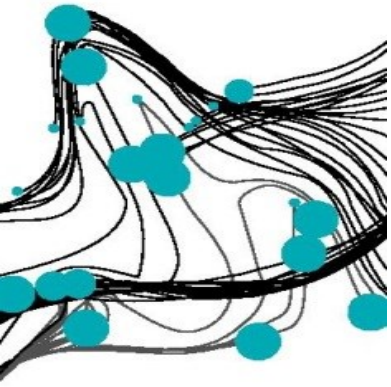
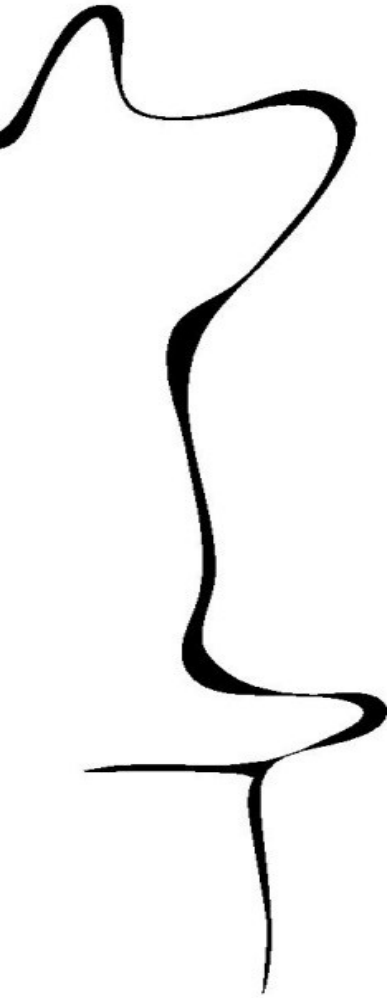


PREDICTING DEMAND OF REPLACEMENT CARS FOR BREAKDOWN CASES USING MACHINE LEARNING TECHNIQUES



Siti Yaumi Salamah

Master Business Information Technology

Faculty of Electrical Engineering, Mathematics & Computer Science

August 2019

GRADUATION COMMITTEE

Maurice van Keulen
Faculty EEMCS, University of Twente

Adina Aldea
Faculty BMS, University of Twente

Chantal Epskamp
ANWB

Remko Stam
ANWB

UNIVERSITY OF TWENTE.

ABSTRACT

Roadside assistance is an emergency service provided to assist people with vehicle breakdown incidents on the breakdown location. Rental cars are usually provided as replacement vehicles for vehicles that cannot be repaired on the spot. Predicting the demand of replacement cars for breakdown cases are essential for the rental car company providing this service. In this research, we aim to investigate to what extent the demand of replacement cars can be predicted. Domain analysis based on literature study and interview with domain experts were conducted to generate a list of potential predictors. Using real world data of replacement car orders in the Netherlands and external data such as weather and calendar data, we compared several machine learning and classical time series models to predict daily demand of replacement cars. Various aggregation levels for spatial level and product type were investigated. The result shows that the best performing model is an XGBoost model trained on a shuffled training and test set, with a 9.49% mean average percentage error. Moreover, we found that prediction performance gradually decreases as the prediction level goes deeper. In addition, we proposed to address the outlier demand by identifying outliers, predicting them separately, and classify a future observation. Empirical comparison of three different approaches was also carried out to produce prediction interval as a means to estimate uncertainty.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 MOTIVATION

A vehicle breakdown is a mechanical defect of a motor vehicle in such a way that the failure prevents the vehicle from running or where continuing being operated is not safe. This incident can arise in various occasions, such as during travelling for holiday or work, even when the vehicle is not in use. There are numerous reasons for a vehicle breakdown, from a dead battery, fuel, ignition, to other mechanical problems. In a case of breakdown, driver and passengers' convenience and safety can be at risk. In this situation, roadside assistance has come in handy, providing breakdown service that oftentimes cannot be done by the driver or passengers of the vehicle themselves and is in need of an expert skill to deal with the failure.

Every year, there are more than 1 million breakdown incidents recorded in the Netherlands (ANWB, 2019). However, not all vehicle faults can be repaired by the road patrols on the spot. In such cases, roadside assistance providers provided transport assistance to tow away the vehicles to a garage for reparation and bring the passengers to a destination. In addition, replacement vehicles (i.e. rental cars) can be provided for the customers while the cars are being repaired.

The whole processes form a complex workflow involving several parties, starting from receiving the incident report from a driver to possibly picking up and returning a rental car. It becomes a challenge for roadside assistance providers to ensure customer satisfaction along the line, starting from the emergency call service, roadside assistance, transport, and rental car services. Roadside assistance providers need to deal with not only resources and capacity planning for the roadside assistance process, but also for the replacement vehicles. Providers of the replacement vehicles need to optimize the number of available rental cars so that customers' needs are fulfilled as fast as possible even during the peak season, while not leaving a lot of cars idle.

On top of that, among the processes along the line, car rental logistics is a complex problem which is difficult to deal with. A lot of effort has gone into studying the optimization of rental cars capacity and fleet management (Fink & Reiners, 2006; Yang et al., 2008; Gupta & Pathak, 2014; Oliviera et al., 2017a; Roy et al., 2014).

Demand forecast is identified as the basis of car rental logistics planning and decision making (Fink & Reiners, 2006; Pachon et al., 2006; Roy et al., 2014; Oliviera et al., 2017a). It acts as a key enabler for an improved customer experience by reducing out-of-stocks situations and reducing costs due to better planning of inventory, among other advantages (Böse et al., 2017). However, issues of uncertainty of demand in car rental industry and the difficulties in forecasting it have been brought to light. Oliviera et al. (2017a) reviewed literature related to fleet and revenue management in car rental industry. They addressed uncertainty issues in fleet management and demand uncertainty is the most recognized among all. Yang et al. (2008) highlighted that the car rental industry has a different characteristics compared to other industries, which are the certain customers´ behavior characters, such as the high rate of no-show, returning cars remotely, and the uncertainty of rental duration contribute to a big part in demand forecast difficulties in the industry. Therefore, there is a need of more accurate demand prediction models for car rental industry considering the unique characteristics it has compared to the demand in other industries such as airline and hotel industries.

Despite some similarities of characteristics with the conventional car rental companies, rental car companies that provide replacement cars as their core business process also have a distinct characteristic and cannot be treated completely the same as the conventional rental car companies. The demand for rental vehicles in the context of vehicles breakdown is not only affected by customer behavior such as travelling pattern and busy commuting hours, but also by the factors that might affect the severity of car breakdown or the capability to quickly repair a car, which in result might end up with the need of replacement vehicles, i.e. the rental vehicles. To the best of our knowledge, little to no literature have covered the prediction of demand for such industry.

Lots of existing literature with car rental logistics as the use case consider the demand forecasting problems as a traditional time series forecasting problem (Pachon et al., 2006; Hong et al., 2007). They consider historical data and attributes of the time series, such as trend and seasonality for the forecast. In another literature, an intervention from analyst is expected to adjust the forecast in the existence of special events (Geraghty & Johnson, 1997).

Besides the traditional time series forecasting technique, time series forecasting can also be analyzed as a machine learning problem, specifically the supervised learning problem. Time series problem can be restructured into a supervised learning problem, thus allowing us to extract more valuable features from a timestamp and include external data as the factors that will affect demand prediction result. Therefore, factors like special events, holidays, and weather can be incorporated into the demand forecasting model and the linear and non-linear relationship of these features can be examined by using machine learning techniques. Research on machine learning-based demand forecasting dates back to the early 2000s and machine learning techniques have been shown to be valuable in getting a more accurate demand forecast.

However, despite the accuracy a machine learning technique can produce, there is a limitation in a predictability of demand given the nature of it. One would expect to find uncertainty in a demand forecast as there are typically discrepancies between forecasts and actual values (Ericsson, 2001) due to the uncertainty in input values (McKay, 1995), which can include a lot of factors, from human behavior to environment related factors. Therefore, there is a need in providing a measure of uncertainty for the decision makers to base their predictions on. One way to communicate uncertainty is by using forecast intervals as they generalize point forecasts to represent and incorporate uncertainty (Hansen, 2006). However, in machine learning forecasting, the capability to specify uncertainty or intervals are rarely included in the research agenda in the field (Makridakis et al., 2018).

Therefore, in this study we want to address the applicability of machine learning forecasting in the specific domain of replacement cars, which has not been a focus of any previous works. We first investigate the predictability of the demand of rental cars as a means of replacement vehicle in the case of unrepairable breakdown by utilizing available machine learning techniques to build a prediction model. In addition, we aim to produce a measure of forecast uncertainty through intervals as an essential part of demand forecast in practice. We compare several approaches to create the intervals and reflect on the strengths and weaknesses of each interval with regards to the quality and the applicability in the practice.

1.2 Case Description

ANWB (The Royal Dutch Touring Club) is an organization for traffic and tourism in the Netherlands with roadside assistance as the main line-of-business. ANWB divide its customers into two market segments, namely the Business-to-business (B2B) and business-to-consumer (B2C) market. B2C

market consists of customers with direct membership subscription to ANWB, while B2B market consists of customers that are entitled to ANWB services through their contracts to another company that has a partnership with ANWB. Depending on the contracts, customers are provided with different kind of services, for example different types of replacement cars for when a breakdown cannot be solved on the spot or different duration for which the cars are rented to the customers.

To provide replacement vehicles for its customers, ANWB partner up with Logicx, its daughter company handling towing and replacement vehicles. Logicx´s fleet for the Netherlands is distributed in 85 locations owned by Logicx and its partners. In the event of unsolved breakdown problems, ANWB send a request for replacement cars to Logicx which then assign an available car from the pick-up location closest to the breakdown location or the most convenient location for the customer.

Each pick-up location has different maximum capacity of cars that it can keep in the location, from 2 up two 120 cars per location. To keep up with the demand, Logicx monitor the demand of replacement cars and the number of cars that are away and returned regularly. In case of shortage of cars in a certain location, Logicx can either transport cars from the other pick-up locations or outsource cars to another car rental company, Hertz, that will take about half a day lead time.

Moreover, the rented cars can be returned in any Logicx and partner locations and it can be different from the pick-up location. To deal with this, Logicx take care of the repositioning of the cars from one location to another location. Logicx plan and monitor this activity weekly and daily. The goal is to have an optimum number of cars in each location to avoid extra outsourcing or daily transportation cost, while keeping a low number of idle cars.

In addition to operational efficiency, ANWB want to ensure high level customer satisfaction with regards to the replacement vehicle service. One way to deal with this is by supporting Logicx with a demand forecast that can help them plan the distribution of cars such that customers can always pick up rental cars from the closest location to the breakdown incidents without long waiting time.

1.3 Research Questions

Given the above-mentioned problem, the objective of this study is to predict the demand of replacement cars using machine learning techniques to enhance car rental logistics planning. The following research question was formulated in order to achieve this objective:

To what extent can we predict the demand of replacement cars in the Netherlands?

To answer the main research question, the following sub questions were defined:

SQ1. What features can be used to predict the demand of replacement cars?

To predict the demand, we need to define the input features that can be used as the predictors. To answer SQ1, we carried out literature review and domain analysis through interview and discussions with domain expert within the company to list the candidate features. Then, feature engineering was done to extract valuable features from the available data. Finally, the features used for prediction were selected based on feature importance using a feature selection technique.

SQ2. Which machine learning model is best suited to predict the demand of replacement cars?

We first conducted a literature review to get an overview of techniques that have been used for the problem. Then, to answer SQ2, a set of available techniques with the most suitable characteristics for the problem were selected. After that, these techniques were applied to build the predictive models and the models were evaluated based on their error performances. We then chose the best performing model and reflect on its strengths and weaknesses in regard to the case.

SQ3. What aggregation level works best for the demand prediction?

To answer SQ3, we built the demand prediction model in several levels of granularity, both for the product type and the spatial aggregation level. We then compared the performance of the prediction model from different aggregation levels.

SQ4. How can we estimate uncertainty of the prediction result?

To answer SQ4, we conducted literature review on the techniques that can be used to provide intervals for machine learning-based predictive model. Then, we selected a suitable measurement, compared several approaches and chose one that is most suitable to the case to provide upper and lower bounds in addition to the prediction results.

After answering each sub-question, we concluded the answer to the main research question by presenting the best performance of the replacement car demand prediction in the Netherlands that can be achieved using the analyzed features and the machine learning models considered.

1.4 Research Methodology

In this research, we used the widely adopted framework for data mining and analysis in industry, namely the CRISP-DM (*Cross-Industry Standard Process for Data Mining*). This standard process model supports best practices and is industry-, tool-, and application-neutral (Shearer, 2000). Figure 1 presents the six phases in the life cycle of a data mining project defined in the CRISP-DM framework.

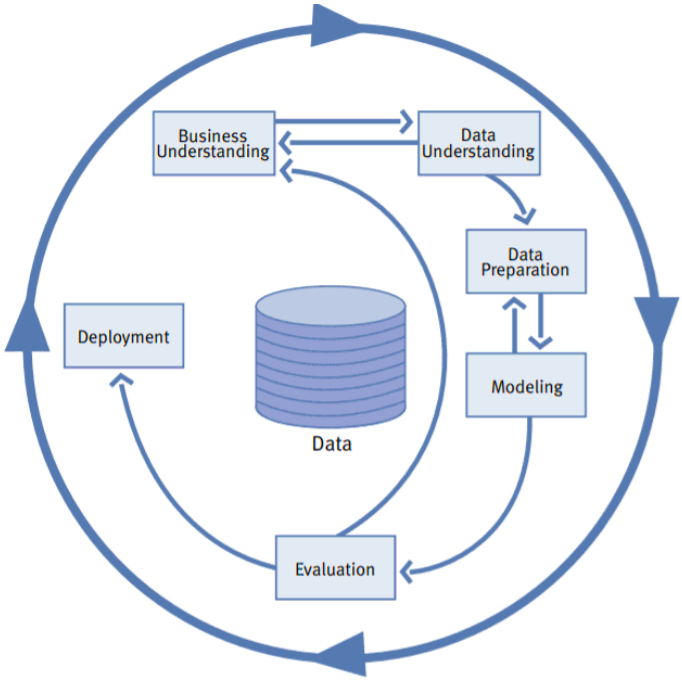


Figure 1 Phases of the CRIS-DM methodology (Chapman et al., 2000)

According to the CRISP-DM, the arrows between the phases show the most important and frequent dependencies between phases. The next process to perform depends on the outcome of the previous one. This concept supports the fact that many data science projects demand an iterative process, allowing one to go back and forth testing different schemes such as to include new features for the model. Therefore, referring to the six phases of the CRISP-DM, we set out an outline of the tasks to be carried out in this study as shown in Figure 2.

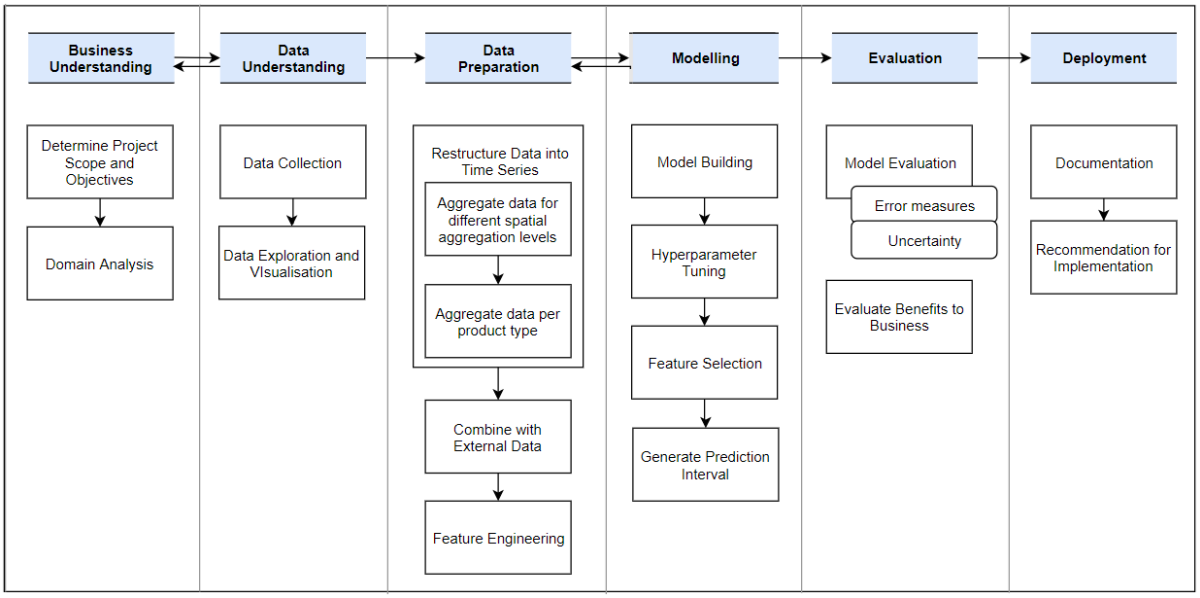


Figure 2 Research outline

The following is the description of the tasks in each phase of CRISP-DM. A more detailed explanation of the methods employed to conduct each step is provided in Chapter 3.

1. Business Understanding
   The initial phase focuses on identifying the scope and objectives of the project. The research starts with a literature review of previous works and related topics to investigate the state of the art of the current research and understand the context of the problem. Then, we determine the project objectives according to the problem. We conduct initial assessment of tools and techniques and create the project plan. Domain analysis via interview with domain experts are also carried out to understand the business and gain the knowledge about the input factors that are important to the problem.

2. Data Understanding
   This phase starts with initial data collection. After that, data exploration and descriptive analytics are conducted to find out the quality of the data, discover first insights, and investigate hypotheses from business understanding phase.

3. Data Preparation
   In this phase, we created the dataset for the modeling phase. It covers the data pre-processing steps, including cleaning and transformation of the data. We process the initial raw data so that it is ready to be fed to the model.

4. Modeling
   At this stage in the project, various modeling techniques are selected after conducting literature review and elimination by aspects to find several techniques suitable for the project. Then, we build and validate each model using the dataset created in the previous phase, including selecting the hyperparameters and the features for the model. We test the model and then produce prediction interval for the model.

5. Evaluation
   The evaluation phase covers the evaluation of several models resulted in the modeling phase by comparing their performances. In this study, in addition to the error of the model, we evaluate the uncertainty of the model represented by the prediction intervals. Then, we relate it to the practice to determine if the business issue has been addressed sufficiently and if the project objectives have been achieved. We review the processes, determine a list of alternative next steps and draw the conclusions of the study.

6. Deployment
   For the deployment stage, we produce a documentation of the model in the form of Jupyter Notebook document, final report and final presentation.

1.5 THESIS STRUCTURE

The remainder of this thesis will be structured as follows.

Chapter 2 Background and Related Works provides a theoretical background of the research topics. Theories and literature are discussed, including theories related to time series forecasting, machine learning techniques. Related works are covered as well.

Chapter 3 Methods describes the research methods chosen for the study, as well as the models, metrics, and tools used in more details.

Chapter 4 Dataset Creation presents the domain analysis to define the predictors, the process of data collection, exploration, and creation of datasets for the demand prediction model. It covers the results for Business Understanding, Data Understanding, and Data Preparation phases of CRISP-DM.

Chapter 5 Demand Prediction Models provides the results of each model and gives a further analysis of the results, including the comparison of prediction intervals. It covers the results for Modelling and Evaluation phases of CRISP-DM.

Chapter 6 Putting the Model into Practice extends Evaluation phase of CRISP-DM by examining the results in relation to the current practice. Recommendations for the Deployment phase are also covered in this chapter.

Chapter 7 Discussion and Future Work discusses the general findings emerged from the study, reviews the limitations of the study and provides recommendations for future research. It is part of CRISP-DM Evaluation phase.

Chapter 8 Conclusions and Contributions summarizes the answers to the research questions and the contribution of the study to theory and practice.

2 BACKGROUND & RELATED WORKS

This chapter consists of background theories used in this study and discussions of previous works related to the study. It starts with an introduction to the domain application, including background about roadside assistance and vehicle breakdown (Section 2.1) and car rental logistics (Section 2.2). Then, Section 2.3 discusses previous research related to car rental demand forecasting to find out the current state of the art in the domain and the candidate features for the prediction model (SQ1). The next two sections serve as an initial exploration and assessment to answer SQ2. Section 2.4 describes the definition of time series forecasting and the comparison of two groups of forecasting techniques. Section 2.5 reviews techniques that have been used in machine learning-based demand forecasting. Finally, Section 2.6 explains about the uncertainty in forecasting and the variety of approaches used in previous works (SQ4).

2.1 ROADSIDE ASSISTANCE & VEHICLE BREAKDOWN

Roadside assistance or breakdown assistance are emergency services provided to assist people who experience vehicle breakdown. A vehicle breakdown is a condition where a motor vehicle suffers from a mechanical failure that prevents it from running or where it is difficult or not safe to continue driving, thus leaving the driver and passengers stranded. On the other hand, vehicles that stall as a result of non-mechanical defect, such as accident, theft, or external fire is not in the definition of a vehicle breakdown.

Roadside assistance mainly consists of services that can be categorized into the following categories [6][25][30][66]:

1. Breakdown service
   Breakdown service is the assistance given by auto mechanic(s) patrolling over an area (from here on will be referred to as road patrol) to repair the defective vehicle on the spot to help the motorists resume their journeys.
2. Transport service
   If the cars cannot be repaired on the spot, roadside assistance providers assist the transportation of the vehicle and its occupants to the nearest automobile repair shop/garage. In the matter of the cars being totally immobile, a tow service will be provided. Passengers can ride the towing vehicle or choose to use other modes of transportation such as public transportation to their destination.
3. Replacement vehicle
   Rental cars are provided as replacement in cases where vehicles with breakdown cannot be repaired on the spot. The cars are rented for a certain period while the broken cars are being repaired in garage or alike.

Roadside assistance is normally provided as a subscription-based service (i.e. to access the service, a customer has to pay a recurring price at regular intervals). This service can be offered by automobile membership associations, roadside assistance specialized companies, car manufacturers, or as a part of car or travel insurance.

A lot of research and development in the industry have been focused on offering breakdown diagnostics, prognostics, as well as predictive and preventive maintenance on vehicles (das Chagas Moura et al., 2011; Prytz, 2014). Das Chagas Moura et al. (2011) studied the prediction of failure and reliability of car engines using machine learning technique on time-to-failure and miles-to-failure

data. They treated the time and miles-to-failure as a time series process as both are ordered in time, thus handling it as time series data. Prytz (2014) investigated machine learning methods to predict upcoming failures of vehicles by making use of either streamed on-board data or historic and aggregated data from off-board sources. Deviations analyzed from the telematics data were associated with the repair history, which resulted in a knowledge base to predict failures on other vehicles with the same deviations. These models aim to make vehicle manufacturers or other affected parties aware of a condition leading to a breakdown in advance and alert drivers of the condition, thus improving the driver safety and reducing vehicle downtime and the cost of maintenance. However, issues related to the optimization of resources of the road patrols, towing company, or vehicle rental company in case of breakdown are rarely addressed.

Despite the limited scientific works on the issue, particularly for the roadside assistance context, over the last few years, roadside assistance providers have reported the use of predictive analytics for breakdown prediction. One developed a predictive technology using machine-learning algorithm that uses historical data, weather and humidity indicators, and real-time traffic and GPS information is claimed to predict the location, time, and type of breakdown with a high accuracy (Jackson, 2018). This is aimed to cut down the waiting times for the road patrol and rescue trucks by placing them within a certain area. Another approach to address similar issue is by developing analytics solution to anticipate where and when roadside incidents will most likely occur based on certain weather and road conditions (Allstate Roadside Services, 2019). A further analysis incorporating geographical data was also reported. One company developed an analytics tool to predict dispatch volume more accurately, especially for extreme weather conditions in winter (O'Donnell, 2014). It utilized weather historical data and data of millions of roadside events and took a more granular look at weather events, including distinguishing profile of events in different geographies. These solutions indicate a significant need of such systems for the industry but at the same time demonstrate the contrasting state of the art between academic and industrial research for the specific problem.

2.2 Car Rental Logistics

Car rental companies in most cases manage a large number of locations to provide rental service for their customers. Therefore, they are faced with logistics management problem in their operation process. Car rental companies need to avoid opportunity loss as a result of shortage or out-of-stocks situation and maintain good customer service quality. Operating a large fleet will be able to prevent shortages but it comes with large holding costs and vehicle depreciation for the company, as not all the cars will be utilized the whole time because of the varying demand of rental cars.

Figure 3 illustrates the life cycle of the rental cars. The whole life cycle covers the procurement of new cars, distribution of cars to locations, repositioning of cars between locations, renting out and retrieval of rented cars, and the disposal of old cars (Fink & Reiners, 2006). Along this whole cycle, the logistics operation mainly concerns with the deployment of rental cars through fleeting, defleeting, and car transfers between locations. These logistics processes aim to optimize the utilization of the whole fleet while still providing high service level to the customer, for instance by delivering flexibility and responsiveness in the rental car process and reducing the waiting times.

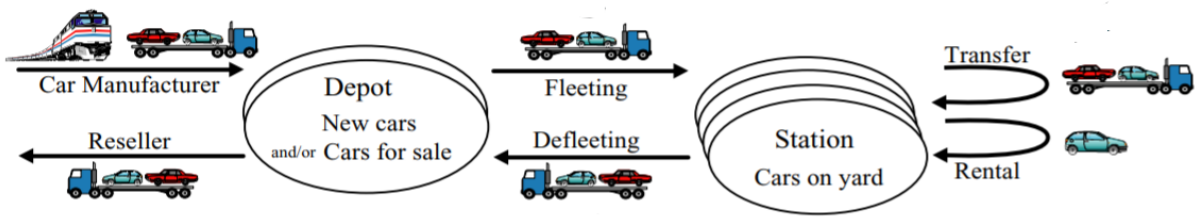


Figure 3 Life cycle of rental cars (Fink & Reiners, 2006)

Roy et al. (2014) defined three strategies for rental cars repositioning between locations, which are no repositioning of either vehicles or customers, repositioning of customers, and repositioning of vehicles. The first configuration serves the customers at certain locations using the cars available at the location without any repositioning. The second strategy proposed to reposition some of the customers to other locations to minimize waiting time, while the last one repositions cars to a location where the cars stay for the remainder of the planning period. The latter allows each rental car location to have access to a larger fleet. However, it comes to an expense for the rental car company, which is the extra cost and resources required for the transportation of cars. Furthermore, it is not always possible to be done in the case of short-term demand changes and the decision is constrained by the capacity of each location.

The current practice in the industry addressed the entire fleet planning problem in three phases (Pachon et al., 2006):

1. Pool segmentation
   Pool segmentation deals with clustering all car rental locations into pools based on distances and variability of aggregated pool demand.
2. Strategic fleet planning
   Fleet planning at the strategic level determines the total fleet size for each pool in the long-term horizon. In this phase, decisions about acquisitions and dismissal of cars from and to manufactures as well as car repositioning between pools are made.
3. Tactical fleet planning
   At the tactical level, the daily operation at each pool is executed. This phase involves determining the optimal number of cars for every location within a pool and the repositioning of cars between the locations in a pool.

Based on the above-mentioned phases, we can classify the scope of our research into the tactical fleet planning phase. As we predict the demand of replacement cars in the Netherlands for a short term operational period (i.e. daily prediction), we expect to provide a high level insight of the variation of demand per day, thus providing an insight of whether adjustment is required for the number of cars available for every locations in the Netherlands.

2.3 Car Rental Demand Forecasting

As the demand varies, car rental businesses may face several challenges, such as income loss and reduced customer satisfaction from unfulfilled demand because of shortage of rental cars, extra cost for outsourcing from third party or transporting cars one at a time to match sudden demand changes, low utilization of cars in case of high inventory level (Fink & Reiners, 2006; Oliveira, 2017a).

An effective way to deal with the difficulty of car rental logistics problem is to forecast the demand and return of rental cars in the near future (Xu & Lim, 2007). Car rental demand forecast can be used as the basis for planning decision to deal with these problems, such as the decision to reposition the cars from one location to another location that may get more demand according to the forecasted values. It has been addressed as an important issue in the car rental industry for long.

According to the period of forecast and its application, the demand forecast can be categorized in the following way (Hyndman & Athanasopoulos, 2018):

1. Short-term forecasts
   Short term forecast is required in operation and management, such as for scheduling, production, and transportation problems.
2. Medium-term forecasts
   Medium-term forecast is useful for determining future resource requirements.
3. Long-term forecasts
   Long-term forecast is used for design and strategic/high level planning.

Detailed demand forecasts can act as a reasonable basis for a short-term planning horizon while medium-term or long-term forecasts in most cases depend on aggregate demand (Fink & Reiners, 2006). Pachon et al. (2006) presented how demand level is important for all levels of fleet planning problems. They included variance of aggregated demand in a pool as one of the variables needed to solve a pool segmentation problem. In the level of strategic planning, a car rental company should consider how cars will be redistributed among pools in a responsive way taking into account the fluctuations of the demand load of the pool. Similarly, the decision to reposition cars between locations is driven by the demand level at each operating location.

Car rental demand forecasting has been studied in different levels of aggregation, in its relation to other influencing factors such as pricing and supply level, and in different combinations of external factors. **Geraghty & Johnson (1997)** classified demand forecast into 2 primary levels of aggregation, namely length-of-rent and on-rent levels. The length-of-rent forecast forecasts the demand for each potential length of rent for each arrival day in the planning horizon, while the on-rent aggregation forecasts the number of cars in use (i.e. the number of cars picked up and already in use) on a specific day. However, the proposed demand forecasts still require an intervention from analysts to defines special events and override the system-generated factors in the existence of special events.

In addition to forecasting the rental car demand, **Xu and Lim (2007)** developed a model that takes into account the supply of the rental car as well. They forecasted the net flow for rental cars, i.e. the difference between supply and demand, by making use the trend of the net flow from historical data and predicting the change in the next period using neural network.

Fink and Reiners (2006) modeled a linear regression function to determine detailed short-term rental demand forecasts over the planning horizon. They identified four main factors that affect the regression function, which are the location, the period of forecast within a week, the lead time between the current period and the period to be forecasted, and the car group. Thus, the function will establish an estimate of the rental car demand per car group at a location in some period. The authors were aware of some influencing factors that can be considered in the model such as seasonality, local events, and weather conditions. However, they are not included in the proposed model due to limited data accessibility.

Lei et al. (2017) studied the rental cars demand prediction, taking into account the increasing trend of car sharing practice using rental cars. With the intention of making the prediction a useful input to

determine new rental car stations to accommodate car sharing practice, they focused on analyzing the demand based on spatial travel patterns by dividing it to several functional regions (e.g. business, entertainment, resident, etc.). They included rental and geographical information such as rental behavior features (i.e. whether the rental is meant for business or individual use, travel distance, time, frequency, and historical data of departure and arrival), destination feature (i.e. number of arrival at each category of region), density of point of interests, and a number of extracted statistical features.

Another study of forecasting for demand and allocation of resource was carried out by **Verma et al. (2006).** Even though their case study was not for the car rental industry, it shows quite a similarity and potential to be implemented in the industry. They introduced a novel approach where they predicted the demand of resource (i.e. service clouds tenant) using various techniques. They classified demand on whether it would increase or not using various classification techniques (i.e. logistic regression, multilayer perceptron, support vector machine, and reduced error pruning tree/REPTree), predicted short-term demand using trend seasonality model and exponential moving average, and predicted long-term demand using ARX/ARMAX. Then, they followed it up with dynamic resource allocation using heuristic rules based on the predicted demand, for example defining the actions to do when the demand is predicted to increase.

Thus far, we have seen several different approaches to forecasting car rental demand. Table 1 gives an overview of the variety of prediction output and aggregation type for car rental demand forecasting from previous works.

Table 1 Comparison of Rental Car Demand Prediction Approaches

Author	Target variable	Aggregation type
Geraghty & Johnson (1997)	Number of cars in use per day	Period (daily)
	Number of cars requested per certain length-of-rent per day	Length-of-rent, period (daily)
Xu & Lim (2007)	Demand of rental car per day	Period (daily)
	Difference between returned car and rented car per day	Period (daily)
Fink & Reiners (2006)	Demand of rental car per car group per location per period	Car group, location, period
Lei et al. (2017)	Demand of rental car per pre-determined functional group	Location
Verma et al. (2006)	Demand will increase vs not-increase	Class of demand (increased or not)

Depending on the goal of the forecast and how the forecasted values will be utilized in the system, these studies have defined various potential variables as the outcome of the prediction related to rental car demand and the aggregation levels to consider. Overall, looking at the objectives and aggregation levels, the car rental demand prediction explored in these literatures concerned for short to medium-term forecast. Geraghty and Johnson (1997) and Xu and Lim (2007) considered the daily demand in particular. In this project, daily demand will also be the main focus given the objective of the research which is to enhance the car rental logistics planning to achieve operational efficiency and maintain customer satisfaction.

Fink and Reiners (2006) mentioned the car (product) group as an important factor to aggregate the demand prediction per car group. Similarly, Oliveira et al. (2017a) developed a framework to give a

clear overview of how to deal with fleet and revenue management in car rental industry and presented how demand is the general input to the process, with a difference in aggregation level needed for each fleet management process, and showed that demand for a specific product is the most relevant uncertain factor related to the fleet and revenue management problems. Therefore, the prediction of demand per car group may need to be investigated in the case of different types of product exist in the business.

Both Fink and Reiners (2006) and Lei et al. (2017) proposed demand prediction per location levels. However, this location or spatial aggregation level may vary according to the requirement, such as per rental car location or per group of locations. In this study, we aim to investigate the feasibility of the replacement cars demand prediction in the context of breakdown incidents starting from the high-level location aggregation, namely per country/the Netherlands.

2.4 Time Series Forecasting

Forecasting is about predicting or estimating the future, given all the available information, including historical data and knowledge of any future occasions that could impact the forecasts (Hyndman & Athanasopoulos, 2018). Time series is a series of values collected at successive times, often at the same intervals, making it a sequence of discrete data. Most quantitative forecasting problems deal with this kind of data. Time series forecasting is thus defined as the use of a model to predict future values based on previously observed time series.

Time series forecasting methods can be divided into univariate and multivariate (Ampazis, 2015). Univariate time series forecasting concerns only with one variable that changes over time, that is the variable to be forecasted. Multivariate time series on the other hand, considers multiple time series simultaneously, i.e. more than one time-dependent variables where each variable depends on its past values and possible dependency on other variables.

With regards to the variables used in the forecast model, Hyndman & Athanasopoulos (2018) defined three types of forecast model: simple, explanatory, and mixed model. First, the simple model is similar to what is defined as univariate time series, i.e. it uses only information from the variable to be forecasted. This type of model extrapolates trend and seasonal patterns but does not aim to find out the factors that affect the patterns. The explanatory model uses predictor variables to help explain what contributes to the variation in the forecasted values, for instance temperature, marketing initiatives, and time of the day. The mixed model combines the explanatory predictors with the past observations of the variable (i.e. the lags) to be forecasted as the other predictors.

There are two families of techniques that are commonly used in time series forecasting. They are the classical time series forecasting techniques and machine learning techniques.

1. Classical Forecasting
   Classical forecasting has been widely used and generally serve as the benchmark methods for the development of a more complex and advanced forecasting techniques nowadays. It has the advantage of its simplicity and interpretability. Most of classical forecasting techniques such as the Moving Average and Holt-Winters Exponential Smoothing work without regressors. Some other techniques like SARIMAX have been developed to handle additional regressors as well. However, these techniques still require the data to satisfy a certain time-dependent structures.

2. Machine Learning Forecasting

   Machine learning techniques are favourable for their characteristics as universal approximators (Carbonneau et al., 2007). In general, machine learning problems can be categorized as follows (Bishop, 2006):

   - *Supervised learning*

     In supervised learning, the training data consists of input vectors along with their corresponding target values where we aim to either assign a discrete category to each input vectors, also known as *classification*, or estimate a continues target value for the inputs, known as *regression.*
   - *Unsupervised learning*

     Training data in unsupervised learning problems does not have any corresponding target values. The tasks in this category include discovering groups of similar samples, determining distribution within an input space, and projecting data from high-dimensional space into a lower dimension.
   - *Reinforcement learning*

     Reinforcement learning concerns with maximizing rewards by finding suitable actions in a given situation through a process of trial and error. Unlike supervised learning, the learning algorithm is not proved samples of optimal outputs.

   Forecasting problems can be defined as supervised learning regression problems, where the outcome is the estimated values of the variable we want to forecast. Techniques from this class have the ability to learn arbitrary function and the ability to control the learning process itself, making it useful in forecasting problems, including one with the presence of noise (Carbonneau et al., 2007). Machine learning forecasting allows more degrees of freedom for the model as it can handle complex interdependencies and non-linear relationships in the data. Furthermore, machine learning forecasting has the capability to handle a large number of variables as predictors which is often required in forecasting from multivariate time series data.

Table 2 summarizes the comparison of several aspects between classical and machine learning forecasting techniques (Carbonneau et al., 2007; Hyndman & Athanasopoulos, 2018; Makridakis et al., 2018; Kharfan & Chan, 2018).

Table 2 Comparison of Classical and Machine Learning Forecasting Techniques

	Classical Forecasting	**Machine Learning Forecasting**
Linearity	Linear model	Non-linear and complex relationships can be modeled
Time series type	Modeling requires the series to be stationary	Any time series can be analyzed
Data preprocessing	A lot of manipulation of time series may be required	Less preprocessing
Number of predictors	Single (historical data) or a few predictors	Unlimited predictors
Data requirements	Requires less amount of data	Requires larger amount of data
Interpretability	Provides insight and information through its parameter	May be difficult to interpret (black box)
Overfitting	No overfitting	Overfitting is possible

Based on the comparison, despite the difficulty to interpret and the possibility of overfitting, machine learning showed the potential to explore more relationships from more predictors. As there are a lot of external data that may be useful to predict the demand, machine learning forecasting is expected to be able to exploit the relationships of the features better than classical forecasting. Therefore, in this study, we decided to focus in utilizing machine learning techniques to build the demand forecasting model.

