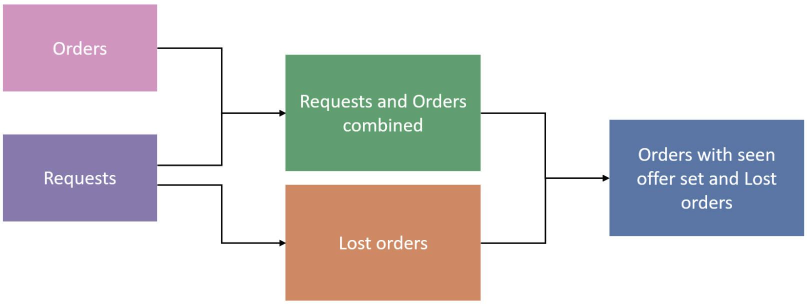

Orders
Requests
Requests and Orders combined
Lost orders
Orders with seen offer set and Lost orders

Figure 4.8: Overview of the process to derive the data set that can be used for predicting if the customer will order and, if so, which time slot they choose.

For estimating the delivery moment, three sequential prediction models are needed, as discussed in section 3.6. To perform the three models, data about the delivery week, day and part of the day are needed. This section already discusses data that can be used to predict the delivery day and part of the day. However, to predict in which week the delivery will take place, more data from other weeks is needed, which is not yet discussed. Based on the filtering technique used to extract the request data, it is not possible to create offer sets or take request data that result in lost orders into account for predicting the delivery week. Nevertheless, order data can be used as this data set is not filtered before extracting and therefore consists of orders in multiple weeks. An overview of the distribution for all order data, including orders with a delivery moment in another week than week 48 can be found in Figure 4.9. Since only order data can be used to predict the delivery week, the models do not consider that an order can be lost; therefore, this option will be included while predicting the delivery day.

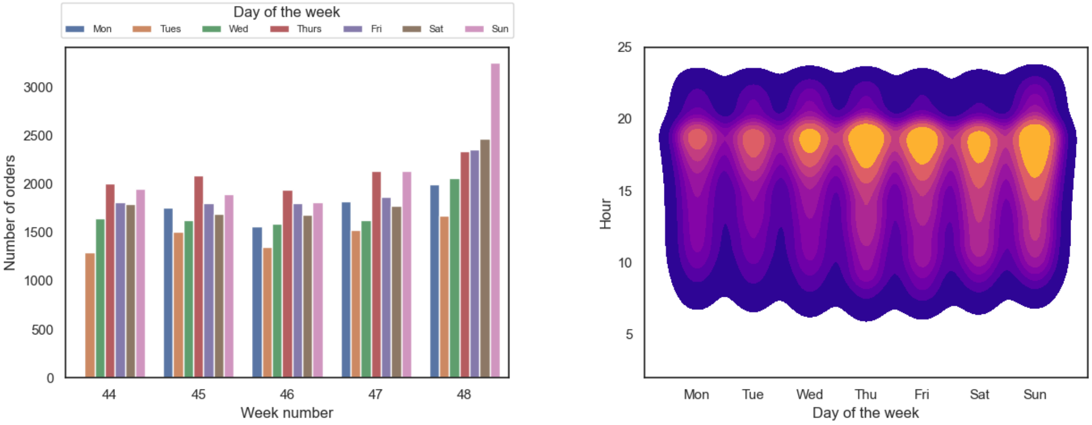


Figure 4.9: The orders' distribution for the different weeks can be found on the left side. On the right side, the activity of the orders is visualised using a kernel density estimate plot where the most activity is found during the eve.

Based on the histogram displayed in Figure 4.9, it can be found that the number of orders follows the same pattern in the first four weeks but that in week 48, the number of orders is increasing. The increase in orders may be because Christmas is coming and in week 48 the time windows for deliveries in this period open. In the kernel density estimate plot on the right side of Figure 4.9, it can be found that most orders are made during the evening. That most orders are placed in the evening was already concluded based on Figure 4.4; however, in this kernel density estimate plot, the activity is even less distributed.

4.2. Class weights

Now the data is gathered, it is possible to check if the data set is imbalanced, as discussed in section 3. This can be established by investigating the number of instances of each category within the set. Table 4.1 provides an overview of the label occurrence in the three used data sets. Based on this table, it is found that weeks 45, 46, 47, and 48 recur more often than weeks 50 and 52 as they are not highly represented in the data set, which makes it possible to consider the data imbalanced. When looking into the distribution of categories used for predicting the days, it is found that there is a slight difference between the weekdays but that the Lost order label has a solid presence. That the executed deliveries over the weekdays are evenly distributed was expected as Figure 4.7, and Figure C.4 already provided these insights. The very high number of Lost orders was not envisaged. Still, it can be explained since an order is already considered lost in this study when the customer sees one available time slot from week 48 but chooses one in a different delivery week, causing the number to rise rapidly. However, based on these results, it can be found that the data are imbalanced. The last distribution of labels is for predicting the part of the delivery day, and it can be found that no deliveries are taking place late at night. Since there are no deliveries at this part of the day, the model does not consider this opportunity a possible outcome. Nevertheless, there is still a difference between the number of morning and night deliveries, which shows that morning deliveries are more popular than night deliveries.

Week	Number	Percentage
44	7670	12,0
45	12319	19,4
46	11535	18,1
47	12080	19,0
48	12177	19,1
49	4515	7,1
50	959	1,5
51	2271	3,6
52	134	0,2

Day	Number	Percentage
Monday	1919	5,2
Tuesday	1712	4,6
Wednesday	1362	3,7
Thursday	1438	3,9
Friday	2052	5,5
Saturday	1769	4,8
Sunday	1795	4,8
Lost order	24993	67,5

Part of the day	Number	Percentage
Early morning	2649	22,0
Morning	5579	46,3
Noon	1846	15,3
Eve	1832	15,2
Night	141	1,2
Late night	0	0

Table 4.1: Overview of distribution of the data within the data set for predicting the delivery week, day and part of the day, respectively.

Based on the outcomes from Table 4.1, it can be concluded that all data sets are imbalanced. As a result, class weights will be assigned when necessary to prevent the model from disregarding the minority classes. Whether it is necessary to use the class weights depends on the model performance of both the model with and without the weights. The class weights can be determined with Equation 4.1 and are based on the complete array of categories and the set of unique labels [56].

$$\textit{Class weight} = \frac{\textit{Number of samples}}{\textit{Number of classes} \text{ x } \textit{Number of occurrence of class in array}} \tag{4.1}$$

4.3. Feature engineering

After preparing the data where it is obtained, analysed and merged into offer sets where the outcome is combined with the seen request set, feature engineering can be done. Feature engineering is a critical process to select and transform data into features that can be used in supervised learning to improve the model's performance [86]. The requests and order files already store numerous data points; however, more information can be found and derived from historical data. The request files, for example, already store the number of provided time slots and performance, and the order files the unique Id number. Both file types store the created date, time and location through longitude and latitude. Before new features are created, whether any data in the files are missing should first be examined.

Missing data can be indicated by employing heatmaps for the two sets and can be found in Figure C.3. Based on Figure C.3, it can be concluded that the order file has no missing data points. However, the request set has some missing points in the *Number of orders in district*. The missing data points can be explained by the fact that consumers from these districts made a request but did not make an order earlier in November and will be replaced by zero. During this research, smaller segments are created based on locations making it possible to replace the missing data points with data from a more extensive segment when the data interpretation allows it.

When there are no missing data points, the data can be scaled, and One-hot encoding can be used, for example. Feature scaling is an essential step before the data is used. The features will be normalised in this research using a min-max normalisation that specifies all feature values between zero and one and does not influence the data distribution. Features representing an element of a finite set can be used for One-hot encoding. With One-hot encoding, all elements of the finite set will be represented by an index, and only one element has a value of one, and all others have a value of zero. The following is an overview of all available data points that can be used to create new features or for the aforementioned One-hot encoding. A more detailed overview of these data points can be found in Appendix D.

Request data:

- Date and time when the request is made
- The estimated weight values for the amount of chilled, frozen and crates
- Delivery location using longitude and latitude
- Postcode
- Population density in the area
- Annual gross income of the area
- The performance (how long it took to create the offer set)
- The number of provided time slots and the number of days
- Date, time, cost and distance of the provided time slots
- The number of time slots per day and part of the day

Order data:

- Date and time when order is made
- The weight values for the amount of chilled, frozen and crates
- Delivery location using longitude and latitude
- Postcode
- Population density in the area
- Annual gross income of the area
- The selected time window
- Id number
- The amount of already made orders
- Start time and day of last delivery
- The amount of time since the last delivery
- The most common delivery day at the address and district
- The most common delivery moment at the address and district

The above overview consists of derived and obtained data points; one of the derived data points is the postcode. The postcode creates customer segments based on location, which can be helpful as longitude and latitude are too specific and unique for a location. To obtain the postcode, the Geopy library in python can help to transform the coordinates of addresses into a postcode, city, or neighbourhood, among others [23]. All this information can be regenerated since the used postcodes' structure has an outward and an inward code and supports geographic layers such as the postcode area, district and sector. Based on the postcode, the area's population density and annual gross income can be derived from the Census where multiple datasets are available, including sets with information about the population density and annual gross income per area. Extracting the postcode can be relevant in more than one way, as it, in addition to the segmentation, indicates the population density and the annual gross income in a particular district. The annual gross income is of interest since the cash-rich and time-poor city residents have a growing desire for online groceries and demand that the delivery takes place on their preferred day and time [7]. Therefore, income might be a helpful feature for getting more insights into the customers' behaviour. The population density feature may not directly impact the customers' behaviour. However, it could affect the time slots provided as companies search for highly dense locations to maximise market share and perhaps offer fewer time slots in less densely populated parts. The availability may differ because the online groceries uptake is driven by age, affluence and access to physical retail opportunities [52][29][81].

Another feature that can help determine the selected time slot is the date and the time when the customer made the order. When using hours or days of the week, it is essential to consider that these features are cyclical. Since otherwise significant information will vanish when the features are not converted. An example where information can be lost is in the day hours. Because a ML model considers 23 and 0 far instead of close to each other. Therefore to preserve the information that 23 and 0 are close to each other, the feature can be represented as coordinates on a circle with an x and y location. The same can be done for the day of the week as, in reality, Saturday is closer to Monday than Wednesday, but when using One-hot encoding, for example, this does not give the same insights. Because One-hot encoding can create a boolean feature for every weekday, giving information about the day but not the relation between the days resulting in the day order no longer matters. As a result, to consider the order, the angular distance of the circle coordinates can be taken into account, of which the cosinus and sinus values are calculated, resulting in unique pairs of values [77][32][57].

In addition to already mentioned derived and obtained data points, historical data might also help to determine the selected time slot in the decision model. The data history in this report is limited to only the November orders delivered before week 48 due to the extracted data. Nevertheless, different historical data can be obtained, such as the last delivery day and moment, the time since the last order, the number of orders made in November and the number of hours the customer booked before the last delivery. Since this historical data can only be obtained from orders before the predetermined delivery week and some new customers did not order before, the previously mentioned data points are also created based on the most common result of the district and area.

4.4. Feature correlation

The features mentioned above are needed to create the best feature set achievable to predict the selected time window by the customer as well as possible. The prediction to determine the chosen time window is divided into three steps. The first prediction step is predicting the delivery week, the second is predicting the delivery weekday, and the last is predicting the part of the day when the delivery is happening. Dividing the prediction steps is done as the choice models have categories as output. When the predictions are made simultaneously, the quality deteriorates because of the many possible classes. Consequently, three optimal feature sets with existing and newly initiated features must be created to perform all predictions as best possible. However, a possible drawback of introducing multiple features is that they can correlate with each other in addition to an expected correlation with the target variable. Accordingly, multicollinearity can occur when the variables are correlated and impact the model's accuracy, as in this case, one variable can be linearly predicted from others with a high degree of accuracy. Therefore it is essential to identify and remove features associated with high multicollinearity. A correlation heatmap can be used to visualise this relationship between the variables, representing the relationship's strength and direction. Since correlation is used to determine if there is a cause-and-effect relationship between two variables, it is a statistical measure that can express the strength of the relationship. The relationship between the two variables can both be negative and positive. If two variables are correlated positively, they move in the same direction, meaning that when the value of one is increasing, the other is also increasing. Two variables are correlated negatively when they move in opposite directions, meaning that if one value increases, the other decreases. Nevertheless, a correlation between variables does not necessarily imply causation, as other aspects may also play a role [61][13]. An overview of the correlations between the features can be found in Figure 4.10.

Based on Figure 4.10, the correlation of the features can be found, and similar heatmaps are used for selecting the features in the filtering method. In this figure, the correlation value is colour coded,

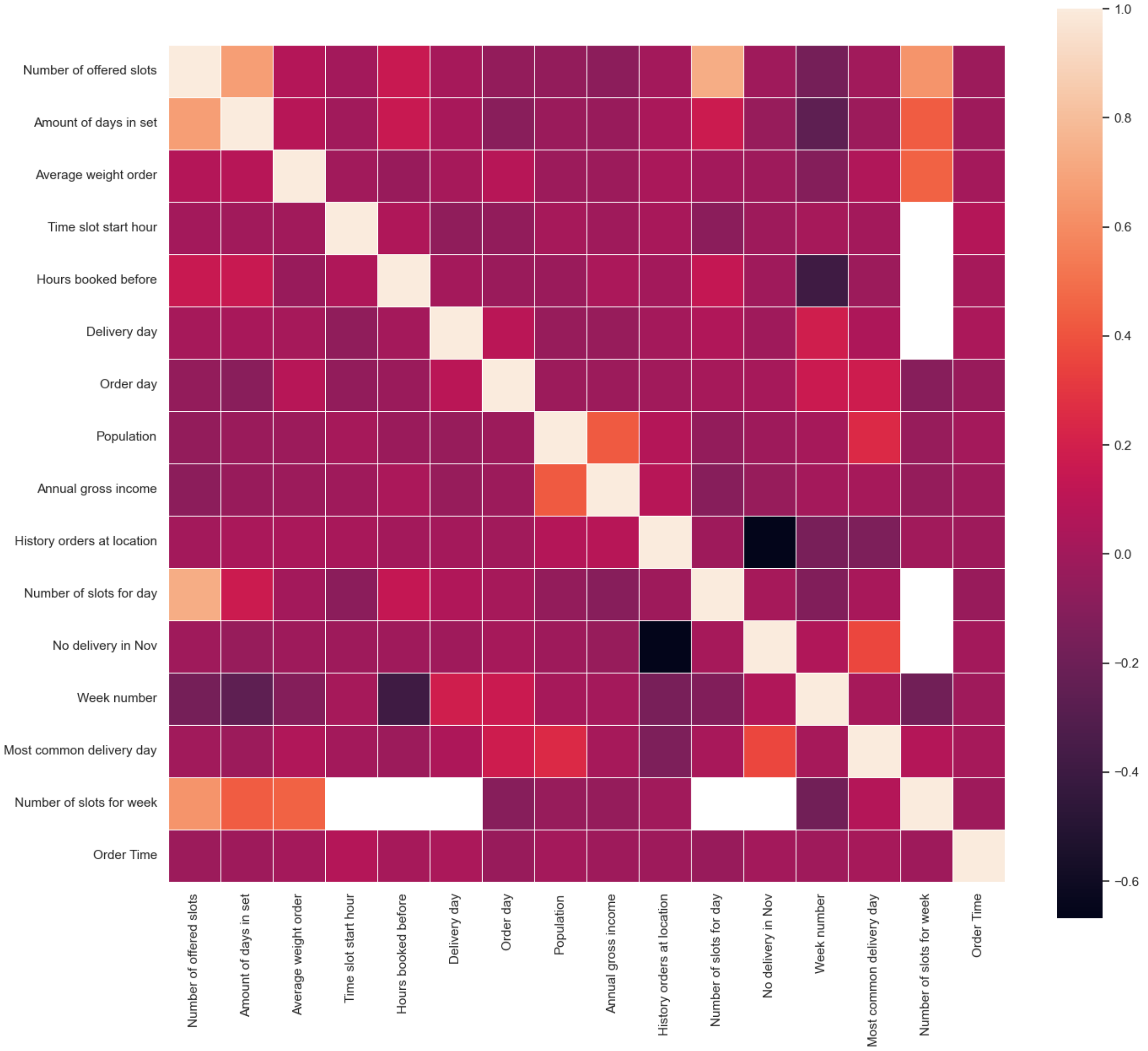


Figure 4.10: Heatmap of correlation between features

meaning that white indicates a positive correlation of one and black has a negative correlation of one. When looking into more detail about the feature *Week number*, it can be found that this feature correlates with the *Number of offered slots, Amount of days in set, Hours booked before and Delivery day* among others. However, not all these features can be used to determine the week number when the delivery will take place since some features such as *Hours booked before* contain more information about the selected time slots, which is not known at the moment of ordering. Hence the features used for predicting the delivery week are based on the correlation matrix in combination with the available variables for all weeks and are:

- Request time, day and week
- The average order weight
- The annual gross income

The feature selection for predicting the delivery weekday is made similarly; only more variables are available after merging the offer sets with the selected order time. Consequently, the following features are used to estimate whether the order is lost and, if not, which day the customer selected for the delivery.

- Request time, day and week
- The average order weight
- The annual gross income
- The delivery week
- The most common delivery day
- The number of previous orders
- The number of available slots per day

For the last prediction model used for estimating the part of the delivery, all previously mentioned features are combined with new ones. New features can be added as more information becomes available per prediction step. The new features that are used are:

- The delivery week and day
- The part of the day from the last delivery
- The number of available slots per part of the day

4.5. Summary

First, an understanding of the ordering process is gained, and data is extracted to identify the customers' behaviour better. After knowing how the process proceeds and which data points are stored, the first analyses can be made. During this analysis, the distributions, performance and peak hours of the orders, requests and deliveries are indicated. During these analyses, it is found that the number of available delivery slots is related to the week number and the system's performance. However, to make the data useful for this research, the requested data need to be merged with the order data, which can only be done in combination with some assumptions. The first assumption is that the order time is within five minutes after and one minute before the request time. The reason the request time can be after the order time may be due to possible delays in the system. Another assumption is that only the last time a time slot is chosen is taken into account, as it is assumed that customers only then know when they want the delivery. The last assumption made is that when the delivery location changes last minute, this order is not taken into account. With these assumptions, it is possible to merge 98% of the order data to the correct offer set, which will be used to check whether the data set is imbalanced. After checking the distribution of the data set, the feature engineering process can start, and information about a possible correlation between the features can be provided.

During the feature engineering, the data is checked for missing points and filled with other data points if needed. From the point where the data contains all the data points, new features can be derived from existing data to provide more insights into the selection process. The derived data are, for example, the postcode, annual gross income and population, and to contain the cyclically of time, the days and hours are converted to two different values. These values are derived using cosinus and sinus to create x and y coordinates on a circle. Based on these two values, information on the time units can be found, and the relation between the different units making the order relevant. This latter information can not be obtained when using One-hot encoding.

Before the parametric and non-parametric models are used, the feature selection is made, after which the data can be normalised with a min-max normalisation which does not influence the data distribution when specifying the feature values between zero and one. The feature selection is made to achieve a feature set that estimates the customers' behaviour as well as possible. In order to create the most optimal set, not all features will be included as multicollinearity may occur then. A correlation heatmap can be used to view the linear relations between the features. Based on this heatmap, the relation between the different variables can be found, making it easier to select the different features during the filtering method for predicting the time slot. Because useful features for prediction are highly correlated with the time slot but are uncorrelated among themselves.

5 Results

With all the gathered data, and after performing some analyses, the parametric and non-parametric choice models can be used to predict the selected time slots. To ensure that both the more advanced models add value to the prediction accuracy, they will be compared with the Benchmark model. The Benchmark model, explained in section 3.6, is a simple model based on the available data and predicts in which week the delivery will occur, on which weekday and part of the day. As mentioned, the predictions are performed in three steps for predicting the selected time slot. The more advanced parametric and non-parametric models will follow an identical structure. This means primarily estimating the delivery week based on several available features. For this estimation, a combination of the deliveries of week 48 and the history orders are used since both are available due to the data extraction. The outcome categories of these models are based on the unique weeks in the data set. After the week of delivery is predicted, this result will be utilised as an additional feature in conjunction with both the previously stated and newly obtained features to predict the delivery day. The features employed to predict the delivery week and day are presented in Figure 5.1. Additionally, this figure showcases the characteristics utilised in the latest prediction stage, which forecasts the specific part of the delivery day. For predicting the delivery day, eight distinct outcome categories are possible: the seven days of week 48 and the category indicating that the order is lost, as not all requested sets result in an order during week 48. The ultimate stage in this estimation procedure entails predicting the delivery time, which divides each day into six equally distributed portions. The used day parts are *late night, early morning, morning, noon, eve* and *night* and are four hours long. The delivery time for each customer can be predicted using these three steps alongside all the utilised features and offer sets.

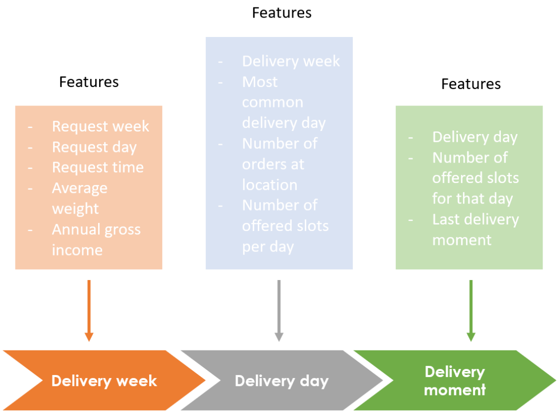
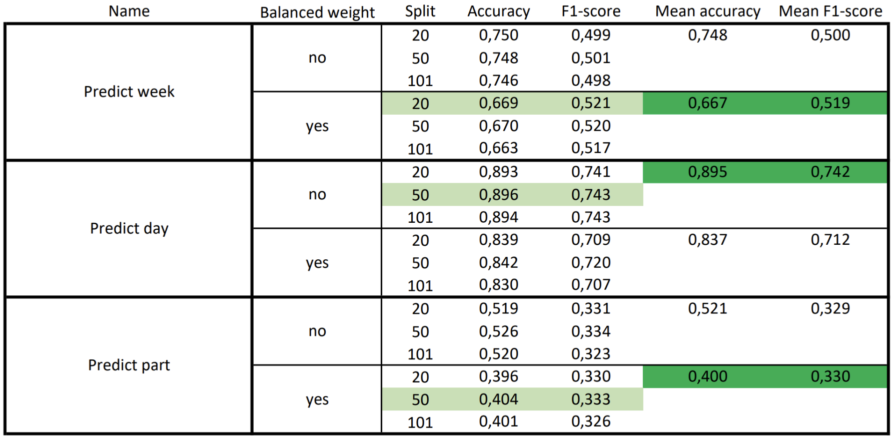


Figure 5.1: Overview of used features per prediction step. Each new step also uses the features from the steps before.

5.1. Multinomial logit models

The initial advanced model utilised is the parametric MNL model, which predicts the delivery week, day, and time segment. MNL models are suitable for multi-class issues and can be implemented using diverse types of solvers, as outlined in section 3.4. In the three prediction stages, the four solvers are employed to establish which one is best suited for the data in this study. The F1-score will be used to evaluate and compare the predicted performance of all four solvers for each prediction step. Based on this score, it will be determined which solver is employed for each prediction. It should be noted that in the MNL model, different solvers can be utilised for each prediction stage. Section E.1 provides an overview of all predictions' outcomes, while Table 5.1 presents the results of the best-performing solvers, with and without class weights. The findings indicate that the newton-cg solver delivers the best performance and will be employed in the MNL model in all stages. Furthermore, the F1-score is also utilised to evaluate whether class weights are required to counteract imbalanced data and enhance performance. Table 5.1 provides an overview of the scores obtained with the newton-cg solver, while the results of the other solvers can be found in Table E.1 and Table E.2. According to the results in Table 5.1, class weights are beneficial for predicting the week and time segment, where imbalanced data is compensated for, and the model's performance is improved. However, for forecasting the delivery day, the F1-score with class weights is inferior to the score without them and, therefore, will not be incorporated into this MNL model.



Name	Balanced weight	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict week	no	20	0,750	0,499	0,748	0,500
		50	0,748	0,501		
		101	0,746	0,498		
	yes	20	0,669	0,521	0,667	0,519
		50	0,670	0,520		
		101	0,663	0,517		
Predict day	no	20	0,893	0,741	0,895	0,742
		50	0,896	0,743		
		101	0,894	0,743		
	yes	20	0,839	0,709	0,837	0,712
		50	0,842	0,720		
		101	0,830	0,707		
Predict part	no	20	0,519	0,331	0,521	0,329
		50	0,526	0,334		
		101	0,520	0,323		
	yes	20	0,396	0,330	0,400	0,330
		50	0,404	0,333		
		101	0,401	0,326		

Table 5.1: Overview of the results for the newton-cg solver in combination with using the balanced class weights and different splits of the MNL model. The model with the highest F1-score for predicting the week, day and part of the day is highlighted in green.

5.1.1. Prediction of delivery week

As explained earlier, the first stage in the process entails forecasting the delivery week through an MNL model. Balanced class weights are employed in the initial prediction to enhance the model's performance and yield a higher F1-score, as illustrated in Table 5.1. The results of the MNL model are depicted in a confusion matrix on the left-hand side of Figure 5.2, and the receiver operating characteristic (ROC) plot is located on the right-hand side. The model's accuracy for predicting the delivery week is $0,669$, indicating that $66,9\%$ of the total predictions are correct. To gain deeper insights into the most influential features in the forecast, feature importance measures can be computed and presented as visual plots, as detailed in subsection G.1.1. These plots provide a comprehensive summary and interpretation of the feature weights assigned to each predicted output week. The feature importance plots indicate that the request week plays a significant role in forecasting the delivery week. The request week feature has a negative impact on the first three weeks, indicating that the probability of delivery during these weeks decreases as the week number increases. From week 47, the week number positively impacts the delivery week. Another essential feature is the average weight order, with a positive impact that increases every week till week 47. From week 48 to week 51, the influence is negative but grows over time. A summary of the absolute coefficient of the feature importance can be found in Figure 5.3, which indicates that the week number of the requests and average weight order have the most impact. Thus, Figure 5.2, subsection G.1.1, and Figure 5.3 offer a comprehensive understanding of the MNL model's delivery week predictions.

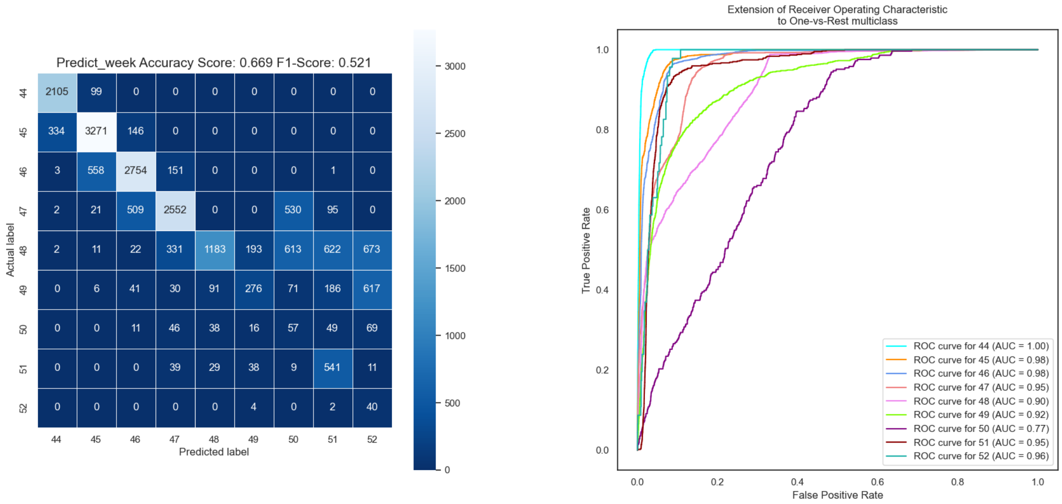


Figure 5.2: On the left side, a summary of the results for predicting the delivery week can be found in the confusion matrix, and on the right side, the ROC curve is displayed.

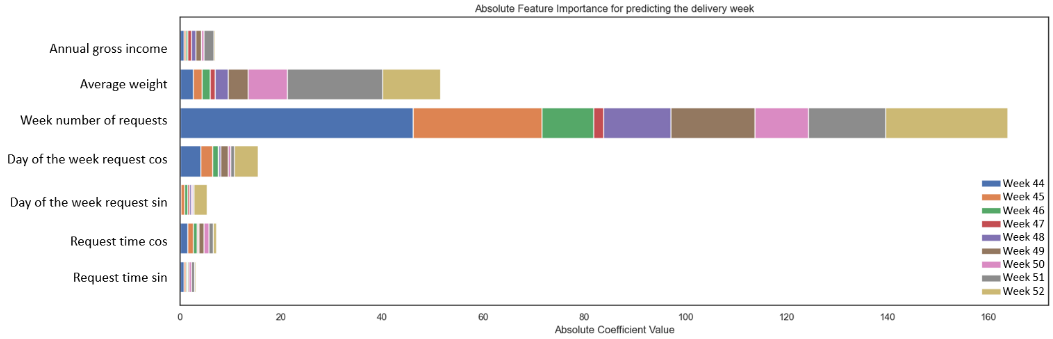


Figure 5.3: Summary of feature importance using the MNL model for predicting the delivery week.

In Figure 5.2 on the left side, the confusion matrix can be found, summarising the prediction results and showing the number of correctly predicted categories and when fault the predicted category. Based on the confusion matrix, it can be found that in the first few weeks (week 44 up until week 47), the predictions are pretty well. However, predicting the correct week in the later weeks is harder. Possible explanations for the decline in performance are that because of Christmas, people behave differently than expected and the number of occurrences in the data set. As reported in Table 4.1, it can be observed that the proportion of data instances belonging to later weeks is smaller, which makes it more challenging to predict these instances accurately. On the other hand, for the earlier weeks, the most significant error is predicting the delivery week either one week early or one week late. One possible reason for this issue could be that the model considers a strict classification of weeks, leading to misclassification, even if the difference in days between the actual and predicted delivery dates is not substantial. For instance, when the model predicts that the delivery will take place in week 48, but instead, it will actually be on Monday in week 49, the difference is relatively small in reality since it differs a few hours. However, the MNL model treats the prediction as completely wrong. To address this issue, the model that predicts the delivery day can include two days before and after the predicted delivery week. This approach can correct the small fault in the week prediction when the correct day is predicted. Despite this, the F1-score of the MNL model is 0.521, indicating the overall performance, and can be compared with the Benchmark model to determine if there is an improvement in prediction performance. Compared to the Benchmark model, the F1-score improved, rising from 0.420 to 0.521, thus indicating a performance enhancement. To obtain more information about the model, predicted probabilities for each class will be used to visually represent the performance of the multi-class classifier by using a binary classification problem. The most commonly adopted one-vs-all approach will be used, where one class is regarded as positive while the rest are considered negative. To create ROC curves that provide more insights, the threshold for the positive class will be varied to determine the true and false positive rates. The ROC curves and the Area Under the Curve (AUC) on the right-hand side of Figure 5.2 will therefore illustrate the trade-off between sensitivity and specificity for each category. The true and false positive rates will be calculated based on the following equations, respectively: $(TP/(TP+FN))$ and $(FP/(TN+FP))$. The ROC plots do not depend on the class distribution and help evaluate the performance as the classifier closest to the top-left indicates the best performance. The AUC will summarise the classifiers' performance into a single measure, where an AUC of one indicates that the model is highly effective in distinguishing between the different classes. At the same time, an AUC close to 0.5 suggests that the model is no better than random guessing, and AUC values below 0.5 indicate that the model is performing worse than random guessing and should be avoided. As a result of the ROC and AUC measures, it can be found that this MNL model can separate week 44 best and week 50 worst, which is in line with the interpretation of the confusion matrix.

To get more insights into the model's performance, the probability density distribution can be visualised using the kernel estimate plot. This plot, visualised in Figure 5.4, contains the probability distribution of the different classes in the MNL model. It can be found that the densities are smoothed, resulting in the plot containing predictions probabilities below zero and above one, which exceeds the possible probability value. The smallest and highest probabilities are mentioned in the description to ensure that this is due to the smoothing and that the values do not overshoot the zero and one probability boundaries.

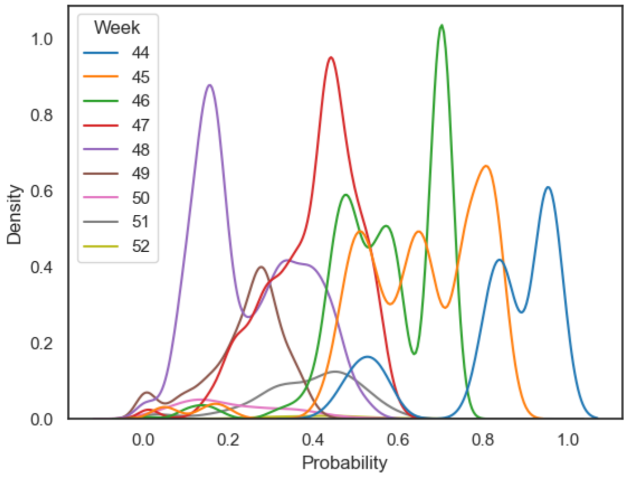
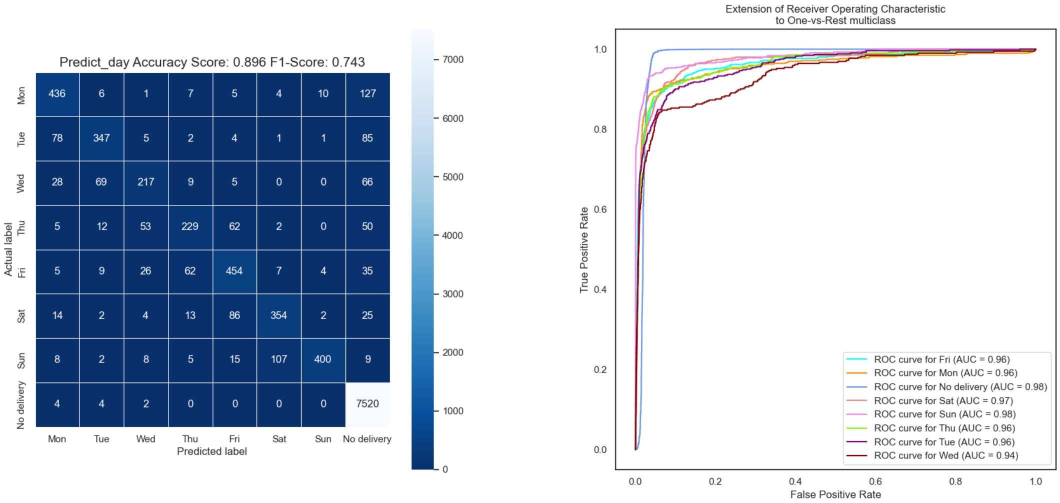


Figure 5.4: Probability distribution for the weeks 44 up until 52. The lowest and highest probabilities are checked to ensure no infeasible probabilities and are 0.000 and 0.968, respectively.

Upon examination of Figure 5.4, it is apparent that the probability distribution for week 44 has the most significant right peak, followed by week 45 and 46. The MNL model calculates these probabilities for each label during the prediction process. However, it is important to note that these probabilities do not necessarily correspond to the correct label predicted by the model. In other words, the probabilities do not reflect what the model should have predicted but what probabilities the model assigned to the correct label during the process. The density of the plot indicates that weeks 51 and 52 have less representation in the dataset, which is in line with the distribution tables from the data analysis. In addition, this figure provides insight into why the MNL model's predictions become more scattered over time, as probability peaks for later weeks are located towards the left side of the plot or may not exist at all. Peaks on the left side of the distribution indicate lower confidence in the predicted week, as the model cannot significantly distinguish the correct label from other possible outputs. This means that the model may have assigned a higher or equal probability to other outputs, resulting in a lack of confidence in the predicted label during the selection process as this is based on the given probabilities. Conversely, earlier weeks, such as week 44, have probability peaks between 0.9 and 0.8, indicating that the model is more confident in selecting the predicted week. As a result, the decreasing trend of the MNL model's accuracy over time can be attributed to decreasing confidence in its predictions as time progresses. This could be due to various factors, such as data availability and upcoming holiday periods.

5.1.2. Prediction of delivery day

The study employed a second MNL model to predict whether a customer placed an order and, if so, the delivery day of the order. The results in Table 5.1 show the comparison of the MNL model with and without class weights and reveal that the model without class weights performs better in terms of the F1-score. Therefore, the MNL model that predicts the delivery day does not contain class weights. As mentioned above, this stage in the prediction process can resolve the fault of the previous model of orders classified to the wrong week by including two days before and after the predicted week. However, because of the data used in this study, including the extra four days does not add value. The confusion matrix evaluating the model's performance can be found on the left side of Figure 5.5 and a ROC plot on the right side. The accuracy of the model for predicting whether the customer made an order and which delivery day they selected is 0.896, indicating that 89.6% of the predictions are correct. However, the confusion matrix shows that the model predicts most of the data points as No delivery, as the dataset has a large number of such points. When examining the performance of the predicted delivery day, it is found that the largest deviation occurs for the adjacent days to the actual delivery day. In subsection G.1.2, a detailed overview of the feature importance can be found, where the number of available slots for the delivery day is the most important feature overall. For example, when Monday is predicted as the delivery day, the probability is positively influenced by the number of available slots. The higher the value, the higher the likelihood that the model will predict that the delivery will take place on Monday. Other important features include the order's average weight and the week the request was made. However, these two features are particularly important in predicting No delivery labels. One interesting finding is the significant impact of the number of available delivery slots on the prediction of lost orders on Monday and Tuesday of the following week. The feature importance plots reveal that a high number of available timeslots on these two days have a considerable negative effect on the probability of the order being classified as lost. To gain further insights into which features are important for each class, a summary plot of the absolute importance can be found in Figure 5.6.


Predict_day Accuracy Score: 0.896 F1-Score: 0.743

Actual label \ Predicted label	Mon	Tue	Wed	Thu	Fri	Sat	Sun	No delivery
Mon	436	6	1	7	5	4	10	127
Tue	78	347	5	2	4	1	1	85
Wed	28	69	217	9	5	0	0	66
Thu	5	12	53	229	62	2	0	50
Fri	5	9	26	62	454	7	4	35
Sat	14	2	4	13	86	354	2	25
Sun	8	2	8	5	15	107	400	9
No delivery	4	4	2	0	0	0	0	7520

Figure 5.5: On the left side, a summary of the results for predicting the delivery day can be found in the confusion matrix, and on the right side, the ROC curve is displayed.

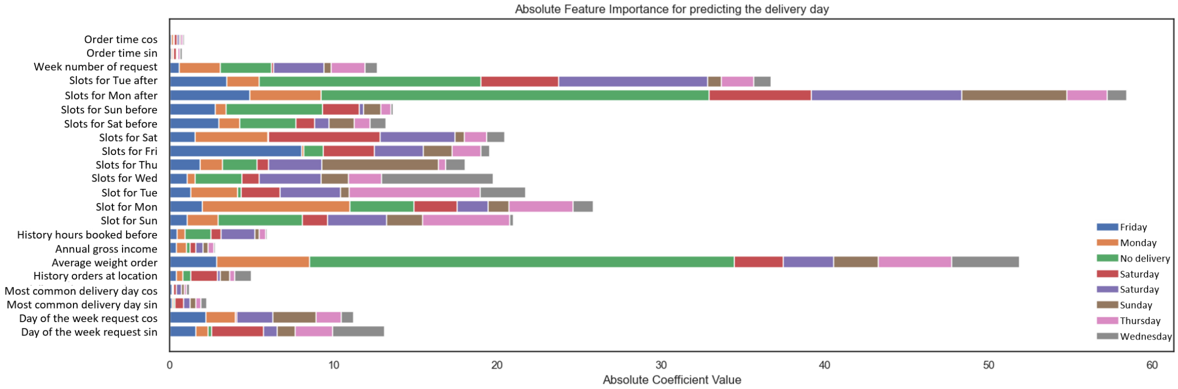


Figure 5.6: Summary of feature importance using the MNL model for predicting the delivery day.

In order to evaluate the performance of the MNL model for predicting the delivery day, the F1-score is computed and compared with the Benchmark model. The F1-score for the MNL model is 0.743, which is higher than the F1-score of the Benchmark model. This suggests that the MNL model improves performance and is valuable to the prediction process. Furthermore, the accuracy of the MNL model is also higher than that of the Benchmark model, indicating a higher percentage of correct predictions. Nonetheless, the accuracy cannot be used for comparing the models as the data used for predicting is imbalanced. However, it is worth noting that the MNL model tends to predict a large number of lost orders and a high proportion of requests that are not completed, as shown in the confusion matrix. Despite this, the model performs well in correctly predicting the delivery day when it makes a prediction. As previously mentioned, the model may struggle to distinguish between two consecutive days. The ROC plot, located on the right side of Figure 5.5, suggests that the No delivery output is the easiest to distinguish, while the deliveries for Friday are the most challenging. Since the No delivery classifier line is closest to the top-left, the line representing the Friday classifier is the furthest away. In addition to providing insights into the model's predicting performance and understanding which features are important for predicting the delivery day, probability density plots can offer a better understanding of how the probabilities are distributed. In Figure 5.7, two probability plots are displayed because the high density of the No delivery label makes the other distributions difficult to see. A second plot on the right side is included to provide a more detailed overview of the probabilities for the executed orders.

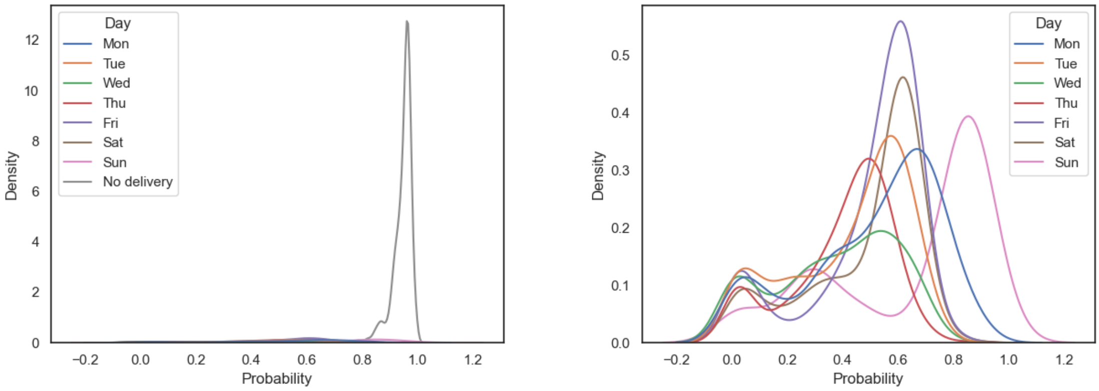


Figure 5.7: Probability distribution for if the order is lost or not. When the order is not lost, the probability of the delivery day is given. The upper and lower bounds were checked and found to be 0.999 and 0.000, respectively, to check for any probabilities that are not feasible.

As shown in Figure 5.7, the dataset contains significantly more instances labelled as No delivery compared to other labels. Moreover, the figure reveals that the model assigns a high probability to the prediction that the order was lost, indicating a high degree of confidence in the No delivery label. Examining the density plot of executed deliveries, it becomes evident that the peak for Sunday deliveries is the farthest to the right, suggesting that the model is most confident in its predictions for this class. However, it is important to note that despite the peak of all classes being on the right side, this does not necessarily indicate perfect performance or accuracy. This is because the class with the highest probability is chosen in this study, and it is still possible that there was a class with a higher probability than the correct one.

5.1.3. Prediction of part of delivery day

The final step in the prediction process involves estimating the specific day part of when the delivery will occur. Once again, the model's performance is compared with and without class weights, and the model with the class weights achieves a higher F1-score. Thus, the class weights will be used to predict the delivery part. The prediction results are presented in Figure 5.8, which includes the confusion matrix on the left and the ROC plot on the right. The accuracy of predicting the delivery time is 0.404, indicating that 40.4% of the predictions are correct. To provide more insights, the feature importance for all five categories can be found in section G.1, which displays the assigned weights to the features per category. Based on these plots, it is clear that the number of slots available for the delivery time has a highly positive impact. Furthermore, it has been observed that an increase in the number of evening and night time slots has a negative effect on the likelihood of selecting early and morning delivery slots. Conversely, for evening and night deliveries, the opposite trend is observed. Another noteworthy finding in the feature plots is that historical information has a greater influence on predicting the delivery time. To aggregate and visualise all possible information from the feature importance plots, a summary plot from the absolute values of all classes is made and displayed in Figure 5.9. By means of this plot, it can be found the number of available time slots in the morning and eve are mainly important.

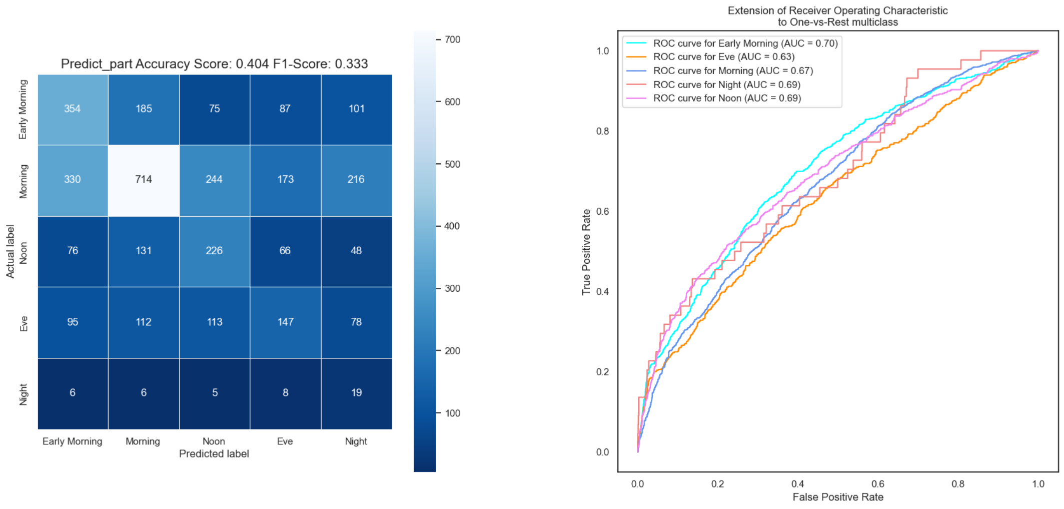


Figure 5.8: On the left side, a summary of the results for predicting the part of the delivery day can be found in the confusion matrix, and on the right side, the ROC curve is displayed.

On the left side of Figure 5.8, the confusion matrix summarises the prediction results and shows the number of correctly predicted categories. When the model predicts the wrong category, the correct label is also provided, offering insights into the relationship between different labels. It can be observed that the MNL model struggles to predict the correct parts of the delivery day, and no clear pattern can be found in the confusion matrix. This indicates that the model has difficulty distinguishing the parts of the day, and a possible explanation could be that the available features do not contain enough information to predict the outcome in this step accurately. The model's F1-score for this step in the prediction process is 0.333, more than double the Benchmark model's F1-score, but the accuracy and thus the number of correctly predicted labels are decreasing due to the imbalanced dataset, where the number of morning deliveries is almost half of the dataset, as shown in Table 4.1. The ROC plot on the right side of Figure 5.8 shows that none of the performances is located on the top-left of the figure, and all are

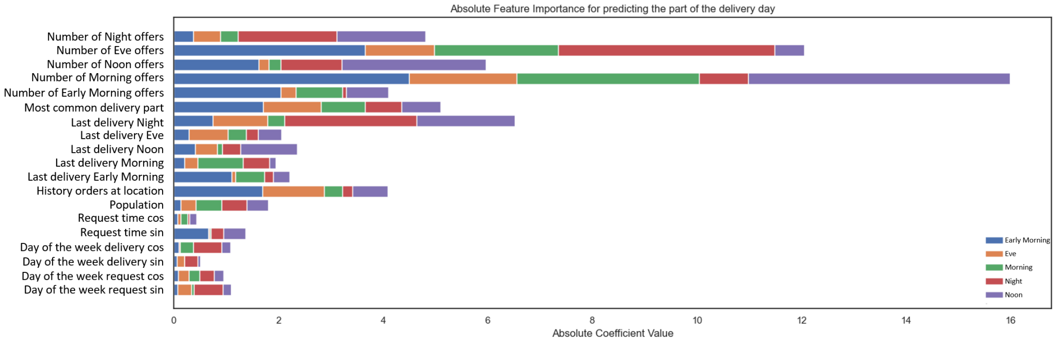


Figure 5.9: Summary of feature importance using the MNL model for predicting the part of the delivery day.

in the same area. This suggests that the model is not accurately differentiating between the various classes, as it only slightly outperforms a model that makes random choices. These outcomes raise another potential explanation for the model's poor performance: the classes may be too comparable.

In order to gain a better understanding of the model's performance, Figure 5.10 displays a probability density distribution. This figure reveals that none of the class peaks is on the right side when the plot is divided at 0.5, implying that the model cannot effectively distinguish the correct outputs, which aligns with the observations based on the confusion matrix and ROC plot. However, it can be found that the model is, in some instances, more confident since there are smaller peaks for the early morning, noon and eve between 0.6 and 0.7.

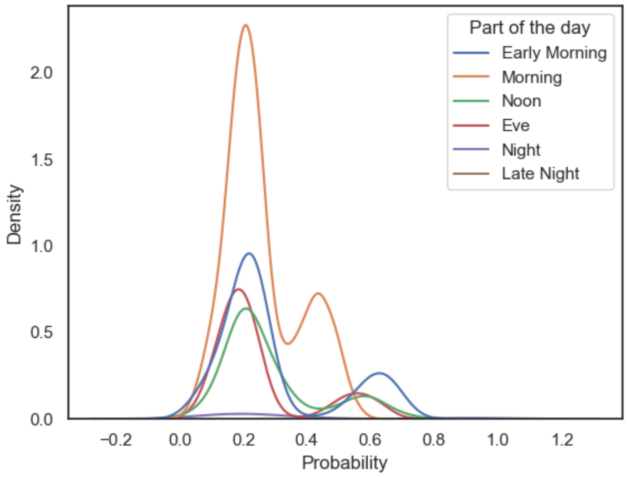


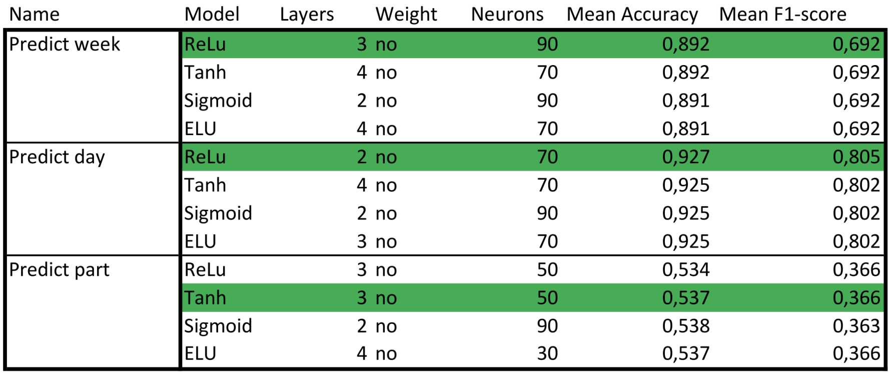
Figure 5.10: Probability distribution for the parts of the day. To ensure that the model does not contain probabilities that are not feasible as indicated due to smoothing, the lowest and highest probabilities are indicated and are 0.001 and 0.954, respectively.

In summary, the results of the hyperparameter tuning indicate that the MNL model with the newton-cg solver performs best across all three prediction stages, based on the F1-score. The F1-score is used for comparison due to the imbalanced nature of the data, which renders metrics such as accuracy unsuitable for evaluation. However, it should be noted that the three models differ in their architecture for predicting the week, part of the delivery, and day, with class weights incorporated only in the first two models. Once the optimal configuration for the models was selected, predictions were made. A series of performance measures, such as confusion matrix, ROC, feature importance, and probability plots, were generated to provide more insight into the model's behaviour. The results suggest that the most important features for predicting the delivery week and day are the week the request was made and the average weight of the order. In contrast, the customer's historical information is the most significant feature for predicting the part of the delivery day. Nevertheless, the number of available slots is an essential feature for predicting both the day and part of the delivery day. In conclusion, the prediction performance of the MNL model is superior to that of the Benchmark models, as determined by the F1-score. Thus, it is recommended to use the more advanced MNL model for prediction. However, it is worth noting that the MNL models for predicting the week and day perform better overall than the MNL model for predicting the delivery moment, as evidenced by their higher F1-scores.

5.2. Neural Networks

The second advanced model employed in this study for predicting the delivery week, day, and part of the day is the non-parametric NN model. Before performing the NN, various characteristics are tested during the hyperparameter tuning, including activation functions, the number of hidden layers, and neurons in these hidden layers. As this study involves the NN performing multi-class predictions, the Softmax activation function is selected for the output layer, as described in section 3.4. However, to determine the rest of the architecture, all prediction performances of the hyperparameter tuning need to be considered. A detailed overview of the accuracy and F1-score indicating the prediction performance of these tests can be found in section E.2. According to the results, the models used in the prediction process do not perform best when sharing similar characteristics and architecture. Specifically, the ReLu solver shows the highest performance for predicting the delivery week and day, while the Tanh solver performs best for predicting the delivery day based on the F1-score.

Furthermore, the number of hidden layers and their width varies among the models. For instance, the NN model used to predict the delivery week has three hidden layers of ninety neurons, while the model for predicting the part of the day has three layers consisting of fifty neurons. Conversely, a different architecture consisting of two hidden layers with seventy neurons is optimal for predicting the delivery day. Interestingly, all three models perform better without incorporating the balanced class weights. A summarised overview of these results is presented in Table 5.2.



Name	Model	Layers	Weight	Neurons	Mean Accuracy	Mean F1-score
Predict week	ReLu	3	no	90	0,892	0,692
	Tanh	4	no	70	0,892	0,692
	Sigmoid	2	no	90	0,891	0,692
	ELU	4	no	70	0,891	0,692
Predict day	ReLu	2	no	70	0,927	0,805
	Tanh	4	no	70	0,925	0,802
	Sigmoid	2	no	90	0,925	0,802
	ELU	3	no	70	0,925	0,802
Predict part	ReLu	3	no	50	0,534	0,366
	Tanh	3	no	50	0,537	0,366
	Sigmoid	2	no	90	0,538	0,363
	ELU	4	no	30	0,537	0,366

Table 5.2: Performance overview of best configuration per solver for all three NN models. The used solver with the best performance is indicated in green for each prediction step.

5.2.1. Prediction of delivery week

For the NN, the same prediction steps will be carried out as for the Benchmark and MNL model. Accordingly, the NN model will first predict the week of delivery. After conducting the hyperparameter tuning, it is determined that the optimal performance for predicting the delivery week is achieved without incorporating balanced class weights, with a combination of the ReLu activation function and three hidden layers comprising ninety neurons each. The results of the NN model for predicting the delivery week are presented in Figure 5.11, where the confusion matrix can be observed on the left and on the right side a plot depicting the losses and accuracy. The right plot provides insights into the learning process of the NN and can help identify overfitting. To minimise the likelihood of overfitting, all NN models employ a dropout function with a probability of 0.2, meaning that 20% of the neurons are randomly set to zero during training. When a neuron is turned to zero, it means that the contribution of this neuron is temporally removed, making the network less sensitive to the specific weight. In addition to the dropout function, an early stopping method is also implemented in the models. The early stopping method is used as it specifies many epochs that prevent underfitting but stops the training if the model's performance starts to degrade on the validation data set. Suppose the dropouts and early stopping methods are not used to prevent overfitting. In that case, the model can stop generalising and start learning the statistical noise during the training, making the model less useful for new predictions on new data.

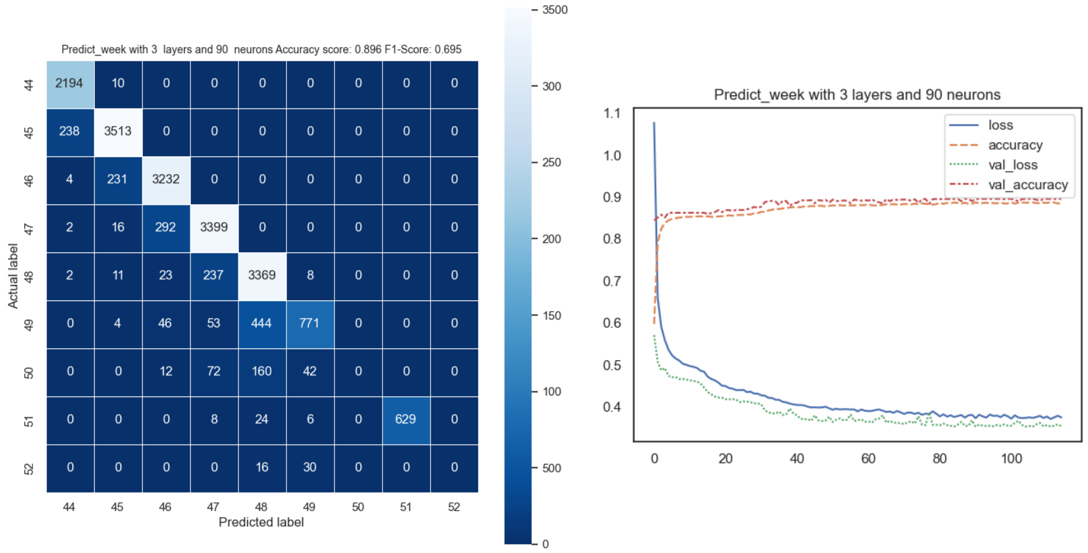


Figure 5.11: On the left side, a summary of the results for predicting the delivery week with the NN can be found. On the right side, the losses and accuracy are displayed, which can be used to determine whether the model is overfitting. The number of utilised epochs during the training process can be found on the x-axis.

Based on the confusion matrix presented in Figure 5.11, a clear pattern emerges in the early weeks, while this is not the case in the later weeks, and the model fails to predict deliveries in weeks 50 and 52. However, the NN model's results are less scattered than the MNL model's, resulting in an accuracy score of 0.896, indicating that the model correctly predicts 89.6% of the data. A possible explanation of why the NN performs better in the first weeks is due to the coming Christmas period and the number of data points for the weeks after week 49. For the first weeks, the model tends to predict the week before the actual delivery week, and to address this issue, two days before and after the selected week can be included during the prediction. The two days after the delivery week can also be included for consistency between the NN and MNL and in the case that the NN might predict the later week. However, due to the data used in this research, no extra days are added in the next prediction step as it does not add value here. To get an understanding of how a NN model works, SHAP values[1] are used to identify the important features with the use of plots, these plots are created for all the different output categories and can be found in subsection G.2.1. In these detailed individual SHAP plots, it has been determined that the most significant feature is the week the request is made. Lower week numbers positively impact the probability of deliveries in the early weeks, while higher week numbers positively influence the probability of later delivery weeks. The second most important feature is the average weight order. When comparing these features with the most important features for predicting the delivery week with the MNL model, it can be found that the same features are indicated as most influential on the prediction. A summary plot of the absolute average SHAP values among all classes can be found in Figure 5.12. This plot provides an overview and confirms that the week number of requests is the most important feature, followed by the average weight. In contrast, the NN model barely utilises the request time cos, annual gross income, and request time sin features, as their values are very

[1] SHAP values which are used to create plots and are calculated with the use of the marginal contribution of a feature value to a given model and, as a result, provide the overall effect on the model.

low compared to the other features. Since all output classes have the same important features, this could indicate that these features are highly informative for predicting all classes or that the classes are closely related and have similar underlying characteristics. The NN model's F1-score is 0.695, which is higher than the scores of both the MNL and Benchmark models, and suggests that the NN model enhances performance compared to the other two models. The loss and accuracy metrics on each epoch in Figure 5.11 indicate that the model is not overfitting and is learning quickly.

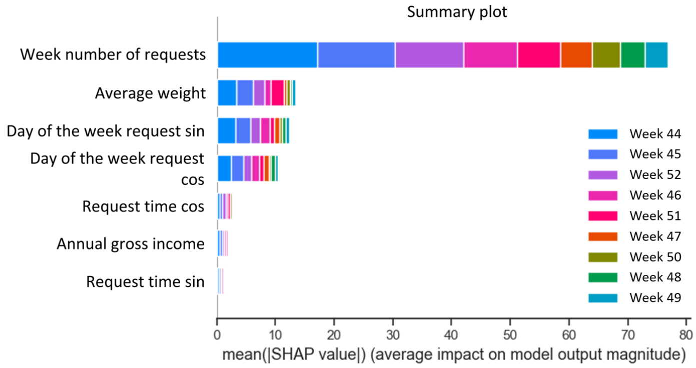


Figure 5.12: Summary of SHAP values indicating the feature importance for all weeks.

In addition to the insights gained through the SHAP values, the probability density plot can also provide valuable information about the presumed black box model. The distribution of probabilities assigned to the various output classes can be observed in Figure 5.13. Similar to the density plots for the MNL model, the densities in this plot are smoothed, resulting in non-zero densities for prediction probabilities below zero and above one. Based on the probabilities assigned by the NN model, it can be observed that weeks 44 through week 48 have high peaks on the right side of the plot, indicating that the model is confident in assigning these labels. Comparing these probabilities with those of the MNL model, a smaller decline in confidence over the weeks can be observed. In other words, the confidence of the NN model in predicting later weeks is higher than that of the MNL model. This difference is also reflected in the confusion matrices, as the results for the NN model are less scattered. Nonetheless, the NN model has a hard time distinguishing later weeks, as there is still a peak for week 50 on the left side of the plot and not even visible for week 52. Therefore the performance can still be improved.

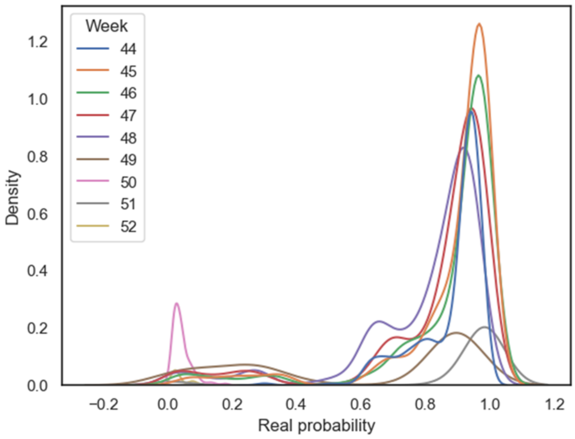


Figure 5.13: Probability distribution of the NN for predicting the delivery week. To ensure that the model does not contain probabilities that are not feasible as indicated due to smoothing, the lowest and highest probabilities are indicated and are 0.000 and 0.998, respectively.

5.2.2. Prediction of delivery day

The second prediction model will estimate the delivery day and comprises two hidden layers of seventy neurons each, in combination with the ReLu solver. These model characteristics are selected based on the F1-score obtained from the hyperparameter tuning results shown in Table 5.2. The model does not contain imbalanced class weights since it performs better without them. In this step of the prediction process, the multi-class NN consists of eight different output classes representing the seven weekdays and the possibility that the order is lost. The results of predicting whether the delivery will take place and, if so, on which day are shown in Figure 5.14. In Figure 5.14, the output results are summarised in a confusion matrix on the left side, while on the right side, the accuracy and loss metrics on each epoch are visualised. The accuracy and loss metrics provide insight into the model's training process, and based on this plot, it is found that the model learns quickly and does not require too many epochs. Additionally, it is found that the accuracy decreases rapidly but not at the same rate as the training set. After a quick drop, it remains stable for several epochs, indicating that the model generalises well to unseen data.