2.5 Machine Learning-Based Demand Forecasting

A large amount of data is available in various sources. Instead of using only the historical data of the demand of rental cars, there is a possibility of combining it with external data such as weather data, both historical and forecast, and calendar data. This data can provide further insight about what factors affect the demand in addition to the pattern of the historical demand. In addition to its power to provide a good prediction, machine learning based demand forecasting has the capability to model the relationships from this data, thus providing a good insight into the influencing factors of the demand.

Demand forecasting using machine learning techniques has been implemented in a large variety of industry, including healthcare (Whitt & Zhang, 2019), retail (Mupparaju et al., 2018; Aburto & Weber, 2007), e-commerce (Tugay & Ögüdücü, 2017; Ampazis, 2015), manufacturing (Carbonneau et al., 2007; Shahrabi et al., 2009, Zhou et al., 2015), tourism (Cankurt & Subasi, 2015; Chen & Wang, 2007), power (Qiu et al., 2015), and water distribution system (Herrera et al., 2010; Antunes et al., 2018). Some of these studies use only the historical demand as input variables, while some others consider a number of explanatory variables such as special events, calendar/holiday, weather, past demand observations (lags), date/time attributes, and time series components. Other than some variables that are available or extractable from every time series in general, some researches have also used context-specific variables, such as product quality for DVD rental demand (Ampazis, 2015) and product price relative to the micro-market price for supermarket products (Aburto & Weber, 2007).

Majority of researches related to machine learning-based demand forecasting either empirically tested the performance of various machine learning algorithms to find the best suited model to a certain case, compare machine learning techniques with classical forecasting techniques, or prove the effectiveness of their proposed model. We review these researches below and summarize them after to provide an overview of the techniques that are available, the data used in each domain application, and the findings about the model performance in general.

Herrera et al. (2010) compared several models to predict water demand in urban area. The results show that Support Vector Regression (SVR) is the most accurate model for the problem, closely followed by Multivariate Adaptive Regression Splines (MARS) and Projection Pursuit Regression (PPR) from Friedman and Stueltze (1981), and Random Forests. They concluded that predicting at hourly level with a medium size demand area, which is a city, was the suitable and sufficient to make management decisions.

Unlike Herrera et al. (2020) that built the prediction models for general period, **Ampazis (2015)** studied horizon-specific demand forecasting for the customer end of a multi-level supply chain. Ampazis (2015) approached the problem as a multivariate problem, combining the historical time series data with another explanatory variable, namely the product quality. They evaluated the effectiveness of Artificial Neural Network (ANN) and Support Vector Machines (SVM) on an online

DVD rental dataset for the peak Christmas season and showed that these machine learning techniques have the ability to lower the uncertainty of supply chain demand forecast.

Whitt & Zhang (2019) studied the forecast of daily arrival of patients in hospital emergency department as a multivariate problem as well, using exogenous variables such as calendar and weather variables. In this study, several models including linear regression, seasonal autoregressive integrated moving average with exogenous variables (SARIMAX), and the multilayer perceptron (MLP) were examined. They found that the highly structured time-series model (SARIMAX) performs better than the extremely flexible neural network model for a relatively small dataset, while regression comes second.

Carbonneau et al. (2007) studied machine learning based forecasting for distorted demand at the upper end of a supply chain. This distortion is a result of increasing order fluctuations as one moves upstream along the supply chain, also known as the bullwhip effect. They compared the machine learning techniques such as Neural Network (NN), Recurrent Neural Network (RNN) and SVM with the more traditional forecasting techniques, such as exponential smoothing, moving average, linear regression and Theta model for single time series. Exponential smoothing turns out as the best performing technique among the traditional ones and none of the ML technique outperforms it. However, with the increase on dimension of the training samples, the supervised learning model SVM is found to be superior.

Both Carbonneau et al. (2007) and Whitt & Zhang (2019) showed that the classical time series models outperform machine learning models for small datasets. However, Makridakis et al. (2018) showed a contrary result. **Makridakis et al. (2018)** evaluated statistical and machine learning techniques on forecasting using large amount of monthly time series data. They found that in their case, the statistical methods such as ARIMA and Holt Winter outperform the machine learning ones. They suggested that this may be an indication that the ML techniques are confused when attempting to optimize specific or heterogeneous patterns in the data, among other possible reasons. They then suggested several practical approaches to improve the forecast performance. One possibility is to cluster data into various categories, such as micro/macro and demographic, or into types of series, such as seasonal/non-seasonal, with trend/without trend, high/low level of randomness, and develop different models for different category.

Carbonneau et al. (2008) studied the distorted demand forecasting further by using the change in demand for each of the past number of periods and that for the current period as the input variables and predicted the total change over the next couple of periods. Contrary to Carbonneau et al. (2007), they found that with the addition of input variables the more advanced machine learning techniques (RNN and SVM) outperform the more traditional techniques (naïve, trend, moving average) but are not significantly better than the linear regression. **Cankurt and Subasi (2015)** also showed how addition of variables improve the performance of machine learning models. They developed multivariate forecasting model for tourism demand using MLP and SVR and showed that the inclusion of additional variables, such as date dimensions and statistics introducing the seasonal, cyclic, and trend components, significantly improve the accuracy.

Antunes et al. (2018) compared ARIMA and NN techniques in forecasting water demand focusing on the short-term demand. Various features are used for the prediction, including the past demand, day of the week, weather data such as temperature and rain, and seasonal events. The machine learning technique was shown to be reliable provided the data contains no significant anomalies.

Carbonneau et al. (2008), Cankurt and Subasi (2015), and Antunes et al. (2018) all showed that machine learning techniques perform better for a multivariate problem where explanatory variables are used in addition to the univariate time series. Besides for a larger dataset and additional explanatory features, some other research found machine learning techniques to result in better performance than the time series model when they are used to forecast long term-demand (Shahrabi et al., 2009), and when a specific algorithm is used to tune the parameters of the machine learning model (Chen & Wang, 2007).

Shahrabi et al. (2009) employed two machine learning techniques, namely ANN and SVM to forecast long-term demand for a car component supplier and compare them with the classic statistical methods (i.e. moving average and exponential smoothing). They found that ANN outperforms the others but moving average still shows a decent result considering. **Chen and Wang (2007)** conducted demand forecasts in tourism domain where they applied SVR technique and compared it with backpropagation NN and the autoregressive integrated moving average (ARIMA). They proposed the use of Genetic Algorithm to find optimal parameters of the SVR model and showed that SVR outperformed the other two models.

Mupparaju et al. (2018) compared moving average, factorization machines, gradient boosting, and several neural network models with different customization to predict demand of products in retail company. Their study showed that factorization machine, a model that is said to handle high-sparse data well, outperforms the others at the cost of significantly higher runtime.

Recently, related works have also seen an increasing trend in the development of novel hybrid and ensemble methods, i.e. the methods that combine multiple learning algorithm to obtain a better predictive performance. **Aburto and Weber (2007)** developed a hybrid intelligent system combining ARIMA models and neural networks for demand forecasting in a retail industry, specifically a supermarket. In their hybrid model, Aburto and Weber (2007) modeled the original time series using SARIMAX and the forecast error as another time series using NN. The study showed that NN outperformed ARIMA and the hybrid model performed better than both of them independently. Qiu et al. (2014) presented a novel approach using ensemble deep learning method consisting of deep belief network (DBN) and SVR in predicting load demand in power industry, while Zhang (2003) proposed a hybrid of ARIMA and ANN. Similarly, both studies showed that their proposed hybrid methods outperform each comprising method individually.

Zhou et al. (2015) proposed a two-step dynamic inventory forecasting for large manufacturing. In the first step, they applied ensemble of six machine learning techniques to forecast the demand, namely Linear Regression, NN, Regression Tree, Gradient Boosting Regression Tree, SVM, and Gaussian Process. After that they consider the characteristics of inventory, such as long-term trend, seasonality, and events factors to get contemporary sales. The final forecasting result is the average from the result of these two steps. They stated that this approach succeeded in acquiring better prediction accuracy and has better interpretability to the analysis results.

Tugay and Ögüdücü (2017) conducted demand prediction on an e-commerce web site on which the same products are sold by different sellers at different price. They compared regression techniques with stacked generalization (stacking ensemble learning) to predict the demand and showed that machine learning methods perform almost as good as the other method.

Table 3 Summary of Machine Learning Forecasting Literature

Author	Industry	Models	Features	Findings
Whitt & Zhang (2019)	Healthcare (hospital ED)	Linear Regression, SARIMAX, MLP	Historical data, calendar, weather	SARIMAX outperforms others on small dataset
Mupparaju et al. (2018)	Retail	Moving Average, Factorization Machine, Gradient Boosting, NN	Historical data	Factorization machine outperforms others in handling high-sparse data
Carbonneau et al. (2007)	Manufacturing	NN, RNN, SVM, Exponential Smoothing, Moving Average, Linear Regression, Theta	Historical data	Exponential smoothing outperforms others, but SVM performs best for larger dataset
Carbonneau et al. (2008)	Manufacturing	RNN, SVM, Naive, Trend, Moving Average, Linear Regression	Historical data	RNN & SVM outperform others but are not significantly better than linear regression
Shahrabi et al. (2009)	Manufacturing (car component)	ANN, SVM, Moving Average, Exponential Smoothing	Historical data	ANN outperforms others but MA still performs well
Tugay & Ögüdücü (2017)	E-commerce	Linear Regression, DT, RF, Gradient Boosting, Stacked Generalization	Historical data	Stacked generalization performs as good as single classifiers
Zhang (2003)		ARIMA, NN, Hybrid ARIMA-NN	Historical data	Hybrid model outperforms each individually
Ampazis (2015)	E-commerce (online DVD rental)	ANN, SVM	Historical data, product quality	Multivariate ML models lower the uncertainty barrier of demand forecast
Aburto & Weber (2007)	Retail (supermarket)	SARIMAX, NN, Hybrid SARIMAX-NN	Lags (pas sales), special days (payment day, holiday, summer), price relative to micro-market	Hybrid model outperforms each individually
Qiu et al. (2014)	Power	Ensemble of DBN and SVR, SVR, DBN, feedforward NN, ensemble feedforward NN	Lags (demand of last 24 hours)	Ensemble of DBN and SVR outperforms others
Cankurt & Subasi (2015)	Tourism	MLP, SVR	Date attributes, time series components (seasonal, cyclic, trend)	Auxiliary variables improve the accuracy
Chen & Wang (2007)	Tourism	GA-SVR, NN, ARIMA	Historical data	SVR outperforms others
Herrera et al. (2010)	Water distribution system	ANN, SVR, MARS, PPR, RF	Historical data, weather (temperature, wind velocity, rain, atmospheric pressure), day of the week	SVR outperforms others
Antunes et al. (2018)	Water distribution system	RF, SVR, KNN, NN, ARIMA	Lags (demand previous 2 weeks), weekday, weekend, weather (temperature, rain), seasonal events	Machine learning models outperform ARIMA
Zhou et al. (2015)	Manufacturing	Ensemble of Linear Regression, NN, Regression Tree, Gradient Boosting, SVM, Gaussian Process	Historical data (1st step), long-term trend, seasonality, events (2nd step)	Two step model provides better accuracy and interpretability
Makridakis et al. (2018)		Naive, Random Walk, Exponential Smoothing, Theta, ARIMA, MLP, Bayesian NN, Generalized Regression NN, KNN, CART, SVR, Gaussian Process, RNN, LSTM		Traditional statistical techniques outperform ML techniques

Table 3 summarizes the works related to machine learning demand forecasting. A large variety of methods have been used for demand forecasting. Instead of limiting to only supervised machine learning techniques, most of these researches have also included classical time series techniques in their studies. Interestingly, some of the researches showed that machine learning can outperform classical forecasting techniques (Carbonneau et al., 2007; Carbonneau et al., 2008; Antunes et al., 2018; Shahrabi et al., 2009; Chen and Wang, 2007; Mupparaju et al., 2018), while some other found classical forecasting techniques to outperform machine learning forecasting, particularly in univariate time series forecasting problem (Carbonneau et al., 2007) and in the case where small dataset is used (Carbonneau et al., 2007; Whitt & Zhang, 2019).

The fact that a simpler method like ARIMA can outperform machine learning techniques can also be explained by the utilization of criterion such as AIC and other optimization processes in ARIMA (Makridakis et al., 2018). They enable effective automatic model selection and parameterization while avoiding or minimizing overfitting, which is often an issue in machine learning techniques. However, despite their result, Makridakis et al., (2018) acknowledged that the findings may be different and that machine learning techniques can offer significant advantage over statistical methods in the case of nonlinear components being present or if other factors dominate the data.

Furthermore, the results of these previous works are case-specific which makes the performance of the techniques depend on the characteristics of the data used, such as the size of the dataset, the presence of significant anomalies in the dataset, and the sparsity of the dataset. Despite the tendency of specificity of machine learning-based demand forecasting performance, Petropoulos et al. (2014) sought to define how types of data influence forecasting accuracy. Based on their investigation involving 14 forecasting methods and their combinations, and seven different time series features, the following are the conclusion of their study:

- For fast-moving data, cycle and randomness have the biggest (negative) impact, followed by forecasting horizon.
- For intermittent data, interval between demand has bigger (negative) effect than the coefficient of variation
- Regardless of the type of data, increasing the length of the data has a small positive effect on the accuracy of the forecast.

Taking everything into account, there is no exact answer on which forecasting techniques (i.e. classical or machine learning) will result in a better performance accuracy. Classical time series techniques are still valuable for demand forecasting despite the availability of many advanced techniques. Therefore, in this study, we will compare machine learning models to classical time series models as our benchmark.

2.6 Uncertainty in Forecasting

Uncertainty is one of the important aspects of forecast quality (Murphy, 1993). Murphy (1993) defined uncertainty in forecast as the variability of observations as described by distribution of observations. However, among several concerns related to machine learning forecasting and its way forward, Makridakis et al. (2018) named the capability to specify uncertainty or intervals as one aspect that is rarely included in the research agenda in the field. In classification, uncertainty can be captured by the probability. However, it is different in the case of regression.

Brando et al. (2018) proposed several categories of uncertainty, as follows:

1. Epistemic uncertainty, if the noise applies to the model parameters.
2. Aleatoric uncertainty, if the noise occurs directly in the output given the input.
    a. Homoscedastic uncertainty, when the noise is constant for all outputs (i.e. measurement errors).
    b. Heteroscedastic uncertainty, when the noise of outputs depends explicitly on the specific input (input-dependent).

Brando et al. (2018) compared the performance of different combination of predictors and uncertainty treatments. They found that heteroscedastic solutions outperform the others. They concluded that taking the variability of the output into account improves the performance of the model.

One way to demonstrate uncertainty of the forecast output is to show it in the form of prediction intervals as they generalize point forecasts to represent and incorporate uncertainty (Hansen, 2006). Prediction interval differs from confidence interval (Heskes, 1997). Confidence interval guarantees with a certain confidence that the mean response lies within the interval by taking into account the standard error of the fit. Prediction interval exhibits with a certain confidence that the new response is between the interval by considering the standard error of the prediction. Thus, confidence interval shows the uncertainty of the model parameters for the whole population (i.e. epistemic uncertainty in Brando et al. (2018)) while prediction interval represents the uncertainty of the estimated values/observations (i.e. aleatoric uncertainty in Brando et al. (2018)).

Makridakis et al. (2018) stated that a lot of researches propose simulating the intervals by generating multiple future sample paths iteratively, but this method would only derive the forecast distribution empirically and not analytically, raising many doubts about its quality. In the following part, we discuss a number of techniques used for generating prediction intervals in previous works.

One technique to measure uncertainty communicated through prediction interval is the quantile regression. Quantile regression estimates the value of the forecasted variable at two percentiles points separated as far as the desired interval. For example, to produce a 90% prediction interval, one can predict the value at 5^{th} and 95^{th} percentiles. Quantile regression-based forecasting method has been implemented for probabilistic or uncertainty in forecasting. Some examples are Mayr et al. (2012), Taieb et al. (2016), and Ziel (2018). Taieb et al. (2016) demonstrated forecasting of electricity consumption for a disaggregated level (i.e. per household). Due to the higher uncertainty expected for a lower level forecast, they proposed the use of probabilistic forecasting method using a non-parametric approach, which is the quantile regression. Unlike parametric approach that uses an estimation of distribution parameter from the data, a non-parametric approach avoids distributional assumptions by estimating the predictive distribution with assumptions on the shape of the distribution, such as smoothness. Mayr et al. (2012) previously demonstrated the use of quantile boosting to generate prediction intervals. This method directly models the borders of the prediction intervals by additive quantile regression, estimated by boosting. The results showed that the quantile regression approach performs better on disaggregated electricity demand, while the traditional approach assuming normality performs better on the aggregated data.

Using statistical models, one can either create prediction interval in frequentist or Bayesian setting. Van Hinsbergen et al. (2009) demonstrated short-term prediction of travel time by combining neural networks using Bayesian inference theory. With Gaussian assumptions for the output distribution and the weights, the proposed method allows accurate estimation of confidence intervals for the

predictions, in fact 97.4% of the actual data fall within the resulted 95% confidence intervals. Gopi et al. (2013) proposed a Bayesian support vector regression to predict traffic speed and providing a measure of uncertainty through error bars, which cannot be provided using the standard SVR. They evaluated the efficiency of BSVR by comparing sensitivity and specificity of prediction errors with variations of MAPE thresholds. Some other studies also proposed probabilistic forecasting with the parametric approach using various techniques to estimate the distribution parameter, such as by estimating conditional mean and conditional variance (Wijaya et al., 2015) or using generalized logit-normal distributions (Pinson, 2012).

2.7 CHAPTER SUMMARY

This chapter discussed the theory and previous works related to the study, including background knowledge about the domain. We have discussed the related works for car rental demand forecasting as well, which introduce us to some factors that have been used or said to be important for car rental demand prediction. Next, we found that both classical time series and machine learning techniques have their own strength and weaknesses. While machine learning techniques are preferable for this project due to their advantages, previous works have shown that classical time series method can outperform machine learning depending on the case it is applied for, thus it is important to include them in the analysis. Therefore, in this study, we will apply machine learning techniques as well as classical time series as a benchmark for the performances of the machine learning models. Lastly, we have seen several possible approaches to generate forecast intervals either with or without distributional assumption. However, the performance of each approach varied from one case to the other and they cannot be easily compared as different measures are used. On that account, in this study we experiment with several approaches with different characteristics and empirically tested their performances to find the most suitable one for our case.

3 METHODS

Following the research outline in Section 1.4 and our findings in Chapter 2, in this chapter we break down the research methods to be used in each research phase, including the methods, approaches, models, measures, and tools for business understanding, data understanding, data preparation, and modelling phases. For modelling phase, we first describe the general steps taken to build each model as well as our proposed approach, and define several models to analyze (Section 3.3). Then, the following chapters explain in more detail the methods used for model selection (Section 3.4) and model evaluation (Section 3.5). After that, we describe several approaches that we use to build prediction interval and the measures to be used to compare each approach (Section 3.6). Finally, Section 3.7 explains the choice of tools we use for the development of the artifact.

3.1 BUSINESS AND DATA UNDERSTANDING

The first two phases conducted in the study are business and data understanding phases. In these phases, we study the background of the problem and determine business objectives for which the solution will be developed. After that, we assess the situation to define the requirements, assumptions, and constraints of the prediction model. This assessment is done iteratively with exploratory data analysis as we collect and explore the data to gain more understanding.

In addition to determining the extent of the case and project, these phases include information gathering for the candidate features for the demand prediction model through domain analysis. Domain analysis is an important step to develop prediction instruments that have sufficient predictive power (van der Spoel et al., 2016). We carried out domain analysis to identify the initial features with the following methods:

1. Literature Review
   Even though there are no literature that specifically predict or discuss the factors that affect the demand of replacement cars, studies related to the demand of rental cars can provide relevant information for our specific problem context. Therefore, a comprehensive review of previous works related to the car rental demand forecasting is performed to obtain a list of potential variables to use as predictors in our model.
2. Interview
   Interview is a commonly used means of capturing domain intelligence. Information gathering through interview can capture the domain knowledge that is not available in the scientific literature or one that may be case-specific. In general, there are three types of interview based on the degree to which questions are pre-defined and the flexibility to proceed with questions emerging from the dialogue, namely unstructured, semi-structured, and structured (DiCicco-Bloom & Crabtree, 2006). In this study, we conducted multiple unstructured and semi-structured interviews with the domain experts within the company. Unstructured interviews allow us to collect background data when little to no information is known about the topic, while semi-structured interviews let us pre-define questions in advance and adapt the questions as the conversation goes. A semi-structured interview is conducted with a road patrol that has years of experience in handling breakdown incidents, while more open unstructured interviews are carried out with the stakeholder of the project and a data scientist of a related project.

In the development of the prediction model, quantitative methods (i.e. statistical and machine learning techniques) are then used to test whether the features proposed in the domain analysis

indeed affect the outcome of the prediction (van der Spoel et al., 2016). Specifically, we carry out this part of the research using exploratory analysis of the data and feature selection, which will be explained in a later section. For the data exploration, we visualize the data using Tableau, a sophisticated data visualization software specialized in business intelligence. We choose to work in Tableau mainly for its ease of use and its ability to extract data from various data sources. We look into the patterns of variables over time and collect basic facts from the data. In addition, we describe the statistics and distribution of the data, the correlation between variables, as well as the quality of the data.

3.2 DATA PREPARATION

3.2.1 Data Restructuring

Raw business data are oftentimes not available in the format that is ready for time series analysis. Thus, they require restructuring in such a way that would fulfill the objective of the analysis.

Time interval

Before training the model, we prepared a dataset in time series format to provide a demand estimation per time point. The demand prediction model can take any level of time intervals, whether it is hourly, daily, weekly, and so on, depending on the available data and the planning horizon. In this study, we restructured historical records of rental car orders into time series with constant intervals. We aggregated these records to get the number of orders per day, as required for the planning activity.

Spatial level

Additionally, to investigate the degree of predictability of the replacement cars demand, the prediction models were developed with several granularities with regards to spatial aggregation. The order records were specified and grouped by the locations where the breakdown incidents happen or where the rental cars being requested. The following are the different levels of detail that are taken into account in this study.

1. High-level: the Netherlands
   The high-level model is a demand prediction model for the whole Netherlands at one point of time. This is an essential level to investigate the feasibility of the demand of replacement cars prediction in the first place.
2. Mid-level: provinces and ANWB work area
   We expect the prediction accuracy for the low level to be of lower standard than the high level. Therefore, it is in our interest to study the demand at an intermediate level as well to thoroughly inspect the effect of more granularity. In this level, we considered two different spatial aggregation levels. First, records of orders were aggregated per province. There are 12 provinces in the Netherlands which means there are 12 point observations to be predicted per time point. In addition, the data were aggregated per ANWB work area which has 33 divisions of area.
3. Low-level: Logicx car rental pick-up locations
   The low-level model predicted the demand of replacement cars per rental locations, are expected to be utilizable as a direct input to plan the right number of cars required at the right location. There are 85 locations located all over the Netherlands.

Product type

Further breakdown of the model is to consider the type of the demand. The car group has been identified as one of the main factors that define the demand estimation function (Fink & Reiners, 2006). Depending on the availability of the data, the demand of replacement cars will be predicted per category of classes of cars and/or per type of customer requesting the car (either B2B or B2C). This is due to the high importance of providing cars from the same category and brand for B2B customers as well as the due consequences the company has to suffer for providing cars from higher classes.

3.2.2 Feature Engineering

Feature engineering is the process of constructing relevant features from raw data based on understanding of domain knowledge and data exploration. It is essential for the success of many machine learning tasks as oftentimes the raw data is not in a from that can be easily learned by the algorithms. A large number of learning algorithms are generic and not designed for a specific purpose, while features are usually domain specific (Domingos, 2012). The features need to take into account the characteristics and limitations of the algorithms to be able to provide useful insights into different aspects of the data (Katz et al., 2016). For example, some algorithms can handle binary variables better than categorical variables with more than two values.

Feature engineering is difficult and time-consuming as it requires domain knowledge and a lot of trial and error to design the features. It is a major part of the iterative process of machine learning. After building the dataset, training the model, and analyzing the results, one can either modify the data or the model and repeat the process to produce a better model. Even with the automation of feature engineering process, for instance by automatically generating large number of candidate features which will then be fed into a feature selection algorithm, in the end the knowledge incorporated into constructing hand-crafted features is still irreplaceable (Domingos, 2012).

Feature engineering can be done by deriving new features from existing features (e.g. from date to day of the week, month, and year), transforming a variable into another format (e.g. from the actual month value to a polar coordinates representation of the month), combining the values of two variables (e.g. ratio and difference of two variables), and creating dummy variables for each value of categorical variables. Most importantly, in classic time series analysis, the algorithms use the previous values of the target output from a certain period of time/rolling window, known as lags, to estimate the future values. Lags can be manually constructed as features in the feature engineering process to transform classic time series problem into a supervised learning problem. In this study, we create a lot of features that we think may be relevant or may improve the learning process even the slightest. We then take measures to prevent overfitting or more noise because of the possibly irrelevant features using the feature selection afterwards.

3.2.3 Data Preprocessing and Dimensionality Reduction

The performance of machine learning algorithms usually depends on the quality of the data, that is to say high quality data will lead to high quality results and faster processing (Kotsiantis et al., 2006). Furthermore, dataset with large number of features (high dimensionality) means more possibility that some of the features are trivial for the model, leading to a potentially unnecessary higher complexity and slow process. The following are preprocessing steps we carried out in this study:

1. Missing Value

Real world data are often incomplete. However, most of existing machine learning algorithms are developed with the assumption that there are no missing values in the data (Somasundaram & Nedunchezhian, 2011). Therefore, missing values can lead to a misclassification or wrong prediction. In this study, we take the following measures to handle missing values in the dataset:

a. Remove feature with large percentage of missing value
   If a feature has a large percentage of missing value, we will remove the feature from the dataset as it will not be reliable for analysis.
b. Imputation
   Missing value will be filled using logical reasoning or approximation from other data rows whenever possible. Depending on the feature that contains the missing value, imputation will be done by either deducting the value based on values of other columns in the row, or taking the value (or mean/median/mode) from the other rows in the dataset (similar case imputation).
c. Remove data row
   If imputation is not possible to do (e.g. it is expected to result in a different effect on the model), and the missing value percentage is relatively low that it will not cause a major data loss, we will exclude the rows with missing value from the analysis.

2. Correlations

High correlations between features means these features likely have similar trends and information. Highly correlated features can affect models differently. Linear models with multicollinearity can have a numerically varying and unstable results (Goldberger & Goldberger, 1991). Interactions between different features that can be analyzed by a tree-based model can be overshadowed by the redundancy of the features.

According to Hall (2000), a good feature subset is a subset of features that are highly correlated with the target value but uncorrelated with each other. Therefore, Pearson correlation (ρ) will be applied to measure the linear dependency between features and an input feature that is highly correlated with the other input feature(s) in the dataset will be removed. Correlation coefficient can also be used to reduce the dimensionality of the dataset by removing features that have low correlation to the output of the model. However, since correlation coefficient does not measure non-linear relationships and an interaction of a feature with other feature(s) can be an important predictor even though the feature alone has a low correlation to the target, we do not use correlations to eliminate features that have low correlation with the target.

3. Standardization and Normalization

Lastly, before we start building the model, feature scaling will be carried out as some machine learning algorithms can produce better performance and learn more effectively when the data attributes are of the same scale or distribution (Kotsiantis et al., 2006). Furthermore, by having the same scale for all attributes, we can directly compare the coefficient values in the linear models. Standardization and normalization are two widely used methods of data scaling (Kotsiantis et al., 2006). Normalization rescales the data from the original scale to a range of 0 to 1, while standardization rescales the distribution by centering the data (i.e. it has a zero mean and a unit variance) assuming that the data are normally distributed.

3.3 MODEL BUILDING

After preparing the dataset, the next process is to train the model on the data. In this study, we compare several machine learning and classical time series techniques. Predicting demand of replacement car per day can be categorized as a regression problem in machine learning taxonomy since the expected output is a continuous value at each point of prediction. In addition to dealing with regression problem, clustering and classification have been demonstrated to have the capacity to support forecasting (Kharfan & Chan, 2018). Bandara et al., 2017 proposed to group time series based on natural grouping with domain knowledge or automatic mechanism such as time series clustering, and build different models for different groups to avoid harm in the overall accuracy because of the use of one general model for heterogeneous series (Bandara et al., 2017).

In car rental domain, companies often experience outlier demand, for instance, because of a surge in demand because of special events (Geraghty & Johnson, 1997). Therefore, in this study, we buit the model with two different approaches. The first approach is a general approach for building, validating, and evaluating machine learning model. We employed this approach to investigate the feasibility of replacement cars demand prediction for different aggregation levels. For the second approach, we proposed to group the time series into outliers and non-outliers dataset and build different models for both groups. This approach consists of three main steps which are outlier identification, classification, and prediction. We evaluated this approach by applying it for the high level prediction and comparing it to the general approach. The steps for each approach is illustrated in Figure 4.

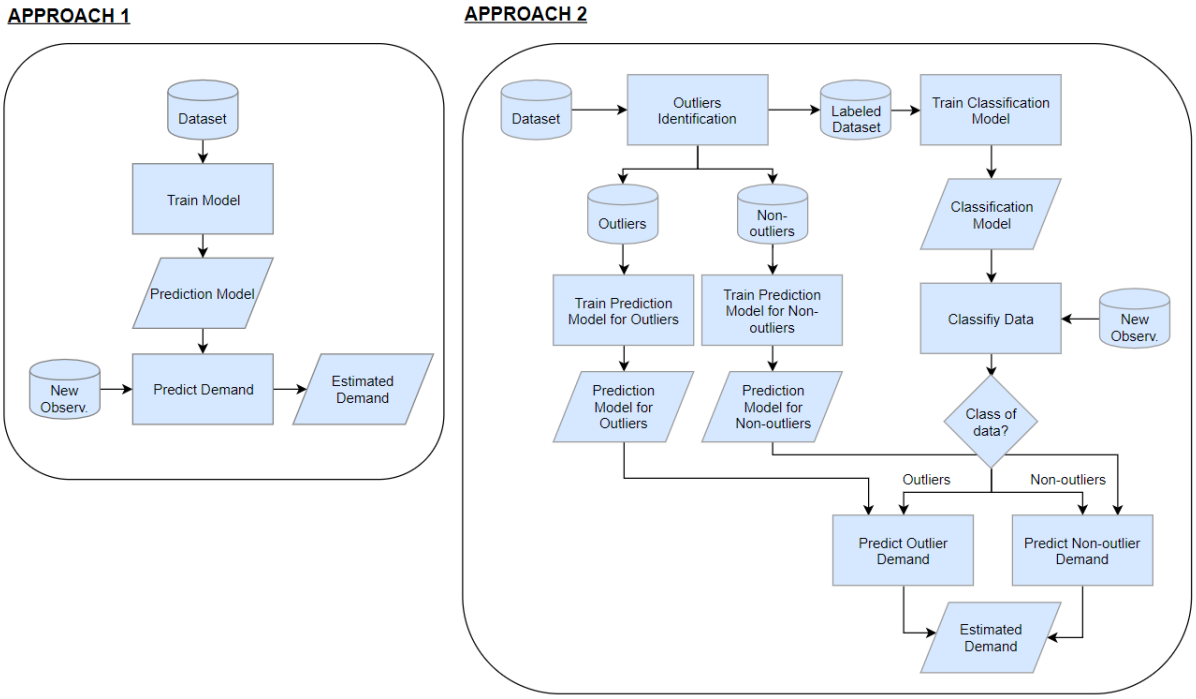


Figure 4 Model building approaches: (1) General approach, (2) Proposed approach

As discussed in Chapter 2, previous works have shown that there is no definite rule of which model performs the best or better than the others. A more sophisticated algorithm does not always produce better performance. Model performance highly depends on the problem and the dataset used in the project. Time series data are also generally more limited since there will only be as many instances as the time series interval in the dataset. With the limited amount of data, we opted to start from the simplest models first. We frame the time series problem as a supervised learning

problem, taking into account various explanatory variables, both from historical and external data. To find the most suitable model for the specific problem, we compared several regression techniques as well as classical time series techniques as the benchmark models, as follows.

Linear Regression

Linear regression, also known as the ordinary least squares, is the simplest linear model for regression. It comprises a linear combination of the input variables to estimate the target value. The key characteristic of this model is that it is a function of regression coefficients that minimize the sum of squares differences between the actual and estimated values.

There are a number of variants of linear regression intended to improve its performance by introducing regularization.

1. Lasso Regression
   Lasso regression applies L1 regularization in the linear regression model. It estimates sparse coefficients and tends to prefer solutions with less parameter values. It imposes a penalty on the sum of the absolute values of the regression coefficients. Thus, the algorithm lowers the coefficient values of the unimportant features or set them to zeros, completely eliminating the features.
2. Ridge Regression
   Ridge regression is a linear regression with L2 regularization. It imposes a penalty term on the size of coefficients (i.e. the sum of squared values of the regression coefficient) and lower the coefficient values when necessary, making the coefficients more stable and robust to collinearity. Thus, it can reduce the impact of the less important features.

Support Vector Regression

Support Vector Regression (SVR) is an implementation of Support Vector Machines (SVM) for regression problems. SVM are based on the structural risk minimization principle that minimizes the true error on an unseen and randomly selected test samples, rather than the error for the currently seen samples, as implemented by models with empirical risk minimization like linear regression. It does so by projecting the data into a higher dimensional space using kernel functions and minimize the error margin. The model produced by SVM depends only on the kernel function evaluated on a subset of the training data. Therefore, SVM is effective for large scale problems. Different kernel functions can be used, such as the linear kernel and the Radial Basis Function (RBF) kernel for non-linear problem. The key property of this technique is that it uses a convex optimization problem to determine the model parameters, thus it guarantees that any local solution is also the global minimum.

Random Forests

Random forest is an ensemble technique where a number of decision trees are trained on various subsamples of the dataset. Furthermore, it splits the node based on the best split among a random subset of the features in place of the best split among all features. It then takes the averaged prediction of the individual trees. On that account, it overcomes the pitfall of a single decision tree, that is the instability and high variance in the result (i.e. a small change in the data can often cause a large change in the final model).

Gradient Boosting

Another variant of the ensemble technique known as boosting involves training multiple weak learners sequentially to create a strong learner. The error function used to train one model depends on the performance of the previous models. Gradient boosting uses multiple decision trees as learners. It builds the first model and calculates an arbitrary differentiable loss function. It then calculates the residual of the loss function using Gradient Descent method and uses this as the new target variable for the next iteration, resulting in an improved model compared to the first model. A more advanced and efficient implementation of Gradient Boosting considers the systems optimization as well to provide a faster and scalable tree boosting system. It is known as XGBoost or Extreme Gradient Boosting.

Classical Time Series

For classical time series techniques, we chose a number of techniques that are widely used and have distinct characteristics to represent this category well. They are ARIMA and its variations, Holt-Winters Exponential Smoothing, and a state-of-the-art time series model Prophet.

ARIMA and variations

ARIMA (Autoregressive Integrated Moving Average) is a linear model where the predictors in the linear equation are the lags of the stationary time series and/or the lags of the forecast errors. The time series can be made stationary by differencing. *Autoregressive* refers to, *Moving Average* refers to the lags of the forecast errors, while Integrated refers to the differenced series which is an integrated version of the stationary series. The notation ARIMA(p,d, q) is commonly used, where:

- p is the number of lags of the stationarized series (*Autoregressive*)
- d is the number of nonseasonal differenced required to make the time series stationary *(Integrated*)
- q is the number of lagged forecast errors (*Moving Average*)

There are several variations of ARIMA. SARIMA (Seasonal Autoregressive Integrated Moving Average) extends ARIMA by explicitly defining a seasonal component. SARIMAX (Seasonal Autoregressive Integrated Moving Average with eXogenous regressors) allows modeling of exogenous variables in addition to univariate time series. SARIMAX(p,d,q)(P,D,Q)m is the notation used to specify the hyperparameters of SARIMAX where P, D, Q, and m are the seasonal elements for autoregressive order, difference order, moving average order, and the number of time steps for a seasonal period, respectively.

Holt-Winters Exponential smoothing

Exponential smoothing calculated the weighted averages of past observations as the forecasts, where the weight is exponentially decreasing as the observations move further in the past. It has an explicit modelling of error, trend, and seasonality.

Prophet

Prophet (Taylor & Letham, 2018) is a model for forecasting time series data based on an additive regression model that models time series as a sum of different components, including non-linear trend, various seasonal components (including yearly, weekly, and daily), and holiday. It can model exogenous variables as extra-regressors as well. The procedure consists of four main components:

- A piecewise logistic or linear growth curve trend with automatic detection and selection of changepoints (i.e. the point where the growth rate is allowed to change) from the data.
- A yearly seasonality modeled using Fourier series.

- A weekly seasonality created using dummy variables.
- A built-in national holidays or user-provided list of important holidays.

3.4 Model Selection

Building prediction model does not end with training the algorithm on the dataset. Model selection is an important step to produce the optimal model for a problem. Model selection can have three different meanings (Luo, 2016):

1. Selection of an effective algorithm for a given problem.
2. Selection of effective hyper-parameter values for a given algorithm.
3. Feature selection.