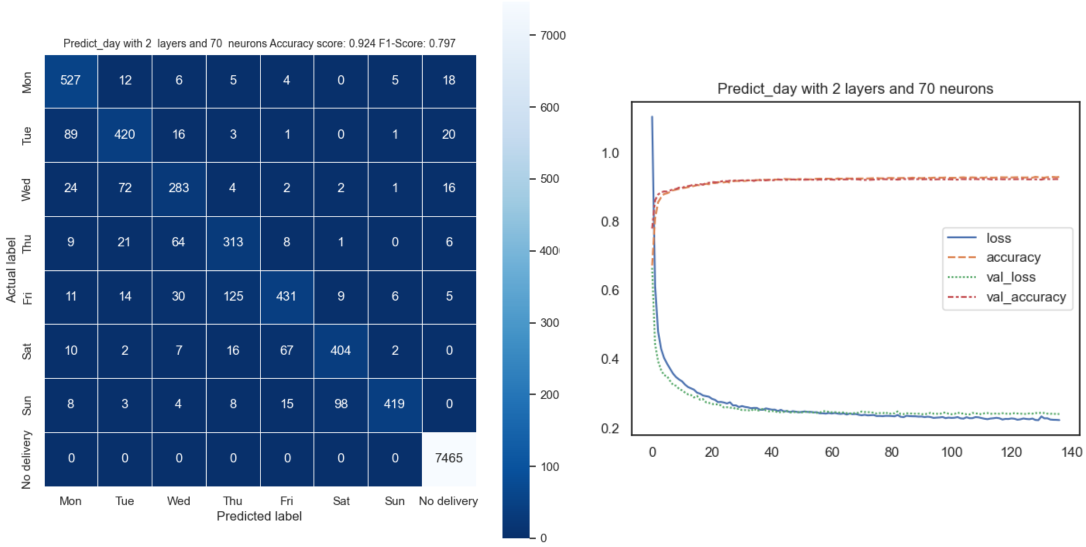


Figure 5.14: On the left side, a summary of the results for predicting if the delivery is occurring, and if so, the predicted day can be found. An overview of the learning rate and if the model is overfitting can be found on the right side. The number of utilised epochs during the training process can be found on the x-axis

The model is trained with a dropout rate of 20% in combination with the early stopping method to minimise the likelihood of overfitting. Analysing the learning performance of the model on the right side of Figure 5.14, reveals that the NN model for predicting the delivery day is not overfitting. The confusion matrix displayed on the left side of Figure 5.14 indicates that many requests are predicted as lost orders, labelled No delivery. If it is predicted that the delivery will occur, a day is also indicated. Since the confusion matrix summarises the results, it provides insights into the model and shows that when the delivery day is not well predicted, it tends to predict the day before as the delivery day. Despite this, the accuracy of the model is 0.924, indicating that 92.4% of the data is predicted correctly. Furthermore, when comparing the F1-score of the NN with the scores of both the Benchmark and the MNL, it is found that the NN model with an F1-score of 0.797 is improving the prediction performance. To get more insight into the behaviour of the NN model predicting the delivery day, a detailed overview of the SHAP values can be found in subsection G.2.2. These plots indicate the feature importance for all predicted delivery days and the No delivery label. Based on these detailed SHAP plots, it is found that the average weight of the order is the most important feature, followed by the day of the week and the number of available slots for the predicted delivery day. A summary plot of the absolute feature importance for all classes is displayed in Figure 5.15 to provide a total overview. Combined with the detailed SHAP plots, this summary plot shows that not all output classes have the same most influential features, suggesting that features' impact on model output depends on the class label. This indicates that each class may have distinct characteristics that are essential to consider. Comparing the feature importance of the NN model with that of the MNL model reveals that the former assigns greater importance to the average weight order and the week of the request. At the same time, the latter prioritises the number of available slots for the delivery day.

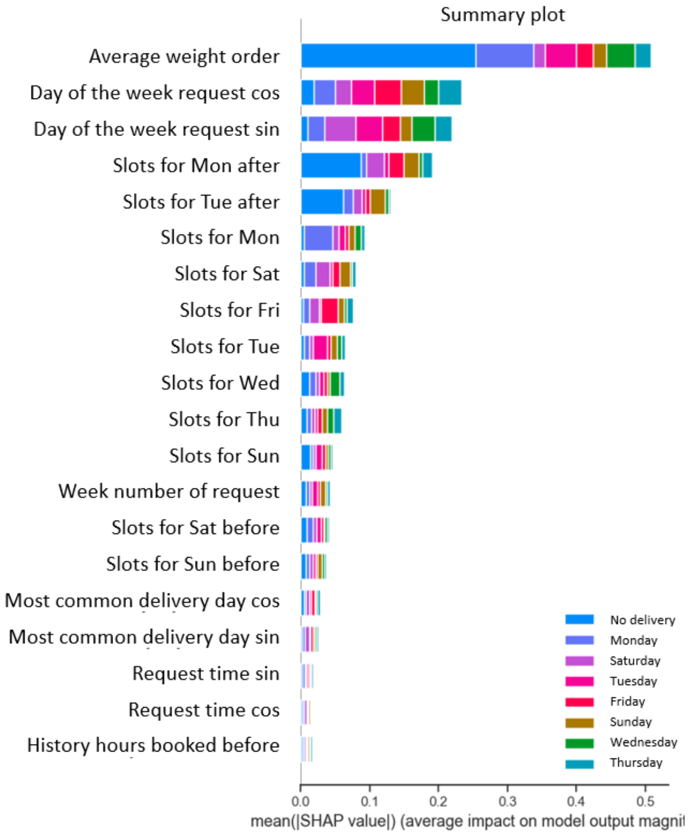


Figure 5.15: Summary of SHAP values indicating the feature importance for all weekdays

To gain further insights into the NN's behaviour, the probability density distribution of the different output classes is obtained and presented in Figure 5.16. This figure includes two probability plots because the No delivery label's high probability density makes it difficult to see the other distributions. Therefore, the figure includes a probability density plot on the right side without the No delivery label. Based on both plots, different results can be observed. For instance, the left plot reveals that the NN model is highly confident in predicting the No delivery label, as the peak is almost at one and very narrow. However, the right plot indicates that, except for the No delivery label, all other labels have the largest peak on the right side, indicating that the model is confident in predicting those labels. Nonetheless, the NN model's predictions are not perfect yet; as a result, some smaller peaks can be found on the left side, indicating that the model was not completely confident about the correct output label.

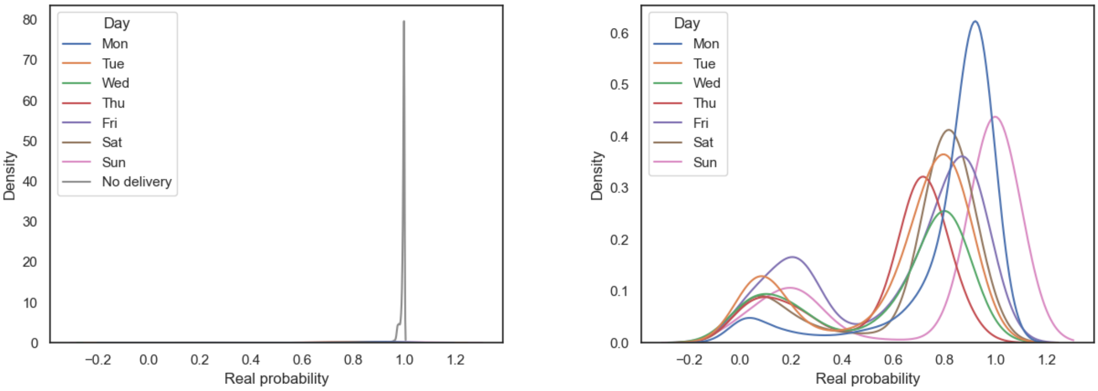


Figure 5.16: Probability distribution for if the order is lost or not. When the order is not lost, the probability of the delivery day is given. The upper and lower bounds were checked and found to be 1.000 and 0.000, respectively, to check for any probabilities that are not feasible.

5.2.3. Prediction of part of delivery day

The final step in identifying customer choice behaviour is to determine the features that help predict the part of the delivery day using a NN model. The NN model used in this study consists of three hidden layers of fifty neurons and does not consider the balanced class weights. In contrast to the prediction models for the week and day, this model uses the Tanh activation function in the hidden layers. This described model architecture is selected after comparing the results of the hyperparameter tuning, as presented in Table 5.2. The results of predicting the part of the delivery day can be found in Figure 5.17, where the confusion matrix is delayed on the left side. The plot indicates that the results of predicting the part of the delivery are scattered. In addition to the confusion matrix summarising the model's results, it also indicates the accuracy value, which for this model is 0.539, meaning that 53.9% of the instances is predicted correctly.

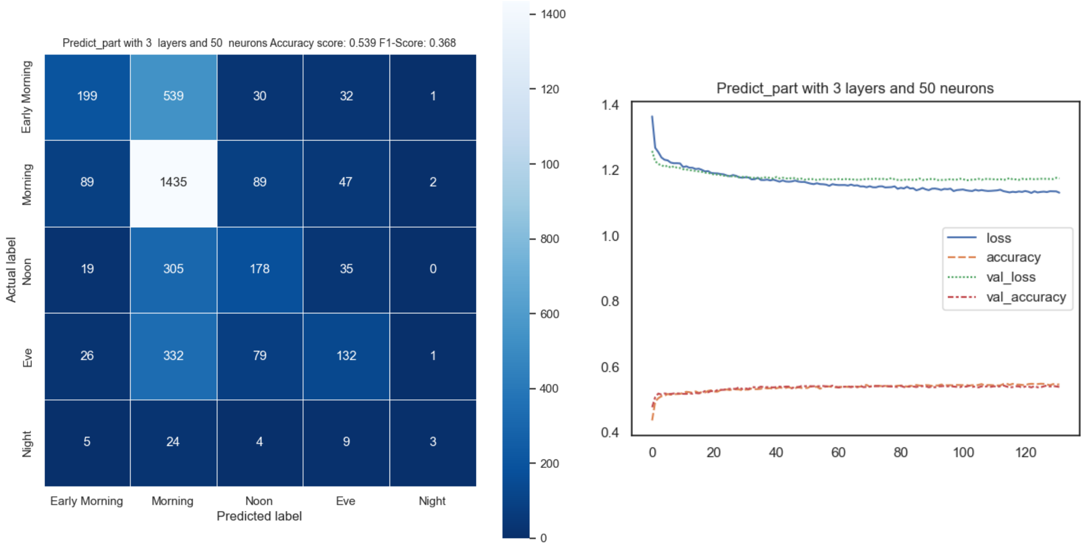


Figure 5.17: On the left side, a summary of the results for predicting the part of the delivery day. On the right side, the learning curve of the NN model for predicting the part of the delivery day can be found with on the x-axis the number of utilised epochs during the training process.

The right side of the Figure 5.17 shows the learning performance of the model, which confirms the suspicion of the model's performance obtained through the confusion matrix and performance metrics. Based on the loss and accuracy metrics in this plot, it can be found that the model learns fast for a short time, but stops learning quickly, suggesting that the model is stopped to prevent overfitting before it is able to improve its learning further since it is slowly increasing. The NN model's inability to reduce losses and improve accuracy may be due to the inadequate predictive power of the selected features for predicting the part of the delivery day. The data used may be too random and not sufficiently related to the output, making it challenging for the model to distinguish between different output classes. Despite the low F1-score of 0.368, compared to the Benchmark and MNL model scores, the NN model improves the prediction performance. To gain more insight into the behaviour of the model, the feature importance weights are obtained, and a detailed overview per output class can be found in subsection G.2.3. The feature importance analysis reveals that historical information significantly influences predicting parts of the delivery day. For instance, a feature containing historical information about the last delivery moment is essential, as customers often select the same time slot as they did last time. A summary plot of the average impact of each feature can be found in Figure 5.18, providing an overview of feature importance.

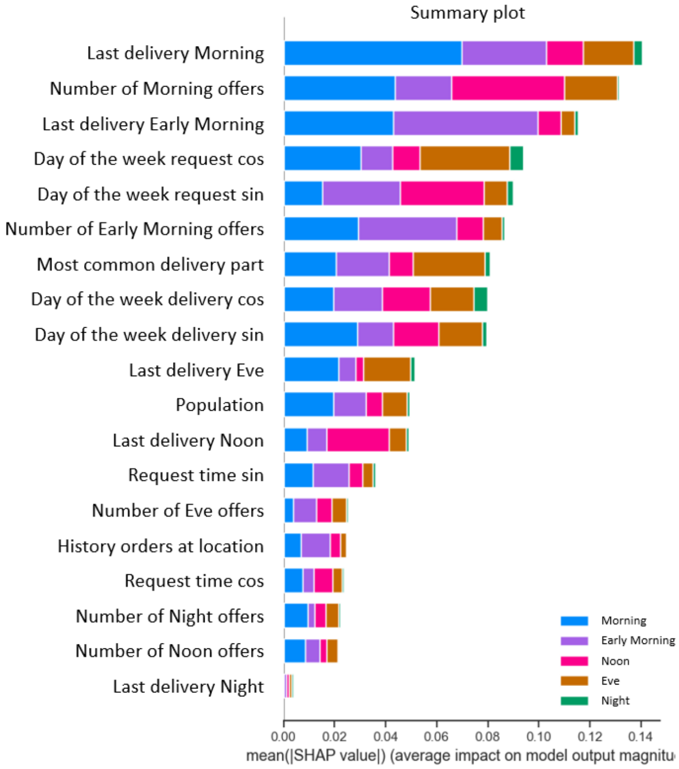


Figure 5.18: Summary of SHAP values indicating the feature importance for all parts of the day.

When examining the summarised SHAP values of the NN model predicting the part of the delivery, it becomes evident that factors related to the morning play a crucial role. Specifically, the most significant general feature is historical information on whether the last delivery was in the morning, followed by the number of available morning time slots. As previously mentioned, morning time slots are popular; thus, it is noteworthy that the day of the week when the request is made is also significant.

A probability density plot is generated using a kernel estimate plot with smoothed densities to gain additional insights into why the model produces scattered results and struggles to distinguish between different classes. However, since the smoothing technique can lead to probabilities that exceed the possible range of values (i.e., below zero or above one), the lowest and highest probabilities are indicated to ensure that the model's results are presented in Figure 5.19 are plausible.

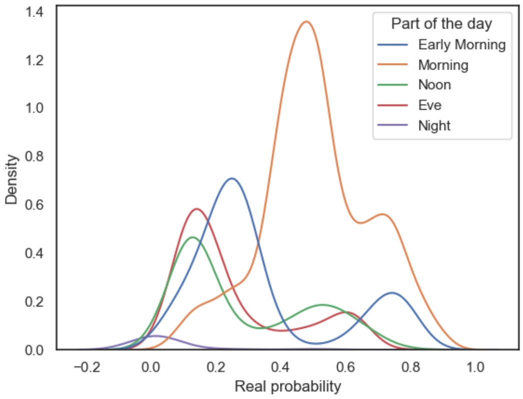


Figure 5.19: Probability distribution for the different parts of the day as output classes. The lower and upper bounds were checked and found to be 0.000 and 0.919, respectively, to check for any probabilities that are not feasible.

Visualising the probability densities can provide additional insights into the model's performance. When a plot shows narrow and high peaks around one, it suggests that the model is highly confident in predicting that output class. However, upon analysing Figure 5.19, it becomes evident that none of the output classes has a narrow peak around one. This confirms the reason behind the model's scattered results. The model is most certain when predicting that the delivery will take place in the morning. In contrast, for predicting the other labels, the NN model lacks certainty, resulting in peaks located towards the left side of the plot.

In summary, the evaluation of the NN models predicting delivery week, day, and part of the day indicate that the NN improves prediction performance compared to the Benchmark and MNL models based on F1-score. However, it was observed that the three models have varying architectures and activation functions in their hidden layers and different hidden layer widths due to hyperparameter tuning. The architecture and activation function with the highest F1-score was selected during tuning, resulting in the use of both ReLu and Tanh activation functions in this study. Nevertheless, all three models utilise the Softmax function in the output layer due to the problem's multi-class nature. Providing insights into the working of NN models, different performance measures, such as confusion matrices, probability densities, and SHAP plots indicating the feature importance, are used to provide information instead of treating the models as black boxes. In addition, the accuracy and losses of the data and validation data can also aid in detecting overfitting. However, these visualisations are mainly used to confirm no overfitting is taken place since the models use a 20% dropout layer and an early stopping method, both employed to minimise the likelihood of overfitting. To determine which features have the most significant influence on the probability of selecting the output class, SHAP values are created. Based on these SHAP plots, it is possible to provide a better understanding, and it is found that the average weight and request time are deemed as important for predicting the delivery week and day. For predicting the part of the delivery day, historical customer information is primarily important since it is found that the last delivery moment significantly influences the selection procedure of a new moment. Next to the historical customer information, the number of available time slots is considered as important.
Since it is found in the SHAP plots visualising feature importance of predicting the part of the delivery day that historical information might be essential to improve the prediction, section H analyses the influence when the models are used only for customers who have ordered before. The influence of historical information is tested as storing more historical data requires much capacity. However, based on the results from section H, it can be found that the prediction performance for both the MNL and NN models predicting the part of the day increases. In contrast, both models that predict the delivery day experience little impact, despite this may be explained by the fact that few features in this prediction step contain historical information. Nonetheless, these results make it interesting to research the options to include more historical data and how it can be acquired, especially to achieve better performance for predicting the part of the delivery and to see the possible improvements in the day prediction step.

5.3. Summary

Before the more advanced MNL and NN models can start performing the three prediction steps, hyperparameter tuning needs to be done. During the tuning of the model, different solvers and the incorporation of balanced class weights are tested and compared. In addition, a different number of layers and neurons will also be tested for the NN model to find the best architecture. After the hyperparameter tuning, the performance results can be compared using the F1-score since all three data sets are assumed to be imbalanced. Based on the F1-scores, it is found that the best performances of the three MNL models are obtained with the newton-cg solver. However, the models for predicting the delivery week and part of the delivery day will include the balanced class weight, whereas the model used for predicting the delivery day not. For the NN models, not all prediction models use the same activation function in the hidden layers; for predicting the delivery week and day, the ReLu function will be utilised and for predicting the part of the delivery day, the Tanhs function. Nevertheless, it is found that none of the NN models uses balanced class weights as it does not add value to the performance. Dropouts and an early stop function are included during the hyperparameter tuning and prediction process to prevent overfitting.

Once the solvers and model architecture are defined, predictions can be made and results obtained. The MNL and NN models are evaluated using the F1-score, which allows for performance comparison. Based on this score, it can be concluded that the NN model performs better than the other models. However, the MNL model outperforms the Benchmark model, indicating that both the advanced MNL and NN models improve prediction performance and can better identify customer behaviour. Further analysis of the models reveals that they have a different order of feature importance for predicting the delivery day, with the MNL model emphasising the number of available slots. In contrast, the NN model prioritises the average weight of the order. Both models also differ in prioritising the features in the final prediction step. The MNL model gives higher weights to the number of available slots, while the NN model emphasises the features containing history. Notably, both models have the same order of feature importance when predicting the delivery week.

The MNL and NN models are expected to assign different weights to the same features, even when trained on comparable data, as both models have different assumptions and learning mechanisms. The MNL model, for example, presumes that the relationship between the independent and dependent variables is linear, whereas a NN model can also model non-linear relationships. Furthermore, NN models can automatically learn features through hidden layers, while MNL models necessitate pre-defined features. Similar results are acquired when the models are trained with only data from customers who ordered before to determine the influence of historical information on the prediction performance. Nevertheless, the mean reason for performing the models with the new data set is to decide if it provides added value to the prediction performance to gather and use more historical data, as it might be the case that both models underestimated the importance of historical features. This is assumed as the MNL and NN models typically assign weights to features based on their ability to predict the outcome of interest. Because this study only has data from November, historical information on most customers is not available, resulting in the models might rely more heavily on features that are consistently available across all customers. However, based on the executed models, it is found that both models improve their performance, especially for predicting the part of the delivery day. That the performance of predicting the delivery day remains almost the same may be due to the lack of historical features in the process. According to these results, gathering and including more historical customer information in the models is assumed to lead to better prediction performances.

6 Simulation

The previous sections in this study have demonstrated that more advanced predictive models can give a better understanding of customer behaviour than the Benchmark models and can identify the most critical customer characteristics. The Benchmark models serve as a reference point for evaluating the effectiveness of the more advanced models in predicting customer behaviour. These baseline models are derived from assumptions found in the data and help ensure that the more complex models add value without requiring excessive computational resources. However, it remains to be determined whether the more advanced MNL and NN models with improved predictive performance add value and truly enhance the simulation closer to reality or whether the results are similar to the Benchmark model. In essence, does the learned behaviour impact the routes and number of offered slots, and if so, could it be utilised to optimise the provided set of offers? A simulation is conducted using the time slots selected by real customers to assess these discrepancies from reality. The outcomes will result in KPIs that the simulator collects to assess its performance. Nevertheless, in AHD for the grocery sector, the KPIs do not have one optimal outcome as it depends on the company's strategy for which trade-offs must be made. Furthermore, the KPIs should be considered together as each provides valuable insights into different aspects of the outcome, and a single KPI does not provide a complete picture of the results.

6.1. Outline

In this study, ORTEC's events-based simulation tool consisting of time slotting and routing optimisation processes will be deployed in combination with an instance generator and event simulator. The instance generator typically generates customers based on behavioural assumptions and arrival distribution. Since real customer data from an online grocery company is available in this study, the real arrival permutation and amount of customers will be used in the instance generator. Instead of the behavioural assumptions, the Benchmark, MNL and NN will individually determine each customer's behaviour during the separate simulations. The event simulator will sequence the arrival of all created customers from the instance generator since the customers and their actions will be a separate event with a timestamp. As the event simulator represents a set of entities that interact with each other by generating and processing events, the different timestamps will be processed in chronological order. Between two consecutive occasions, it is assumed that no changes occur in the system, allowing it to jump to the next event directly. Event-based simulators are particularly useful in this type of problem where the timing of events is important and allows for more efficient use of computer resources. Because the simulator only processes events when they occur, it can avoid wasting computational resources on idle periods.

The event simulator not only lists subsequent events based on timestamps but also collects the companies' KPIs to assess the modelled system's performance and identify areas for improvement. These KPIs can provide decision-makers with valuable insights into the system's operation and help them make informed decisions to optimise it. However, it is important to compare all the KPIs together as each one provides valuable insights into different aspects of the simulation, and a single KPI cannot provide a complete picture. The KPIs used in this study's simulations include:

- Number of offered time slots per customer
- Total executing time of the routes
- Number of assigned people

Based on these KPIs, different simulations can be performed and compared to determine the impact of implementing customer choice models. A schematic overview is illustrated in Figure 6.1 to explain better how the simulation process works. This figure shows that the instance generator and event simulator are deployed first, after which the first event will be indicated. The first timestamp will always be the action that the customer arrives in the process, after which an offer request event is made. This request will be sent to the time slotter that will create an offer set which will be sent back to the customer labelled as an offer response. Now the customer can select a time slot from the provided offer set. Instead of predefined customer behaviour that can be used to choose a time slot, the customer choice models will be employed. In this step, the trained customer choice models are implemented in the simulation tool and either return the selected part of day or indicates that the order is lost. After selecting the final result, the customer leaves the process if it is predicted that the order is lost or the customer selects one of the preferred time slots for the given part of the day. The time slots of a delivery day part will be randomly ranked since the actual window is one hour. Based on the preference list of the customer, the time slot will receive a booking request with this list, checks the availability, and selects the highest possible preference. The availability needs to be rechecked, as another customer may place an order during the selection, making the time slot unavailable. Routing optimisation will be done periodically during the simulation, and a final optimisation will be performed when the window is fully booked or the cut-off time is reached.

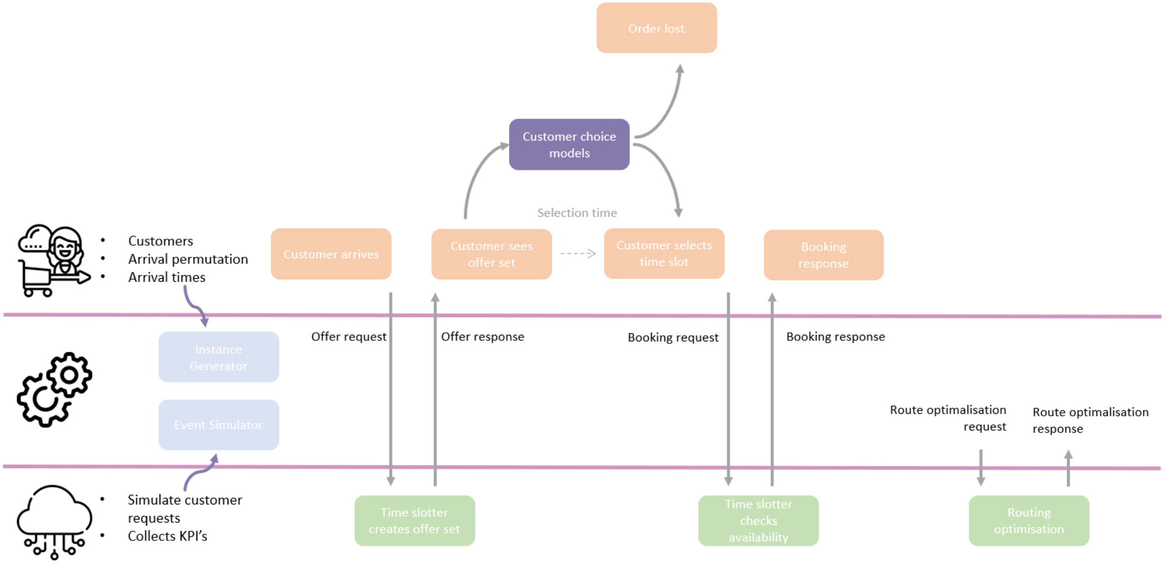


Figure 6.1: Overview of the used simulation process.

6.2. Performance evaluation

To evaluate whether the more advanced MNL and NN models truly enhance the simulation and add value, simulations are performed where the MNL, NN and Benchmark models are included instead of predefined customer behaviour. The KPIs of these simulations will be compared with the KPIs of a simulation conducted with real-chosen time slots based on the same simulation scheme. An enhanced simulation by the MNL and NN models indicates a closer alignment between their outcomes and the simulation results utilising real-selected time slots, as compared to the Benchmark model. The simulation with real-selected time slots is considered necessary because the time slotter or the route optimiser may be updated between the new simulations and November, affecting the KPIs. Therefore, the simulation with the real-selected time slots serves as a more reliable reference point for evaluating the impact of the choice models.

Before conducting the simulation it needs to be decided how to take the prediction uncertainty of the choice models into account. Since the more advanced used customer choice models are MNL or NN models, the returns are probability matrices that indicate the likelihood of each possible class label for a given input. To account for uncertainty in the prediction, the obtained probabilities will be used as probability weights for each class to sample the outcomes and generate a distribution. The most selected outcome after sampling it several times will be determined. In addition, it can help to reduce the impact of any individual prediction that may be incorrect due to random chance and to obtain a more robust prediction that is less sensitive to random fluctuations. Ensuring the sampling process with multiple samples is beneficial for predicting customer behaviour as sometimes customers do unexpected things and select another time slot. The sampling process has also been repeated once to account for unexpected behaviour and to analyse whether these results are closer to reality. This test is deemed necessary since the time slots with a small probability will likely never be selected in the previously mentioned sample method while the possibility still exist. For this reason, a modified sampling process was implemented, wherein a single random number was generated and used to select the corresponding time slot. In this way, the process ensures that time slots with a higher probability still have a greater chance of selection. The chance of selecting a time slot with a low probability is higher than the original method, though still lower than other time slots. Despite this, according to Figure 6.2, it is determined that sampling multiple times is preferable as it leads to more realistic results of modelled customer behaviour, which is concluded after combining the results of the three KPIs and comparing them to the values of the actual model's KPIs. For instance, it is observed in the leftmost plot, showing the average number of offered time slots, that the simulations where the probabilities are sampled multiple times produce results closer to the actual results than when the probabilities are only sampled once. Nevertheless, to determine which model is preferred, the average duration per route and the total number of assigned customers need to be evaluated similarly. After analysing all KPIs results, it can be concluded that sampling multiple times leads to more accurate and realistic outcomes. Therefore, it is decided to continue sampling the probabilities multiple times when employing the MNL and NN models.

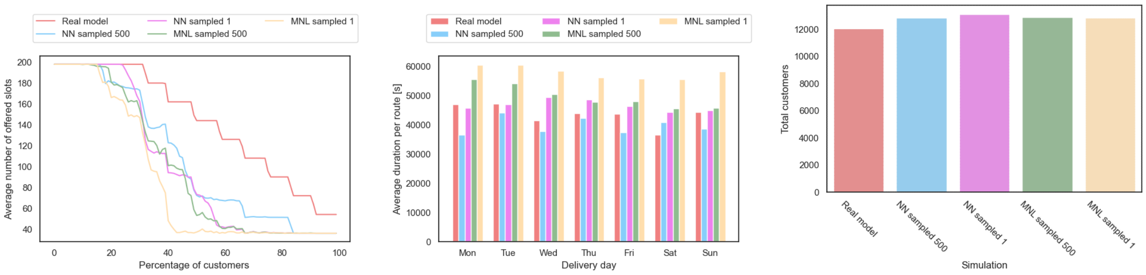


Figure 6.2: Comparison of the KPIs for sampling the MNL and NN model five hundred times and one time.

As described above, several simulations will be conducted using ORTEC's event-based simulation tool to evaluate the influence of the introduced customer choice models. First, a simulation with the real customer-selected time slots is performed to generate the KPIs for comparison purposes. Subsequently, the simulations using the Benchmark, NN and MNL models are performed. The performed simulations are compared using the same three KPIs used before the number of offered time slots per customer, the total execution time of the routes, and the number of assigned customers. An overview of these KPIs can be found in Table 6.1, where the costs are also displayed. The computation of the average route cost per customer in a routing problem involves the multiplication of the incremental cost of each route by the number of executed routes, followed by the division of the result by the total number of customers served in the routes. The execution costs of the routing problem include the cost per unit, cost per hour, and cost per order and grows with each additional customer leading to an increase in the overall execution costs of the routing problem. Based on these results, the different simulations will be compared, and it will be examined whether the results using the MNL or NN are further or closer to the simulation results with the real-selected time slots compared to the simulation using the Benchmark model. The comparison will be made using all outcomes to have the best possible overview since each provides valuable insights into different aspects of the outcome.

	Real model	Benchmark model	NN	MNL
Average number of seen slots	143.496	140.639	107.965	97.897
Average duration of the routes [s]	43496	15554	39671	49601
Total assigned customers	12047	7382	12897	12904
Average number of customers per route	30.678	17.494	22.968	23.014
Average duration per customer [s]	1417	893	1728	2154
Average route cost per customer	6.524	11.275	8.708	8.690
Average execution cost per customer	11.806	7.443	14.397	17.947

Table 6.1: Overview of the KPIs from the simulations performed with the real-selected time slots, Benchmark model, NN and MNL models.

The set of KPIs obtained from the simulations indicates that the MNL and NN models have a positive impact on the simulation and are effective in bringing it closer to reality compared to the Benchmark model. In order to explain this, the visualised results in Table 6.1 can be used. These indicate that simulations utilising the real-selected time slots or the Benchmark model offer more time slots on average. However, for the average duration of the route and total assigned customers, the simulation results utilising the NN and MNL models yield results closer to the actual results than the simulation using the Benchmark model. Additionally, it can be perceived that the routes generated based on the real-selected time slots are the most efficient regarding the average number of customers per route, while routes generated in the simulation using the Benchmark model are the least efficient. Despite the low number of customers per route, the simulation utilising the Benchmark yields the lowest duration per customer. This can be explained because more efficient routes can be generated since fewer customers are assigned, allowing for fewer stops and travel distance. Also the length of the time slot is a limiting factor during the optimisation process with more customers as there is less flexibility to create efficient routes. Another interesting finding is the cost per customer.
To start with the average routes cost, here it is found that the more people are on a route, the more customers share the cost of initialising a route, leading to lower costs per customer. However, the results for the average execution cost per customer can be found initially counterintuitive, as simulations with fewer customers on a route resulted in lower costs per customer. There are multiple reasons for this, and despite the higher cost it is also possible that the route is more profitable. The first explanation for the higher cost is that the variable costs do not occur on a pro-rata basis, which means that the cost does not increase linearly. This occurs because the cost structure includes rates that depend on various factors such as order size, execution time, and travel distance to the customer, all of which are dependent on the resources required to serve each customer. Therefore, it is also directly related to the higher average duration per customer and the fact that less efficient routes can be created. A reason why the route can still be profitable is that the operating margin [1] might be positive allowing for a higher profit when more customers are assigned. Nevertheless, since the profit depends on the realised revenue, costs and fees, this needs to be known first before it can be determined whether assigning more or less customers is advantageous. In addition, it is possible that a grocery store is willing to accept losses and only assigns more customers for strategic reasons to remain competitive and retain customers as a worthwhile investment in terms of both revenue and customer satisfaction.

Based in the table, a comprehensive summary of all the outcomes is presented, enabling a comparison of the KPIs to conclude that both the MNL and NN models enhance the simulation in comparison to the Benchmark model. However, these results do not offer insights into the distribution of the results and to obtain more information the various KPIs will be discussed in more detail. To start with the average number of time slots options offered per day per customer percentage is displayed in Figure 6.3.

[1] Operating margin = $\frac{\text{Revenue - Cost of goods sold - Operating expenses}}{\text{Revenue}}$

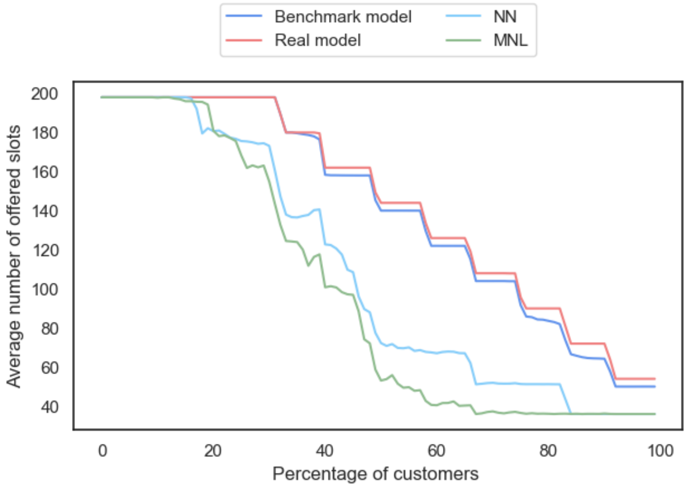


Figure 6.3: This plot indicates the average number of offered times slots per customer per cent.

This plot confirms that the simulations using the real-selected time slot and the Benchmark model offer the most time slots on average. However, simulations with the more advanced customer choice model offered fewer time slots earlier in the process on average. Nevertheless, it is found in Figure J.1 that the spread between the minimum and the maximum number of slots offered per day is broader, which implies that some customers still see more time slots than the average customer. That the simulation with the more advanced models offers fewer time slots may be because customers in the simulation reserve a delivery moment earlier, resulting in fewer consumers being considered lost and more time slots already fully booked in the beginning. To support this assumption and to obtain more information about the number of served customers, an analysis can be conducted based on the number of assigned orders to assess whether the number of offered slots impacts customer satisfaction and the likelihood of losing orders. The KPI for assigned customers, as displayed in Figure 6.4, can be evaluated to validate the number of customers served during the simulation. This is particularly important as one of the company's objectives is to minimise lost customers.

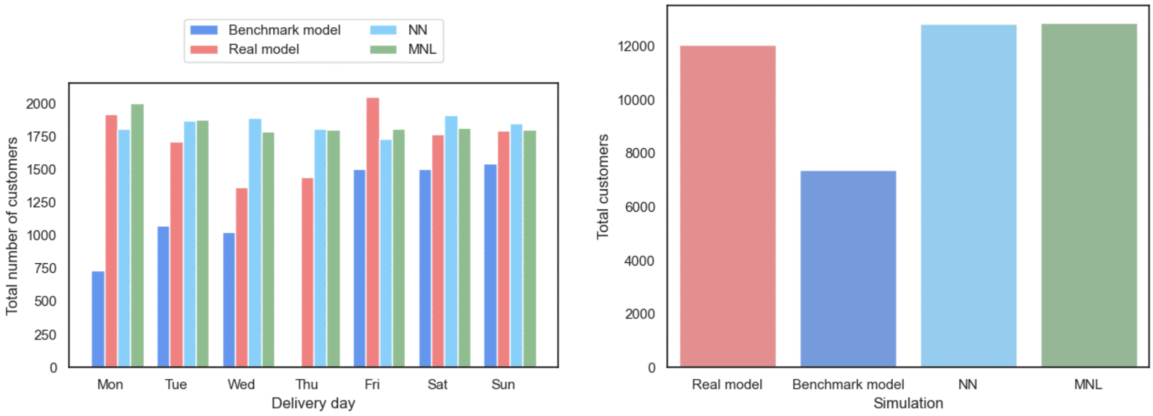


Figure 6.4: Comparison of daily and total number of served customers across the simulations.

Figure 6.4 displays the total number of customers served daily on the left side and the overall number of customers assisted for each simulation on the right side. These figures confirm that in the Benchmark model simulation, significantly fewer customers are served than in the actual results, and no customers are assigned for deliveries on Thursday. Conversely, on average, the MNL and NN models help more customers than the actual results. Accordingly, this proves that although fewer time slots are offered on average, more customers are served. A possible explanation for why the simulations with the MNL and NN models predict that more customers will be served might be because no orders are placed on the days before and after the delivery week during the simulation. As a result, the number of time slots offered for the weekend before and Monday and Tuesday after will always stay the maximum. The feature importance plots of both models can be analysed in Appendix G to determine the influence of these features on the different days. In these plots, it can be found that the number of offered slots for Monday and Tuesday positively affects the prediction that delivery will occur. Another reason for the discrepancy may be that in the simulation, customers are presented with all possible time slots from the beginning, whereas, in reality, customers can only place orders a twenty-five days in advance. Consequently, customers in the simulation may place orders earlier than they can do in reality. To see in more detail how these additional customers affect the delivery routes, the distribution of the executing time per route and per customer are visualised in Figure 6.5.

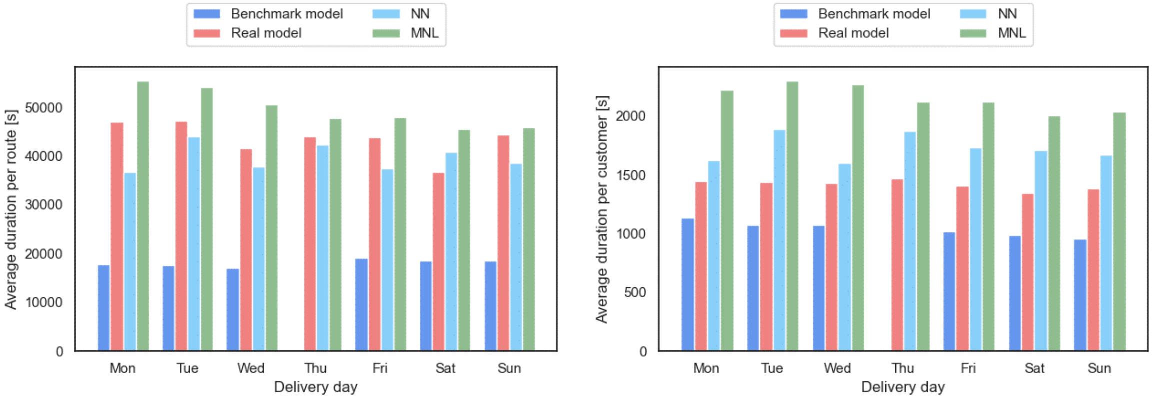


Figure 6.5: Comparison of the average duration per route and the average duration per customer in seconds across the simulations.

As indicated in Table 6.1 there is a difference in the average duration per route in seconds between the simulation using real-time slots and the simulation using advanced customer choice models and the Benchmark model. To better understand this, Figure 6.5 provides an overview of the distribution. The plots show that the average duration per route decreased on Thursdays and throughout the week. The average duration per customer visualised on the right side of Figure 6.5 also reveals the same results. As mentioned, this might be because optimising the routes becomes more complex when it is fully booked compared to when more space is available. This is because when the route is fully booked, there is less flexibility in changing the order of deliveries, as all the stops and their associated small time windows are already fixed. In contrast, when more space is available, there is greater flexibility to optimise the route by adjusting the order of deliveries or combining multiple deliveries into a single trip.

Considering the observed discrepancy in customer allocation between the models utilised in the simulation, it may be not appropriate to compare the total cost of routes. In such instances, it may be more suitable to compare the average route cost per customer, average executing cost per customer, and average number of customers per route.

The plots presented in Figure 6.6 display the average execution and route cost per customer and the average number of customers per route. The figures show that there is no direct correlation between the number of customers assigned to a route and the average execution cost, as the cost is lower for simulations using real-selected time slots than for those using the MNL and NN models, but higher than the costs from the Benchmark model. However, a linear relation can be found between the average route cost and the number of customers per route.

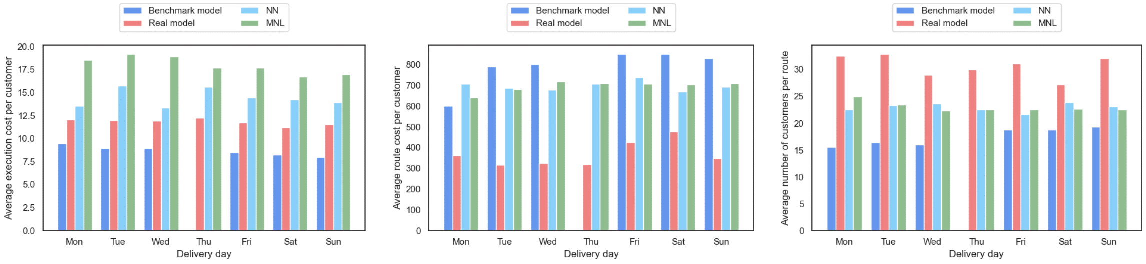


Figure 6.6: Comparison of the average execution and route cost per customer and the average number of customers per route.

To summarise, the simulation results using ORTEC's event-based simulation tool evaluate the impact of the advanced predictive models, the Benchmark model and real-selected time slots. The simulations are compared concerning all results of the KPIs together as each provides valuable insights into different aspects of the outcome. Analysing the set KPIs it can be found that the MNL and NN improve the simulation results closer to reality than with the Benchmark model. When looking into more detail to the results of the simulations it can be seen that the real-selected time slots and the Benchmark model offered the most time slots, while simulations with more advanced customer choice models offered fewer time slots. However, the MNL and NN models predicted that more customers would be served, possibly due to the time slots offered for the weekend before, Monday, and Tuesday after the delivery week or due to earlier availability of all time slots in the simulation. The analysis also revealed that the initial cost per route decreases as the number of customers on a route increases, resulting in lower costs per customer. Nevertheless, simulations with fewer customers on a route showed lower execution costs per customer, which may seem counterintuitive at first glance. The higher costs in this case can be attributed to the fact that variable costs do not occur on a pro-rata basis and depend on the resources including order size, execution time, and distance to travel to the customer. As a result, the higher average duration per customer and the less efficient routes created may contribute to higher costs. However, the profitability of the route may still be positive due to the operating margin, which can be calculated when the revenues, costs and fees are known. In addition, assigning more customers for strategic reasons, even if it leads to losses, can be a worthwhile investment in retaining customers and maintaining competitive.

6.3. Test scenarios

The study reveals till this point two important findings. Firstly, the MNL and NN models demonstrate improved prediction performance in customer choice models. Secondly, the simulation results indicate that utilising these models brings the simulation closer to reality compared to the Benchmark model. Given these results, it is worth investigating whether these more advanced customer choice models can be utilised to optimise offer strategies and provide added value. This can be done using the simulation tool as it can assess whether the customer choice models confer additional value in optimising the offered sets. The dynamic slotting approach is augmented as the offer sets become more tailored to individual customers, requiring more decisions per customer request throughout the booking horizon. Two test scenarios are created and will be compared based on the KPIs mentioned earlier and take both supply and demand management into account. It is decided to create two test scenarios since the simulations take approximately two days to complete because the actual company data, described in section 4, contains many customers spread over thirty-two days requiring multiple route optimisations. Route optimisation requires the most time, and based on the data provided, up to eighty routes will be made available each day. It is important to note that during the optimisation process, customer choice models were utilised to both adjust the offer set and select the time slots. However, utilising the same model to adjust different components of the optimisation process makes it challenging to compare the results. Although it is challenging this approach was chosen. Potentially a future study that builds on the results of this study could be extended on this. An overview of the created test scenarios can be found in Table 6.2. The test scenarios are explained in more detail afterwards.

Test scenario	Considerations	Expected outcomes	Customer choice model	Removed base on	Name
1	Execution costs and probabilities of selecting a time slot are considered. Time slots with high execution costs and low probabilities of selection are excluded to reduce execution costs.	Minimal impact on customer satisfaction, supportive influence on lowering execution costs, limited loss of time slots, sufficiently large data set for analysis.	MNL	20% lowest probabilities 20% highest cost	MNL 20/20
				30% lowest probabilities 30% highest cost	MNL 30/30
			NN	20% lowest probabilities 20% highest cost	NN 20/20
				30% lowest probabilities 30% highest cost	NN 30/30
2	Only probabilities of selecting a time slot are considered. All time slots with lowest likelihood of selection are discarded to reduce the offer set size.	Larger reduction in offer set size, routes become less diverse, and more customers predicted as lost.	MNL	20% lowest probabilities	MNL 20
				30% lowest probabilities	MNL 30
			NN	20% lowest probabilities	NN 20
				30% lowest probabilities	NN 30

Table 6.2: Overview of the created test scenarios to determine if the offer strategy can be optimised with the use MNL and NN models.

In the initial test scenario, the expenses related to the company's execution are considered in conjunction with the probabilities assigned by choice models to different time slots. Suppose the calculated execution cost is high while the likelihood of a customer selecting a given time slot is low. In that case, it raises the question of whether excluding this time slot would impact customer convenience. To investigate the effect of such exclusions on KPIs, time slots with high execution costs and low probabilities of selection will not be offered to customers during the simulation. This approach is expected to have a limited effect on customer satisfaction, as the set offered is not empty, and precisely the slots offered will be minimally different from customer preferences. Furthermore, the loss of time slots that meet these two requirements will be limited, ensuring a sufficiently large data set for analysis. On the contrary, it is expected that these changes will have a supportive influence on lowering execution costs resulting in a reduction in the thin profit margins. To ensure that all customers can still select a time slot by preventing excluding customers with higher costs due to their location, the maximum cost value depends on individual costs rather than a universal value. Based on this test scenario, two different sorts of simulations are performed. In the first simulation, the top twenty per cent most expensive and the lowest twenty per cent likelihoods are indicated, and when a time slot complies with both rules, it would be removed from the offer set shown to the customer. The same construction is used for a second test on the first simulation. In the second test the top thirty per cent most expensive and the lowest thirty per cent likelihoods will be found.

The second test scenario focuses solely on the probability of selecting a time slot. In this case, the offer set is created based only on the likelihood of selection, and the cost is not considered. As a result, it is assumed that the size of the offer set will be reduced, with only the top twenty or thirty per cent time slots with the lowest likelihood of selection being discarded to ensure that all customers are still presented with offers. This simulation is expected to result in changes in the KPIs since the reduction in offer set size may affect attractiveness and execution costs. Despite the smaller offer sets containing the preferred time slots, the routes are expected to become less diffuse as removing the time slots with low likelihood eliminates the small probability of selection. Consequently, the routes become less diverse, allowing for more customers in one route. Although more customers may be predicted as lost due to the smaller offer sets, leading to experience less convenience. In this study it is a useful way to gauge the impact of the offer set size only, which can be used for later simulations.

To measure the impact of the test scenario as accurate as possible, each simulation results will be compared with the results of the simulation using the real-selected time slots. This is done since, as above-mentioned, two changes have been applied in the same simulation.

6.4. Test scenario results

In order to determine whether the advanced choice models can be employed to optimise the offer set, the same KPIs will be used to compare the results with the ones of the real-selected time slot stimulation. An overview of all the results can be found in Table 6.3.

	Real model	MNL 20/20	MNL 30/30	NN 20/20	NN 30/30	MNL 20	MNL 30	NN 20	NN 30
Average number of seen slots	143.496	98.958	117.913	111.549	114.976	96.631	87.497	98.997	86.624
Average duration of the route [s]	43496	49169	49899	36091	32957	37862	29091	38926	28274
Total assigned customers	12047	12630	13125	11504	11757	9139	7344	4780	4773
Average number of customers per route	30.678	22.511	23.395	22.139	24.415	20.643	16.010	16.366	14.924
Average duration per customer [s]	1417	2181	2134	1622	1351	1867	2462	3220	2945
Average route cost per customer	6.524	8.885	8.549	8.428	8.192	8.527	9.029	7.462	7.479
Average execution cost per customer	11.806	18.177	17.786	13.518	11.260	15.559	20.520	26.829	24.544

Table 6.3: Overview of the KPIs from the simulations performed for all test scenarios.

Analysing the outcomes presented in Table 6.3, it is observed that the simulations utilising the second test scenario have significantly lower customer assignment compared to the simulation employing the real-selected time slots. This is particularly noticeable in the results of MNL 30, NN 20 and NN 30 simulations. When considering the relatively high execution cost and duration per customer, as well as the low number of customers per route, it is assumed that these scenarios are not suitable for optimising the offering strategy in this study. It is worth noting that the assumed sub-optimal strategies (MNL 30, NN 20, NN 30) have a low average duration of the route, and the number of offers does not reduce further next to the decrease due to the strategy, resulting in a relatively large number of time slots compared to the real results. This is due to the low number of consumers served, which means fewer time slots are occupied, and the strategy mainly affects the number, and more efficient routes can be created. Another remarkable result is that in the first scenario, the average number of offered slots and assigned customers are higher when thirty per cent is eliminated compared to the scenario where only twenty per cent is eliminated. A possible explanation could be that by removing more slots, the offer set is more balanced, which means that maybe not only time slots are removed that have a positive influence on the probability that the order is placed but also time slots that have a negative influence. To determine whether this is the reason, more simulations need to be performed with a combination of different choice models to better define the influences. However, the remaining simulation results of the test scenarios can be compared separately with the actual results. It has been determined through comparisons that the employment of NN 30/30 may enhance the offering strategy, given the slight decrease in the number of assigned customers and a comparatively more substantial reduction in the average duration of routes. As anticipated, the average number of seen slots decreases, aligning with the implemented strategy. Moreover, it is noteworthy that the average execution cost and duration per customer show a decrease. The relevance of the costs depends on the availability of information concerning revenue, cost, and delivery fees. Upon comparing the other strategies, it was observed that the MNL 20/20 and MNL 30/30 strategies resulted in the assignment of more customers, but at the cost of increased average route duration (per customer). Moreover, the number of customers per route decreased, which might have been expected to enable the creation of more efficient routes.However, the costs increased in a similar manner as the duration. The reverse is indicated for the NN 20/20 and MNL 30 as the number of served customers and average duration decrease. Nevertheless, it is indicated that the average duration per customer increases, which was not expected as it was expected that more efficient routes could be made.

Based on the above-discussed results, the first test scenario reveals some interesting results. While it may be difficult to directly compare the various test scenarios, an attempt is made to compare their route related outcomes with the simulation results in which no offer strategy is applied, using the same customer choice model. However, it is crucial to consider the number of assigned customers, as previous findings have shown that generating more efficient routes may be feasible with fewer served customers.

	MNL	MNL 20/20	MNL 30/30
Total assigned customers	12904	12630	13125
Average number of customers per route	23.014	22.511	23.395
Average duration per customer [s]	2154	2181	2134
Total average cost per customer	26.637	27.062	26.335

Table 6.4: Summary overview of the results form the simulations with and without offering strategy utilising the MNL model.

In the presented summary Table 6.4, an assessment of the results reveals that improving the overall routing efficiency might be possible based on the tested offering strategy. The improvement has the

most chance to be achieved by excluding the thirty most expensive time slots from the simulation when they are also among the bottom thirty per cent regarding the likelihood of selection. A possible reason more customers are assigned when using thirty per cent and more time slots are deleted from the offer set is that it leads to a more balanced offer set, which might benefit the customer choice model, as previously discussed. But, more simulations need to be performed to substantiate these assumptions and determine the strategy's precise influence.
The results indicate that with this strategy, the total number of customers and the average number of customers per route increases, and the average duration and the total average cost decrease. The total average cost per customer is the sum of the average route cost per customer and the average execution cost per customer. Although, the insights into the effects could be improved when the revenues, costs and delivery fees are known. Based on the above test results, improving the overall routing efficiency by using the same choice model is possible. The same comparison is made in the first test scenario using the NN customer choice model, and an overview of the results can be found in Table 6.5.

	NN	NN 20/20	NN 30/30
Total assigned customers	12897	11504	11757
Average number of customers per route	22.968	22.139	24.415
Average duration per customer [s]	1728	1622	1351
Total average cost per customer	23.105	21.946	19.452

Table 6.5: Summary overview of the results form the simulations with and without offering strategy utilising the NN model.

Based on the visualised results regarding the route in Table 6.5, it can be analysed that by conducting the strategies, the number of assigned customers compared to the simulation with the same choice model decreases, allowing for generating more efficient routes and reducing the average duration per customers. Moreover, it is found that with the NN 30/30 model, the average duration reduces more, while in comparison with the NN 20/20, more customers are served. Further, another remarkable result is that the total average cost per customer decreases, while it is expected that both the average route and execution costs will increase. Both findings could be a consequence of the strategy since it removes the most expensive time slots, and it is possible that the most expensive also take up the most time. Therefore, this strategy might be very promising based on comparing the routing result. The total effect on the profitability of this scenario can only be determined when revenues, costs, and fees are known.

In summary, comparing the outcomes of the strategies with the real outcomes suggest that advanced customer choice models might optimise the offer strategy, with the NN 30/30 strategy showing particular best result. Because, based on the results, the most potential is seen in the first test scenario, the route-related results of these test scenarios are compared with the results of the simulation using the same customer choice model to gain a better understanding. It is indicated that after analysing the results in both comparisons, the average duration and the total average cost per customer are reduced, which could be a consequence of the applied strategy. Since it removes the most expensive time slots that possibly take the most time to execute and that had a low chance of being selected. Further simulations are necessary to assess the precise impact of the offer strategy and the prediction process, as the same choice model was employed for optimising the offer set and modelling customer behaviour. This can be done by using two different choice models in order to better identify the effect of the strategy. Moreover, additional simulations could, for instance, explore the effects of different probabilities or strategies based on different features and the impact of price incentives now that the probability of selecting a time slot can be more accurately determined. Also, similar tests could be performed using green labels, but dynamic pricing should not be included, as the combination results in less influence. Another possibility could be to train the customer choice models again with fewer time slots to determine the effect. It is good to take into account that only an initial exploration is conducted to investigate the potential of optimising offer strategies. Based on this study, future studies can delve much deeper into this topic and compare different approaches in different manners.

7 Conclusion

The main aim of this study is to determine whether customer choice behaviour can be better identified than a Benchmark model based on assumptions and used to optimise the provided offer set in the context of online groceries.

To determine how customer choice behaviour can be better improved, a literature review is conducted to understand why customers order online groceries in the first place, how this process works, and which techniques are currently used to model their behaviour. During this research, it was found that the steps in the delivery process are strongly interlinked and interdependent. Therefore, it is of interest to obtain a better insight into customer behaviour to optimise the correlations between these steps using demand management. Demand management aims to manage trade-offs between customer satisfaction and the company's logistics and profit by shaping and generating customer demand while keeping sight on efficiency and profit. This requires obtaining the customers' behaviour using DCMs, which are often used for this problem. In recent years theory-driven parametric DCMs have been conducted as they are simple and readable as the estimated weights allied to customer characteristics can be obtained directly. The most used parametric DCM is the MNL model. Unfortunately, parametric models also have several limitations since manual specifications are needed, which are laborious and prone to errors which can affect the prediction performance. In combination with the increasing amount of data, the parametric DCMs are nowadays replaced with ML and other data-driven methods, making the selection process more effective and less vulnerable to subjective prejudice. However, it is found that non-parametric models lack interpretability and often function as black boxes. In addition, not all models are as applicable and accurate as the others; therefore, based on the literature, it is decided to use the NN model with multiple hidden layers to identify and determine the characteristics that influence customer behaviour. To improve interpretation, as it is paramount to understand customers' decision-making process, SHAP values will be derived to provide these insights.

The parametric MNL model and non-parametric NN model has been employed in this study after hyperparameter tuning to analyse if it is possible to identify customer behaviour better. A Benchmark model ensures that the MNL and NN add value in indicating the behaviour and prediction performance. The results will be compared based on the F1-score since all three models follow the same prediction process and use the same customer features. The prediction process includes an initial prediction of the week of delivery, followed by a subsequent prediction of the exact day of delivery, and finally, a prediction of the expected delivery time. Based on the results, it is found that the MNL and NN models have better prediction performances than the Benchmark model, indicating that both models add value to the performance. However, when comparing the MNL with the NN model, it is found that the NN model even has a better performance. Therefore it is decided that the NN model can model customer behaviour the best of these models. When looking in detail at the results, it is found that the models consider the exact features necessary in the first step in the process. The most important feature is the week number in which the request is made, followed by the average weight of the order. In the second and third step, the MNL and NN models evaluate the importance of the features differently. In the models where it is predicted if the delivery is occurring, and if so, at which day the delivery is taking place, the MNL models consider the number of available slots for the delivery days as most important, followed by the average weight of the order and in which week the request is made. In contrast to the MNL, the NN model considers the average weight of the order as the most influential, followed by which day the request is made and the number of available slots per day. Similar results of different importance are found in the step predicting the part of the delivery day. The MNL model assigns the highest weight to the number of available slots per day, followed by the features containing historical information. In comparison, the NN model gives more weight to the historical features, followed by the number of available time slots and the day the request is made. That the MNL and NN models assign various weights to the same features, even when trained on the same data, is expected as both models fundamentally differ with other assumptions and learning mechanisms. For instance, an MNL model assumes that the relationship between the independent and dependent variables is linear, while a NN model can also model non-linear relationships. Additionally, NN models can automatically learn features through hidden layers, whereas MNL models require pre-defined features.

The SHAP plots depicting feature importance for predicting the delivery day component suggest that historical information could be crucial in improving the prediction accuracy. As a result, an investigation of the impact of exclusively utilising the models for customers who have previously placed orders is performed. The analysis tests the influence of historical information, given the potential capacity limitations associated with storing more data. Nonetheless, based on the outcomes, it is evident that the prediction performance of both the MNL and NN models improved significantly in predicting the delivery day moment. In contrast, little impact is observed on the models predicting the delivery day, which can be attributed to the limited historical information in this prediction step's features. These findings need further exploration into ways to incorporate more historical data and how it can be acquired, particularly to enhance performance in predicting the delivery day component and to ascertain the possible improvements in the day prediction step.

Given that the advanced customer choice models offer improved prediction performance and a better understanding of customer behaviour than the Benchmark model, the question remains whether these models add value and truly enhance ORTEC's event-based simulation closer to reality. To evaluate this, the simulation tool integrates the MNL, NN, and Benchmark models separately, and the results are compared to a simulation based on the real-selected time slots. For this comparison three KPIs are introduced to assess the modelled systems' performance and must be used together as each KPI provides valuable insights into different aspects. The used KPIs include: the number of offered time slots per customer, the total execution time of the routes and the number of assigned customers.

Based on the analysis of the KPIs obtained from the simulations, it can be concluded that the MNL and NN models positively impact the simulation, improving the results closer to reality when compared to the Benchmark model. A more comprehensive analysis of the simulations shows that while the actual-selected time slots and the Benchmark model offered the most time slots, simulations utilising the MNL and NN models offered fewer time slots. On the other hand, the MNL and NN models predicted that more customers would be served, possibly due to the time slots offered for the weekend before, Monday, and Tuesday after the delivery week or earlier availability of all time slots in the simulation. The analysis also revealed that the initial cost per route decreases as the number of customers on a route increases, resulting in lower costs per customer. On the other hand, simulations with fewer customers on a route showed lower execution costs per customer, which may seem counterintuitive at first glance. In this case, the higher costs can be attributed to the fact that variable costs do not occur on a pro-rata basis and depend on the resources, including order size, execution time, and distance to travel to the customer. Additionally, fixed costs could arise in certain step-up values. Consequently, an additional customer will lead to a higher average fixed cost than was applicable for the preceding customer. As a result, the higher average duration per customer and the less efficient routes created contribute to higher costs. However, the route's profitability may still be positive due to the operating margin, which can be calculated when the revenues, costs, and fees are known[1]. In addition, assigning more customers for strategic reasons, even if it leads to losses, can be a worthwhile investment in retaining customers and maintaining competitiveness. To conclude, when all findings are combined as they must be used as a whole, it is indicated that the MNL and NN models improve the simulation results since the outcomes are closer to the values of the actual-model than from the Benchmark model.

[1] This is the economic definition of profit maximisation by calculating the marginal costs versus the marginal revenues. When these two are equal, the maximum profit will be made.

Based on the finding that the MNL and NN models improve the prediction performance in customer choice models and are able to bring the simulation closer to reality compared to the Benchmark model, a start can be made to investigate whether both models can be used to optimise the offer strategy. Two test scenarios are created to analyse the impact of excluding time slots. The first test scenario excludes time slots with high execution costs and low probabilities of selection, and the second test scenario focuses solely on the probability of selecting a time slot. To accurately measure the impact of the test scenarios, each simulation result was compared with the results of the simulation using the real-selected time slots. Results revealed that the created test scenarios might optimise the offer strategies when comparing the KPIs with the real outcomes. The first test scenario indicated the most potential. Therefore, the route-related results of this test scenario are compared with the results of the simulation using customer choice models to obtain more information. Evaluating this comparison reveals that the average duration and the total average cost per customer are reduced, which could be a consequence of the applied strategy. Since it removes the most expensive time slots that possibly take the most time to execute and that had a low chance of being selected. However, to be more certain and conclude, further investigation needs to be done with the use of more simulations utilising other choice models. Additionally, other simulations could be performed to determine the effect of strategies created based on different features or the influence of price incentives now that the probability of selecting a time slot can be more accurately determined.

In conclusion, this study aimed to identify and model customer choice behaviour in the context of online groceries to optimise the offer set. The literature review found that demand management is crucial for optimising customer satisfaction and company logistics. The parametric MNL and non-parametric NN models were employed to evaluate customer behaviour. After hyperparameter tuning, it was found that both models outperformed the Benchmark model in predicting customer behaviour, with the NN model showing the best performance. The MNL and NN models evaluated feature importance differently. Overall the study concluded that advanced customer choice models offer improved prediction performances and a better understanding of customer behaviour and can enhance simulation tools to provide more realistic results. Furthermore, a first step is made to analyse whether the advanced customer choice models can be utilised to optimise offer strategies where promising results are found, but further research is needed.

Based on the findings of this study, several recommendations can be made to enhance the prediction performance and optimise the offer set in the context of online groceries. Firstly, it may be helpful to store and use more historical customer data to improve the accuracy of the predictions. Besides, further research could explore applying different choice models, such as the RF, to evaluate whether the prediction performance can be improved. Additionally, incorporating more data or including data from other companies could enhance the predictions and provide new insights or features that may impact the overall performance of the analysis. Because other companies can have other processes where, for example, the groceries are selected, and only then the delivery moment is selected. Besides, it could be advantageous to gain further insights by retraining the customer choice models based on simulation data and analysing which information may be lost. Furthermore, now it is found that the more advanced models can identify customer behaviour better, and different varieties of incentives can be included to determine the impact since this was not possible in this research due to the available data. Incentives that can be used are, for example, green labels or price changes and could be implemented on the time slots with a positive selection change to adjust the provided offer set. Finally, further testing can be performed to investigate whether alternative offering strategies based on other features can significantly improve the KPIs in the simulation.

Bibliography

[1] Niels Agatz, Yingjie Fan, and Daan Stam. "Going green: the effect of green labels on delivery time slot choices". In: *Available at SSRN 3656982* (2020).

[2] Niels Agatz, Yingjie Fan, and Daan Stam. "The impact of green labels on time slot choice and operational sustainability". In: *Production and Operations Management* 30.7 (2021), pp. 2285–2303.

[3] Niels Agatz et al. "Challenges and opportunities in attended home delivery". In: *The vehicle routing problem: Latest advances and new challenges* (2008), pp. 379–396.

[4] Niels Agatz et al. "Demand management opportunities in e-fulfillment: What internet retailers can learn from revenue management". In: (2008).

[5] Niels Agatz et al. "Revenue management opportunities for Internet retailers". In: *Journal of Revenue and Pricing Management* 12.2 (2013), pp. 128–138.

[6] Niels Agatz et al. "Time slot management in attended home delivery". In: *Transportation Science* 45.3 (2011), pp. 435–449.

[7] Emel Aktas, Michael Bourlakis, and Dimitris Zissis. "Collaboration in the last mile: evidence from grocery deliveries". In: *International Journal of Logistics Research and Applications* 24.3 (2021), pp. 227–241.

[8] Ahmad Alwosheel, Sander van Cranenburgh, and Caspar G Chorus. "Is your dataset big enough? Sample size requirements when using artificial neural networks for discrete choice analysis". In: *Journal of choice modelling* 28 (2018), pp. 167–182.

[9] Pedro Amorim et al. "Customer preferences for delivery service attributes in attended home delivery". In: *Chicago Booth Research Paper* 20-07 (2020).

[10] Kursad Asdemir, Varghese S Jacob, and Ramayya Krishnan. "Dynamic pricing of multiple home delivery options". In: *European Journal of Operational Research* 196.1 (2009), pp. 246–257.