To avoid selection bias and overfitting in a given modelling problem, in general, cross-validation is used. Cross-validation is a resampling procedure that creates combinations of training-test sets to evaluate machine learning models and give a more accurate estimation of how the model will generalize to an unseen dataset. However, in time series problem, the conventional cross-validation cannot be used due to the structure of the data. First, the data is time sensitive and changing the order of the data will affect the training and the results (Antunes et al., 2018). Secondly, cross-validation technique has a fundamental assumption that the data are independent and identically distributed (Arlot & Celisse, 2010). As the lagged values of the time series are used both as the input variables and the reference data, the training and test sets are not statistically independent if chosen randomly and the time series might be produced by a process that changes over time (Bergmeir & Benitez, 2012) hence violating the fundamental assumptions of cross-validation.

For this reason, we first split the dataset into training and test sets by respecting the order of the observations. Then for model selection, we use a variation of k-fold cross-validation designed for time series data on the whole training set and evaluate the selected model on the held-out test set. Time series cross validation (TSCV) does not shuffle the dataset. Instead, for the k-th split, it assigns the first k folds as train set and the k+1-th fold as test set. The training set in the succeeding split is a superset of the previous training sets. The schema of the procedure is depicted in Figure 5.

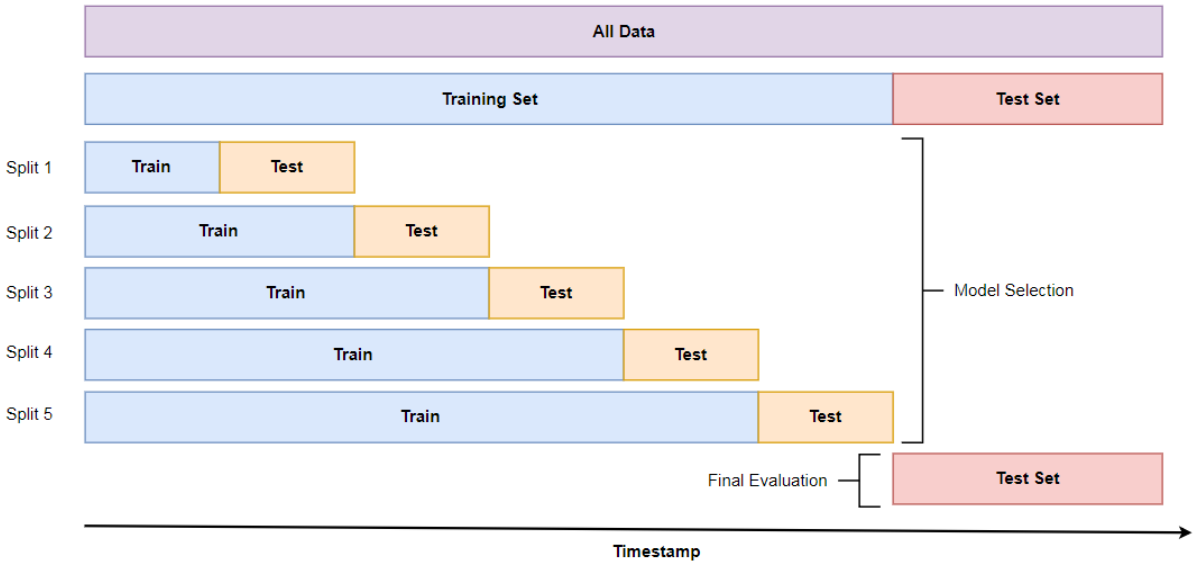


Figure 5 Time series cross-validation

3.4.1 Hyper-parameter Tuning

There are a number of techniques to tune the hyper-parameter values. The most common one is using grid-search where we define several potential values of the parameters. Grid-search will run every combinations of hyper-parameter values and returns the values that gives the best average performance from the cross-validation. The drawback of grid-search is that it involves human intuition to define and redefine the search area. To overcome this issue, an automated global optimization method for model selection, the Bayesian optimization is widely utilized recently (Liu et al., 2017; Ru et al., 2017; Kandasamy et al., 2019). Snoek et al. (2012) introduced practical Bayesian optimization for hyperparameter selection of machine learning algorithms. This method uses the Bayes' theorem by taking into account the result of the previous selected values of the hyperparameters (prior belief) to select the value for the next iteration. However, Bayesian optimization can get stuck with many iterations without improvement or result in a local minimum of the objective function (Brochu et al., 2010). Considering the trade-off between the above-mentioned methods, in this research grid-search is mainly used, while Bayesian optimization can be an alternative for algorithms that require significant effort and time for tuning due to the large number of hyper-parameters that needs to be tuned.

3.4.2 Feature Selection

In machine learning, there is a phenomenon known as the curse of dimensionality, which shows that generalizing correctly becomes more difficult as the dimensionality (i.e. the number of features) increases. Adding more features may appear harmless as they provide no new information at worst, but the curse of dimensionality may outweigh the benefits (Domingos, 2012). Some features may only introduce noise instead of improving the predictive power of a model and increasing features means increasing the processing power and the amount of training data needed.

There are two main approaches to reduce the number of features, namely feature selection and dimensionality reduction. Dimensionality reduction creates new combinations of all the features to use as the inputs of the model. On the other hand, feature selection works by including and excluding features in the data without modifying them, which will be more useful as insights for the business domain. Some features may be irrelevant individually but yield a predictive power when trained together with other features. Therefore, rather than merely evaluating feature importance or the relationship of each input feature with the target, we employ feature selection method that selects the best subset of features, specifically the Recursive Feature Elimination (RFE).

RFE is one technique that examines feature subset ranking criterion instead of feature ranking criterion individually. The iterative procedure of recursive feature elimination is as follows (Guyon et al., 2002):

1. Train the model, that is, the model optimizes the weights of each feature with respect to its cost function.
2. Compute the ranking criterion (i.e. the effect of removing one feature at a time on the objective function) for all features.
3. Remove the feature(s) with the lowest ranking criterion.
4. Repeat until the desired number of features is reached.

In addition, we use RFE with cross-validation (RFECV) to select the best number of features in the final subset.

3.5 MODEL EVALUATION

Model evaluation in this study comes in twofold. First, we evaluate different models in the test (evaluation) set and compare their performances to find the best performing model for the problem. Next, we evaluate the performance of the best model for demand forecasting in the future, that is, we run a forecast for one week period that includes multi-step forecasting and uses weather forecast data instead of the actual weather data.

3.5.1 Performance Measures

The performances of the trained models are evaluated using several error metrics. In particular, there are three metrics that are widely used to evaluate point observations in regression problem over the years (Botchkarev, 2018):

1. Mean Squared Error (MSE) and Root Mean Squared Error (RMSE)
   MSE is the most widely used error metrics. It measures the squared error point distance and is scale dependent. RMSE is the root of the MSE which can be used to keep the dimension of the metrics to the actual values. In other words, it shows the average distance. With e_i denoting the error or difference between the actual and predicted value for the i-th observation, MSE and RMSE are formulated as follows.

$$MSE = \frac{\sum_{i=1}^{n} {e_i}^2}{n}$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{\sum_{i=1}^{n} {e_i}^2}{n}}$$

2. Mean Absolute Error (MAE)
   MAE is another scale-dependent metrics. It differs from the MSE in terms of the point distance it measures, which is the absolute error.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |e_i|$$

3. Mean Absolute Percentage Error (MAPE)
   MAPE normalized the absolute error measure by actual values. Due to the intuitive explanation it offers, MAPE is the most commonly used measure for assessing forecasts in organisations and has seen an increasing trend of use (Botchkarev, 2018).

$$MAPE = \frac{100}{n}\sum_{i}^{n} \frac{|e_i|}{|A_i|}$$

Besides the three most commonly used metrics, we measure the performance using the median absolute error (Median AE) because of its robustness to outliers and the coefficient of determination (R^2). Median AE calculates the median of all absolute differences between the target and the prediction. R^2 measures the proportion of explained variance, that is the amount of variance in the target variable explained by the factors included in the model. There are several different formulas to calculate R^2. In this study we implemented the formalization in Kvålseth (1985) as follows.

$$R^2 = 1 - \frac{\sum_{i=1}^{n} e_i^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

3.6 PREDICTION INTERVAL

As discussed in Section 2.6, allowing estimation of the uncertainty for the point forecasts can bring machine learning method one step further in the application. Specifically, prediction interval can provide an extended insight for the business domain, such as the trade-off between the range of interval and the granularity of prediction or to find out given the highest possible accuracy, how far lower or higher could the actual demand deviates. In this study, we consider several techniques to generate prediction interval:

1. Assuming normality, one step ahead prediction interval (constant variance) is calculated as follows.

   $$Prediction\ interval = \hat{y} \pm Z\ score\ \times\ \hat{\sigma}$$

   where
   $\hat{y}$ = forecast value
   $\hat{\sigma}$ = estimate of the standard deviation of the forecast distribution
   $Z\ score$ = multiplier that depends on the coverage probability (e.g. 1.28 for 80% intervals, 1.96 for 95% intervals)

2. Assuming normality, calculate prediction interval with non-constant variance by fitting input variables not only to target variable, but also to error values.
3. Perform quantile regression at low and high quantile.

We compare which technique is the most suitable for the problem by evaluating the resulted intervals based on the concept of accuracy and precision. The more actual values fall within the interval, the more accurate the interval is. Whereas for the precision, the narrower the interval, the more precise it is. These measures are formalized as follows.

- Prediction Interval Coverage Probability (PICP)

$$PICP = \frac{1}{N}\sum_{i=1}^{N} h_i, \text{ where } h_i = 1 \text{ if } l(x_i) \leq y_i \leq u(x_i)$$

$$h_i = 0 \text{ otherwise}$$

- Mean Prediction Interval Width (MPIW)

$$MPIW = \frac{1}{N}\sum_{i=1}^{N} |u(x_i) - l(x_i)|$$

3.7 DEVELOPMENT TOOLS

We have taken into account a selection of representative machine learning methods for the prediction model. Python will be used as the programming language to implement the entire data processing and machine learning algorithms as it is a general-purpose high-level programming language and is widely used in the machine learning community. It is supported by a large number of libraries to perform machine learning tasks, including the *Scikit-learn*, a library that provides a wide range of state-of-the-art implementations of machine learning algorithms while maintaining easy-to-use interface (Pedregosa et al., 2011). We mainly use this library for its simple and efficient way for building the model.

We develop the model using Jupyter Notebook, a web application that supports literate programming paradigm introduced by Knuth (1992). Literate programming combines a programming with a documentation using natural language showing the thoughts behind the program. By utilizing this methodology, we can produce a more robust and easily maintained program.

3.8 CHAPTER SUMMARY

In summary, we carried out this study by following the CRISP-DM framework. We selected several machine learning and classical time series models to be applied and compared. We described the general approach to build models for various aggregation levels and proposed an approach to handle outlier demand to be validated in this study. Then, we define four approaches to generate prediction interval for the model which will be compared using the standard performance metrics for prediction interval.

4 DATASET CREATION

This chapter describes the results for the first part of the research which consists of the Business Understanding, Data Understanding, and Data Preparation phases. The whole chapter focuses on the creation of dataset for Model Building phase. It starts with a description of the business context, a review of the findings from related works, as well as interview for domain analysis, to obtain initial features for the prediction. Then, we proceed to explore the raw datasets that are available. After that, we apply data preparation steps to create the final dataset for modelling phase.

4.1 BUSINESS CONTEXT

ANWB (The Royal Dutch Touring Club) is an organization for traffic and tourism in the Netherlands. ANWB provide services related to assistance, insurance, traffic safety, travel publishing, advice and information. Among these services, roadside assistance is the major business line of ANWB. The service consists of three major elements, which are breakdown assistance by ANWB road patrols, transport service, and replacement vehicles. ANWB provided these services to customers with breakdown incidents in the Netherlands as well as abroad. The focus of this study is in the replacement vehicle provision process in the Netherlands.

ANWB partner up with Logicx, their subsidiary company in charge of handling transport (towing) and replacement vehicles, to provide replacement cars to their customers. In the Netherlands, Logicx´s fleet is distributed in 85 locations owned by Logicx and its partners. Besides ANWB, Logicx provide their service to external customers as well, including other roadside assistance providers, government, and police. The demand from these external customers amount to 30% of Logicx total demand.

The simplified process of ordering replacement cars for customers with breakdown is illustrated in Figure 6. When a breakdown incident occurs, customers contact the ANWB to request for a roadside assistance. ANWB receive the information of the breakdown incidents, such as the location, and assign a road patrol nearby to handle the incident. Road patrol receive the assignment and proceed to assist the customers. When a vehicle cannot be repaired or a reparation is estimated to take or has taken too much time, road patrol can request for a replacement car for the customers if the customers choose to use the service. Logicx receive the order from the ANWB system and will contact the ANWB road patrol to arrange the order. During this arrangement, Logicx will find a rental car that match the customers' cars at the most convenient pick-up location to the customers, be it the closest location to the breakdown or any location preferred by the customers. This arrangement may end up with a cancelled order if there is no match. If a fitting rental car is found, the expected return date and location will also be defined in communication with the road patrol and the customer. In most cases, road patrols will then take the customers to the garage and rental location.

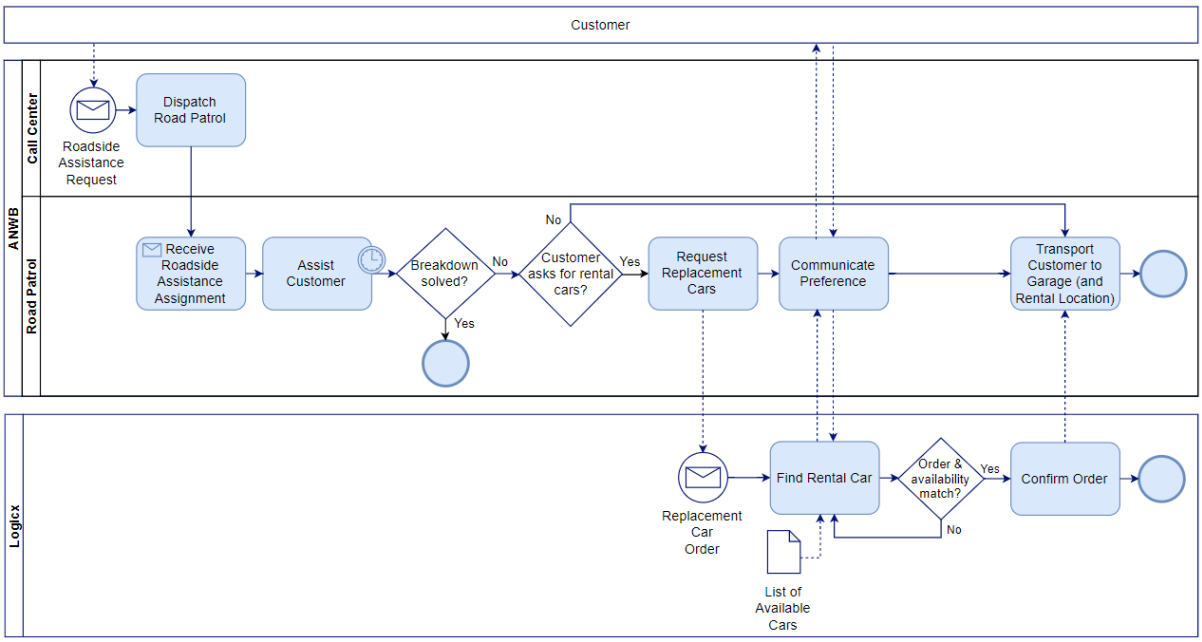


Figure 6 Business Process of Replacement Car Order

Logicx use a predefined mapping of cars into several classes to provide cars from the same classes to the customers. The detailed categorization is attached in Appendix G.1. However, the choice of rental cars is also affected by some business rules. ANWB divide their customers into two market segments, the Business-to-Consumer (B2C) and Business-to-Business (B2B) market, where there are a number of different rules that may apply to customers of each market segment.

1. B2C market
   For the B2C market, ANWB offers the roadside assistance as a membership-based service. The service applies to the members regardless of the vehicles that they own or drive. For customers from B2C market, it is predetermined that the provided rental cars will not be higher than C class cars regardless of the class of the customers vehicles.
2. B2B market
   B2B market segment consists of customers that are entitled to ANWB services through their contracts to another company that has a partnership with ANWB. Different policies apply for different customers depending on the agreement between ANWB and the partners. For example, for customers from company X, ANWB may only assist the customers with transport and replacement cars service without the breakdown reparation. Another policy may state that ANWB are obligated to provide customers from a manufacturer company Y with replacement cars from the same manufacturer/brand.

Furthermore, the replacement cars are rented for a certain period for the customers while the broken cars are being repaired in garage or alike. There is a maximum rental duration that applies to each customer depending on the type of subscription that they have or the contract between ANWB and the B2B partners in case of B2B segment customers.

4.2 Domain Analysis

In the previous section, we have seen the common process of the replacement cars order generation. To put it simply, the demand of replacement cars is a portion of the demand of roadside assistance where further in the process the breakdown cannot be solved on the spot due to various

reasons. In this chapter, the factors that may potentially affect the demand and can be used as predictors of the demand will be defined through domain analysis.

In Section 2.3, we discussed a number of literatures related to rental car demand forecasting. While there has been a number of studies on the case, they are intended for the general car rental industry which serve rental cars as a means of transportation for business or individual travel and tourism purpose (Geraghty & Johnson, 1997; Fink & Reiners, 2006). In addition to historical data of rental orders, several other factors that may affect the demand of rental cars were mentioned. They are seasonality, special (local) events, and weather conditions. Another more recent research studied rental car demand forecasting for the growing car sharing business model (Lei et al., 2017). They defined spatial-related factors such as the main function of a region where the demand is to be predicted (i.e., business, entertainment, resident, etc.), the density of point of interests of the location, and factors such as rental behavior features (e.g. whether the trip is a business or individual trip).

Besides literature review, we conducted a couple rounds of interview to extend and/or validate the candidate features, i.e. the factors mentioned in the related works, including the demand forecasting predictors from other use cases mentioned in Table 3. The first and second interviews were carried out with domain experts with different background within ANWB. During the interviews, several potential factors were brought up:

1. Seasonal pattern
   During summer, a higher demand of rental car is expected.
2. Weather, whether it is hot or cold
   A lot of breakdown incidents happen during cold days because of battery problems but majority of this problem can be fixed easily and do not require replacement cars. On the other hand, in the summer, there seems to be more engine problem occurring and causing breakdowns due to cooling difficulty. This problem is more difficult (or takes too much time) to solve, thus oftentimes end up with a replacement car order.
3. Day of the week, whether it is weekday or weekend
   One of the reason road patrols send a request for a replacement car is that they have no required spare parts on the roadside assistance car and no ANWB shops open close by. Some of the shops open all time, while some others open only on weekdays. The opening hours also differ based on time of the day.
4. Holiday and long weekend
   More people are expected to travel during holiday, the day before, or the day after holiday and long weekend, which means higher possibility of breakdown incidents and requests of replacement cars.
5. Road patrol's experience
   The skills of the road patrol might affect their call. Supposedly, more replacement car requests come from the less experienced road patrols.
6. Type of customers, whether they are classified as a part of B2B/B2C market
   Cars of customers from B2B market are generally more modern and more difficult to repair (e.g. electric cars), thus they are expected to require replacement cars. Moreover, there are regulations for some B2B partners where replacement cars should be provided to the B2B customers directly without reparation assistance

7. Type of cars
   Conditions that lead to the request of rental cars are mainly those related to the car type. For instance, a specific type of cars needs a certain component that is not available in the roadside assistance car or ANWB shops.

The third interview was conducted with a data scientist that developed a model for similar case but with the focus of analyzing the demand in two holiday destination countries, France and Spain. As their focus is more towards building a geospatial model for the demand prediction, they use similar spatial-related features with those mentioned in Lei et al. (2017), which are the number of hotel and campsite in an area, and the length of motorway, where a longer motorway is expected to induce higher breakdown occurrences. While these features are interesting for the model, they are intended for a detailed spatial analysis. The focus of our research is more towards laying the groundwork for the demand prediction in the Netherlands in general. Therefore, the characteristics of a location is out of the scope of the study and will be left out for future work.

Table 4 Overview of Potential Features

Features	Related Work	Interview
Historical rental car orders	[27], [28]	1, 3
Seasonality	[27]	1
Special events	[27], [28]	-
Weather conditions	[27]	1, 2, 3
Weekday		1
Customer type	[45]	1
Road patrols experience		1, 2
Public holiday		1
School holiday		1, 3
Car-related features		2
Sun cycle		3
Length of motorway		3
Number of point of interests	[45]	3

Table 4 summarizes the factors mentioned in literature and interviews. Some of the features mentioned in the interviews are quite hypothetical. Therefore, a further exploratory data analysis for the features will be carried out in the next chapter. Furthermore, car-related features may be one of the factors that causes a car to be unrepairable, resulting in a request for replacement car. However, the replacement car services are not assigned to certain cars but to the member with subscription. The breakdown and replacement vehicle services are applicable to any car driven by the member. Therefore, the car-related feature cannot be used as a predictor. Features like road patrol's experience and the type of customers may not work as predictors either as their future values are unknown. However, a constant/relative number can be used for the feature if a certain pattern or seasonality is found in the data which will be explored in the next part.

4.3 DATA EXPLORATION

4.3.1 Rental Car Demand

Rental car orders are used to represent the rental car demand. We use the data of rental car orders from ANWB to Logicx, which comprise approximately 70% of Logicx total rental car orders. It is important to take into account that the actual demand may not be fulfilled due to unavailability of cars and this unfulfilled demand is not registered in the data of rental car orders. While this may affect the accuracy of the prediction to some extent, based on the experience, Logicx expect that this number is really small.

The dataset is created from a total of 253,713 rental car orders from 2014 to March 2019. Data from before 2014 are excluded due to the change in customers contracts that include a replacement vehicle as a provided service. The demand of rental car has seen an increasing trend since 2014, with 6.8% increase from 2017 to 2018. Figure 7 shows the trend of yearly rental car demand over time.

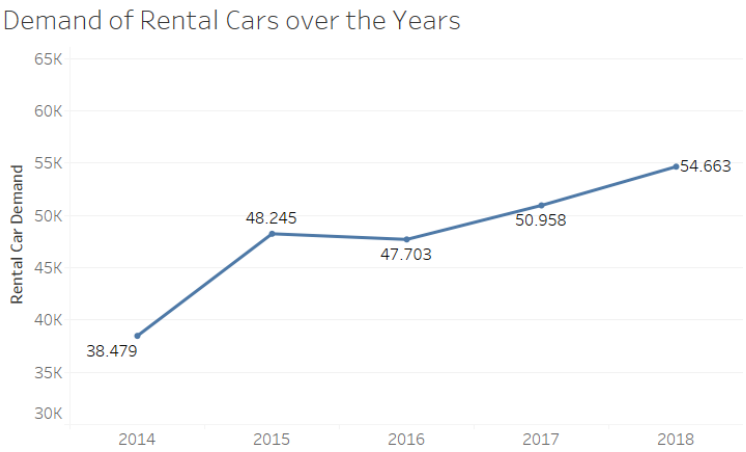


Figure 7 Demand of rental cars over the years

Furthermore, over the last 5 years, the demand of rental cars in total is the lowest during the first quartile (January-March) and reaches its highest during the fourth quarter (October-December) (Figure 8). The demand for the third quarter (July-September) are lower compared to the second and fourth quarter. This is a period of summer school holiday where most people are on vacation. Moreover, more engine problems (which is the most frequent problem leading to replacement car orders) are expected to occur during the summer. However, instead of resulting in a higher demand, the demand is lower than the second quarter. This is also contradictory with the number of breakdowns (which seems to yield a higher number on the third quartile) even though more breakdowns are expected to lead to a higher chance of requesting replacement cars. One possible explanation is that more breakdowns that happen in Q3 are more repairable than those that happen in Q2.

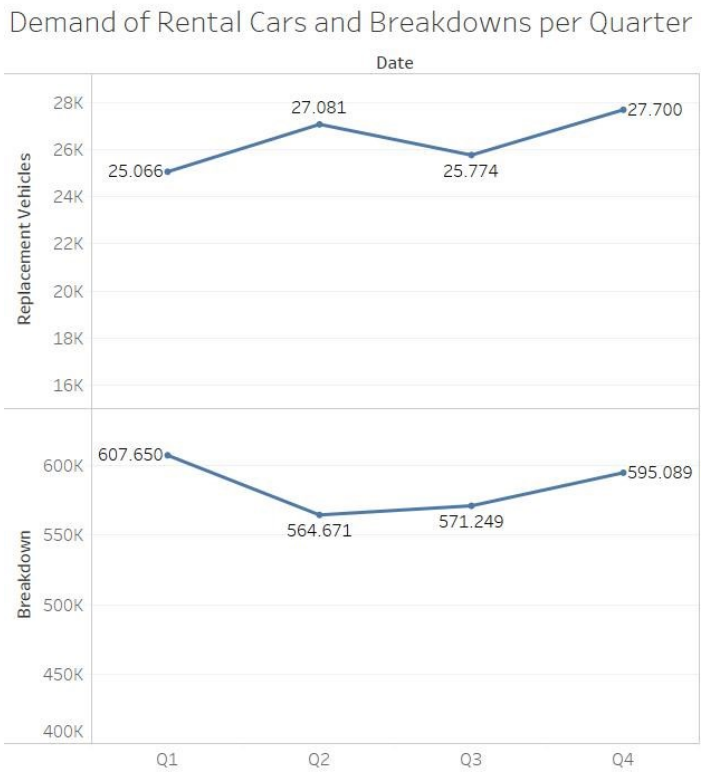


Figure 8 Demand of rental cars and breakdowns per quarter

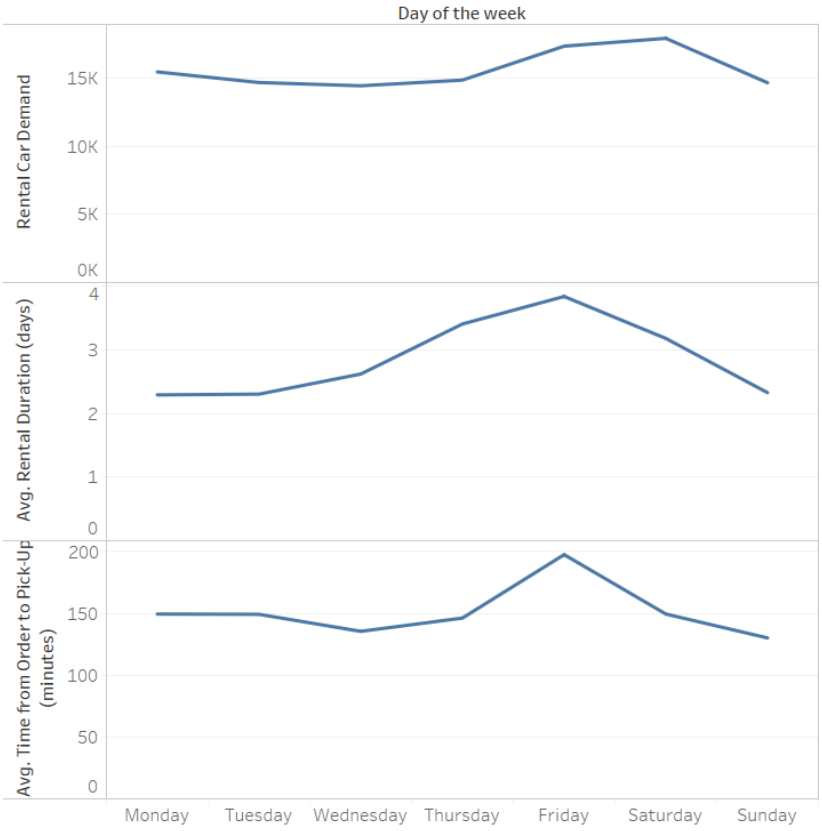


Figure 9 Demand, rental duration, and lead time per day of the week

In terms of the demand on each day of the week, demand of replacement car is higher on Friday and Saturday. On these days, the average rental duration is higher as well. It can be explained by the fact that some of the pick-up/drop-off locations are closed on weekend and the rental duration on the contract only counts the working days. Another interesting insight is that the lead time between the request of replacement cars and the pick-up time is about 50 minutes longer on Friday compared to the other days. This may be an indication that there is a shortage of rental cars at pick-up locations close by due to the high demand on Friday. This condition would require customers to get the rental cars from a further location (hence the increase in the time between the placement of order to the arrival at pick-up locations). Another possibility is that customers tend to travel on a longer distance on Friday and request to pick-up the cars at their destination. It means that the further the destination, the longer the time to arrive to the pick-up locations from the time of order.

Additionally, in Figure 10, we can see the spread of the demand for every day of the week. It shows that despite the tendency of occurences of higher demand on Friday and Saturday, there seems to be quite a number of outliers in the other days, especially Monday and Thursday. It may be a sign of some occassionally high demand on a long weekend. The difference in the distribution of the demand values over the weekend (Friday, Saturday, and Sunday) and weekday also show how demand over the weekend are more varied compared to the more closely spread demand values on weekdays. However, there are more potential outliers during the weekdays.

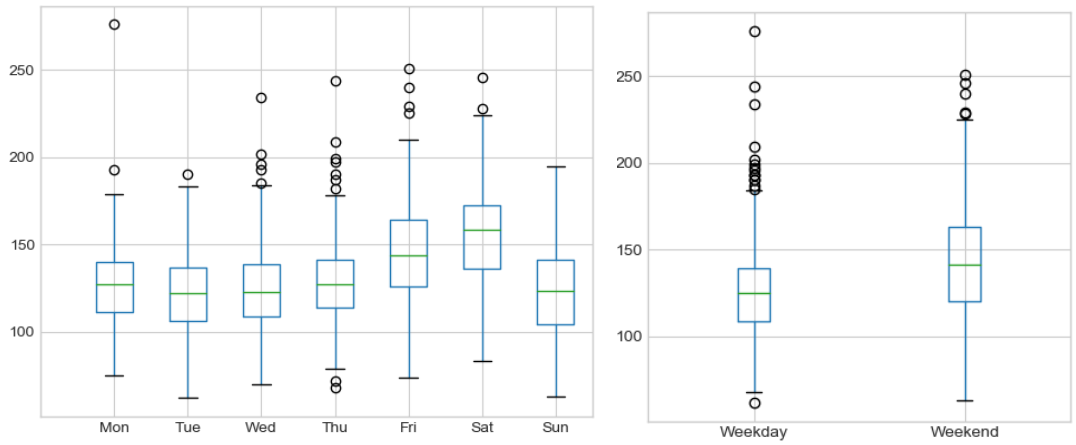


Figure 10 Boxplot of demand per day of the week

Based on the visualizations with various time intervals, we expect multiple seasonality patterns on the dataset, which are weekly with the high demand on the weekend and yearly with lower demand on the first quartile and higher demand on the last quartile. Finally, since the goal is to predict the demand per day, we restructured the data into daily time series. A total of 1916 instances are available to build the prediction model as seen in Figure 11.

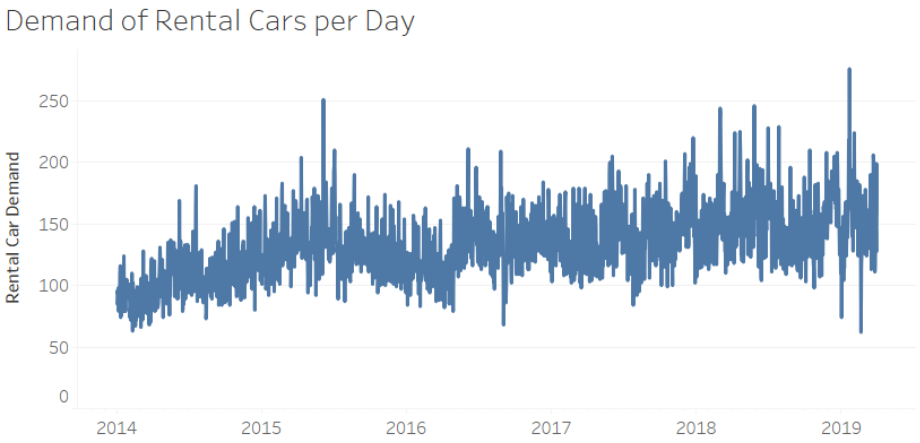


Figure 11 Daily demand of rental cars

From Figure 11 we can see again the increasing trend in general from 2014 to 2019. The demand seems to fluctuate a lot. However, more fluctuations are found in the latter year, with some extremely low and high demand. If we take a look at the distribution of the data in Figure 12 and Figure 13, we can see that the distribution of the demand is nearly symmetrical but with a quite long right tail. Figure 13 shows that demand over 200 on this long tail are the candidates of outlier from the data.

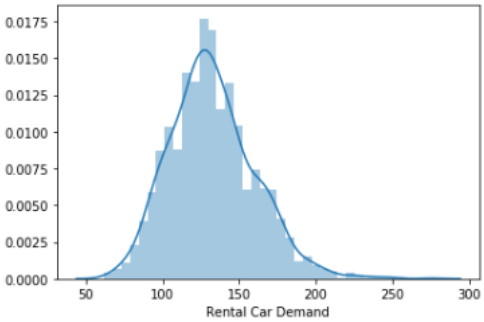


Figure 12 Density plot of daily rental car demand

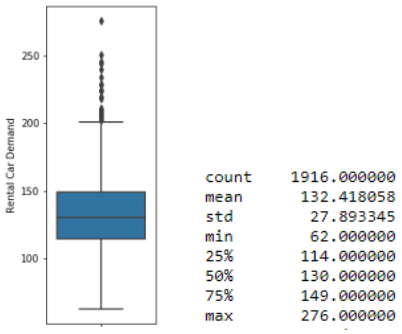


Figure 13 Boxplot of daily rental car demand

There could be several reasons for the existence of these potential outliers, such as noise, human error or error in the data, or other factors that may be important to the model. These outliers can be excluded without further analysis if they are found to be an errror or mistake in the data (Laurikkala, et al., 2000). Therefore, we tried to verify some extreme high and low demand numbers to the domain experts to see if there are indeed some errors in the data or some known reasons where we can exclude the observation (Detailed comments for each sampled outlier cases are attached in Appendix A).

Overall, it seems that factors related to weather and holiday are still expected to explain the occurences of extremely low and high demand values. From the extreme low cases, we also found 2 weeks consecutively low demand which turned out to be caused by the replacement cars orders being recorded in two separate systems due to system migration. We have then dealt with this by aggregating the data from both systems/database. For extreme demand cases, the reasons for some of the cases are not as obvious as the others. Interestingly, in one case, the actual demand for roadside assistance was also much higher than the forecasted demand of roadside assistance, while in another case the demand for roadside assistance looks normal and is close to the expected demand but the number of replacement cars is high. Since none of the inspected outliers appear to be an error in the data and they are reckoned to be caused by the weather and holiday factors that are going to be incorporated in our model, we decided to keep the outliers in the analysis.

4.3.2 Weather

Weather data for the Netherlands are collected from open data provided by The Royal Netherlands Meteorological Institute, KNMI[1]. There are several weather attributes available on daily level as can be seen in Table 5. These data are available in 50 weather stations of the Netherlands.

Table 5 Description of KNMI Weather Data

Attribute	Description
FG	Daily mean windspeed (in 0.1 m/s)
TG	Daily mean temperature in (0.1 degrees Celsius)
TN	Minimum temperature (in 0.1 degrees Celsius)
TX	Maximum temperature (in 0.1 degrees Celsius)
RH	Daily precipitation amount (in 0.1 mm) (-1 for <0.05 mm)
PG	Daily mean sea level pressure (in 0.1 hPa) calculated from 24 hourly values
UG	Daily mean relative atmospheric humidity (in percent)

Out of 50 weather stations, we excluded 17 weather stations. Four of them do not have a complete record of weather data for all dates since 2014 (i.e. they are either new or discontinued), while the rests record only the daily mean wind speed out of all variables (See Appendix B for the complete percentage of missing weather data).

For the demand prediction at the high level, we use weather data from De Bilt weather station, which is located in central Netherlands and is also the oldest weather station in the country. At province and work area level, we take the data from the weather station closest to the center of the area. Similarly, the closest weather station to each Logicx pick-up location provides the value of the weather variables for the corresponding location every day.

De Bilt weather station has no missing data. However, some of the other weather stations still have a certain percentage of missing values. For these missing values, we fill in the values with values from the next closest station to the location. For instance, if in one of the days the minimum temperature from the weather station closest to a province is missing, we take the minimum temperature value from the second closest station. If the value from the second closest station is still missing, we replace it with the value from the third closest station. It goes on until there is no more missing value in the data.

[1] http://projects.knmi.nl/klimatologie/daggegevens/

To see how the weather may affect the demand of replacement cars, a plot of each weather variable from De Bilt weather station and the rental car demand is presented in Figure 14. It can be seen from Figure 14 that some extremely high demand happens on the day with either a very low temperature, extremely high temperature, or extremely low atmospheric humidity.