[11] Ayuya. *Parametric versus Non-Parametric Models*. Feb. 2021. URL: `https://www.section.io/engineering-education/parametric-vs-nonparametric/`.

[12] Barbara Baarsma and Jesse Groenewegen. "COVID-19 and the demand for online grocery shopping: Empirical evidence from the Netherlands". In: *De Economist* 169.4 (2021), pp. 407–421.

[13] Will Badr. *Why Feature Correlation Matters .... A Lot! - Towards Data Science*. Dec. 2021. URL: `https://towardsdatascience.com/why-feature-correlation-matters-a-lot-847e8ba439c4`.

[14] Pragati Baheti. *Activation Functions in Neural Networks [12 Types amp; Use Cases]*. Feb. 2023. URL: `https://www.v7labs.com/blog/neural-networks-activation-functions#:~:text=Similar20to20the20sigmoid2Flogistic,case20of20multi2Dclass20classi`

[15] Fernando Bernstein, Sajad Modaresi, and Denis Sauré. "A dynamic clustering approach to data-driven assortment personalization". In: *Management Science* 65.5 (2019), pp. 2095–2115.

[16] Jose Blanchet, Guillermo Gallego, and Vineet Goyal. "A markov chain approximation to choice modeling". In: *Operations Research* 64.4 (2016), pp. 886–905.

[17] Pieter S Bouwstra et al. "Stochastic and dynamic routing with flexible deliveries for an e-grocer". In: *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE. 2021, pp. 3354–3359.

[18] Pedro Quelhas Brito et al. "Customer segmentation in a large database of an online customized fashion business". In: *Robotics and Computer-Integrated Manufacturing* 36 (2015), pp. 93–100.

[19] Brownlee. *How to Choose a Feature Selection Method For Machine Learning*. Nov. 2019. URL: https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/.

[20] J Brownlee. *Why Is Imbalanced Classification Difficult?* Feb. 2020. URL: https://machinelearningmast com/imbalanced-classification-is-hard/.

[21] Jason Brownlee. *How to Choose an Activation Function for Deep Learning*. Jan. 2021. URL: https://machinelearningmastery.com/choose-an-activation-function-for-deep-learning/.

[22] Norman Buysse. *Enorme groei van online boodschappen doen is voorbij: ouderen gaan graag zelf op pad*. Nederlands. July 3, 2021. URL: https://www.ad.nl/koken-en-eten/enorme-groei-van-online-boodschappen-doen-is-voorbij-ouderen-gaan-graag-zelf-op-pad (visited on 10/06/2022).

[23] Eunjoo Byeon. *Introduction to Geopy: Using Your Latitude Longitude Data in Python*. Dec. 2021. URL: https://towardsdatascience.com/things-to-do-with-latitude-longitude-data-using-geopy-python-1d356ed1ae30.

[24] Ann Melissa Campbell and Martin Savelsbergh. “Incentive schemes for attended home delivery services”. In: *Transportation science* 40.3 (2006), pp. 327–341.

[25] Ann Melissa Campbell and Martin WP Savelsbergh. “Decision support for consumer direct grocery initiatives”. In: *Transportation Science* 39.3 (2005), pp. 313–327.

[26] Marco Casazza, Alberto Ceselli, and Lucas Létocart. “Optimizing Time Slot Allocation in Single Operator Home Delivery Problems”. In: *Operations Research Proceedings 2014*. Springer, 2016, pp. 91–97.

[27] Centraal Bureau voor de Statistiek. *Bevolkingsteller*. Nederlands. Feb. 25, 2022. URL: https://www.cbs.nl/nl-nl/visualisaties/dashboard-bevolking/bevolkingsteller (visited on 10/10/2022).

[28] Neha Chaudhuri et al. “On the platform but will they buy? Predicting customers’ purchase behavior using deep learning”. In: *Decision Support Systems* 149 (2021), p. 113622.

[29] Graham Clarke, Christopher Thompson, and Mark Birkin. “The emerging geography of e-commerce in British retailing”. In: *Regional Studies, Regional Science* 2.1 (2015), pp. 371–391.

[30] Paul Covington, Jay Adams, and Emre Sargin. “Deep neural networks for youtube recommendations”. In: *Proceedings of the 10th ACM conference on recommender systems*. 2016, pp. 191–198.

[31] Ayan Kumar Dhar. *Understanding Activation Functions and Hidden Layers in Neural Networks*. Jan. 2022. URL: https://medium.com/analytics-vidhya/understanding-activation-functions-and-hidden-layers-in-neural-networks-4fca2b980917.

[32] Christopher Dossman. *Top 6 Errors Novice Machine Learning Engineers Make*. Mar. 2021. URL: https://medium.com/aiC2B3-theory-practice-business/top-6-errors-novice-machine-learning-engineers-make-e82273d394db.

[33] EliteDataScience. *How to Handle Imbalanced Classes in Machine Learning*. July 2022. URL: https://elitedatascience.com/imbalanced-classes.

[34] Vivek F Farias, Srikanth Jagabathula, and Devavrat Shah. “A nonparametric approach to modeling choice with limited data”. In: *Management science* 59.2 (2013), pp. 305–322.

[35] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. “From data mining to knowledge discovery in databases”. In: *AI magazine* 17.3 (1996), pp. 37–37.

[36] Jacob B Feldman and Huseyin Topaloglu. “Revenue management under the Markov chain choice model”. In: *Operations Research* 65.5 (2017), pp. 1322–1342.

[37] Qi Feng, J George Shanthikumar, and Mengying Xue. “Consumer choice models and estimation: A review and extension”. In: *Production and Operations Management* 31.2 (2022), pp. 847–867.

[38] Guillermo Gallego, Richard Ratliff, and Sergey Shebalov. “A general attraction model and sales-based linear program for network revenue management under customer choice”. In: *Operations Research* 63.1 (2015), pp. 212–232.

[39] Laurie A Garrow and Frank S Koppelman. “Multinomial and nested logit models of airline passengers’ no-show and standby behaviour”. In: *Journal of Revenue and Pricing Management* 3.3 (2004), pp. 237–253.

[40] Valerio Gatta et al. “E-groceries: A channel choice analysis in Shanghai”. In: *Sustainability* 13.7 (2021), p. 3625.

[41] Manas Gaur, Shruti Goel, and Eshaan Jain. “Comparison between nearest Neighbours and Bayesian network for demand forecasting in supply chain management”. In: *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*. IEEE. 2015, pp. 1433–1436.

[42] Tobias Gawor and Kai Hoberg. “Customers’ valuation of time and convenience in e-fulfillment”. In: *International Journal of Physical Distribution & Logistics Management* (2018).

[43] *GfK-onderzoek: websuper worstelt met hele bestelling en Crisp de beste*. Nederlands. URL: https://www.gfk.com/insights/Crisp-wint-E-commerce-FMCG-Rapport (visited on 10/06/2022).

[44] Soumi Ghosh and Chandan Banerjee. “A predictive analysis model of customer purchase behavior using modified random forest algorithm in cloud environment”. In: *2020 IEEE 1st International conference for convergence in engineering (ICCE)*. IEEE. 2020, pp. 239–244.

[45] Gupta. *Feature Selection Techniques in Machine Learning*. Dec. 2020. URL: https://www.analyticsvidhya.com/blog/2020/10/feature-selection-techniques-in-machine-learning/.

[46] Arushi Gupta and Daniel Hsu. “Parameter identification in Markov chain choice models”. In: *Theoretical Computer Science* 808 (2020), pp. 99–107.

[47] Brij Gupta, Dharma P Agrawal, and Shingo Yamaguchi. *Handbook of research on modern cryptographic solutions for computer and cyber security*. IGI global, 2016.

[48] Isha Gupta. *5 Tips for Consumer Choice Models - Towards Data Science*. Dutch. Dec. 14, 2021. URL: https://towardsdatascience.com/consumer-preference-models-85e297887b1b (visited on 10/03/2022).

[49] Liana van der Hagen et al. “Machine Learning-Based Feasability Checks for Dynamic Time Slot Management”. In: *Available at SSRN 4011237* (2022).

[50] Julian Hagenauer and Marco Helbich. “A comparative study of machine learning classifiers for modeling travel mode choice”. In: *Expert Systems with Applications* 78 (2017), pp. 273–282.

[51] Jeff Hale. *Don’t Sweat the Solver Stuff - Towards Data Science*. Dec. 2021. URL: https://towardsdatascience.com/dont-sweat-the-solver-stuff-aea7cddc3451.

[52] Nick Hood et al. “Sociodemographic and spatial disaggregation of e-commerce channel use in the grocery market in Great Britain”. In: *Journal of Retailing and Consumer Services* 55 (2020), p. 102076.

[53] Chih-Wei Hsu and Chih-Jen Lin. “A comparison of methods for multiclass support vector machines”. In: *IEEE transactions on Neural Networks* 13.2 (2002), pp. 415–425.

[54] IBM. *What is the k-nearest neighbors algorithm? | IBM*. Sept. 2022. URL: https://www.ibm.com/topics/knn.

[55] IBM Cloud Education. *Neural Networks*. Aug. 2020. URL: https://www.ibm.com/cloud/learn/neural-networks.

[56] Angel Igareta. *Dealing with Imbalanced Data in TensorFlow: Class Weights*. Mar. 2022. URL: https://towardsdatascience.com/dealing-with-imbalanced-data-in-tensorflow-class-weights-60f876911f99#:~:text=multiple20output20model.-,Generating20class20weights,error20than20the20majority20class..

[57] D Kaleko. *Feature Engineering - Handling Cyclical Features*. Oct. 2017. URL: `http://blog.davidkaleko.com/feature-engineering-cyclical-features.html`.

[58] Philipp ‘Phil’ Klaus and Stan Maklan. “Towards a better measure of customer experience”. In: *International journal of market research* 55.2 (2013), pp. 227–246.

[59] Robert Klein et al. “Differentiated time slot pricing under routing considerations in attended home delivery”. In: *Transportation Science* 53.1 (2019), pp. 236–255.

[60] Charlotte Köhler, Jan Fabian Ehmke, and Ann Melissa Campbell. “Flexible time window management for attended home deliveries”. In: *Omega* 91 (2020), p. 102023.

[61] Ajitesh Kumar. *Correlation Concepts, Matrix Heatmap using Seaborn*. Apr. 2022. URL: `https://vitalflux.com/correlation-heatmap-with-seaborn-pandas/`.

[62] Kuo. *Explain Your Model with the SHAP Values - Dataman in AI*. Sept. 2014. URL: `https://medium.com/dataman-in-ai/explain-your-model-with-the-shap-values-bc36aac4de3d`.

[63] Katherine N Lemon and Peter C Verhoef. “Understanding customer experience throughout the customer journey”. In: *Journal of marketing* 80.6 (2016), pp. 69–96.

[64] Alix Lhéritier et al. “Airline itinerary choice modeling using machine learning”. In: *Journal of choice modelling* 31 (2019), pp. 198–209.

[65] Stanley Frederick WT Lim and Matthias Winkenbach. “Configuring the last-mile in business-to-consumer e-retailing”. In: *California Management Review* 61.2 (2019), pp. 132–154.

[66] Renzhi Lu and Seung Ho Hong. “Incentive-based demand response for smart grid with reinforcement learning and deep neural network”. In: *Applied energy* 236 (2019), pp. 937–949.

[67] Scott M Lundberg and Su-In Lee. “A unified approach to interpreting model predictions”. In: *Advances in neural information processing systems* 30 (2017).

[68] Jochen Mackert. “Choice-based dynamic time slot management in attended home delivery”. In: *Computers & Industrial Engineering* 129 (2019), pp. 333–345.

[69] Jochen Mackert, Claudius Steinhardt, and Robert Klein. “Integrating customer choice in differentiated slotting for last-mile logistics”. In: (2019).

[70] Gianluca Malato. *How to explain neural networks using SHAP | Your Data Teacher*. May 2021. URL: `https://www.yourdatateacher.com/2021/05/17/how-to-explain-neural-networks-using-shap/`.

[71] Wilson E Marcílio and Danilo M Eler. “From explanations to feature selection: assessing SHAP values as feature selection mechanism”. In: *2020 33rd SIBGRAPI conference on Graphics, Patterns and Images (SIBGRAPI)*. Ieee. 2020, pp. 340–347.

[72] Claudio Marcus. “A practical yet meaningful approach to customer segmentation”. In: *Journal of consumer marketing* (1998).

[73] Marius. *Multiclass Classification with Support Vector Machines (SVM), Dual Problem and Kernel Functions*. June 2020. URL: `https://towardsdatascience.com/multiclass-classification-with-support-vector-machines-svm-kernel-trick-kernel-functions-f9d5377d6f02`.

[74] Al-Masri. *How Does Backpropagation in a Neural Network Work?* Oct. 2022. URL: `https://builtin.com/machine-learning/backpropagation-neural-network`.

[75] Andrew McCallum, Kamal Nigam, et al. “A comparison of event models for naive bayes text classification”. In: *AAAI-98 workshop on learning for text categorization*. Vol. 752. 1. Madison, WI. 1998, pp. 41–48.

[76] Yuan Meng et al. “What makes an online review more helpful: an interpretation framework using XGBoost and SHAP values”. In: *Journal of Theoretical and Applied Electronic Commerce Research* 16.3 (2020), pp. 466–490.

[77] Mikulskibartosz. *How to deal with days of the week in machine learning*. Mar. 2021. URL: `https://www.mikulskibartosz.name/time-in-machine-learning/`.

[78] Christoph Molnar. *9.6 SHAP (SHapley Additive exPlanations) | Interpretable Machine Learning*. 2022. URL: https://christophm.github.io/interpretable-ml-book/shap.html.

[79] Alejandro Mottini and Rodrigo Acuna-Agost. “Deep choice model using pointer networks for airline itinerary prediction”. In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 2017, pp. 1575–1583.

[80] Mwende. *The Relationship Between Naive Bayes and Bayesian Network*. May 2021. URL: https://studyexcell.com/the-relationship-between-naive-bayes-and-bayesian-network/.

[81] Andy Newing et al. “‘Sorry we do not deliver to your area’: geographical inequalities in online groceries provision”. In: *The International Review of Retail, Distribution and Consumer Research* 32.1 (2022), pp. 80–99.

[82] NU.nl. *De onlineboodschappentaart is veel groter, het stuk van AH kleiner*. Nederlands. Feb. 13, 2021. URL: https://www.nu.nl/economie/6115598/de-onlineboodschappentaart-is-veel-groter-het-stuk-van-ah-kleiner.html (visited on 10/10/2022).

[83] ORTEC. *ORTEC*. 2022.

[84] Nicola Ortelli et al. “Assisted specification of discrete choice models”. In: *Journal of choice modelling* 39 (2021), p. 100285.

[85] Nicolas Pasquier and Sujoy Chatterjee. “Customer Choice Modelling: A Multi-Level Consensus Clustering Approach”. In: *Annals of Emerging Technologies in Computing (AETiC)* 5.2 (2021), pp. 103–120.

[86] Harshil Patel. *What is Feature Engineering — Importance, Tools and Techniques for Machine Learning*. Jan. 2022. URL: https://towardsdatascience.com/what-is-feature-engineering-importance-tools-and-techniques-for-machine-learning-2080b0269f10.

[87] Laura Patterson. *Predict Customer Choices – How Cool is That?!* Dutch. Sept. 7, 2021. URL: https://visionedgemarketing.com/choice-modelling-predicts-customer-preferences/ (visited on 10/03/2022).

[88] Serhat Peker, Altan Kocyigit, and P Erhan Eren. “LRFMP model for customer segmentation in the grocery retail industry: a case study”. In: *Marketing Intelligence & Planning* (2017).

[89] Derek A Pisner and David M Schnyer. “Support vector machine”. In: *Machine learning*. Elsevier, 2020, pp. 101–121.

[90] Raguzin et al. *READY FOOD*. Nov. 2020. URL: https://www.oliverwyman.com/content/dam/oliver-wyman/v2/publications/2020/November/Ready_Food.pdf (visited on 10/03/2022).

[91] Kim Ramus and Niels Asger Nielsen. “Online grocery retailing: what do consumers think?” In: *Internet research* (2005).

[92] Raschka. *Intro to Machine Learning*. 2018. URL: https://sebastianraschka.com/pdf/lecture-notes/stat479fs18/02_knn_notes.pdf.

[93] Susan Rose et al. “Online customer experience in e-retailing: an empirical model of antecedents and outcomes”. In: *Journal of retailing* 88.2 (2012), pp. 308–322.

[94] Sahar F Sabbeh. “Machine-learning techniques for customer retention: A comparative study”. In: *International Journal of Advanced Computer Science and Applications* 9.2 (2018).

[95] *Shopping Behavior 2022: Of shocks and accelerators*. Nederlands. URL: https://discover.gfk.com/story/shopping-behavior-2022/page/4/3 (visited on 10/06/2022).

[96] Brian Sifringer, Virginie Lurkin, and Alexandre Alahi. “Enhancing discrete choice models with neural networks”. In: *Proceedings of the 18th Swiss Transport Research Conference (STRC), Monte Verità/Ascona, Switzerland*. 2018, pp. 16–18.

[97] A Serdar Şimşek and Huseyin Topaloglu. “An expectation-maximization algorithm to estimate the parameters of the markov chain choice model”. In: *Operations Research* 66.3 (2018), pp. 748–760.

[98] Reema Singh and Magnus Söderlund. “Extending the experience construct: an examination of online grocery shopping”. In: *European Journal of Marketing* (2020).

[99] *sklearn.linear*$_{m}odel.LogisticRegression$. URL: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html.

[100] Slundberg. *GitHub - slundberg/shap: A game theoretic approach to explain the output of any machine learning model.* June 2022. URL: https://github.com/slundberg/shap#citations.

[101] Statista. *Online grocery shopping in the Netherlands - statistics facts*. Nederlands. Sept. 6, 2022. URL: https://www.statista.com/topics/6479/online-grocery-shopping-in-the-netherlands/#dossierKeyfigures (visited on 10/06/2022).

[102] Arne Strauss, Nalan Gülpınar, and Yijun Zheng. “Dynamic pricing of flexible time slots for attended home delivery”. In: *European Journal of Operational Research* 294.3 (2021), pp. 1022–1041.

[103] Arne K Strauss, Robert Klein, and Claudius Steinhardt. “A review of choice-based revenue management: Theory and methods”. In: *European journal of operational research* 271.2 (2018), pp. 375–387.

[104] Kenneth E Train. *Discrete choice methods with simulation*. Cambridge university press, 2009.

[105] C-Y Tsai and C-C Chiu. “A purchase-based market segmentation methodology”. In: *Expert systems with applications* 27.2 (2004), pp. 265–276.

[106] Chih-Fong Tsai and Mao-Yuan Chen. “Variable selection by association rules for customer churn prediction of multimedia on demand”. In: *Expert Systems with Applications* 37.3 (2010), pp. 2006–2015.

[107] S Van Cranenburgh et al. “Choice modelling in the age of machine learning”. In: *arXiv preprint arXiv:2101.11948* (2021).

[108] Garrett Van Ryzin and Gustavo Vulcano. “A market discovery algorithm to estimate a general class of nonparametric choice models”. In: *Management Science* 61.2 (2015), pp. 281–300.

[109] Rohit Verma. “Customer choice modeling in hospitality services: A review of past research and discussion of some new applications”. In: *Cornell Hospitality Quarterly* 51.4 (2010), pp. 470–478.

[110] Rohit Verma and Gerhard Plaschka. “Predicting customer choices: recent research has greatly improved management’s ability to anticipate customer wants”. In: *MIT Sloan Management Review* 47.1 (2005), pp. 7–11.

[111] Qiuping Wang, Hao Sun, and Qi Zhang. “A bayesian network model on the public bicycle choice behavior of residents: a case study of Xi’an”. In: *Mathematical Problems in Engineering* 2017 (2017).

[112] Katrin Waßmuth et al. “Demand Management for Attended Home Delivery–A Literature Review”. In: *ERIM Report Series Reference Forthcoming* (2022).

[113] Mary Wolfinbarger and Mary C Gilly. “eTailQ: dimensionalizing, measuring and predicting etail quality”. In: *Journal of retailing* 79.3 (2003), pp. 183–198.

[114] Chi Xie, Jinyang Lu, and Emily Parkany. “Work travel mode choice modeling with data mining: decision trees and neural networks”. In: *Transportation Research Record* 1854.1 (2003), pp. 50–61.

[115] Xinan Yang and Arne K Strauss. “An approximate dynamic programming approach to attended home delivery management”. In: *European Journal of Operational Research* 263.3 (2017), pp. 935–945.

[116] Xinan Yang et al. “Choice-based demand management and vehicle routing in e-fulfillment”. In: *Transportation science* 50.2 (2016), pp. 473–488.

[117] Liu Yue et al. “Demand forecasting by using support vector machine”. In: *Third International Conference on Natural Computation (ICNC 2007)*. Vol. 3. IEEE. 2007, pp. 272–276.

[118] Xilei Zhao et al. “Modeling Stated preference for mobility-on-demand transit: a comparison of Machine Learning and logit models”. In: *arXiv preprint arXiv:1811.01315* (2018).

[119] Huiqi Zhu, Shuihai Dou, and Ying Qiu. “Joint model for last-mile delivery service selection in China: evidence from a cross-nested logit study”. In: *IEEE Access* 7 (2019), pp. 137668–137679.

A

Scientific paper

The scientific paper start on the next page.

Investigating different models that can be used to define the characteristics that influence customer behaviour in the online grocery sector

M. Middelweerd*, B. Atasoy*, P. Zattoni Scroccaro*, W. Merkx**

** Faculty of Mechanical, Maritime and Materials Engineering, department of Maritime and Transport Technology*
Delft University of Technology, The Netherlands
*** ORTEC, Math Innovation Team, Zoetermeer, The Netherlands*

Abstract—**In the Netherlands, online groceries are becoming increasingly popular, as are the challenges grocery companies face in meeting customers' rising demand for smaller and cheaper time slots while maintaining thin profit margins due to a highly competitive market. Customer choice modelling will be used to identify customers' behaviour and control the trade-offs between customer attractiveness and profitability with demand management. As there are parametric and non-parametric models, including Machine Learning, to identify the behaviour, they will be compared to define which model represents customer behaviour best. Based on the F1-score, it is evaluated that the Multinomial logit (MNL) and Neural Network (NN) models outperform the created Benchmark model, derived from assumptions from the data, to ensure added value in predicting performance. Since it is found that the advanced predictive models can provide a better understanding of customer behaviour and identify critical customer characteristics, it will be tested whether these models enhance the simulation closer to reality or produce similar results to the Benchmark model, and if so, whether they can be used to optimise the offer strategy. In essence, does the learned behaviour impact the routes and number of offered slots, and can it be utilised to optimise the provided set of offers? A simulation tool is conducted to determine this, which uses real-selected customer time slots to assess any differences from reality and collect key performance indicators (KPIs) for evaluation. The results of the simulations show that incorporating advanced choice models in the simulations adds value and brings the outcomes closer to reality. As a result, a first attempt is made with two test scenarios to determine whether the advanced choice models can be used to optimise the offer strategy. Promising outcomes are found by analysing the results. However, further research is needed to assess the exact impact of the strategies. In future research, it is recommended to determine the influence of using more historical data and data from other companies in combination with other choice models. Also, it is recommended to consider different strategies for optimising the offer set based on incentives or other features.**

Keywords—Customer choice modelling, Customer behaviour, parametric models, non-parametric models, Machine learning, SHAP values, Multinomial logit, Neural network

I. Introduction

The demand for online grocery shopping has increased significantly in recent years, primarily driven by the desire for convenience and time savings among customers [1]. Online grocery shopping eliminates the need to factor in store opening hours, allowing customers to purchase groceries anytime and anywhere, making it an effortless and time-efficient experience [2]. To further enhance this experience, companies offer Attended Home Delivery (AHD) services, where groceries are delivered to the customer's door at a pre-arranged time slot of their choosing to ensure product quality. Nonetheless, allowing customers to select their preferred time slots poses logistical challenges for the companies, as the chosen slots can directly impact operational efficiency. To maintain control over the process, companies determine which time slots to offer and for what price. Regardless a more extensive offer set with cheaper and shorter time slots is associated with more convenience but adversely affects the delivery's execution costs. Therefore, the company has to face complex trade-offs between customer preferences and operational efficiency as it can temporarily close popular time slots. Doing this can reduce customer attractiveness as it does not meet their expectations. Nevertheless, AHD also benefits by reducing delivery failures by ensuring customers are home during the delivery window [3]. To generate suitable time slots and to minimise the attractiveness reduction, the following three conditions need to be considered:

1) The potential of acceptance by the customer has the highest priority, meaning that an empty offer set should be avoided as it will lead to disappointment.
2) The collection of time slots should deviate as little as possible from the customer's preferences regarding the time of the day.
3) The length of the offered time slots is crucial as length and availability affect customer satisfaction and, thus, suitability [4][5].

As above-mentioned, trade-offs must be made between customer preferences and efficiency to manage profitably and suitability. Customers can be nudged by influencing their behaviour and convincing them to select other more cost-efficient slots when needed [4]. Demand management can be used to seek cost-efficient fulfilment on the supply side and focuses, on the other hand, on meeting customer demand in the most efficient way possible. However, customer choice behaviour needs to be known to do this and can be obtained with customer choice models. These models are commonly used in retail for various purposes, including assortment decisions, pricing, and profit optimisation. For these goals, easily understandable parametric models are often used to give insights into the characteristics that influence customer behaviour. Although, these are more often redeemed for more advanced non-parametric data-driven models, including Machine Learning (ML), as they have a higher accuracy and predictive power. Despite that, these ML models are commonly used as black boxes, aiming to maximise prediction accuracy rather than understanding the underlying motivations behind customer choice behaviour. Nevertheless, there is value in understanding the main drivers of the customer-choosing process.

Based on these findings and that prior research primarily focused on the supply chain, this research will address the following question: *How can customer choice behaviour be better identified and used to optimise the provided offer set in the context of online groceries?* Four subquestions are formulated to explore and clarify different characteristics, aiming to break down the primary question and provide a comprehensive answer. The subquestions are:

- *What is customer choice behaviour in Attended Home Delivery, and how can it be identified?*
- *Which customer characteristics influence the behaviour the most?*
- *What is the difference between modelling with Machine Learning models and parametric models?*
- *Does the learned behaviour affect the simulation's routes and the number of offered slots?*

As above-mentioned, earlier research mainly focused on the supply chain, but after a recent shift, now its attention is on demand management and modelling customer behaviour. Nevertheless, the currently deployed customer choice models in retail are often simplistic. Therefore, this research seeks to extend the existing literature by using advanced customer choice models to model customer behaviour in the retail sector while providing insights into the model and identifying which features are essential.

The following sections of this review are organised as follows. In section II, the information found in literature is gathered to provide an understanding of how the process works and how the behaviour of customers selecting a time slot can be modelled. The used customer choice models will be further explained in section III, where the used Benchmark will also be introduced to ensure added value of the more advanced models. In section IV, a case study is performed where the customer choice models are included in ORTEC's events-based simulation to evaluate if the models add value and truly enhance the simulation to reality compared to the Benchmark model. Additionally, it analyses if the models can be used to optimise the provided set of offers based on two new strategies. Section V contains the conclusions and recommendations for future research.

II. Literature review

The literature review is divided into two parts. The first part focuses on how the online grocery process works, and the second part focuses on how customer behaviour can be modelled. The methodology for identifying customer behaviour will be determined using the found literature and will be applied in the case study.

A. *Order process*

The research focus has shifted from supply chain to demand management, leading to advanced technologies for a better understanding of customer behaviour. In the retail sector, these new technologies that identify behaviour better have already led to different service and offer strategies since a company cannot excel in all key performance indicators (KPIs) [6]. Demand management can be employed for these trade-offs as it maximises the overall profit, and the planning in AHD can be seen as an assortment of delivery options only with an impact on the delivery costs [7]. To manage the trade-offs, it will intend to shape and generate customer demand to benefit the fulfilment process. In this context, the most significant part of the fulfilment process is the customer's order decoupling point, which includes three primary steps: order capture, assembly, and delivery [8]. During order capture, customers select their preferred time slot from different time slots provided by the company. This booking process needs to be smooth, so the company must provide the time slots in at least a few seconds, after which the customer can select the preferred one [9]. Assembly is scheduled once an order is placed, followed by delivery within the selected time window. See Figure 1 for an overview of the possible ordering process.

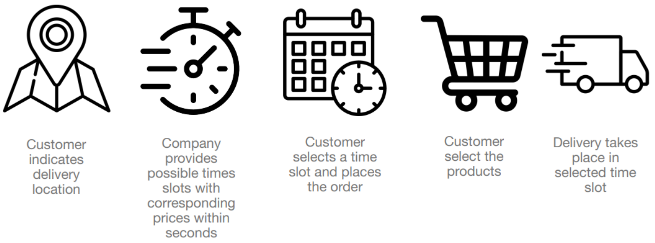

Customer indicates delivery location
Company provides possible times slots with corresponding prices within seconds
Customer selects a time slot and places the order
Customer select the products
Delivery takes place in selected time slot

Fig. 1. Ordering process from a customer perspective.

The first step in the ordering process shown in the figure is for the customer to specify the delivery location. Then, the company offers feasible delivery time slots with corresponding prices. The customer selects a time slot and orders, after which products are added to the online basket. This step can be conducted during the selling horizon, where a finite set of products is offered to heterogeneous customers [10]. Once the customer has selected the products, the company confirms the delivery, starts the assembly and prepares for delivery in the predetermined time slot [3]. However, the order process may vary depending on the company, as some start with the customer selecting products instead of a time slot.

Availability of time slots for delivery is checked first by anticipating and rapid assessment of the delivery step of the order based on a vehicle routing problem with time windows to see whether a particular time slot is feasible given already accepted customers, which may result in some customers being shown different options. Additionally, the availability depends on the company's KPIs and factors, such as the potential profit from the customer's request, their expected behaviour, and the opportunity costs of fulfilling the request at a particular time slot. After all, it may be more advantageous to reserve a specific time slot for more attractive future customers or direct the customer to a more suitable option.

Demand management in online grocery involves determining the time slots and the corresponding prices for delivery, which can be dynamic or static. Dynamic slotting and pricing refer to the process of making choices about which time slots to offer and at what price during the booking process, while static slotting is when these choices are determined prior to the booking horizon based on previous data and are not updated during the process [11]. In other words, dynamic slotting can use incentives to balance demand over the week and day based on popularity, while static slotting relies on predetermined data. The number of offered time slots might be affected by demand volumes, and geographic areas with low demand may receive fewer time slots to maintain efficient delivery routes.

Demand management aims to maximise profits by exploiting market heterogeneities. Depending on the heterogeneity, the market can be partitioned into segments with various sensitivities and preferences, which is already done in other sectors to optimise the offer strategy [12][13]. Based on demand management, the offered time slots to customers and price need to be determined. Where the geographic demand is comprehensible, as a minimum demand can be required to justify the area, determining the available capacity is less evident than initially appearing. Within the capacity also, the picking capacity in the warehouse and available driving time are included next to the physical fleet size. Resulting in that clustering orders is directly linked to transportation planning. That is only part of it, as with demand management, the capacity is not sold with a first-come-first-serve mentality, and the segments based on heterogeneity require more differentiation between orders [14]. Another essential tool for demand management is pricing, but determining the magnitude of discounts and premiums can be challenging as unexpected changes are received as unfair. On the contrary, customers will learn to anticipate limiting the effect when they follow a regular pattern. Nevertheless, demand management may offer smaller and cheaper time slots without affecting efficiency or the risk of failed deliveries. A good understanding of customer behaviour is needed, which can be obtained with customer choice models.

B. Customer choice model

To comprehend customer behaviour and preferences for selecting one option among several alternatives, discrete choice models (DCMs) are used. Understanding customer behaviour can help a company maintain efficiency while ensuring customer satisfaction and attractiveness [15]. DCMs are widely used in various sectors, such as marketing, economics and operational studies. With the growth in online retail, the use of DCMs to model customer choice behaviour has increased in recent years. It can help identify behaviour and make efficient pricing and revenue management decisions. It is a scientific approach to finding critical market drivers by analysing choices made by customers among various options. This involves measuring customer preferences and identifying patterns between the offers and choices, resulting in a better understanding of customer behaviour [16][17][18]. During this process, customers are assumed to see all the options together at a certain time and decide based on preferences[19].

Two models utilised to predict customer behaviour by incorporating utility are the parametric model, which is commonly used, and the increasingly common non-parametric model [20]. The parametric model is embedded in random utility theory and characterised by simplifying the function to a known form where the data is summarised through a fixed amount of parameters. Random utility theory suggests customers associate a particular utility with each product and make decisions based on maximising utility. The various time slots in this study are the products, and a DCM can be derived. The utility of a choice option consists of a deterministic component, u_j, which represents the mean utility of the alternative, and a random component, ϵ_j, with a zero mean. These two components combine to form the overall utility, as shown in Equation1. The notation j represents one of the alternatives from a set of products presented to a customer, while l denotes a specific segment within the customer population, which is assumed to comprise $\mathcal{L} := 1, ..., L$ segments.

$$U_j^l = u_j^l + \epsilon_j^l \tag{1}$$

The probability of a customer choosing a product j can be calculated using $P_j(S) = P\left(U_j = max\left\{U_{j'} : j' \in S \cup \{0\}\right\}\right)$, where S indicates the offer set $S \subseteq \mathcal{J}(c_t)$ with c_t the available inventory at time t. It should be noted that there is a possibility that a customer may not make a purchase, which is represented by U_0 [10].

Various parametric models, including Multinomial Logit (MNL), Nested Logit, Generalized Attraction, and Markov Chain choice model, are available for modelling customer behaviour. However, in this research, the focus is on the widely used utility-based parametric MNL model. The MNL models are applied in different sectors such as the retail, airline and travel industries to model customers' behaviour to, for example, optimise the profit with time slot pricing [21]. The benefit of using a theory-based parametric model is that it enables the extrapolation of choice predictions to unobserved alternatives, and the estimated weights related to the characteristics can be obtained directly. However, the significance of the model's parameters is only valid if it is assumed that the theory and the model are appropriately specified [22][23][24].

Parametric models require a fixed set of parameter specifications based on the assumed data distribution for making predictions. In contrast, non-parametric models do not rely on specific parameter settings and choose a functional form based on training data [25]. Non-parametric models, such as ML models, are becoming increasingly popular due to their ability to increase predictive power and create non-linear relationships between characteristics. However, ML models can be perceived as black boxes since the result are not directly interpretable [26]. For modelling customer behaviour, it is of utmost importance to understand the estimated parameters [27]. To address this, SHaply Additive exPlanation (SHAP) values are used to interpret ML models. The SHAP value is proposed for model interpretability by clarifying the ML model with a unified approach where the collective SHAP values can indicate how much each feature contributes to the target variable [28][29].

This research will employ ML as a classification model for a supervised learning problem. Various ML models, such as Random Forest, k-Nearest Neighbor, Support Vector Machine, Bayesian Network, and Neural Network (NN), have been proposed for analysing customer behaviour. However, not all models are equally effective, and as a result, the NN model with multiple hidden layers has been selected based on its accuracy and applicability in other sectors. The NN model is based on the biological brain and requires a large training dataset. The data is then summarised, processed, and weighted through connections to generate a network output [30]. The NN model includes an input layer, one or more hidden layers consisting of neurons, and a final layer of output neurons. The input layer represents the independent variable, such as the alternatives' attributes, customers' characteristics, and contextual factors, and the output layer represents the choice probabilities of all options. Between these layers, the hidden layer connects the input and output layers, and when the model consists of four or more layers, it is referred to as a deep NN [31][32]. The number of neurons included in the NN model depends on the complexity, and the number of neurons in each layer will be determined during the hyperparameter tuning. Since there is no standard structure for the NN model, the optimal number of neurons in each layer will be determined based on the results of the tuning process.

III. Customer choice modelling

With more companies and customers' preferences entering the highly competitive industry, customer segmentation is encouraged to ensure attractiveness by offering the right time slots. In this research, customer segmentation will be carefully done as the behaviour is assumed to be homogeneous within a segment and significantly impacts model performance. However, companies may reject models based on segmentation that perform price discrimination, for example, and prefer more refined segmentation techniques [20]. Customer segmentation is achieved in this research by assigning a feature value based on demographic and historical characteristics. In addition, the increasing amount of accessible customer information has made it necessary to regulate and identify a selection as not all features are essential to model behaviour [33]. Selecting features can be done based on classification techniques categorised in wrapper, filter, embedded and hybrid methods [34][35]. A combination of filter and wrapper techniques is chosen in this research to reduce the dimensionality of data and improve computational efficiency by eliminating irrelevant features during pre-processing. Moreover, when the MNL and NN models are utilised, they assign weights to the features during training to achieve optimal classification performance and enhance model interpretation. The filtering process of features is performed just before hyperparameter tuning, which involves testing and evaluating multiple combinations of hyperparameters to find the optimal architecture resulting in the best model performance. The tuned hyperparameters involve identifying the appropriate solver and activation function, determining whether to include balanced class weights and specifying the number of hidden layers and their width when applicable. One must compare activation functions and solvers appropriate for multiclass problems, as there is no single best choice for all optimisation problems. The MNL model tests commonly used solvers, including newton-cg, lbfgs, sag, and saga, and the NN model tests activation functions such as ReLu, ELU, Tanh, and Sigmoid. However, the NN's output layer will consist of Softmax. To determine the best architecture, the performance of multiple combinations will be evaluated based on the performance metrics, including the F1-score. Based on these performances, different structures are found for the models, as the prediction process consists of three steps.

The customer's preferred time window will be estimated using three distinct models. First, the MNL and NN model will predict the delivery week, followed by the day and the time of day. Because when the delivery moment is predicted at once, it will lead to a rapid decline in prediction performance due to many output classes. A Benchmark model is created to ensure that the more advanced MNL and NN models contribute to the prediction performance. This model is divided into the same three prediction steps to establish if all three models add value and are created based on the most commonly chosen option for that ordering moment.

	Benchmark	MNL	NN
Delivery week	0.420	0.521	0.695
Delivery day	0.123	0.743	0.797
Part of the delivery day	0.127	0.333	0.368

TABLE I

F1-scores of the Benchmark, MNL and NN models for evaluating the prediction performance.

The comparison of the Benchmark model results with the predictions of the MNL and NN models based on the F1-score reveals that both advanced models enhance the performance of predictions. In other words, using the MNL and NN models enables better identification of customer behaviour compared to the Benchmark model. Nevertheless, it is found that the NN has the best prediction performance. Table I shows an overview of the results. Understanding the estimated parameters of the choice models is crucial; therefore, feature importance plots have been generated to gain more insight into the model. As mentioned, the importance cannot be directly obtained from ML models. However, in this research, SHAP values were introduced to obtain them. By evaluating the feature importance, it can be noted that the models show different rankings. For example, the MNL model for predicting the delivery day prioritises the number of available slots, whereas, in contrast, the NN model emphasises the average weight of the order. Besides, the importance of the final prediction step also differs as the MNL model gives higher weights to the number of available slots again. At the same time, the NN focused more on the features containing history. Both models have the same order of feature importance for predicting the delivery week. That the models distribute their importance differently over the same features was noted since both models have different assumptions and learning mechanisms, resulting in different feature weights. The difference is, for example, that the MNL model presumes that the relationship between the independent and dependent variables is linear. In contrast, the NN models can also model non-linear relationships.

Considering that the historical features were emphasised for the last prediction step, the models were also performed with only customer data who ordered before, as it is possible that the models underestimated the importance of historical features. The reason for assuming that the MNL and NN models rely more on consistently available features across all customers is that they assign weights to features based on their ability to predict the outcome of interest. However, since this study only has data from November, historical information on most customers is not available. Adding more historical information could improve the prediction performance based on the new models' prediction, particularly for predicting the delivery day part. Table II shows an overview of the results. The current lack of historical features in the process may be the reason for the delivery day prediction's almost unchanged performance.

	Benchmark	MNL	NN
Delivery day	0.210	0.733	0.801
Part of the delivery day	0.166	0.500	0.549

TABLE II

F1-SCORES OF SIMULATIONS EVALUATING THE PREDICTION PERFORMANCE WITH ONLY CUSTOMERS THAT ORDERED IN WEEKS BEFORE.

IV. CASE STUDY: ORTEC

The results show that the MNL and NN models have advanced predictive performance, better understand customer behaviour than the Benchmark model, and can consequently identify critical customer characteristics. Nevertheless, it remains to be determined whether these models add value and bring the simulation closer to reality or whether the results are comparable to those of the Benchmark model. The simulation uses actual customer slots from one delivery week to assess these differences using KPIs obtained during the simulation. The KPIs include the number of time slots offered per customer, the total route execution time, and the number of people assigned to assess performance.

ORTEC's event-based simulation tool, consisting of a time slotting and routing optimisation process, will be deployed with an instance generator and event simulator. Figure 2 shows an overview of the process. For this research, actual customer data from an online grocery company is used in the instance generator to determine the influence of including the Benchmark, MNL and NN in the simulation focusing on one delivery week. The event-based simulator processes customer events in chronological order and assumes that between two consecutive occasions, no changes occur in the system. It allows the simulator to jump to the next event directly, avoiding wasting computational resources on idle periods. Simulations are performed where the MNL, NN and Benchmark models are separately included to analyse whether the MNL and NN models bring the outcomes closer to reality. The outcomes will be evaluated and compared with results from a simulation performed with the real-chosen time slots. This simulation is considered necessary because the time slotter or route optimiser may be updated between the new simulations and when the orders are placed, affecting the KPIs. An improved simulation by including the MNL and NN can be indicated when the KPIs are closer to the results of the real-selected simulation than using the Benchmark models, and an overview of these results is displayed in Table III.

	Real model	Benchmark model	NN	MNL
Average number of seen slots	143.496	140.639	107.965	97.897
Average duration of the routes [s]	43496	15554	39671	49601
Total assigned customers	12047	7382	12897	12904
Average number of customers per route	30.678	17.494	22.968	23.014
Average duration per customer [s]	1417	893	1728	2154
Average route cost per customer	6.524	11.275	8.708	8.690
Average execution cost per customer	11.806	7.443	14.397	17.947

TABLE III

OVERVIEW OF THE KPIS FROM THE SIMULATIONS PERFORMED WITH THE REAL-SELECTED TIME SLOTS, BENCHMARK, NN AND MNL MODELS.

The results in Table III need to be evaluated together as each KPI provides valuable insights into different aspects of the outcome. When analysing the set of KPIs, they indicate that the MNL and NN models positively impact the simulation and bring it closer to reality than the Benchmark model as the results are closer to the real simulation. When analysing the results in more detail, it can be seen that the real-selected time slots and the Benchmark model offer the most time slots, while simulations with more advanced customers offered fewer time slots. However, the MNL and NN models predicted that more customers would be served, possibly due to the used features indicating the time slots offered for the weekend before, Monday, and Tuesday after the delivery week, since these do not change as the simulation focuses on one delivery week or because of the earlier availability of all time slots in the simulation. The analysis also revealed that the initial cost per route decreases as the number of customers on a route increases, resulting in lower costs per customer. Nevertheless, simulations with fewer customers on a route showed lower execution costs per customer, which may seem counterintuitive at first glance. In this case, the higher costs can be attributed to the fact that variable costs do not occur pro-rata and depend on the resources, including order size, execution time, and distance to travel to the customer. As a result, the higher average duration per customer and the less efficient routes created may contribute to higher costs. However, the route's profitability may still be positive due to the operating margin, which can be calculated when the revenues, costs and fees are known. In addition, assigning more customers for strategic reasons, even if it leads to losses, can be a worthwhile investment in retaining customers

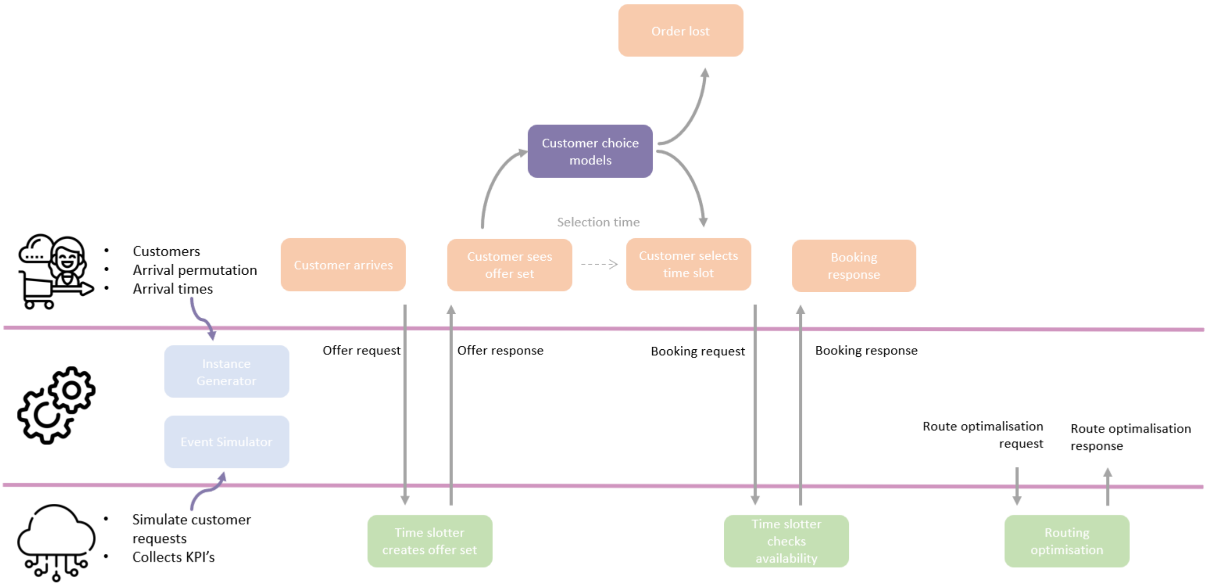


Fig. 2. Ordering process from a customer perspective.

and maintaining competitiveness.

Now that it is indicated that the MNL and NN models improve the prediction performance in identifying customer behaviour and that the simulation results demonstrate that utilising these models brings the simulation closer to reality than the Benchmark model, two test scenarios are created. With these two test scenarios, it can be investigated whether these more advanced models confer additional value in optimising the offered sets. The first test scenario considers the expenses related to the company's execution in conjunction with the probabilities the choice models assign to different time slots. This test scenario determines if excluding time slots with high costs and low selection probabilities will affect customers' attractiveness and KPIs. The second test scenario solely focuses on the assigned probabilities to the time slots. Both scenarios will be run twice, once based on twenty per cent and once based on thirty per cent. This means that for the first scenario, the time slots among the top twenty most expensive and belonging to the lowest twenty per cent based on probability will be left out, and the same is done only based on thirty per cent. For the second scenario, this means that time slots with a probability belonging to the lowest twenty per cent are removed from the offer set and repeated for thirty per cent. An overview of the created test scenarios can be found in Table IV. It is important to note that during the optimisation process, customer choice models were utilised to adjust the offer set and select the time slots. However, utilising the same model to adjust different components of the optimisation process makes it challenging to compare the results. Although it is challenging, this approach was chosen. Potentially, a future study that builds on the results of this study could be extended on this. In order to determine whether the advanced choice models can be employed to optimise the offer set in this research, the same KPIs will be used to compare the results with the ones of the real-selected time slot stimulation. An overview of all the results can be found in Table V.

Test scenario	Considerations	Model	Removed base on	Name
1	Execution costs and probabilities of selecting a time slot are considered. Time slots with high execution costs and low probabilities of selection are excluded to reduce executive costs.	MNL	20% lowest probabilities 20% highest cost	MNL 20/20
			30% lowest probabilities 30% highest cost	MNL 30/30
		NN	20% lowest probabilities 20% highest cost	NN 20/20
			30% lowest probabilities 30% highest cost	NN 30/30
2	Only probabilities of selecting a time slot are considered. All time slots with lowest likelihood of selection are discarded to reduce the offer set size.	MNL	20% lowest probabilities	MNL 20
			30% lowest probabilities	MNL 30
		NN	20% lowest probabilities	NN 20
			30% lowest probabilities	NN 30

TABLE IV

OVERVIEW OF THE CREATED TEST SCENARIOS.

Table V shows that the MNL 30, NN 20 and NN 30 simulations have significantly lower customer assignment, relatively high execution and route costs, and duration per customer, as well as a low number of customers per route compared to the simulation with real-selected time slots. As a result, the second test scenarios can be considered not suitable for optimising this research's offering strategy. It is worth noting that the assumed sub-optimal strategies have a low average duration of the route and offer a relatively large number of time slots compared to the real results. This might be because the number of offered time slots does not reduce further due to the low number of customers served, which means fewer time slots are occupied and are only affected by the strategy of removing time slots. The duration of the route might be low since more efficient routes can be created when fewer customers need to be served. In the simulations applying the first scenario, the average number of offered slots and assigned customers are higher when thirty per cent is eliminated compared to the test where only twenty per cent is eliminated. A possible explanation can be that removing more time slots may lead to a more balanced offer set, as it may eliminate not only time slots with positive influence but also those with negative influence on the probability of order placement. To determine whether this is the reason, more simulations need to be performed with a combination of different choice models to define the influences better. However, the remaining simulation results of the test scenarios can be compared separately with the actual results. The outcomes of the NN 30/30 simulation indicate a slight decrease in the number of assigned customers and a comparatively more substantial reduction in the average duration of routes, making it a better choice for enhancing

	Real model	MNL 20/20	MNL 30/30	NN 20/20	NN 30/30	MNL 20	MNL 30	NN 20	NN 30
Average number of seen slots	143.496	98.958	117.913	111.549	114.976	96.631	87.497	98.997	86.624
Average duration of the route [s]	43496	49169	49899	36091	32957	37862	29091	38926	28274
Total assigned customers	12047	12630	13125	11504	11757	9139	7344	4780	4773
Average number of customers per route	30.678	22.511	23.395	22.139	24.415	20.643	16.010	16.366	14.924
Average duration per customer [s]	1417	2181	2134	1622	1351	1867	2462	3220	2945
Average route cost per customer	6.524	8.885	8.549	8.428	8.192	8.527	9.029	7.462	7.479
Average execution cost per customer	11.806	18.177	17.786	13.518	11.260	15.559	20.520	26.829	24.544

TABLE V

OVERVIEW OF THE KPI RESULTS FROM THE SIMULATIONS PERFORMED FOR ALL TEST SCENARIOS.

the offering strategy. The average execution cost and duration per customer also show a decrease. However, the MNL 20/20, MNL 30/30, and MNL 30 strategies resulted in the assignment of more customers, but at the cost of increased average route duration and decreased number of customers per route.

Based on the results of the test scenarios and the fact that the first scenario reveals some interesting results, an attempt is made to compare the route-related results with the outcomes of the simulation in which no offer strategy is applied, using the same customer choice model. Given previous outcomes demonstrating that fewer served customers may result in more efficient route generation, it is essential to consider the number of customers assigned. To start with simulations using the MNL from which the results are displayed in Table VI. The analysis of the results shows that the first scenario, excluding the thirty per cent most expensive slots when they are also among the bottom thirty per cent in terms of probability of being selected, can lead to the most considerable improvement. This may be due to a more balanced offer set, which could benefit the customer choice model. However, further simulations are needed to confirm this. With this strategy, the total and average number of customers per route increased while the average duration and total average cost decreased. The total average cost is the sum of the average route and execution cost per customer. Improvement in revenue, cost, and delivery fees could provide more insight into the strategy's effects.

	MNL	MNL 20/20	MNL 30/30
Total assigned customers	12904	12630	13125
Average number of customers per route	23.014	22.511	23.395
Average duration per customer [s]	2154	2181	2134
Total average cost per customer	26.637	27.062	26.335

TABLE VI

SUMMARY OVERVIEW OF THE RESULTS FORM THE SIMULATIONS WITH AND WITHOUT OFFERING STRATEGY UTILISING THE MNL MODEL.

Additionally, the same comparison of the first test scenario is made using the NN model, of which the results are visualised in Table VII. The displayed results indicate that the tested strategies decreased the number of assigned customers, which resulted in more efficient routes with a reduction in average duration per customer compared to the simulation without a strategy. Moreover, the NN 30/30 model showed a greater reduction in average duration while serving more customers than the NN 20/20. Surprisingly, the total average cost per customer decreased despite the expectation that both the average route and execution costs would increase. This may be due to the strategy removing the most expensive time slots, which could take up the most time. Further investigation of revenues, costs, and fees is necessary to determine the strategy's overall profitability.

Analysing the results of the simulations, it evaluated that the MNL and NN models may optimise the offer strategy, with the NN 30/30 strategy showing the best results. Since the most potential is seen in the first test scenario, based on results, the route-related outcomes of these test scenarios are compared with the outcomes of simulations utilising the same model but without a modified offer set. When evaluating the results, it is indicated that the average duration and total average cost per customer are reduced, possibly due to the applied strategy, possibly because the most expensive time slots take the most time to execute and have a low chance of selection removed. However, further simulations using different choice models, exploring different probabilities or strategies based on different features, and training customer choice models again with fewer time slots are necessary to determine the precise impact of the offer strategy. It is important to note that this research is an initial exploration, and future studies can compare different approaches in different ways.

	NN	NN 20/20	NN 30/30
Total assigned customers	12897	11504	11757
Average number of customers per route	22.968	22.139	24.415
Average duration per customer [s]	1728	1622	1351
Total average cost per customer	23.105	21.946	19.452

TABLE VII

SUMMARY OVERVIEW OF THE RESULTS FORM THE SIMULATIONS WITH AND WITHOUT OFFERING STRATEGY UTILISING THE NN MODEL.

V. CONCLUSIONS

This research determines whether customer choice behaviour can be better identified than a Benchmark model based on assumptions and used to optimise the provided offer set in the context of online groceries. Customer behaviour needs to be understood to optimise the interdependent process using demand management, as the aim is to manage the trade-offs between customer satisfaction and the company's logistics and profit. The behaviour can be obtained with parametric or non-parametric DCMs where previously parametric ones were most commonly used but are now replaced for non-parametric models. The non-parametric models such as ML are data-driven, making it easier to process the increasing amount of data but are harder to interpret. Therefore SHAP values will be derived to provide these insights. For this research, it is decided to employ the parametric MNL model and the non-parametric NN model with multiple hidden layers to analyse if it is possible to identify customer behaviour better than the Benchmark model. The models' prediction performance is compared based on the F1-score, and it is found that both the MNL and NN models outperform the Benchmark model, with the NN model having the best performance. This suggests that the NN and MNL model can better identify customer behaviour. When looking into detail at the results, it can be found that the MNL and NN consider different features important for predicting the delivery steps. The MNL model assigns the highest weight to the number of available slots, while the NN model considers the average weight of the order as the most influential. That the models assign different weights to the same features, even when trained on the same data, is expected as both models fundamentally differ with other assumptions and learning mechanisms. The importance of predicting the part of the delivery day suggests that historical information could improve prediction accuracy. Testing the influence of historical information, the prediction performance of both the MNL and NN models improved significantly in predicting the part of the delivery day. The findings recommend further exploration into ways to incorporate more historical data to enhance performance.

Since it is found that the MNL and NN models can better identify customer behaviour than the Benchmark model, the question remains whether these models add value and truly enhance ORTEC's event-based simulation closer to reality compared to the Benchmark model. Simulation results indicated that advanced models predicted more customers would be assigned despite the lower average number of offered time slots which was expected to reduce customers' attractiveness and satisfaction. Nevertheless, the MNL and NN models add value and enhance the simulation closer to reality than the Benchmark model. Consequently, two test scenarios are created to compare the KPIs and evaluate whether the models can be utilised to optimise the provided offer set. When analysing the KPIs, it is evaluated that the first test scenario indicates the most potential. Further evaluation by comparing the results of this test scenario with simulations using the same customer choice model but without applying a strategy showed that in the simulation applying the strategy reduced the average duration and total average cost per customer, potentially due to removing the most expensive and time-consuming time slots that had a low chance of being selected. However, more simulations using other choice models are necessary to confirm these findings. Additionally, other simulations could explore strategies based on different features or the impact of price incentives.

This research found that customer behaviour can be better identified using MNL and NN models since both outperform the Benchmark model in predicting customer behaviour. Moreover, the NN model showed the best performance. Based on these results, simulations are performed to determine whether ORTEC's simulation tool can be enhanced to be more realistic using advanced models compared to a simulation utilising the Benchmark model. Since the MNL and NN models also add value to the simulation tool by bringing the results closer to reality, a first step in analysing whether the choice models can be used for strategies to optimise the offer set is made. Promising results are found, but further research is needed. Nevertheless, several recommendations can be made to improve the prediction performance and strategies to optimise the offer set in the context of online groceries. Firstly, using more historical customer data could improve prediction accuracy. Secondly, exploring different choice models like the Random Forest could improve prediction performance. Thirdly, incorporating data from other companies could enhance predictions and provide new insights. Because other companies can have other processes where, for example, the groceries are selected, and only then the delivery moment is selected. Besides, it might be advantageous to gain further insights by retraining the customer choice models based on simulation data and analysing which information may be lost. Fourthly, since the advanced models can identify customer behaviour better, the effect of incentives such as green labels or price changes could be indicated since this was not possible in this research due to the available data. Lastly, further testing can be performed to investigate whether alternative offering strategies based on other features can significantly improve the KPIs in the simulation.

References

[1] Tobias Gawor and Kai Hoberg. "Customers' valuation of time and convenience in e-fulfillment". In: *International Journal of Physical Distribution & Logistics Management* (2018).

[2] Valerio Gatta et al. "E-groceries: A channel choice analysis in Shanghai". In: *Sustainability* 13.7 (2021), p. 3625.

[3] Charlotte Köhler, Jan Fabian Ehmke, and Ann Melissa Campbell. "Flexible time window management for attended home deliveries". In: *Omega* 91 (2020), p. 102023.

[4] Niels Agatz et al. "Challenges and opportunities in attended home delivery". In: *The vehicle routing problem: Latest advances and new challenges* (2008), pp. 379–396.

[5] Marco Casazza, Alberto Ceselli, and Lucas Létocart. "Optimizing Time Slot Allocation in Single Operator Home Delivery Problems". In: *Operations Research Proceedings 2014*. Springer, 2016, pp. 91–97.

[6] Rohit Verma and Gerhard Plaschka. "Predicting customer choices: recent research has greatly improved management's ability to anticipate customer wants". In: *MIT Sloan Management Review* 47.1 (2005), pp. 7–11.

[7] Fernando Bernstein, Sajad Modaresi, and Denis Sauré. "A dynamic clustering approach to data-driven assortment personalization". In: *Management Science* 65.5 (2019), pp. 2095–2115.

[8] Ann Melissa Campbell and Martin WP Savelsbergh. "Decision support for consumer direct grocery initiatives". In: *Transportation Science* 39.3 (2005), pp. 313–327.

[9] Liana van der Hagen et al. "Machine Learning-Based Feasability Checks for Dynamic Time Slot Management". In: *Available at SSRN 4011237* (2022).

[10] Arne K Strauss, Robert Klein, and Claudius Steinhardt. "A review of choice-based revenue management: Theory and methods". In: *European journal of operational research* 271.2 (2018), pp. 375–387.

[11] Jochen Mackert. "Choice-based dynamic time slot management in attended home delivery". In: *Computers & Industrial Engineering* 129 (2019), pp. 333–345.

[12] Niels Agatz et al. "Demand management opportunities in e-fulfillment: What internet retailers can learn from revenue management". In: (2008).

[13] Niels Agatz et al. "Revenue management opportunities for Internet retailers". In: *Journal of Revenue and Pricing Management* 12.2 (2013), pp. 128–138.

[14] Arne Strauss, Nalan Gülpınar, and Yijun Zheng. "Dynamic pricing of flexible time slots for attended home delivery". In: *European Journal of Operational Research* 294.3 (2021), pp. 1022–1041.

[15] Niels Agatz et al. "Time slot management in attended home delivery". In: *Transportation Science* 45.3 (2011), pp. 435–449.

[16] Rohit Verma. "Customer choice modeling in hospitality services: A review of past research and discussion of some new applications". In: *Cornell Hospitality Quarterly* 51.4 (2010), pp. 470–478.

[17] Laura Patterson. *Predict Customer Choices – How Cool is That?!* Dutch. Sept. 7, 2021. URL: https://visionedgemarketing.com/choice-modelling-predicts-customer-preferences/ (visited on 10/03/2022).

[18] Raguzin et al. *READY FOOD*. Nov. 2020. URL: https://www.oliverwyman.com/content/dam/oliver-wyman/v2/publications/2020/November/Ready_Food.pdf (visited on 10/03/2022).

[19] Isha Gupta. *5 Tips for Consumer Choice Models - Towards Data Science*. Dutch. Dec. 14, 2021. URL: https://towardsdatascience.com/consumer-preference-models-85e297887b1b (visited on 10/03/2022).

[20] Xinan Yang et al. "Choice-based demand management and vehicle routing in e-fulfillment". In: *Transportation science* 50.2 (2016), pp. 473–488.

[21] Xinan Yang and Arne K Strauss. "An approximate dynamic programming approach to attended home delivery management". In: *European Journal of Operational Research* 263.3 (2017), pp. 935–945.

[22] Garrett Van Ryzin and Gustavo Vulcano. "A market discovery algorithm to estimate a general class of nonparametric choice models". In: *Management Science* 61.2 (2015), pp. 281–300.

[23] Alejandro Mottini and Rodrigo Acuna-Agost. "Deep choice model using pointer networks for airline itinerary prediction". In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 2017, pp. 1575–1583.

[24] S Van Cranenburgh et al. "Choice modelling in the age of machine learning". In: *arXiv preprint arXiv:2101.11948* (2021).

[25] Nicola Ortelli et al. "Assisted specification of discrete choice models". In: *Journal of choice modelling* 39 (2021), p. 100285.

[26] Brian Sifringer, Virginie Lurkin, and Alexandre Alahi. "Enhancing discrete choice models with neural networks". In: *Proceedings of the 18th Swiss Transport Research Conference (STRC), Monte Verità/Ascona, Switzerland*. 2018, pp. 16–18.

[27] Ayuya. *Parametric versus Non-Parametric Models*. Feb. 2021. URL: https://www.section.io/engineering-education/parametric-vs-nonparametric/.

[28] Scott M Lundberg and Su-In Lee. "A unified approach to interpreting model predictions". In: *Advances in neural information processing systems* 30 (2017).

[29] Kuo. *Explain Your Model with the SHAP Values - Dataman in AI*. Sept. 2014. URL: https://medium.com/dataman-in-ai/explain-your-model-with-the-shap-values-bc36aac4de3d.

[30] Julian Hagenauer and Marco Helbich. "A comparative study of machine learning classifiers for modeling travel mode choice". In: *Expert Systems with Applications* 78 (2017), pp. 273–282.

[31] Ahmad Alwosheel, Sander van Cranenburgh, and Caspar G Chorus. "Is your dataset big enough? Sample size requirements when using artificial neural networks for discrete choice analysis". In: *Journal of choice modelling* 28 (2018), pp. 167–182.

[32] Renzhi Lu and Seung Ho Hong. "Incentive-based demand response for smart grid with reinforcement learning and deep neural network". In: *Applied energy* 236 (2019), pp. 937–949.

[33] Sahar F Sabbeh. "Machine-learning techniques for customer retention: A comparative study". In: *International Journal of Advanced Computer Science and Applications* 9.2 (2018).

[34] Gupta. *Feature Selection Techniques in Machine Learning*. Dec. 2020. URL: https://www.analyticsvidhya.com/blog/2020/10/feature-selection-techniques-in-machine-learning/.

[35] Brownlee. *How to Choose a Feature Selection Method For Machine Learning*. Nov. 2019. URL: https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/.

B

Benchmark model

B.1. Delivery week Benchmark model

The first created Benchmark model is the model for predicting the delivery week. To create the Benchmark model the most common delivery week for all request weeks is indicated. After indicating the most common delivery week it is assumed that the predcition of the delivery week equals the most common option. The results of the most common delivery week per request week can be found in .

Request week	Most common delivery week
44	44
45	45
46	46
47	47
48	48

Table B.1: Most common delivery week per request week

B.2. Delivery day Benchmark model

When no additional assumptions are made regarding the number of lost orders, it is found that the most common delivery day for each request day is No delivery. However, when leaving out the option that the order is lost, the most common delivery day of each request day can be found in Table B.2. The results of Table B.2 are used for creating the Benchmark model as it is assumed that one-third of the time the request results in a delivery instead of being assumed to be lost.

Request day	Most common delivery day
Monday	Tuesday
Tuesday	Wednesday
Wednesday	Friday
Thursday	Friday
Friday	Saturday
Saturday	Sunday
Sunday	Monday

Table B.2: Most common selected delivery day for all request days.

B.3. Delivery part of the day benchmark model

The most common part of the delivery day for all parts of the requested day is the Morning. An overview of these results can be found in Table B.3.