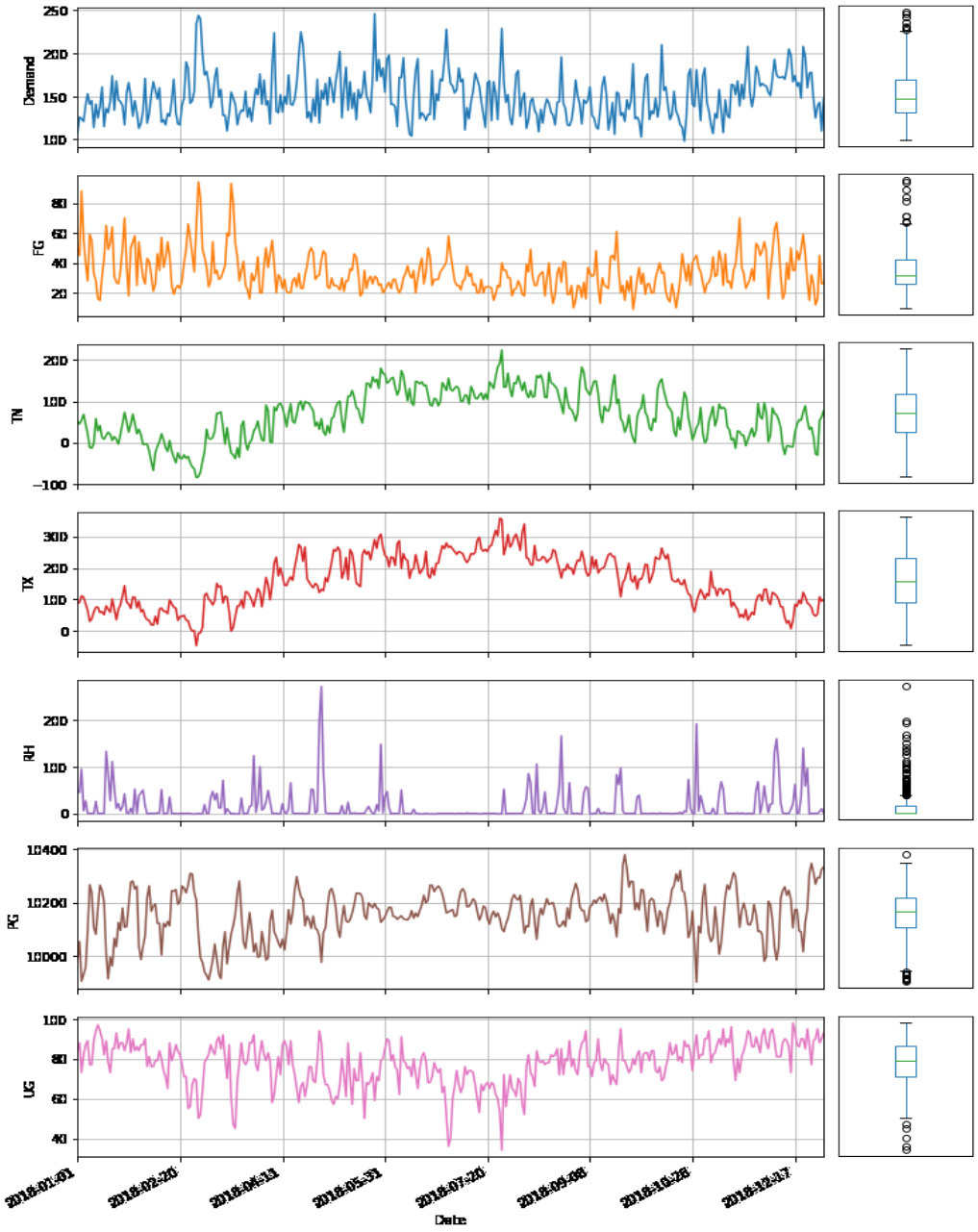


Figure 14 Visualization of Weather Variables in comparison to Demand in 2018

4.3.3 Other features

In the previous chapter, we have explored historical orders, seasonality, weather conditions, and weekday. There are some other features that have been mentioned as potential factors for rental car demand according to the literature review and domain analysis (Table 4). Special events, public holiday, school holiday, and sun cycle features could be created during the feature engineering phase, thus they would be explored directly with the machine learning model. Information regarding customer type, in this case the B2B and B2C customers, could be retrieved from the historical rental

order data. However, the number of each type of customers in the future is unknown and would require a separate prediction. In addition, during the exploration process, we have experimented with various constant/relative number of the customer type so that it can be used as an input for future prediction, such as the ratio of each type of customers to all customers, ratio of customers from a certain type that request replacement cars to ones with breakdowns in general, and various other possible values. However, the number of customers over various periods (e.g. daily, weekly, monthly) is rather stable and does not exhibit any striking pattern. Therefore, we decided to drop this factor and investigate the feasibility of predicting them instead.

Road patrols experience is another factor that is difficult to quantify. For instance, data related to the working years of road patrols are difficult to obtain as they contain private information. Another factor with almost similar issue is the car-related features, such as the age, type, or model of the car. These data are available for analysis. However, as it has been discussed in Section 4.2, the way the current policy are makes the future value of these features extremely difficult to determine. Therefore, we decided to exclude the road patrols experience and the car-related features from the analysis. In addition, length of motorway and the number of point of interests were also excluded as they are more spatial related and would be out of scope of the study.

4.4 Data Preparation

4.4.1 Feature Engineering

To create the full dataset with the required features, we derived new features from the raw univariate time series data of rental car demand and the weather data. Using the date feature from the original datasets, we created features that can be used for supervised learning, including the year, month, and weekday. Sun cycle features are also created according to the date, using Python Astral library[2]. Holidays, school holidays, and special events in the Netherlands are generated using an in-house Python script with date as an input. Lags for the demand are generated by taking the demand values of the previous 7 days. Seven is chosen as the number of lags given the weekly seasonality expected from the data (also exhibited by the autocorrelations plot in Appendix C).

Next, we created more features using interaction of some features by taking the difference, summation, counts of occurrences, average, standard deviation, or other possible form of feature interaction. This is done iteratively with the support of domain knowledge and experiments. For instance, we created the feature *isHoliday* which is a single representation of whether a day is a day off, regardless of what holiday it is or if it is only a weekend day-off and not a calendar holiday. The average and standard deviation of the demand from the previous week are considered as well to extract a better summary of the demand trend compared to using only the values from the previous days as 7 individual features. Features such as temperature difference with the day before and a count of consecutive cold and warm days are the other examples.

We also took into account some possible forms which a feature can be used in. For example, Month feature that has the value of 1-12 can be presented in a polar format, that is a sine-cosine representation, to give an information of its cyclical nature (e.g. Month 1 comes after Month 12). Some algorithms can handle binary variables better than categorical variables with more than two values. Therefore, we encoded categorical features using several encoding strategies. Originally, dummy variables will be created for each value of these categorical features (also known as the one-hot encoding technique). However, due to the high sparsity induced in the dataset with one-hot

[2] Astral documentation https://astral.readthedocs.io/en/stable/module.html

encoded features, we considered comparing it with other techniques, including hashing (Weinberger, et al., 2009), binary encoding, and the above-mentioned polar technique. The results of this comparison will be discussed in Section 5.1.4. We created a total of 90 features for the dataset.

Table 6 summarizes the features on the dataset after feature engineering (See Appendix D for a full description of the features).

Table 6 Summary of Input Features

Category	Features
Target lags	Demand of replacement car per day, average, and standard deviation for the previous 7 days
Datetime attributes	Weekday, month, year, sun cycle
Public holiday	New Year's day, Easter, King's day, liberation day, ascension day, Pentecost, Christmas
School holiday	School holiday in the Northern, Central, and Southern regions
Other special day	Good Friday, National remembrance day, New Year's eve, Saint Nicholas' eve, Carnival
Day before holiday	Day before holidays, i.e. day before Good Friday, Easter Sunday, King's day, Liberation day, Remembrance day, Whit Sunday, Christmas day, New Year's eve
Day after holiday	Day after holidays, i.e. day after New Year's day, Easter Monday, King's day, Liberation day, Remembrance day, Whit Monday, Boxing day, New Year's day, first Monday after New Year's day
Holiday attributes	Whether a day is a holiday/weekend, first day of a long holiday, last day of a long holiday, a day in a long weekend, a day before a long holiday
Weather	Windspeed, maximum, minimum, and mean temperature, precipitation, sea level pressure, atmospheric humidity
Weather attributes	Number of consecutive cold/warm days, total cold/warm days in the previous 7 days, temperature difference with the day before

4.4.2 Dimensionality Reduction

Dataset with large number of features may not be effective to build a model on. High dimension means high complexity and more possibility that some of the features provide similar information or only introduce noise to the model. With 90 features on the dataset, we eliminated a number of features with high percentage of missing values, low variance, and high correlation with other feature(s).

First, no features have a high percentage of missing value. Secondly, using the low variance filter, we removed one holiday feature which is *holiday_bevrijdingsdag_off* (i.e. Liberation day as an official paid holiday), along with the day before and after this holiday, which only has the value True once every 5 year and has the value False for the rest of the rows. Finally, we run correlation test on all the features.

The following relationships were found to have high correlations.

1. *Minimum temperature* is highly correlated with *Average temperature* (ρ = 0.93361)
2. *Maximum temperature* is highly correlated with *Average temperature* (ρ = 0.97271)
3. *Time-to-sunrise* is highly correlated with *Month (cosine)* (ρ = 0.9376)
4. *Time-to-sunset* is highly correlated with *Daylight* (ρ = 0.99264)

We removed *Time-to-sunrise*, *Time-to-sunset* and *Average temperature* and kept the other features. We opted to keep both minimum and maximum temperature instead of the average temperature due to the previous hypothesis from the domain analysis that suspects a hot day as the trigger to an

engine failure that is difficult to repair on the spot (hence resulting in a request for replacement cars).

Lastly, before training and testing the model, which will be discussed in detail in the next chapter, feature scaling was carried out on the dataset. Due to most of the features being categorical and the fact that some of the numeric features such as *Daylight* do not exhibit Gaussian distribution, maximum-minimum normalization technique was used instead of standardization to avoid the distributional assumption. Normalization was done on the training set and then applied on the test set. This is to avoid information leak to a test set that should represent unseen observations. In this manner, one could expect the generalization ability demonstrated from the evaluation on the test set to reflect the true generalizability to an unseen data in the future. The values in all input features were transformed to a range of 0-1. Thus, no feature will dominate the other features just because of the larger range of values one has.

4.5 CHAPTER SUMMARY

In this chapter, we listed a number of candidate features based on domain analysis through literature review and interviews. Then, we selected the most relevant features and eliminated features that have no data available or are deemed to be out of scope of the study. Through data exploration, we gained more insights about the characteristics and quality of the data, and the potential effect of a feature on demand prediction. Then, we created more features that are not directly available in the raw dataset, features that are likely to be valuable as predictors and features that can help the models to learn the relationships. Lastly, we pre-processed the data accordingly so that they are ready to be processed in the modelling phase.

5 DEMAND PREDICTION MODEL

This chapter covers the results of the modelling phase in the following structure.

Section 5.1 provides the results that will answer SQ2 by following the first approach for model building defined on Section 3.3. We first present the results of the prediction models using classical time series techniques, followed by the ones using machine learning techniques. After that, we show a comparison of different data preprocessing (i.e. encoding categorical variable) strategies as an effort to improve the models. Then, the section after presents the result of the automated feature selection and the performance of the models retrained using the automatically selected subset of features. We analyzed the result of the best performing model. Figure 15 summarizes the experiments and comparisons that were conducted for the highest level prediction model.

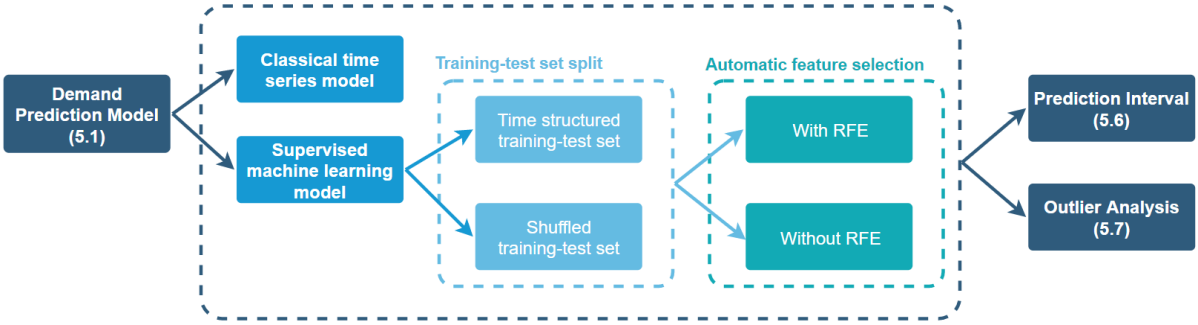


Figure 15 Outline of experiment on high level prediction

Section 5.2 to 5.5 discuss the results for the deeper aggregation levels to answer SQ3, by applying the approach that perform best according to the results of the high level model in Section 5.1. We discuss the demand prediction model per market segment, demand prediction per province, ANWB work area, and Logicx rental location in order. Figure 16 outlines of the prediction levels investigated in this study and their respective sections.

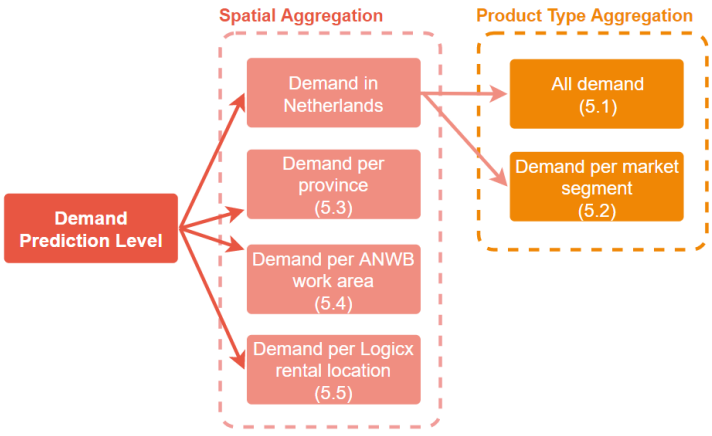


Figure 16 Demand prediction level

Afterwards, we proceed to examine the proposed components to enhance the demand prediction model, which are the prediction interval and outlier analysis. Section 5.2 describes the prediction intervals built on the highest aggregation level model to answer SQ4. Methods defined on Section 3.6 were applied in this section. Lastly, we implemented the second approach of model building that was proposed on Section 3.3 and presented the results on Section 5.7.

5.1 Demand Prediction for the Netherlands

5.1.1 Classical Time Series Models

Before we build the model, the univariate time series data can be decomposed into its trend, seasonal, and residual components as seen in Figure 17. It can be seen that there is an increasing trend and a yearly seasonality on the data.

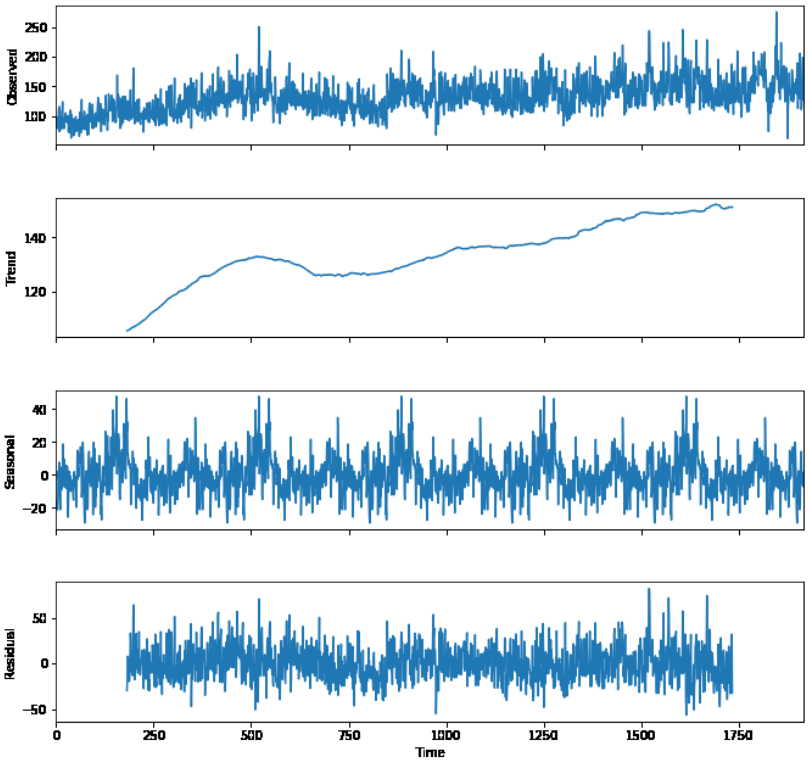


Figure 17 Time series decomposition

Next, we train the classical time series models using ARIMA (or variations), Holt-Winters Exponential Smoothing, and Prophet. To be able to compare the results with the machine learning performance, we train the model using the first 70% data and leave the last 30% of the data to predict and measure the evaluation metrics on. We conduct one-step ahead prediction for the classical time series model as the prediction on the testing set for the machine learning models will also use a one step ahead actual value as the lags. Unlike machine learning model, for classical time series model, this procedure requires predicting the data one by one (i.e. one day at a time) and retraining the model after adding the data of the predicted day to the training data.

Auto-ARIMA (with exogenous variables)

We used Auto-ARIMA to select the most suitable models from ARIMA and its variation along with the hyperparameters. We run Auto-ARIMA with the stepwise parameter selection method. Exogenous variables (i.e. all input features except the lags features) are also included in the training. SARIMAX(1,0,0)(0,0,0)7 is selected as the best performing model.

Exponential Smoothing

We manually varied the configuration of the Exponential Smoothing model to get the best result. Additive type of trend and seasonality are selected. Seasonal periods of 7 is used as we want to

analyze it with the weekly pattern of the daily data. Exponential smoothing analyzes only the univariate time series without exogenous variables.

Prophet without exogenous variables

We tuned the Fourier order for the yearly seasonality and the seasonality mode for the model. Prophet decomposed the data as can be seen in Figure 18 (by default without the holidays and extra regressors decomposition) and perform the prediction using these components. Unlike ARIMA, Prophet decomposed the time series weekly seasonality in addition to yearly.

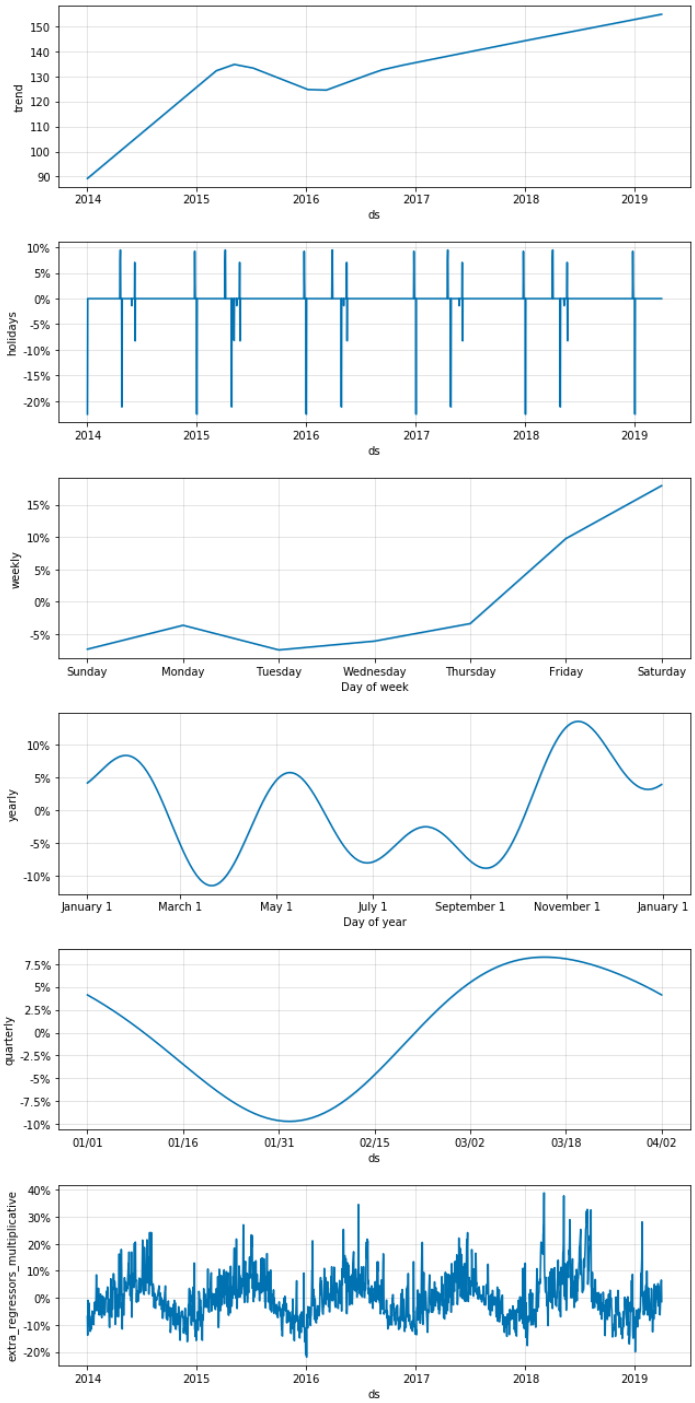


Figure 18 Time series decomposition using Prophet with exogenous variables

Prophet with exogenous variables (holidays and extra regressors)

After tuning the Fourier orders for yearly seasonality and custom-defined quarterly seasonality, the seasonality mode, and prior scales for the holiday and other exogenous variables, we train the Prophet model. Prophet analyzes holiday as one feature for all different holidays, similar to our *isHoliday* feature. Figure 18 shows the decomposition of the time series trained with the Prophet model, which include decompositions of the effect of holiday and extra regressors (other input features) as well.

Table 7 shows the performance of each model based on various metrics. It can be seen that SARIMAX model outperforms the other classical models on all measures.

Table 7 Performance of TIme Series Models

	MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
Auto-ARIMA: SARIMAX(1,0,0)x(0,0,0,7)	13.9534	9.6573	346.0921	18.6036	11.1395	0.5143
Exponential Smoothing	15.3365	17.7173	410.2963	20.2558	12.799	0.4242
Prophet (with exog)	15.4519	10.534	418.9895	20.4692	12.423	0.4119
Prophet (without exog)	16.5318	11.3131	488.031	22.0914	13.024	0.3151

5.1.2 Train Models with Time Structured Dataset

We have evaluated some classical time series models as the benchmark for the machine learning models. After that, as discussed in Section 3.4, we trained the prediction model by splitting the training and test set keeping the time structure and using time series cross validation (TSCV) approach for the cross-validation folds. Five-fold TSCV were used to have at least around one year data for each training fold. Table 8 shows the performance of each model on the test set.

Table 8 Performance of machine learning models with time structured dataset

	MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
Linear Regression	14.1083	9.80982	353.395	18.7988	11.4258	0.504017
Ridge	14.6449	10.1152	384.949	19.6201	11.913	0.459731
SVR (RBF)	14.8243	10.1424	391.446	19.785	12.0018	0.450613
SVR (Linear)	15.0387	10.3556	404.058	20.1012	12.0534	0.432912
Lasso	15.167	10.3459	411.08	20.2751	12.2842	0.423057
Gradient Boosting	15.7201	10.2446	445.146	21.0985	12.7781	0.375246
XGBoost	16.7681	10.7704	506.486	22.5053	12.8608	0.289156
Random Forest	16.9345	10.8908	521.575	22.838	12.8449	0.267979

From Table 8, it can be seen that no model can outperform SARIMAX(1,0,0)(0,0,0)7 for all performance metrics. When we investigated further, the time series cross-validation (TSCV) splits have resulted in an unstable performance for each cross-validation fold. This may have caused the hyperparameters tuned using grid search with cross-validation unable to generalize well to the test set. It can be seen from the difference in the performance of the models on the training and test set (See Appendix F.1 for a complete performance on both training and test set). There is a clear overfitting on all models as the performances on the training set are much better than on the testing set, particularly demonstrated by the MSE. It could also be an indication that there are more outliers on the test set compared to those learned by the model from the training set. Therefore, shuffling

the dataset for training and testing are expected to improve the performance as parts of the highly fluctuating demand in the more recent years are included in the learning process as well.

5.1.3 Train Models with Shuffled Dataset

Considering the nature of supervised learning, it is possible to train the model on a shuffled training and test set splits. Theoretically, this is not the most proper approach for time series prediction as discussed in Section 3.4. However, Bergmeir and Benitez (2012) has shown that in practice, shuffling the dataset has not cause any issue with the solution. Therefore, we shuffled the dataset when splitting training and test set and retrain the models. Five-fold cross validation was used to select the best parameters of each model instead of the time series cross validation. Table 9 shows the performance of each model on the test set (See Appendix F.2 for a complete performance on both training and test set).

Table 9 Performance of machine learning models with randomized dataset

	MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
XGBoost	12.0562	9.47926	243.541	15.6058	10.0967	0.689771
Gradient Boosting	12.1378	9.58108	244.092	15.6234	9.72451	0.68907
SVR (RBF)	12.3349	9.60096	250.975	15.8422	10.1916	0.680301
Ridge	12.1955	9.5713	252.68	15.8959	9.76322	0.67813
Lasso	12.3308	9.71742	257.207	16.0377	9.78918	0.672364
Linear Regression	12.3535	9.68606	258.898	16.0903	9.8125	0.670209
SVR (Linear)	12.352	9.65136	261.649	16.1756	9.83929	0.666705
Random Forest	13.0045	10.2575	283.94	16.8505	10.2666	0.63831

It can be seen that there is a major improvement on all models. The highest performing model has 69% of the target variance explained by the input features and a significantly lower MSE. However, we still observed an overfitting for all the tree-based models, which are XGBoost, Gradient Boosting, and Random Forest. Two out of the three models top the performances among all models. However, there are not that much difference between their performances with the simpler linear regression. One would expect that the model performance of gradient boosting tree models will be improved if the overfitting can be controlled.

Table 10 XGBoost performance

XGBoost	**Training**	11.067	8.62665	198.766	14.0984	9.14	0.743072
	Testing	12.1754	9.64069	246.139	15.6888	10.1648	0.686462

Overfitting on XGBoost can be controlled by controlling the complexity of the model and adding randomness to the model. Both techniques have been taken into account by tuning the related hyperparameters. Table 11 shows the selected combination of parameters and the negative mean squared error values for each cross-validation split. The average performance on the cross-validation splits do not differ far from the performance on the test set. However, retraining the data on the whole training set introduced the overfitting.

Table 11 XGBoost performance for each cross-validation split

param_gamma	0.5
param_max_depth	3
param_min_child_weight	5
param_subsample	0.5
split0_test_score	-320.132
split1_test_score	-228.553
split2_test_score	-246.486
split3_test_score	-289.736
split4_test_score	-256.546
mean_test_score	-268.329
std_test_score	32.6984

The result of overfitted model often has poor generalization (Bishop, 1991). Therefore, avoiding overfitting is necessary to have a better generalization performance to hold-out test set. In our case, reducing overfitting by reducing the complexity of the model has worsened the performance on the test set as well. It is however worsened only by a little, compared to how much the training set performance has decreased. The performance of both training and test set can be controlled to be more similar at the cost of lower performance in the test set.

5.1.4 Comparison of Encoding Strategies

Overfitting in a tree-based model can occur when the tree ends up with strict rules of sparse data on the training set. Considering the dataset that is used to train the model consists of a large number of one-hot-encoded features (i.e. binary features where most of the values are False), it may have caused the model to overfit. To inspect if a denser dataset can improve the performance, we experimented by comparing various encoding strategies to the one hot encoding. The performances of the models with different types of weekday and month features are presented in Table 12.

Table 12 Performance of different encoding strategies

Model	Ordinal	One hot	Binary encoding	Hashing	Polar
Random Forest	293,481	288,129	309,822	319,489	280,176
Gradient Boosting	247,077	251,288	266.65	278.06	248,831
XGBoost	254,584	243,541	280,122	281.2	256,132
SVR_RBF	312,806	250,975	254.83	285,033	257,623
SVR_linear	323,426	261,649	297,567	298,032	294,156
Lasso	316,342	254,484	284,235	281,151	285,332
Ridge	316,617	252,415	284,239	284,405	283,735
Linear Regression	318.25	258.92	285,811	295,231	288,262

Table 12 shows that in general, the other strategies still cannot outperform the performance of the model trained with the initial encoding technique, which is one hot encoding, for weekday and month features. Although, there is an exception for Random Forest and Gradient Boosting. Polar encoding outperforms one hot encoding technique for these two machine learning algorithms. Also, for Gradient Boosting, ordinal encoding appears to outperform all the others, including the polar technique. This result may be attributable to the ability of the tree-based model Random Forest and Gradient Boosting to treat an ordinal encoded feature as a categorical feature, unlike for example a linear regression that would take the value as a continuous value. Besides, polar technique also brings additional information of the cyclical nature of the features which may further improve the performance. These differences show that the best performance of each algorithm may be obtained

by customizing the processing techniques for each algorithm. However, since the difference between these algorithms with one hot encoding technique are quite small and one hot encoding is favorable for all the other algorithms, we proceeded to use one hot encoded weekday and month features for our model to compare these machine learning algorithms in the same manner.

5.1.5 Feature Selection and Feature Importance

The current models are still trained on the full feature set. As mentioned in Section 3.4.2, we perform the Recursive Feature Elimination with Cross Validation (RFECV) to automatically determine the number of features and which feature can result in the best performance. We perform the RFECV using for every model with the exception of SVR with RBF kernel due to the limitation that RFE can only run on models with coefficient (linear models) or feature importance (tree-based models). Figure 19 shows the performance of each iteration of the automatic feature selection on all features and the number of final selected feature set for each model.

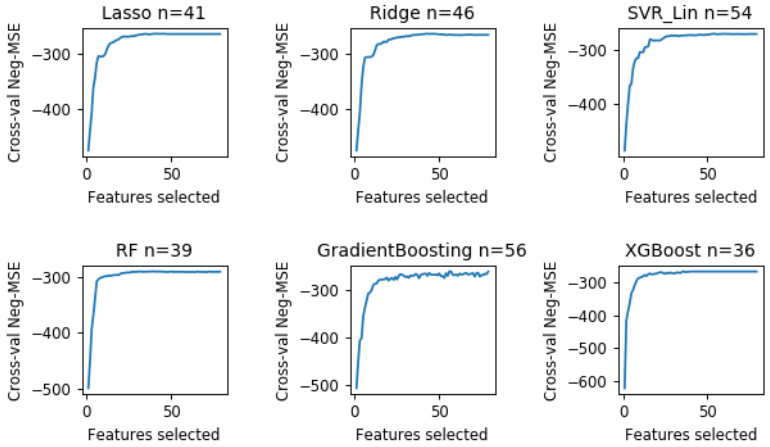


Figure 19 Performance per RFE iteration

Among the selected feature set for all models, there are 15 agreed upon features by all models, which are *Consecutive_cold_days, Consecutive_warm_days, Month_5, RH, Rolling_mean, Target_t-1, Target_t-2, temp_diff_1_days_ago_TX, TN, TX, Weekday_1, Weekday_4, Weekday_5, Weekday_6,* and *Year*. These features are included in the final subset by all models.

With the new subset of feature, we transform the dataset to include only the selected features and retrained the models. The performances of all models using the new subsets of features are presented in Table 13. XGBoost came out as the best-performing model with 69.2% explained variance by using the least number of features, in fact 36 features. The XGBoost performance after applying RFE barely has any difference with the performance before RFE. However, it has reduced the number of features from 42 to 36 which will lead to a more efficient processing.

Table 13 Performance after RFE

	MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
XGBoost	12.0608	9.49054	241.797	15.5498	10.1423	0.691993
Linear Regression	12.2346	9.67313	250.608	15.8306	9.80661	0.68077
Gradient Boosting	12.2624	9.67205	251.737	15.8662	9.6451	0.679331
Lasso	12.308	9.68269	254.854	15.9642	9.97939	0.675361
Ridge	12.305	9.67334	255.151	15.9735	9.8918	0.674982
SVR (Linear)	12.3439	9.64133	262.694	16.2078	9.87814	0.665373
Random Forest	13.0943	10.3242	288.397	16.9823	10.1547	0.632632

Before RFE, XGBoost included 42 features in the model. The ranking of feature importance before RFE feature selection are as follows.

1. Rolling_mean (0.156673)
2. Target_t-7 (0.109680)
3. Weekday_5 (0.054925)
4. Target_t-1 (0.045274)
5. Weekday_4 (0.038948)
6. holiday_schoolvakantie_zuid (0.037628)
7. Weekday_6 (0.036103)
8. Year (0.028780)
9. TX (0.028020)
10. Consecutive_cold_days (0.024936)
11. TN (0.022128)
12. Consecutive_warm_days (0.021046)
13. Target_t-2 (0.021036)
14. temp_diff_1_days_ago_TX (0.018108)
15. Cold_days_prev_week (0.017567)
16. Month_6 (0.017362)
17. Month_10 (0.017199)
18. UG (0.016262)
19. Target_t-6 (0.015945)
20. Weekday_1 (0.015483)
21. holiday_schoolvakantie_midden (0.014504)
22. RH (0.014422)
23. FG (0.014110)
24. Month_7 (0.013891)
25. Warm_days_prev_week (0.013848)
26. Rolling_std (0.013729)
27. Month_5 (0.013706)
28. holiday_hemelvaartsdag-1 (0.013558)
29. Month_11 (0.013163)
30. isHoliday (0.013034)
31. Day_before_long_holiday (0.012785)
32. PG (0.012750)
33. Target_t-4 (0.012149)
34. temp_diff_1_days_ago_TN (0.012006)
35. Weekday_2 (0.009801)
36. Daylight (0.009720)
37. Target_t-5 (0.009411)
38. Target_t-3 (0.009391)
39. Month_1 (0.008584)
40. holiday_schoolvakantie_noord (0.008361)
41. Month_9 (0.008159)
42. Weekday_3 (0.005816)

After feature selection, six features are further excluded from the model. They are *Month_10, Holiday_hemelvaartsdag-1, Target t-5, holiday_schoolvakantie_noord, Month_9,* and *Weekday_3*. XGBoost feature ranking based on its importance after feature selection are as follows.

1. Rolling_mean (0.149491)
2. Target_t-7 (0.118423)
3. Weekday_5 (0.059693)
4. Target_t-1 (0.048088)
5. Weekday_6 (0.041002)
6. holiday_schoolvakantie_zuid (0.040833)
7. Weekday_4 (0.039193)
8. Year (0.033965)
9. TX (0.033221)
10. Consecutive_cold_days (0.025869)
11. temp_diff_1_days_ago_TX (0.023572)
12. Target_t-2 (0.022929)
13. TN (0.020471)
14. Consecutive_warm_days (0.019900)
15. Month_7 (0.018544)
16. Month_6 (0.018485)
17. Target_t-6 (0.018156)
18. Cold_days_prev_week (0.017586)
19. Weekday_1 (0.017523)
20. UG (0.017329)
21. RH (0.016986)
22. holiday_schoolvakantie_midden (0.016191)
23. FG (0.015505)
24. Month_11 (0.015282)
25. Warm_days_prev_week (0.014654)
26. temp_diff_1_days_ago_TN (0.014509)
27. isHoliday (0.014128)
28. Rolling_std (0.014083)
29. Day_before_long_holiday (0.013498)
30. PG (0.012810)
31. Target_t-4 (0.012809)
32. Weekday_2 (0.011720)
33. Month_5 (0.011318)
34. Month_1 (0.011231)
35. Daylight (0.010567)
36. Target_t-3 (0.010437)

One thing that is important to note is the fact that XGBoost excluded most holiday features. After RFE, we do not see any individual holiday feature listed for the model. This can be explained by the sparsity of the data. These Boolean holiday features have very few True values. It makes the gain in purity per split very insignificant thus the tree is very unlikely to select the holiday feature. More holiday features are included in the subset of selected features in the linear models. However, the linear models cannot outperform the tree-based model.

5.1.6 Prediction Results

After comparing the models and the different approach used in training them, the best results are achieved by XGBoost with RFE selected subset trained on randomized training and test set split. Figure 20 shows the visualization of actual vs predicted value for the XGBoost model. From this plot, we can see that the model performs quite well by the pattern of the scattered value that follows the straight diagonal line. There is a good correlation between the actual and predicted values. However, we can see that the model tends to underestimate high value and a bit overestimate the low demand value. Overall, it performs well for the demand value around average demand. Furthermore, if we see the error, the high error/residuals are not only found for the extremely low and high values, but also for some of the average values.

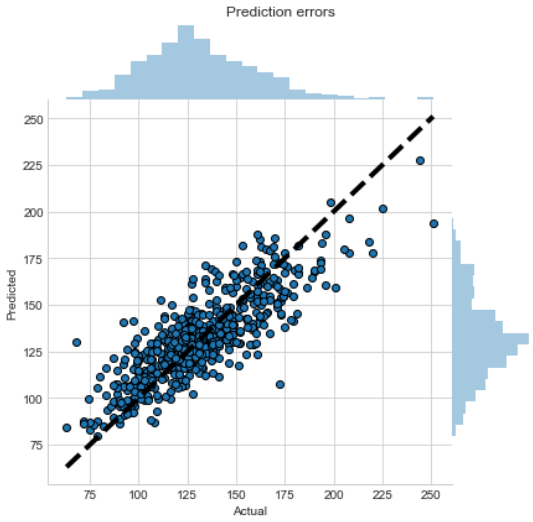


Figure 20 Actual vs predicted demand for XGBoost

Figure 21 further shows the plot of the predicted values and the residuals. The residuals seem to be quite decent considering that it has no clear pattern or trend and are quite randomized and symmetrically distributed. Thus, the model may not be perfectly accurate but it represents the case well enough.

Figure 21 Residuals for XGBoost

Compared to the result in a recent study by Whitt and Zhang (2019), the best model in our case produced a comparable error rate with their best model, in fact a MAPE of 9.5% compared to 8.4%.

Even though the prediction is done for different domain and dataset, there are similarities to our study. First, a lot of features used in our study are found in Whitt and Zhang (2019), such as the calendar and weather features among others. Furthermore, both studies are intended to predict daily demand one step ahead. Thus, this gives us a good insight of how our model has performed in comparison to the other state-of-the-art demand prediction.

However, based on the preliminary interviews with domain experts, it is known that a machine learning model can predict breakdown incidents with an error of around 3-4%, while our replacement car demand prediction model can only perform with MAPE around 9%. This may be caused by the inability of the current model to learn certain patterns from barely 5 years of data. It mostly applies for the holiday features. As discussed in the previous chapter, the model excluded most holiday features, especially those that only happen once a year. With 5 years of data, we only have 5 occurrences of the holiday. Thus, it is normal that the model cannot learn the pattern from only 5 occurrences out of 1916 instances.

However, it is also important to note that this percentage score is the average score for all days. The 50th percentile of the absolute error values is 10 cars (i.e. 7.5% percentage error). It means that half of the cases are well predicted. The 80th percentile has an absolute error value around 20 cars (i.e. 15% percentage error) which show that for 80% of the model predicted the demand quite decently. However, for some of the days in the dataset, we observed some quite high errors as presented in Table 14.

Table 14 Days with highest error

Date	Actual	Prediction	Error
2016-09-09	172.0	107.450798	64.549202
2016-09-01	68.0	130.119324	62.119324
2015-06-05	251.0	193.533569	57.466431
2015-05-17	92.0	140.833572	48.833572
2015-12-18	97.0	141.477493	44.477493
2017-12-23	220.0	177.924530	42.075470
2017-10-14	201.0	159.039780	41.960220
2018-07-09	111.0	152.818359	41.818359
2017-03-10	179.0	141.384277	37.615723
2017-10-26	161.0	123.465561	37.534439

Interestingly, even though there is a tendency that the high demand values are underestimated as seen in Figure 20, some extreme high/low demand can be predicted better than others. For instance, in Figure 22, one of the day in 2018 with 244 actual demand value is predicted with only 16 cars error while one of the day in 2015 with 251 demand is predicted with 57 cars error. When we investigated the features, the temperature on the day with 57 errors is quite high, in fact 31.8^{o}C which is also an increase of 9.8^{o}C from the day before. However, it seems that due to the average demand being only 139 cars and the highest demand in the previous week is not more than 160 cars, the model failed to increase the prediction accordingly. It only increased the prediction to 193 cars.

With the existence of such cases, we experimented with another approach to deal with the outliers which is later discussed in Section 5.7.

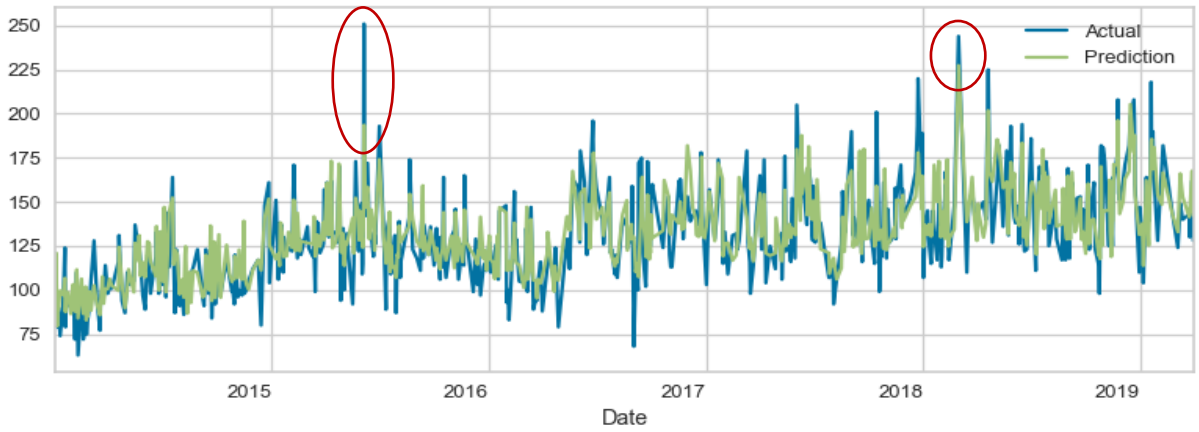


Figure 22 Time series plot of the actual and predicted demand in the Netherlands

The error in the prediction can also be explained by the absence of other factors that may affect the demand. This is demonstrated by the R^2 of 0.69 for the model. This value means that the variance in the current input features can explain 69% of variance of the replacement car demand. There are approximately 30% variance in the target demand that cannot be explained by the current predictors.

Furthermore, through our experiment on a smaller dataset of the demand of replacement cars, we found that a model trained on 2 years dataset have shown an even lower performance in terms of R^2 and MSE, which indicates the lower percentage of variance that can be explained by the model and the existence of occasional errors that are worse than those in the model with 5 years dataset. However, there is not much difference in terms of MAPE. Table 15 shows the comparison across different sizes of the dataset.

Table 15 Performance comparison across different sizes of dataset

Dataset	MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
2017-2019	13.84	9.37	330.32	18.17	12.26	0.5533
2014-2019	12.06	9.48	243.54	15.61	10.10	0.6898

Compared to the prediction of the number of breakdown per day, prediction of the demand of replacement cars has the repairability of the car to consider in addition to general condition like weather and holiday. In our preliminary interview with a domain expert, conditions of cars seem to be an important factor that affects the request of replacement cars. However, the replacement car service is provided at the basis of membership instead of entitled to a specific car. Customers can use the service regardless of the car they use. Therefore, the number of cars with certain conditions (e.g. certain age group, certain models, cars with new technology, etc.) cannot be used as predictors and may contribute to a part of the unexplained variance in the demand of replacement car. There could also be some rare events in the Netherlands that we have not taken into account, that may be the reason of some extreme demand.

Besides the abovementioned issues, the historical data used as the target variable could be inaccurate. There is always a chance that the data is wrong somewhere we are unaware of. Additionally, in the beginning we have made an assumption that the rental car orders represent the

demand of replacement cars. In fact, this may not always be true. Customers are entitled to limited number of times for when they can use the replacement car service. Customers can also opt out to having the replacement cars provided for various personal reasons. Therefore, there may be times where replacement cars are needed following breakdowns that cannot be repaired on the spot but orders are not made. All things considered, we have to bear in mind that there is always a random effect to the demand of replacement cars every day as well that the model cannot possibly predict accurately.

5.2 Demand Prediction per Market Segment

Car group or car type is one of the main characteristics of a rental car request (Oliveira, 2017a) and is one of the main factors to consider in rental car demand forecast (Fink & Reiners, 2006). The unavailability of a certain car group often puts car rental companies in a position where they have to offer cars from higher level group with the same price to avoid lost sales, or cars from lower level group with a discounted price (Oliveira, 2017a), leading to the need of forecast at car group level. A similar strategy applies to a replacement cars company like Logicx. Out of consideration for customer satisfaction, Logicx can upgrade the level of replacement cars at their own cost when cars of the same group as the customers' cars are not available. Likewise, cars from a lower level can be offered if the customers are willing to accept. Therefore, it is vital to have an insight of the right number of cars from different car groups in addition to the number of cars in general.

Logicx categorize cars in several classes (Appendix G.1). These classes can further be grouped into several bigger categories for planning purpose (Appendix G.2). During the operation, ANWB is obliged to provide Logicx the information of these classes of cars so that Logicx can assign cars from the same class to the customers'. However, only 3% of the total orders in 2017-2018 have the required information available in the database. Due to the unavailability of the data at the moment, another alternative of a deeper level forecast that can help company in their planning for different car groups is the demand forecast per market segment. In general, market segmentation in car rental business is inherent in different car types since the demand for certain car types typically falls into specific market segment (Geraghty & Johnson, 1997). In ANWB and Logicx case, there are two major market segments, namely B2B and B2C segments. By default, customers from B2C segment are provided with cars from class C regardless of the class of their cars. Thus, predicting the demand of replacement cars for B2C segment would already provide an insight of the demand of class C cars. In the following sections, we discuss the results of the exploratory data analysis and demand prediction for the demand of replacement cars for the B2B and B2C segments.

5.2.1 Data Preparation and Exploration

Orders for replacement cars are grouped by the customers' market segment (i.e. B2B and B2C segment). Figure 23 shows the demand of replacement cars from ANWB per market segment. In average, there are more demand from B2C segment, in fact an average of 83.35 compared to 49.43. Furthermore, the demand from B2C segment have been gradually increasing over the last 5 years, while the demand from B2B segment have seen an apparent drop around 2016.

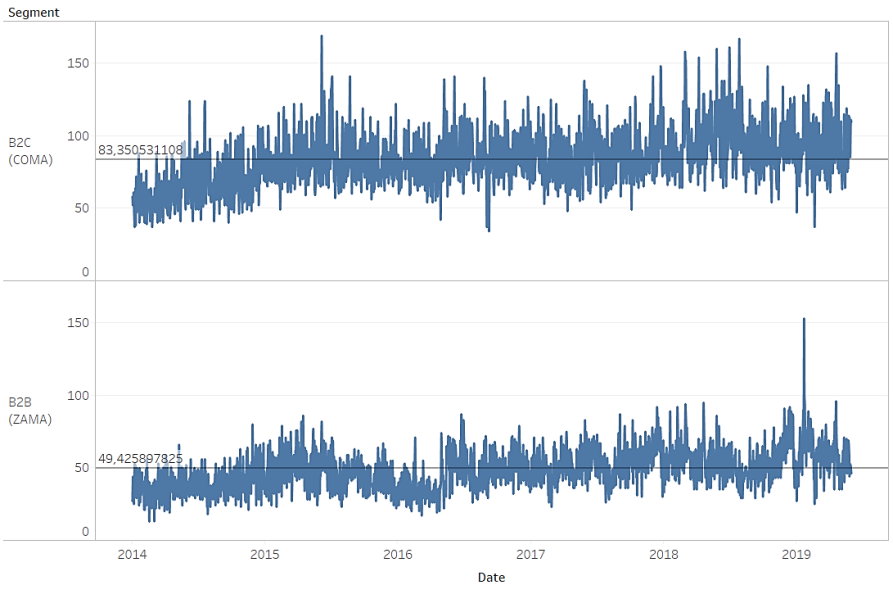


Figure 23 Daily Demand of rental cars per market segment

Besides the trend, another prominent feature is that the B2C segment have seen more fluctuation and extreme peaks every now and then. On the other hand, demand from B2B are more stable in the daily basis, except for one evident outlier in 2019. Figure X shows the different spread of the demand from both segments.

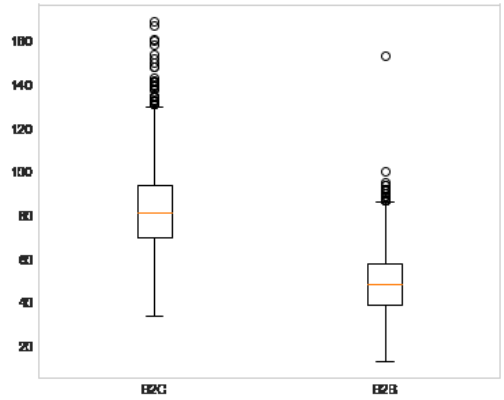


Figure 24 Boxplot of daily rental car demand per market segment

Applying the same data preparation steps as the demand prediction in the Netherlands, we create a dataset for each segment with the same input features as the high level prediction. Furthermore, despite the existence of an obvious outlier candidate in B2B segment in 2019, we keep this data in the initial analysis and treat the outliers as possible extreme values that may be explained by the input features.

5.2.2 Model Performance per Market Segment

B2C

With the new dataset for each market segment, we build the models again using the selected machine learning algorithms on a shuffled training-test set. Table 16 shows the performance of each model for B2C segment on the test set (See Appendix F.4 for a complete performance on both training and test set).

Table 16 Model performance for B2C segment

	MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
Gradient Boosting	9.6592	12.4034	151.5030	12.3086	7.9630	0.6197
XGBoost	9.7015	12.5307	152.9030	12.3654	8.4889	0.6162
Ridge	9.8380	12.7097	155.4410	12.4676	8.3329	0.6099
Lasso	9.8215	12.7211	156.5600	12.5124	8.4838	0.6070
SVR (RBF)	9.9572	12.7221	158.2500	12.5798	8.6607	0.6028
SVR (Linear)	9.9458	12.7792	159.2220	12.6183	8.4226	0.6004
Random Forest	9.9276	12.7629	164.5260	12.8268	8.3224	0.5870
Linear Regression	10.4109	13.6021	183.2190	13.5359	8.8185	0.5401

For B2C segment, the best performing model is Gradient Boosting with an mean absolute deviation of 9.66 cars per day (12.4% of the actual demand for B2C), followed closely by XGBoost. Figure 25 shows the actual and predicted demand for B2C segment with Gradient Boosting model. From the figure, it can be seen that the model predicted the demand more accurately around the average demand value, with only a couple of cases having high deviation. There is also a tendency to underestimate extreme high demand and overestimate low demand even though it does not apply to all cases.

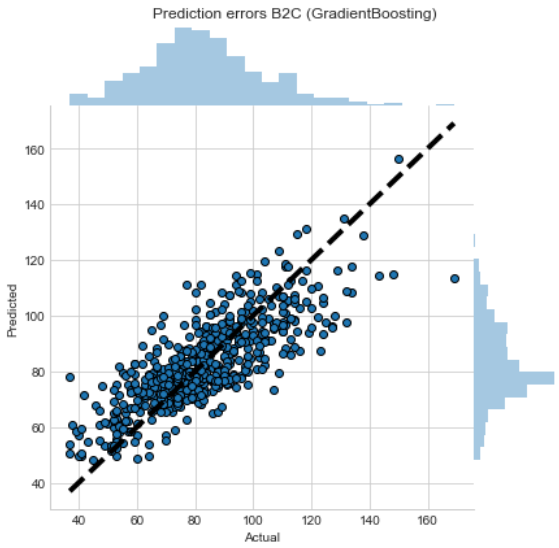


Figure 25 Actual vs predicted demand for B2C segment

In Figure 26, we can observe that similar to the demand prediction for the Netherlands in general, the model cannot accurately predict an extreme demand that has a high difference with the demand around the period, such as the highest demand in 2015 and the lowest demand in 2016.

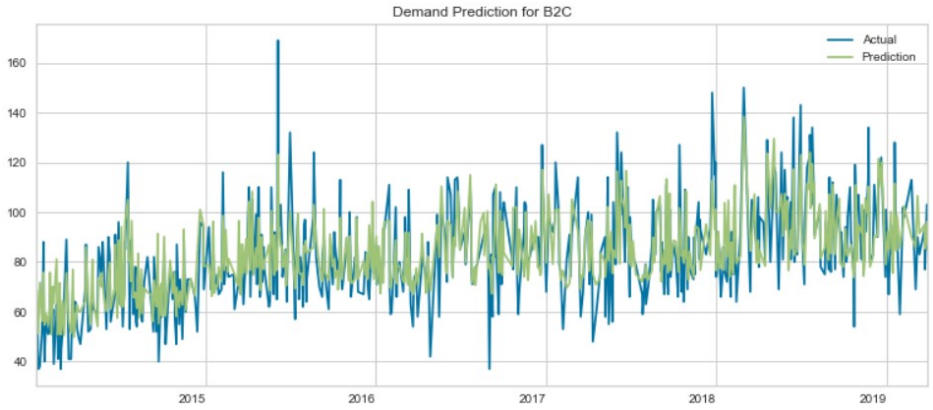


Figure 26 Time series plot of the actual and predicted demand for B2C segment

B2B

The same steps were applied to train and test the models on the B2B dataset. Table 17 shows the performance of each model for B2B segment on the test set (See Appendix F.5 for a complete performance on both training and test set).

Table 17 Model performance for B2B segment

	MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
SVR (RBF)	6.8411	14.7507	79.5612	8.9197	5.2753	0.5649
Lasso	6.8949	15.1608	80.6627	8.9812	5.3371	0.5588
Ridge	6.9132	15.1636	80.7130	8.9840	5.4153	0.5586
SVR (Linear)	6.9435	15.1708	81.6669	9.0370	5.7921	0.5534
XGBoost	7.1644	15.6546	83.1704	9.1198	5.7477	0.5451
Gradient Boosting	7.1470	15.6392	84.0733	9.1692	5.9011	0.5402
Random Forest	7.1360	15.6676	85.6179	9.2530	5.7273	0.5317
Linear Regression	8.0948	18.0444	105.2700	10.2601	6.7425	0.4243

In general, the non tree-based models outperformed the tree-based model on this dataset. It indicates that the dataset may have different characteristics and underlying relationships of the features that make it better explained by regularized models such as SVR, lasso, and ridge regression. For this dataset, we obtained a mean absolute error of 6.8 cars a day with the SVR RBF kernel model. Considering the actual demand of B2B segment, this value results in a higher relative error, as exhibited by the 14.78% MAPE compared to 12.4% for B2C and 9.66% for the high level demand in general. Figure 27 further shows that the predictions for B2B segment are less precise compared to B2C and general predictions. The model underestimated the high demand in general, regardless of the period/year, as depicted more clearly in Figure 28.

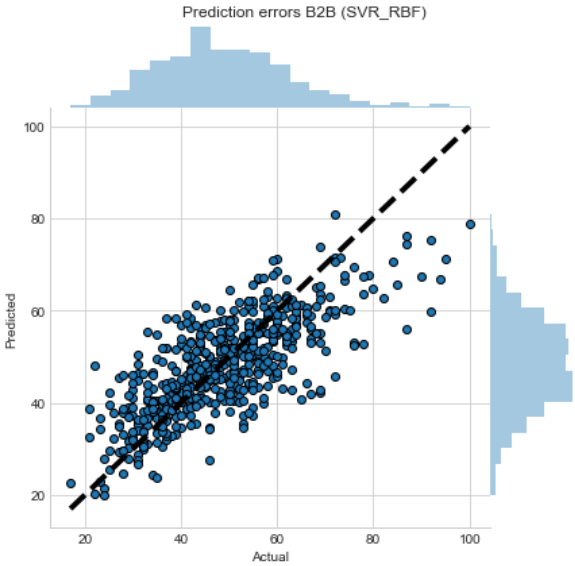


Figure 27 Actual vs predicted demand for B2B segment

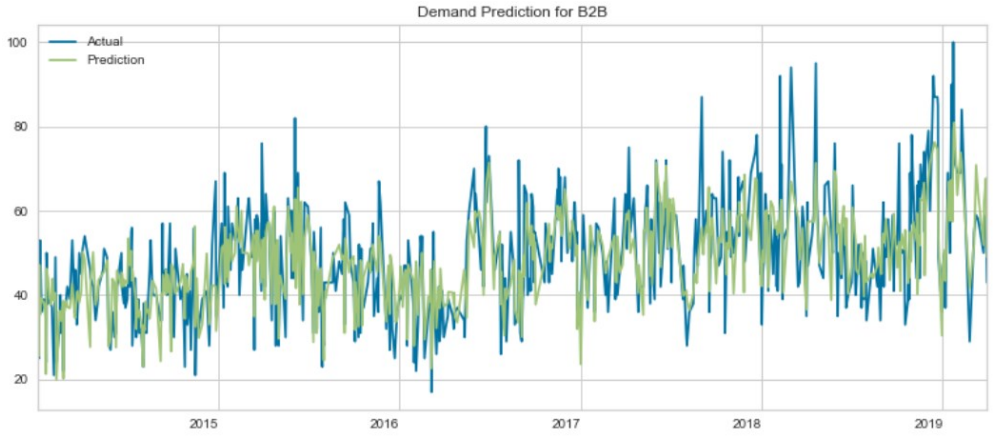


Figure 28 Time series plot of the actual and predicted demand for B2B segment

To summarize, the performance of the models become lower as we go into a deeper aggregation level with regards to the product type, in this case a market segment level. Table 18 summarizes the comparison of the performances. The model for B2C segment produced a better performance compared to the B2B segment when we observe their relative errors to their own actual demand values, which indicates that the fluctuation in B2B demand is less predictable. Furthermore, the R^2 scores suggest that there is less percentage of variance of the B2C demand, and even less of the B2B demand, that can be explained by the same input features used to predict the demand in general. It shows that the same set of features may be good predictors for a certain aggregation level but less so for a different prediction level and product type. Therefore, adding features that are more specific to the case may be necessary besides the features used for the high level.

Table 18 Comparison of the performance of the best model per market segment

Segment	Model	MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
All	**XGBoost**	12.0608	9.4905	241.7970	15.5498	10.1423	0.6920
B2C	**Gradient Boosting**	9.6592	12.4034	151.5027	12.3086	7.9630	0.6197
B2B	**SVR (RBF)**	6.8411	14.7507	79.5612	8.9197	5.2753	0.5649

5.3 Demand Prediction per Province

In addition to the high level prediction (i.e. demand prediction for the whole Netherlands), deeper spatial aggregation levels could be more suitable and beneficial for operational planning. For example, with a good prediction per Logicx rental location, ANWB can distribute the cars more accurately. It is also interesting to investigate to what aggregation level a demand is feasible to predict using machine learning techniques. Furthermore, it is possible that the demand of replacement cars are more predictable in certain locations and are less predictable in others. This section discusses the result of the demand prediction per province as one of the variation of spatial aggregation level for the prediction.

5.3.1 Data Preparation

Rental car orders are mapped to one of the 12 provinces of the Netherlands based on the coordinates of the breakdown. We use a geographical data[3] of the provinces of the Netherlands containing attributes such as central coordinate and sets of coordinates that define a polygon (area) for each province. Then, a rental car order from a breakdown that happens inside the area of a province will be marked as a demand for the province.

There are two types of coordinates data in the ANWB databases, which are the GPS coordinate (i.e. longitude and latitude) and the *Rijksdriehoeks* or RD coordinate (i.e. X-Y points). Data from the old system (i.e. system used to record the orders before 2017) stored the RD coordinates of the breakdown locations but only 25% of these records have the translation to the GPS coordinates. As the external data sources (i.e. geographical data of weather stations and provinces) contain only GPS coordinates and the data processing steps carried out up to this point are constructed to use GPS coordinates, an extra step is required to convert the RD coordinates of all rental car orders into GPS coordinates. For this study, we decided to first exclude the older data and proceed with the data from the second quartile of 2017 (after a full migration to the new system) that contains complete GPS coordinates of the breakdown locations. Figure 29 shows the total demand per province from April 2017 to March 2019.

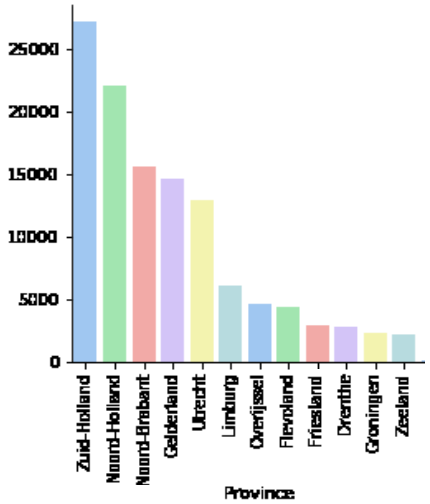


Figure 29 Total demand per province April 2017-March 2019

[3] Hietbrink, Joost. "GeoJSON Data of The Netherlands." We Build Internet, 9 July 2015, https://www.webuildinternet.com/2015/07/09/geojson-data-of-the-netherlands/provinces.geojson

In Figure 29, it can be seen that the demand of rental cars vary per province, with Zuid-Holland having the highest demand over the last two years. Figure 30 further illustrates the distribution of the demand of rental cars per day in each location. In the less busy locations, the demand seem to be quite stable, as shown by the difference between the first and third quartile of the demand that are less than 5 cars, with the existence of some high demand identified as potential outliers. On the other hand, the demand per day for the busier locations such as Zuid-Holland and Noord-Holland vary quite a lot. There are also quite a number of high demand and a few low demand suspected as outliers. With the differences in the characteristics of the demand value for each province, we will build one model for every province.

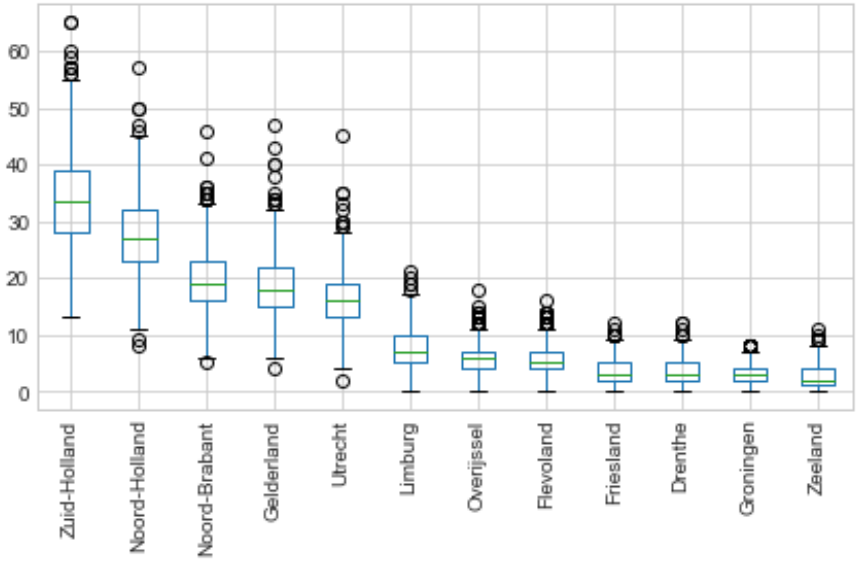


Figure 30 Boxplot of daily rental car demand per province

To build a prediction model specifically tuned for each province, we create a separate dataset for each province. The same input features as the high level prediction are used. We use the same steps to create the features for each dataset. The only difference is on the data that are used for the weather features. Previously, we take the data from one weather station (i.e. De Bilt weather station) to represent the weather data for the whole Netherlands. For the lower spatial level, in this case the province level, we map a weather station to each province based on the euclidean distance between the weather station location and the center of the province. The closest weather station to the center of a province is then assigned for the province and its weather data are used as the input data for the dataset of the province. The following section discusses the selected models for every province and the performance of each model trained on different datasets for different provinces.

5.3.2 Model Performance per Province

Table 19 shows the performance of the best performing model for each province. Out of 8 machine learning models trained on shuffled training set (i.e. simple linear regression, lasso, ridge, SVR linear, SVR RBF, Random Forest, sklearn gradient boosting, and XGBoost), Ridge regression is selected for 5 provinces, while Random Forest, XGBoost, SVR with linear kernel and Lasso regression are selected for the other provinces. The results show that the best prediction performance was achieved by the province with the highest average and total demand, which is Zuid-Holland. The model resulted in an average of 5.37 cars error per day, which is 17.54% of the actual demand per day in average. This performance is lower than those of the prediction models at country level. The results further show that the lower the average daily demand of a province is, the lower the model performance it tends

to get. For instance, the prediction models for Gelderland, Utrecht, and Noord-Brabant that have an average demand of 16-20 cars, resulted in MAPE scores around 23%, which is lower than the MAPE of Zuid-Holland and Noord-Holland. Likewise, the average percentage error of provinces with average daily demand of 7.57 or lower, are all higher than 39%.

Table 19 Model performance per province

Province	Selected Model	MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2	Average demand
Zuid-Holland	Ridge	5.37	17.54%	45.07	6.71	4.49	0.2897	34.09
Noord-Holland	XGBoost	4.59	18.46%	31.76	5.64	3.85	0.2047	27.71
Gelderland	Random Forest	3.60	22.55%	22.66	4.76	3.01	0.2047	18.27
Utrecht	XGBoost	3.59	23.08%	20.26	4.50	3.05	0.1021	16.14
Noord-Brabant	Lasso	4.11	23.27%	25.25	5.03	3.53	0.1058	19.59
Flevoland	Ridge	1.71	39.29%	4.69	2.16	1.31	0.0174	5.46
Limburg	Random Forest	2.41	40.05%	9.76	3.12	2.02	0.1549	7.57
Overijssel	Random Forest	1.93	51.24%	5.95	2.44	1.54	0.0864	5.72
Groningen	SVR (linear)	1.23	58.60%	2.38	1.54	1.05	-0.0058	2.89
Drenthe	Ridge	1.59	61.27%	4.20	2.05	1.31	0.0563	3.40
Zeeland	Ridge	1.39	66.24%	3.00	1.73	1.19	0.1506	2.70
Friesland	Ridge	1.74	70.63%	4.86	2.20	1.56	0.0261	3.67

Besides the prediction errors, the R^2 scores are considerably lower than the scores for the high level prediction, in general as well as per market segment. The highest R^2 for province level is 0.2897 which implies that at most, only 28.97% of the variability of daily demand can be explained by the input features (predictors) collectively. These low R^2 scores may also be a consequence of the low range of demand values for each location. Low range of values gives a low standard deviation, which in turn is likely to result in a rather low R^2 score due to the difficulty of obtaining an even lower residual for the prediction. In Table 19, it can also be seen that the prediction model for Groningen resulted in a negative R^2. It indicates that the predictions by the model perform worse than the predictions by the mean value of the demand. Figure 31 shows the demand predictions plotted against the actual values of the demand for the province of South Holland and Groningen. We can see that for South Holland, the predictions still form a correlation to the actual demand even though there are high deviations for many cases. Meanwhile, for Groningen, the predictions form a horizontal trend line near the average demand value of 2.89 cars per day but with an even higher sum of squared error compared to the sum of squared of error from a constant line of the mean value.

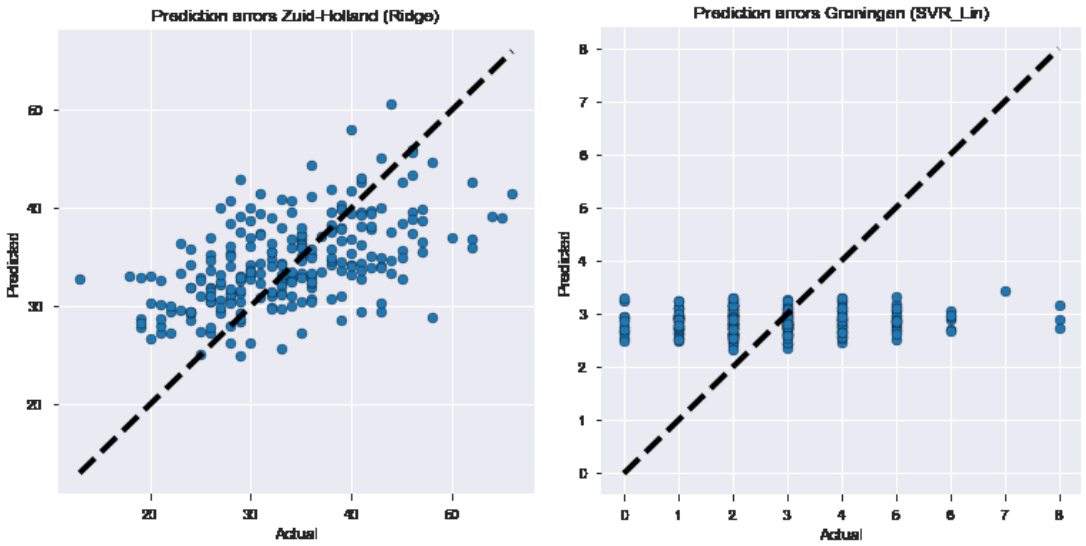


Figure 31 Actual vs predicted demand for Zuid-Holland and Groningen province

Overall, the provinces appear to have varying distributions and range of the demand of replacement cars and hence different performances of the prediction models. The prediction models in some provinces are more acceptable than in some others. Provinces with low average demand tend to be more difficult to predict, partly due the low range and low variation of the demand values. Furthermore, the performance of the prediction models for provinces with average demand lower than 6 cars tend be similar to or even worse than the predictions using the mean value of the demand.

5.4 Demand Prediction per Work Area

In the previous section, we have seen the results of the demand prediction at province level. Another potential spatial aggregation level for predicting the demand of replacement cars is the ANWB work area, which is the division of the Netherlands into several area in which ANWB manage the roadside assistance operations. In the same way as the demand prediction per province, the following section explains the result of our study on the daily demand per work area and their predictability.

5.4.1 Data Preparation

ANWB divided the Netherlands into 33 work areas as depicted in Figure 32. For every breakdown, ANWB define the work area where the breakdown happens and record the information in the database. We use this information to aggregate the total demand per day for all work area. There are missing work area data in 128 breakdown cases during April 2017 to March 2019. This value amounts to 0.0011% of all breakdown cases during the period. In addition, there are 64 cases where the work area is undefined, labeled as “Onbekend” and “Niet gevonden” in the database. Since the occurrences of these missing and undefined cases are really low, we exclude the records with unknown work area value when aggregating the demand. Figure 33 shows the total demand per work area from April 2017 to March 2019.

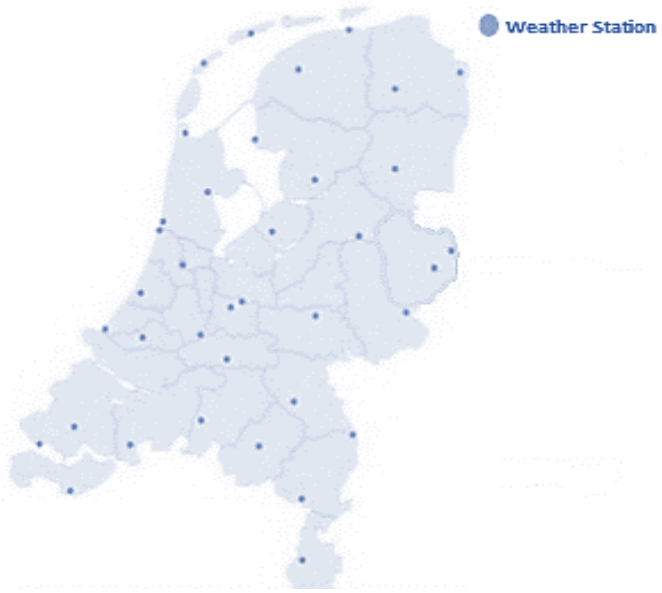


Figure 32 ANWB Work Area

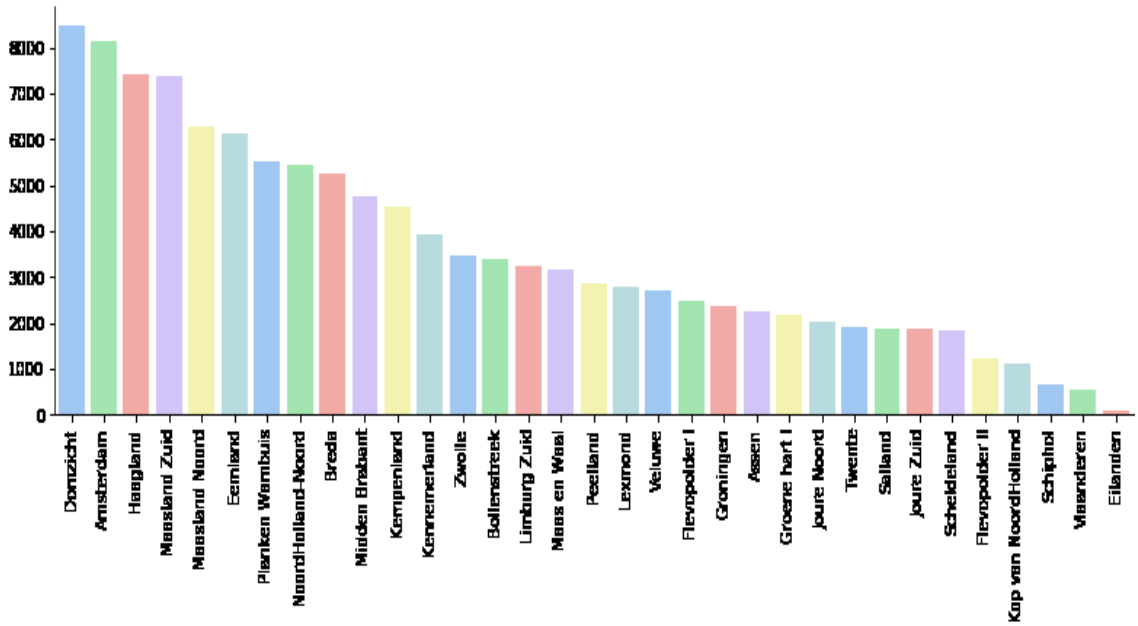


Figure 33 Total demand per ANWB work area April 2017-March 2019

Figure 33 shows that, similar to the demand per province, the demand of rental cars also vary per work area. Domzicht area has the highest demand in total over the last two years, followed closely by Amsterdam area. The average demand in these two areas are in fact nearly the same. However, in Figure 34, we can see that Domzicht area has some days with significantly higher demand compared to the other days in Domzicht or any daily demand in Amsterdam. We can also observe that in some work areas, the outliers are more apparent than in the others. Moreover, some work areas also seem to be more skewed than the others. For Eilanden, it is evident that there is no demand of replacement cars in most days, while some outlier days only have either 1 or 2 breakdowns that ended up with a request of replacement cars.