Part of requested day	Most common part of the delivery day
Early morning	Morning
Morning	Morning
Noon	Morning
Eve	Morning
Night	Morning
Late night	Morning

Table B.3: Most commonly selected part of the delivery day for all parts of the requested days.

C

Graphical analysis of request and order data

C.1. Data extraction

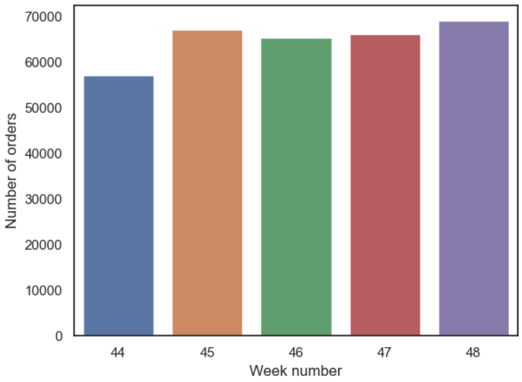


Figure C.1: Orders per week before filtering

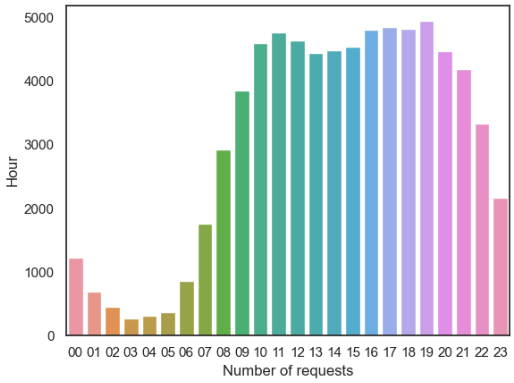


Figure C.2: Total number of requests made per hour

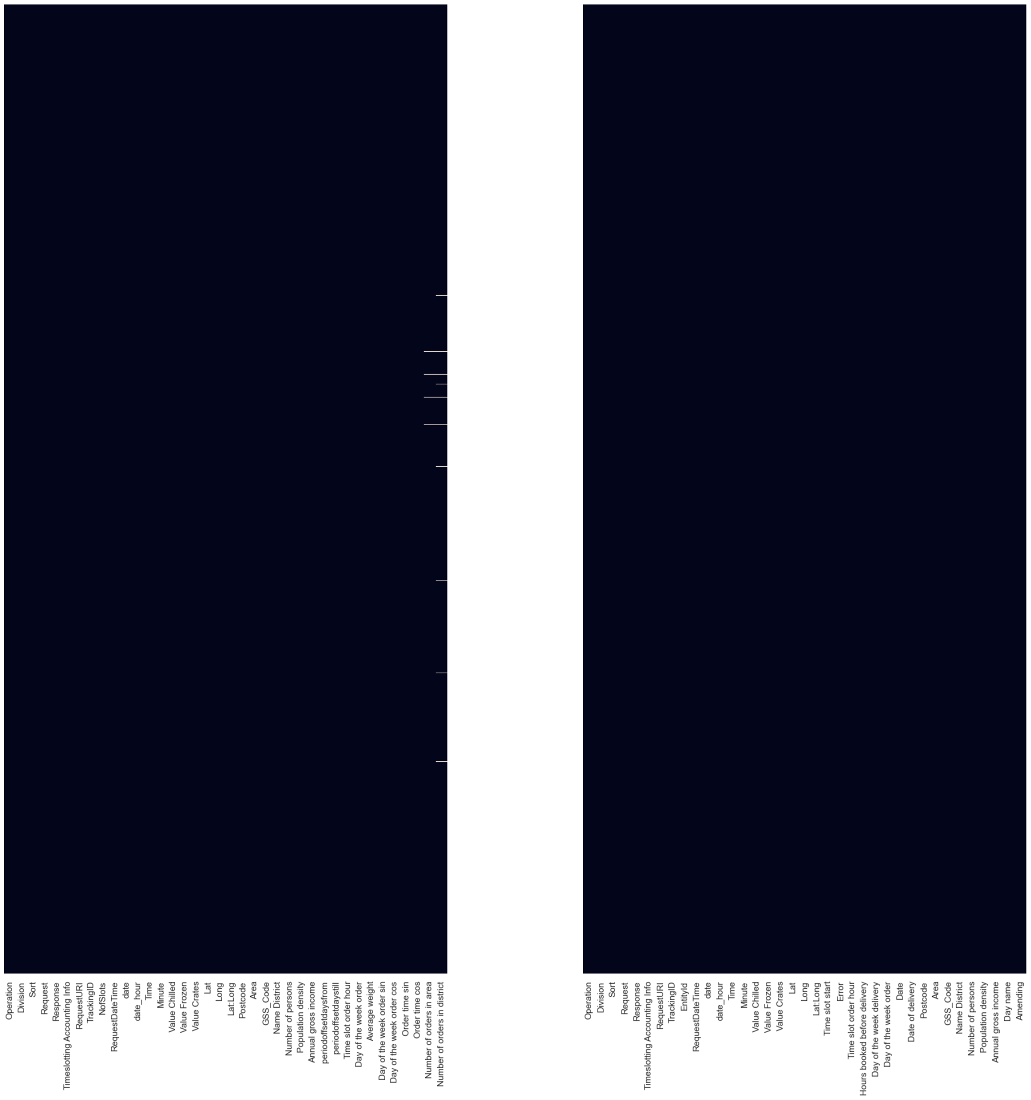


Figure C.3: Heatmaps of missing data points in the request and order data, respectively.

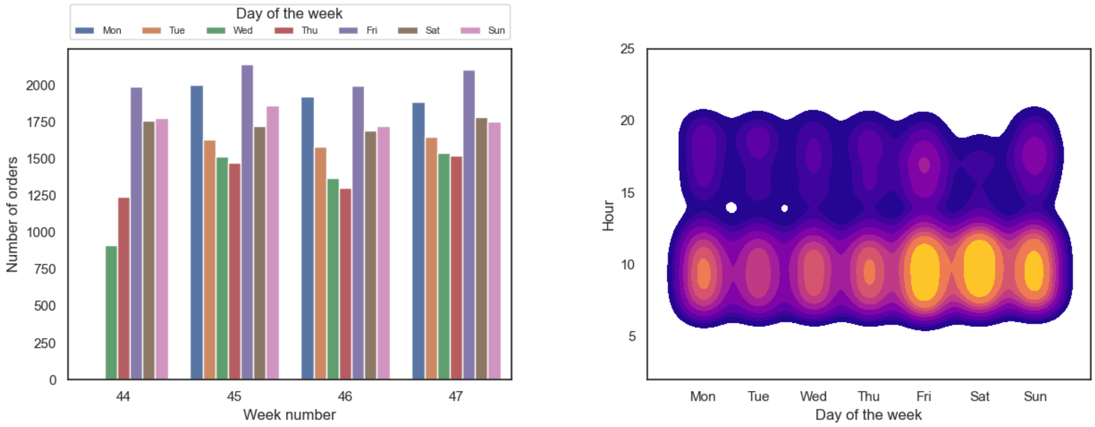


Figure C.4: On the left side, the distribution of deliveries before week 48 per day can be found. On the right side, the activity of the deliveries is visualised using a kernel density estimate plot.

D
Feature overview

Here an overview of all available features can be found from the combined data set of the orders and requests and the lost order.

- Lat:Long
- EntityId
- Selected Timeslot
- Timeslot length
- Number of offered slots
- Offer set
- Time
- Amount of days in set
- Order time
- Request time
- Value order
- Average weight order
- Value Chilled
- Value Frozen
- Value Crates
- TrackingID order
- TrackingID request
- Time slot start hour
- Hours booked before
- Delivery day
- Delivery date
- Order day
- Postcode
- Population
- Annual gross income
- History orders in area
- History orders in district
- History orders at location
- Start time of last delivery
- Delivery day of last delivery
- Average days between delivery
- Amount of offers for delivery day
- Consists time slot
- check date
- check time
- Number of Late Night offers
- Number of Early Morning offers
- Number of Morning offers
- Number of Noon offers
- Number of Eve offers
- Number of Night offers
- Delta cost
- Date last delivery
- Slots for Sun
- Slots for Mon
- Slots for Tue
- Slots for Wed
- Slots for Thu
- Slots for Fri
- Slots for Sat
- Slots for Sat before
- Slots for Sun before
- Slots for Mon after
- Slots for Tue after
- Most common delivery day of district
- Most common delivery moment of district
- History hours booked before of district
- Most common delivery day at location
- Most common delivery moment at location
- History hours booked before at location

- Day of the week delivery sin
- Day of the week delivery cos
- Day of the week order sin
- Day of the week order cos
- Order time sin
- Order time cos
- Last start of delivery sin
- Last start of delivery cos
- Day name
- Part of the day
- Parts of the day
- Last delivery Parts of the day
- Last delivery Early Morning
- Last delivery Morning
- Last delivery Noon
- Last delivery Eve
- Last delivery Night
- Delivery on Fri
- Delivery on Mon
- Delivery on Sat
- Delivery on Sun
- Delivery on Thu
- Delivery on Tue
- Delivery on Wed
- Delivery Early Morning
- Delivery Morning
- Delivery Noon
- Delivery Eve
- Delivery Night
- Last delivery on Sun
- Last delivery on Mon
- Last delivery on Tue
- Last delivery on Wed
- Last delivery on Thu
- Last delivery on Fri
- Last delivery on Sat
- No delivery in Nov
- Parts of the day Order
- Name of order day
- Day and part delivery
- Day and part order
- Fri Early Morning
- Fri Eve
- Fri Morning
- Fri Night
- Fri Noon
- Mon Early Morning
- Mon Eve
- Mon Morning
- Mon Night
- Mon Noon
- Sat Early Morning
- Sat Eve
- Sat Morning
- Sat Night
- Sat Noon
- Sun Early Morning
- Sun Eve
- Sun Morning
- Sun Night
- Sun Noon
- Thu Early Morning
- Thu Eve
- Thu Morning
- Thu Night
- Thu Noon
- Tue Early Morning
- Tue Eve
- Tue Morning
- Tue Night
- Tue Noon
- Wed Early Morning
- Wed Eve
- Wed Morning
- Wed Night
- Wed Noon
- Order placed on Fri
- Order placed on Mon
- Order placed on Sat
- Order placed on Sun
- Order placed on Thu
- Order placed on Tue
- Order placed on Wed
- Ordered in the Late Night
- Ordered in the Early Morning
- Ordered in the Morning
- Ordered in the Noon
- Ordered in the Eve
- Ordered in the Night
- Week number
- Weeks till delivery
- Days till delivery
- Days between last delivery
- Most common delivery day
- Most common delivery day as number
- Most common delivery day as number sin
- Most common delivery day as number cos
- Most common delivery part
- History hours booked before
- Most common delivery part in numbers
- Income groups
- Order during week
- No selection
- Days till delivery week
- No delivery
- Number of slots in delivery week
- Date

E

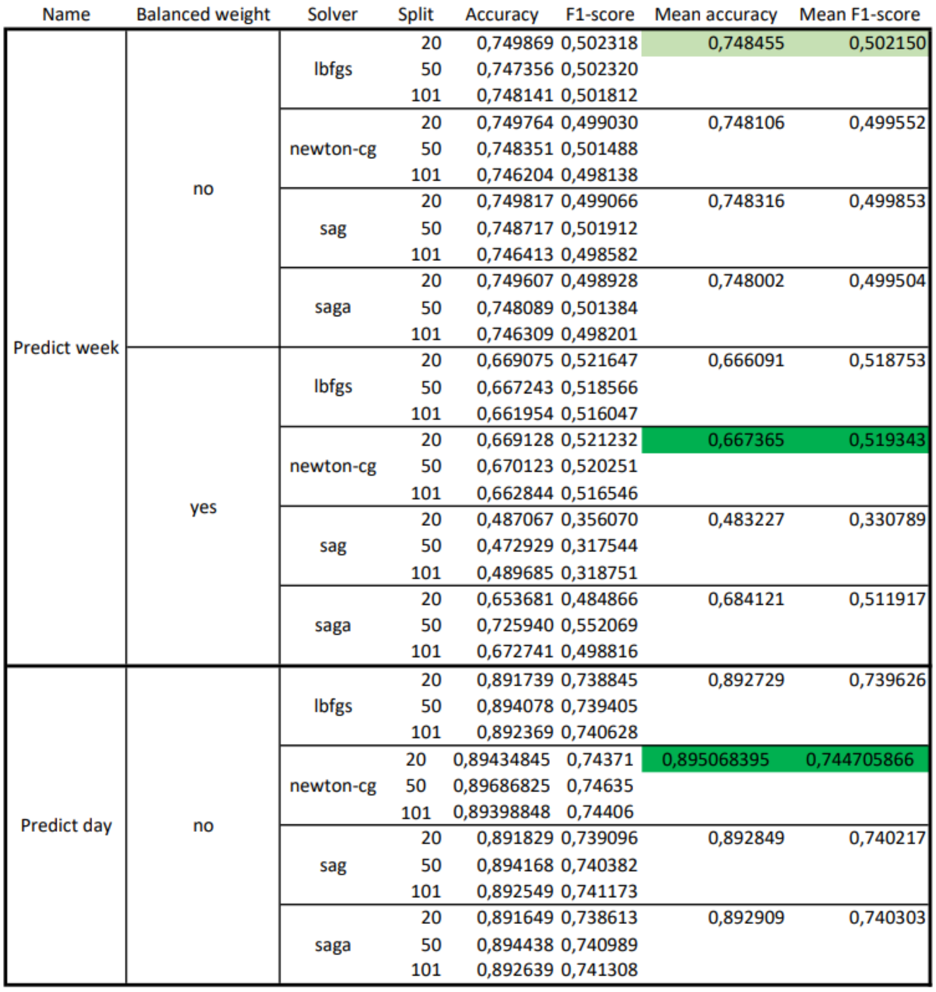
Hyperparameter tuning

In this part, the more comprehensive results of the different solvers for the MNL and NN models can be found. Both models are performed based on data that is divided into a train and test set with different splits. The MNL model is executed to predict the delivery week, day and part of the day with the lbfgs, newton-cg, sag and saga solvers. The different solvers are also performed with and without the balanced class weights. Based on the F1-score results, it can be found that the newton-cg solver performs as best for all three predictions. However, it is also found that including balanced class weights does not always positively affect performance and will not be used for predicting the delivery day.

The NN models are also used to predict the delivery week, day and part of the day in combination with the ReLU, Tanh, Sigmoid and Elu activation functions. Next to diverse functions and data split in different ways, the number of neurons in hidden layers differs, and all models are performed with and without the usage of balanced class weights. After executing all different architecture combinations, it is found that for predicting in which week the delivery will take place, the ReLu activation function with three hidden layers and ninety neurons in these layers performs slightly better than other functions. For predicting the delivery day, it is found that the ReLu function with two hidden layers containing seventy neurons also outperforms other functions. As a result, the NN will use the ReLu activation function to predict the week and day of the delivery. When looking into which activation functions perform best for predicting the part of the delivery day, it is found that not the ReLu function but the Tanh function with three hidden layers of fifty neurons has the highest F1-score. Therefore, the NN will use the Tanh activation function when predicting the part of the delivery day. Despite that, the NN will not use the same activation function in the hidden layers for the whole prediction process; all three prediction steps will be performed without class weights.

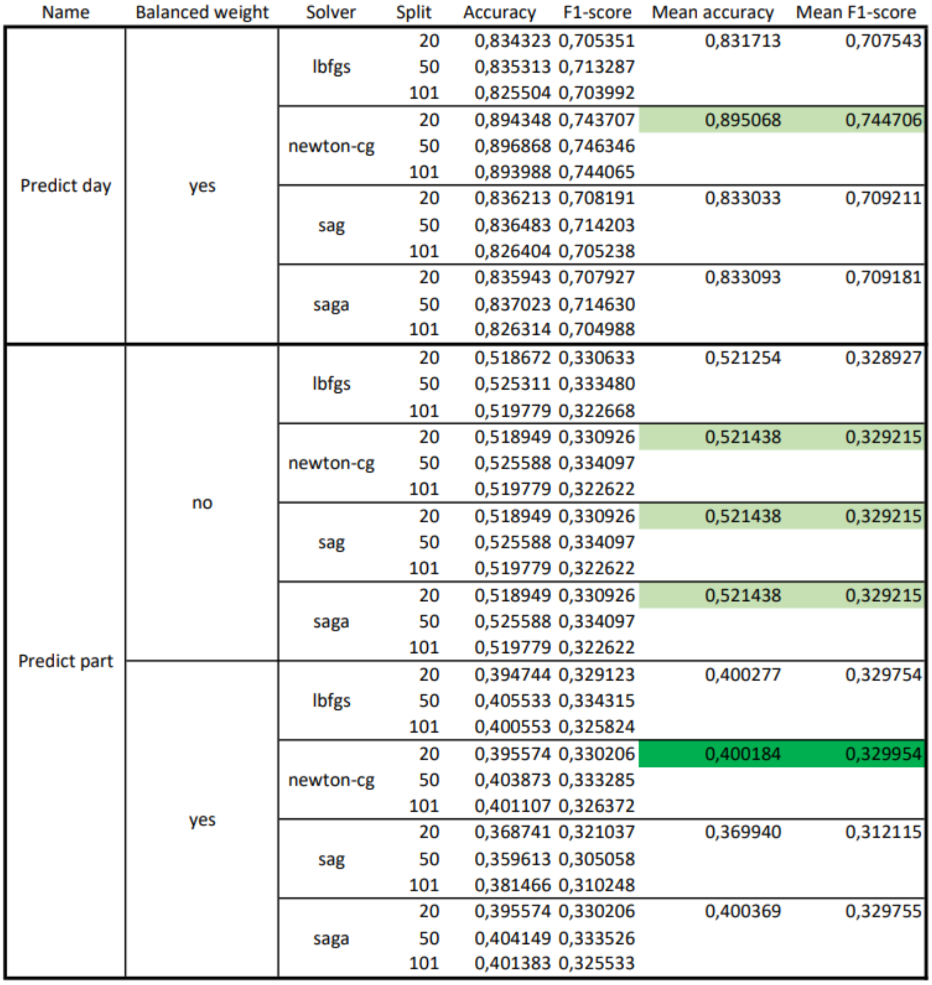
All the results of the hyperparameter tuning for both the MNL and NN can be found in section E.1 and section E.2. In these tables, the architecture with the highest F1-score is highlighted in green, as the configuration with the highest score is assumed to have the best prediction performance.

E.1. MNL



Name	Balanced weight	Solver	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict week	no	lbfgs	20	0,749869	0,502318	0,748455	0,502150
			50	0,747356	0,502320		
			101	0,748141	0,501812		
		newton-cg	20	0,749764	0,499030	0,748106	0,499552
			50	0,748351	0,501488		
			101	0,746204	0,498138		
		sag	20	0,749817	0,499066	0,748316	0,499853
			50	0,748717	0,501912		
			101	0,746413	0,498582		
		saga	20	0,749607	0,498928	0,748002	0,499504
			50	0,748089	0,501384		
			101	0,746309	0,498201		
	yes	lbfgs	20	0,669075	0,521647	0,666091	0,518753
			50	0,667243	0,518566		
			101	0,661954	0,516047		
		newton-cg	20	0,669128	0,521232	0,667365	0,519343
			50	0,670123	0,520251		
			101	0,662844	0,516546		
		sag	20	0,487067	0,356070	0,483227	0,330789
			50	0,472929	0,317544		
			101	0,489685	0,318751		
		saga	20	0,653681	0,484866	0,684121	0,511917
			50	0,725940	0,552069		
			101	0,672741	0,498816		
Predict day	no	lbfgs	20	0,891739	0,738845	0,892729	0,739626
			50	0,894078	0,739405		
			101	0,892369	0,740628		
		newton-cg	20	0,89434845	0,74371	0,895068395	0,744705866
			50	0,89686825	0,74635		
			101	0,89398848	0,74406		
		sag	20	0,891829	0,739096	0,892849	0,740217
			50	0,894168	0,740382		
			101	0,892549	0,741173		
		saga	20	0,891649	0,738613	0,892909	0,740303
			50	0,894438	0,740989		
			101	0,892639	0,741308		

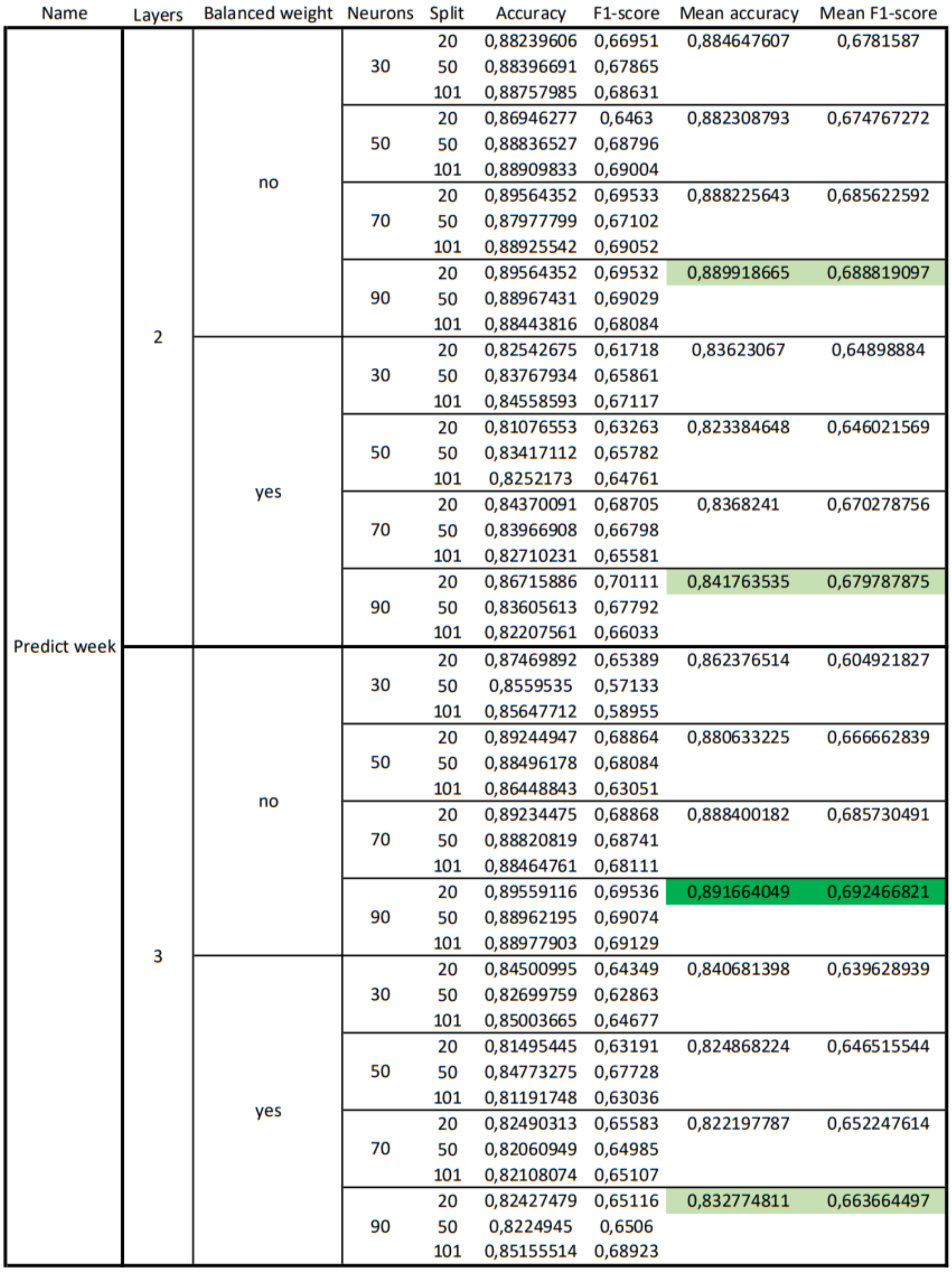
Table E.1: Overview of the results for all prediction steps performed with the MNL model with different solvers and with and without balanced class weights.



Name	Balanced weight	Solver	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict day	yes	lbfgs	20	0,834323	0,705351	0,831713	0,707543
			50	0,835313	0,713287		
			101	0,825504	0,703992		
		newton-cg	20	0,894348	0,743707	0,895068	0,744706
			50	0,896868	0,746346		
			101	0,893988	0,744065		
		sag	20	0,836213	0,708191	0,833033	0,709211
			50	0,836483	0,714203		
			101	0,826404	0,705238		
		saga	20	0,835943	0,707927	0,833093	0,709181
			50	0,837023	0,714630		
			101	0,826314	0,704988		
Predict part	no	lbfgs	20	0,518672	0,330633	0,521254	0,328927
			50	0,525311	0,333480		
			101	0,519779	0,322668		
		newton-cg	20	0,518949	0,330926	0,521438	0,329215
			50	0,525588	0,334097		
			101	0,519779	0,322622		
		sag	20	0,518949	0,330926	0,521438	0,329215
			50	0,525588	0,334097		
			101	0,519779	0,322622		
		saga	20	0,518949	0,330926	0,521438	0,329215
			50	0,525588	0,334097		
			101	0,519779	0,322622		
	yes	lbfgs	20	0,394744	0,329123	0,400277	0,329754
			50	0,405533	0,334315		
			101	0,400553	0,325824		
		newton-cg	20	0,395574	0,330206	0,400184	0,329954
			50	0,403873	0,333285		
			101	0,401107	0,326372		
		sag	20	0,368741	0,321037	0,369940	0,312115
			50	0,359613	0,305058		
			101	0,381466	0,310248		
		saga	20	0,395574	0,330206	0,400369	0,329755
			50	0,404149	0,333526		
			101	0,401383	0,325533		

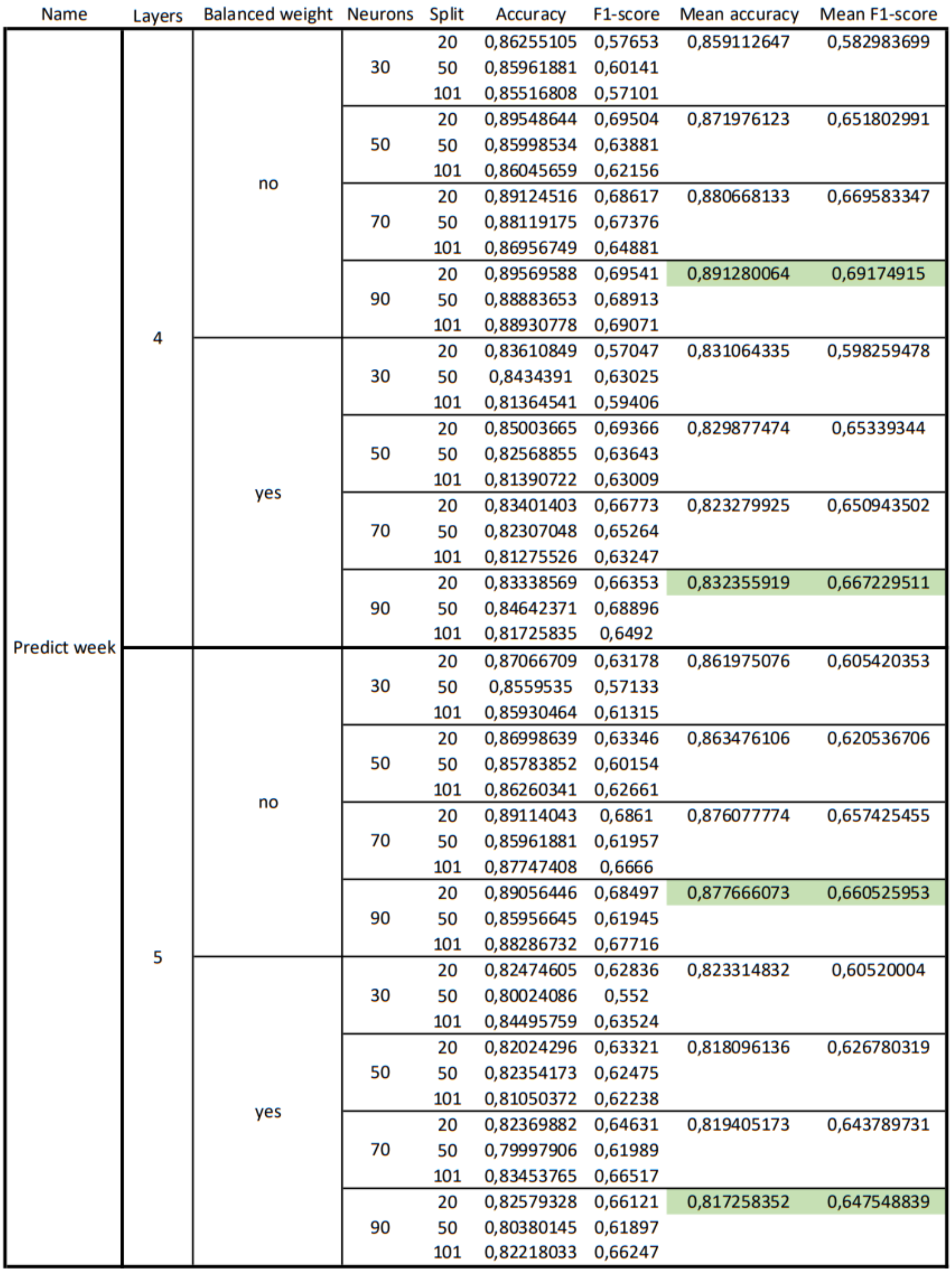
Table E.2: Overview of different solvers for MNL model.

E.2. NN



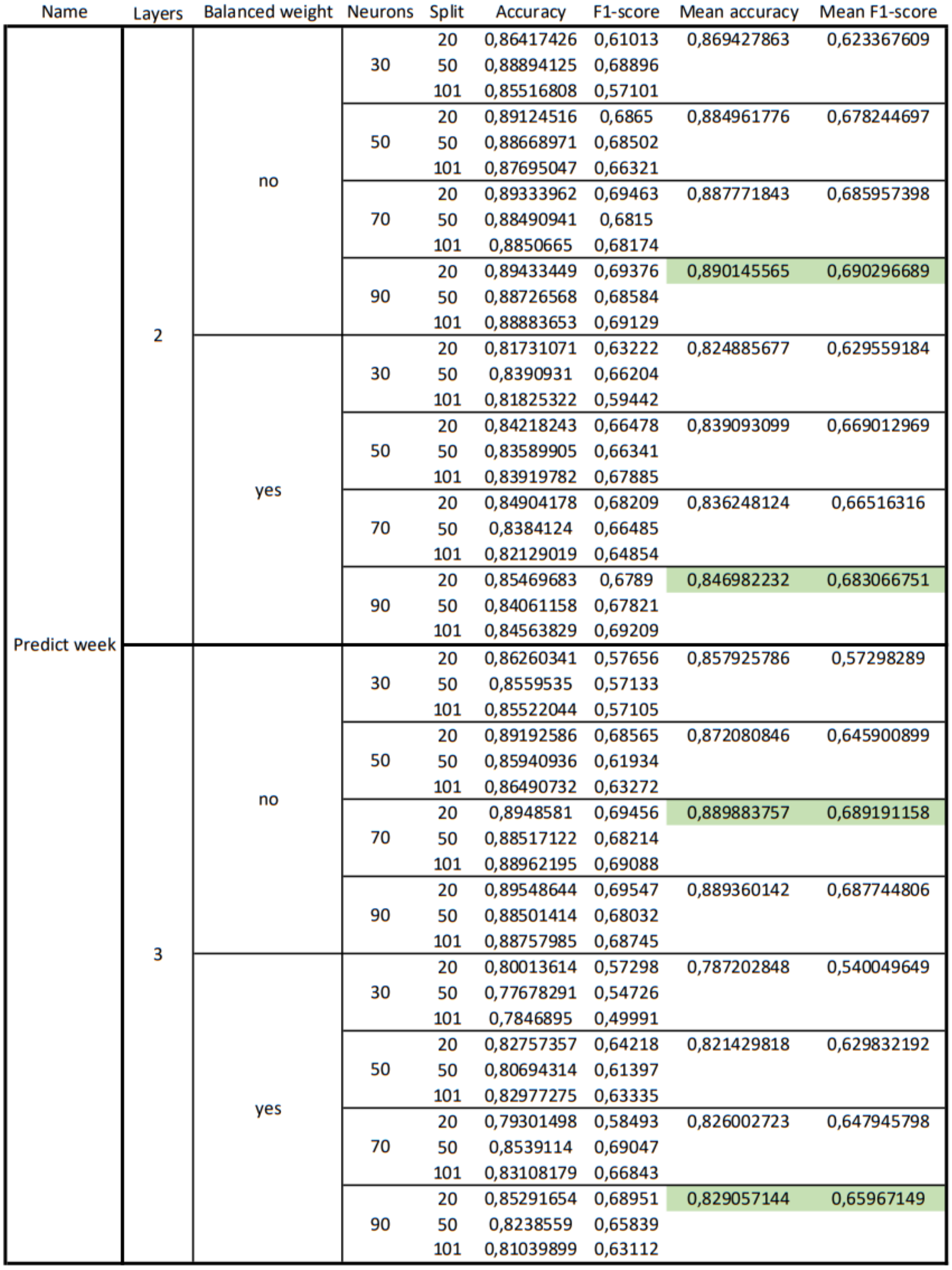
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict week	2	no	30	20	0,88239606	0,66951	0,884647607	0,6781587
				50	0,88396691	0,67865		
				101	0,88757985	0,68631		
			50	20	0,86946277	0,6463	0,882308793	0,674767272
				50	0,88836527	0,68796		
				101	0,88909833	0,69004		
			70	20	0,89564352	0,69533	0,888225643	0,685622592
				50	0,87977799	0,67102		
				101	0,88925542	0,69052		
			90	20	0,89564352	0,69532	0,889918665	0,688819097
				50	0,88967431	0,69029		
				101	0,88443816	0,68084		
		yes	30	20	0,82542675	0,61718	0,83623067	0,64898884
				50	0,83767934	0,65861		
				101	0,84558593	0,67117		
			50	20	0,81076553	0,63263	0,823384648	0,646021569
				50	0,83417112	0,65782		
				101	0,8252173	0,64761		
			70	20	0,84370091	0,68705	0,8368241	0,670278756
				50	0,83966908	0,66798		
				101	0,82710231	0,65581		
			90	20	0,86715886	0,70111	0,841763535	0,679787875
				50	0,83605613	0,67792		
				101	0,82207561	0,66033		
	3	no	30	20	0,87469892	0,65389	0,862376514	0,604921827
				50	0,8559535	0,57133		
				101	0,85647712	0,58955		
			50	20	0,89244947	0,68864	0,880633225	0,666662839
				50	0,88496178	0,68084		
				101	0,86448843	0,63051		
			70	20	0,89234475	0,68868	0,888400182	0,685730491
				50	0,88820819	0,68741		
				101	0,88464761	0,68111		
			90	20	0,89559116	0,69536	0,891664049	0,692466821
				50	0,88962195	0,69074		
				101	0,88977903	0,69129		
		yes	30	20	0,84500995	0,64349	0,840681398	0,639628939
				50	0,82699759	0,62863		
				101	0,85003665	0,64677		
			50	20	0,81495445	0,63191	0,824868224	0,646515544
				50	0,84773275	0,67728		
				101	0,81191748	0,63036		
			70	20	0,82490313	0,65583	0,822197787	0,652247614
				50	0,82060949	0,64985		
				101	0,82108074	0,65107		
			90	20	0,82427479	0,65116	0,832774811	0,663664497
				50	0,8224945	0,6506		
				101	0,85155514	0,68923		

Table E.3: Results of predicting the delivery week with the ReLu activation function part 1.



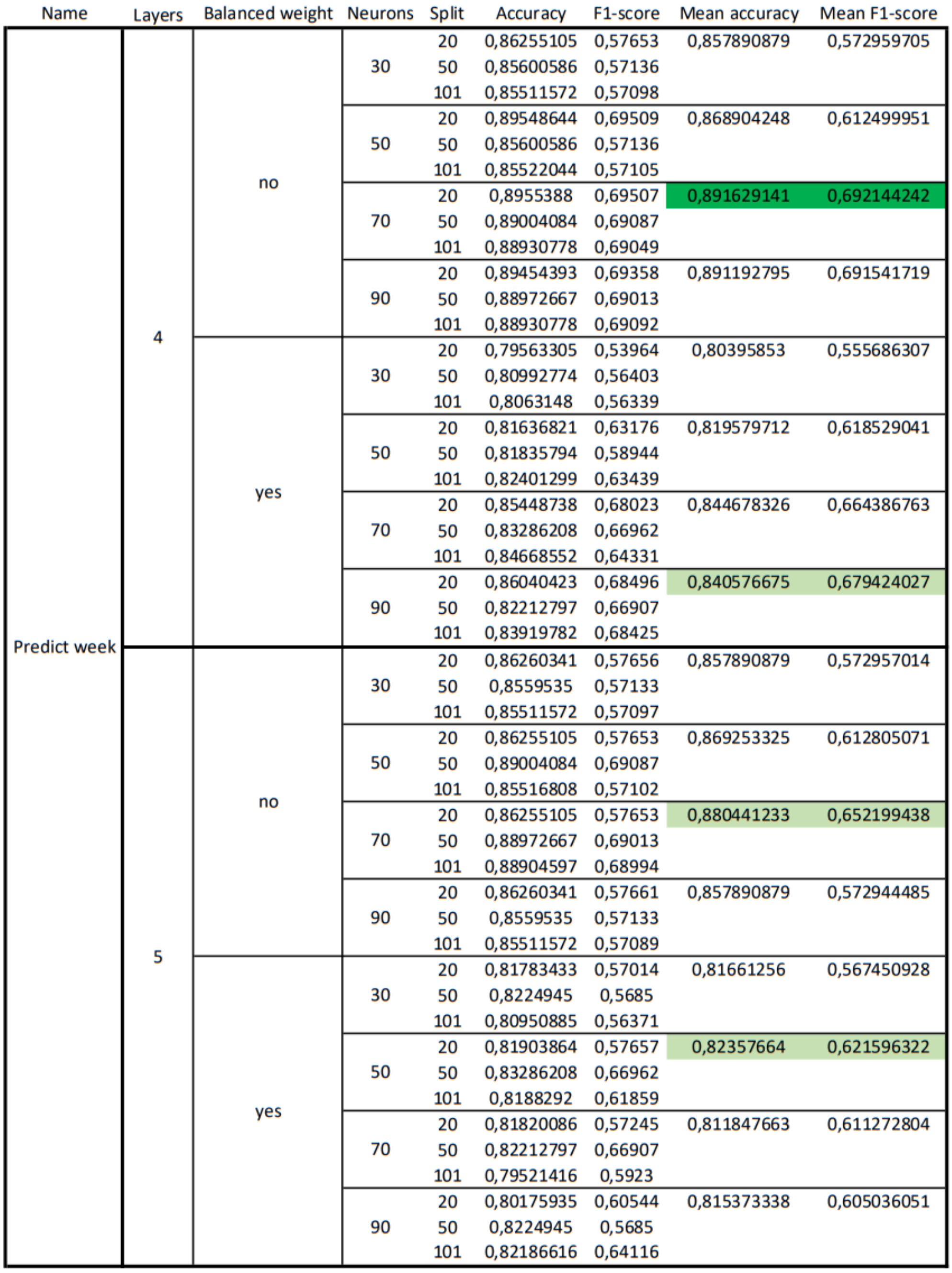
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict week	4	no	30	20	0,86255105	0,57653	0,859112647	0,582983699
				50	0,85961881	0,60141		
				101	0,85516808	0,57101		
			50	20	0,89548644	0,69504	0,871976123	0,651802991
				50	0,85998534	0,63881		
				101	0,86045659	0,62156		
			70	20	0,89124516	0,68617	0,880668133	0,669583347
				50	0,88119175	0,67376		
				101	0,86956749	0,64881		
			90	20	0,89569588	0,69541	0,891280064	0,69174915
				50	0,88883653	0,68913		
				101	0,88930778	0,69071		
		yes	30	20	0,83610849	0,57047	0,831064335	0,598259478
				50	0,8434391	0,63025		
				101	0,81364541	0,59406		
			50	20	0,85003665	0,69366	0,829877474	0,65339344
				50	0,82568855	0,63643		
				101	0,81390722	0,63009		
			70	20	0,83401403	0,66773	0,823279925	0,650943502
				50	0,82307048	0,65264		
				101	0,81275526	0,63247		
			90	20	0,83338569	0,66353	0,832355919	0,667229511
				50	0,84642371	0,68896		
				101	0,81725835	0,6492		
	5	no	30	20	0,87066709	0,63178	0,861975076	0,605420353
				50	0,8559535	0,57133		
				101	0,85930464	0,61315		
			50	20	0,86998639	0,63346	0,863476106	0,620536706
				50	0,85783852	0,60154		
				101	0,86260341	0,62661		
			70	20	0,89114043	0,6861	0,876077774	0,657425455
				50	0,85961881	0,61957		
				101	0,87747408	0,6666		
			90	20	0,89056446	0,68497	0,877666073	0,660525953
				50	0,85956645	0,61945		
				101	0,88286732	0,67716		
		yes	30	20	0,82474605	0,62836	0,823314832	0,60520004
				50	0,80024086	0,552		
				101	0,84495759	0,63524		
			50	20	0,82024296	0,63321	0,818096136	0,626780319
				50	0,82354173	0,62475		
				101	0,81050372	0,62238		
			70	20	0,82369882	0,64631	0,819405173	0,643789731
				50	0,79997906	0,61989		
				101	0,83453765	0,66517		
			90	20	0,82579328	0,66121	0,817258352	0,647548839
				50	0,80380145	0,61897		
				101	0,82218033	0,66247		

Table E.4: Results of predicting the delivery week with the ReLu activation function part 2.



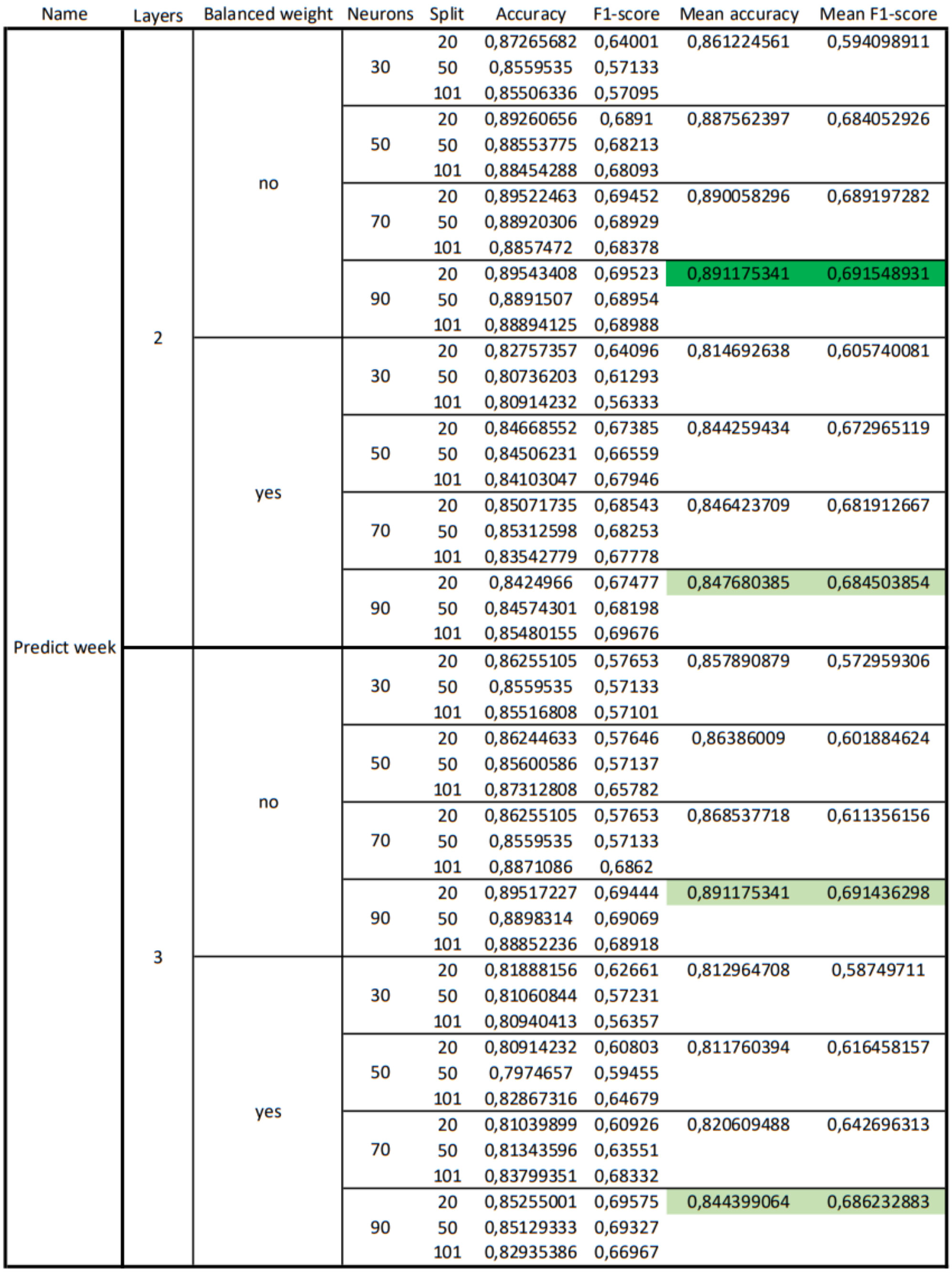
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict week	2	no	30	20	0,86417426	0,61013	0,869427863	0,623367609
				50	0,88894125	0,68896		
				101	0,85516808	0,57101		
			50	20	0,89124516	0,6865	0,884961776	0,678244697
				50	0,88668971	0,68502		
				101	0,87695047	0,66321		
			70	20	0,89333962	0,69463	0,887771843	0,685957398
				50	0,88490941	0,6815		
				101	0,8850665	0,68174		
			90	20	0,89433449	0,69376	0,890145565	0,690296689
				50	0,88726568	0,68584		
				101	0,88883653	0,69129		
		yes	30	20	0,81731071	0,63222	0,824885677	0,629559184
				50	0,8390931	0,66204		
				101	0,81825322	0,59442		
			50	20	0,84218243	0,66478	0,839093099	0,669012969
				50	0,83589905	0,66341		
				101	0,83919782	0,67885		
			70	20	0,84904178	0,68209	0,836248124	0,66516316
				50	0,8384124	0,66485		
				101	0,82129019	0,64854		
			90	20	0,85469683	0,6789	0,846982232	0,683066751
				50	0,84061158	0,67821		
				101	0,84563829	0,69209		
	3	no	30	20	0,86260341	0,57656	0,857925786	0,57298289
				50	0,8559535	0,57133		
				101	0,85522044	0,57105		
			50	20	0,89192586	0,68565	0,872080846	0,645900899
				50	0,85940936	0,61934		
				101	0,86490732	0,63272		
			70	20	0,8948581	0,69456	0,889883757	0,689191158
				50	0,88517122	0,68214		
				101	0,88962195	0,69088		
			90	20	0,89548644	0,69547	0,889360142	0,687744806
				50	0,88501414	0,68032		
				101	0,88757985	0,68745		
		yes	30	20	0,80013614	0,57298	0,787202848	0,540049649
				50	0,77678291	0,54726		
				101	0,7846895	0,49991		
			50	20	0,82757357	0,64218	0,821429818	0,629832192
				50	0,80694314	0,61397		
				101	0,82977275	0,63335		
			70	20	0,79301498	0,58493	0,826002723	0,647945798
				50	0,8539114	0,69047		
				101	0,83108179	0,66843		
			90	20	0,85291654	0,68951	0,829057144	0,65967149
				50	0,8238559	0,65839		
				101	0,81039899	0,63112		

Table E.5: Results of predicting the delivery week with the Tanh activation function part 1.



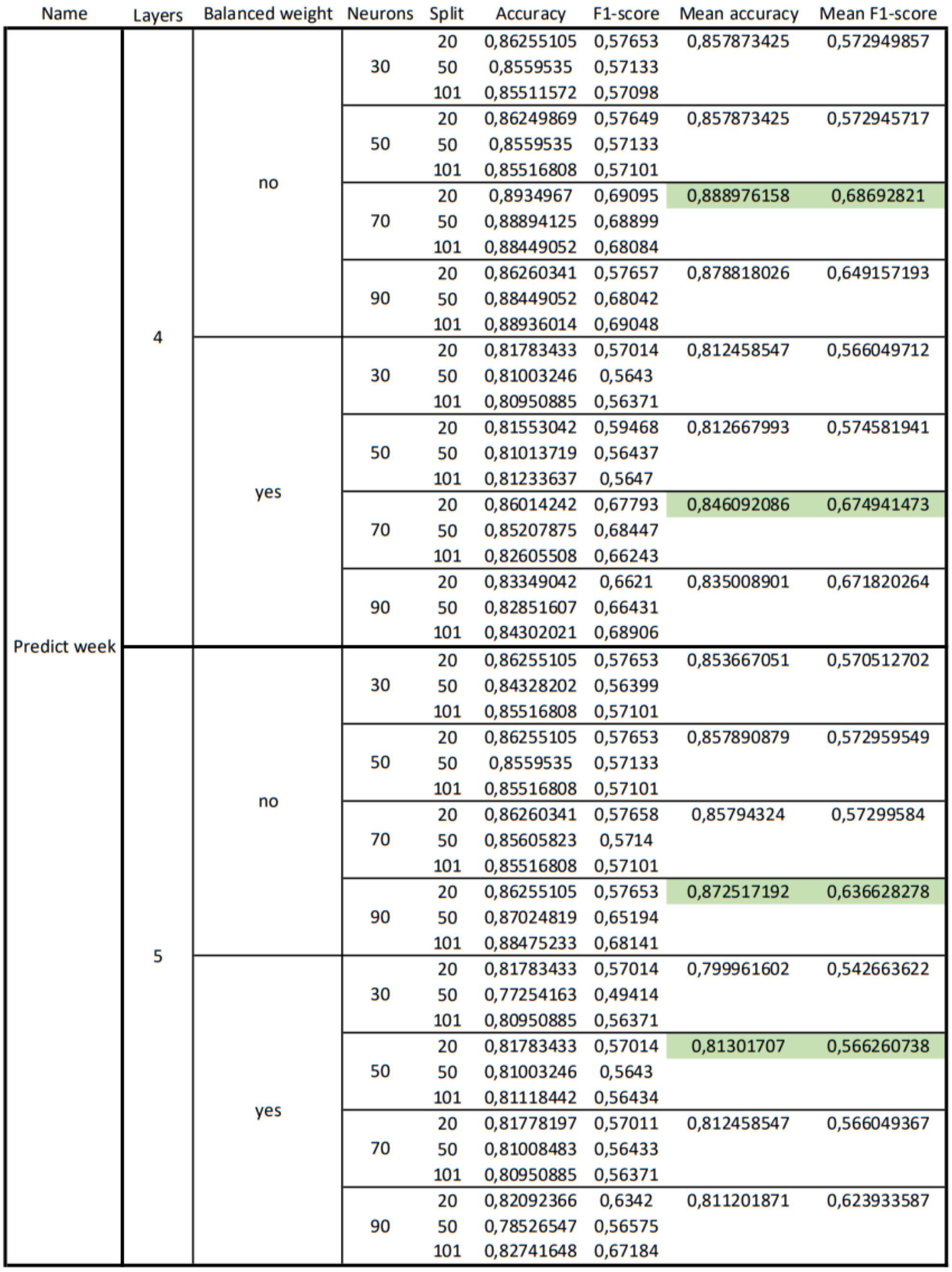
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict week	4	no	30	20	0,86255105	0,57653	0,857890879	0,572959705
				50	0,85600586	0,57136		
				101	0,85511572	0,57098		
			50	20	0,89548644	0,69509	0,868904248	0,612499951
				50	0,85600586	0,57136		
				101	0,85522044	0,57105		
			70	20	0,8955388	0,69507	0,891629141	0,692144242
				50	0,89004084	0,69087		
				101	0,88930778	0,69049		
			90	20	0,89454393	0,69358	0,891192795	0,691541719
				50	0,88972667	0,69013		
				101	0,88930778	0,69092		
		yes	30	20	0,79563305	0,53964	0,80395853	0,555686307
				50	0,80992774	0,56403		
				101	0,8063148	0,56339		
			50	20	0,81636821	0,63176	0,819579712	0,618529041
				50	0,81835794	0,58944		
				101	0,82401299	0,63439		
			70	20	0,85448738	0,68023	0,844678326	0,664386763
				50	0,83286208	0,66962		
				101	0,84668552	0,64331		
			90	20	0,86040423	0,68496	0,840576675	0,679424027
				50	0,82212797	0,66907		
				101	0,83919782	0,68425		
	5	no	30	20	0,86260341	0,57656	0,857890879	0,572957014
				50	0,8559535	0,57133		
				101	0,85511572	0,57097		
			50	20	0,86255105	0,57653	0,869253325	0,612805071
				50	0,89004084	0,69087		
				101	0,85516808	0,57102		
			70	20	0,86255105	0,57653	0,880441233	0,652199438
				50	0,88972667	0,69013		
				101	0,88904597	0,68994		
			90	20	0,86260341	0,57661	0,857890879	0,572944485
				50	0,8559535	0,57133		
				101	0,85511572	0,57089		
		yes	30	20	0,81783433	0,57014	0,81661256	0,567450928
				50	0,8224945	0,5685		
				101	0,80950885	0,56371		
			50	20	0,81903864	0,57657	0,82357664	0,621596322
				50	0,83286208	0,66962		
				101	0,8188292	0,61859		
			70	20	0,81820086	0,57245	0,811847663	0,611272804
				50	0,82212797	0,66907		
				101	0,79521416	0,5923		
			90	20	0,80175935	0,60544	0,815373338	0,605036051
				50	0,8224945	0,5685		
				101	0,82186616	0,64116		

Table E.6: Results of predicting the delivery week with the Tanh activation function part 2.



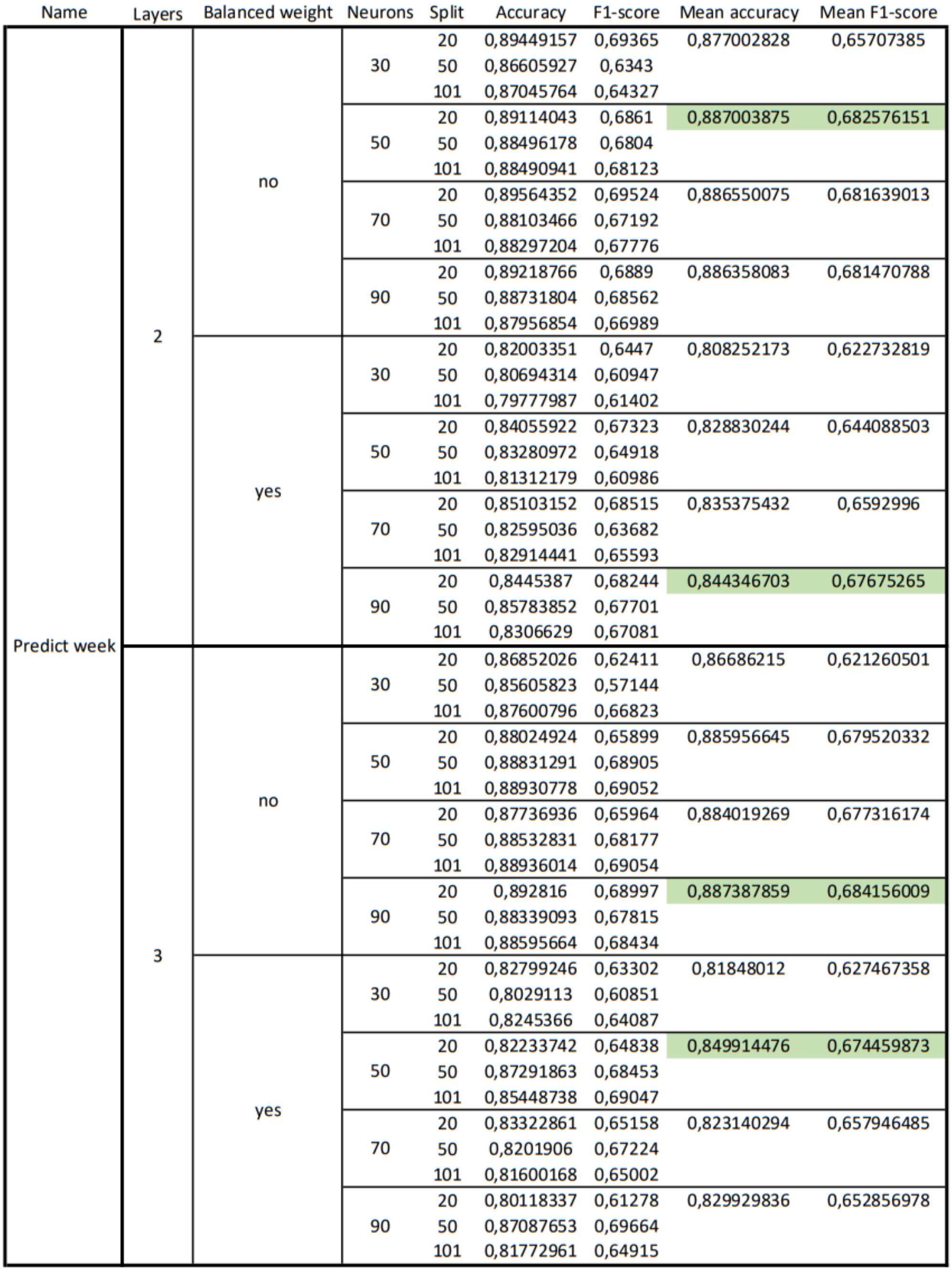
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict week	2	no	30	20	0,87265682	0,64001	0,861224561	0,594098911
				50	0,8559535	0,57133		
				101	0,85506336	0,57095		
			50	20	0,89260656	0,6891	0,887562397	0,684052926
				50	0,88553775	0,68213		
				101	0,88454288	0,68093		
			70	20	0,89522463	0,69452	0,890058296	0,689197282
				50	0,88920306	0,68929		
				101	0,8857472	0,68378		
			90	20	0,89543408	0,69523	0,891175341	0,691548931
				50	0,8891507	0,68954		
				101	0,88894125	0,68988		
		yes	30	20	0,82757357	0,64096	0,814692638	0,605740081
				50	0,80736203	0,61293		
				101	0,80914232	0,56333		
			50	20	0,84668552	0,67385	0,844259434	0,672965119
				50	0,84506231	0,66559		
				101	0,84103047	0,67946		
			70	20	0,85071735	0,68543	0,846423709	0,681912667
				50	0,85312598	0,68253		
				101	0,83542779	0,67778		
			90	20	0,8424966	0,67477	0,847680385	0,684503854
				50	0,84574301	0,68198		
				101	0,85480155	0,69676		
	3	no	30	20	0,86255105	0,57653	0,857890879	0,572959306
				50	0,8559535	0,57133		
				101	0,85516808	0,57101		
			50	20	0,86244633	0,57646	0,86386009	0,601884624
				50	0,85600586	0,57137		
				101	0,87312808	0,65782		
			70	20	0,86255105	0,57653	0,868537718	0,611356156
				50	0,8559535	0,57133		
				101	0,8871086	0,6862		
			90	20	0,89517227	0,69444	0,891175341	0,691436298
				50	0,8898314	0,69069		
				101	0,88852236	0,68918		
		yes	30	20	0,81888156	0,62661	0,812964708	0,58749711
				50	0,81060844	0,57231		
				101	0,80940413	0,56357		
			50	20	0,80914232	0,60803	0,811760394	0,616458157
				50	0,7974657	0,59455		
				101	0,82867316	0,64679		
			70	20	0,81039899	0,60926	0,820609488	0,642696313
				50	0,81343596	0,63551		
				101	0,83799351	0,68332		
			90	20	0,85255001	0,69575	0,844399064	0,686232883
				50	0,85129333	0,69327		
				101	0,82935386	0,66967		

Table E.7: Results of predicting the delivery week with the Sigmoid activation function part 1.



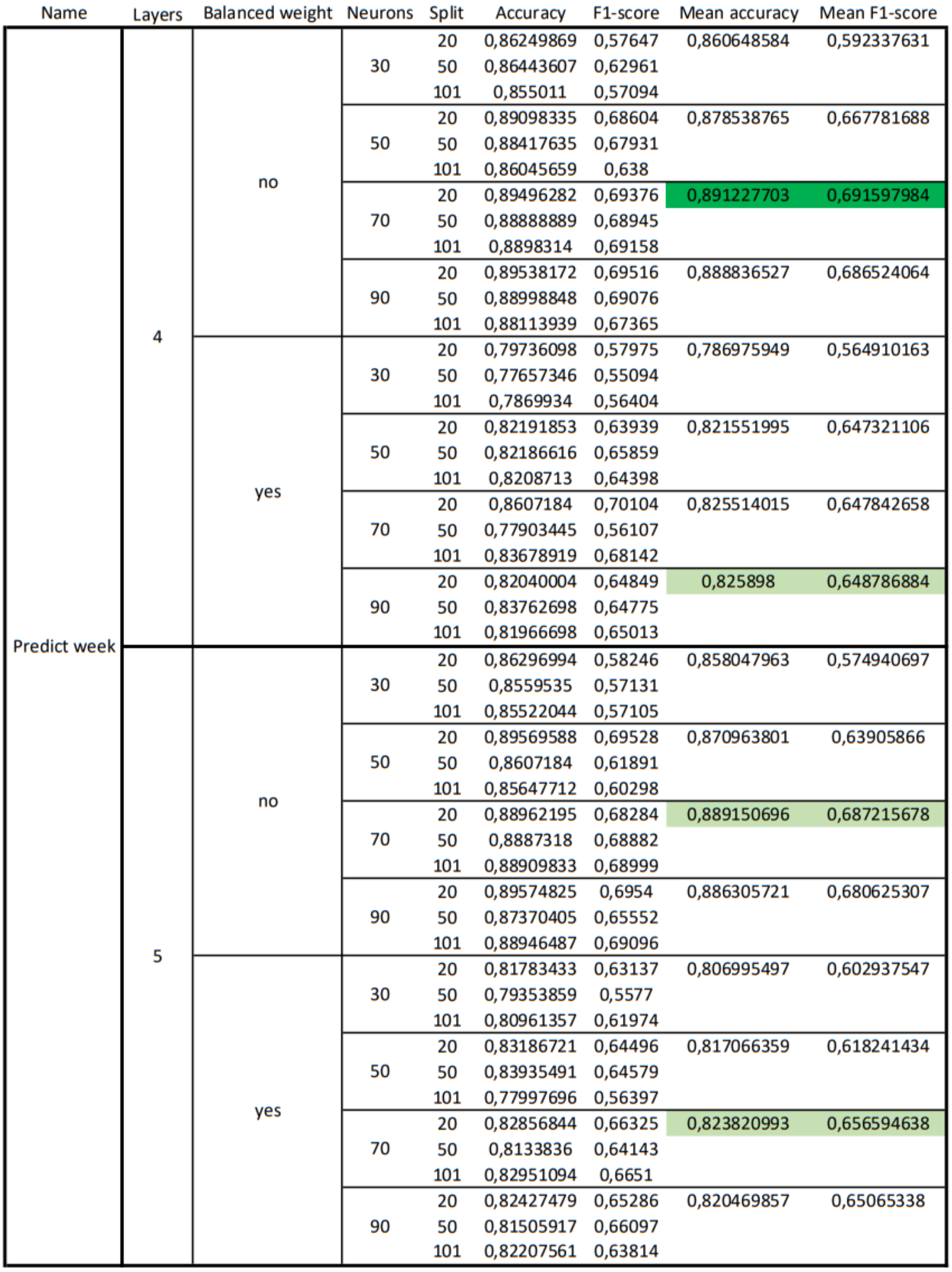
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict week	4	no	30	20	0,86255105	0,57653	0,857873425	0,572949857
				50	0,8559535	0,57133		
				101	0,85511572	0,57098		
			50	20	0,86249869	0,57649	0,857873425	0,572945717
				50	0,8559535	0,57133		
				101	0,85516808	0,57101		
			70	20	0,8934967	0,69095	0,888976158	0,68692821
				50	0,88894125	0,68899		
				101	0,88449052	0,68084		
			90	20	0,86260341	0,57657	0,878818026	0,649157193
				50	0,88449052	0,68042		
				101	0,88936014	0,69048		
		yes	30	20	0,81783433	0,57014	0,812458547	0,566049712
				50	0,81003246	0,5643		
				101	0,80950885	0,56371		
			50	20	0,81553042	0,59468	0,812667993	0,574581941
				50	0,81013719	0,56437		
				101	0,81233637	0,5647		
			70	20	0,86014242	0,67793	0,846092086	0,674941473
				50	0,85207875	0,68447		
				101	0,82605508	0,66243		
			90	20	0,83349042	0,6621	0,835008901	0,671820264
				50	0,82851607	0,66431		
				101	0,84302021	0,68906		
	5	no	30	20	0,86255105	0,57653	0,853667051	0,570512702
				50	0,84328202	0,56399		
				101	0,85516808	0,57101		
			50	20	0,86255105	0,57653	0,857890879	0,572959549
				50	0,8559535	0,57133		
				101	0,85516808	0,57101		
			70	20	0,86260341	0,57658	0,85794324	0,57299584
				50	0,85605823	0,5714		
				101	0,85516808	0,57101		
			90	20	0,86255105	0,57653	0,872517192	0,636628278
				50	0,87024819	0,65194		
				101	0,88475233	0,68141		
		yes	30	20	0,81783433	0,57014	0,799961602	0,542663622
				50	0,77254163	0,49414		
				101	0,80950885	0,56371		
			50	20	0,81783433	0,57014	0,81301707	0,566260738
				50	0,81003246	0,5643		
				101	0,81118442	0,56434		
			70	20	0,81778197	0,57011	0,812458547	0,566049367
				50	0,81008483	0,56433		
				101	0,80950885	0,56371		
			90	20	0,82092366	0,6342	0,811201871	0,623933587
				50	0,78526547	0,56575		
				101	0,82741648	0,67184		

Table E.8: Results of predicting the delivery week with the Sigmoid activation function part 2.