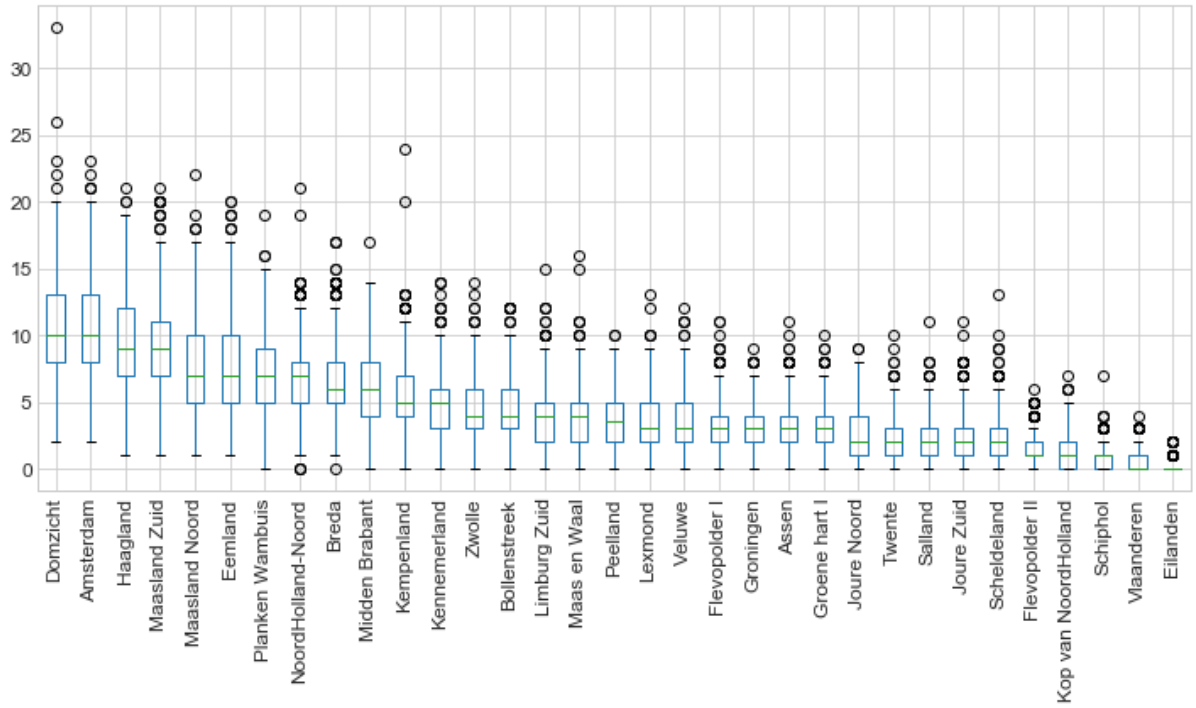


Figure 34 Boxplot of daily rental car demand per ANWB work area

Similar to the demand prediction per province, we will build different model for different work area. Therefore, 33 datasets, one for each work area, are created using the same input features as the previous prediction models. Similar to the dataset for province level, we use weather data from weather stations closest to the center of each work area based on the euclidean distance. The rest of the features are created in the same manner as the higher level datasets. In the following section, we describe the selected models for every work area and the performance of each model trained on different datasets for different work areas.

5.4.2 Model Performance per Work Area

For every work area, eight machine learning algorithms were used to train the prediction model on the new dataset per work area. The performance on the test set were compared and the best performing model were selected. Table 20 shows the performance of the selected model for each work area.

Overall, the MAE scores for the demand prediction at work area level are ranging from 0.24 to 2.82 cars. It may seem to be quite low. However, the average demand per work area is also rather low. They vary from 0.13 cars in Eilanden to 10.63 cars in Domzicht. In terms of MAPE, the lowest percentage error is shown by the model for work area with the highest average demand and it tends to be higher the less the demand in a work area is. This is in line with the results from the demand prediction per province.

However, the highest R^2 is not obtained by the best model in terms of error rate (i.e. the prediction model for work area with the highest demand). The highest R^2 is produced by the prediction model for Scheldeland, followed by NoordHolland-Noord, with R^2 of 16.21% and 14.97% respectively. This is likely due to the R^2 being related to the MSE loss function which makes it more sensitive to occasional large errors. Figure 35 visualizes the actual and predicted demand for work area with the lowest MAPE, Domzicht, and work area with the highest R^2, Scheldeland. It can be seen that the predictions for Domzicht, including those for the extreme high demand, spread around the average

demand value more closely. Moreover, the prediction for the highest demand has a higher deviation than if the demand is predicted at the demand value which has worsened the R^2 of the predictions. While they appear to lead to a better average percentage error, they resulted in an R^2 close to zero. On the other hand, predictions for Scheldeland, especially for the potential outliers, are closer to the diagonal line than the average demand line (and hence the higher R^2). Besides, the average demand of Scheldeland is very low, in fact 2.31 cars per day. Therefore, even with deviation averaging around 1-2 cars, the MAPE score is bound to be high.

Table 20 Model performance per work area

Work Area	Selected Model	MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2	Average demand
Domzicht	Random Forest	2.75	29.52%	12.35	3.51	2.44	0.0520	10.63
Amsterdam	SVR (RBF)	2.59	31.79%	10.04	3.17	2.25	0.0769	10.19
Haagland	Lasso	2.53	38.14%	10.37	3.22	2.06	0.0879	9.30
NoordHolland-Noord	Lasso	1.96	38.32%	6.13	2.48	1.66	0.1497	6.84
Maasland Noord	SVR (linear)	2.39	38.40%	8.57	2.93	2.22	0.0468	7.84
Eemland	XGBoost	2.26	39.49%	8.24	2.87	1.86	0.0861	7.68
Maasland Zuid	LinearReg	2.82	41.00%	11.51	3.39	2.56	0.1110	9.25
Breda	Lasso	2.15	45.53%	7.14	2.67	1.80	0.0519	6.60
Planken Wambuis	Lasso	2.28	48.14%	8.33	2.89	1.99	0.0609	6.93
Kempenland	Random Forest	2.05	51.90%	6.26	2.50	1.78	0.0224	5.69
Zwolle	Lasso	1.54	52.42%	3.95	1.99	1.30	0.0611	4.35
Midden Brabant	Lasso	2.00	54.11%	6.33	2.52	1.66	0.0362	5.97
Kop van NoordHolland	Ridge	0.99	55.71%	1.51	1.23	0.72	0.0342	1.39
Flevopolder II	Ridge	1.06	55.81%	1.67	1.29	0.70	0.0071	1.53
Flevopolder I	Ridge	1.28	56.20%	2.63	1.62	1.08	0.0099	3.11
Veluwe	Random Forest	1.50	57.44%	3.66	1.91	1.11	0.0520	3.39
Schiphol	SVR (RBF)	0.70	58.50%	0.79	0.89	0.80	0.0041	0.80
Limburg Zuid	Ridge	1.71	58.57%	4.98	2.23	1.46	0.0550	4.06
Kennemerland	Ridge	1.88	58.89%	5.54	2.35	1.60	0.0204	4.90
Scheldeland	XGBoost	1.32	58.94%	2.93	1.71	1.05	0.1621	2.31
Twente	Ridge	1.31	60.25%	2.73	1.65	1.11	0.0446	2.40
Groningen	SVR (linear)	1.23	61.64%	2.33	1.53	1.07	-0.0189	2.98
Lexmond	Ridge	1.65	61.70%	4.26	2.06	1.40	0.0102	3.51
Assen	Ridge	1.39	61.76%	2.98	1.73	1.24	0.0508	2.84
Maas en Waal	Ridge	1.73	61.89%	4.81	2.19	1.44	0.0652	3.99
Salland	Ridge	1.23	62.68%	2.23	1.49	1.12	0.0351	2.36
Joure Noord	Ridge	1.38	63.00%	2.78	1.67	1.36	0.0219	2.56
Bollenstreek	SVR (RBF)	1.76	64.42%	4.84	2.20	1.51	-0.0062	4.28
Groene hart I	Lasso	1.42	65.53%	3.36	1.83	1.17	0.0178	2.75
Peelland	GradientBoosting	1.50	67.01%	3.60	1.90	1.27	0.0807	3.59
Joure Zuid	Lasso	1.28	67.56%	2.48	1.57	1.13	0.0077	2.35
Vlaanderen	Lasso	0.62	71.25%	0.52	0.72	0.59	0.0239	0.67
Eilanden	Ridge	0.24	97.91%	0.17	0.41	0.11	0.0202	0.13

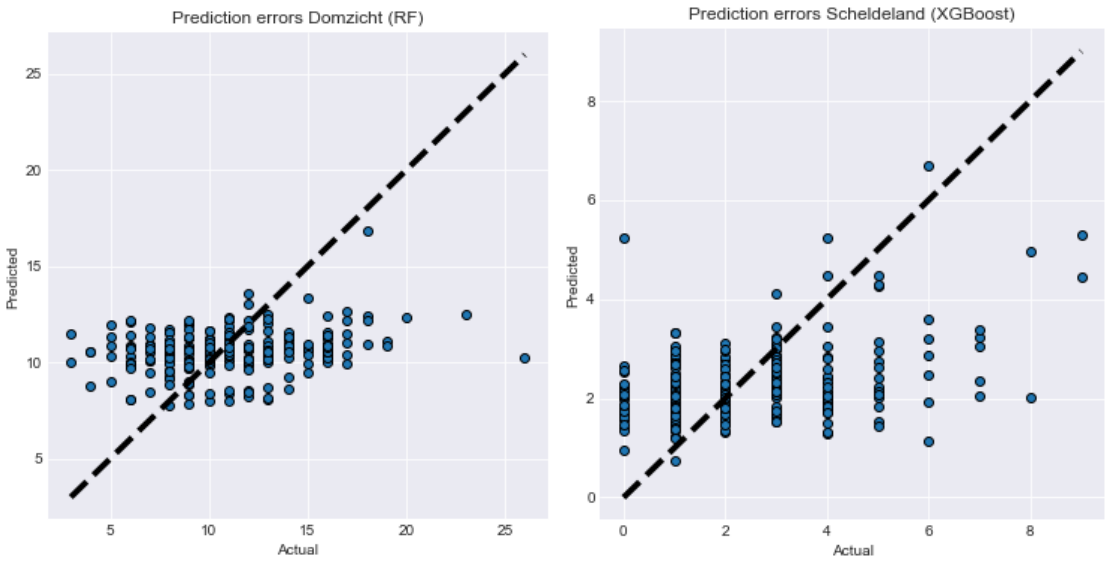


Figure 35 Actual vs predicted demand for work area

With almost 60% average percentage error for Scheldeland that only has 2.31 average cars per day, it is even more difficult to get a good performance for work area with less average demand per day. The results show that for these work area, the percentage error are all higher than 60%. Even worse, for Vlaanderen and Eilanden with average demand of 0.67 and 0.13 respectively, the predictions are simply unusable. This is due to the intermittent demand pattern (i.e. demand where there are many zero values with very little occurences of non-zero demand) occuring in these area. Figure 36 illustrates how there are only a few days with either 1 or 2 demand of replacement cars. Due to the rare occurrences and the low range of demand values, the machine learning models along with the input features that are not specifically designed to handle this kind of demand, only managed to predict the demand at less than 0.25 cars most of the times.

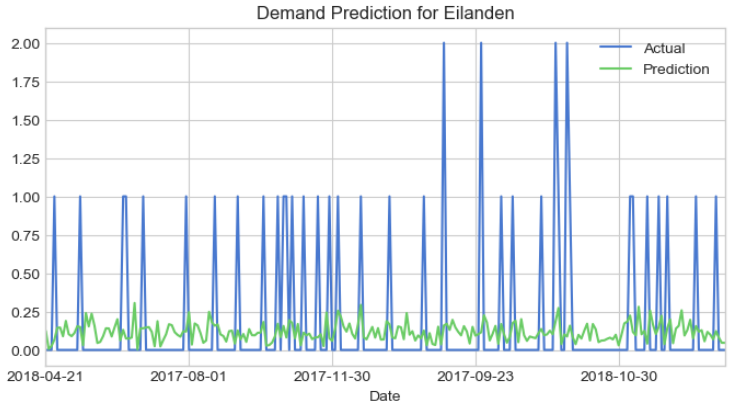


Figure 36 Time series plot of the actual and predicted demand for Eilanden

To sum up, demand prediction at work area level shows a similar results to the demand prediction at province level where the range of the demand at one area contributes to the difficulty to predict the demand. Furthermore, some work areas have a really low average demand, and some of them can be categorized as intermitten demand where the prediction models basically predict the demand at near zero value due to the frequency of zero actual demand. In addition, the percentages of explained variance in the demand at work area level are even lower than the higher level predictions, with the highest being 16.21% compared to provinces' 29.87% and country's 69.2%,

partly due to the more narrow range of demand values per work area that leads the models to predict around the average value.

5.5 Demand Prediction per Rental Location

The ultimate goal of predicting the demand of replacement cars is to have the right number of cars available in the right rental location at the right time. To achieve this, one of the most straightforward approachesa is to predict the demand of rental cars per rental location. In this section, we discuss the demand prediction at this spatial level and compare it in comparison to the demand prediction at the other spatial aggregation levels.

5.5.1 Data Preparation

Logicx fleet for the Netherlands are distributed in 85 Logicx and partners locations all over the country. Customer can pick up and return the cars at these locations. In some cases, replacement cars can also be transported to the customers' desired place by Logicx and picked up at an agreed location at the end of rental period, provided that the customers have the roadside assistance package that allow this service.

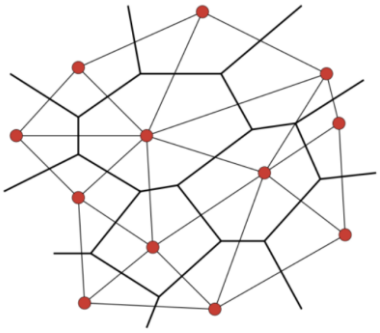
Logicx assign the pick up location of the replacement car based on the following aspects:

- Distance to breakdown location
- Availability of the rental cars at the location
- Customer's preference

For ANWB road patrols, the most convenient location is usually the closest location to the breakdown location since they can bring the customers the the rental car pick up location in the shortest time possible thus allowing them to continue their work with other breakdown cases. However, if the right cars are not available at the closest location, another location that is further away will be selected. In some cases, customers can also choose to pick up the rental cars at another location according to their preferences, such as the one near their home or destination, or the one along the way to the destination.

The information available in ANWB database regarding the pick up location is the name of the city. There is no information about the exact Logicx rental location that provides the car. Therefore, to get the number of replacement car to predict per Logicx location, it is necessary to assign a Logicx location for each breakdown case with replacement car order. Due to the many possible variables that affect the decision of which rental locations the customers need to pick up the cars at, assuming that the closest location is acceptable for both customers and road patrols (i.e. there is no special request from the customers), as well as that Logicx always has the right car at the right location (which is the ultimate goal of the prediction), we used the the following rule:

A rental car order from a certain breakdown location is a demand for the closest rental location regardless of the customers preferences and the capacity and availability of the cars at the rental location.



Thiessen polygons (thick lines)

Delaunay triangulation (thin lines)

Figure 37 Thiessen polygons and delaunay triangulation (Burrough et al., 2015)

To map rental car orders as the demand of exactly one rental location, we divided up the Netherlands using the Thiessen (also known as Dirichlet or Voronoi) polygons of Logicx rental locations. Thiessen polygons are widely used as a method for relating point data to space in geographical analysis and Geographical Information Systems (Burrough et al., 2015). An example of Thiessen polygons is illustrated in Figure 37. Each partition space created using this approach contains a specific data point, in our case a Logicx rental location, and all points that are closer to this rental location than to the other rental locations. In this study, we excluded 2 out of 85 Logicx and partners rental locations, which are:

- *112 Autoberging,* which only serves emergency call centers other than ANWB (i.e. Logicx other customers).
- *ABC Amsterdamse Bergings Combinatie,* which only delivers BMW cars.

Logicx call center can reserve a car for ANWB customers in these two locations. However, they only do this when other issuing locations cannot help, hence the exclusion in this analysis. Figure 38 shows the partition of the Netherlands for 83 Logicx rental locations based on the Thiessen polygons.

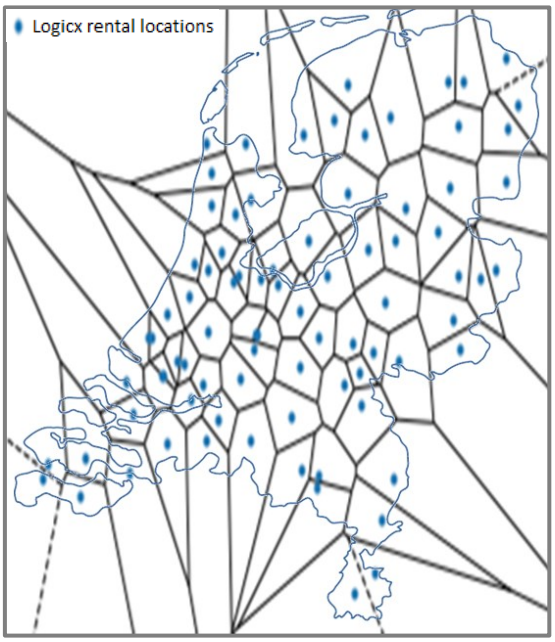


Figure 38 Thiessen polygons of Logicx rental locations

After defining the polygons, the demand can be assigned to rental locations the same way the demand is assigned to provinces, that is, every rental car order that occurs inside one region/polygon is the demand for the rental location corresponding to the polygon. Figure 39 shows the total demand per Logicx rental locations from April 2017 to March 2019. Again, just like the demand at province and work area level, the demand of replacement cars vary per Logicx location, from the highest of 4726 cars in ANWB Servicecentrum Utrect to the lowest of 31 cars in Autoberging Dallinga in the span of 2 years.

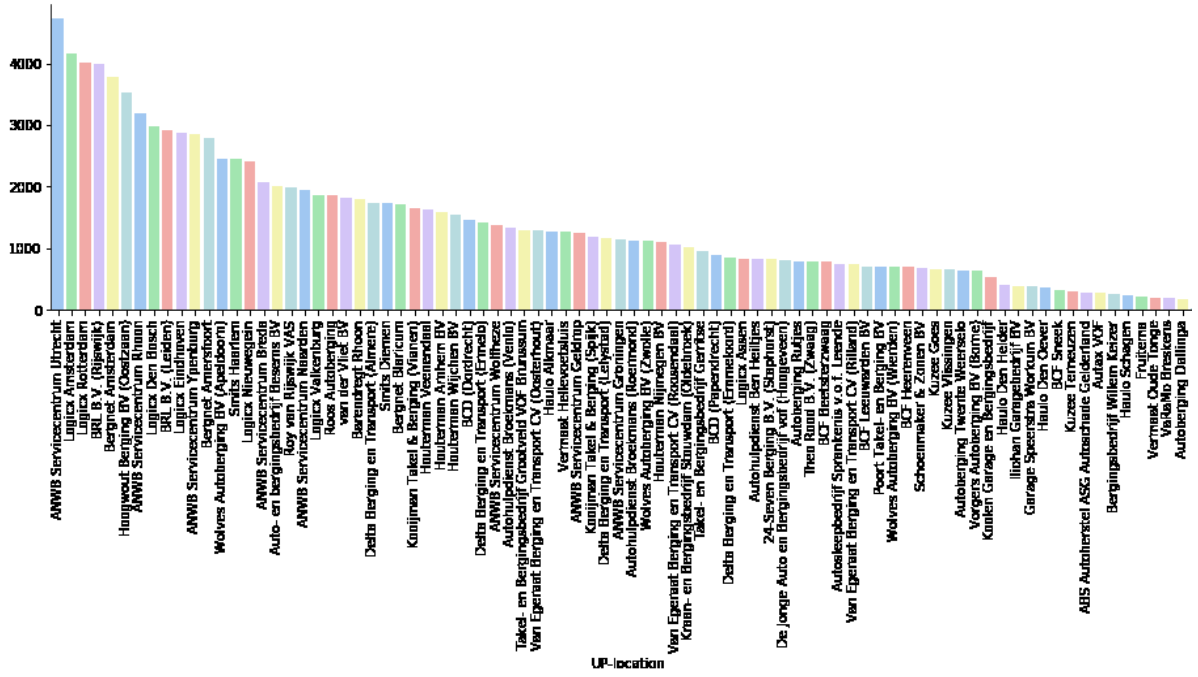


Figure 39 Total demand per Logicx pick-up location April 2017-March 2019

In Figure 40, we can see that more than half of the locations recorded a median value of 0 or 1 car per day which indicates the high possibility of the demand being an intermittent demand. However, in quite a number of locations, the demand can have a median value around 5 cars a day, with some outlier days where the demand can reach up to 17 cars. It is also important to note that the maximum capacity of cars that the locations can hold at one time differs per location. Some locations such as *Haulo Den Helder* and *Autosleepbedrijf Sprankenis v.o.f. Leende* can hold a maximum of 2 cars while some other locations like *BRL B.V. (Leiden)*, *van der Vlier BV*, and *Kuzee Vlissingen* have 100 or higher capacity.

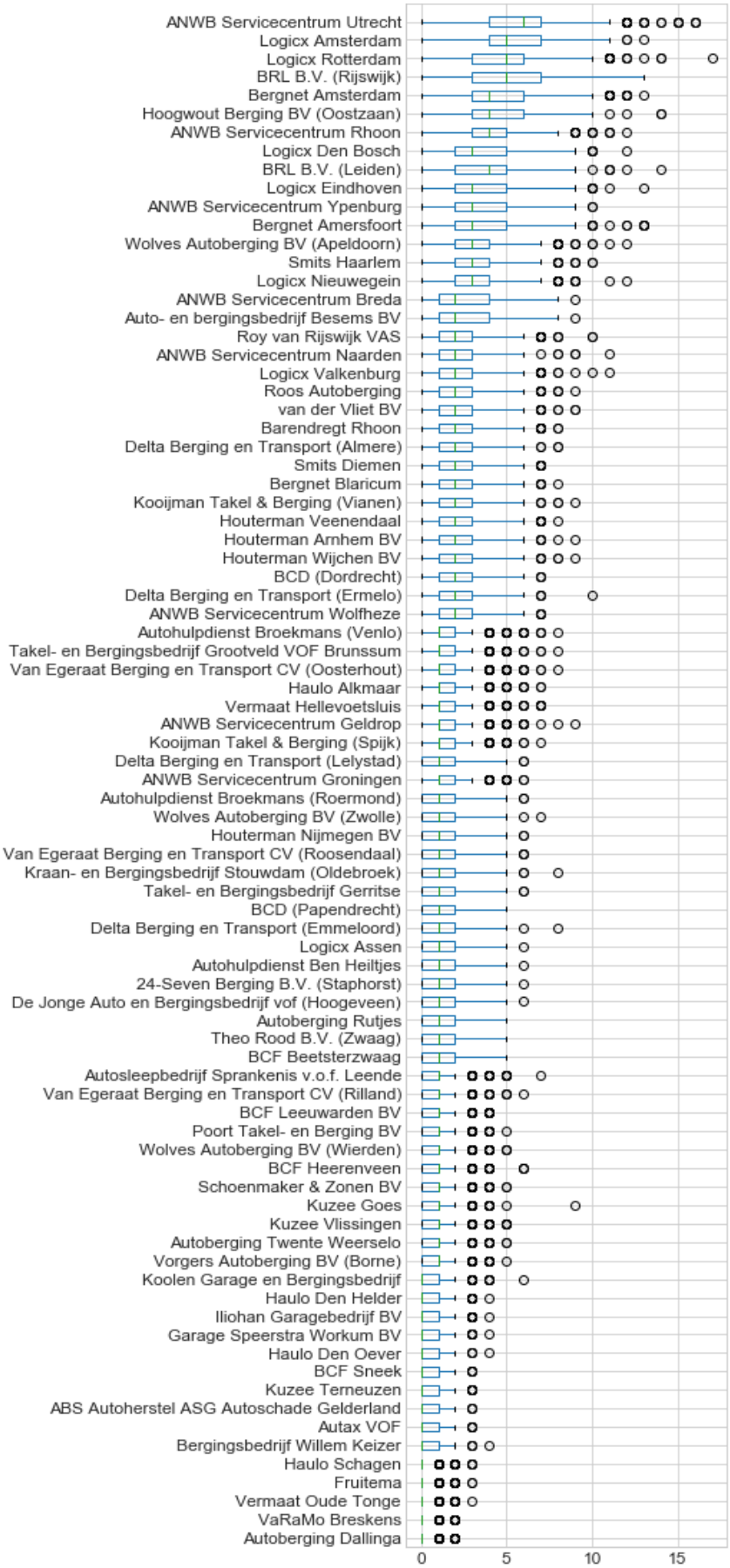


Figure 40 Boxplot of daily rental car demand per Logicx location

Table 21 Rental locations with demand exceeding capacity

Rental location		Average demand	Highest demand	Max capacity
861	ANWB Servicecentrum Utrecht	5.93	16	8
853	ANWB Servicecentrum Rhoon	4.00	12	4
940	ANWB Servicecentrum Ypenburg	3.58	10	5
957	Roy van Rijswijk VAS	2.50	10	6
937	ANWB Servicecentrum Naarden	2.44	11	9
671	Roos Autoberging	2.32	9	6
944	ANWB Servicecentrum Geldrop	1.58	9	6
943	ANWB Servicecentrum Groningen	1.43	6	5
675	Takel- en Bergingsbedrijf Gerritse	1.21	6	5
945	Autosleepbedrijf Sprankenis v.o.f. Leende	0.94	7	2
879	Haulo Den Helder	0.52	4	2

Looking at the extreme demand value per location, there are some cases where the high demand exceeded the maximum capacity of the location, as listed in Table 21. On the one hand, these cases may be actual representations of unavoidable cases where frequent repositioning of cars in a day or repositioning of customers are required when outlier demand takes place at a location. For example, the demand for *Haulo Den Helder* are not more than 1 car 75% of the time, thus having a maximum capacity of 2 cars is reasonable. However, Logicx will still need to redirect customers to a further location when the demand exceed 2 cars in some rare days. On the other hand, these demand-exceeding-capacity cases may have been a result of our approach to strictly divide the area using polygons. For instance, *ANWB Servicecentrum Utrect* can only hold up to 8 cars but has been experiencing demand higher than 8 for a couple of times. In reality, these high number of rental car orders would most likely be assigned to *Logicx Nieuwegein* instead of *ANWB Servicecentrum Utrecht* due to both being less than 2 km apart while *Logicx Nieuwegein* having much higher capacity, in fact a maximum of 40 cars. As the current Logicx process in assigning pick-up locations would be able to deal with this kind of problem, we decided to proceed with this demand aggregation.

Another interesting finding from the demand aggregation at the location level is that the daily demand for every location suggests that a high demand in a certain day in the Netherlands may not be reflected equally in every location. In Figure 41, a high demand with 276 cars appear to result in a high demand for *ANWB Servicecentrum Utrecht* and *Logicx Amsterdam,* with 15 and 11 cars compared to the daily average of 5.93 and 5.2 respectively. On contrary, the demand for *Logicx Rotterdam* is only 2 cars, which is below its average level around 5. For this reason, it is important to build a model for each Logicx rental location using a separate dataset per location. The datasets are prepared in similar fashion to the dataset creation for demand prediction at province and work area level.

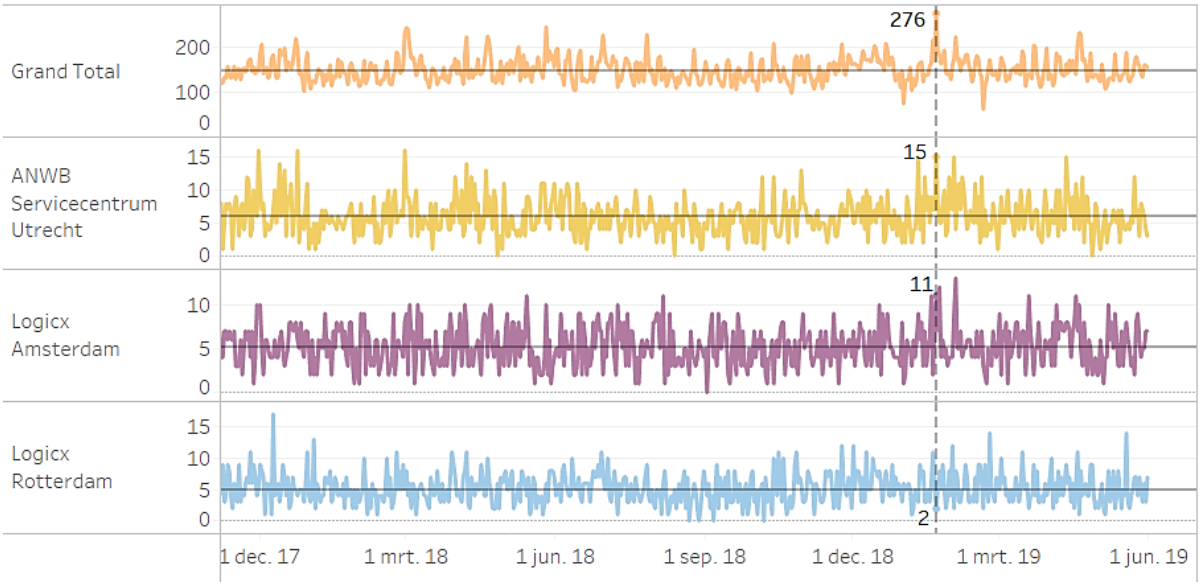


Figure 41 Daily demand of rental cars for 3 rental locations with the highest demand

5.5.2 Model Performance per Rental Location

With average demand ranging from 0.23-5.93 cars per day, the average absolute deviation per day, represented by MAE scores, from the best machine learning model at each location are ranging from 0.33-2.08 cars per day (See Appendix F.6 for detailed model performances of all Logicx and partner rental locations). An error of 0.33 car per day (rounded up to 1) may seem to be quite low. However, performance around this level was resulted from a prediction of intermittent demand that are found in many Logicx rental locations. In day-to-day basis, the model predicted the demand at around average demand value, similar to the result of intermittend demand at work area level (Figure 36).

In terms of percentage error, the lowest score obtained from the predictions at Logicx locations level is no less than 47.03%. Table 22 shows the model performances for 5 locations with the lowest MAPE scores. For predictions at province and work area level, the lowest MAPE scores are dominated by area with the highest average demand. While the same condition still holds at rental locations level to some extend, it is less so than at the higher level. For instance, we can see *Bergnet Blaricum* and *Takel-en Bergingsbedrijf Gerritse* that have an average demand of 2.16 and 1.21 cars respectively, listed in the top 5 locations with lowest percentage error. This is due to the range of demand value that are generally low for all locations. With low range of demand value, similar to what we found in predictions at work area level, there is a tendency of the machine learning models to predict around the average demand value, as seen in Figure 42. As a result, the distribution of the demand plays an important part in the final average of percentage error (MAPE). In Figure 42, we can see that *ANWB Servicecentrum Utrect* (i.e. location with the lowest MAPE and the highest average demand) as well as *Bergnet Blaricum* and *Takel-en Bergingsbedrijf Gerritse* (i.e. locations with MAPE in best 5 but low average demand) are all having the highest frequency of the demand close to the average demand value, thus resulting in a low percentage error.

Table 22 Model performance per rental location - Top 5 based on MAPE

Rental location		Selected Model	MAE	MAPE	MSE	RMSE	MED_AE	R^2
861	**ANWB Servicecentrum Utrecht**	Random Forest	2.08	47.03%	7.03	2.65	1.73	0.0295
615	**Logicx Rotterdam**	Ridge	1.68	49.35%	4.39	2.10	1.42	0.0396
613	**Bergnet Blaricum**	Lasso	1.07	49.57%	1.92	1.39	0.99	0.0143
917	**Logicx Amsterdam**	Ridge	1.74	51.40%	4.89	2.21	1.48	0.0013
675	**Takel- en Bergingsbedrijf Gerritse**	SVR (RBF)	0.91	51.75%	1.47	1.21	0.80	0.0017

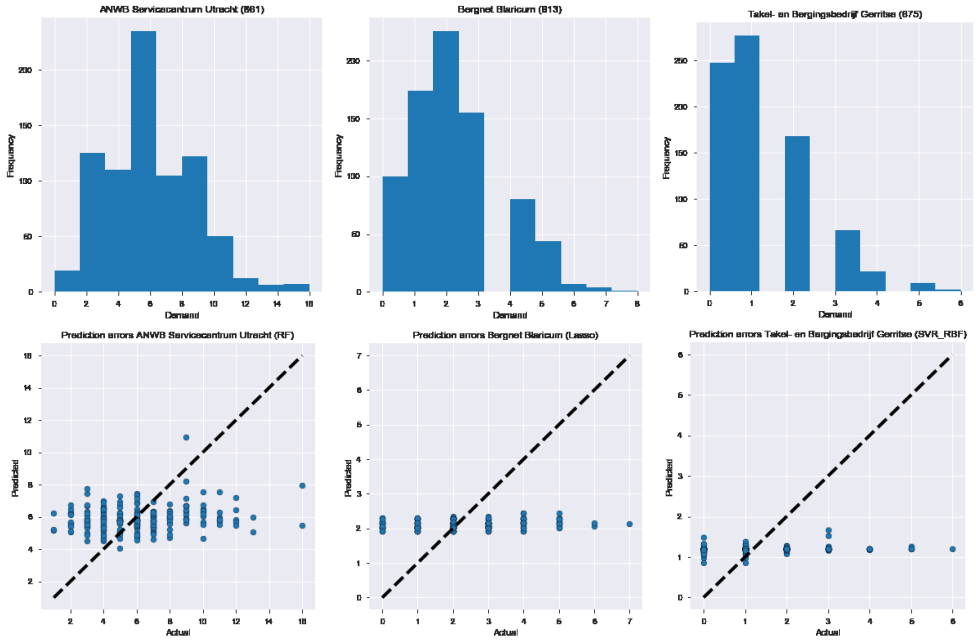


Figure 42 Distribution of demand per rental location

According to the R^2, the highest percentage of explained variance of a demand in a location was obtained from the predictions in *Autohulpdienst Broekmans (Venlo)* using XGBoost, that is an R^2 of 9.37%. The next highest R^2 scores that could be achieved in demand prediction per rental location are 7.6%, 7.28%, 5.81%, and 5.53%. Table 23 shows the performances of locations with the best 5 R^2 scores. This numbers show that there are at least 90% of the demand variance per location that cannot be explained by the current input features that we used. This could be put down to the lack of samples of extreme demand that could lead to a better learning, the absence of more location-specific features that we did not incorporate in the model, or due to randomness of the location of breakdown in general. In addition, we found a lot of negative R^2 scores in many other locations, which suggest that a constant prediction using the average demand value will produce a better prediction performance compared to the current predictions (see Appendix F.6 for the details).

Table 23 Model performance per rental location - Top 5 based on R^2

Rental location		Selected Model	MAE	MAPE	MSE	RMSE	MED_AE	R^2
620	**Autohulpdienst Broekmans (Venlo)**	XGBoost	1.02	55.97%	1.72	1.31	0.86	0.0937
947	**BRL B.V. (Rijswijk)**	Lasso	1.93	54.03%	5.55	2.36	1.64	0.0760
928	**Hoogwout Berging BV (Oostzaan)**	Random Forest	1.64	54.50%	4.06	2.01	1.39	0.0728
955	**Kuzee Goes**	Random Forest	0.75	64.91%	0.86	0.93	0.69	0.0581
631	**Logicx Eindhoven**	Random Forest	1.58	64.11%	3.81	1.95	1.45	0.0553

All in all, we saw that the ability of machine learning models trained on less than two years of data using the current input features, to predict the demand per Logicx rental location, is really limited. The performances of the models are either worse or not significantly better than predictions using average demand values. Furthermore, after looking at the results from various demand aggregation levels, we found that the range of demand values and the distribution of the demand affect the performance of the model. MAPE appears to be a good metric to compare different time series as well as to measure performance in an intuitive way. However, it barely holds any meaning for a really low range of demand values and intermittent demand, as it tends to reach a really large value.

5.6 Prediction Interval

As we cannot expect the model to perform accurately for the whole year, we generate a prediction interval to estimate uncertainty of the prediction. In this study, the following strategies are compared to build the prediction interval.

Approach 1: Prediction interval with constant variance

In this approach, we run 5-fold cross-validation on the training set. Then, the upper and lower bound of the prediction is calculated as follows.

$$Prediction\ interval = \hat{y} \pm Z\ score\ \times\ \hat{\sigma}$$

$\hat{\sigma}$ represents the estimate of the standard deviation of the forecast distribution. For one step ahead forecasting, the standard deviation of the forecast distribution is almost the same as or may be slightly larger than the standard deviation of the residuals, but oftentimes this difference is ignored (Hyndman, 2018). Therefore, we took the RMSE value of the 5 cross-validation splits as an estimate of standard error of the prediction. This approach assumes that the observations are normally distributed. Thus, to generate 95% prediction interval, we used Z-score = 1.96.

As the approach used the variance of the performance of the cross-validation splits, the prediction interval using this approach directly represents the stability of the model when it is trained with different sets of data. Figure 43 illustrates the prediction interval performed on two months car rental demand prediction. The results show the MPIW value of 64.89 which means the demand of replacement cars on a day is the predicted demand ±32 cars. This is a relatively large interval width. However, it comes with a good coverage of actual value, in fact a PICP value of 95.83%. It shows that with 95% prediction interval (which means we can be 95% certain that the true value is within the interval), the prediction interval actually includes the true value 95.83% of the time. This level of accuracy is very good since there is less than 5% chance that the interval does not contain the actual value. However, the constant interval is too large to make it effective for the practice.

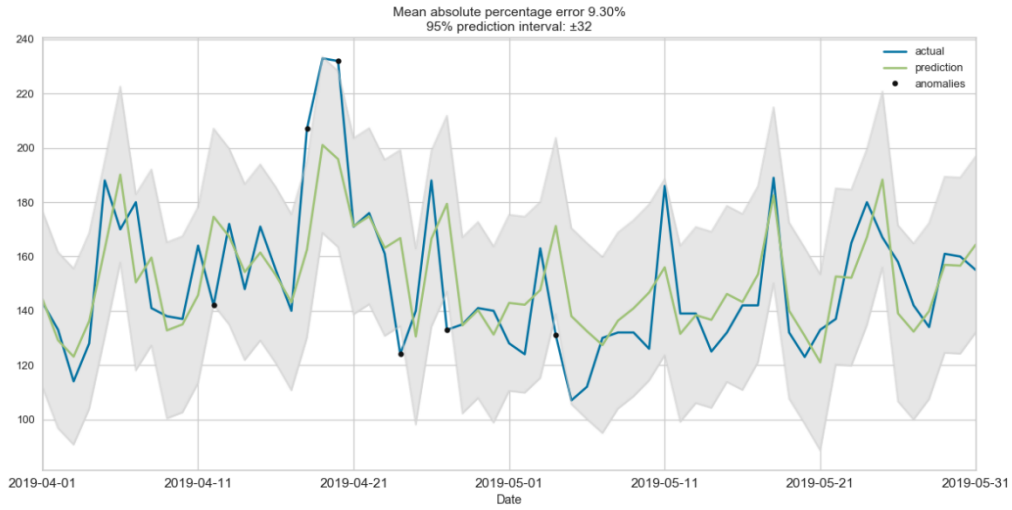


Figure 43 Prediction Interval with Constant Variance

Approach 2: Prediction interval with error model

Prediction interval with error model is carried out by fitting the input features to the residuals after testing the model. First, we do 70:30 split on the training set. We train on the 70% split and predict the rest 30% split. Then, we calculated the error of this prediction as the validation error.