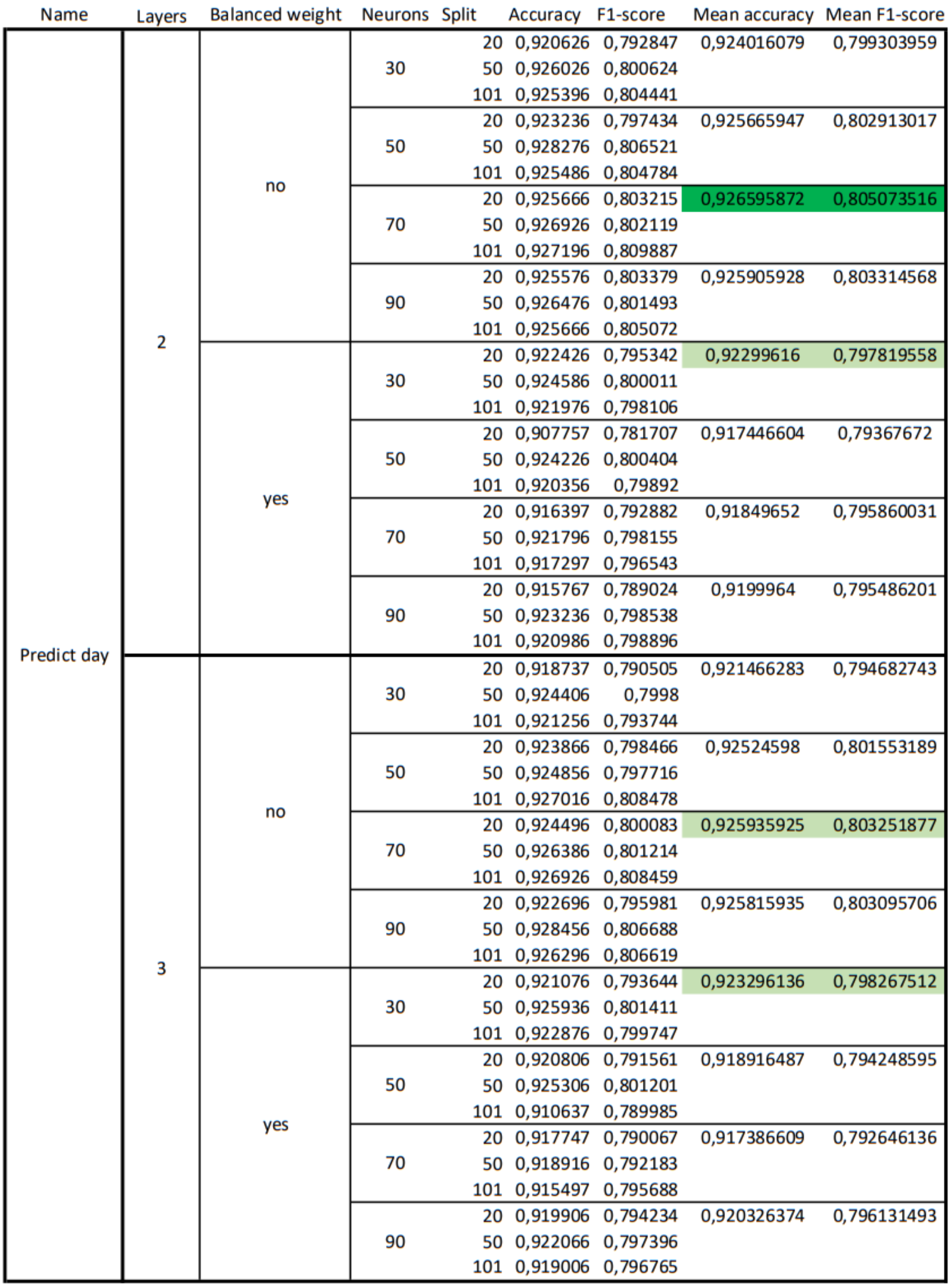
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict week	2	no	30	20	0,89449157	0,69365	0,877002828	0,65707385
				50	0,86605927	0,6343		
				101	0,87045764	0,64327		
			50	20	0,89114043	0,6861	0,887003875	0,682576151
				50	0,88496178	0,6804		
				101	0,88490941	0,68123		
			70	20	0,89564352	0,69524	0,886550075	0,681639013
				50	0,88103466	0,67192		
				101	0,88297204	0,67776		
			90	20	0,89218766	0,6889	0,886358083	0,681470788
				50	0,88731804	0,68562		
				101	0,87956854	0,66989		
		yes	30	20	0,82003351	0,6447	0,808252173	0,622732819
				50	0,80694314	0,60947		
				101	0,79777987	0,61402		
			50	20	0,84055922	0,67323	0,828830244	0,644088503
				50	0,83280972	0,64918		
				101	0,81312179	0,60986		
			70	20	0,85103152	0,68515	0,835375432	0,6592996
				50	0,82595036	0,63682		
				101	0,82914441	0,65593		
			90	20	0,8445387	0,68244	0,844346703	0,67675265
				50	0,85783852	0,67701		
				101	0,8306629	0,67081		
	3	no	30	20	0,86852026	0,62411	0,86686215	0,621260501
				50	0,85605823	0,57144		
				101	0,87600796	0,66823		
			50	20	0,88024924	0,65899	0,885956645	0,679520332
				50	0,88831291	0,68905		
				101	0,88930778	0,69052		
			70	20	0,87736936	0,65964	0,884019269	0,677316174
				50	0,88532831	0,68177		
				101	0,88936014	0,69054		
			90	20	0,892816	0,68997	0,887387859	0,684156009
				50	0,88339093	0,67815		
				101	0,88595664	0,68434		
		yes	30	20	0,82799246	0,63302	0,81848012	0,627467358
				50	0,8029113	0,60851		
				101	0,8245366	0,64087		
			50	20	0,82233742	0,64838	0,849914476	0,674459873
				50	0,87291863	0,68453		
				101	0,85448738	0,69047		
			70	20	0,83322861	0,65158	0,823140294	0,657946485
				50	0,8201906	0,67224		
				101	0,81600168	0,65002		
			90	20	0,80118337	0,61278	0,829929836	0,652856978
				50	0,87087653	0,69664		
				101	0,81772961	0,64915		

Table E.9: Results of predicting the delivery week with the ELU activation function part 1.



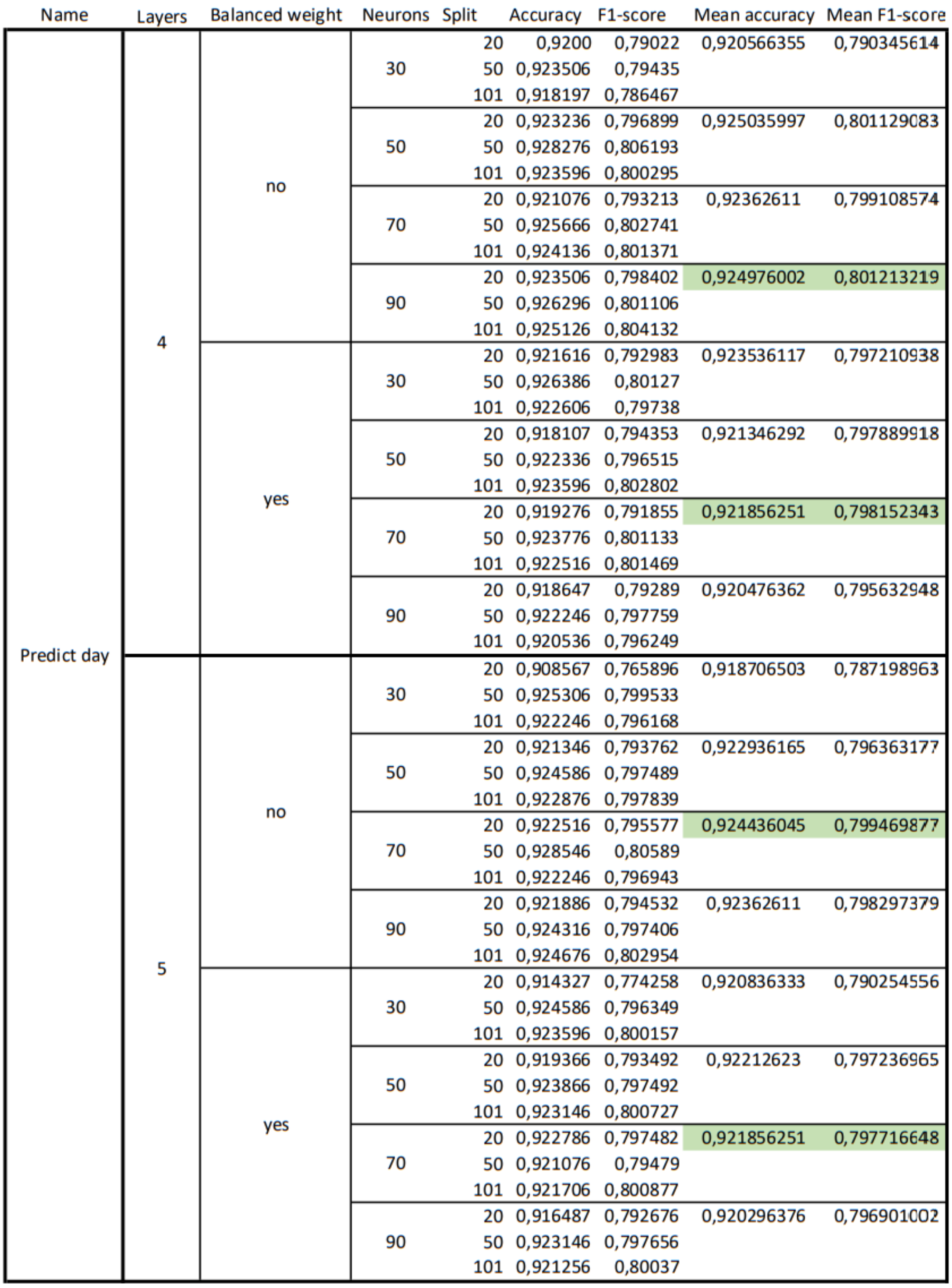
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict week	4	no	30	20	0,86249869	0,57647	0,860648584	0,592337631
				50	0,86443607	0,62961		
				101	0,855011	0,57094		
			50	20	0,89098335	0,68604	0,878538765	0,667781688
				50	0,88417635	0,67931		
				101	0,86045659	0,638		
			70	20	0,89496282	0,69376	0,891227703	0,691597984
				50	0,88888889	0,68945		
				101	0,8898314	0,69158		
			90	20	0,89538172	0,69516	0,888836527	0,686524064
				50	0,88998848	0,69076		
				101	0,88113939	0,67365		
		yes	30	20	0,79736098	0,57975	0,786975949	0,564910163
				50	0,77657346	0,55094		
				101	0,7869934	0,56404		
			50	20	0,82191853	0,63939	0,821551995	0,647321106
				50	0,82186616	0,65859		
				101	0,8208713	0,64398		
			70	20	0,8607184	0,70104	0,825514015	0,647842658
				50	0,77903445	0,56107		
				101	0,83678919	0,68142		
			90	20	0,82040004	0,64849	0,825898	0,648786884
				50	0,83762698	0,64775		
				101	0,81966698	0,65013		
	5	no	30	20	0,86296994	0,58246	0,858047963	0,574940697
				50	0,8559535	0,57131		
				101	0,85522044	0,57105		
			50	20	0,89569588	0,69528	0,870963801	0,63905866
				50	0,8607184	0,61891		
				101	0,85647712	0,60298		
			70	20	0,88962195	0,68284	0,889150696	0,687215678
				50	0,8887318	0,68882		
				101	0,88909833	0,68999		
			90	20	0,89574825	0,6954	0,886305721	0,680625307
				50	0,87370405	0,65552		
				101	0,88946487	0,69096		
		yes	30	20	0,81783433	0,63137	0,806995497	0,602937547
				50	0,79353859	0,5577		
				101	0,80961357	0,61974		
			50	20	0,83186721	0,64496	0,817066359	0,618241434
				50	0,83935491	0,64579		
				101	0,77997696	0,56397		
			70	20	0,82856844	0,66325	0,823820993	0,656594638
				50	0,8133836	0,64143		
				101	0,82951094	0,6651		
			90	20	0,82427479	0,65286	0,820469857	0,65065338
				50	0,81505917	0,66097		
				101	0,82207561	0,63814		

Table E.10: Results of predicting the delivery week with the ELU activation function part 2.



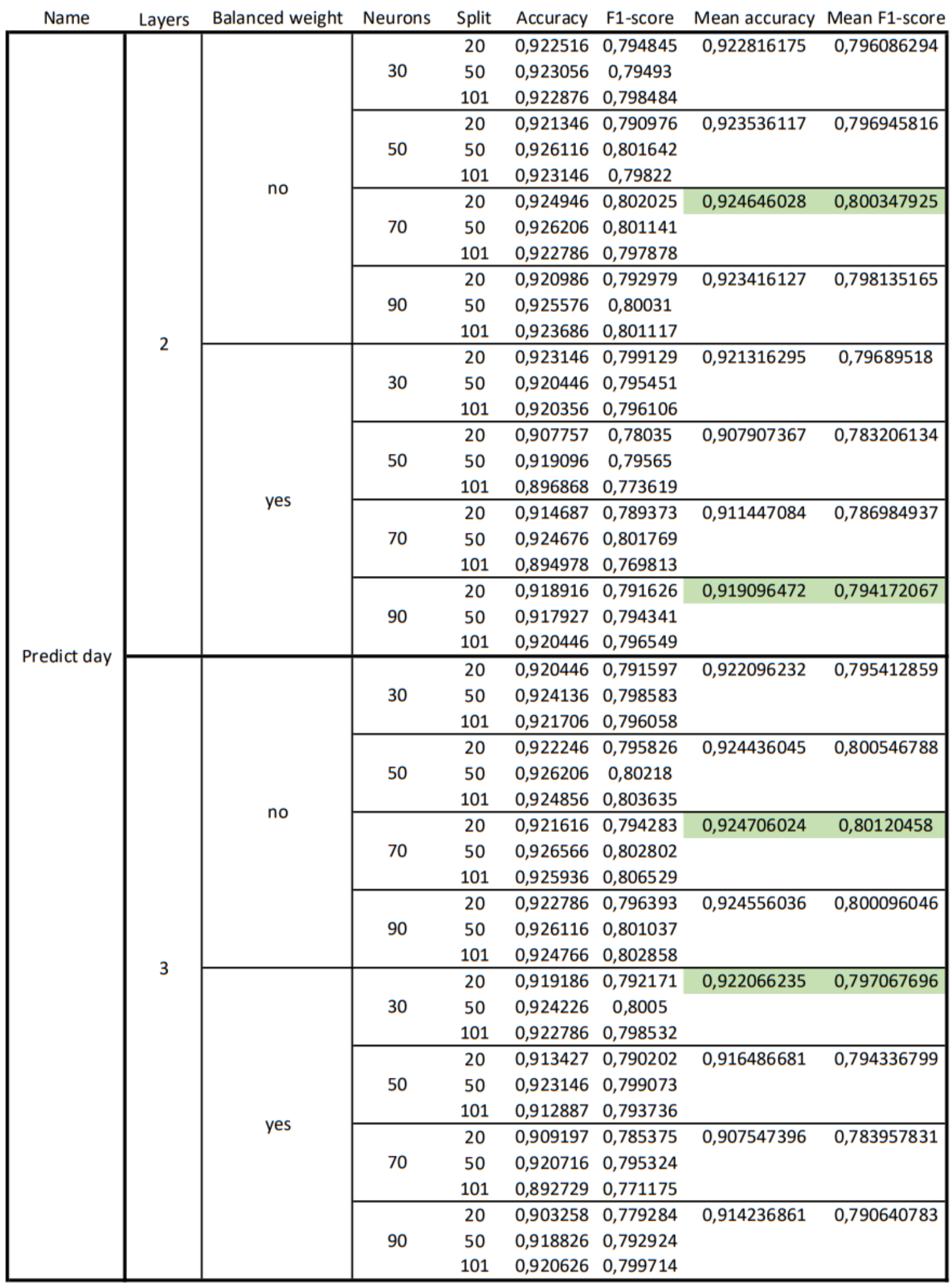
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict day	2	no	30	20	0,920626	0,792847	0,924016079	0,799303959
				50	0,926026	0,800624		
				101	0,925396	0,804441		
			50	20	0,923236	0,797434	0,925665947	0,802913017
				50	0,928276	0,806521		
				101	0,925486	0,804784		
			70	20	0,925666	0,803215	0,926595872	0,805073516
				50	0,926926	0,802119		
				101	0,927196	0,809887		
			90	20	0,925576	0,803379	0,925905928	0,803314568
				50	0,926476	0,801493		
				101	0,925666	0,805072		
		yes	30	20	0,922426	0,795342	0,92299616	0,797819558
				50	0,924586	0,800011		
				101	0,921976	0,798106		
			50	20	0,907757	0,781707	0,917446604	0,79367672
				50	0,924226	0,800404		
				101	0,920356	0,79892		
			70	20	0,916397	0,792882	0,91849652	0,795860031
				50	0,921796	0,798155		
				101	0,917297	0,796543		
			90	20	0,915767	0,789024	0,9199964	0,795486201
				50	0,923236	0,798538		
				101	0,920986	0,798896		
	3	no	30	20	0,918737	0,790505	0,921466283	0,794682743
				50	0,924406	0,7998		
				101	0,921256	0,793744		
			50	20	0,923866	0,798466	0,92524598	0,801553189
				50	0,924856	0,797716		
				101	0,927016	0,808478		
			70	20	0,924496	0,800083	0,925935925	0,803251877
				50	0,926386	0,801214		
				101	0,926926	0,808459		
			90	20	0,922696	0,795981	0,925815935	0,803095706
				50	0,928456	0,806688		
				101	0,926296	0,806619		
		yes	30	20	0,921076	0,793644	0,923296136	0,798267512
				50	0,925936	0,801411		
				101	0,922876	0,799747		
			50	20	0,920806	0,791561	0,918916487	0,794248595
				50	0,925306	0,801201		
				101	0,910637	0,789985		
			70	20	0,917747	0,790067	0,917386609	0,792646136
				50	0,918916	0,792183		
				101	0,915497	0,795688		
			90	20	0,919906	0,794234	0,920326374	0,796131493
				50	0,922066	0,797396		
				101	0,919006	0,796765		

Table E.11: Results of predicting the delivery day with the ReLu activation function part 1.



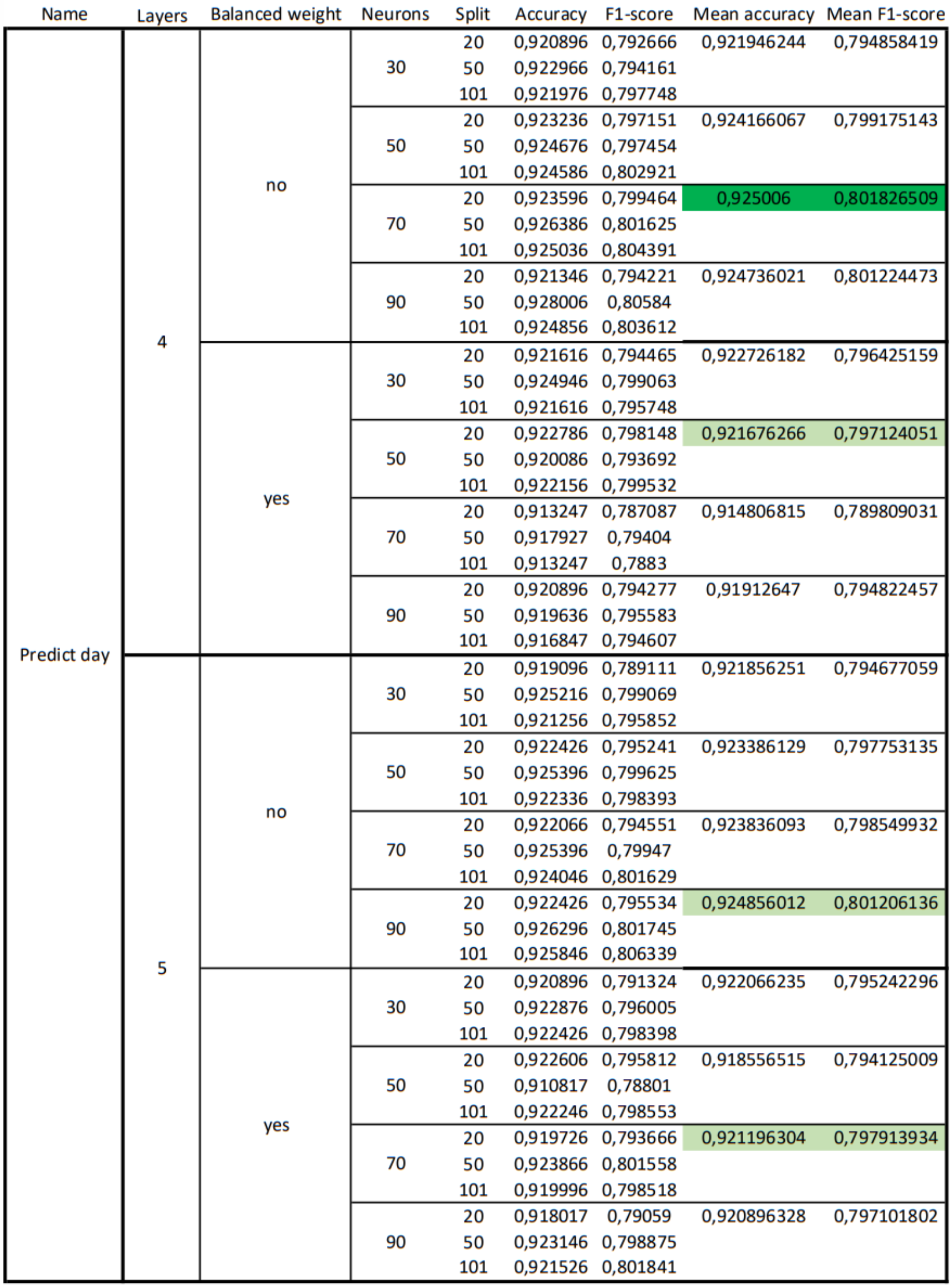
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict day	4	no	30	20	0,9200	0,79022	0,920566355	0,790345614
				50	0,923506	0,79435		
				101	0,918197	0,786467		
			50	20	0,923236	0,796899	0,925035997	0,801129083
				50	0,928276	0,806193		
				101	0,923596	0,800295		
			70	20	0,921076	0,793213	0,92362611	0,799108574
				50	0,925666	0,802741		
				101	0,924136	0,801371		
			90	20	0,923506	0,798402	0,924976002	0,801213219
				50	0,926296	0,801106		
				101	0,925126	0,804132		
		yes	30	20	0,921616	0,792983	0,923536117	0,797210938
				50	0,926386	0,80127		
				101	0,922606	0,79738		
			50	20	0,918107	0,794353	0,921346292	0,797889918
				50	0,922336	0,796515		
				101	0,923596	0,802802		
			70	20	0,919276	0,791855	0,921856251	0,798152343
				50	0,923776	0,801133		
				101	0,922516	0,801469		
			90	20	0,918647	0,79289	0,920476362	0,795632948
				50	0,922246	0,797759		
				101	0,920536	0,796249		
	5	no	30	20	0,908567	0,765896	0,918706503	0,787198963
				50	0,925306	0,799533		
				101	0,922246	0,796168		
			50	20	0,921346	0,793762	0,922936165	0,796363177
				50	0,924586	0,797489		
				101	0,922876	0,797839		
			70	20	0,922516	0,795577	0,924436045	0,799469877
				50	0,928546	0,80589		
				101	0,922246	0,796943		
			90	20	0,921886	0,794532	0,92362611	0,798297379
				50	0,924316	0,797406		
				101	0,924676	0,802954		
		yes	30	20	0,914327	0,774258	0,920836333	0,790254556
				50	0,924586	0,796349		
				101	0,923596	0,800157		
			50	20	0,919366	0,793492	0,92212623	0,797236965
				50	0,923866	0,797492		
				101	0,923146	0,800727		
			70	20	0,922786	0,797482	0,921856251	0,797716648
				50	0,921076	0,79479		
				101	0,921706	0,800877		
			90	20	0,916487	0,792676	0,920296376	0,796901002
				50	0,923146	0,797656		
				101	0,921256	0,80037		

Table E.12: Results of predicting the delivery day with the ReLu activation function part 2.



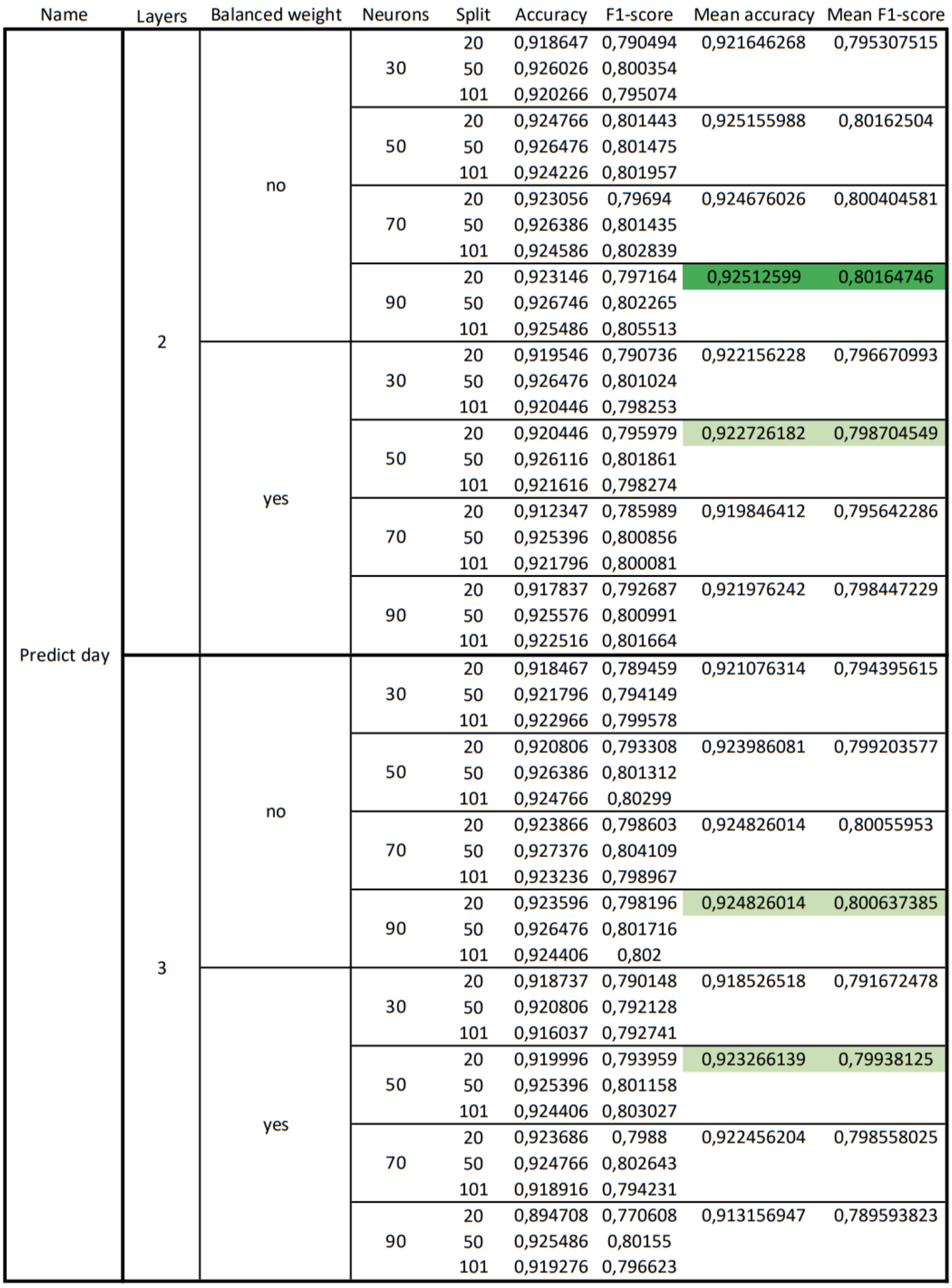
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict day	2	no	30	20	0,922516	0,794845	0,922816175	0,796086294
				50	0,923056	0,79493		
				101	0,922876	0,798484		
			50	20	0,921346	0,790976	0,923536117	0,796945816
				50	0,926116	0,801642		
				101	0,923146	0,79822		
			70	20	0,924946	0,802025	0,924646028	0,800347925
				50	0,926206	0,801141		
				101	0,922786	0,797878		
			90	20	0,920986	0,792979	0,923416127	0,798135165
				50	0,925576	0,80031		
				101	0,923686	0,801117		
		yes	30	20	0,923146	0,799129	0,921316295	0,79689518
				50	0,920446	0,795451		
				101	0,920356	0,796106		
			50	20	0,907757	0,78035	0,907907367	0,783206134
				50	0,919096	0,79565		
				101	0,896868	0,773619		
			70	20	0,914687	0,789373	0,911447084	0,786984937
				50	0,924676	0,801769		
				101	0,894978	0,769813		
			90	20	0,918916	0,791626	0,919096472	0,794172067
				50	0,917927	0,794341		
				101	0,920446	0,796549		
	3	no	30	20	0,920446	0,791597	0,922096232	0,795412859
				50	0,924136	0,798583		
				101	0,921706	0,796058		
			50	20	0,922246	0,795826	0,924436045	0,800546788
				50	0,926206	0,80218		
				101	0,924856	0,803635		
			70	20	0,921616	0,794283	0,924706024	0,80120458
				50	0,926566	0,802802		
				101	0,925936	0,806529		
			90	20	0,922786	0,796393	0,924556036	0,800096046
				50	0,926116	0,801037		
				101	0,924766	0,802858		
		yes	30	20	0,919186	0,792171	0,922066235	0,797067696
				50	0,924226	0,8005		
				101	0,922786	0,798532		
			50	20	0,913427	0,790202	0,916486681	0,794336799
				50	0,923146	0,799073		
				101	0,912887	0,793736		
			70	20	0,909197	0,785375	0,907547396	0,783957831
				50	0,920716	0,795324		
				101	0,892729	0,771175		
			90	20	0,903258	0,779284	0,914236861	0,790640783
				50	0,918826	0,792924		
				101	0,920626	0,799714		

Table E.13: Results of predicting the delivery day with the Tanh activation function part 1.



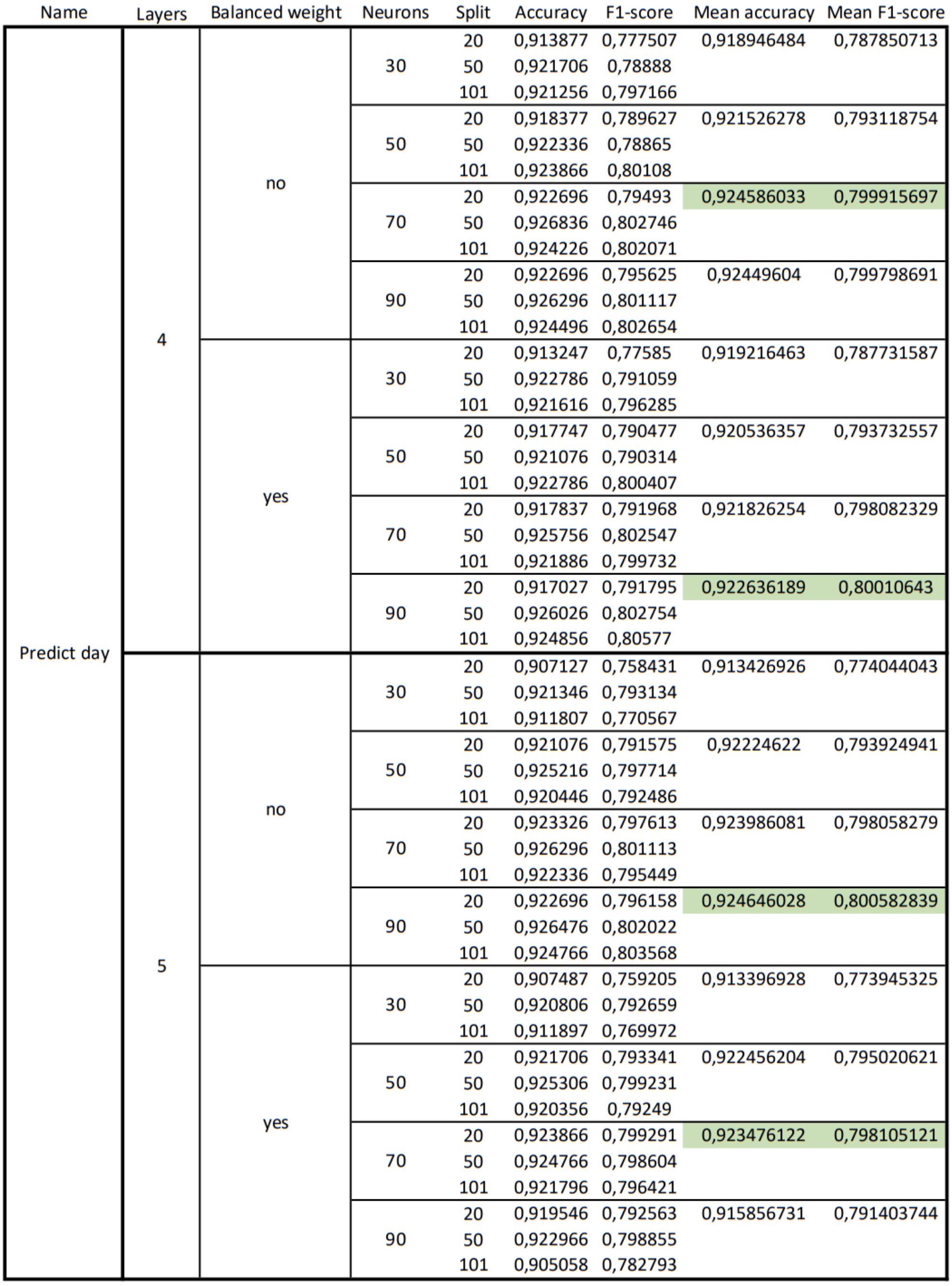
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict day	4	no	30	20	0,920896	0,792666	0,921946244	0,794858419
				50	0,922966	0,794161		
				101	0,921976	0,797748		
			50	20	0,923236	0,797151	0,924166067	0,799175143
				50	0,924676	0,797454		
				101	0,924586	0,802921		
			70	20	0,923596	0,799464	0,925006	0,801826509
				50	0,926386	0,801625		
				101	0,925036	0,804391		
			90	20	0,921346	0,794221	0,924736021	0,801224473
				50	0,928006	0,80584		
				101	0,924856	0,803612		
		yes	30	20	0,921616	0,794465	0,922726182	0,796425159
				50	0,924946	0,799063		
				101	0,921616	0,795748		
			50	20	0,922786	0,798148	0,921676266	0,797124051
				50	0,920086	0,793692		
				101	0,922156	0,799532		
			70	20	0,913247	0,787087	0,914806815	0,789809031
				50	0,917927	0,79404		
				101	0,913247	0,7883		
			90	20	0,920896	0,794277	0,91912647	0,794822457
				50	0,919636	0,795583		
				101	0,916847	0,794607		
	5	no	30	20	0,919096	0,789111	0,921856251	0,794677059
				50	0,925216	0,799069		
				101	0,921256	0,795852		
			50	20	0,922426	0,795241	0,923386129	0,797753135
				50	0,925396	0,799625		
				101	0,922336	0,798393		
			70	20	0,922066	0,794551	0,923836093	0,798549932
				50	0,925396	0,79947		
				101	0,924046	0,801629		
			90	20	0,922426	0,795534	0,924856012	0,801206136
				50	0,926296	0,801745		
				101	0,925846	0,806339		
		yes	30	20	0,920896	0,791324	0,922066235	0,795242296
				50	0,922876	0,796005		
				101	0,922426	0,798398		
			50	20	0,922606	0,795812	0,918556515	0,794125009
				50	0,910817	0,78801		
				101	0,922246	0,798553		
			70	20	0,919726	0,793666	0,921196304	0,797913934
				50	0,923866	0,801558		
				101	0,919996	0,798518		
			90	20	0,918017	0,79059	0,920896328	0,797101802
				50	0,923146	0,798875		
				101	0,921526	0,801841		

Table E.14: Results of predicting the delivery day with the Tanh activation function part 2.



Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict day	2	no	30	20	0,918647	0,790494	0,921646268	0,795307515
				50	0,926026	0,800354		
				101	0,920266	0,795074		
			50	20	0,924766	0,801443	0,925155988	0,80162504
				50	0,926476	0,801475		
				101	0,924226	0,801957		
			70	20	0,923056	0,79694	0,924676026	0,800404581
				50	0,926386	0,801435		
				101	0,924586	0,802839		
			90	20	0,923146	0,797164	0,92512599	0,80164746
				50	0,926746	0,802265		
				101	0,925486	0,805513		
		yes	30	20	0,919546	0,790736	0,922156228	0,796670993
				50	0,926476	0,801024		
				101	0,920446	0,798253		
			50	20	0,920446	0,795979	0,922726182	0,798704549
				50	0,926116	0,801861		
				101	0,921616	0,798274		
			70	20	0,912347	0,785989	0,919846412	0,795642286
				50	0,925396	0,800856		
				101	0,921796	0,800081		
			90	20	0,917837	0,792687	0,921976242	0,798447229
				50	0,925576	0,800991		
				101	0,922516	0,801664		
	3	no	30	20	0,918467	0,789459	0,921076314	0,794395615
				50	0,921796	0,794149		
				101	0,922966	0,799578		
			50	20	0,920806	0,793308	0,923986081	0,799203577
				50	0,926386	0,801312		
				101	0,924766	0,80299		
			70	20	0,923866	0,798603	0,924826014	0,80055953
				50	0,927376	0,804109		
				101	0,923236	0,798967		
			90	20	0,923596	0,798196	0,924826014	0,800637385
				50	0,926476	0,801716		
				101	0,924406	0,802		
		yes	30	20	0,918737	0,790148	0,918526518	0,791672478
				50	0,920806	0,792128		
				101	0,916037	0,792741		
			50	20	0,919996	0,793959	0,923266139	0,79938125
				50	0,925396	0,801158		
				101	0,924406	0,803027		
			70	20	0,923686	0,7988	0,922456204	0,798558025
				50	0,924766	0,802643		
				101	0,918916	0,794231		
			90	20	0,894708	0,770608	0,913156947	0,789593823
				50	0,925486	0,80155		
				101	0,919276	0,796623		

Table E.15: Results of predicting the delivery day with the Sigmoid activation function part 1.

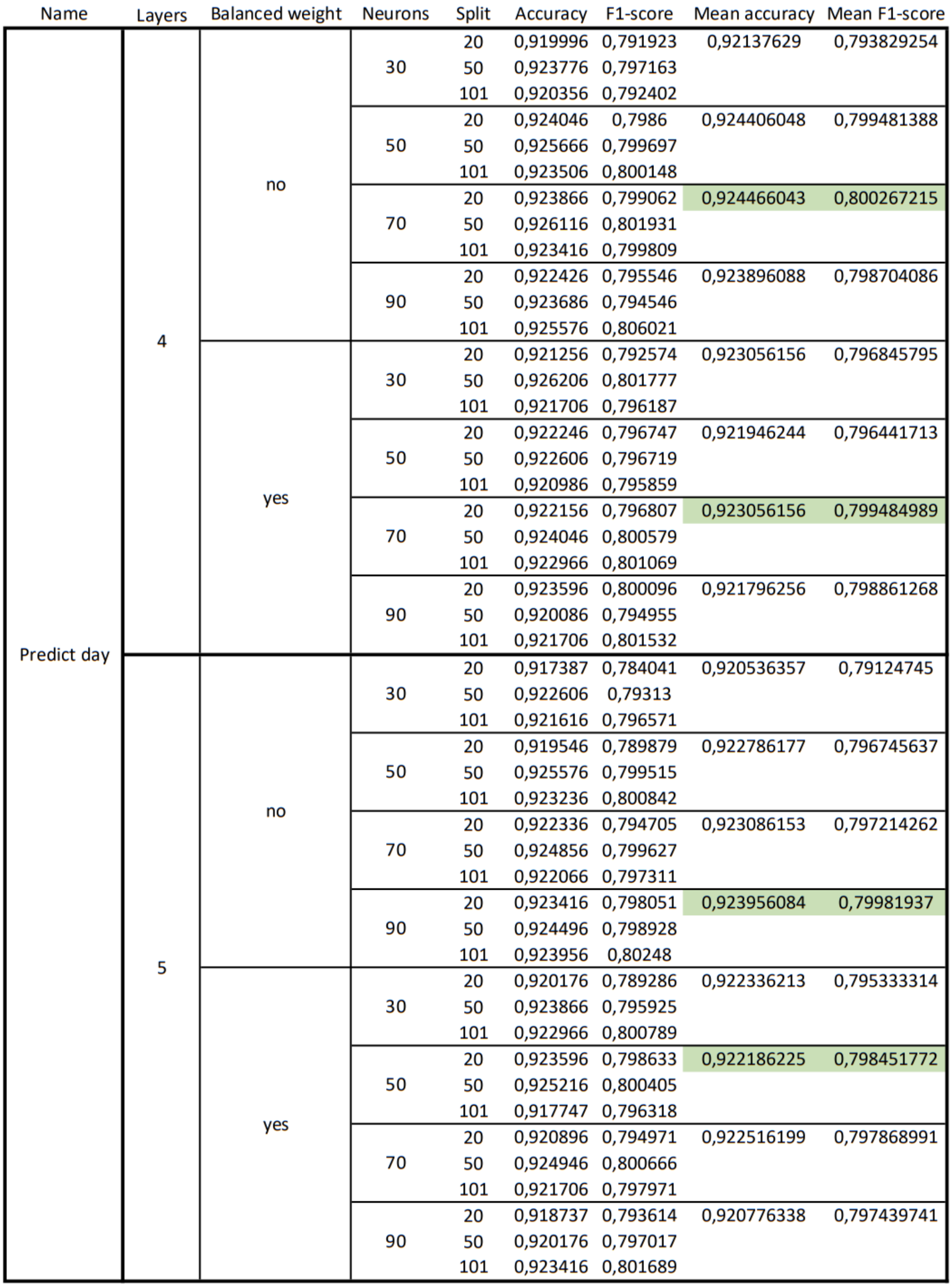


Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict day	4	no	30	20	0,913877	0,777507	0,918946484	0,787850713
				50	0,921706	0,78888		
				101	0,921256	0,797166		
			50	20	0,918377	0,789627	0,921526278	0,793118754
				50	0,922336	0,78865		
				101	0,923866	0,80108		
			70	20	0,922696	0,79493	0,924586033	0,799915697
				50	0,926836	0,802746		
				101	0,924226	0,802071		
			90	20	0,922696	0,795625	0,92449604	0,799798691
				50	0,926296	0,801117		
				101	0,924496	0,802654		
		yes	30	20	0,913247	0,77585	0,919216463	0,787731587
				50	0,922786	0,791059		
				101	0,921616	0,796285		
			50	20	0,917747	0,790477	0,920536357	0,793732557
				50	0,921076	0,790314		
				101	0,922786	0,800407		
			70	20	0,917837	0,791968	0,921826254	0,798082329
				50	0,925756	0,802547		
				101	0,921886	0,799732		
			90	20	0,917027	0,791795	0,922636189	0,80010643
				50	0,926026	0,802754		
				101	0,924856	0,80577		
	5	no	30	20	0,907127	0,758431	0,913426926	0,774044043
				50	0,921346	0,793134		
				101	0,911807	0,770567		
			50	20	0,921076	0,791575	0,92224622	0,793924941
				50	0,925216	0,797714		
				101	0,920446	0,792486		
			70	20	0,923326	0,797613	0,923986081	0,798058279
				50	0,926296	0,801113		
				101	0,922336	0,795449		
			90	20	0,922696	0,796158	0,924646028	0,800582839
				50	0,926476	0,802022		
				101	0,924766	0,803568		
		yes	30	20	0,907487	0,759205	0,913396928	0,773945325
				50	0,920806	0,792659		
				101	0,911897	0,769972		
			50	20	0,921706	0,793341	0,922456204	0,795020621
				50	0,925306	0,799231		
				101	0,920356	0,79249		
			70	20	0,923866	0,799291	0,923476122	0,798105121
				50	0,924766	0,798604		
				101	0,921796	0,796421		
			90	20	0,919546	0,792563	0,915856731	0,791403744
				50	0,922966	0,798855		
				101	0,905058	0,782793		

Table E.16: Results of predicting the delivery day with the Sigmoid activation function part 2.

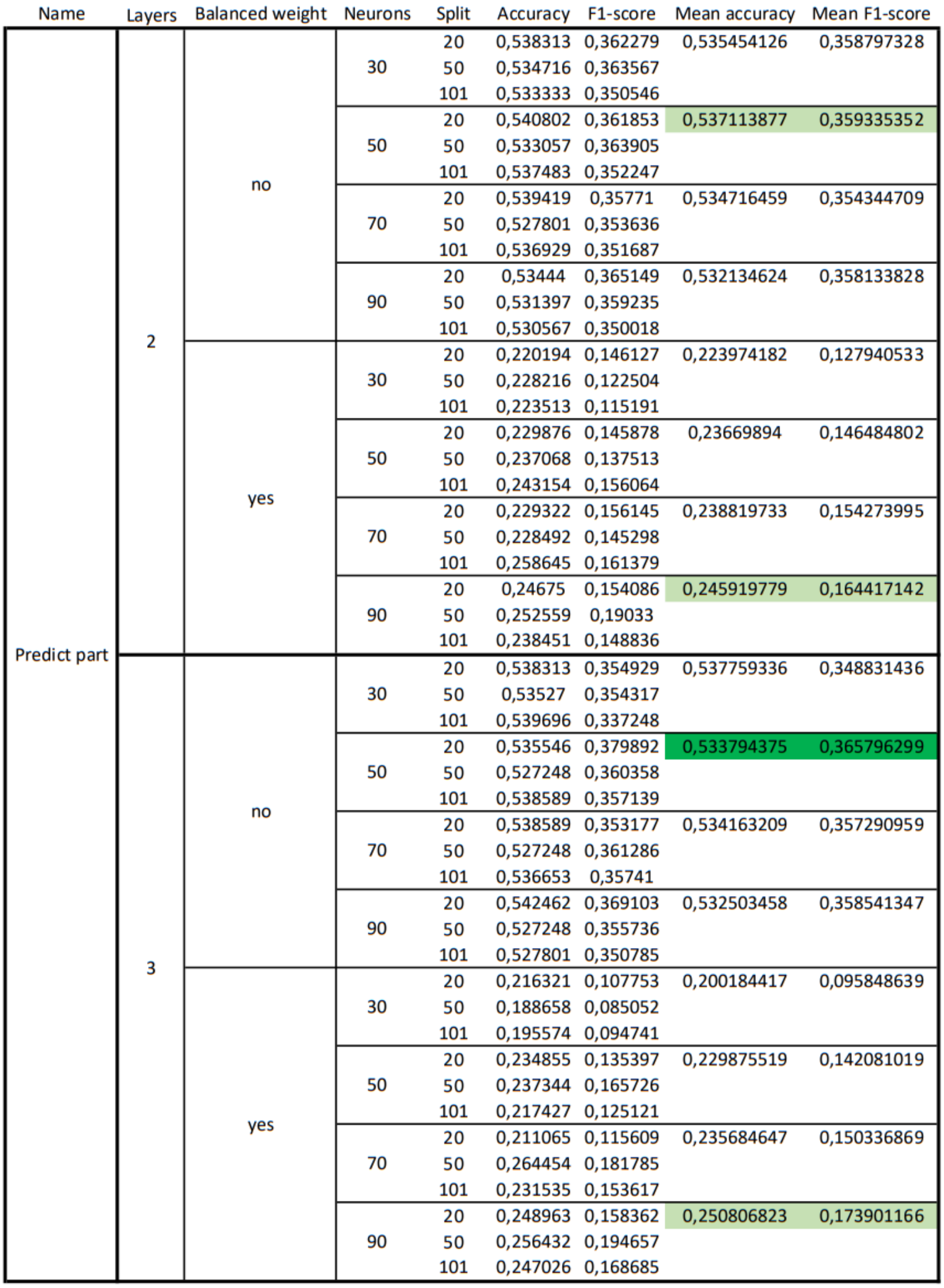
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict day	2	no	30	20	0,920806	0,793434	0,922276218	0,796081139
				50	0,925846	0,800538		
				101	0,920176	0,794272		
			50	20	0,922966	0,796633	0,92425606	0,799206152
				50	0,926026	0,800671		
				101	0,923776	0,800314		
			70	20	0,923236	0,79763	0,924526038	0,800479183
				50	0,925216	0,798955		
				101	0,925126	0,804853		
			90	20	0,922786	0,795455	0,924646028	0,800247866
				50	0,927466	0,805381		
				101	0,923686	0,799908		
		yes	30	20	0,920356	0,792781	0,920686345	0,795856252
				50	0,926206	0,802065		
				101	0,915497	0,792723		
			50	20	0,916487	0,789674	0,921046316	0,796917725
				50	0,925756	0,801322		
				101	0,920896	0,799757		
			70	20	0,911087	0,783145	0,897978162	0,773913613
				50	0,912887	0,7885		
				101	0,86996	0,750096		
			90	20	0,910187	0,785316	0,900407967	0,777993796
				50	0,86888	0,747838		
				101	0,922156	0,800827		
	3	no	30	20	0,921346	0,793531	0,922306216	0,795108594
				50	0,922426	0,793208		
				101	0,923146	0,798586		
			50	20	0,923236	0,798587	0,924466043	0,800274718
				50	0,925396	0,798693		
				101	0,924766	0,803545		
			70	20	0,923146	0,797243	0,925095992	0,802088254
				50	0,927196	0,804894		
				101	0,924946	0,804128		
			90	20	0,921256	0,792264	0,923596112	0,798172179
				50	0,924766	0,798559		
				101	0,924766	0,803694		
		yes	30	20	0,915677	0,790703	0,921226302	0,79686654
				50	0,926476	0,803965		
				101	0,921526	0,795932		
			50	20	0,910277	0,785517	0,915946724	0,792026268
				50	0,914957	0,789498		
				101	0,922606	0,801064		
			70	20	0,918287	0,790942	0,916336693	0,792416111
				50	0,910727	0,787556		
				101	0,919996	0,79875		
			90	20	0,901728	0,775197	0,915226782	0,790715594
				50	0,921526	0,796079		
				101	0,922426	0,800871		

Table E.17: Results of predicting the delivery day with the ELU activation function part 1.



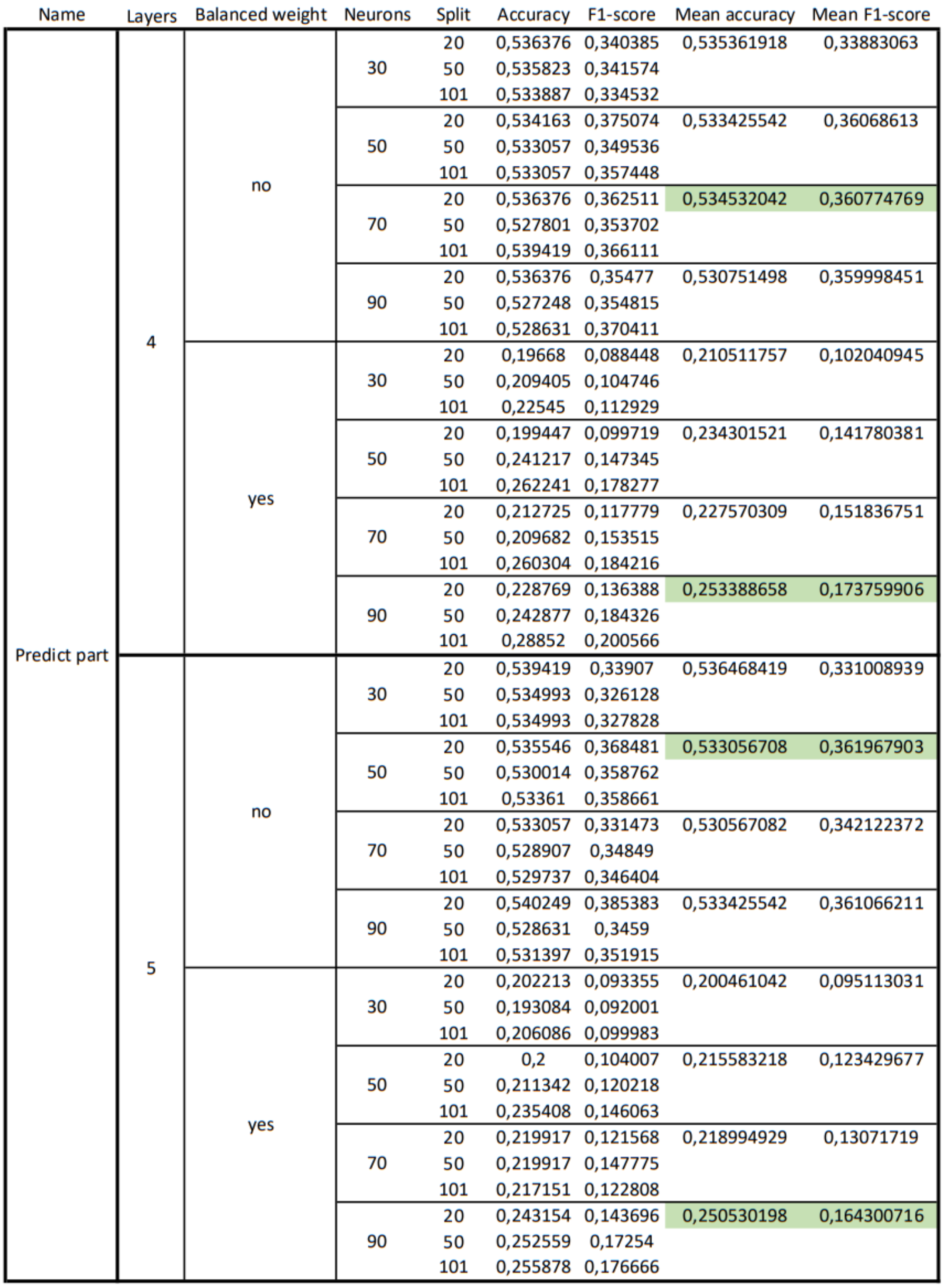
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict day	4	no	30	20	0,919996	0,791923	0,92137629	0,793829254
				50	0,923776	0,797163		
				101	0,920356	0,792402		
			50	20	0,924046	0,7986	0,924406048	0,799481388
				50	0,925666	0,799697		
				101	0,923506	0,800148		
			70	20	0,923866	0,799062	0,924466043	0,800267215
				50	0,926116	0,801931		
				101	0,923416	0,799809		
			90	20	0,922426	0,795546	0,923896088	0,798704086
				50	0,923686	0,794546		
				101	0,925576	0,806021		
		yes	30	20	0,921256	0,792574	0,923056156	0,796845795
				50	0,926206	0,801777		
				101	0,921706	0,796187		
			50	20	0,922246	0,796747	0,921946244	0,796441713
				50	0,922606	0,796719		
				101	0,920986	0,795859		
			70	20	0,922156	0,796807	0,923056156	0,799484989
				50	0,924046	0,800579		
				101	0,922966	0,801069		
			90	20	0,923596	0,800096	0,921796256	0,798861268
				50	0,920086	0,794955		
				101	0,921706	0,801532		
	5	no	30	20	0,917387	0,784041	0,920536357	0,79124745
				50	0,922606	0,79313		
				101	0,921616	0,796571		
			50	20	0,919546	0,789879	0,922786177	0,796745637
				50	0,925576	0,799515		
				101	0,923236	0,800842		
			70	20	0,922336	0,794705	0,923086153	0,797214262
				50	0,924856	0,799627		
				101	0,922066	0,797311		
			90	20	0,923416	0,798051	0,923956084	0,79981937
				50	0,924496	0,798928		
				101	0,923956	0,80248		
		yes	30	20	0,920176	0,789286	0,922336213	0,795333314
				50	0,923866	0,795925		
				101	0,922966	0,800789		
			50	20	0,923596	0,798633	0,922186225	0,798451772
				50	0,925216	0,800405		
				101	0,917747	0,796318		
			70	20	0,920896	0,794971	0,922516199	0,797868991
				50	0,924946	0,800666		
				101	0,921706	0,797971		
			90	20	0,918737	0,793614	0,920776338	0,797439741
				50	0,920176	0,797017		
				101	0,923416	0,801689		

Table E.18: Results of predicting the delivery day with the ELU activation function part 2.



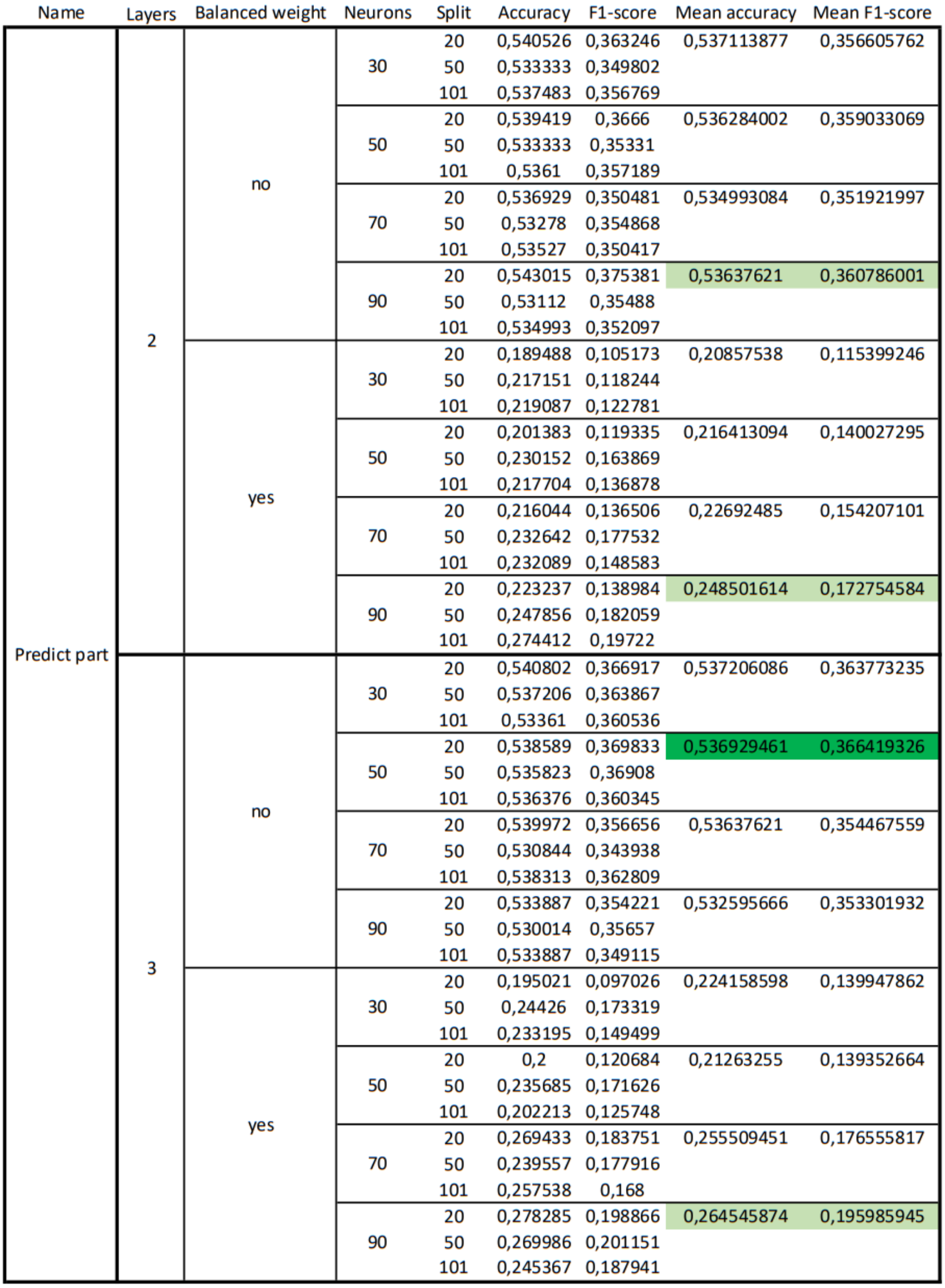
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict part	2	no	30	20	0,538313	0,362279	0,535454126	0,358797328
				50	0,534716	0,363567		
				101	0,533333	0,350546		
			50	20	0,540802	0,361853	0,537113877	0,359335352
				50	0,533057	0,363905		
				101	0,537483	0,352247		
			70	20	0,539419	0,35771	0,534716459	0,354344709
				50	0,527801	0,353636		
				101	0,536929	0,351687		
			90	20	0,53444	0,365149	0,532134624	0,358133828
				50	0,531397	0,359235		
				101	0,530567	0,350018		
		yes	30	20	0,220194	0,146127	0,223974182	0,127940533
				50	0,228216	0,122504		
				101	0,223513	0,115191		
			50	20	0,229876	0,145878	0,23669894	0,146484802
				50	0,237068	0,137513		
				101	0,243154	0,156064		
			70	20	0,229322	0,156145	0,238819733	0,154273995
				50	0,228492	0,145298		
				101	0,258645	0,161379		
			90	20	0,24675	0,154086	0,245919779	0,164417142
				50	0,252559	0,19033		
				101	0,238451	0,148836		
	3	no	30	20	0,538313	0,354929	0,537759336	0,348831436
				50	0,53527	0,354317		
				101	0,539696	0,337248		
			50	20	0,535546	0,379892	0,533794375	0,365796299
				50	0,527248	0,360358		
				101	0,538589	0,357139		
			70	20	0,538589	0,353177	0,534163209	0,357290959
				50	0,527248	0,361286		
				101	0,536653	0,35741		
			90	20	0,542462	0,369103	0,532503458	0,358541347
				50	0,527248	0,355736		
				101	0,527801	0,350785		
		yes	30	20	0,216321	0,107753	0,200184417	0,095848639
				50	0,188658	0,085052		
				101	0,195574	0,094741		
			50	20	0,234855	0,135397	0,229875519	0,142081019
				50	0,237344	0,165726		
				101	0,217427	0,125121		
			70	20	0,211065	0,115609	0,235684647	0,150336869
				50	0,264454	0,181785		
				101	0,231535	0,153617		
			90	20	0,248963	0,158362	0,250806823	0,173901166
				50	0,256432	0,194657		
				101	0,247026	0,168685		

Table E.19: Results of predicting the day part of the delivery with the ReLu activation function part 1.