This validation error has now become the new target for the error model. For this error model, we fit the input features and the validation error. Then, we predict the hold-out test set using this error model. The prediction interval can then be calculated by assuming normality and defining the standard error of the prediction by the square root of this prediction result. The remaining calculation is the same as the previous approach, that is:

$$Prediction\ interval = \hat{y} \pm Z\ score\ \times\ \hat{\sigma}$$

The intuition behind this approach is that the errors of a model are also affected by the input features. With this approach, instead of using a constant variance for all the predicted samples, each sample has its own deviation, meaning some instances will have large intervals while the others have smaller intervals. Figure 44 shows how the prediction interval looks on the prediction of replacement cars demand for April-May 2019. For a 95% prediction interval (Z-score of 1.96), the performance of this approach reaches a coverage PICP of 88.52% with 44.89 interval width. It means that we can be 95% certain that with adjustment of ±23 cars in average, the estimated value will fall within this interval 88.52% of the time.

This approach has resulted in a slightly lower accuracy in comparison to first approach at the expense of having a much lower deviation as well. Despite the trade-off, this level of performance may be more useful and acceptable for planning or adjusting the predicted demand as they do not have to adjust the value too far from the model prediction.

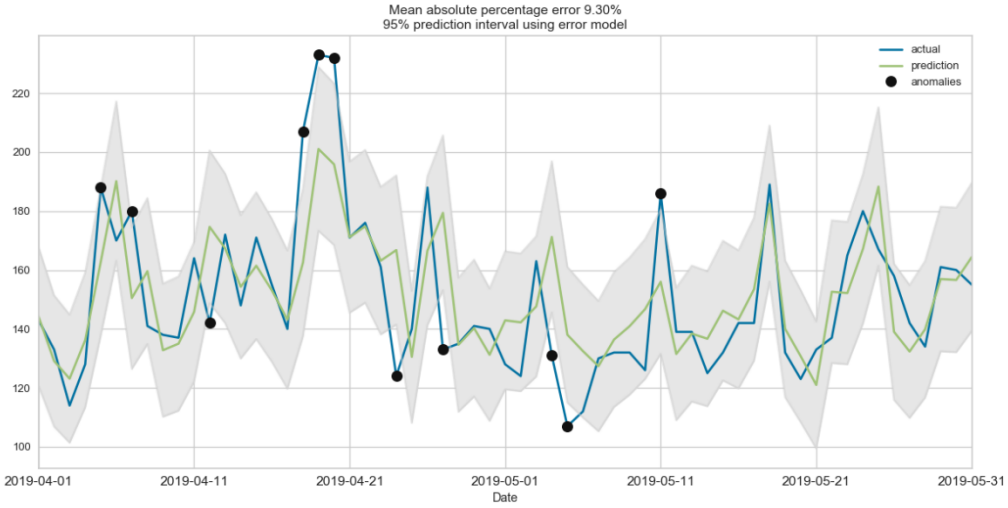


Figure 44 Prediction Interval with Error Model

Approach 3: Prediction interval with quantile regression

Based on our literature review in Section 2.6, we generated prediction interval using quantile regression that does not require assumption of distribution parameter. Using the implementation of quantile loss function on XGBoost, we can perform two XGBoost regressions on different levels (i.e. upper and lower quantile) as the upper and lower bound of the prediction. For a 95% prediction interval, we performed quantile regression at 0.025 and 0.975-quantile.

Figure 45 visualizes the prediction interval using quantile regression. It can be seen that quantile regression at extreme quantile such as 0.025, tends to produce values far off to the actual data. This quantile regression values succeed in covering the extremely low demand but provides very distant values to the other demand that are not extremely low. This results in the MPIW similar to the first approach, in fact average of 64.52 or ±33 cars per day. However, it produces 90% PICP, which is lower than the first approach. The difference in coverage probability between this approach and the error model approach is not that notable, in contrast to the difference in interval width.

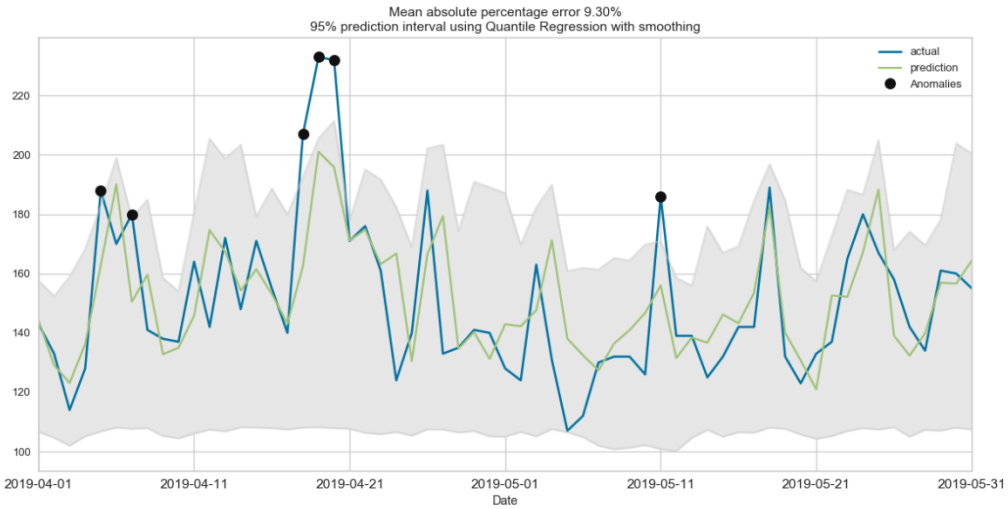


Figure 45 Prediction Interval with Quantile Regression

Table 24 shows the summary of the performance for the prediction intervals on the test set. Based on the performance of each technique, we can see that there is a clear tradeoff between coverage and interval width. While the MPIW for the first approach represents the deviation for all days in the forecast horizon, both prediction intervals with error model and quantile regression produce non-constant deviation for the demand, hence varying prediction interval width. Table 25 describes the statistics for the varying width.

Table 24 Performance of prediction interval methods

Approach	Coverage (PICP)	Width (MPIW)
1. Constant variance, assuming normality	95.823%	64.8941
2. Error model, assuming normality	88.522%	44.8949
3. Quantile regression	90.783%	64.1995

Table 25 Statistics of interval width for PI with Error Model and Quantile Regression

	Error Model	Quantile Regression
mean	44.894859	64.628367
std	3.840358	15.128772
min	34.9519	31.818756
25%	42.515541	53.749458
50%	44.736912	62.099396
75%	47.071878	73.993835
max	59.113995	106.597511

For error model, the standard deviation for the interval width in different days is low compared to the one for quantile regression. The interval width is ranging from approximately 35 to 59 per day, while in quantile regression it ranges from 32 to 107. As prediction interval is a means of presenting uncertainty of the prediction, this variation of width can indicate the reliability of the forecast. On certain days where the prediction interval is higher than normal, we can say that the prediction is not too reliable. This can act as a signal to the forecaster team to take a look and check whether they need to make adjustment to the numbers provided by the model. This is an important aspect to a demand forecast. Due to the nature of the case, one cannot expect to have an accurate forecast of the replacement car demand for all days in the whole year. Therefore, accuracy on average does not say much for the implementation in practice. According to the experience of domain expert, some forecast result on certain days like those on/around holidays are usually not as reliable as the others, thus the capability to distinguish certain days where they need to pay attention on and adjust the forecast will make the work more efficient.

Looking at the varying interval width for both error model and quantile regression approach, error model seems to provide a better upper and lower bound in general. Quantile regression tends to provide an average low value for an extreme low quantile. As a consequence, some of the high interval width in quantile regression may be induced by this nearly uniform low quantile estimation rather than the uncertainty of the forecast. Therefore, the error model approach may be preferable to use in practice. Despite that, since the variation itself is not too high for day to day basis, it may be difficult to differentiate or categorize which level of interval width can be regarded as unreliable forecast.

5.7 OUTLIERS ANALYSIS

Besides the prediction interval, we proposed an approach to handle the outliers to enhance the result of the demand predictions. Up to this point, we found that the demand predictions at the Netherlands level was able to perform decently in average, in fact with less than 10% MAPE and 12 cars per day. However, this performance does not hold for each and every day that was predicted. On some cases, the error of the prediction can reach up to 68 cars, which is around 50% of the actual average demand. Therefore, an analysis of such cases is essential to improve the prediction results and to support the implementation of the demand prediction model in practice.

Based on our exploratory data analysis, in the dataset, there are outliers that appear from time to time and are expected to be the result of extreme weather condition and/or holidays rather than because of an error in the data. Thus, they cannot easily removed without further analysis. Moreover, previous study has shown that in forecasting short-term demand, machine learning techniques were shown to be reliable provided the data contains no significant anomalies (Antunes et al., 2018). Therefore, in this study, we propose the use of similar approach to the three steps approach in Kharfan & Chan (2018) (i.e. clustering-prediction-classification). However, instead of clustering the data, followed by interpreting the characteristics of the data in each cluster, we propose to distinguish the data based on the characteristics, which are outliers and non-outliers, using outliers identification techniques.

5.7.1 Outlier Identification

There are several possible definitions of outliers, among them are the following (Aguinis et al., 2013):

1. *Single construct outliers* which are extremely large or small data values of the same construct/distribution.
2. *Error outliers* which are outliers lying at a distance from other data points because of errors in observation, recording, or data processing.
3. *Interesting outliers* which are accurate values far from other data points but may contain valuable knowledge.
4. *Discrepancy outliers* which are data points with large residual values that possibly affect model fit or parameter estimates.
5. *Influential time series additive outliers* which are data points that deviate from surrounding values in time series analysis.

Different type of outliers have different identification techniques and typically different approach to handle them. Previously, in Section 4.3.1, we have identified the *single construct outliers* from our dataset using the *box plot* approach. We applied the box plot to identify outliers (i.e. extreme demand values) in 2017-2018. After presenting these outliers to the domain experts, we concluded that these outliers are most likely not *error outliers* and therefore we treated them as *interesting outliers* by keeping them in the dataset for model building.

However, this method uses the assumption that the data has one identical distribution (Laurikkala et al. 2000). Meanwhile, we have observed that adding more historical data seems to violate this assumption as there is a clear difference in the range of demand in 2014 compared to the more recent period. As a result, with this approach, the lower outliers are gathered mostly in 2014 when in fact these values could be a normal demand in this year, as seen in Figure 46. This is in fact a known challenge in modelling outlier in time series or temporal data, due to the dynamic nature and evolutionary patterns of the data (Gupta et al., 2014).

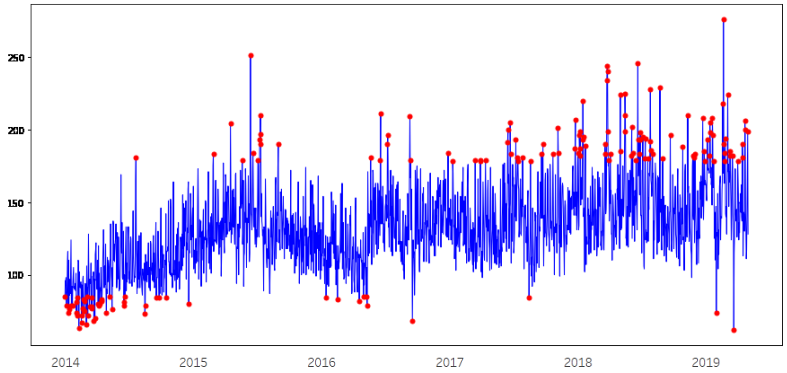


Figure 46 Outliers identifed using box plot

To identify the extreme values for all period of our data (2014-2019), we employed the *peak detection* approach which is a common task to identify sudden high values in time-series analysis and signal processing. These outliers can then be defined as those of the *influential time series additive outliers.* However, peak detection also has a drawback where it often results in a large number of false positives due to peaks occuring with different amplitudes (strong and weak peaks) and at different scales (Palshikar, 2009), which is also expected in our dataset given the different range of values over the years.

A number of libraries[4] with different algorithms that incorporate distance (time window) and height (threshold of normal values) in detecting peaks are available in Python. Using these algorithms, we identified the peaks (and valleys) in dataset of the demand of replacement cars. Considering our purpose of distinguishing outliers and non-outliers, which is to provide valuable insights into the cases with high prediction errors and propose an approach to handle such cases, we decided to focus on identifying the *discrepancy outliers* in the dataset. Therefore, we evaluated the detected peaks from each algorithm against the results of the best performing demand prediction model (i.e. whether the peaks match the cases with high deviations). We found that not all the detected peaks have high errors in the prediction results and not all cases with high errrors are detected as peaks. It might be that some of the cases with high prediction errors are anomalies because of the different underlying distributions or the lack of samples with similar patterns, instead of because of them being sudden extreme values.

Using prediction model to identify discepancy outliers

Thus far, we have investigated the extreme data values that lie far from the other data points and concluded that they are interesting outliers that may carry valuable knowledge instead of one that can be removed without analysis. Moreover, the peak detection algorithms show that the peaks in the data are not always the data points with large residual values (also known as discrepancy outliers). Therefore, we took a direct approach to identify discrepancy outliers by using the results of a prediction model. This approach has been mentioned in Gupta et al. (2014), where one can predict the value at time t using for instance a regression model, and define whether a data point is an outlier based on its deviation from the predicted values. Figure 47 ilustrates the identification approach.

[4] Tournade, Y., Overview of the peaks detection algorithms available in Python (2013), GitHub repository, https://github.com/MonsieurV/py-findpeaks

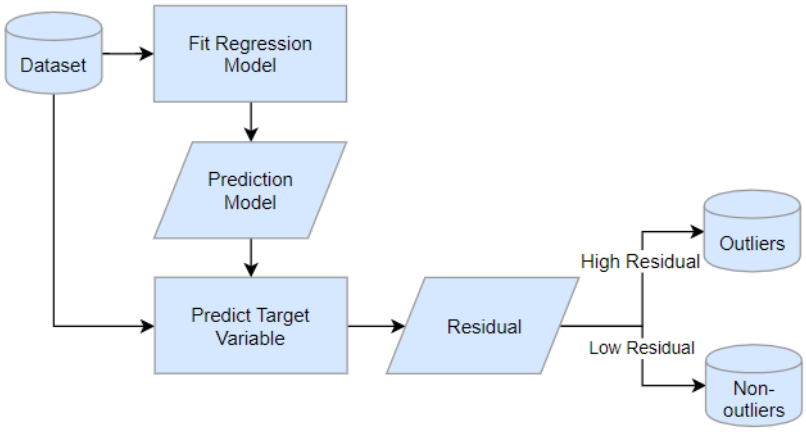


Figure 47 Outliers identification

Using the best performing demand prediction model for the Netherlands, we predicted the demand for the entire dataset of the demand of replacement cars in 2014-2019. Afterwards, we sorted the predictions based on the error/deviation from their actual demand values and labeled a certain top percentile of the errors as outliers. In this case, we applied three different threshold values to find the best distinction of the outliers and non-outliers, which are the values at the top 30th, 25th, and 20th percentile of the absolute errors. The first threshold gives us a 70:30 split of non-outliers and outliers where cases with errors higher than 13.27 cars labeled as outliers. The second one resulted in a 75:25 split where the absolute errors are higher than 14.72 cars, while the last one with 80:20 split separated outliers and non-outliers at 16.12 cars deviation. This choice of thresholds allow us to put together outlier datasets that contain at least 383 instances, which is a reasonable number to train and test a separate outlier model on. In the following sections, we discuss the results of the implementation of separate models for outliers and non-outliers, the comparison of the models trained on different composition of outliers and non-outliers, the classification of an instance into one of the datasets, and the recommendation for implementation in practice.

5.7.2 Prediction Models for Outliers and Non-outliers

After identifying the outliers, we divided the dataset into outliers and non-outliers datasets. We retrained the best performing XGBoost model for each dataset. The performance of the model is shown in Table 26.

Table 26 Performance comparison of separate and combined models for outliers and non-outliers

Model		MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
General model		12.06	9.49	241.80	15.55	10.14	0.6919
80-20 split (threshold = 16.12 cars)	**Non-outliers**	8.16	6.43	98.13	9.91	7.47	0.8414
	Outliers	27.45	21.50	1036.15	32.19	24.40	0.163
	Combined results	12.02	9.44	285.73	16.90	9.13	0.6176
75-25 split (threshold = 14.72 cars)	**Non-outliers**	7.59	5.89	89.68	9.47	6.67	0.8509
	Outliers	24.55	19.01	756.98	27.51	23.86	0.3878
	Combined results	11.83	9.17	256.50	16.02	8.60	0.6666
70-30 split (threshold = 13.27 cars)	**Non-outliers**	7.74	5.95	94.81	9.74	6.61	0.8612
	Outliers	24.47	18.47	756.15	27.50	22.99	0.1289
	Combined results	12.77	9.71	293.44	17.13	9.48	0.6065

The results show that in general, removing the outliers, which are instances with high prediction errors, have resulted in a significantly better performance across all the metrics used. However, the outliers dataset becomes much less predictable compared to the non-outliers. Particularly, the MSE that amplifies the impact of outliers or large errors is notably higher for the outlier models. Nevertheless, using these two separate models can still produce a better combined results compared to the results of a general model that does not analyze outliers and non outliers separately.

The results from different thresholds show that the 14.72 cars threshold (i.e. defining 25% highest errors as outliers) produced the best combined results of the separate models. The 80:20 split resulted in a quite high error metrics scores for the outliers dataset, thus affecting the performance of the combined results. This is reflected especially in the MSE of the combined results that becomes notably higher than the general model. The other metrics demonstrate a comparable performance to the general model. Meanwhile, the 75:25 split have shown an improved performance for the outliers dataset. It could be due to the increase in the number of samples used to train the data. The performances of the non-outlier models are also better than the 80:20 split. All of the predictions of the 75:25 non-outliers and outliers models combined, an improved performance compared to the general model is achieved, with an exception on the MSE that is marginally higher. However, further addition of samples to the outliers dataset by including all data points having errors higher than 13.72 cars produces a really small improvement for the outliers dataset but does not increase the performance of the non outliers and the combined results. To sum up, our results suggest that a 14.72 absolute prediction error is a good lower threshold to label a data point as an outlier for the demand of replacement cars in the Netherlands in a sense that it improves the average error of the test set.

5.7.3 Outliers Classification: a feasibility study

The last step to complete the process of outliers handling is to classify whether an observation in the future is a normal data point or an outlier case. To use separate models for outliers and non-outliers, one needs to be able to predict whether the demand in a certain day will be a normal or an outlier demand. It can be done by classifying a case into either an outlier (i.e. data that cannot be predicted well by the machine learning model) or non-outlier category, followed by predicting the demand using the corresponding model.

In the previous section, we have also seen that using separate models for outliers and non-outliers have improved the performance of the predictions but it does not show significant difference. Another alternative is to still use one sigle model for all data but supporting it with the same classification model to give an idea if the demand in a certain day is an outlier demand. If forecasters can get an insight about this occurence, we can expect a human intervention for this certain day. This section will carry out a first try into such classification.

We used the same dataset that is used for the general prediction model with a change in the target variable, from the demand to the label of the dataset (i.e. outliers and non-outliers). The same pipeline of processes was used to build and test the model. Gradient Boosting algorithm was applied for the classification task. As we want to emphasize on identifying the outliers, *recall* (i.e. the ratio of the number of instances that are correctly classified as outliers out of all the outliers in the dataset) was used as the evaluation metric to select the hyperparameters of the model through cross-validated grid search.

Table 27 shows the number of actual and predicted cases for both outlier and non-outlier classes. Out of 135 outliers in the test set, only 21 were correctly identified while the remaining 114 were classified as non-outliers. This has resulted in a *recall* of only 0.16 for the outlier class (see Appendix F.7 for a complete evaluation score for both non-outlier and outlier class). While the accuracy of the classifications reach 71.48%, the classification model still needs a significant improvement for it to be able to effectively classify the outliers. Further tuning of the model and the threshold for classifying an instance into either outlier or non outlier may improve the model. In addition, a comparison with other classification algorithms may also be useful to find the most suitable model for this task.

Table 27 Confusion matrix of outlier and non-outlier classification

		Predicted		
		Outlier	Non-outlier	
Actual	Outlier	21 (TP)	114 (FN)	135
	Non-outlier	50 (FP)	390 (TN)	440
		71	204	

Overall, the proposed approach has shown a good potential to improve the performance of the demand prediction. However, a good classification result would be required to effectively use this approach. This is because the significant improvement is brought mostly by the improvement in the non-outlier predictions. Meanwhile, the performance of the outlier model seems to be weak and may be the cause of a little increase in the MSE scores that are more sensitive to outliers. Thus, it is important to identify the outlier well from an unseen observation in the future. Furthermore, the ability to identify the outlier will provide more benefit in enhancing the prediction result and the overal planning activity as will be discussed in Section 6.3.

5.8 Chapter Summary

Up to this point, we have presented the results of the demand prediction model and the prediction interval. Our findings show that for the demand of replacement cars in the Netherlands, machine learning models perform better than classical time series when the models are trained by ignoring the time structure. Handling the categorical variables (i.e. weekday, month) by one-hot-encoding them resulted in the best performance in general. Furthermore, the use of Recursive Feature Elimination (RFE) for feature selection produced a better performance with less numbers of input features required. In general, there are common features that are included by all models for the prediction, but some other selected features differ for each model. On top of the prediction, we compared several approaches to generate prediction interval in addition to point prediction. We found that there is a trade-off between the coverage of the interval and the interval width. Besides the trade-off, we discovered that a high uncertainty of a model does not necessarily lead to a higher error, and vice versa. Finally, we proposed a framework to handle the outliers (i.e. non well-predicted cases) by identifying them, demonstrating the improvement in the model without outliers, and providing a first look into the classification of the outliers.

6 PUTTING THE MODEL INTO PRACTICE

In the previous chapter, we have discussed various models and approaches to improve and support the use of the model in practice. In this chapter, we will specifically address what models are recommended to be implemented and how they should be used in practice, and relate them to the goal of the business.

6.1 ROLE OF THE DEMAND PREDICTIONS IN ANWB & LOGICX

In our background study in Chapter 2, we have recognized the focus of our study as the tactical fleet planning problem where a demand prediction model is built as an input for fleet planning at the tactical level. In addition, our findings can also provide an insight for the other phases of planning problem, i.e. pool segmentation and strategic fleet planning. We will discuss how the findings from this study can assist each problem.

Tactical fleet planning

For the daily operation in the Netherlands, the demand prediction for the whole Netherlands is applicable as an input to determine the optimal number of cars for every location within the country and the repositioning of cars between the locations in the Netherlands. As we predict the daily demand of replacement cars in the Netherlands, we provide an insight of how busy a day will be for Logicx, thus indicating whether adjustment is required for the number of cars available in every rental location. This benefit is acknowledged by Logicx management.

In the current process, Logicx monitor the current stock against the minimum, maximum, and the number of cars expected to be returned at each location. Demand prediction contributes as an indicator of whether there is a need to adjust the minimum stock required at each location to handle the change of demand. For instance, when the predicted demand in the Netherlands is higher than a certain level, Logicx need to increase the stock at each location at a certain number. Currently, the focus of this strategy is for the weekend or holiday weekend as an influx of order often occurs during these periods. Therefore, the demand predictions in the Netherlands can contribute in Logicx planning process as follows:

1. *Predict daily demand for all days in the upcoming week at the end of each week.*
   In this way, the required stock on the weekend can be determined several days in advance. Given that Logicx have partial control over which location a car should be returned to (as long as the customers agree), Logicx can use this capacity to try to have the cars rented at the beginning of the week returned at the location that would require more cars around the weekend.
2. *Predict daily demand per day every day in a week*.
   In addition to the weekly prediction, an update of the demand prediction every day can provide a better insight for operational level as one can expect the demand prediction to be more accurate due to the more accurate weather forecast. As a result, Logicx have a more accurate idea of the demand the day after and can decide for further repositioning or outsourcing. This is particularly important since for the outsourcing process, there is around half a day lead time between ordering and arrival of cars at Logicx locations.

Besides as an indicator to adjust stock at each location accordingly. Demand prediction at lower spatial level such as rental location can be an even more useful input for planning per location. This is due to the fact that a high demand in the Netherlands may be centralized in some locations

instead of equally spread over the Netherlands, as we can see in Figure 41. However, the current results suggest that the daily demand prediction model per location is not in a good state to be used. As an alternative, Logicx can apply the high-level demand prediction and project it to the lower level locations. It is also important to note that it may be useful to define several different projections for different periods instead of using constant projections for the whole year due to, as an example, the demand that is possibly more focused near the seaside during summer holiday period.

Strategic fleet planning

Strategic fleet planning requires demand prediction with a more aggregated period than the daily prediction as it concerns with the total fleet size for the Netherlands in the long-term horizon, including the decisions about acquisitions and dismissal of cars for the whole fleet. Besides a long term demand prediction, a lot of other information may also be useful for this long-term planning, such as the data of the number of cars that are currently rented (away) for a certain day, the number of cars that are going to be returned, and the number of cars that are going to be rented. These number would tell the maximum number of cars needed to be available in a day, which is an important information for fleet size problem. Demand prediction per rental duration may be required for this purpose. However, it is out of the scope of this study. Demand per group of cars which were alternatively investigated as demand per market segment in our study, would also be a useful insight for this matter, as suggested in the car rental fleet management framework by Oliveira et al. (2017) (Appendix H).

Pool segmentation

Currently, Logicx manage the repositioning between all locations spread over the Netherlands. Managing the fleet in a couple of more disaggregated pools (i.e. divisions of area that group some rental locations together) is suggested in a lot of fleet management problems (Pachon et al., 2006). In this study, we investigated the feasibility and performances of various spatial aggregation levels in the Netherlands. These results can be useful as an insight in considering potential levels or grouping for pool segmentation in the future. Furthermore, demand per location along with other data such as pool design requirements and network of stations are suggested as an input for a pool segmentation (Oliveira et al., 2017).

6.2 Benefit Analysis

Among the studied levels for the demand prediction, the most aggregated model (i.e. demand prediction for the Netherlands in general) are the one with the best performance in terms of the error rate. The demand predictions per market segment may still be acceptable in practice, while the demand predictions for the deeper spatial levels would require significant improvement or a different approach to be able to be effectively used.

Table 28 Performance comparison for existing and proposed model

	MAE	MAPE	MSE	RMSE	MEDIAN_AE
Existing	23.15	15.01	887.40	29.79	20.00
Model	13.43	9.31	281.42	16.77	11.97

In the current process, Logicx are provided with a demand forecast based on a certain proportion of the expected number of vehicle breakdowns per day. Table 28 shows the comparison of the error metrics from the prediction results for 2017-2019. Compared to the existing demand predictions, the prediction model developed in this study is 6% more accurate as implied by the decrease in

MAPE. This percentage means that the model is able to reduce the average error (MAE) by 10 cars out of average demand of 146 cars per day. In particular, the machine learning model generally predict extreme high demand better than the existing predictions which is implied by the significant difference in the MSE scores. Table 29 shows the predictions and errors of the existing and the machine learning model for the highest actual demand from the test set. The evaluation results show that except for one instance (March 1st, 2018), in general the model cut the error of the existing forecast by half. The average percentage error of these highest 10 demand reach 33% (ranging from 23-57% error) while the model only has 15% percentage error (with a range from 6-22%).

Table 29 Demand predictions for the highest actual demand (2017-2019)

Date	Actual	Model	Existing	Error model	Error existing
2-6-2017	205	179.78	149	25.22	56
14-10-2017	201	159.04	111	41.96	90
23-12-2017	220	177.92	103	42.08	117
1-3-2018	244	227.41	247	16.59	3
20-4-2018	225	201.8	156	23.2	69
24-11-2018	208	196.2	134	11.8	74
21-12-2018	208	178.08	141	29.92	67
19-1-2019	218	183.93	138	34.07	80
18-4-2019	207	162.47	159	44.53	48
19-4-2019	233	201.07	165	31.93	68
20-4-2019	232	195.86	156	36.14	76

On certain days that are expected to have high demand such as during the weekend, holiday, or days with high temperature, Logicx usually replace the current estimation with an intuitive prediction that usually appeared to be more accurate than the estimation. However, it is possible that some unexpected days get ignored and in this case an error of 23-57% percent will occur for the related days. With the use of the model, Logicx can rely on the predictions that are data-driven rather and minimize the need for individual assumptions.

Given the potential improvement that the model can bring, the next interesting question would be how much this model can offer in terms of cost savings or increased benefit. There are a lot of possible scenarios and variable costs involved in the process of car repositioning between locations. Due to the complexity and the limited access to the data for the whole process, we approached this problem using the utilization rate of Logicx fleet. An improvement of demand prediction is expected to lead to a better availability of cars at the location, thus leading to a higher utilization of Logicx fleet in the Netherlands. However, Logicx goal is not to get a 100% utilization as in rental car business the company needs to have a certain stock at each location to guarantee that the replacement cars can always be provided for the customers. Furthermore, commonly in a car rental business, it is less likely that all the cars returned on a certain day can be re-rented on the same day due to the timing or down time like cleaning.

Therefore, to estimate the increase in utilization, we assumed a highest possible utilization to be close to 80% when a demand forecast can perform almost accurately. This is deemed as an optimistic case. Realistically, we assume 70% utilization in average will be achieved in ideal condition, which is the highest monthly utilization rate achieved by Logicx once in the last 2 years. In pessimistic scenario, it could be the case that even with a 100% accurate forecast, there will not be a significant difference in the utilization due to the ineffective fleet size, uncertainty in distribution, or the many considerations for repositioning of cars. By assuming these values for a perfect model with 0% prediction error, interpolation was used to estimate the utilization for the machine learning model with 9% error. In Logicx, it is known that 1% higher utilization will increase the margin by

approximately € 50,000 per year. Accordingly, with the use of the demand prediction model, Logicx could expect an increase in margin as set out in Table 30.

Table 30 Estimated increased margin

Current average utilization	62.37%
Increased margin per 1% increase of utilization	€ 50,000

	Optimistic	Realistic	Pessimistic
Utilization with 100% accurate model	80%	70%	65%
Utilization with proposed model	69.067%	65.269%	63.370%
Increased utilization	6.694%	2.896%	0.998%
Increased margin	**€ 334,695**	**€ 144,822**	**€ 49,885**

6.3 Recommendations for Implementation

We have previously analyzed the role and the potential of the studied demand prediction models in practice. In addition, we demonstrated that the high-level prediction model outperforms Logicx existing forecast, hence the potential improvement for customer satisfaction and financial gain. Furthermore, in this research, we have studied and proposed some approaches to enhance the demand prediction by taking into account the functionality and limitations of the developed models. We summarized the proposed usage of the models in the following implementation scheme (Figure 48).

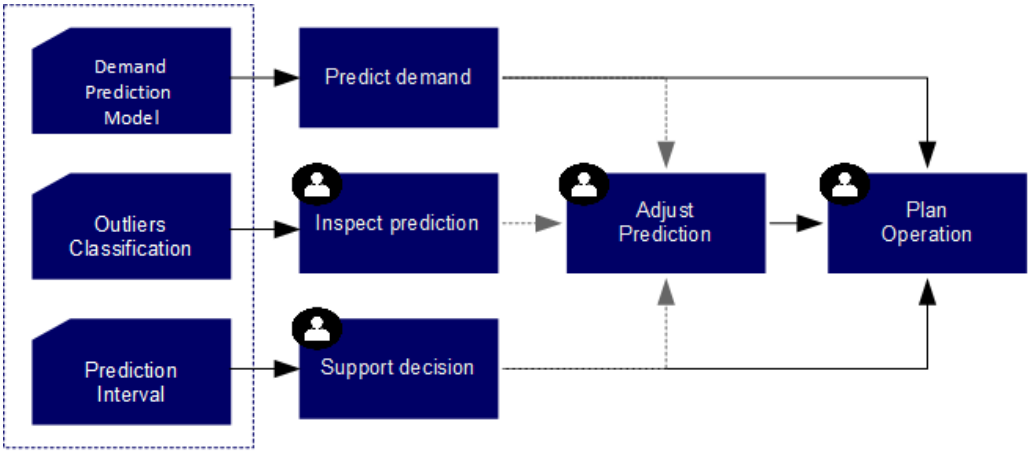


Figure 48 Implementation scheme

Demand prediction model is used to predict the demand of replacement cars. However, considering the current model cannot predict the demand satisfactorily for 100% of the cases, we proposed the use of *outlier classification* and *prediction interval* in combination with the knowledge of the domain experts. *Outlier classification* can be used to detect if a day is an outlier day (i.e. one that cannot be predicted well by the *demand prediction model*). If a day is classified as an outlier day, it is an indicator for the user of the model to inspect the prediction and decide whether the demand predictions need to be adjusted. With a good classification model, it will help the users to more effectively focus their attention on the difficult cases. Lastly, *prediction interval,* which points out the how uncertain the model is towards its prediction value, could be used to support the decision. Prediction interval shows the possible range of values (i.e. upper bound and lower bound) the demand may fall within according to the model. In practice, for instance, if the prediction value suggests that a large increase in stock would be required, the user can find out to what extent the prediction should be trusted before making a decision to adjust the stock.

7 DISCUSSION, LIMITATIONS, AND FUTURE WORK

This chapter consists of discussions of the findings as well as limitations and recommendations for future work. In Section 7.1, we interpret and discuss the new understanding and significance of our findings. After that, we reflect on several limitations of the study and propose potential future works accordingly.

7.1 DISCUSSION

The results of this study have shown that the classical time series model, SARIMAX, perform better than the machine learning models when the time structure is respected (i.e. the training set and testing set are split according to time order). This is in line with the findings in Whitt and Zhang (2019) where they found that SARIMAX outperform linear regression and MLP on a 1400 days dataset (comparable to our 1916 days dataset). However, this study has also demonstrated that machine learning models outperform the classical time series model when the training and testing set are split by also shuffling the data, disregarding the time order. One possible reason for this is that there is a change in the trend and variations of the demand over the years which make the year an important predictor of the demand. Therefore, training the model on older years and testing it on the more recent year(s) has brought down the effectiveness of the models. On that account, detrending the data may be an alternative approach to shuffling the data as the latter have raised concern over the potential violation of assumptions in the cross-validation procedure (Arlot & Celisse, 2010; Bergmeir & Benitez, 2012).

Ultimately, machine learning models, particularly XGBoost model, provided the best performance for the prediction of replacement car demand in the Netherlands. This may be an indication of the presence of nonlinear relationships that cannot be modeled by the linear models, including the classical time series models. By using machine learning model, company can have a better insight on the number of requests for replacement cars they can expect in a day, whether a day will be busy or if there will be less activities. Company can be more confident in planning their operation as the estimation of the predictions are based on real data from the past and expected values of the affecting factors in the future.

Despite the performance of the model, there are a number of shortcomings of the machine learning-based prediction for the demand of replacement cars. Performance of machine learning methods depends on the input data they are trained on. Business data oftentimes are very limited compared to other applications where machine learning is typically used, such as energy forecasting, which may affect the performance of the demand prediction as proper training may be difficult because of the short time series (Makridakis et al., 2018). We have seen this difficulty in our results since there are very limited occurrences of extreme demand values and specific holidays from 5 years historical data, that are expected to better estimate the actual demand. As a result, the model produced some serious errors in a small percentage of the cases.

For the above-mentioned reason, it may be preferable to have forecasters intervene with the machine learning predictions for a few particular cases where the model cannot perform well. Therefore, it is important for the company to be able to distinguish the cases where the model can perform well and the ones where we cannot expect a reliable and accurate estimation of the demand. Prediction intervals are one possible solution to this as it communicates the uncertainty level of the model. The wider the interval, the higher the uncertainty of the model towards the estimated demand for each day.

Unfortunately, we found that the applicability of this uncertainty interval in practice is not as straightforward as it seems. Our results show that high uncertainty does not necessarily mean high error between the prediction and the actual demand. Instead, it only shows how uncertain the model is of the estimated value it provides. Thus, intervention of users may not be necessary even though a really wide interval is provided in a certain day. Conversely, a day with really narrow interval may end up with a high error.

Ericsson (2001) addressed this issue in terms of numerical accuracy (i.e. whether a forecast has a low error) and statistical accuracy (i.e. whether the uncertainty interval can contain the actual value). He further defined two conditions, a *poor forecast* as one with numerical inaccuracy and *forecast failure* as one with statistical inaccuracy. Therefore, it may be more desirable to avoid forecast failure first to make sure that the uncertainty interval always covers the actual value. When a model is statistically accurate, then we can expect the uncertainty interval to enable the users to plan different strategies for the range of possible outcomes indicated by the interval, as suggested by Chatfield (1993). Our second approach in model building, which is to deal with the outliers to improve the prediction would be a step forward towards this direction. By separating the outliers and non-outliers, we hope to avoid forecast failure in the first place, and after that gain a further insight about the uncertainty by looking at the prediction interval.

Another interesting point of the results of this study is the possible alternatives for the use of the demand prediction model. In practice, it is often in the interest of a company to predict demand multiple days ahead. In this study, we have only tested the model for a one step ahead prediction. However, previous work with similar approach showed that increasing the steps from 1 to 7 days ahead increase the error rate gradually for each day added but does not make it substantially bad (Whitt and Zhang, 2019). Accordingly, the company should take into account this trade-off between the ability to do their planning several days in advance with less accuracy or daily planning with more accurate prediction.

7.2 Limitations & Future Work

7.2.1 Limitations

With regards to the data and the methods used, there are some limitations we can defined from this research.