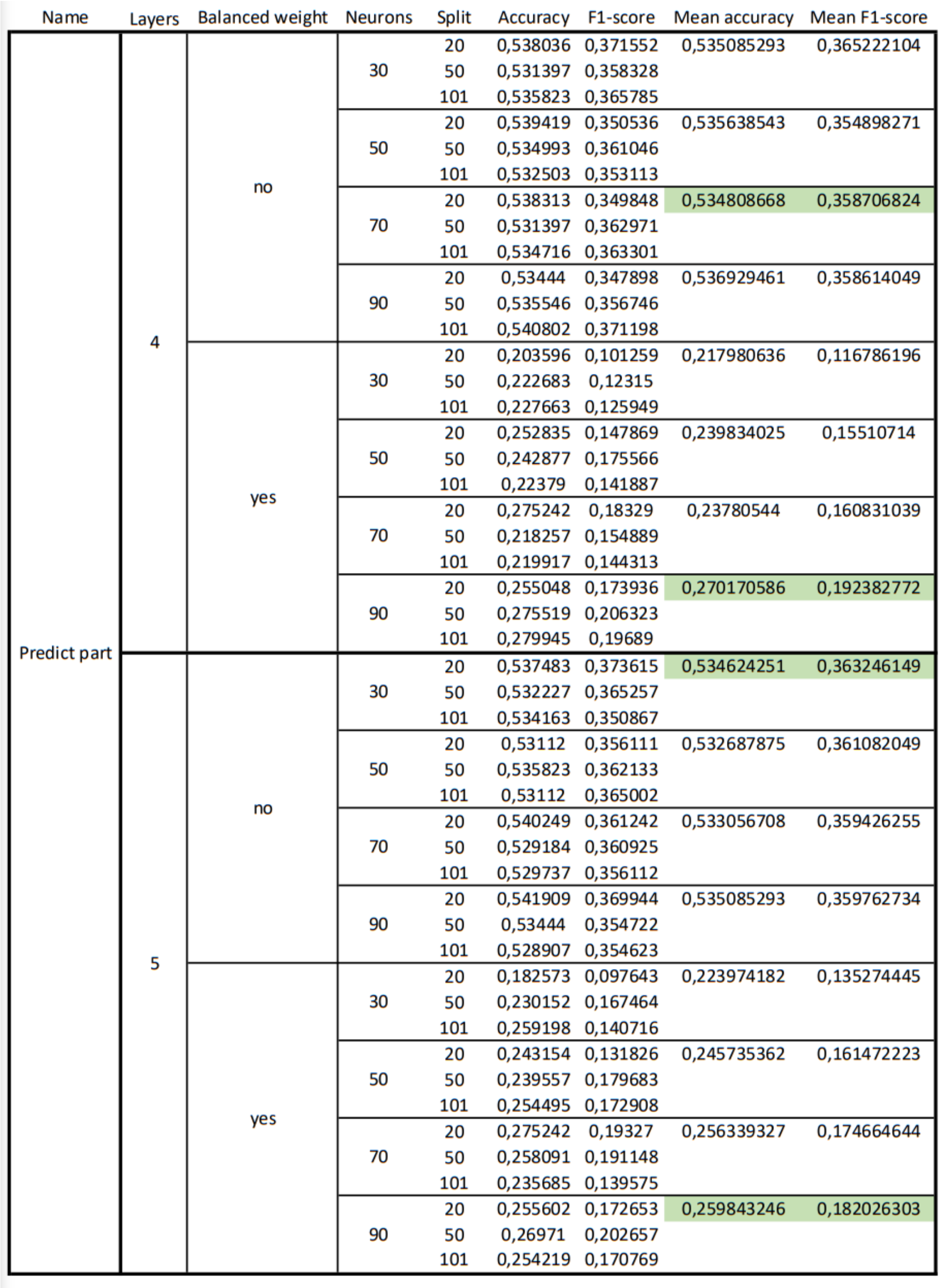
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict part	4	no	30	20	0,536376	0,340385	0,535361918	0,33883063
				50	0,535823	0,341574		
				101	0,533887	0,334532		
			50	20	0,534163	0,375074	0,533425542	0,36068613
				50	0,533057	0,349536		
				101	0,533057	0,357448		
			70	20	0,536376	0,362511	0,534532042	0,360774769
				50	0,527801	0,353702		
				101	0,539419	0,366111		
			90	20	0,536376	0,35477	0,530751498	0,359998451
				50	0,527248	0,354815		
				101	0,528631	0,370411		
		yes	30	20	0,19668	0,088448	0,210511757	0,102040945
				50	0,209405	0,104746		
				101	0,22545	0,112929		
			50	20	0,199447	0,099719	0,234301521	0,141780381
				50	0,241217	0,147345		
				101	0,262241	0,178277		
			70	20	0,212725	0,117779	0,227570309	0,151836751
				50	0,209682	0,153515		
				101	0,260304	0,184216		
			90	20	0,228769	0,136388	0,253388658	0,173759906
				50	0,242877	0,184326		
				101	0,28852	0,200566		
	5	no	30	20	0,539419	0,33907	0,536468419	0,331008939
				50	0,534993	0,326128		
				101	0,534993	0,327828		
			50	20	0,535546	0,368481	0,533056708	0,361967903
				50	0,530014	0,358762		
				101	0,53361	0,358661		
			70	20	0,533057	0,331473	0,530567082	0,342122372
				50	0,528907	0,34849		
				101	0,529737	0,346404		
			90	20	0,540249	0,385383	0,533425542	0,361066211
				50	0,528631	0,3459		
				101	0,531397	0,351915		
		yes	30	20	0,202213	0,093355	0,200461042	0,095113031
				50	0,193084	0,092001		
				101	0,206086	0,099983		
			50	20	0,2	0,104007	0,215583218	0,123429677
				50	0,211342	0,120218		
				101	0,235408	0,146063		
			70	20	0,219917	0,121568	0,218994929	0,13071719
				50	0,219917	0,147775		
				101	0,217151	0,122808		
			90	20	0,243154	0,143696	0,250530198	0,164300716
				50	0,252559	0,17254		
				101	0,255878	0,176666		

Table E.20: Results of predicting the day part of the delivery with the ReLu activation function part 2.



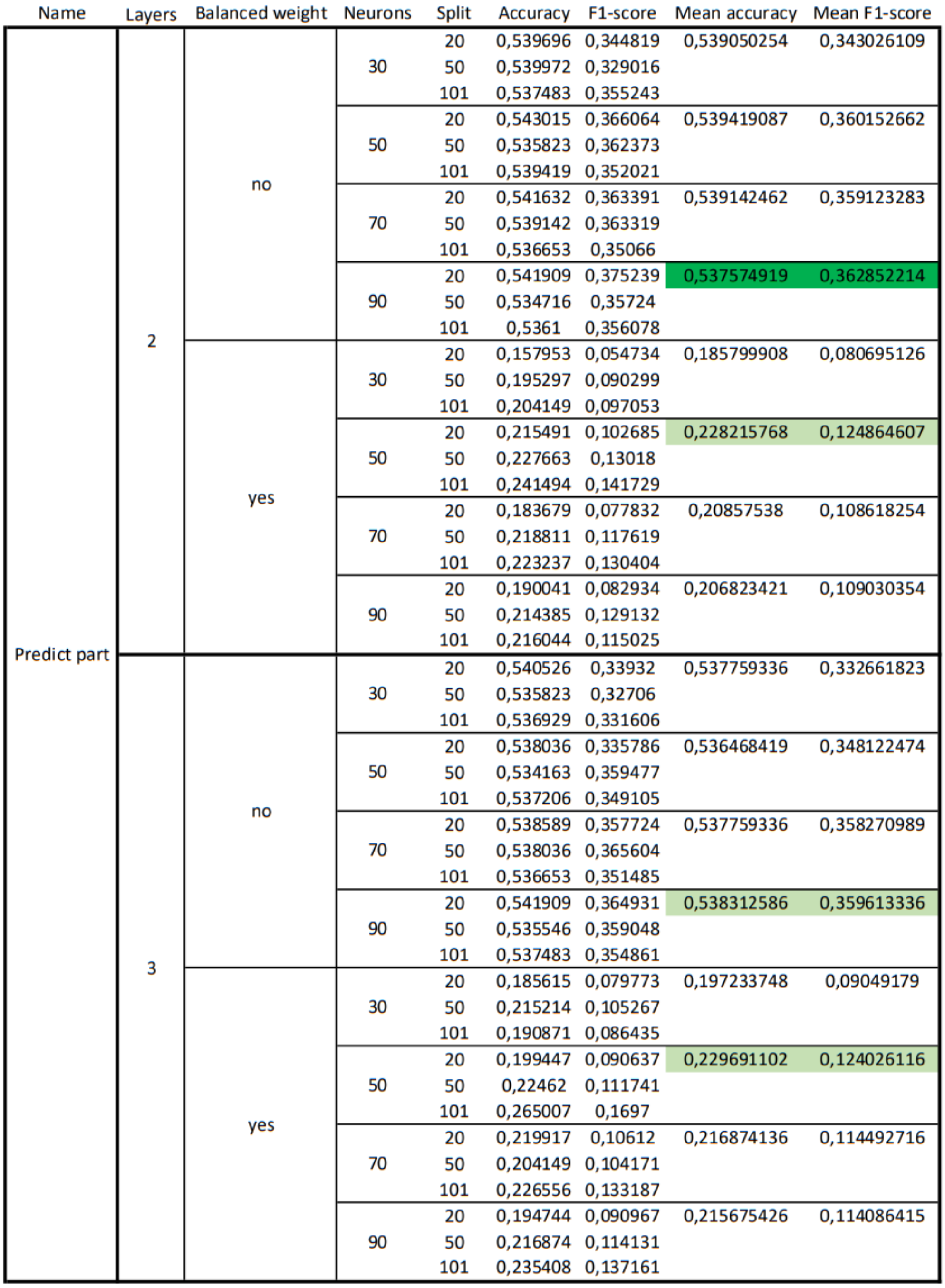
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict part	2	no	30	20	0,540526	0,363246	0,537113877	0,356605762
				50	0,533333	0,349802		
				101	0,537483	0,356769		
			50	20	0,539419	0,3666	0,536284002	0,359033069
				50	0,533333	0,35331		
				101	0,5361	0,357189		
			70	20	0,536929	0,350481	0,534993084	0,351921997
				50	0,53278	0,354868		
				101	0,53527	0,350417		
			90	20	0,543015	0,375381	0,53637621	0,360786001
				50	0,53112	0,35488		
				101	0,534993	0,352097		
		yes	30	20	0,189488	0,105173	0,20857538	0,115399246
				50	0,217151	0,118244		
				101	0,219087	0,122781		
			50	20	0,201383	0,119335	0,216413094	0,140027295
				50	0,230152	0,163869		
				101	0,217704	0,136878		
			70	20	0,216044	0,136506	0,22692485	0,154207101
				50	0,232642	0,177532		
				101	0,232089	0,148583		
			90	20	0,223237	0,138984	0,248501614	0,172754584
				50	0,247856	0,182059		
				101	0,274412	0,19722		
	3	no	30	20	0,540802	0,366917	0,537206086	0,363773235
				50	0,537206	0,363867		
				101	0,53361	0,360536		
			50	20	0,538589	0,369833	0,536929461	0,366419326
				50	0,535823	0,36908		
				101	0,536376	0,360345		
			70	20	0,539972	0,356656	0,53637621	0,354467559
				50	0,530844	0,343938		
				101	0,538313	0,362809		
			90	20	0,533887	0,354221	0,532595666	0,353301932
				50	0,530014	0,35657		
				101	0,533887	0,349115		
		yes	30	20	0,195021	0,097026	0,224158598	0,139947862
				50	0,24426	0,173319		
				101	0,233195	0,149499		
			50	20	0,2	0,120684	0,21263255	0,139352664
				50	0,235685	0,171626		
				101	0,202213	0,125748		
			70	20	0,269433	0,183751	0,255509451	0,176555817
				50	0,239557	0,177916		
				101	0,257538	0,168		
			90	20	0,278285	0,198866	0,264545874	0,195985945
				50	0,269986	0,201151		
				101	0,245367	0,187941		

Table E.21: Results of predicting the day part of the delivery with the Tanh activation function part 1.



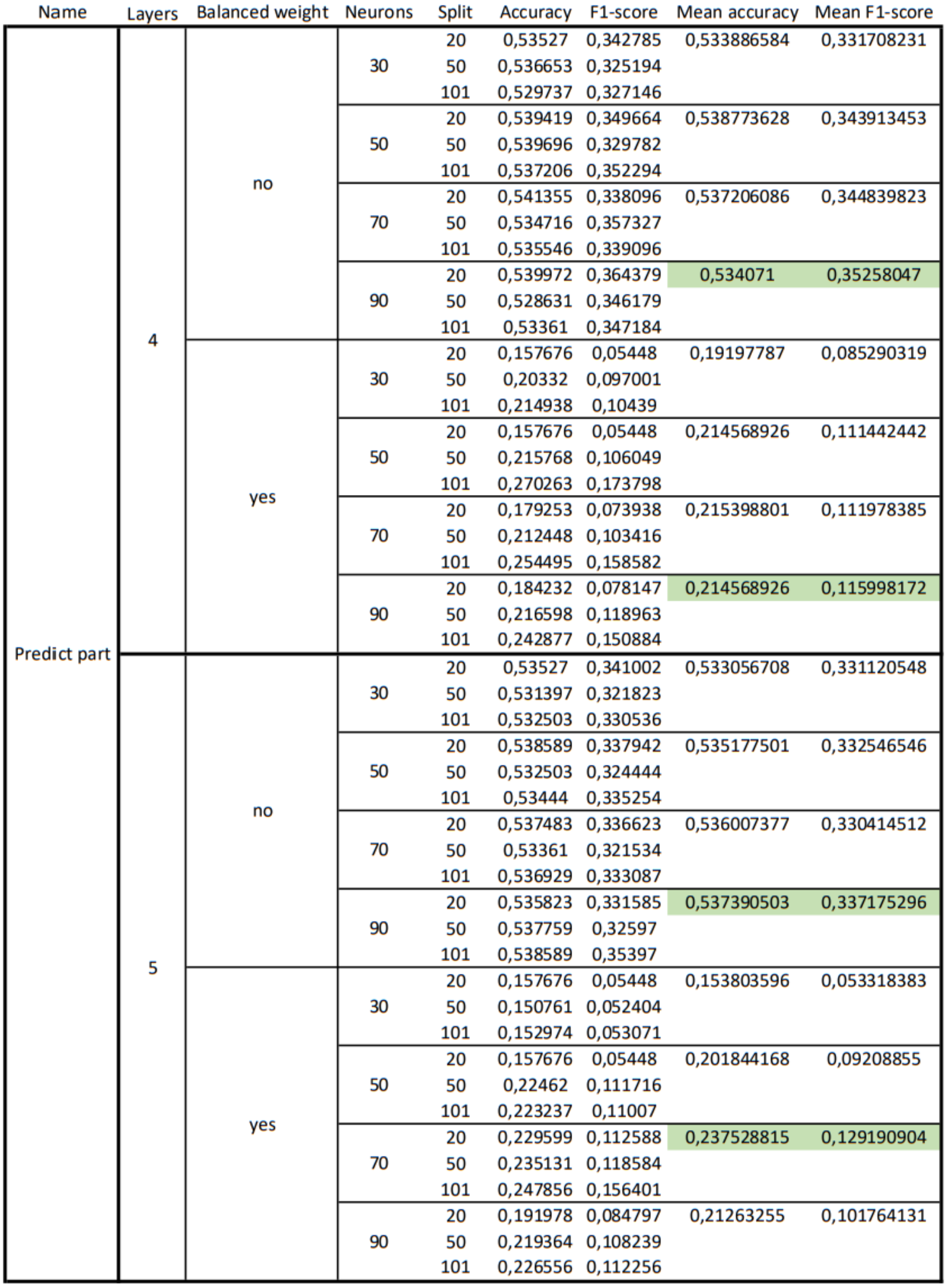
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict part	4	no	30	20	0,538036	0,371552	0,535085293	0,365222104
				50	0,531397	0,358328		
				101	0,535823	0,365785		
			50	20	0,539419	0,350536	0,535638543	0,354898271
				50	0,534993	0,361046		
				101	0,532503	0,353113		
			70	20	0,538313	0,349848	0,534808668	0,358706824
				50	0,531397	0,362971		
				101	0,534716	0,363301		
			90	20	0,53444	0,347898	0,536929461	0,358614049
				50	0,535546	0,356746		
				101	0,540802	0,371198		
		yes	30	20	0,203596	0,101259	0,217980636	0,116786196
				50	0,222683	0,12315		
				101	0,227663	0,125949		
			50	20	0,252835	0,147869	0,239834025	0,15510714
				50	0,242877	0,175566		
				101	0,22379	0,141887		
			70	20	0,275242	0,18329	0,23780544	0,160831039
				50	0,218257	0,154889		
				101	0,219917	0,144313		
			90	20	0,255048	0,173936	0,270170586	0,192382772
				50	0,275519	0,206323		
				101	0,279945	0,19689		
	5	no	30	20	0,537483	0,373615	0,534624251	0,363246149
				50	0,532227	0,365257		
				101	0,534163	0,350867		
			50	20	0,53112	0,356111	0,532687875	0,361082049
				50	0,535823	0,362133		
				101	0,53112	0,365002		
			70	20	0,540249	0,361242	0,533056708	0,359426255
				50	0,529184	0,360925		
				101	0,529737	0,356112		
			90	20	0,541909	0,369944	0,535085293	0,359762734
				50	0,53444	0,354722		
				101	0,528907	0,354623		
		yes	30	20	0,182573	0,097643	0,223974182	0,135274445
				50	0,230152	0,167464		
				101	0,259198	0,140716		
			50	20	0,243154	0,131826	0,245735362	0,161472223
				50	0,239557	0,179683		
				101	0,254495	0,172908		
			70	20	0,275242	0,19327	0,256339327	0,174664644
				50	0,258091	0,191148		
				101	0,235685	0,139575		
			90	20	0,255602	0,172653	0,259843246	0,182026303
				50	0,26971	0,202657		
				101	0,254219	0,170769		

Table E.22: Results of predicting the day part of the delivery with the Tanh activation function part 2.



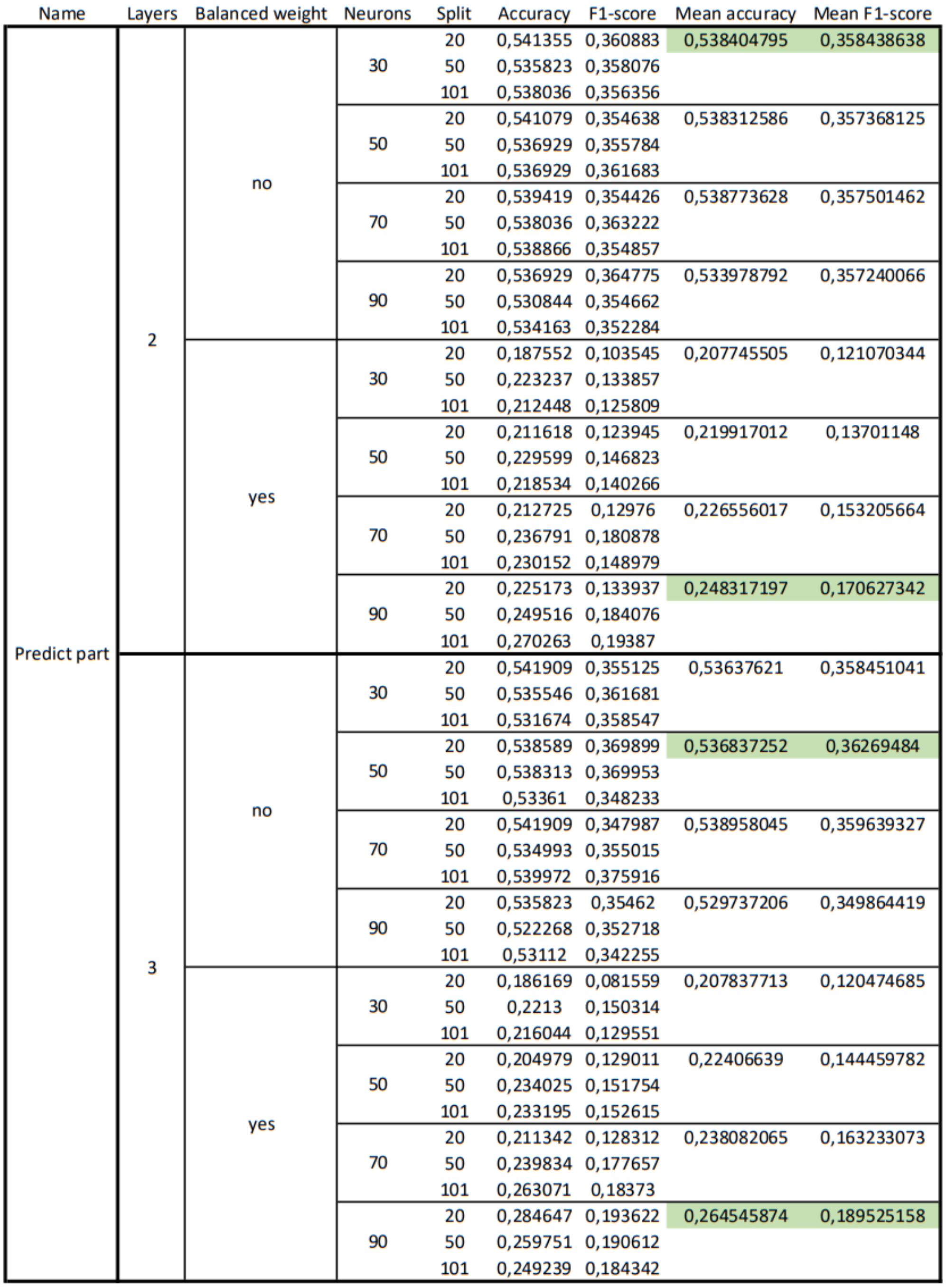
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict part	2	no	30	20	0,539696	0,344819	0,539050254	0,343026109
				50	0,539972	0,329016		
				101	0,537483	0,355243		
			50	20	0,543015	0,366064	0,539419087	0,360152662
				50	0,535823	0,362373		
				101	0,539419	0,352021		
			70	20	0,541632	0,363391	0,539142462	0,359123283
				50	0,539142	0,363319		
				101	0,536653	0,35066		
			90	20	0,541909	0,375239	0,537574919	0,362852214
				50	0,534716	0,35724		
				101	0,5361	0,356078		
		yes	30	20	0,157953	0,054734	0,185799908	0,080695126
				50	0,195297	0,090299		
				101	0,204149	0,097053		
			50	20	0,215491	0,102685	0,228215768	0,124864607
				50	0,227663	0,13018		
				101	0,241494	0,141729		
			70	20	0,183679	0,077832	0,20857538	0,108618254
				50	0,218811	0,117619		
				101	0,223237	0,130404		
			90	20	0,190041	0,082934	0,206823421	0,109030354
				50	0,214385	0,129132		
				101	0,216044	0,115025		
	3	no	30	20	0,540526	0,33932	0,537759336	0,332661823
				50	0,535823	0,32706		
				101	0,536929	0,331606		
			50	20	0,538036	0,335786	0,536468419	0,348122474
				50	0,534163	0,359477		
				101	0,537206	0,349105		
			70	20	0,538589	0,357724	0,537759336	0,358270989
				50	0,538036	0,365604		
				101	0,536653	0,351485		
			90	20	0,541909	0,364931	0,538312586	0,359613336
				50	0,535546	0,359048		
				101	0,537483	0,354861		
		yes	30	20	0,185615	0,079773	0,197233748	0,09049179
				50	0,215214	0,105267		
				101	0,190871	0,086435		
			50	20	0,199447	0,090637	0,229691102	0,124026116
				50	0,22462	0,111741		
				101	0,265007	0,1697		
			70	20	0,219917	0,10612	0,216874136	0,114492716
				50	0,204149	0,104171		
				101	0,226556	0,133187		
			90	20	0,194744	0,090967	0,215675426	0,114086415
				50	0,216874	0,114131		
				101	0,235408	0,137161		

Table E.23: Results of predicting the day part of the delivery with the Sigmoid activation function part 1.



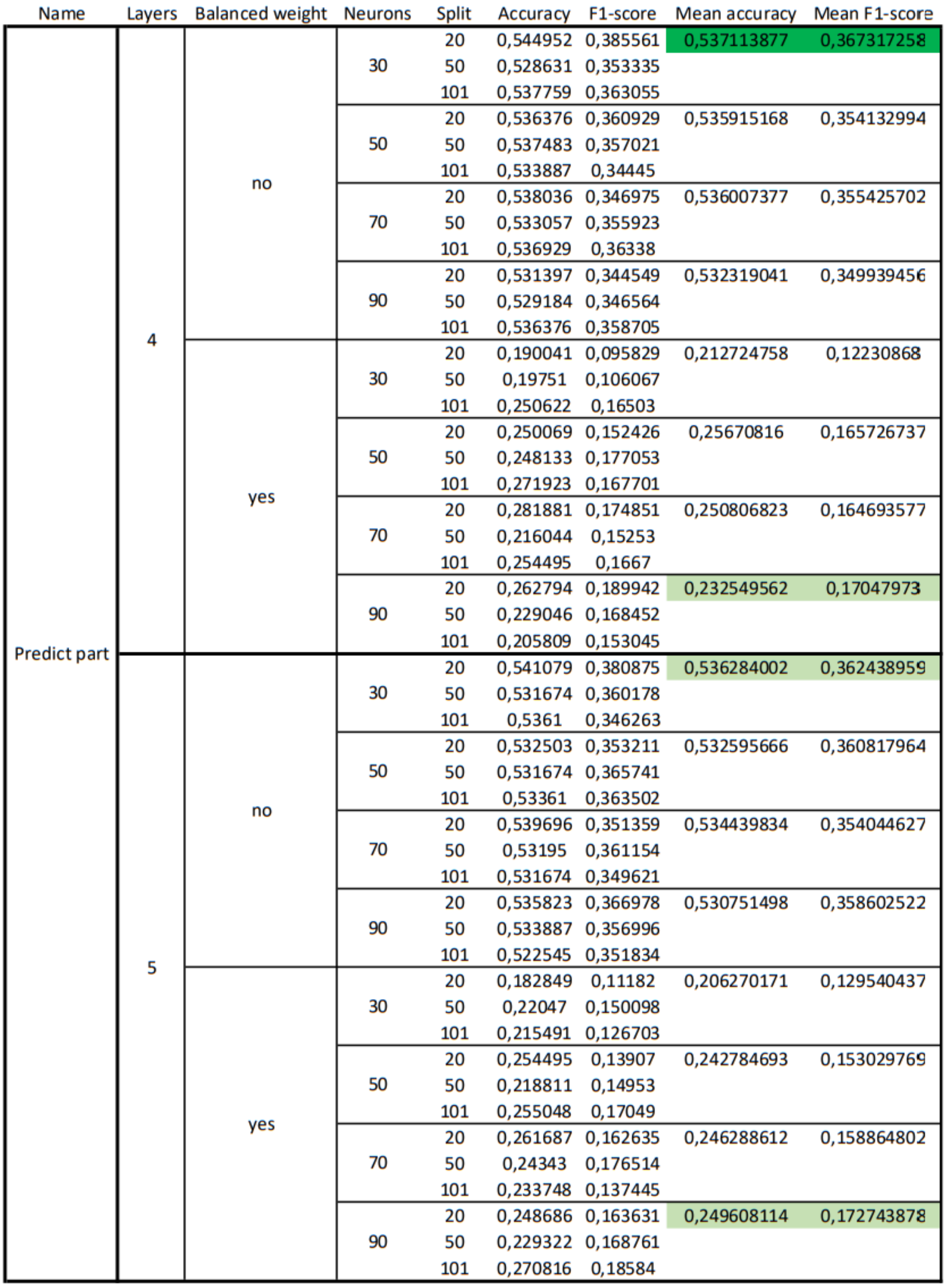
Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict part	4	no	30	20	0,53527	0,342785	0,533886584	0,331708231
				50	0,536653	0,325194		
				101	0,529737	0,327146		
			50	20	0,539419	0,349664	0,538773628	0,343913453
				50	0,539696	0,329782		
				101	0,537206	0,352294		
			70	20	0,541355	0,338096	0,537206086	0,344839823
				50	0,534716	0,357327		
				101	0,535546	0,339096		
			90	20	0,539972	0,364379	0,534071	0,35258047
				50	0,528631	0,346179		
				101	0,53361	0,347184		
		yes	30	20	0,157676	0,05448	0,19197787	0,085290319
				50	0,20332	0,097001		
				101	0,214938	0,10439		
			50	20	0,157676	0,05448	0,214568926	0,111442442
				50	0,215768	0,106049		
				101	0,270263	0,173798		
			70	20	0,179253	0,073938	0,215398801	0,111978385
				50	0,212448	0,103416		
				101	0,254495	0,158582		
			90	20	0,184232	0,078147	0,214568926	0,115998172
				50	0,216598	0,118963		
				101	0,242877	0,150884		
	5	no	30	20	0,53527	0,341002	0,533056708	0,331120548
				50	0,531397	0,321823		
				101	0,532503	0,330536		
			50	20	0,538589	0,337942	0,535177501	0,332546546
				50	0,532503	0,324444		
				101	0,53444	0,335254		
			70	20	0,537483	0,336623	0,536007377	0,330414512
				50	0,53361	0,321534		
				101	0,536929	0,333087		
			90	20	0,535823	0,331585	0,537390503	0,337175296
				50	0,537759	0,32597		
				101	0,538589	0,35397		
		yes	30	20	0,157676	0,05448	0,153803596	0,053318383
				50	0,150761	0,052404		
				101	0,152974	0,053071		
			50	20	0,157676	0,05448	0,201844168	0,09208855
				50	0,22462	0,111716		
				101	0,223237	0,11007		
			70	20	0,229599	0,112588	0,237528815	0,129190904
				50	0,235131	0,118584		
				101	0,247856	0,156401		
			90	20	0,191978	0,084797	0,21263255	0,101764131
				50	0,219364	0,108239		
				101	0,226556	0,112256		

Table E.24: Results of predicting the day part of the delivery with the Sigmoid activation function part 2.



Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict part	2	no	30	20	0,541355	0,360883	0,538404795	0,358438638
				50	0,535823	0,358076		
				101	0,538036	0,356356		
			50	20	0,541079	0,354638	0,538312586	0,357368125
				50	0,536929	0,355784		
				101	0,536929	0,361683		
			70	20	0,539419	0,354426	0,538773628	0,357501462
				50	0,538036	0,363222		
				101	0,538866	0,354857		
			90	20	0,536929	0,364775	0,533978792	0,357240066
				50	0,530844	0,354662		
				101	0,534163	0,352284		
		yes	30	20	0,187552	0,103545	0,207745505	0,121070344
				50	0,223237	0,133857		
				101	0,212448	0,125809		
			50	20	0,211618	0,123945	0,219917012	0,13701148
				50	0,229599	0,146823		
				101	0,218534	0,140266		
			70	20	0,212725	0,12976	0,226556017	0,153205664
				50	0,236791	0,180878		
				101	0,230152	0,148979		
			90	20	0,225173	0,133937	0,248317197	0,170627342
				50	0,249516	0,184076		
				101	0,270263	0,19387		
	3	no	30	20	0,541909	0,355125	0,53637621	0,358451041
				50	0,535546	0,361681		
				101	0,531674	0,358547		
			50	20	0,538589	0,369899	0,536837252	0,36269484
				50	0,538313	0,369953		
				101	0,53361	0,348233		
			70	20	0,541909	0,347987	0,538958045	0,359639327
				50	0,534993	0,355015		
				101	0,539972	0,375916		
			90	20	0,535823	0,35462	0,529737206	0,349864419
				50	0,522268	0,352718		
				101	0,53112	0,342255		
		yes	30	20	0,186169	0,081559	0,207837713	0,120474685
				50	0,2213	0,150314		
				101	0,216044	0,129551		
			50	20	0,204979	0,129011	0,22406639	0,144459782
				50	0,234025	0,151754		
				101	0,233195	0,152615		
			70	20	0,211342	0,128312	0,238082065	0,163233073
				50	0,239834	0,177657		
				101	0,263071	0,18373		
			90	20	0,284647	0,193622	0,264545874	0,189525158
				50	0,259751	0,190612		
				101	0,249239	0,184342		

Table E.25: Results of predicting the day part of the delivery with the ELU activation function part 1.



Name	Layers	Balanced weight	Neurons	Split	Accuracy	F1-score	Mean accuracy	Mean F1-score
Predict part	4	no	30	20	0,544952	0,385561	0,537113877	0,367317258
				50	0,528631	0,353335		
				101	0,537759	0,363055		
			50	20	0,536376	0,360929	0,535915168	0,354132994
				50	0,537483	0,357021		
				101	0,533887	0,34445		
			70	20	0,538036	0,346975	0,536007377	0,355425702
				50	0,533057	0,355923		
				101	0,536929	0,36338		
			90	20	0,531397	0,344549	0,532319041	0,349939456
				50	0,529184	0,346564		
				101	0,536376	0,358705		
		yes	30	20	0,190041	0,095829	0,212724758	0,12230868
				50	0,19751	0,106067		
				101	0,250622	0,16503		
			50	20	0,250069	0,152426	0,25670816	0,165726737
				50	0,248133	0,177053		
				101	0,271923	0,167701		
			70	20	0,281881	0,174851	0,250806823	0,164693577
				50	0,216044	0,15253		
				101	0,254495	0,1667		
			90	20	0,262794	0,189942	0,232549562	0,17047973
				50	0,229046	0,168452		
				101	0,205809	0,153045		
	5	no	30	20	0,541079	0,380875	0,536284002	0,362438959
				50	0,531674	0,360178		
				101	0,5361	0,346263		
			50	20	0,532503	0,353211	0,532595666	0,360817964
				50	0,531674	0,365741		
				101	0,53361	0,363502		
			70	20	0,539696	0,351359	0,534439834	0,354044627
				50	0,53195	0,361154		
				101	0,531674	0,349621		
			90	20	0,535823	0,366978	0,530751498	0,358602522
				50	0,533887	0,356996		
				101	0,522545	0,351834		
		yes	30	20	0,182849	0,11182	0,206270171	0,129540437
				50	0,22047	0,150098		
				101	0,215491	0,126703		
			50	20	0,254495	0,13907	0,242784693	0,153029769
				50	0,218811	0,14953		
				101	0,255048	0,17049		
			70	20	0,261687	0,162635	0,246288612	0,158864802
				50	0,24343	0,176514		
				101	0,233748	0,137445		
			90	20	0,248686	0,163631	0,249608114	0,172743878
				50	0,229322	0,168761		
				101	0,270816	0,18584		

Table E.26: Results of predicting the day part of the delivery with the ELU activation function part 2.

F

SHAP values

In recent years, the explainability of ML models has become one of the most discussed topics in ML [71]. SHAP values are a method for explaining the predictions of ML models and eliminating the black box idea, as they do not directly provide information about the feature's importance. The SHAP values are created based on a feature attribution method, where a particular value is assigned to each feature, making it possible to interpret the predictions [76]. In other words, contribution values for each feature of each data point will be assigned during this approach. Based on these values, the given importance that a model provides to a feature is encoded as contribution information and can be used to determine its importance. To be more precise, SHAP values approximate Shapley values, a concept from game theory that solves the problem of calculating the contribution to a model's prediction from every subset of features using a dataset with a given number of features. As an illustration, Shapley quantifies each player's contribution in a game, and SHAP quantifies each feature's contribution to the prediction model. Calculating the exact solution of Shapley values is infeasible due to the exponential nature of the problem. However, SHAP provides an approximation using special weighted linear regression, which can be applied to any model. The approximation is obtained by leveraging the local explainability property to construct surrogate models for ML models. To achieve this, SHAP slightly changes the input and tests the impact on the prediction. If the model prediction remains relatively stable despite the small input changes for a particular feature, that feature may not be a significant predictor for that specific data point [71]. However, the main concept is that the contribution of one feature does not depend only on a single feature but rather on the entire feature set.

The SHAP values can be calculated using the formula displayed in Equation F.1 where M indicates the number of features, ϕ_j is the feature attribution value of feature j the Shapley values. And z_j' represents the coalition vector that indicates whether feature j is being observed. In the coalition vector, one suggests that the feature is present, and zero indicates that the feature is absent.

$$g(z') = \phi_0 = \sum_{j=1}^{M} \phi_j z_j' \tag{F.1}$$

The Shapley values indicates as ϕ_j can be determined with Equation F.2, where S represents the a set containing non-zero indexes in z' and given a model f.

$$\phi_j = \sum_{S \subseteq N \setminus \{j\}} \frac{|S|!(M - |S| - 1)!}{m!} \left[f\left(S \cup \{j\} - f\left(S\right)\right) \right] \tag{F.2}$$

where N indicates the set of all input features.

To get insights into the model, the feature importance will be obtained and is based on the absolute Shapley values. The features with the average most significant values are considered the most important and can be determined with the following equation [67][76].

$$I_j = \frac{1}{n}\sum_{i=1}^{n}\left|\phi_j^{(i)}\right| \tag{F.3}$$

Based on this formula, the summary plots of SHAP values will be acquired. For more detailed information about which features are important for each class, more detailed plots will be provided. In the more detailed plots, each point indicated in red or blue is a Shapley value for a feature and an instance. The feature determines the position of the points on the y-axis, while the x-axis position represents the corresponding value. Low values will be indicated with blue shades, and the higher values with red. Points that are overlapping will be slightly jittered along the y-axis, making it possible to observe the distribution [78][70][100].

G

Feature importance

G.1. Feature importance MNL

In this part, an overview of the MNL's feature importance plots can be found. The feature importance plots are displayed for each possible output category, meaning that all days of the week, for example, have their feature importance plot. In addition, these plots can be used to compare the importance of the features between the MNL and NN models to indicate if the features are used similarly in both models.

G.1.1. Week

Feature importance for predicted deliveries in week 44

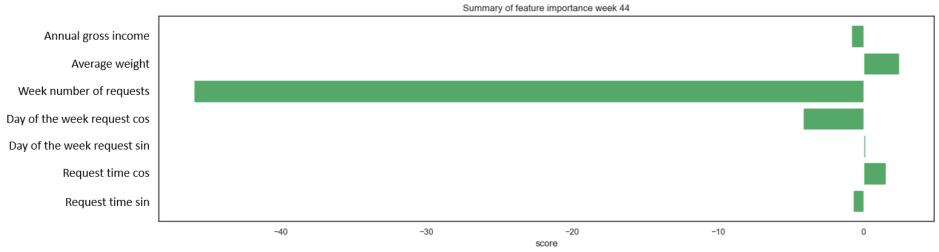


Figure G.1: Feature importance for predicted week 44

In this figure, an overview of the feature importance for predicted deliveries in week 44 can be found. Based on this figure, it can be found that the week number has a high impact on the prediction.

Feature importance for predicted deliveries in week 45

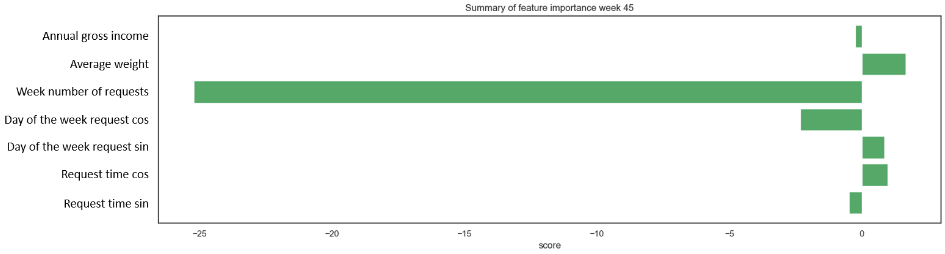


Figure G.2: Feature importance for predicted week 45

In this figure, the feature importance of predicted deliveries in week 45 can be found. When this figure is compared with the feature importance of week 44, it can be found that the impact of the week number when the request is made is declining. However, it is still the most important feature for predicting that the delivery will take place in week 45.

Feature importance for predicted deliveries in week 46

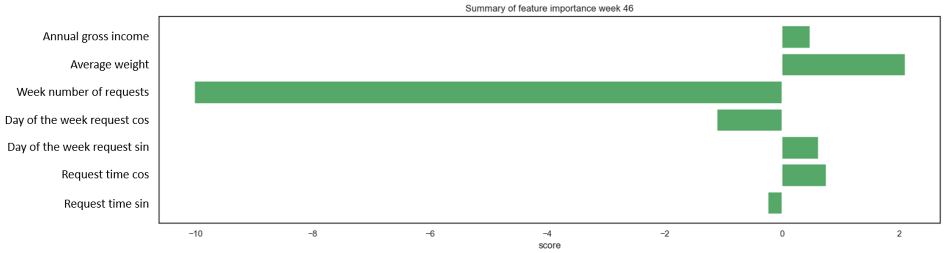


Figure G.3: Feature importance for predicted week 46

Based on the feature importance for predicted deliveries in week 46, a further decline in the importance of the request week can be found. Also, the importance of the Annual gross income shifts from negative to positive impacts on the predicted delivery week.

Feature importance for predicted deliveries in week 47

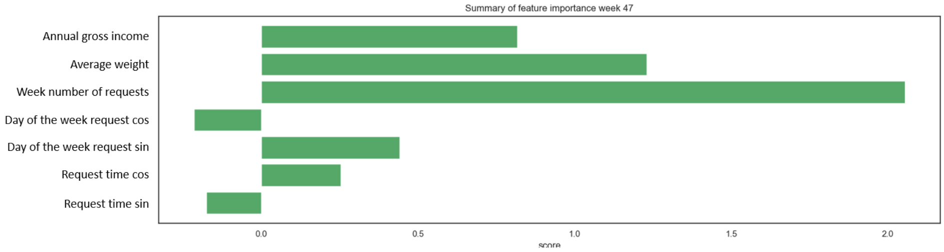


Figure G.4: Feature importance for predicted week 47

For the predicted deliveries in week 47, the feature importance plot changes a lot relative to the earlier seen feature importance plots. To predict that the delivery is taken place this week, the most important feature is still the week in which the request is made; however, the importance of the average weight and the annual gross income is proportionally growing.

Feature importance for predicted deliveries in week 48

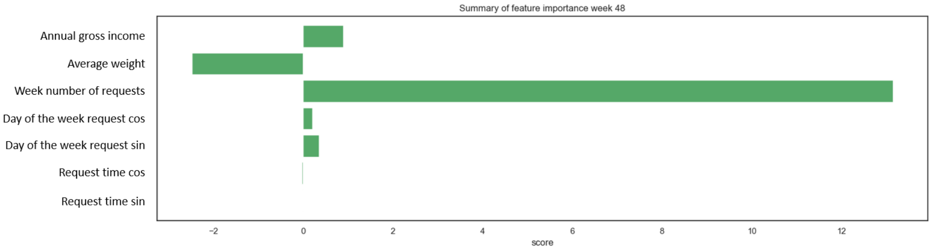


Figure G.5: Feature importance for predicted week 48

For deliveries predicted in week 48, it is found that the cyclical indicated request time and day are not that important. Another remarkable thing is that the impact of the average weight order switches from positive to negative.

Feature importance for predicted deliveries in week 49

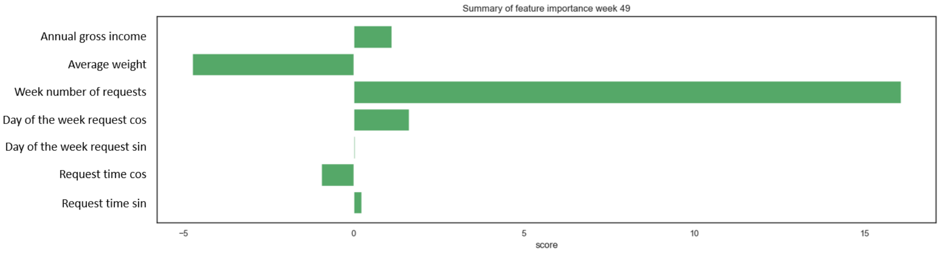


Figure G.6: Feature importance for predicted week 49

Based on the feature importance plot for predicted deliveries in week 49, it can be found that the importance of the request week is increasing again. Also, it can be found that the requested day gets more influence on the prediction.

Feature importance for predicted deliveries in week 50

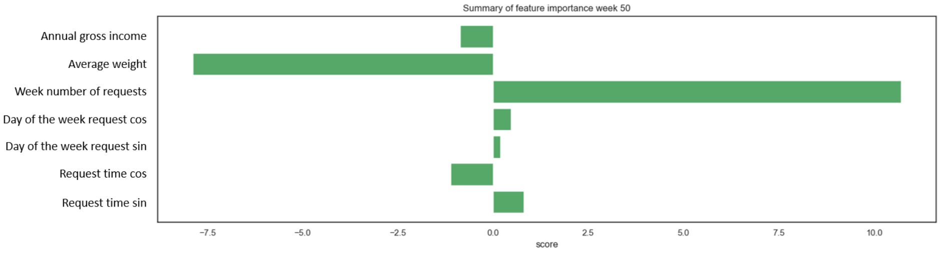


Figure G.7: Feature importance for predicted week 50

For week 50, the predicted deliveries mainly depend on the average weight of the order and the week number when the request is made. However, it can be found that the annual gross income gets a negative influence instead of a positive one.

Feature importance for predicted deliveries in week 51

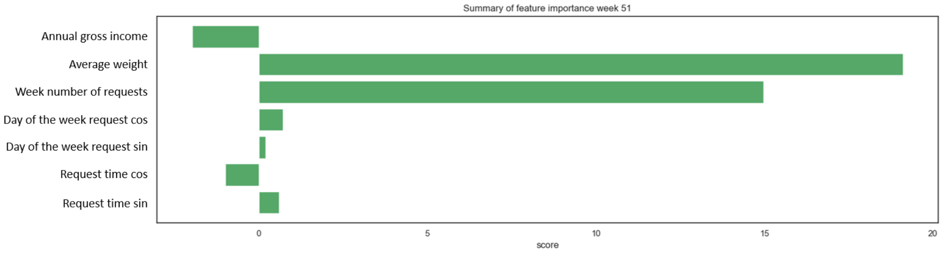


Figure G.8: Feature importance for predicted week 51

The feature importance plot for week 51 indicates that the average weight order has a negative impact and that importance of the cyclical request time is more significant.

Feature importance for predicted deliveries in week 51

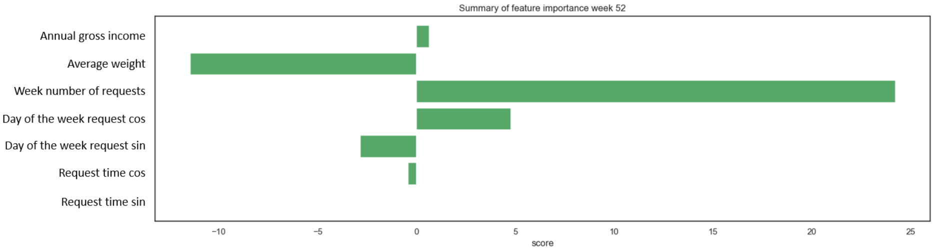


Figure G.9: Feature importance for predicted week 52

For the deliveries predicted in the last week of the year, the average weight order plays an important part in the predicting process. This can be explained by the fact that Christmas is coming, and customers want to order more than they normally do.

When comparing the feature importance plots over the week, it can be found that annual gross income, average weight order and the week in which the request is made the most important feature are. However, when the request time and day were left out as features for predicting the delivery week, the performance decreased. Therefore it was decided to include them as features in this prediction step.

G.1.2. Day

Feature importance for predicted deliveries on Monday

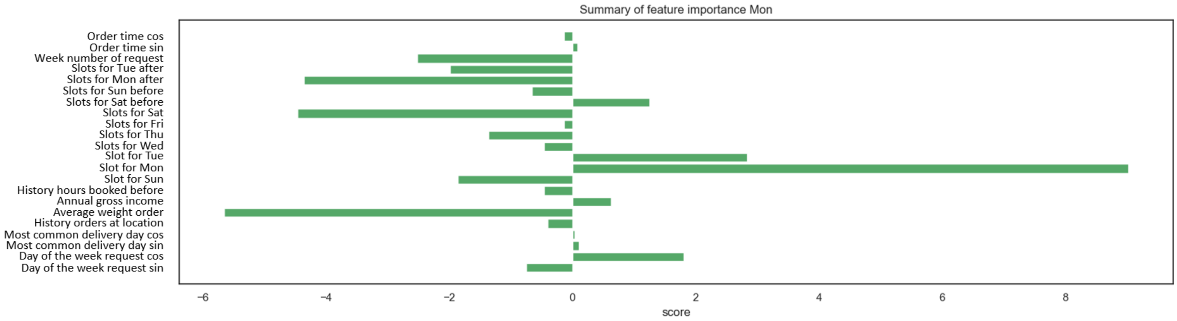


Figure G.10: Feature importance for predicted deliveries on Monday.

For predicting that the delivery will take place on Monday, the number of offered slots for Monday that week has a large positive impact compared to the Monday after the predicted delivery week. Since the model also includes the next Monday to compensate for possible small errors when predicting the delivery week. Another feature that seems to be important is the average weight order, which has a negative influence.

Feature importance for predicted deliveries on Tuesday

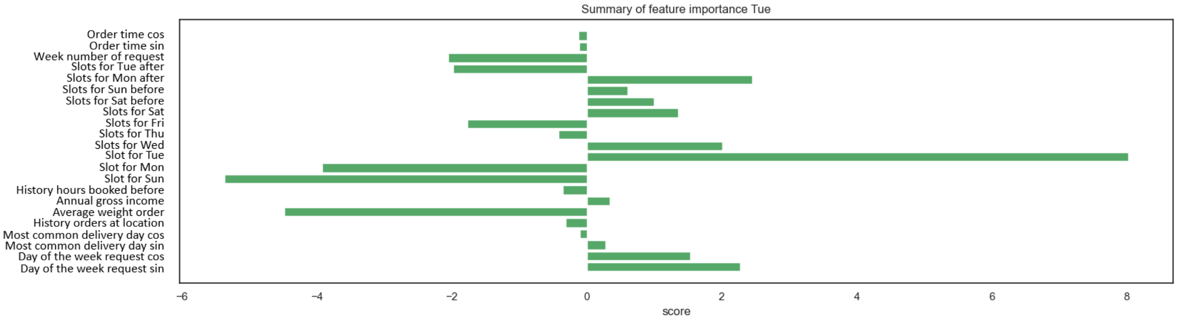


Figure G.11: Feature importance for predicted deliveries on Tuesday.

The most important feature is the number of offered slots for the day itself and the number of slots for Sunday in combination with the day of the week the request is made.

Feature importance for predicted deliveries on Wednesday

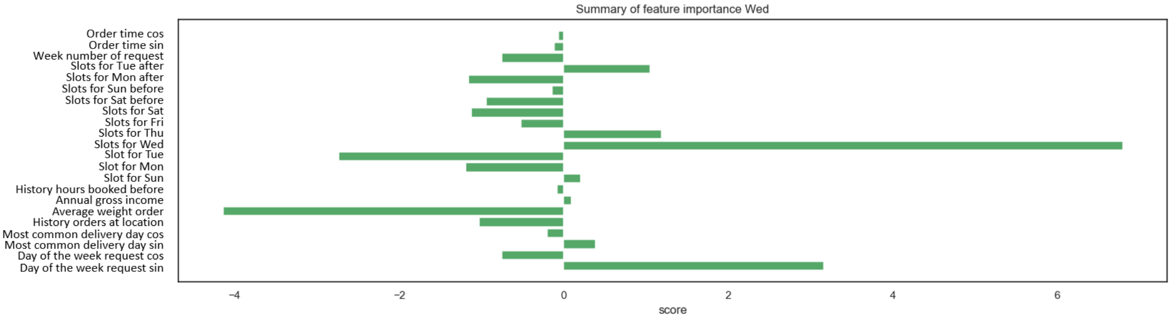


Figure G.12: Feature importance for predicted deliveries on Wednesday.

For predicting that the delivery takes place on Wednesday, it is found based on the feature importance plot that mainly the available slots on Wednesday are important. However, the requested day also impacts the delivery day.

Feature importance for predicted deliveries on Thursday

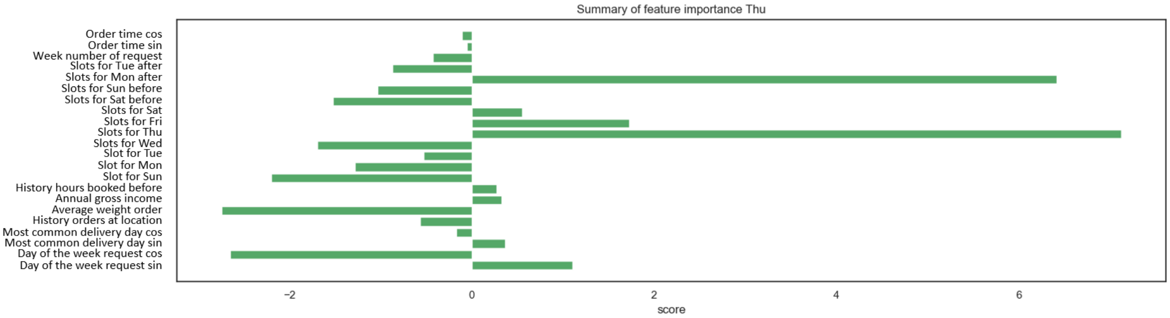


Figure G.13: Feature importance for predicted deliveries on Thursday.

The most important features that influence that the delivery is going to take place on Thursday are the number of available slots of that Thursday and the slots for the Monday after the selected delivery week. It is found that both features have a positive impact on the prediction of the delivery day.

Feature importance for predicted deliveries on Friday

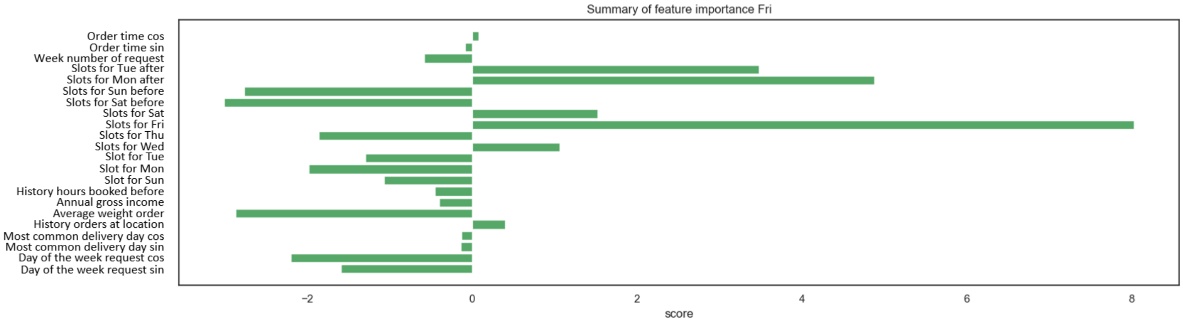


Figure G.14: Feature importance for predicted deliveries on Friday.

Based on the feature importance plot, it can be found that the number of available slots on Monday and Tuesday after the predicted delivery week and the Friday have the most influence on the predicting. Also, it is found that the number of available slots for most other delivery days is negative.

Feature importance for predicted deliveries on Saturday

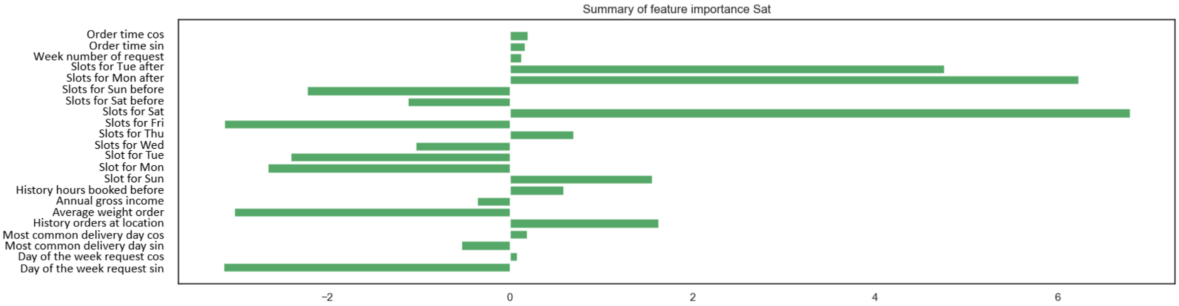


Figure G.15: Feature importance for predicted deliveries on Saturday.

Based on this plot, it can be found that the influence of the two additional days included after the delivery week gets more influence in predicting that the delivery takes place on Saturday. However, the most important feature is still the number of available slots for the day itself.

Feature importance for predicted deliveries on Sunday

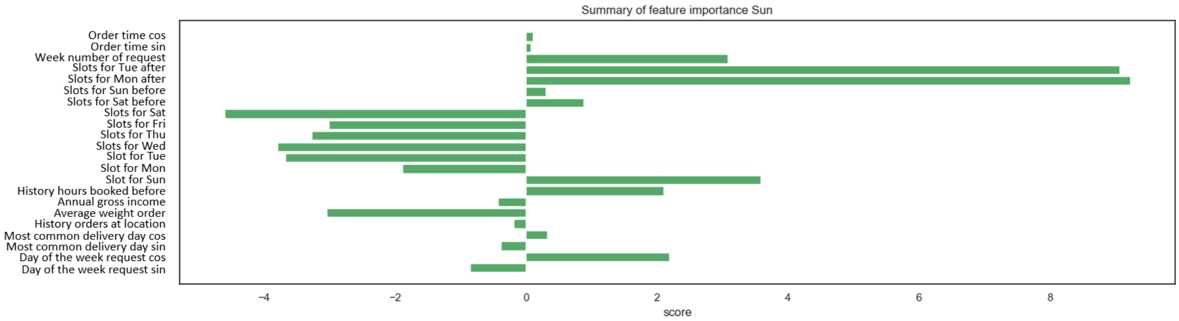


Figure G.16: Feature importance for predicted deliveries on Sunday.

For predicting that the delivery is taking place on Sunday, the most important features are instead of the number of available days for Sunday, the number of available slots for Monday and Tuesday after the selected delivery week. In addition to these two important features, the number of slots offered for all other days has a negative influence.

Feature importance for predicting that the requests results in a Lost order

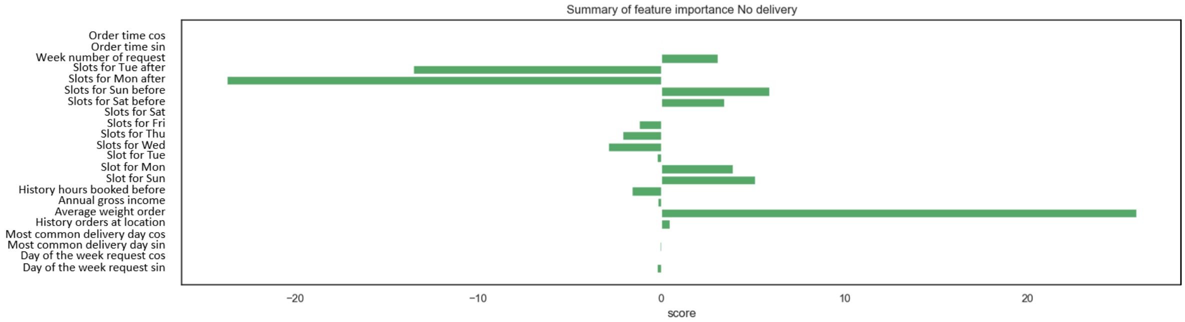


Figure G.17: Feature importance for predicted when the order is assumed to be lost

In contrast to when the order is not predicted as lost and the delivery days are predicted for this outcome, the average weight order is the most important feature and has a positive influence. Other important features to predict if the order is lost are the number of slots offered for Monday and Tuesday that negatively impact if the order is lost.

G.1.3. Part

Feature importance for predicted deliveries in the early morning

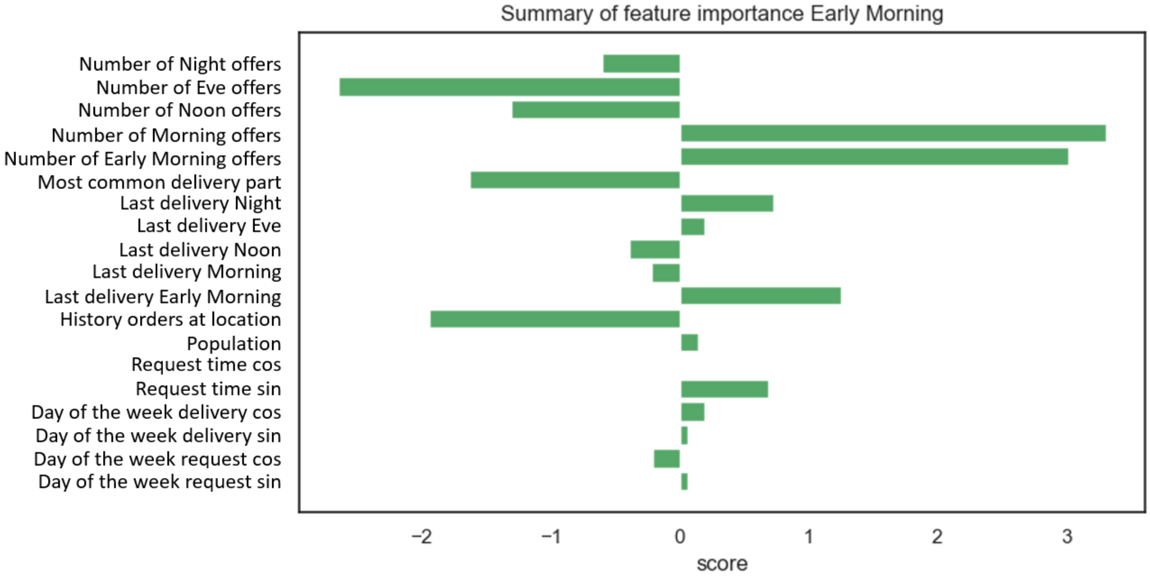


Figure G.18: Feature importance for predicted part of the day early morning

When predicting the part of the delivery day, new features are added, and the most important features for predicting if the order will be delivered in the early morning are the number of offered slots for the early morning and morning time window. Other important features are the number of orders already made by the customer and the number of offered night slots, which both have a negative influence. Also, the time the last order was delivered helps predict the new delivery moment.

Feature importance for predicted deliveries in the morning

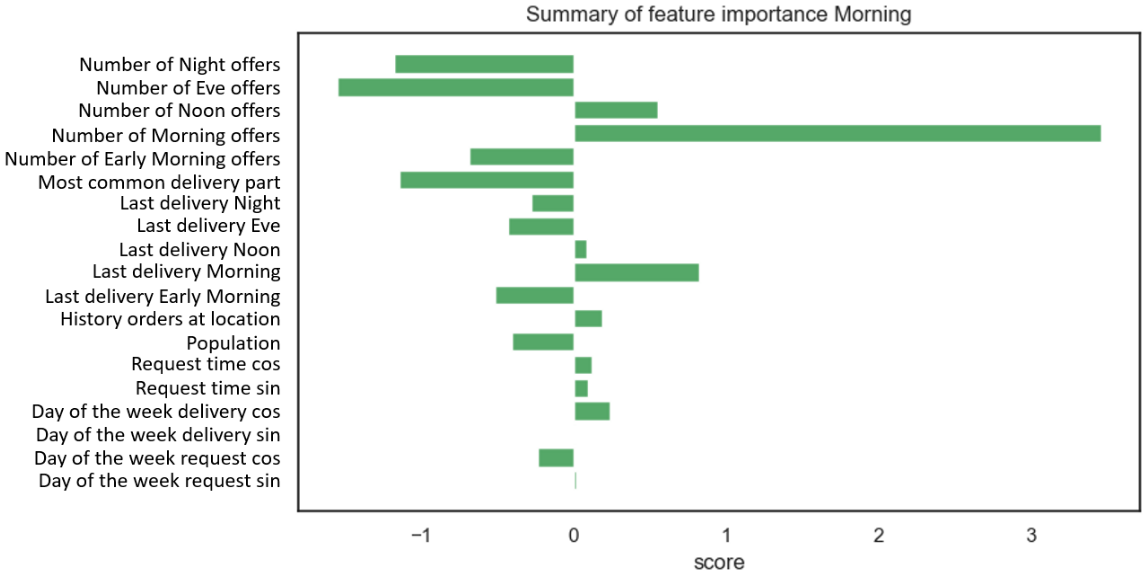


Figure G.19: Feature importance for predicted part of the day morning

For predicting if the delivery will occur in the morning, the feature indicating the number of morning offers is the most important. Other features are less important but can negatively influence the number of night and eve offers.

Feature importance for predicted deliveries in the noon

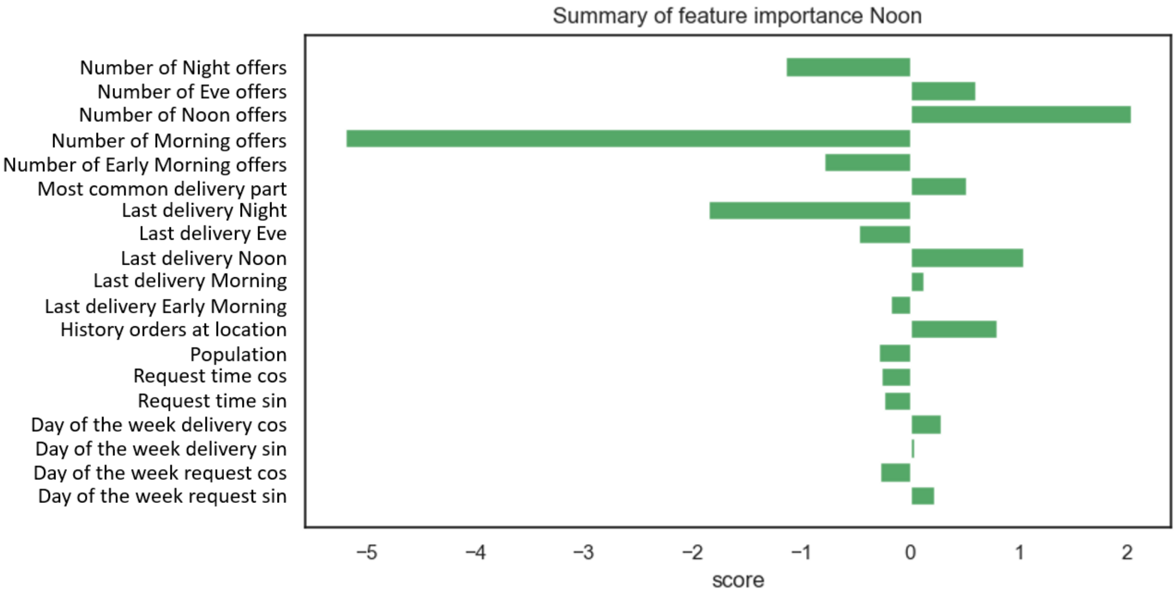


Figure G.20: Feature importance for predicted part of the day noon

The most important feature for predicting if the delivery will take place in the noon is not the number of offered slots for this delivery moment but the number of offered morning slots instead. This feature, the number of offered morning slots, negatively impacts the prediction. The second most important feature is the number of offered slots for delivery in the noon, which has a positive impact.

Feature importance for predicted deliveries in the eve

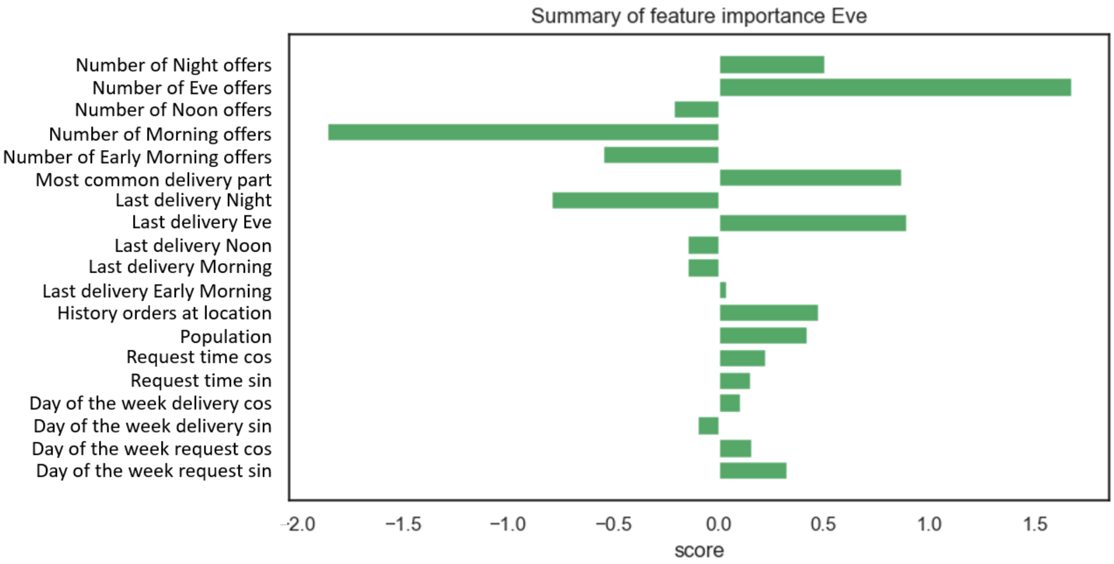


Figure G.21: Feature importance for predicted part of the day eve

Based on the feature plot for predicting if the delivery is taking place in the eve, it can be found that the feature that indicates the number of morning slots has the highest negative importance. However, the second most important feature has a positive influence and represents the number of offered eve slots.

Feature importance for predicted deliveries in the night

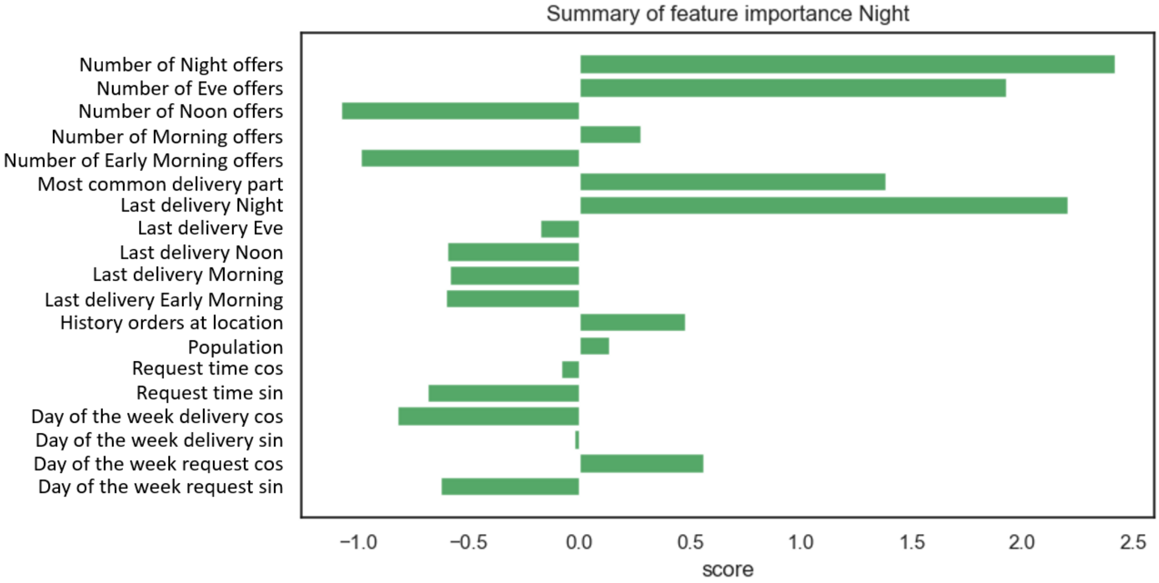


Figure G.22: Feature importance for predicted part of the day night

When looking into which features are important for predicting if the delivery is taking place in the night, it can be found that the number of offered night and eve offers and features based on the history have a strong influence. The features based on history indicate, for example, that the last delivery was delivered during the night.

G.2. Feature importance NN

To give more insights into which features are important in the NN, SHAP values are used to identify the influence of the used features. For indicating the influence of a feature, the SHAP calculates the impact of every feature on the target variable, the process is described in more detail in Appendix F. The arrangement of features in SHAP value plots is based on their level of influence, with the most influential feature placed at the top. These plots can also convey whether a feature has a positive or negative effect when the value is high or low. Positive effects are depicted on the right side of the y-axis, while negative effects are illustrated on the left. Additionally, the colour of the feature value indicates its magnitude, with warm colours (e.g. red) indicating high values and cool colours (e.g. blue) representing low values.

G.2.1. Week

Feature importance for predicted deliveries in week 44

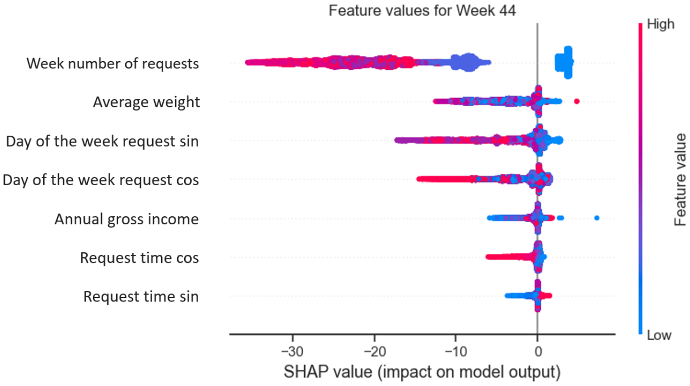


Figure G.23: SHAP values indicating the feature importance for deliveries predicted in week 44

Feature importance for predicted deliveries in week 45

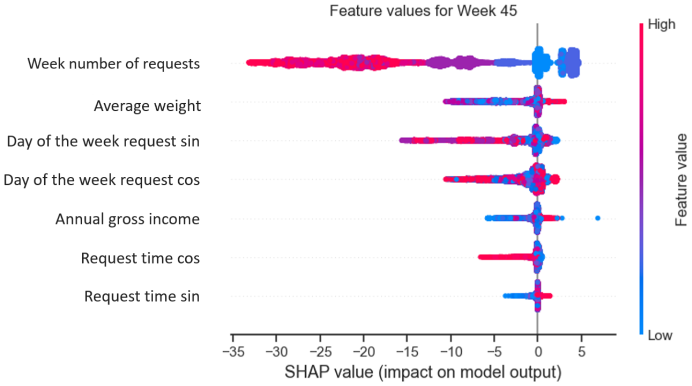


Figure G.24: SHAP values indicating the feature importance for deliveries predicted in week 45

Feature importance for predicted deliveries in week 46

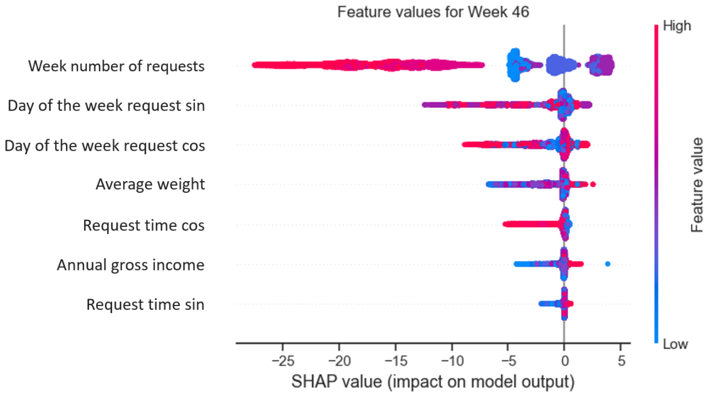


Figure G.25: SHAP values indicating the feature importance for deliveries predicted in week 46

Feature importance for predicted deliveries in week 47

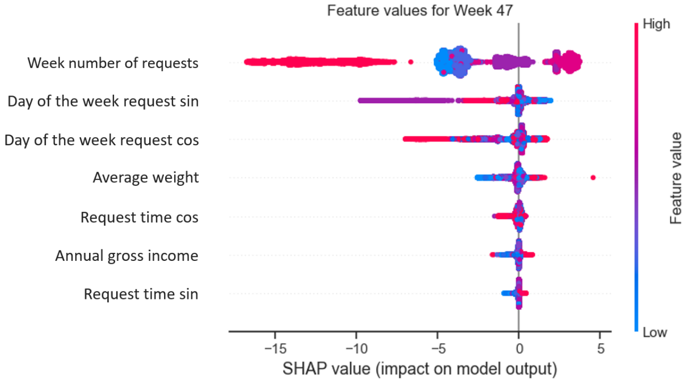


Figure G.26: SHAP values indicating the feature importance for deliveries predicted in week 47

Feature importance for predicted deliveries in week 48

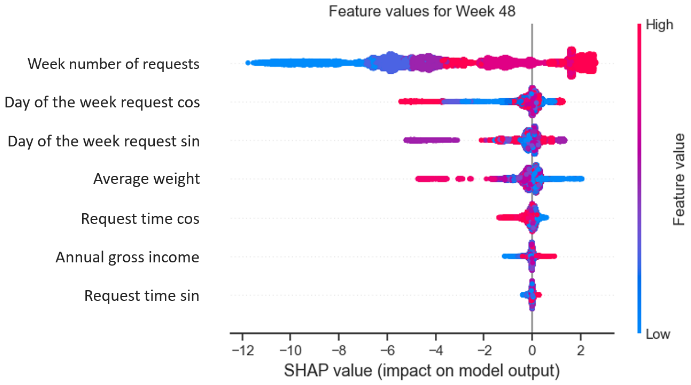


Figure G.27: SHAP values indicating the feature importance for deliveries predicted in week 48

Feature importance for predicted deliveries in week 49

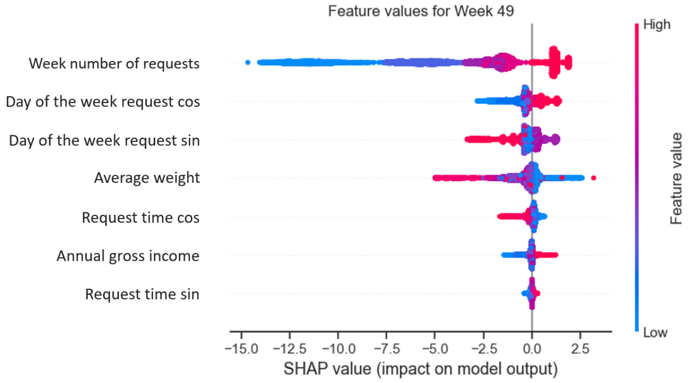


Figure G.28: SHAP values indicating the feature importance for deliveries predicted in week 49

Feature importance for predicted deliveries in week 50

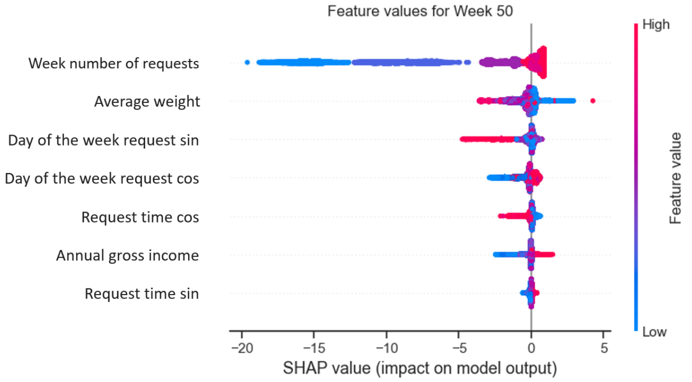


Figure G.29: SHAP values indicating the feature importance for deliveries predicted in week 50

Feature importance for predicted deliveries in week 51

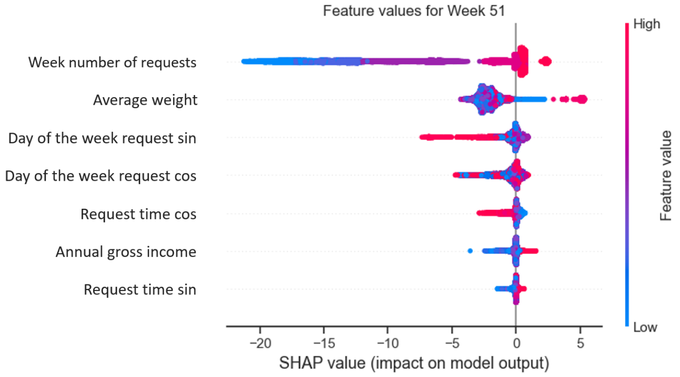


Figure G.30: SHAP values indicating the feature importance for deliveries predicted in week 51

Feature importance for predicted deliveries in week 52

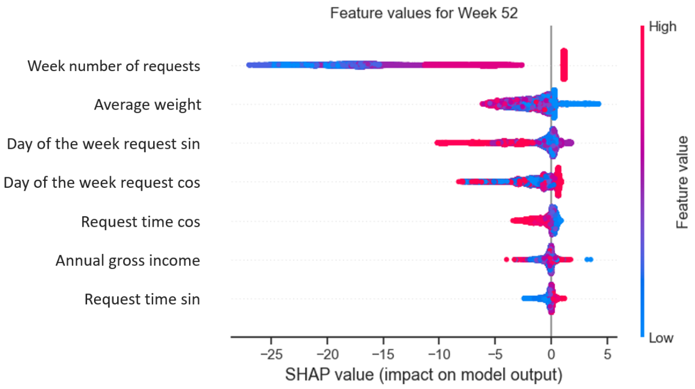


Figure G.31: SHAP values indicating the feature importance for deliveries predicted in week 52

G.2.2. Day

Feature importance for predicted deliveries on Monday

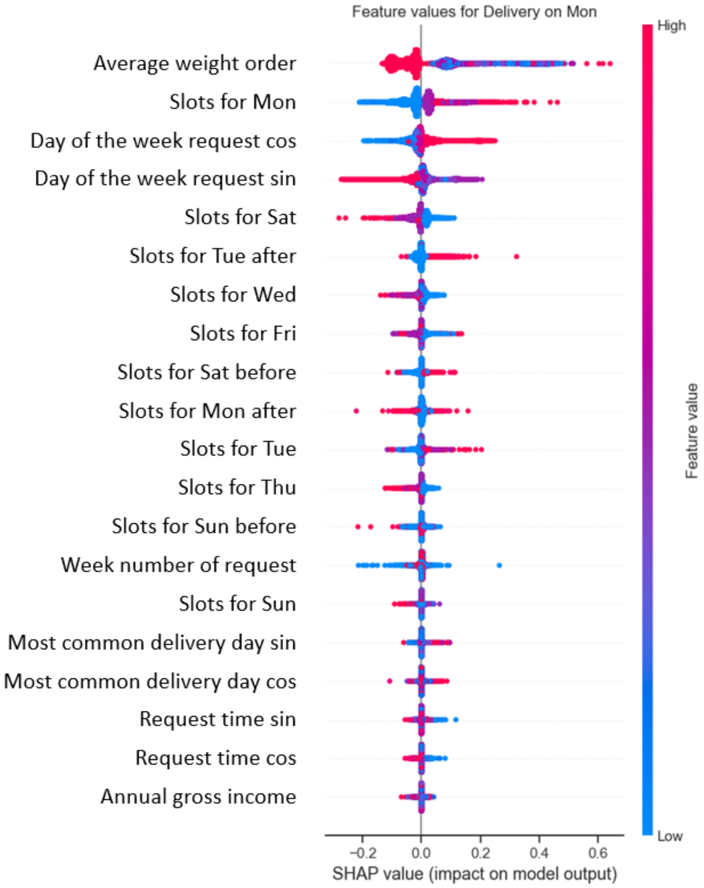


Figure G.32: SHAP values indicating the feature importance for deliveries predicted for Monday

Feature importance for predicted deliveries on Tuesday

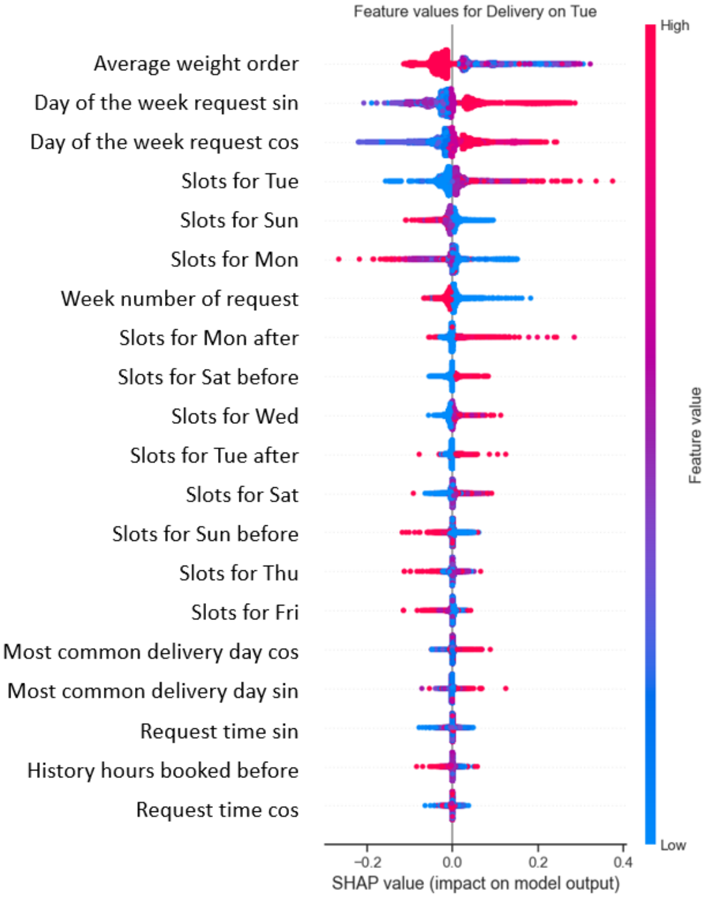


Figure G.33: SHAP values indicating the feature importance for deliveries predicted for Tuesday

Feature importance for predicted deliveries on Wesnesday

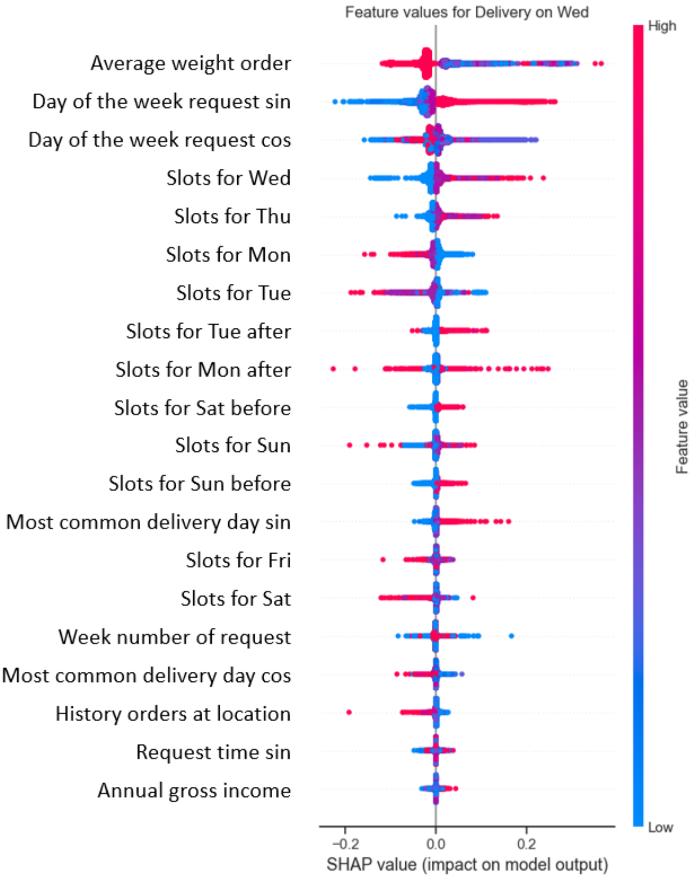


Figure G.34: SHAP values indicating the feature importance for deliveries predicted for Wednesday

Feature importance for predicted deliveries on Thursday

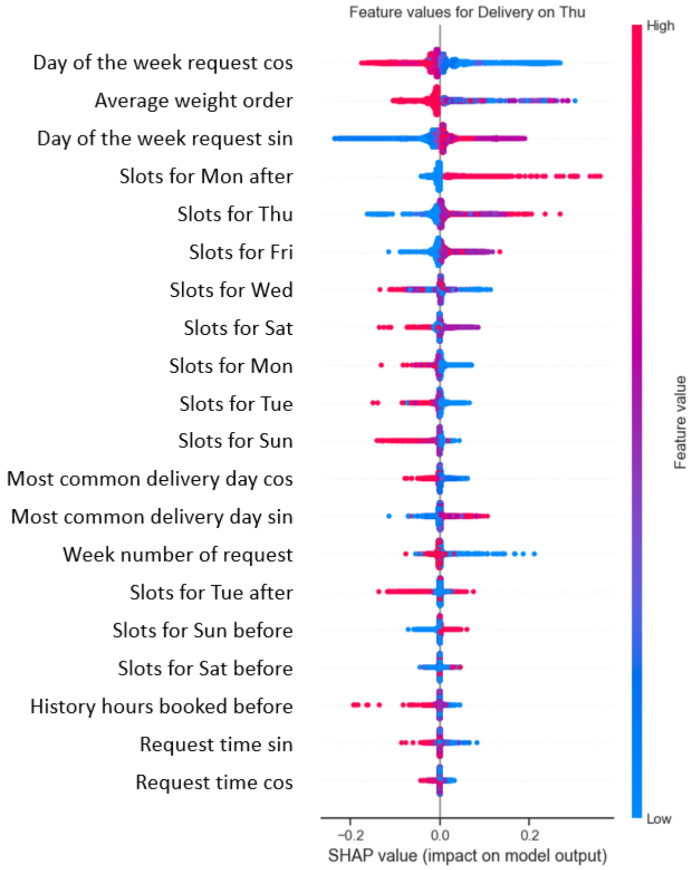


Figure G.35: SHAP values indicating the feature importance for deliveries predicted for Thursday

Feature importance for predicted deliveries on Friday

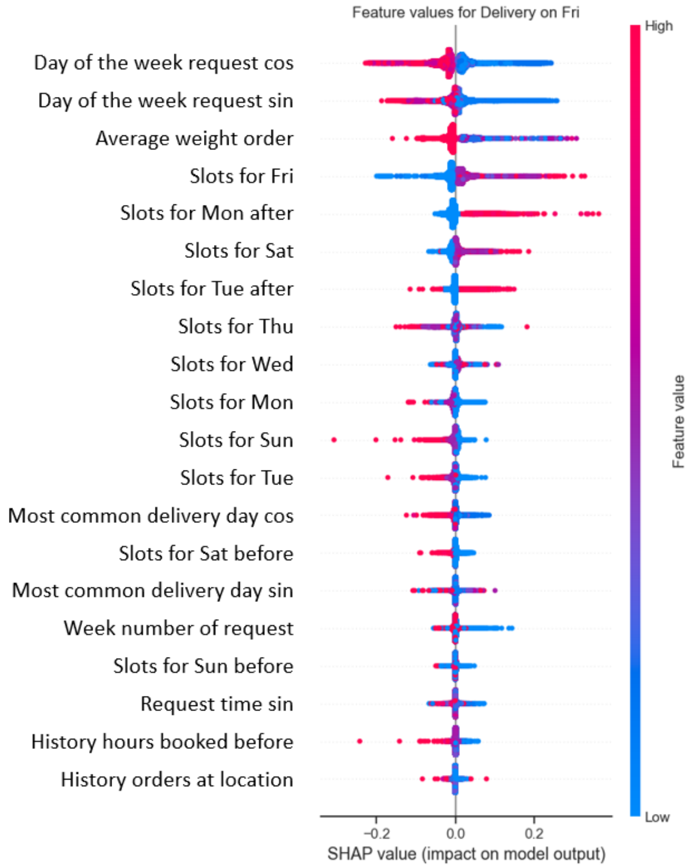


Figure G.36: SHAP values indicating the feature importance for deliveries predicted for Friday

Feature importance for predicted deliveries on Saturday

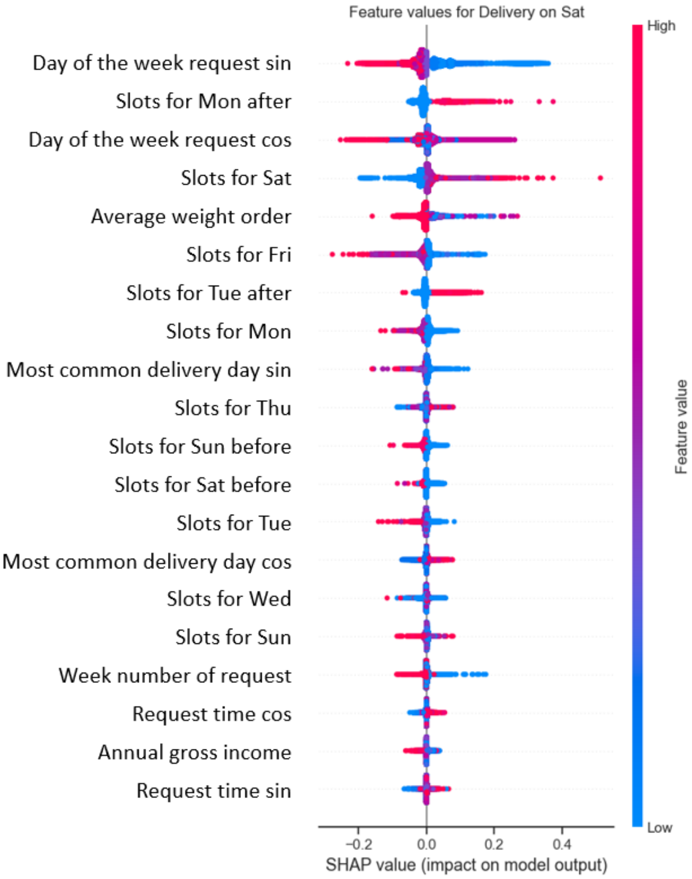


Figure G.37: SHAP values indicating the feature importance for deliveries predicted for Saturday

Feature importance for predicted deliveries on Sunday

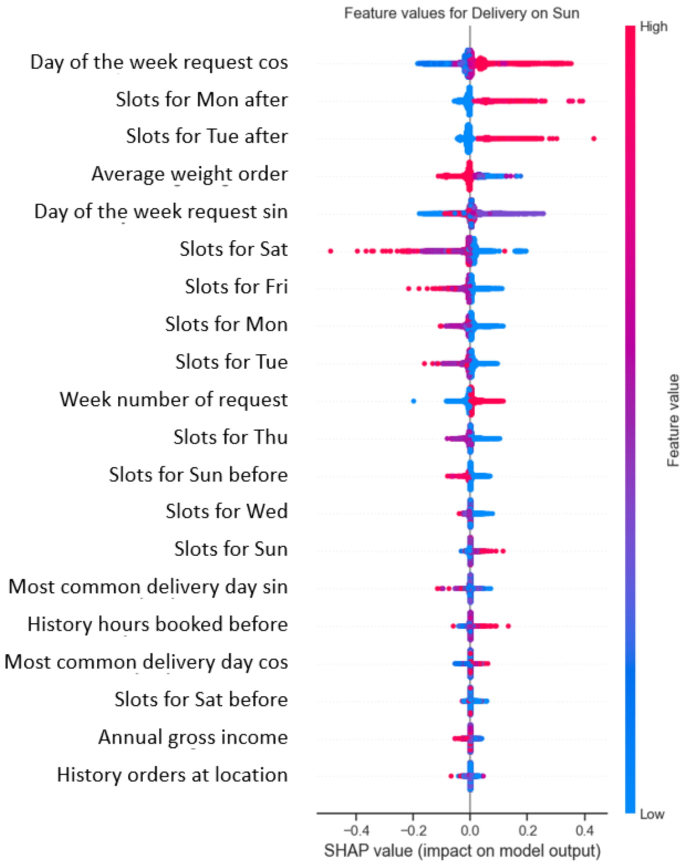


Figure G.38: SHAP values indicating the feature importance for deliveries predicted for Sunday

Feature importance for predicting that the requests results in a Lost order

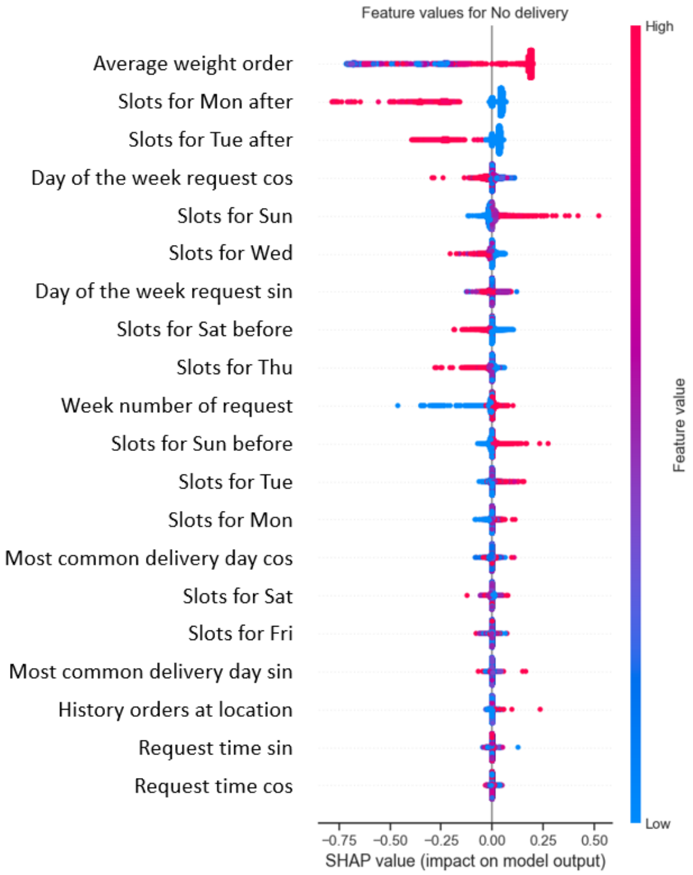


Figure G.39: SHAP values indicating the feature importance for deliveries predicted as Lost

G.2.3. Part of the day

Feature importance for predicted deliveries in the early morning

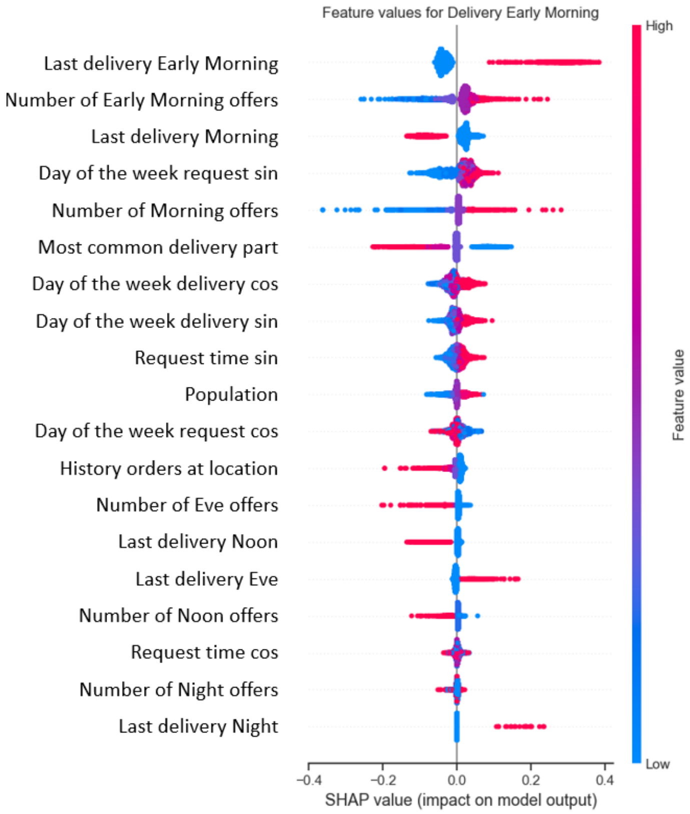


Figure G.40: SHAP values indicating the feature importance for deliveries predicted in the Early Morning

Feature importance for predicted deliveries in the morning

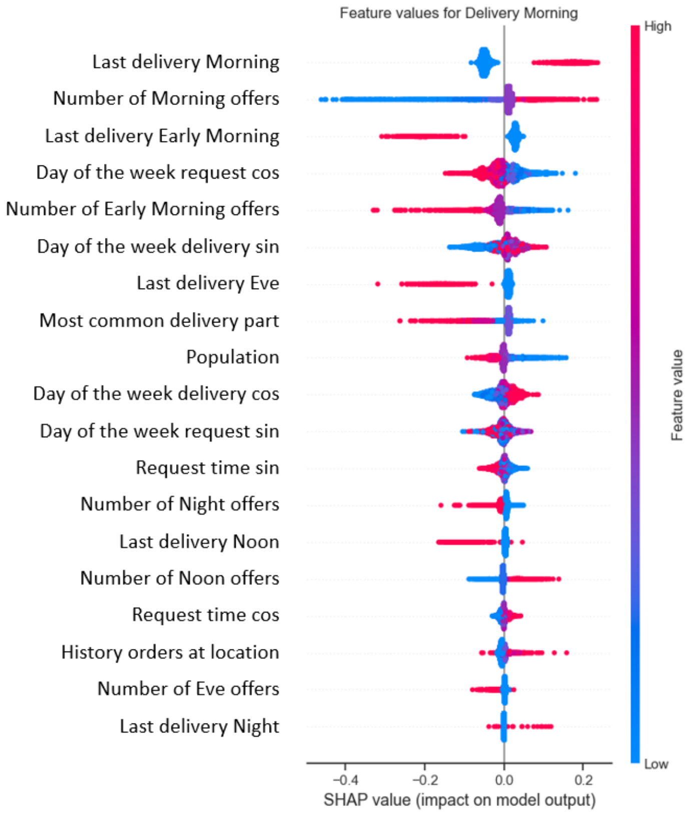


Figure G.41: SHAP values indicating the feature importance for deliveries predicted in the Morning

Feature importance for predicted deliveries in the noon

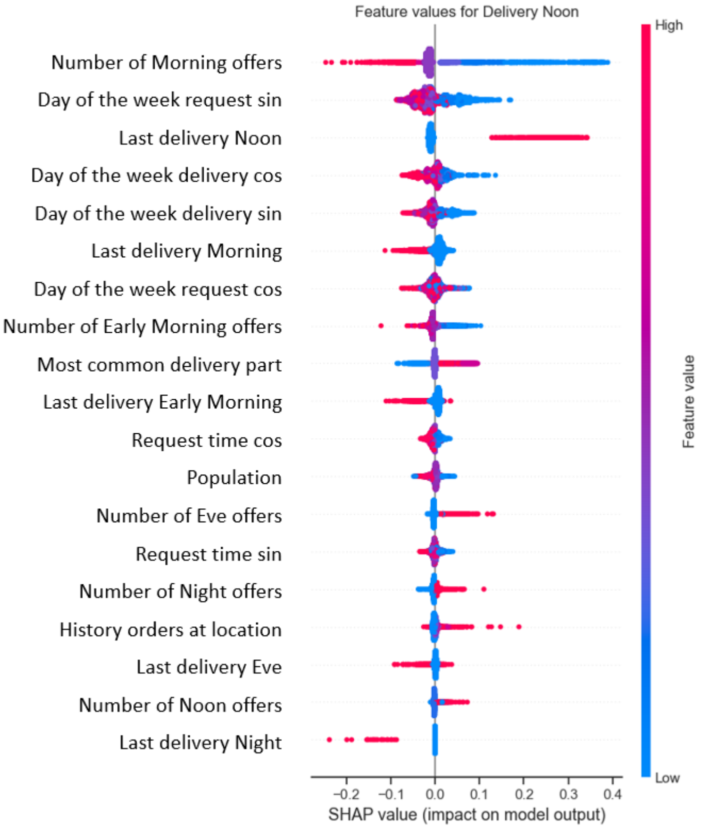


Figure G.42: SHAP values indicating the feature importance for deliveries predicted in the Noon

Feature importance for predicted deliveries in the eve

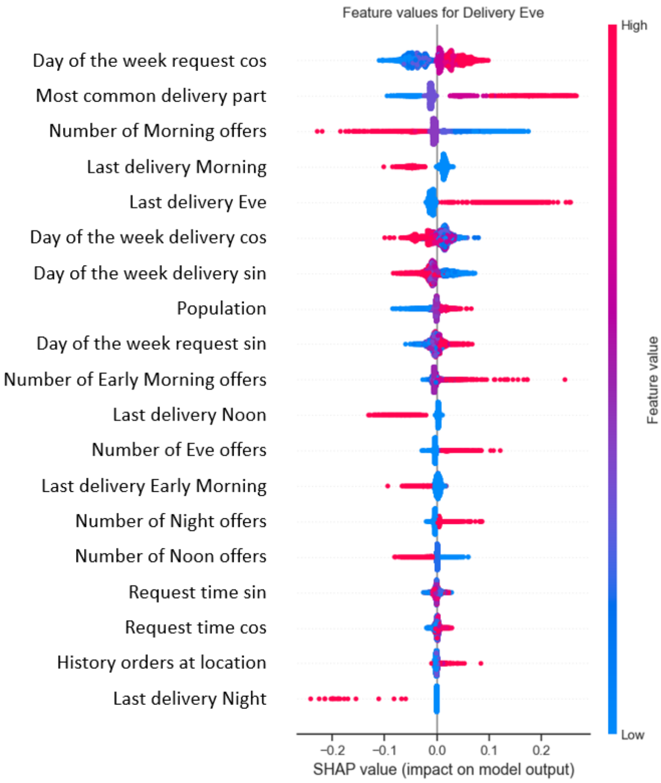


Figure G.43: SHAP values indicating the feature importance for deliveries predicted in the Eve

Feature importance for predicted deliveries in the night

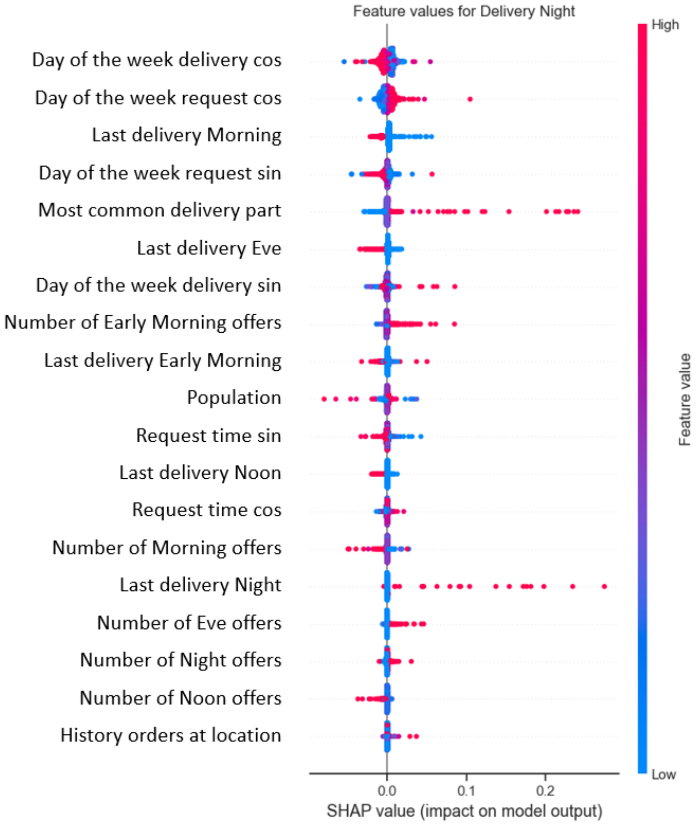


Figure G.44: SHAP values indicating the feature importance for deliveries predicted in the Night

H

Segmentation

The executed advanced models employ segmentation by adding clusters as a feature. These clusters are derived from customer demographic and order history characteristics and can offer valuable insights into customer behaviour and preferences. However, during the prediction of the part of the day, it is observed that historical features are deemed to be of greater importance. Nevertheless, it might be possible that this form of customer segmentation alone may have limited utility as multiple variables and characteristics beyond the features representing clusters are involved. Moreover, since the data used in this research only covered November, historical information on most customers was not available. As a result, the MNL and NN models might have down-weighted the importance because they often assign weights to different features based on their ability to predict the outcome of interest and, therefore, rely more heavily on other features consistently available across all customers. Combined with the desire to enhance the prediction performance, it is decided to explore whether incorporating more historical information could improve the models' predictive capabilities. To do this, only customers' behaviour is predicted that ordered before. In this way, it is also possible to determine whether it is advantageous to store more customer details to perform predictions in the future, as this requires more space.

For this reason, the Benchmark, MNL and NN model will be determined and trained with only data from customers that ordered before. Since historical information is not available for predicting the week, this step will be omitted, and only the two other prediction steps will be performed per model. Due to time limitations, the model architecture for all these models will remain unchanged since hyperparameter tuning takes a long time. However, the results of the Benchmark model predicting the delivery day and part of the day can be found in Figure H.1. Following the presentation of the results for the Benchmark model, the outcomes of the MNL and NN models will be shown in Figure H.2, Figure H.5, Figure H.8 and Figure H.11, respectively.

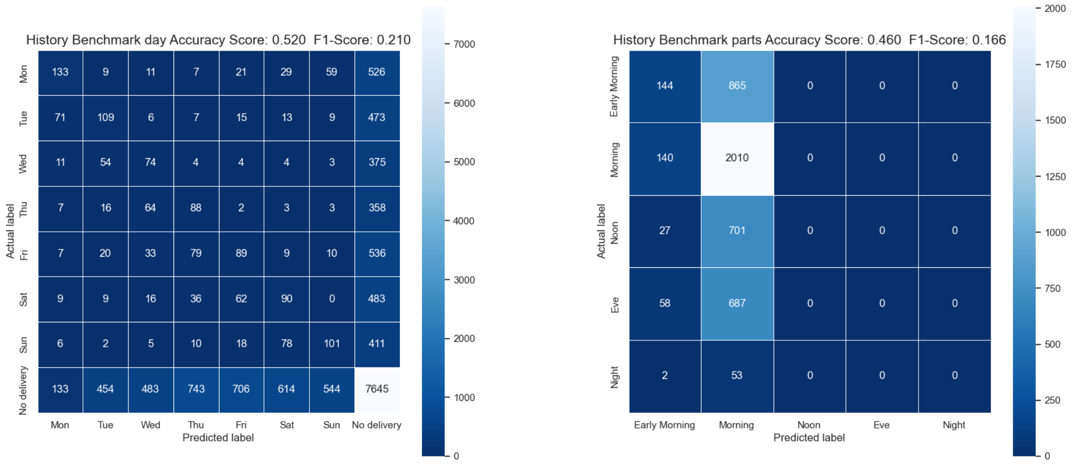


Figure H.1: The summarised outcomes of the Benchmark model are presented, which solely consider customers who have placed orders before. The results for the predicted delivery day are illustrated on the left-hand side, while the predicted part of the delivery day is displayed on the right-hand side.

After analysing the evaluation metrics of the Benchmark model, it is observed that both prediction steps' performances have improved compared to the ones developed using all available data. The F1-score for predicting the delivery day has increased from 0.123 to 0.210, while for predicting the part of the delivery day, the F1-score has improved from 0.127 to 0.166. The most striking things in these results are that for predicting the part of the delivery day, not only the Morning is predicted anymore but also the Early Morning label and for predicting the delivery day, Thursday is now predicted. Similar to the previous Benchmark models, these models will also be utilised to determine if the MNL and NN models add value to the prediction process.

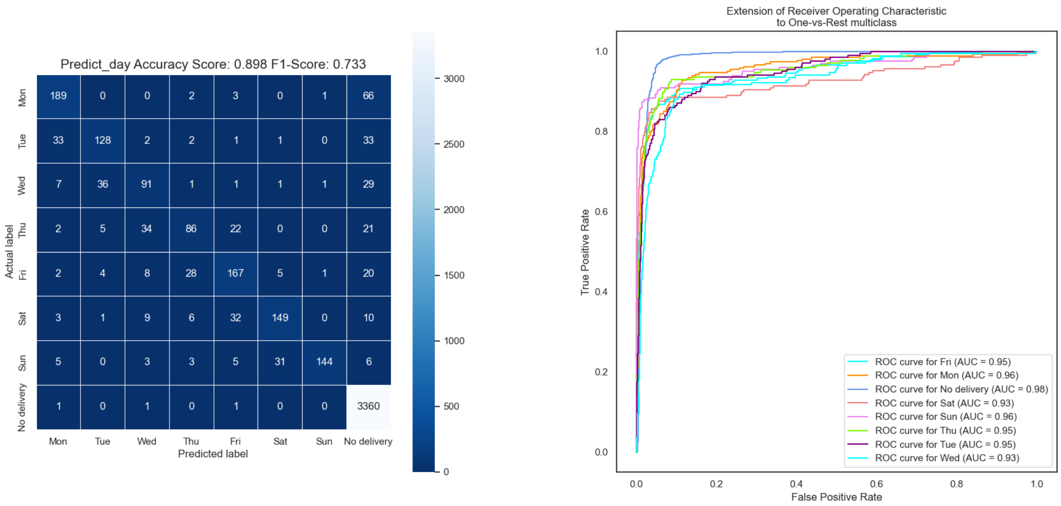


Figure H.2: On the left side, a summary of the results for predicting the delivery day can be found in the confusion matrix, and on the right side, the ROC curve is displayed. For both plots, only customer data utilised who ordered before in November.

The MNL model is the first advanced model used in this study, which predicts both the delivery day and the part of the delivery day sequentially. The performance of the MNL model for the first prediction step, which predicts whether the delivery will take place and on which day, is presented in Figure H.2. The

F1-score for this step is 0.733, and the accuracy is 0.898, indicating that 89.9% of the predictions are correct. However, an improvement is observed in the F1-score compared to the Benchmark model. On the other hand, when the MNL model using all the data is compared to the MNL model utilising only data of customers with history, no performance improvement is observed, and the ROC and AUC values remain almost the same. This suggests that the MNL model does not have an advantage when only data of customers with history are used and is even less confident in assigning the No Delivery label and all other labels, as depicted in Figure H.3.

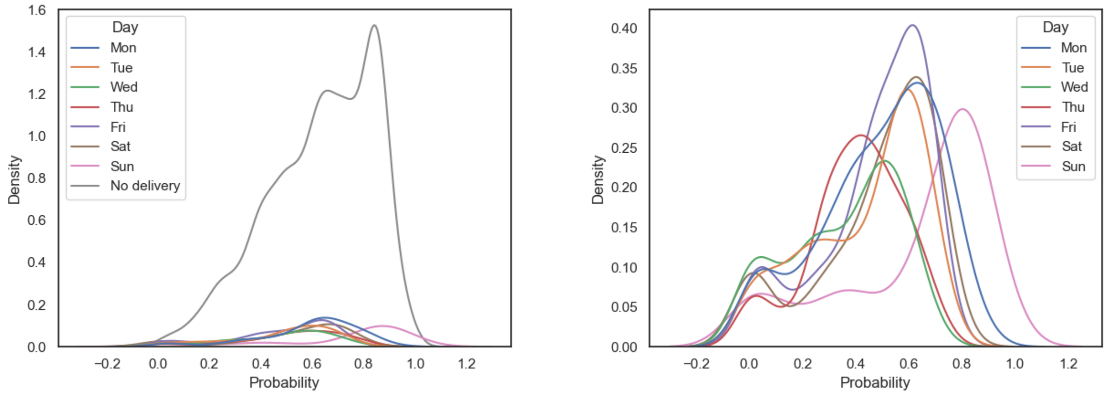


Figure H.3: Probability distribution for if the order is lost or not. When the order is not lost, the probability of the delivery day is given. The upper and lower bounds were checked and found to be 0.999 and 0.000, respectively, to check for any probabilities that are not feasible.

To further analyse why the MNL models have almost the same prediction performance, feature values can provide more insights into the model's behaviour. The summary plot of all classes containing the absolute coefficient value is displayed in Figure H.4. Based on this plot and the more detailed feature importance plots presented in Appendix I for all separate classes, it can be found that the importance distribution is changed. In order words, features became comparatively less or more important. Nevertheless, it is indicated that the features used for predicting whether a customer placed an order and, if so, which day do not contain much historical information. Of the features used, only the *History hours booked before* and the *Most common delivery* contain historical information. On this information, a possible explanation for why the performance does not change when more focus is placed on historical information could be improved when adding new features. Fore, a possible explanation for why the performance does not change when more emphasis is placed on historical information is the lack of these features, which could be improved by adding new/more historical features.

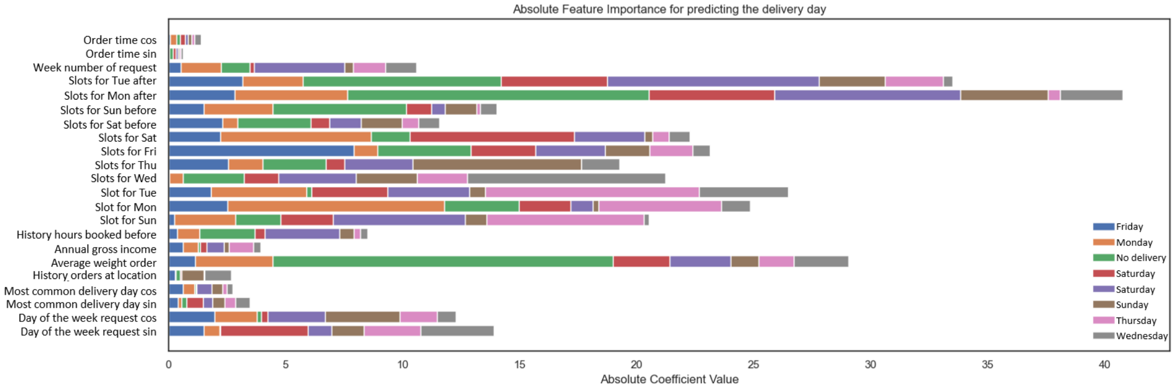


Figure H.4: Summary of feature importance using the MNL model for predicting the delivery day.

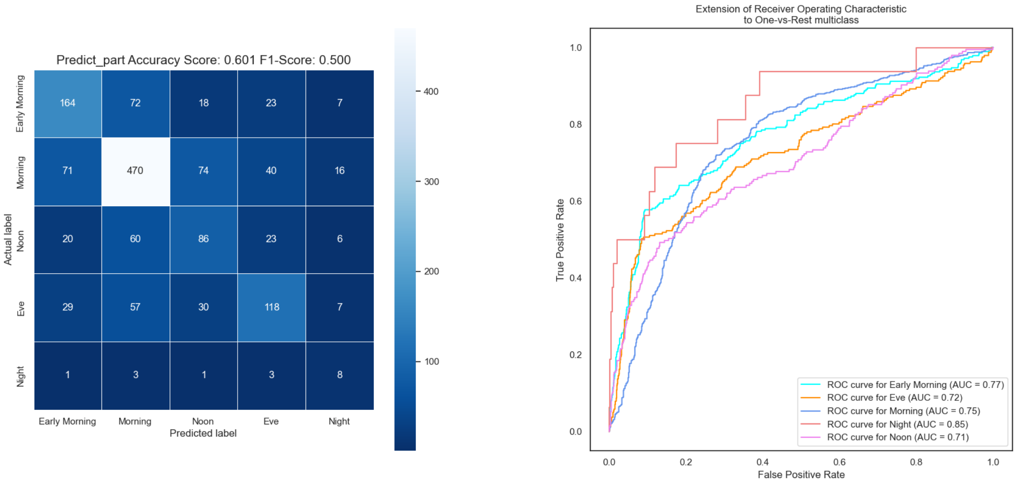


Figure H.5: On the left side, a summary of the results for predicting the part of the delivery day can be found in the confusion matrix, and on the right side, the ROC curve is displayed. For both plots, only customer data utilised who ordered before in November.

The second step in the prediction process involves using the MNL model to determine the part of the delivery day, and the results are presented in Figure H.5. The confusion matrix indicates that all labels are predicted, and the accuracy is 0.601, implying that 60.1% of the predicted data is correct. When comparing the F1-score of this model with the Benchmark model, an improvement can be observed as the score increased to 0.500. Moreover, when these results are compared to the MNL model utilising all data, a performance increase can be found. When looking into more detail, it can be found in the ROC plot that the model better distinguishes the output class, especially the Night category. Furthermore, by examining the probability density plot depicted in Figure H.6, it can be observed that additional peaks appear on the right side of the distribution when using 0.5 as a threshold. This suggests that the MNL model is becoming more confident in assigning the correct labels and explains its improved ability to distinguish between different output classes.

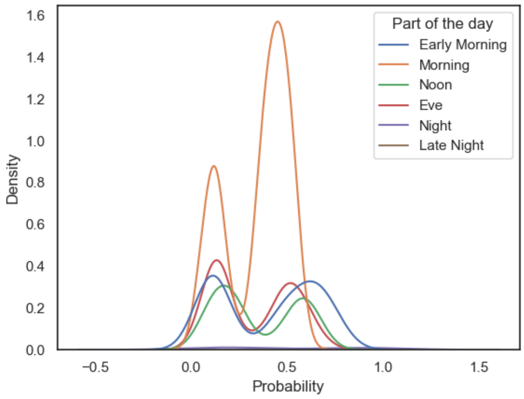


Figure H.6: Probability distribution for the different parts of the day as output classes predicted with only customers that ordered earlier. The lower and upper bounds were checked and found to be 0.006 and 0.964, respectively, to check for any probabilities that are not feasible.

To gain a better understanding of the MNL model's performance, feature importance plots can be employed. A summary plot of all classes can be observed in Figure H.7. This plot reveals that certain features, such as the *Most common delivery part*, are considered more critical than the MNL model predicting the delivery based on all data. Furthermore, the MNL model may be improving relative to

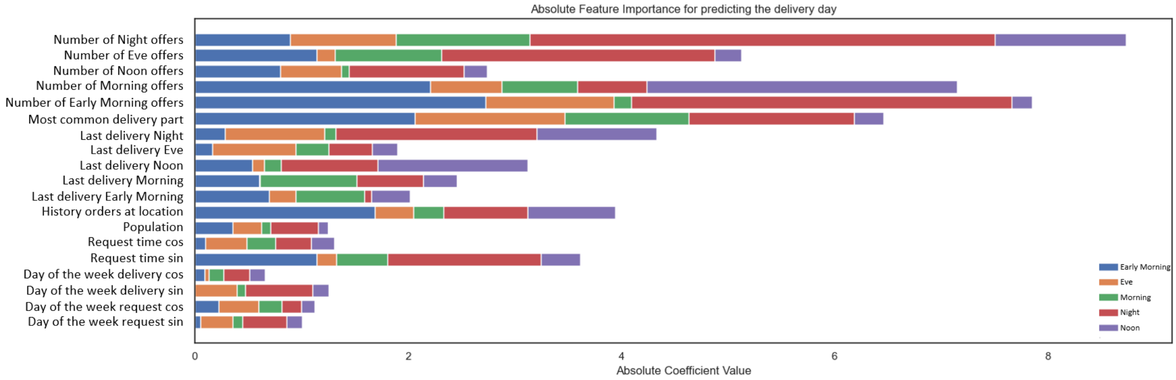
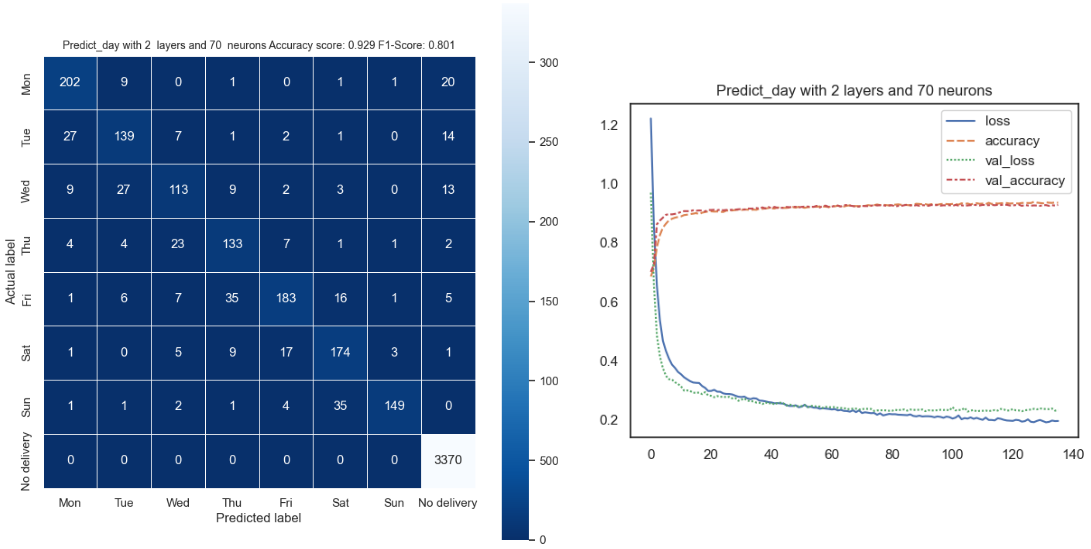


Figure H.7: Summary of feature importance using the MNL model for predicting the part of the delivery day.

the other MNL model since more features with historical information have been included, which was not the case when comparing the delivery day. The plots visualised in Appendix I can provide a more detailed overview of the used features and their importance.



	Mon	Tue	Wed	Thu	Fri	Sat	Sun	No delivery
Mon	202	9	0	1	0	1	1	20
Tue	27	139	7	1	2	1	0	14
Wed	9	27	113	9	2	3	0	13
Thu	4	4	23	133	7	1	1	2
Fri	1	6	7	35	183	16	1	5
Sat	1	0	5	9	17	174	3	1
Sun	1	1	2	1	4	35	149	0
No delivery	0	0	0	0	0	0	0	3370

Figure H.8: On the left side, a summary of the results for predicting the delivery day with only customers that ordered earlier. On the right side, the learning curve of the NN model for predicting the delivery day can be found with on the x-axis the number of utilised epochs during the training process.

Following the evaluation of the MNL models for predicting the delivery day and part of the day using only historical customer data, a similar analysis was performed for the NN models. Starting with the NN model predicting the delivery day, the confusion matrix displayed on the left side of Figure H.8 indicates an accuracy of 0.929, suggesting that 92.2% of the data is predicted correctly. To compare its performance with the Benchmark, MNL, and NN models using all customer data, the F1-score was determined and found to be 0.801. This result suggests that this NN model outperforms the other three models. To minimise the likelihood of overfitting during training, this model contains similar to the earlier NN models, a dropout rate of 20%, an early stopping method and a plot to analyse the learning performance of the model on the right side of Figure H.8.

In order to further investigate the performance of the model, the probability density plot displayed in Figure H.9 can be utilised. The plot shows that the peaks of the different output classes have moved slightly to the right, indicating that the model has become more certain about the predictions, which also explains the better prediction performance.

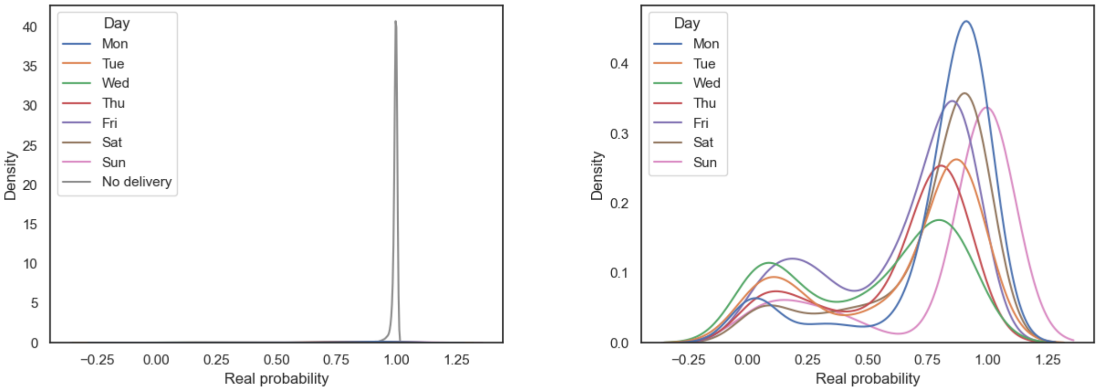


Figure H.9: Probability distribution for if the order is lost or not. When the order is not lost, the probability of the delivery day is given. The upper and lower bounds were checked and found to be 1.000 and 0.000, respectively, to check for any probabilities that are not feasible.

To better understand why the NN model performs slightly better than the model using all customer data, even when only a few historical features are included, a summary plot of SHAP values is presented in Figure H.10. The SHAP values represent the contribution of each feature to the prediction for each sample and thus provide insight into the model's behaviour. The plot shows that the feature importance order is mainly unchanged between the two models, which could explain why their prediction performance is almost similar. However, a more detailed view of the feature importance can be obtained by examining the category-specific feature importance plots shown in Figure I. Overall, the SHAP value analysis provides valuable insights into the behaviour of the NN model and highlights the importance of certain features for accurate predictions.

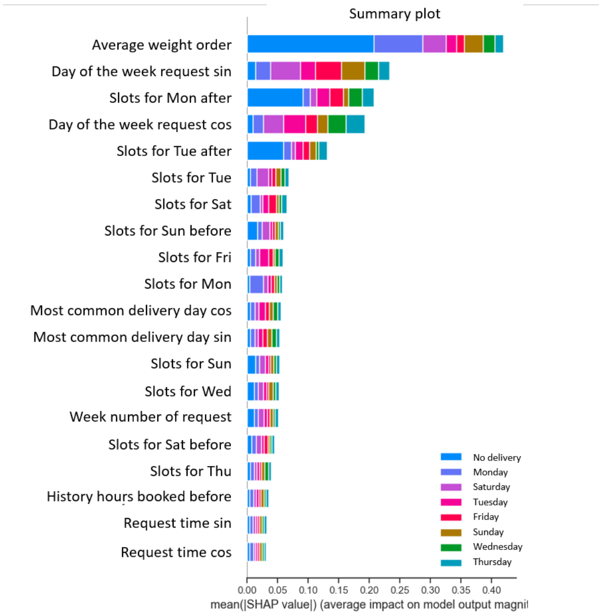


Figure H.10: Summary of SHAP values indicating the feature importance for all weekdays and the option that the order is lost.

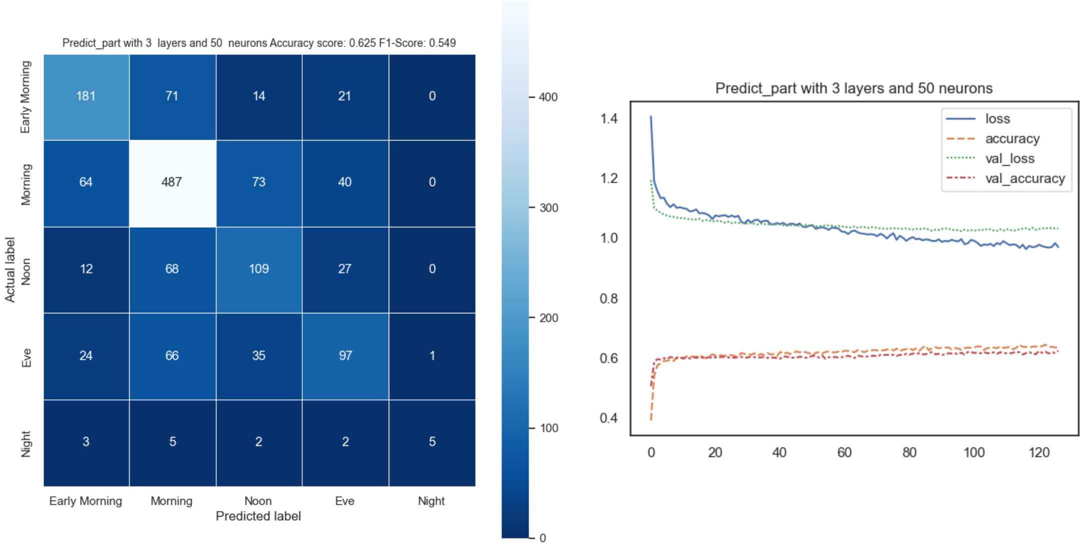


Figure H.11: On the left side, a summary of the results for predicting the part of the delivery day with only customers that ordered earlier. On the right side, the learning curve of the NN model for predicting the part of the delivery day can be found with on the x-axis the number of utilised epochs during the training process.

The final model to evaluate the effect of using data only from customers who have made an order previously is the NN model for predicting the part of the delivery day. The model's performance is summarised in the confusion matrix shown on the right side of Figure H.11. The accuracy value of 0.625 indicates that 62.5% of the data is predicted correctly. However, to compare the model's performance with the Benchmark, MNL, and NN models trained on all the data, the F1-score is used, which has a value of 0.549. This indicates that the NN model for predicting the part of the delivery day performs better than the other three models. To further analyse the performance, the learning performance is visualised on the right side of Figure H.11. It can be observed that the accuracy increases, and the losses decrease more rapidly than in the other NN model. Nevertheless, the model is stopped before the learning process ends to avoid overfitting. Additionally, based on the probability density plots displayed in Figure H.12, the model becomes more confident in predicting the categories compared to the NN model that uses all the data.

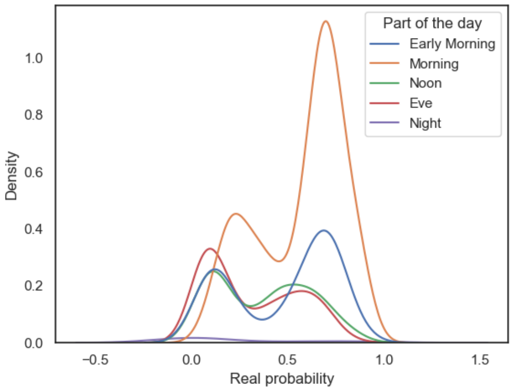


Figure H.12: Probability distribution for the different parts of the day as output classes predicted with only customers that ordered earlier. The lower and upper bounds were checked and found to be 0.001 and 0.951, respectively, to check for any probabilities that are not feasible.

The feature importance is also determined using SHAP values for the final prediction step. The summarised SHAP value plot is displayed in Figure H.13, giving an overview of the prediction of all different classes. When comparing this plot with the summarised feature importance plot of the NN model based on all customer data, it can be found that especially the features containing historical information are indicated as more important. Features considered more important incorporate, for example, information about the most common and last delivery moments. A more detailed overview of the feature importance per output category can be found in Figure I.

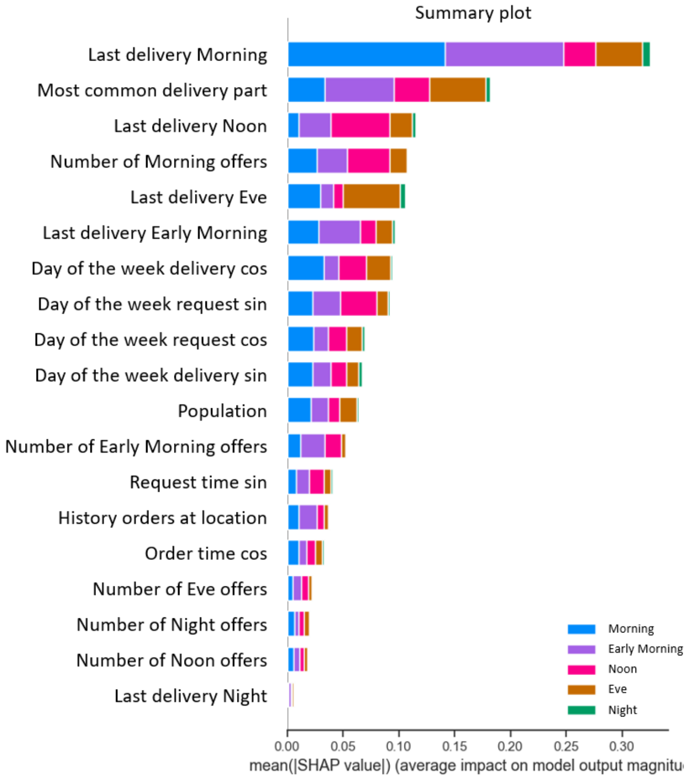


Figure H.13: Summary of SHAP values indicating the feature importance for all parts of the day.

To summarise, the analysis results show that both the MNL and NN models outperform the Benchmark model, indicating that they provide better predictions and add value to the process by improving the performance. These findings suggest that the models are better capable of capturing customer behaviour and more accurate than the Benchmark model. Furthermore, when comparing the performance of the MNL and NN models, it is observed that both models that predict the part of the delivery day outperform the other MNL and NN models. However, when comparing the results for predicting the delivery day, the performance is similar, allowing for the assumption that they perform equally well. One plausible reason why the models predicting the part of the delivery day demonstrate better performance could be the result of the inclusion of additional historical information features. Only a few used features contain that information for the step predicting the day. Hyperparameter tuning and additional new historical features may further enhance the performance of the models. These results demonstrate the potential of using historical data in the MNL and NN models to improve prediction performance, especially as more historical data becomes available.

Feature importance of the clustered MNL and NN

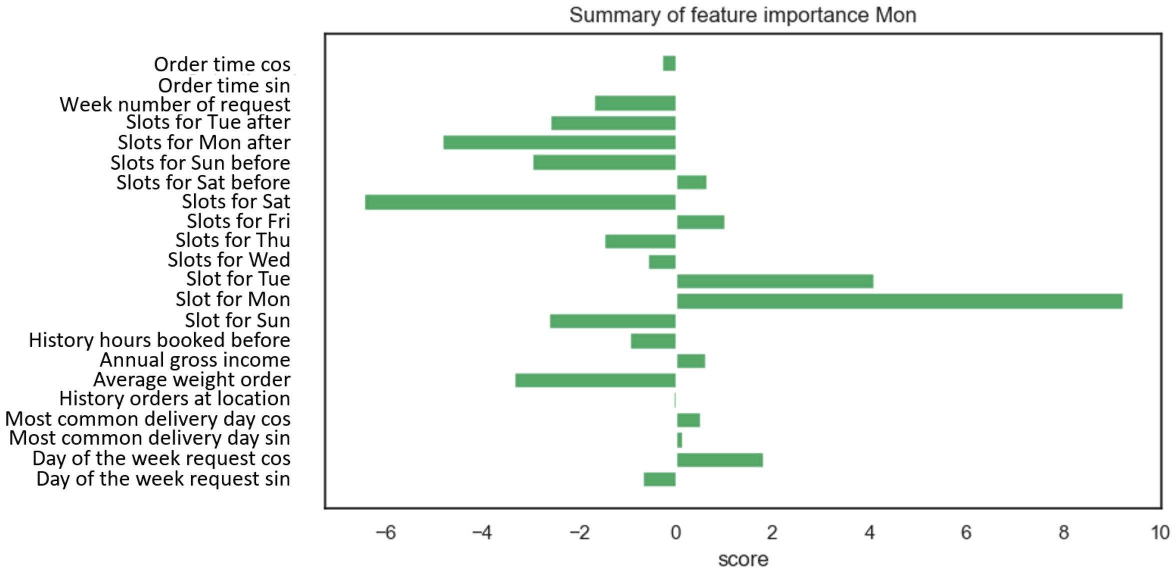


Figure I.1: Feature importance for predicted deliveries on Monday.

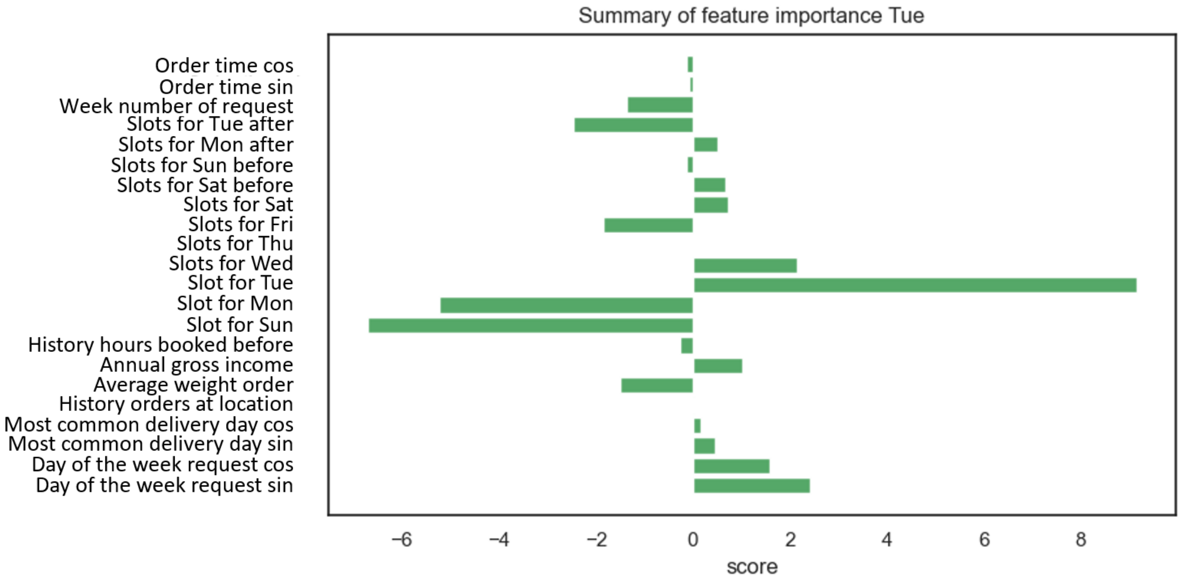


Figure I.2: Feature importance for predicted deliveries on Tuesday.

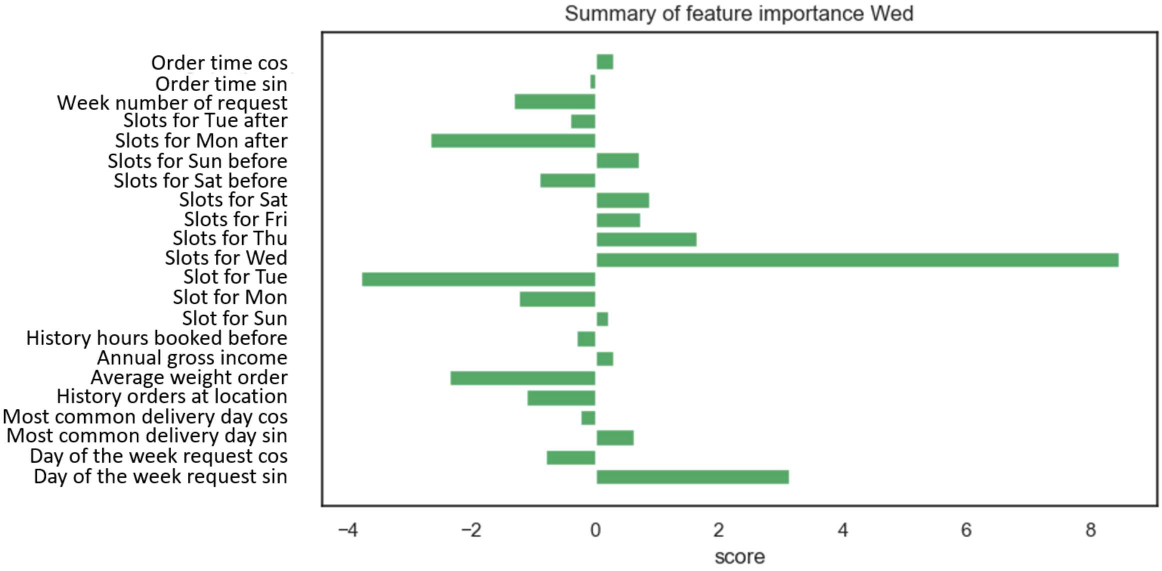


Figure I.3: Feature importance for predicted deliveries on Wednesday.

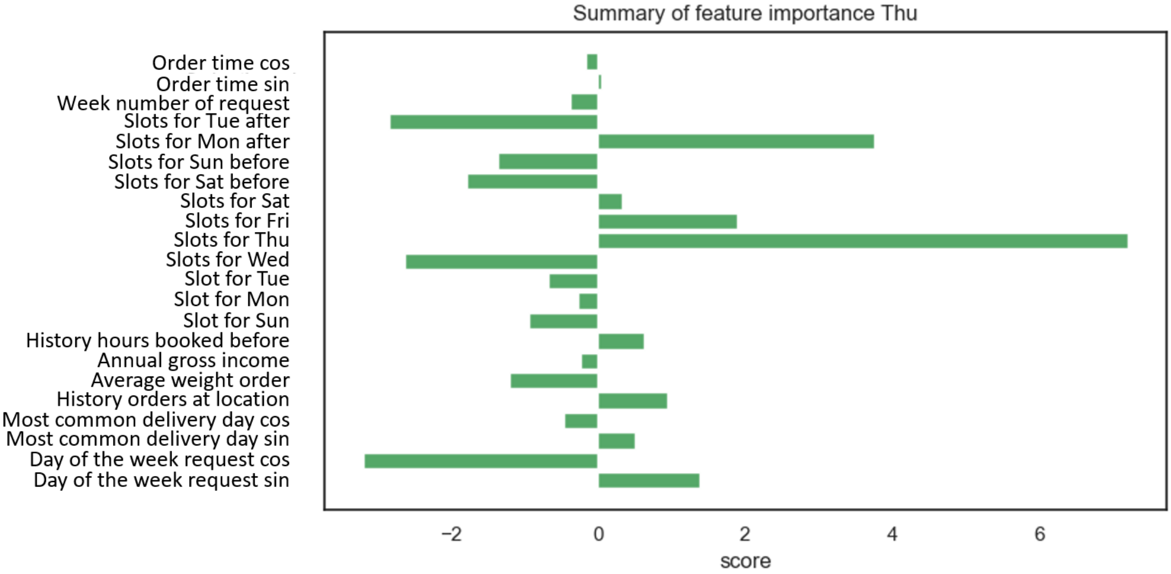


Figure I.4: Feature importance for predicted deliveries on Thursday.

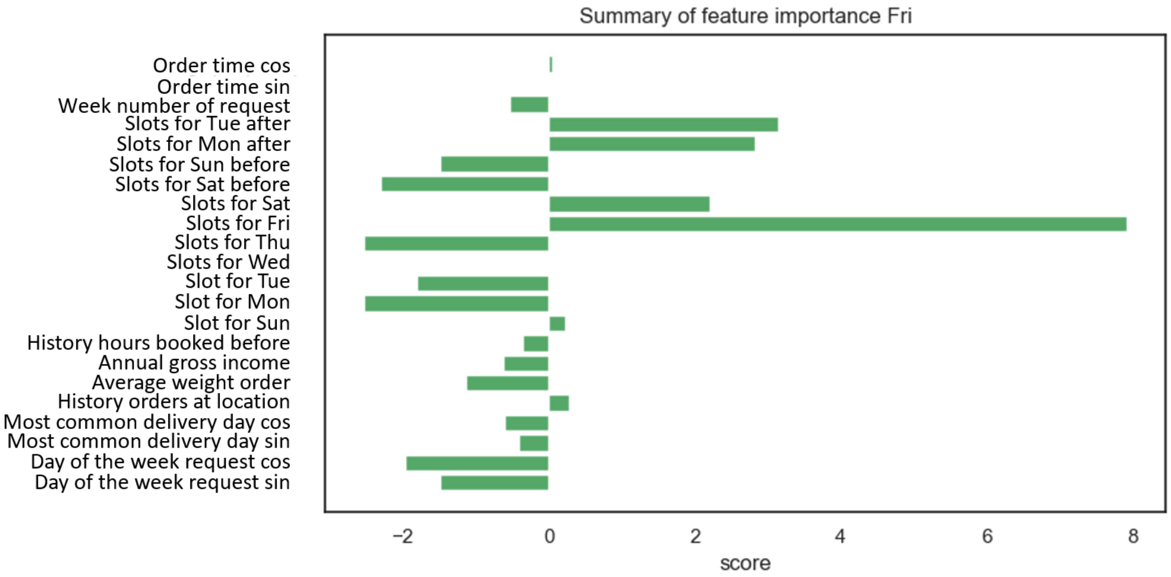


Figure I.5: Feature importance for predicted deliveries on Friday.

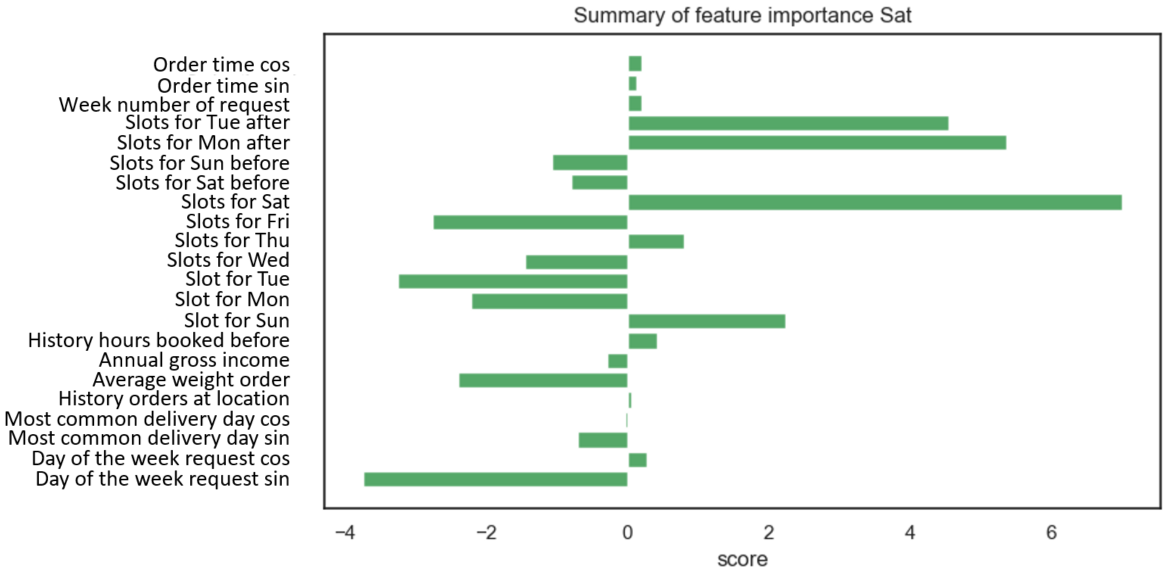


Figure I.6: Feature importance for predicted deliveries on Saturday.

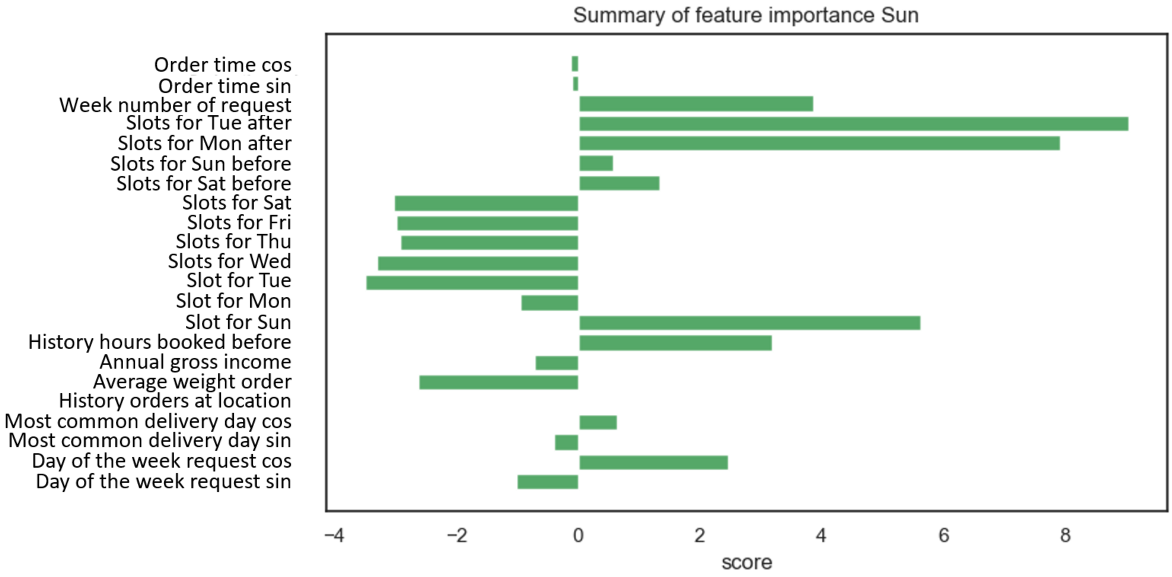


Figure I.7: Feature importance for predicted deliveries on Sunday.

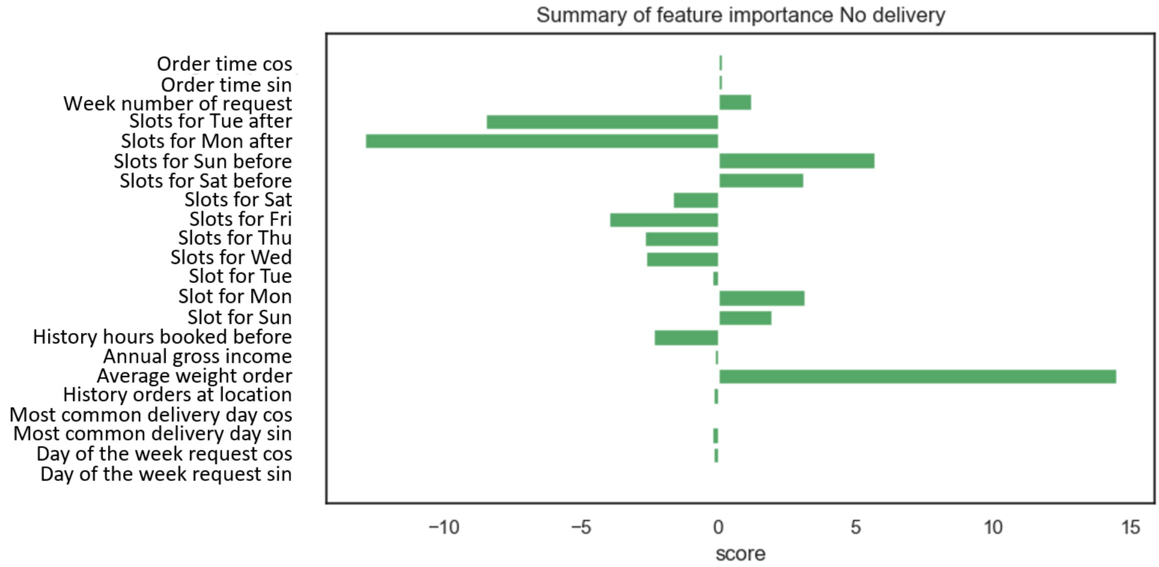


Figure I.8: Feature importance for predicted when the order is assumed to be lost.

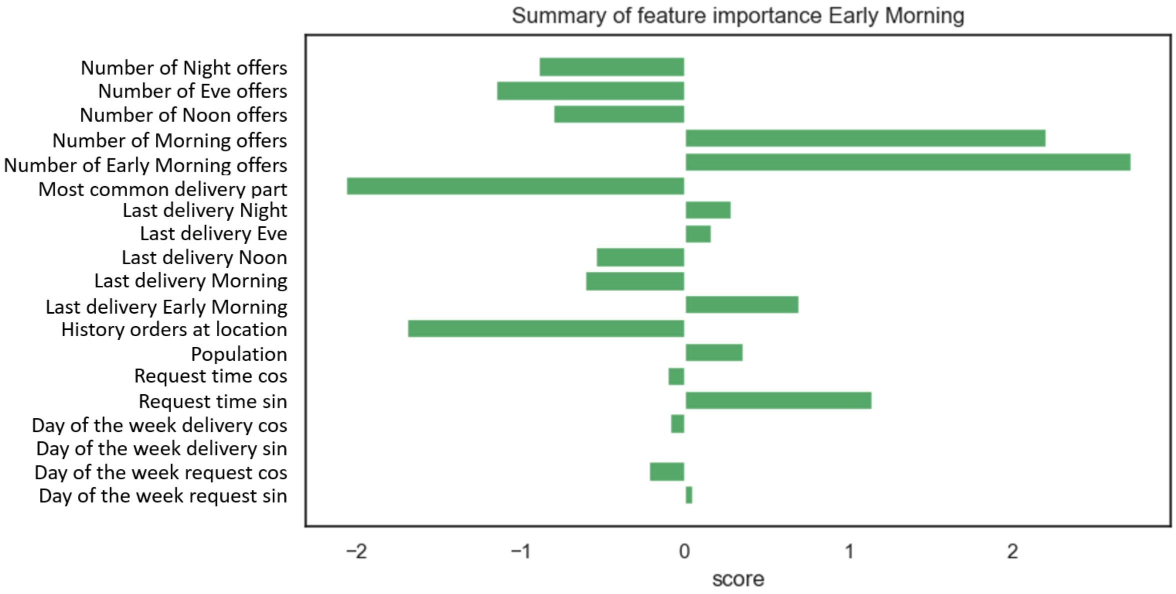


Figure I.9: Feature importance for predicting that the delivery will take place in the Early Morning.

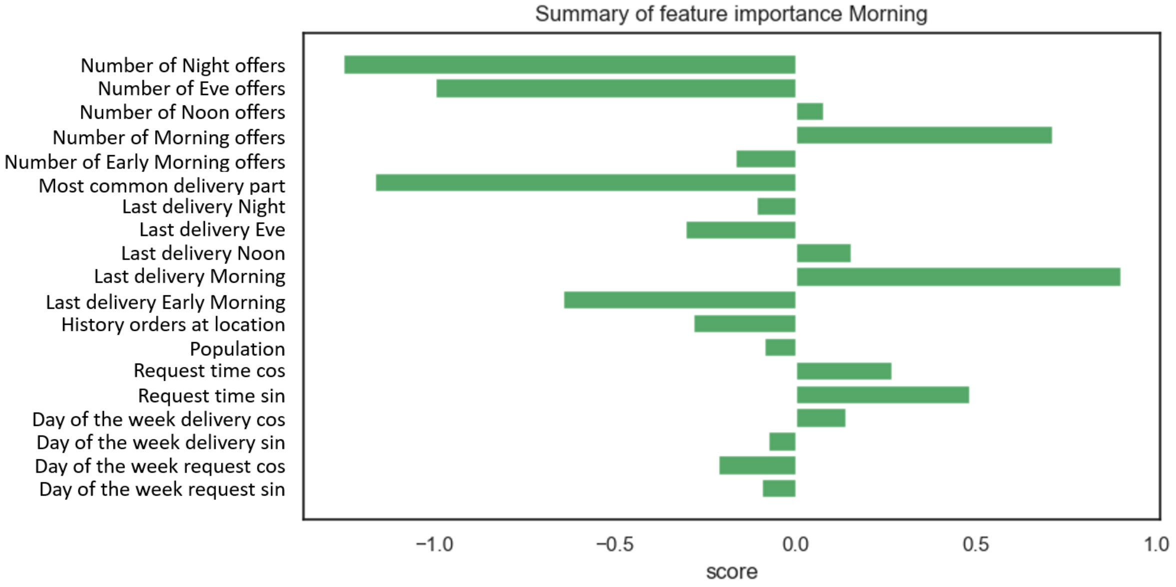


Figure I.10: Feature importance for predicting that the delivery will take place in the Morning.

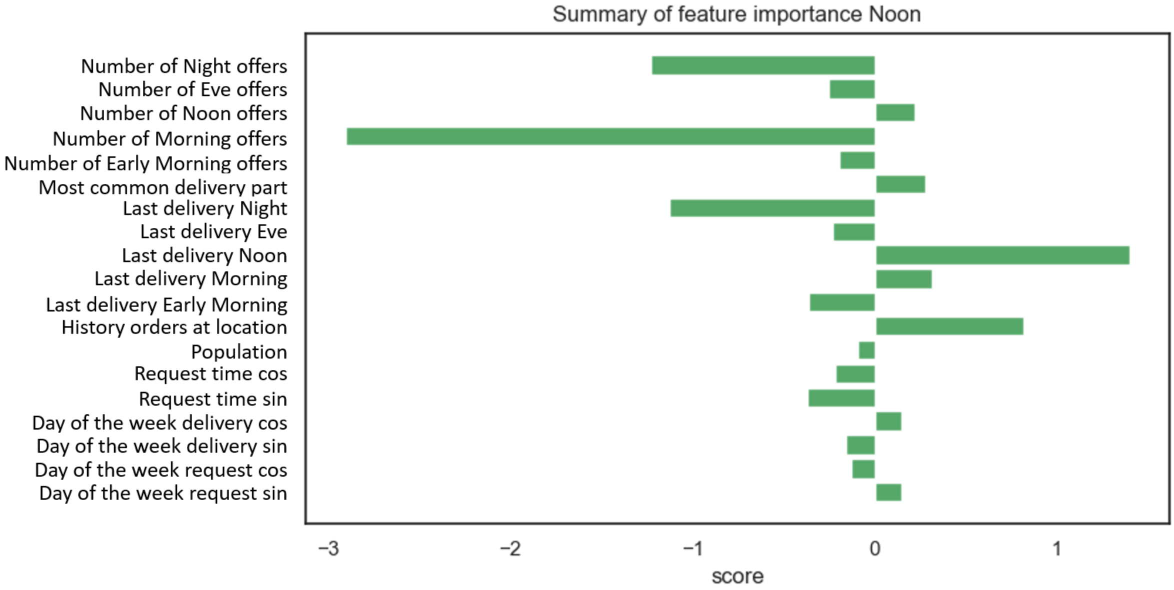


Figure I.11: Feature importance for predicting that the delivery will take place in the Noon.

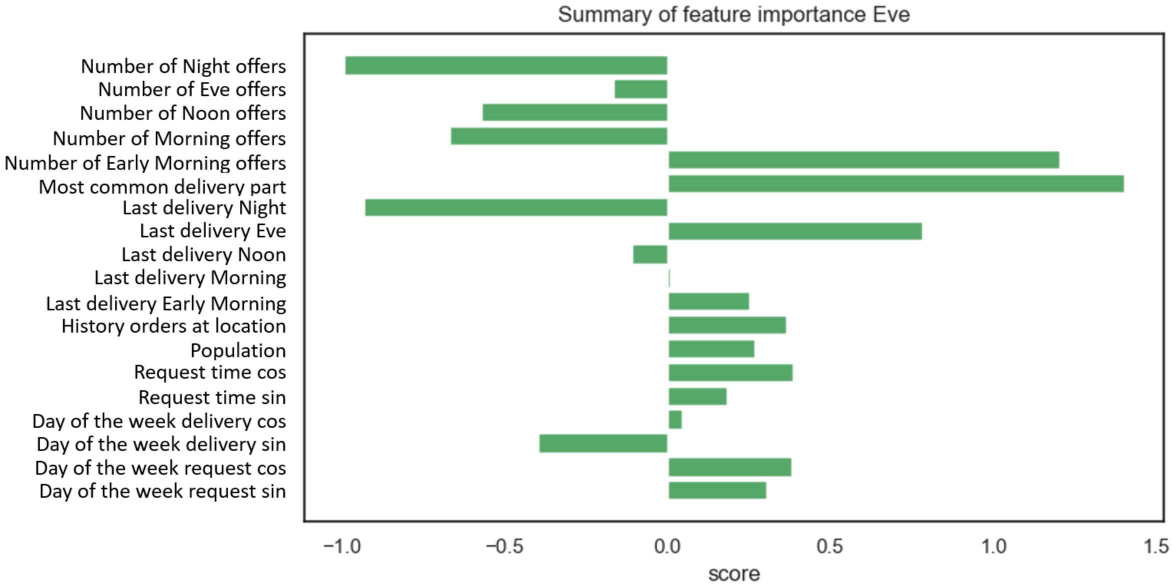


Figure I.12: Feature importance for predicting that the delivery will take place in the Eve.

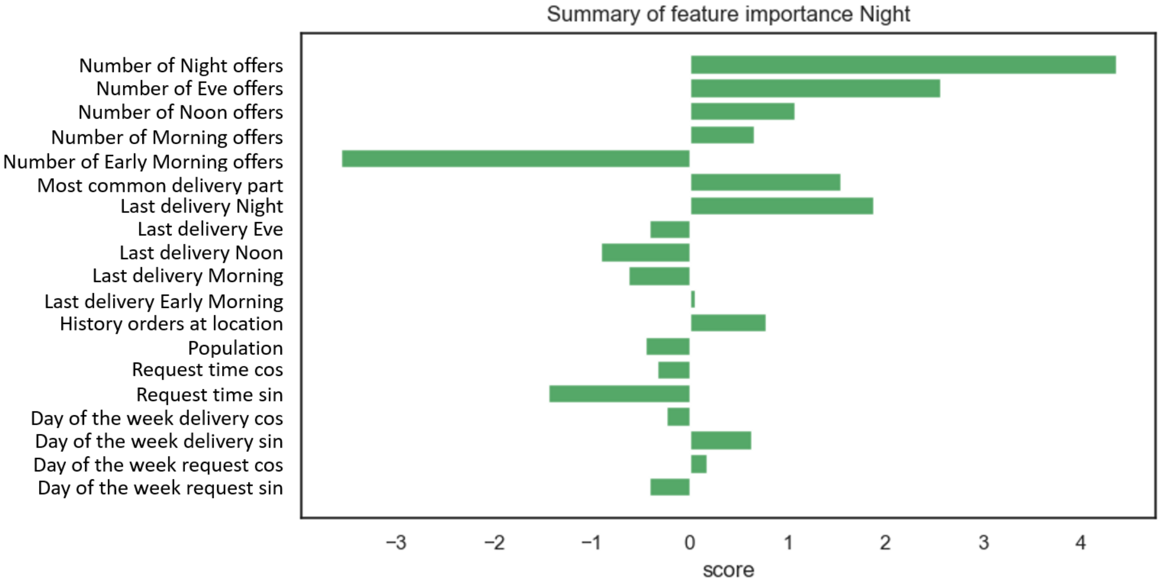


Figure I.13: Feature importance for predicting that the delivery will take place in the Night.

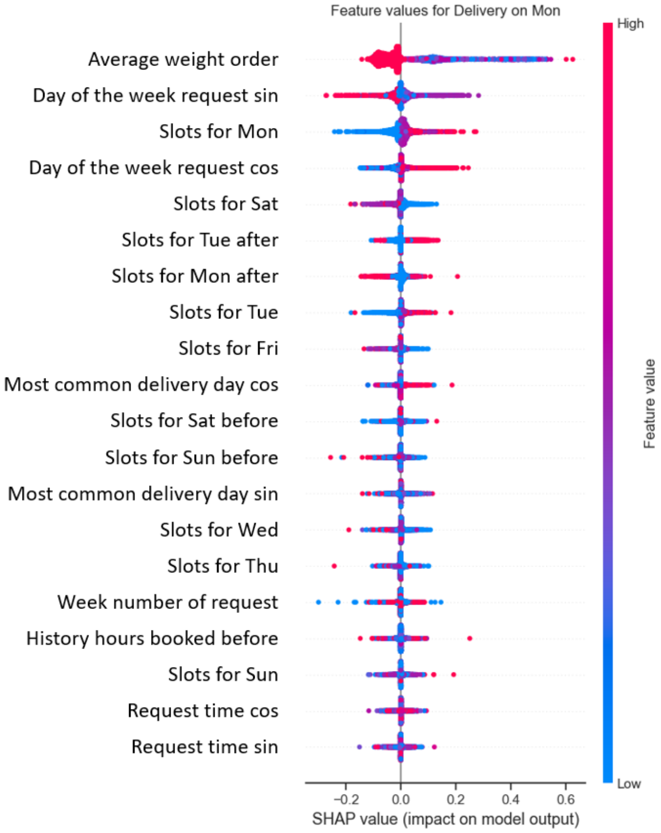


Figure I.14: Feature importance for predicting that the delivery will take place on Monday.

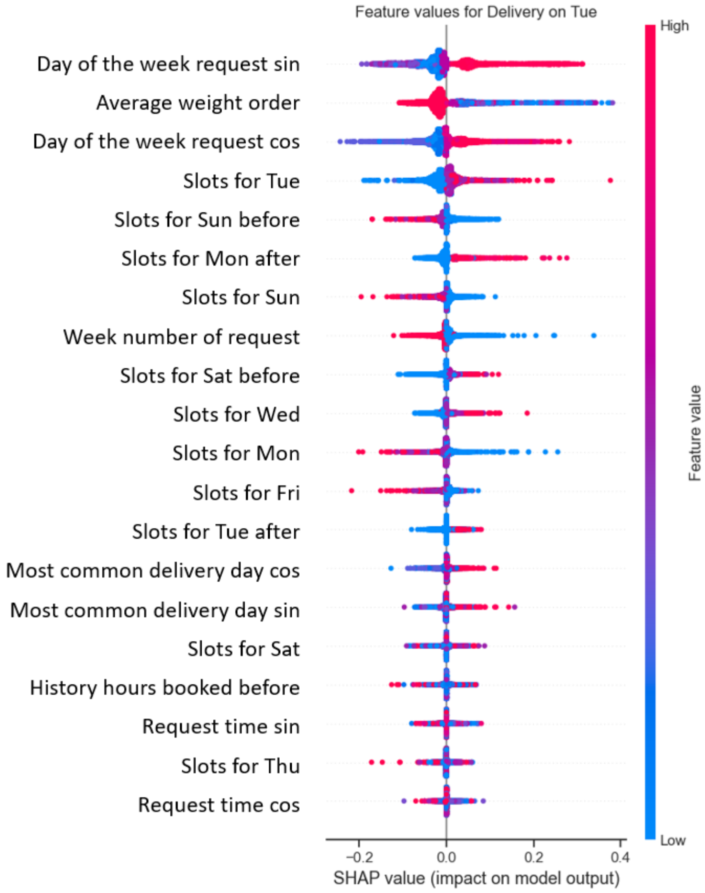


Figure I.15: Feature importance for predicting that the delivery will take place on Tuesday.

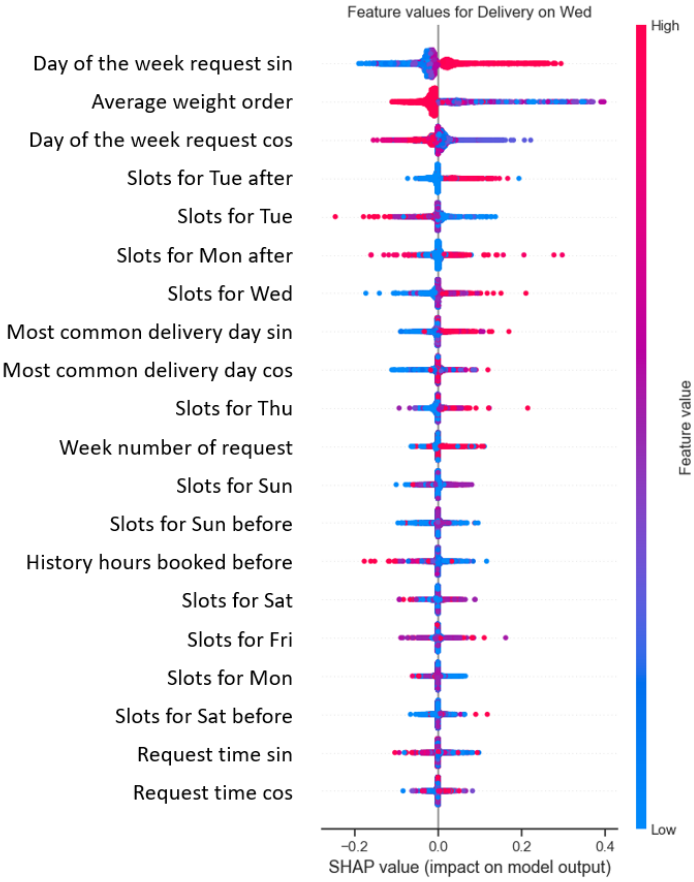


Figure I.16: Feature importance for predicting that the delivery will take place on Wednesday.

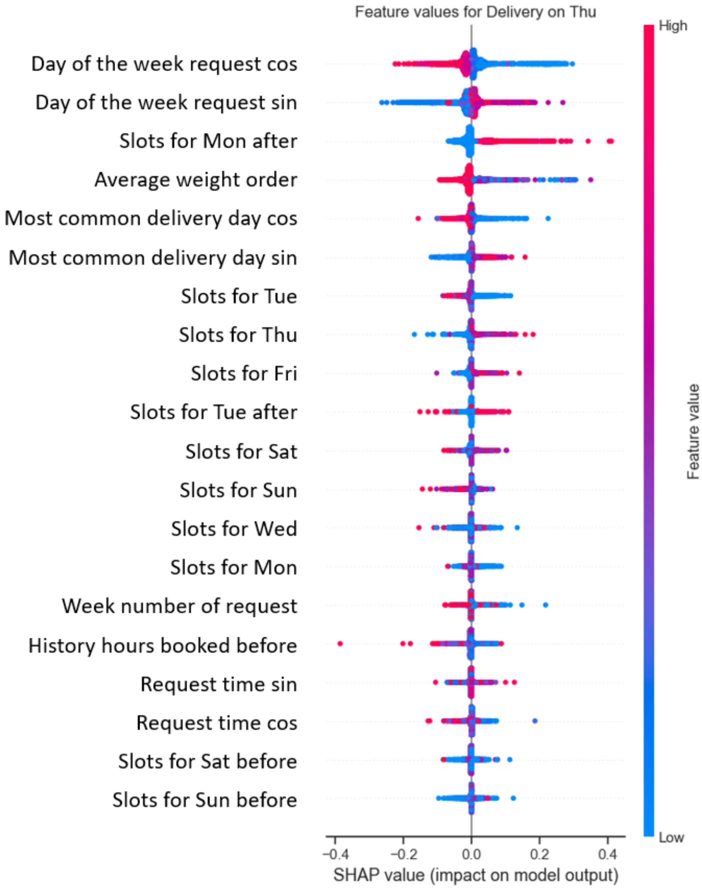


Figure I.17: Feature importance for predicting that the delivery will take place on Thursday.

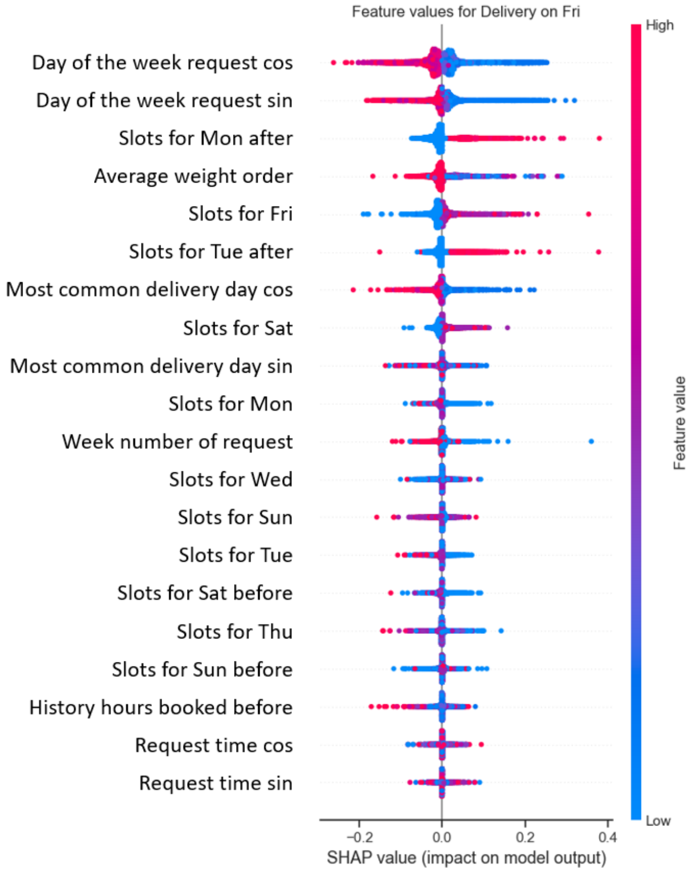


Figure I.18: Feature importance for predicting that the delivery will take place on Friday.

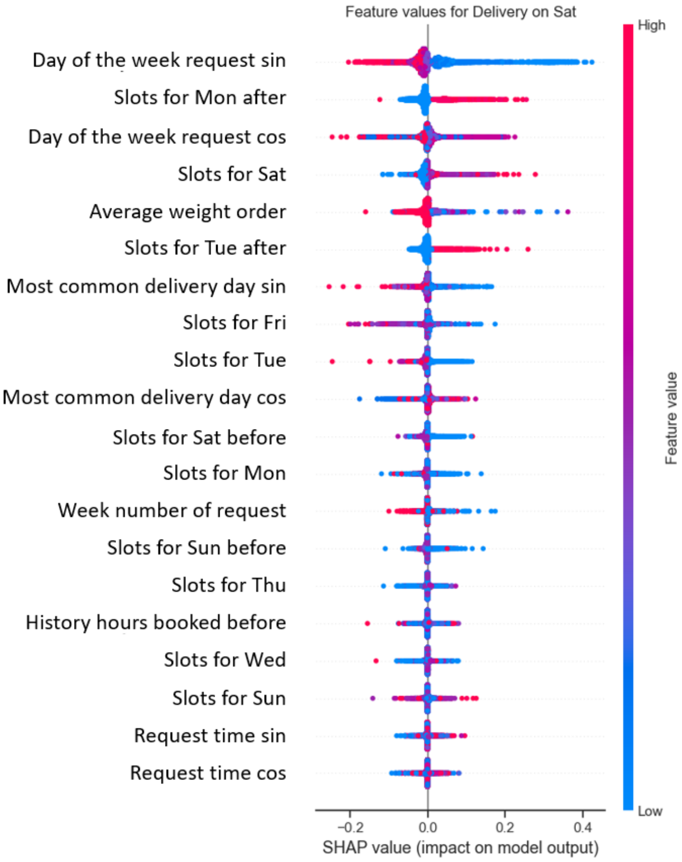


Figure I.19: Feature importance for predicting that the delivery will take place on Saturday.

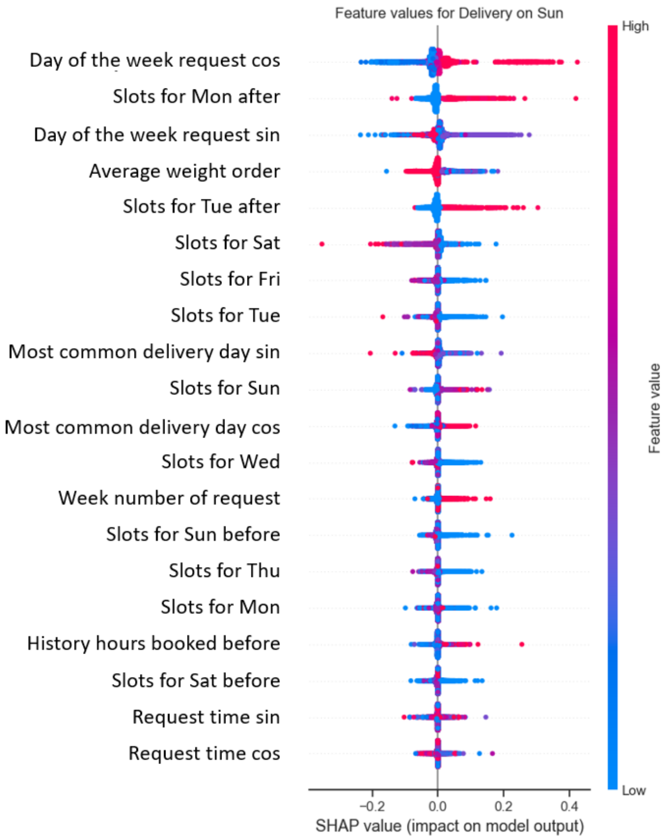


Figure I.20: Feature importance for predicting that the delivery will take place on Sunday.

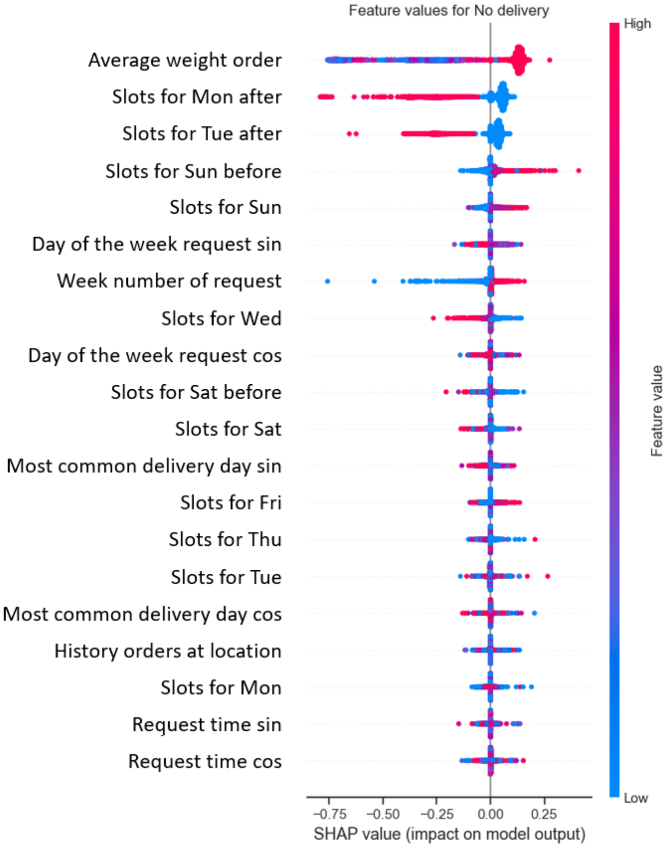


Figure I.21: Feature importance for predicting if the order is placed or lost.

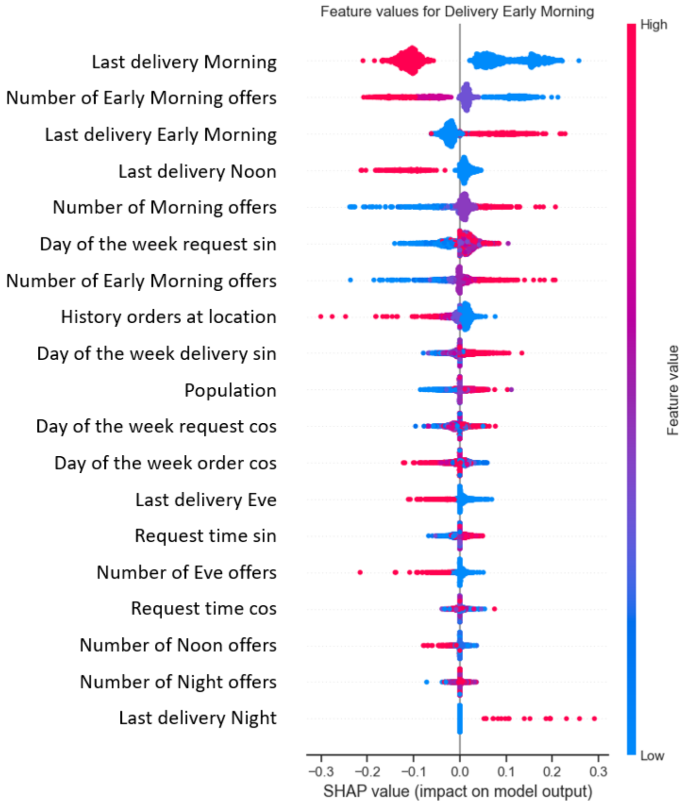


Figure I.22: Feature importance for predicting that the delivery will take place in the Early Morning.

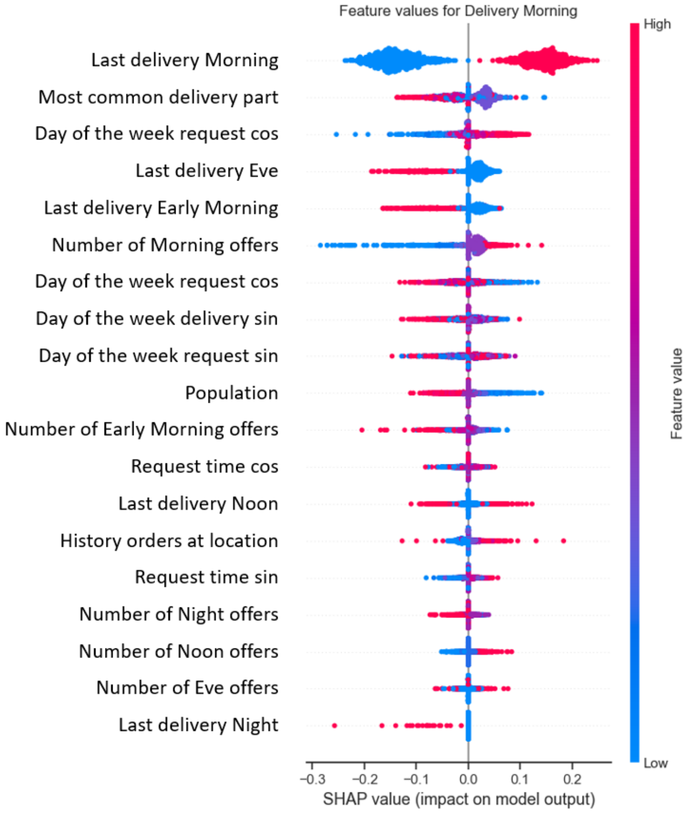


Figure I.23: Feature importance for predicting that the delivery will take place in the Morning.

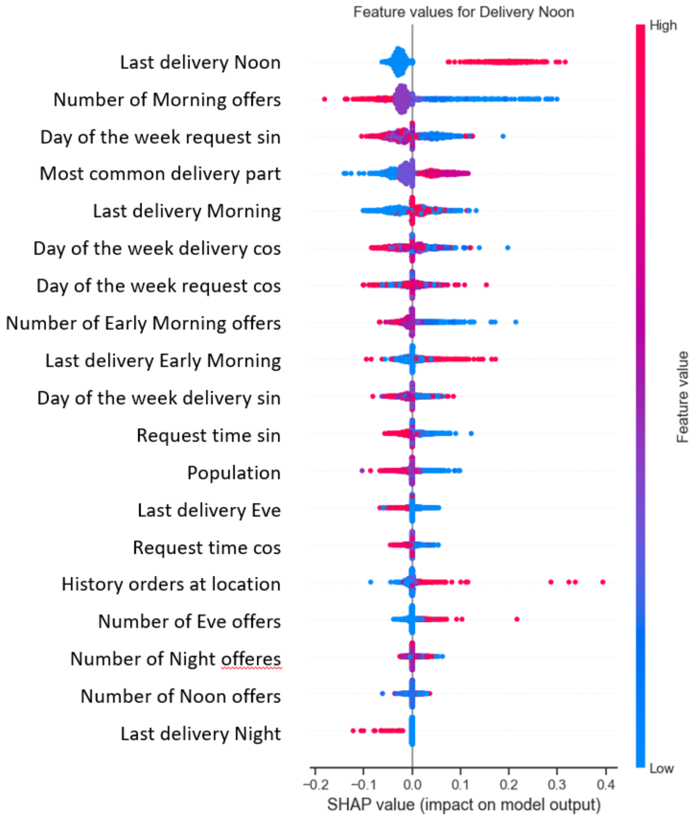


Figure I.24: Feature importance for predicting that the delivery will take place in the Noon.

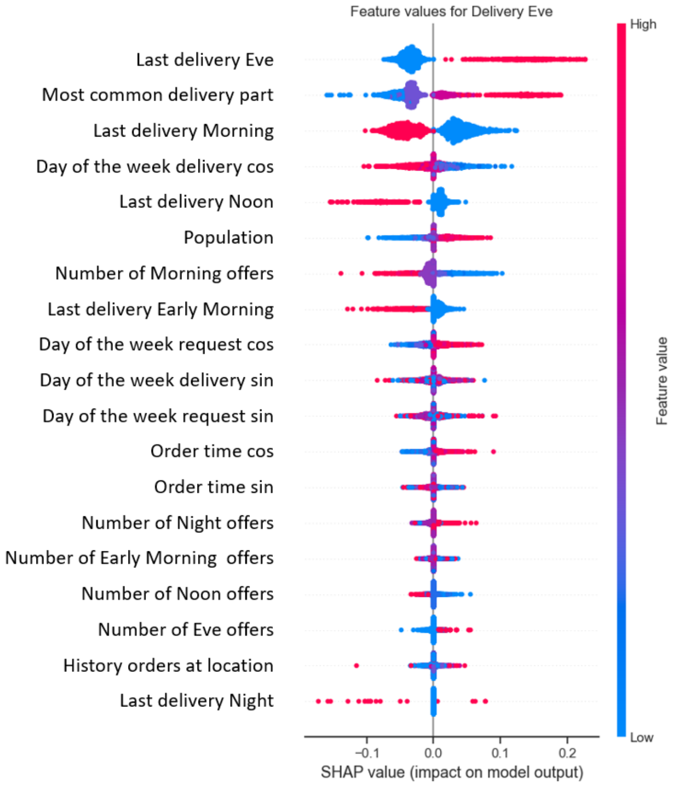


Figure I.25: Feature importance for predicting that the delivery will take place in the Eve.

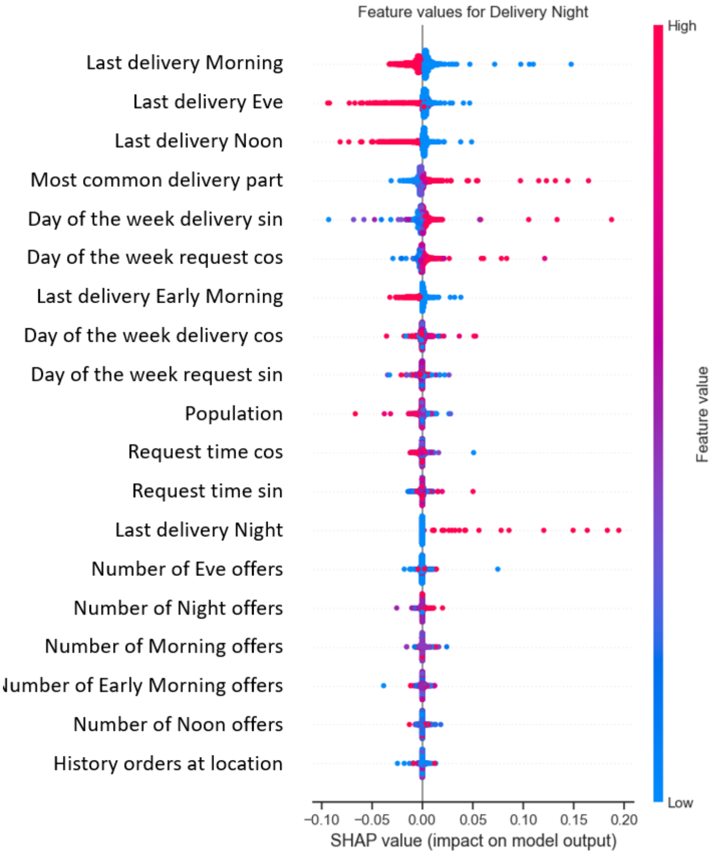
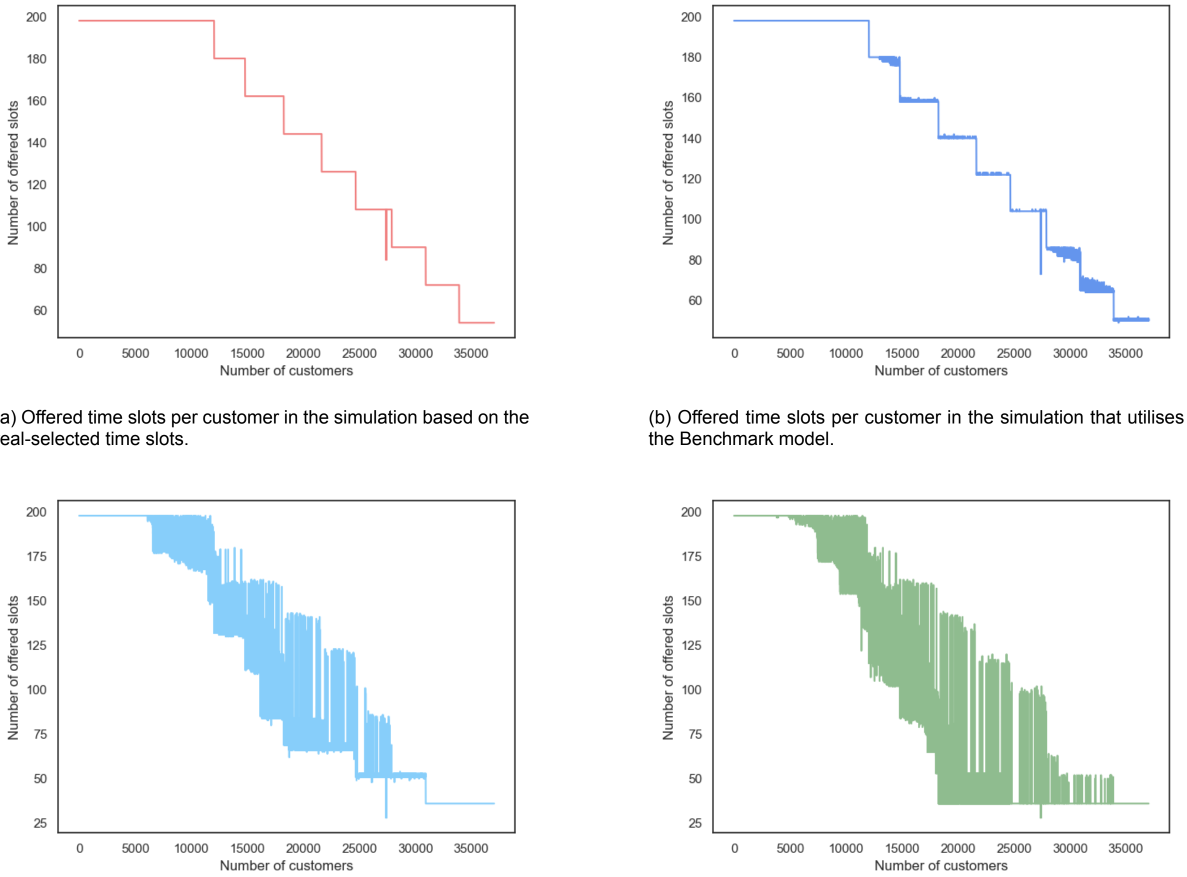


Figure I.26: Feature importance for predicting that the delivery will take place in the Night.

J Additional simulation figures



(a) Offered time slots per customer in the simulation based on the real-selected time slots.

(b) Offered time slots per customer in the simulation that utilises the Benchmark model.

(c) Offered time slots per customer in the simulation that utilises the NN model

(d) Offered time slots per customer in the simulation that utilises the MNL model

Figure J.1: Overview of the distributions of offered time slots per simulation.