1. Data limitations

The historical data for the demand of replacement car were taken from two different systems. These systems were in place for different time period and have different attributes in their record. We took the data from the database by filtering the data based on the location and the providers of the replacement cars. We used slightly different filters in the data due to the difference in the attributes available in each system. As a result, the data that we took may not have represented the exact same case. Furthermore, we use only 5 years of data because the relevancy of the older data to the current business. As a result, the features representing events that only happen once a year do not have a lot of instances which the model can learn the pattern on.

Besides the historical order data, the models used an actual historical weather data instead of a weather forecast due to the availability of weather forecast data for past observations. Meanwhile, the input features used for the prediction in the future will be based on a weather forecast.

Therefore, there may be an additional error resulted from the error of the weather forecast that is not evaluated in this study.

For benefit evaluation, we have also encountered the difficulty in collecting a more detailed data regarding the costs, partly also because of the complexity of the dynamic process related to the fleet management at operational level.

2. Methods limitations

In the model building phase, we proposed the use of Recursive Feature Elimination (RFE) as an automated feature selection technique. However, feature selection using RFE can only be done for models that has a coefficient or feature importance attribute. Therefore, the study did not report the selected subset of features for SVR with RBF kernel as we had to leave it out for not having the required attribute. Another method or extension of RFE for SVR RBF should have been implemented for a thorough comparison of all models.

In addition, using RFE when the input features contain dummy variables has introduced a tricky situation. On the one hand, there is an issue of interpretability of k-1 dummy variables (from one hot encoding) are included in the input. This is because of dropping another feature from the k-1 variables will lead to a loss of information of every categorical value in the original variables, thus the interpretability of the features is difficult to explain. On the other hand, using all k encoded dummy variables without removing any of the resulted binary variables will allow a better interpretation after RFE. Using all k dummy variables means that we are treating the features not as one feature, but independently. For instance, for weekday feature, the interpretation becomes “if a day is Sunday” instead of “what day of the week it is”. However, this choice will introduce a perfect multicollinearity in the data. Even so, this dilemma would not affect the result of our study as Neter et al. (1996) states that the existence of the multicollinearity would not affect the ability to get a good fit.

Besides RFE, the second approach for creating prediction interval, where we estimated the standard error of prediction with a prediction of error values (done by fitting the input features to the residuals) does not have a strong theoretical foundation supporting it as of now. This kind of limitation for prediction interval has also been mentioned in Makridakis et al. (2018) that points out the difficulty of defining a fully acceptable prediction interval method for a lot of machine learning algorithms due to the assumption of distribution that has to be used. In this study, we have only provided an empirical evidence for this approach as the scope of this study was rather to apply available techniques and find the most suitable technique for the case.

Another limitation is introduced by the weakness of MAPE as error measurement. At the later stage of the analysis, we found that demand in some Logicx locations can be categorized as intermittent demand, that is when 0 (zero) demand occurs frequently. As a consequence, the intuitive metric MAPE will result in an undefined/infinite value for such cases, which is a known disadvantage of MAPE (Hyndman, 2006). To compare the forecast results of the lower level time series to the main high-level forecast results, we used an adjusted value of MAPE for the zero-demand cases by assigning 100% error for every forecast error for the zero-actual demand. Nonetheless, this adjusted MAPE does not affect the conclusion of our study as we can still conclude that the daily demand of replacement cars on low level is difficult to predict accurately as shown by the highly unexplained variance proportion (R^2) among other things. However, even though the MAPE scores gave us an insight on approximately how far the percentage error differs in different spatial aggregation levels

and different area, MAPE is not the most suitable metric to use when comparing several time series where an intermittent series exists (Hyndman, 2006; Kim & Kim, 2016).

7.2.2 Future Work

Based on the limitations and the current state of the model, we defined several recommendations for future work.

1. Apply more sophisticated approach to deal with holiday features
   Due to the high cardinality of holiday features that is translated into a highly sparse dataset, the tree-based model, XGBoost, does not work quite well with the holiday data even though it outperforms the others. Therefore, a more advanced approach with hybrid model is worth to try. It can be by blending several models (i.e. taking average value of all models) where the models should have enough diversity, such as having different base classifiers, different feature set, different samples for the training set, and different parameters, or by stacking models. Stacking can be done by conducting "out-of-fold" prediction for all the one hot-encoded holiday features and use the result as an input for the tree-based model.

2. Investigate different time interval for lower level demand prediction
   As the results show that the lower level predictions do not perform well, it may be interesting to investigate if a longer period is used, such as weekly prediction per location.

3. Study the prediction of lower level demand as an intermittent demand prediction
   We have seen that the daily demand at lower spatial level resembles an intermittent demand pattern. Therefore, future research can focus on applying models that are more specialized for intermittent demand or create features that are more customized for predicting intermittent demand with machine learning. Furthermore, for a proper comparison metric over different series that includes intermittent-demand series, several other metrics were proposed in other studies instead of using MAPE, such as the Mean Absolute Scaled Error (MASE) (Hyndman, 2006) and the Mean Arctangent Absolute Percentage Error (MAAPE) (Kim & Kim, 2016). It may be interesting for further research to use these metrics to report the difference of forecast accuracies in different aggregation levels.

4. Investigate the feasibility of predicting the demand per rental duration
   Predicting demand for every potential rental length has been pointed out as one of the primary levels of aggregation in a demand forecast (Geraghty & Johnson, 1997). The effectiveness of this approach has not been proven and rental duration is known as a major contributor of demand forecast difficulties in industry (Yang et al., 2008). However, it has the potential to bring the demand prediction one step further to the optimization of the number of cars needed to be available. For example, the demand of replacement cars can be transformed into the demand per each rental duration per day. We can then predict the demand of cars that will be rented out for 1-2 days tomorrow, the demand of cars that will be rented out for more than 3 days the day after tomorrow, and so on.

5. Investigate correlation between the demand of replacement cars and the number of customers with replacement cars in their contract
   For the dataset, part of the reason that the data from earlier than 2014 were not used is because of the change in the roadside assistance package (i.e. from 2014, more customers

have replacement vehicles covered in their packages). Therefore, a further investigation on the correlation between the demand of replacement cars and the number of customers with replacement cars in their contract may be useful to improve the model. If they are correlated, it may be the case that the change in the distributions of the data are subject to the change in the number of customers. In this case, there is a possibility that the demand per number of users can be less fluctuating or more stationary than the demand data, thus making it easier to predict. In this case, we may be able to use a larger dataset with additional data from before 2014. It comes with the difficulty to also predict the unknown future values of the customers but since the number of customers are not dynamically changing every week, an estimate may work for the model.

8 CONCLUSIONS & CONTRIBUTIONS

This chapter summarizes the results of this study. Section 8.1 answers the sub questions formulated in Section 1.3 and concludes the answer to the main research question. Section 8.2 briefly describes the contributions of the research for theory and practice.

8.1 CONCLUSIONS

In this study, we developed a model to predict demand of replacement cars using machine learning techniques with Python, by following the CRISP-DM framework. The answers to the main research question and its sub-questions can be summarized as follows.

SQ1. What features can be used to predict the demand of replacement cars?

Based on literature study, interviews with domain experts, and data exploration, we listed a number of potential factors that can be a predictor of the replacement car demand (Table 4). Out of these factors, we selected a set of features that have the data available and/or predictable in the future. We created a dataset containing 90 features related to the historical data of replacement cars, time of the year, holiday, and weather (Appendix D). To reduce the dimensionality, we eliminated 6 features with high correlations and low variance. The remaining 84 features were then used to build prediction models using 8 machine learning algorithms. Out of the remaining 84 features, we found 15 features agreed upon by all machine learning models. They are mostly features related to historical data, time, and weather. The rest of the features that are used as predictors differ per model. The best performing model appears to use 42 features in total and eventually 36 features were selected after an automated feature selection using Recursive Feature Elimination.

SQ2. Which machine learning model is best suited to predict the demand of replacement cars?

We conducted a literature review to get an overview of machine learning techniques that are available and their characteristics (Table 2). There are two categories of techniques that are commonly used for time series forecasting, namely classical time series and machine learning forecasting. For daily demand prediction problem, supervised machine learning model is more efficient since it does not require retraining every day, unlike classical time series model. However, our literature study of the state-of-the-art machine learning forecasting (Table 3) has shown that classical time series can outperform machine learning model at some occasions. Therefore, we compared several machine learning models that work for regression problems, namely simple linear regression, lasso regression, ridge regression, Support Vector Regression with linear and RBF kernel, Random Forest, Gradient Boosting Regression, and XGBoost, as well as classical time series models as the benchmark for the demand prediction problem.

We trained machine learning models in two variations, which are training on a time-ordered and training on a shuffled training-test set split. Machine learning models trained on the shuffled train-test set split perform best, while the models trained on time-structured train-test set split cannot outperform classical time series model SARIMAX. The best performing model to predict the demand of replacement cars is XGBoost with 9.49% MAPE. For interpretability, linear regression model, including Lasso and Ridge, are more interpretable. However, their performances are lower than that of XGBoost. Nonetheless, interpretation for XGBoost can be provided in the form of feature ranking and importance.

SQ3. What aggregation level works best for the demand prediction?

Several possible aspects for aggregation level for a demand prediction model were found through literature study. Among them, the time interval, product type and spatial aggregation are the most relevant to our study. For the time interval, we used daily aggregate of the data as required for the planning activity that becomes the focus of our research. For the product type and spatial level, demand prediction models were built and compared for various aggregation levels. For product type, two demand prediction models for B2B and B2C market segments were developed. The evaluation shows that the performance of the demand prediction per market segment are lower than the high-level prediction (i.e. aggregated segment), with the mean average percentage error of 12.4% and 14.8% for B2C and B2B segment respectively, compared to 9.49% MAPE for the aggregated segment. For spatial aggregation, three deeper aggregation levels were considered, which are province, work area, and rental locations. Each of this level has a varying number of area and there are diverse ranges of demand values for every area which lead to varying performances per area. Table 31 shows the best performance out of all divisions within each aggregation level.

Table 31 Summary of the best performance of different aggregation level models

Aggregation level	Total division	Best MAPE	Best R^2
High level	1	9.49%	69.20%
Market segment	2	12.40%	61.97%
Province	12	17.54%	28.97%
Work area	33	29.52%	16.21%
Rental locations	83	47.03%	9.37%

We found that the higher the granularity, the less satisfactory the prediction performance is. For the low-level prediction with relatively low average demand, the models tend to predict around the average demand value. For some area, the R^2 scores even suggest that the models perform worse than predictions using the average value. This condition is found in all the spatial aggregation levels considered. Therefore, it is clear that for predictions with daily time interval, the aggregation level that works best is the highest level, which is all demand in the Netherlands for all market segment. The second best would be the predictions per market segment since it exhibits an acceptable performance while offering more detailed information for operational planning.

SQ4. How can we estimate uncertainty of the prediction result?

Prediction interval is widely suggested in literature as a means of presenting uncertainty. We compared three techniques to provide lower and upper bound of the daily demand prediction: using constant variance from the prediction model, variance of a fitted error model (both assuming observations are normally distributed) and quantile regression. We measured the performance of each approach based on the prediction interval coverage probability (PICP) and mean prediction interval width (MPIW). There is a clear trade-off between PICP and MPIW demonstrated in the performance of all prediction interval techniques. The constant variance approach results in the highest coverage compared to the others, in fact 95.82%, followed by quantile regression with 90.783%, and error model with 88.52%. However, the error model approach produced the most reasonable result in terms of the MPIW. Among the three approaches, the error model and quantile regression approaches offer more information compared to the constant variance. The variations in interval width from day to day produced in the two non-constant approach may indicate when prediction is less reliable as a wider interval implies a higher uncertainty. Furthermore, as we found that high uncertainty does not always mean a high error, we proposed an outlier separation and

classification approach that can be utilized as an indicator of the days on which the error of the prediction is expected to be high, hence suggesting the need of forecasters to intervene with the results.

Finally, we can conclude the answer to the main research questions as follows.

RQ: To what extent can we predict the demand of replacement cars in the Netherlands?

By using regression models and framing the time series problem as a supervised learning problem, we can generate a prediction for the daily demand of replacement cars in the Netherlands. Historical data and external data like weather and calendar data are proven to be useful for the demand prediction. The best performing model can predict the demand of replacement cars per day with 9.49% average percentage error in the country level. The distribution of the error for has shown that the model performed decently for almost 80% of the cases but it has high errors for the remaining cases. In fact, based on the proposed outlier analysis, the exclusion of the outliers has resulted in a 5.89% average percentage error.

The performance of the demand prediction gradually decreases each time we add more granularity to the prediction (i.e. predicting at lower aggregation level). For prediction per market segment at country level, the performance seems to be acceptable despite the decrease from the performance of the highest level. For province level, some provinces still demonstrate acceptable performances, but the less busy provinces suggest differently. The same goes for the even deeper work area and rental locations level. For this granularity, we found at some area resembles that of an intermittent demand, which the current input features and machine learning approach cannot predict effectively.

Moreover, in terms of the explained variance, the input features in the best performing model can explain up to 69% variance in the demand value. However, it still has a considerable percentage of unexplained variance, in fact 31%, which shows that the case has a significant random effect and/or there are other important predictors that cannot or has yet been incorporated in our model that can improve the performance of the model. According to the different approach that we implemented to generate the prediction interval, this performance of 9.49% error and 69% explained variance have an uncertainty where the half width of the interval in average is at least two times the average error of the prediction. To sum up, it is not possible to predict the demand of the replacement cars completely accurately. However, with the support of knowledge from the domain experts, supplementary components to the prediction like prediction interval and outlier analysis can be effectively used to enhance the demand prediction.

8.2 CONTRIBUTIONS

This research contributed to an enhancement of the current knowledge on the existing literature on rental cars demand prediction and demand prediction using machine learning techniques in general. This has been achieved by empirically testing a number of prediction models on a real-life dataset. This study provides valuable knowledge on what factors are important to predict the demand of replacement cars on a daily level. To the best of our knowledge, no literature has explored this topic for the specific domain application.

Furthermore, we have presented empirical comparison of several techniques for generating a prediction interval as an estimate of uncertainty for a demand prediction model and discussed the possibilities of utilizing it to help improve the effectiveness of demand predictions for planning. We empirically compared the performance of different approaches from different

category/characteristics (i.e. parametric, non-parametric, supervised learning on residuals), which can be generally applied to a wide variety of machine learning models. Previous research for demand forecasting using machine learning tend to focus on providing only the point forecast, while some others focus on proposing methods to generate prediction interval on top of a specific algorithm.

In this study, we have also directly compared and discussed the effect of deeper location level or more granular spatial aggregation. As far as we can tell, most research focused on investigating the different time aggregation for the prediction period or time window. Little has explicitly analyzed the effect of various location aggregation level in predicting time series demand.

For practice, this research has laid the groundwork for replacement car prediction model in the Netherlands and provided valuable information for the company to implement demand prediction and incorporate the result for their planning. All findings in this research are based on real life data. Therefore, the results are applicable for real world circumstances, increasing this research's contributions to practice.

REFERENCES

[1] Aburto, L., & Weber, R. (2007). Improved supply chain management based on hybrid demand forecasts. *Applied Soft Computing*, *7*(1), 136-144.

[2] Aguinis, H., Gottfredson, R. K., & Joo, H. (2013). Best-Practice Recommendations for Defining, Identifying, and Handling Outliers. Organizational Research Methods, 16(2), 270–301. doi:10.1177/1094428112470848

[3] Allstate Roadside Services. (2019). Data & Analytics, the future of roadside is here. Retrieved July 12, 2019, from https://www.allstateroadsideservices.com/roadside-solutions/data-analytics.aspx

[4] Ampazis, N. (2015). Forecasting Demand in Supply Chain Using Machine Learning Algorithms. *International Journal of Artificial Life Research (IJALR)*, *5*(1), 56-73.

[5] Antunes, A., Andrade-Campos, A., Sardinha-Lourenço, A., & Oliveira, M. S. (2018). Short-term water demand forecasting using machine learning techniques. *Journal of Hydroinformatics*, *20*(6), 1343-1366.

[6] ANWB. (2019, July 04). Pechhulp van de Wegenwacht: ANWB Wegenwacht®. Retrieved July 12, 2019, from https://www.anwb.nl/wegenwacht

[7] ANWB. (2019, January 28). The Royal Dutch Touring Club ANWB. Retrieved July 13, 2019, from https://www.anwb.nl/over-anwb/vereniging-en-bedrijf/organisatie/english-page

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

[9] Bergmeir, C., & Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. Information Sciences, 191, 192-213. Bishop, C. (1991). Improving the generalization properties of radial basis function neural networks. Neural computation, 3(4), 579-588.

[10] Bishop, C. M. (2006). Pattern recognition and machine learning. springer.

[11] Böse, J. H., Flunkert, V., Gasthaus, J., Januschowski, T., Lange, D., Salinas, D., ... & Wang, Y. (2017). Probabilistic demand forecasting at scale. Proceedings of the VLDB Endowment, 10(12), 1694-1705.

[12] Bontempi, G., Taieb, S. B., & Le Borgne, Y. A. (2012, July). Machine learning strategies for time series forecasting. In *European business intelligence summer school* (pp. 62-77). Springer, Berlin, Heidelberg.

[13] Botchkarev, A. (2018). Performance Metrics (Error Measures) in Machine Learning Regression, Forecasting and Prognostics: Properties and Typology. *arXiv preprint arXiv:1809.03006*.

[14] Brando, A., Rodríguez-Serrano, J. A., Ciprian, M., Maestre, R., & Vitrià, J. (2018, September). Uncertainty Modelling in Deep Networks: Forecasting Short and Noisy Series. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 325-340). Springer, Cham.

[15] Brochu, E., Cora, V. M., & De Freitas, N. (2010). A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*.

[16] Burrough, P. A., McDonnell, R., McDonnell, R. A., & Lloyd, C. D. (2015). Principles of geographical information systems. Oxford university press.

[17] Cankurt, S., & Subasi, A. (2015). Developing tourism demand forecasting models using machine learning techniques with trend, seasonal, and cyclic components. *Balkan Journal of Electrical and Computer Engineering*, *3*(1), 42-49.

[18] Carbonneau, R., Vahidov, R., & Laframboise, K. (2007). Machine learning-Based Demand forecasting in supply chains. *International Journal of Intelligent Information Technologies (IJIIT)*, *3*(4), 40-57.

[19] Carbonneau, R., Laframboise, K., & Vahidov, R. (2008). Application of machine learning techniques for supply chain demand forecasting. *European Journal of Operational Research*, *184*(3), 1140-1154.

[20] Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). CRISP-DM 1.0 Step-by-step data mining guide.

[21] Chatfield, C. (1993). Calculating interval forecasts. Journal of Business & Economic Statistics, 11(2), 121-135.

[22] Chen, K. Y., & Wang, C. H. (2007). Support vector regression with genetic algorithms in forecasting tourism demand. *Tourism Management*, *28*(1), 215-226. das Chagas Moura, M., Zio, E., Lins, I. D., & Droguett, E. (2011). Failure and reliability prediction by support vector machines regression of time series data. *Reliability Engineering & System Safety*, *96*(11), 1527-1534.

[23] DiCicco-Bloom, B., & Crabtree, B. F. (2006). The qualitative research interview. *Medical education*, *40*(4), 314-321.

[24] Domingos, P. M. (2012). A few useful things to know about machine learning. *Commun. acm*, *55*(10), 78-87.

[25] Dosser, M. (2016, December 13). ÖAMTC Mitgliedschaft. Retrieved July 12, 2019, from https://www.oeamtc.at/mitgliedschaft/

[26] Ericsson, N. R. (2001). Forecast uncertainty in economic modeling. FRB International Finance Discussion Paper, (697). Board of Governors of the Federal Reserve System (U.S.).

[27] Fink, A., & Reiners, T. (2006). Modeling and solving the short-term car rental logistics problem. *Transportation Research Part E: Logistics and Transportation Review*, *42*(4), 272-292.

[28] Geraghty, M. K., & Johnson, E. (1997). Revenue management saves national car rental. *Interfaces*, *27*(1), 107-127.

[29] Gopi, G., Dauwels, J., Asif, M. T., Ashwin, S., Mitrovic, N., Rasheed, U., & Jaillet, P. (2013, October). Bayesian support vector regression for traffic speed prediction with error bars. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)* (pp. 136-141). IEEE.

[30] Green Flag. (2019). Breakdown cover, assistance and recovery, whenever you need us: Green Flag. Retrieved July 12, 2019, from https://www.greenflag.com/breakdown-cover

[31] Gupta, M., Gao, J., Aggarwal, C. C., & Han, J. (2014). Outlier detection for temporal data: A survey. IEEE Transactions on Knowledge and Data Engineering, 26(9), 2250-2267.

[32] Gupta, R., & Pathak, C. (2014). A machine learning framework for predicting purchase by online customers based on dynamic pricing. *Procedia Computer Science*, *36*, 599-605.

[33] Hall, M. A. (2000). Correlation-based feature selection of discrete and numeric class machine learning.

[34] Hansen, B. E. (2006). Interval forecasts and parameter uncertainty. *Journal of Econometrics*, *135*(1-2), 377-398.

[35] Herrera, M., Torgo, L., Izquierdo, J., & Pérez-García, R. (2010). Predictive models for forecasting hourly urban water demand. *Journal of hydrology*, *387*(1-2), 141-150.

[36] Hong, W. C., Lai, Y. J., Pai, P. F., Lee, S. L., & Yang, S. L. (2007, September). Composite of support vector regression and evolutionary algorithms in car-rental revenue forecasting. In 2007 IEEE Congress on Evolutionary Computation (pp. 2872-2878). IEEE.

[37] Hyndman, R. J. (2006). Another look at forecast-accuracy metrics for intermittent demand. Foresight: The International Journal of Applied Forecasting, 4(4), 43-46.

[38] Hyndman, R.J., & Athanasopoulos, G. (2018) *Forecasting: principles and practice*, 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2. Accessed on 14 April 2019.

[39] Jackson, B. (2018, September 27). AI-powered CAA roadside assistance will be dispatched before you break down. Retrieved July 12, 2019, from https://www.itworldcanada.com/article/ai-powered-caa-roadside-assistance-will-be-dispatched-before-you-break-down/409330

[40] Kandasamy, K., Vysyaraju, K. R., Neiswanger, W., Paria, B., Collins, C. R., Schneider, J., ... & Xing, E. P. (2019). Tuning Hyperparameters without Grad Students: Scalable and Robust Bayesian Optimisation with Dragonfly. *arXiv preprint arXiv:1903.06694*.

[41] Kharfan, M., & Chan, V. W. (2018). Forecasting Seasonal Footwear Demand Using Machine Learning (Master's thesis, Massachusetts Institute of Technology, 2018). Cambridge: MIT. Retrieved from https://dspace.mit.edu/.

[42] Kim, S., & Kim, H. (2016). A new metric of absolute percentage error for intermittent demand forecasts. International Journal of Forecasting, 32(3), 669-679.

[43] Knuth, D. E. (1992). Literate Programming. Number 27 in CSLI Lecture Notes. *Center for the Study of Language and Information*, 349-358.

[44] Kotsiantis, S. B., Kanellopoulos, D., & Pintelas, P. E. (2006). Data preprocessing for supervised leaning. International Journal of Computer Science, 1(2), 111-117.

[45] Kvålseth, T. O. (1985). Cautionary note about R 2. The American Statistician, 39(4), 279-285.

[46] Laurikkala, J., Juhola, M., Kentala, E., Lavrac, N., Miksch, S., & Kavsek, B. (2000, August). Informal identification of outliers in medical data. In Fifth international workshop on intelligent data analysis in medicine and pharmacology (Vol. 1, pp. 20-24).

[47] Lei, S., Wang, H., Yang, C., Du, B., Zhong, R., & Huang, R. (2017, August). Forecasting car rental demand based temporal and spatial travel patterns. In *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)* (pp. 1-8). IEEE.

[48] Liu, C. B., Chamberlain, B. P., Little, D. A., & Cardoso, Â. (2017, September). Generalising random forest parameter optimisation to include stability and cost. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 102-113). Springer, Cham.

[49] Luo, G. (2016). A review of automatic selection methods for machine learning algorithms and hyper-parameter values. *Network Modeling Analysis in Health Informatics and Bioinformatics*, *5*(1), 18.

[50] Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). Statistical and Machine Learning forecasting methods: Concerns and ways forward. *PloS one*, *13*(3), e0194889.

[51] Mayr, A., Hothorn, T., & Fenske, N. (2012). Prediction intervals for future BMI values of individual children-a non-parametric approach by quantile boosting. *BMC Medical Research Methodology*, *12*(1), 6.

[52] McKay, M. D. (1995). Evaluating prediction uncertainty (No. NUREG/CR--6311). Nuclear Regulatory Commission.

[53] Mupparaju, K., Soni, A., Gujela, P., & Lanham, M.A. (2018). A Comparative Study of Machine Learning Frameworks for Demand Forecasting.

[54] Murphy, A. H. (1993). What is a good forecast? An essay on the nature of goodness in weather forecasting. Weather and forecasting, 8(2), 281-293.

[55] Neter, J., Kutner, M. H., Nachtsheim, C. J., & Wasserman, W. (1996). Applied linear statistical models (Vol. 4, p. 318). Chicago: Irwin.

[56] O'Donnell, A. R. (2014, November 19). Agero Introduces Enhanced Analytics to Support Roadside Assistance Services. Retrieved July 12, 2019, from http://iireporter.com/agero-introduces-enhanced-analytics-to-support-roadside-assistance-services/

[57] Oliveira, B. B., Carravilla, M. A., & Oliveira, J. F. (2017a). Fleet and revenue management in car rental companies: A literature review and an integrated conceptual framework. Omega, 71, 11–26.

[58] Oliveira, B. B., Carravilla, M. A., & Oliveira, J. F. (2017b, June). A Dynamic Programming Approach for Integrating Dynamic Pricing and Capacity Decisions in a Rental Context. In *Congress of APDIO, the Portuguese Operational Research Society* (pp. 297-311). Springer, Cham.

[59] Pachon, J., Iakovou, E., & Chi, I. (2006). Vehicle fleet planning in the car rental industry. *Journal of Revenue and Pricing Management*, *5*(3), 221-236.

[60] Palshikar, G. (2009, June). Simple algorithms for peak detection in time-series. In Proc. 1st Int. Conf. Advanced Data Analysis, Business Analytics and Intelligence (Vol. 122).

[61] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *Journal of machine learning research*, *12*(Oct), 2825-2830.

[62] Petropoulos, F., Makridakis, S., Assimakopoulos, V., & Nikolopoulos, K. (2014). 'Horses for Courses' in demand forecasting. *European Journal of Operational Research*, *237*(1), 152-163.

[63] Pinson, P. (2012). Very-short-term probabilistic forecasting of wind power with generalized logit–normal distributions. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *61*(4), 555-576.

[64] Prytz, R. (2014). Machine learning methods for vehicle predictive maintenance using off-board and on-board data (Doctoral dissertation, Halmstad University Press).

[65] Qiu, X., Zhang, L., Ren, Y., Suganthan, P. N., & Amaratunga, G. (2014, December). Ensemble deep learning for regression and time series forecasting. In *2014 IEEE symposium on computational intelligence in ensemble learning (CIEL)* (pp. 1-6). IEEE.

[66] RAC. (2019). Breakdown cover. Retrieved July 12, 2019, from https://www.rac.co.uk/breakdown-cover

[67] Roy, D., Pazour, J. A., & De Koster, R. (2014). A novel approach for designing rental vehicle repositioning strategies. IIE Transactions, 46(9), 948-967.

[68] Ru, B., McLeod, M., Granziol, D., & Osborne, M. A. (2017). Fast information-theoretic Bayesian optimisation. *arXiv preprint arXiv:1711.00673*.

[69] Shahrabi, J., Mousavi, S. S., & Heydar, M. (2009). Supply chain demand forecasting: A comparison of machine learning techniques and traditional methods. *Journal of Applied Sciences*, *9*(3), 521-527.

[70] Shearer, C. (2000). The CRISP-DM model: the new blueprint for data mining. *Journal of data warehousing*, *5*(4), 13-22.

[71] Snoek et al. Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems* (pp. 2951-2959).

[72] Somasundaram, R. S., & Nedunchezhian, R. (2011). Evaluation of three simple imputation methods for enhancing preprocessing of data with missing values. International Journal of Computer Applications, 21(10), 14-19.

[73] Taieb, S. B., Huser, R., Hyndman, R. J., & Genton, M. G. (2016). Forecasting uncertainty in electricity smart meter data by boosting additive quantile regression. *IEEE Transactions on Smart Grid*, *7*(5), 2448-2455.

[74] Taylor, S. J., & Letham, B. (2018). Forecasting at scale. *The American Statistician*, *72*(1), 37-45.

[75] Tugay, R., & Ögüdücü, S. G. (2017). Demand Prediction using Machine Learning Methods and Stacked Generalization. In *Proceedings of the 6th International Conference on Data Science, Technology and Applications - Volume 1: DATA* (pp. 216-222).

[76] van der Spoel, S., Amrit, C. A., & van Hillegersberg, J. (2016, March). The role of domain analysis in prediction instrument development. In *Big Data Interoperability for Enterprises (BDI4E) Workshop 2016*.

[77] van Hinsbergen, C. I., Van Lint, J. W. C., & Van Zuylen, H. J. (2009). Bayesian committee of neural networks to predict travel times with confidence intervals. *Transportation Research Part C: Emerging Technologies*, *17*(5),

[78] Verma, M., Gangadharan, G. R., Narendra, N. C., Vadlamani, R., Inamdar, V., Ramachandran, L., Calheiros, R. N. & Buyya, R. (2016). Dynamic resource demand prediction and allocation in multi-tenant service clouds. *Concurrency and Computation: Practice and Experience*, *28*(17), 4429-4442.

[79] Weinberger, K., Dasgupta, A., Attenberg, J., Langford, J., & Smola, A. (2009). Feature hashing for large scale multitask learning. arXiv preprint arXiv:0902.2206.

[80] Whitt, W., & Zhang, X. (2019). Forecasting arrivals and occupancy levels in an emergency department. *Operations Research for Health Care*.

[81] Wijaya, T. K., Sinn, M., & Chen, B. (2015, April). Forecasting uncertainty in electricity demand. In *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[82] Xu, J. X., & Lim, J. S. (2007, September). A new evolutionary neural network for forecasting net flow of a car sharing system. In *2007 IEEE Congress on Evolutionary Computation*(pp. 1670-1676). IEEE.

[83] Yang, Y., Jin, W., & Hao, X. (2008, October). Car rental logistics problem: a review of literature. In *2008 IEEE International Conference on Service Operations and Logistics, and Informatics* (Vol. 2, pp. 2815-2819). IEEE.

[84] Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, *50*, 159-175.

[85] Zhou, Q., Han, R., & Li, T. (2015, December). A two-step dynamic inventory forecasting model for large manufacturing. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)* (pp. 749-753). IEEE.

[86] Ziel, F. (2018). Quantile regression for the qualifying match of GEFCom2017 probabilistic load forecasting. *International Journal of Forecasting*.

APPENDIX

A. OUTLIERS INSPECTION

1. Low demand

Date	Roadside Assistance		Replacement Cars	Comment
	Forecast	Actual		
01/01/2017	2.453	2.703	81	New year's day
31/01/2017	3.539	3.019	79	2 weeks low numbers. Partial data in FLOW?
08/02/2017	3.482	3.146	81	2 weeks low numbers. Partial data in FLOW?
09/02/2017	3.699	3.427	83	2 weeks low numbers. Partial data in FLOW?
13/02/2017	3.778	4.090	83	2 weeks low numbers. Partial data in FLOW?
19/05/2018	3.261	2.987	75	Saturday in long Pentecost weekend
24/10/2018	2.944	2.783	99	Wednesday autumn break

2. High demand

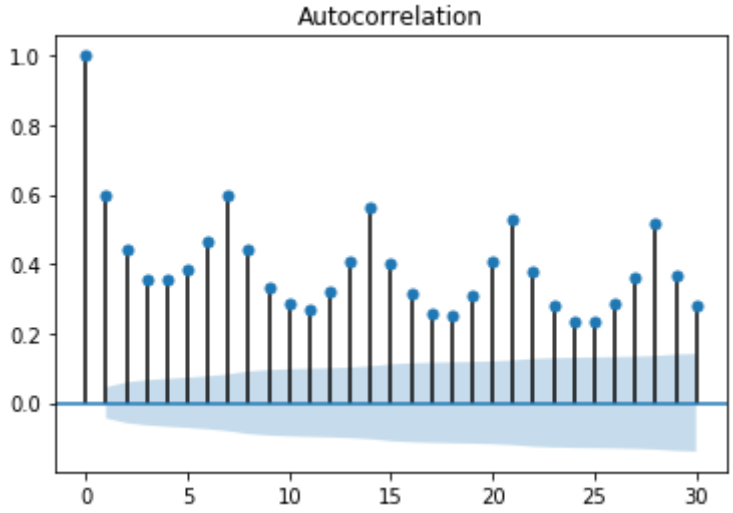
Date	Roadside Assistance		Replacement Cars	Comment
	Forecast	Actual		
02/06/2017	3.725	3.536	206	Friday before long Pentecost weekend
02/12/2017	3.603	4.373	205	Busy day also for road patrol
23/12/2017	3.414	3.430	220	Saturday first day of christmas holiday
28/02/2018	5.605	6.167	235	Extreme busy because of winter weather
01/03/2018	5.529	7.145	243	Extreme busy because of winter weather
02/03/2018	5.654	5.853	241	Extreme busy because of winter weather
26/05/2018	3.673	3.769	250	?
27/07/2018	4.119	3.778	230	Start of summer holiday part of the Netherlands

B. Percentage of Missing Values per Weather Station

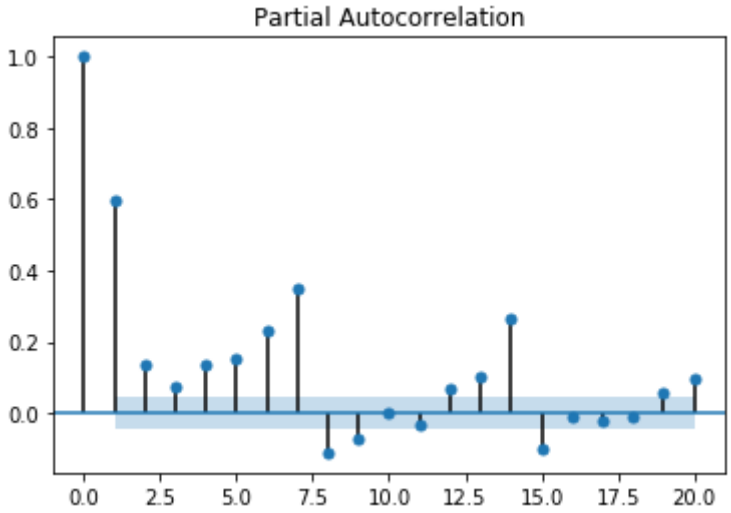
STN	Longitude	Latitude	Altitude	Name	Date	FG	TG	TN	TX	PG	RH	UG	Total
323	3.884	51.527	1.40	WILHELMINADORP	73.70	75.05	75.05	75.05	75.05	75.05	75.05	75.05	74.88
210	4.430	52.171	-0.20	VALKENBURG	55.48	55.48	55.48	55.48	55.48	55.48	55.48	55.48	55.48
311	3.672	51.379	0.00	HOOFDPLAAT	50.26	60.28	100.00	100.00	100.00	100.00	100.00	100.00	88.82
215	4.437	52.141	-1.10	VOORSCHOTEN	10.18	10.23	10.23	10.23	10.23	10.23	10.23	10.23	10.22
285	6.399	53.575	0.00	HUIBERTGAT	0.00	24.58	100.00	100.00	100.00	100.00	100.00	100.00	78.07
316	3.694	51.657	0.00	SCHAAR	0.00	9.66	100.00	100.00	100.00	100.00	100.00	100.00	76.21
209	4.518	52.465	0.00	IJMOND	0.00	5.17	100.00	100.00	100.00	100.00	100.00	100.00	75.65
313	3.242	51.505	0.00	VLAKTE V.D. RAAN	0.00	2.97	100.00	100.00	100.00	100.00	100.00	100.00	75.37
312	3.622	51.768	0.00	OOSTERSCHELDE	0.00	2.71	100.00	100.00	100.00	100.00	100.00	100.00	75.34
331	4.193	51.480	0.00	THOLEN	0.00	0.47	100.00	100.00	100.00	100.00	100.00	100.00	75.06
258	5.401	52.649	7.30	HOUTRIBDIJK	0.00	0.31	100.00	100.00	100.00	100.00	100.00	100.00	75.04
225	4.555	52.463	4.40	IJMUIDEN	0.00	0.00	100.00	100.00	100.00	100.00	100.00	100.00	75.00
248	5.174	52.634	0.80	WIJDENES	0.00	0.00	100.00	100.00	100.00	100.00	100.00	100.00	75.00
308	3.379	51.381	0.00	CADZAND	0.00	0.00	100.00	100.00	100.00	100.00	100.00	100.00	75.00
315	3.998	51.447	0.00	HANSWEERT	0.00	0.00	100.00	100.00	100.00	100.00	100.00	100.00	75.00
324	4.006	51.596	0.00	STAVENISSE	0.00	0.00	100.00	100.00	100.00	100.00	100.00	100.00	75.00
343	4.313	51.893	3.50	R'DAM-GEULHAVEN	0.00	0.00	100.00	100.00	100.00	100.00	100.00	100.00	75.00
257	4.603	52.506	8.50	WIJK AAN ZEE	0.00	100.00	0.00	0.00	0.00	100.00	0.00	0.00	25.00
242	4.921	53.241	10.80	VLIELAND	0.00	1.72	1.77	1.77	1.77	1.72	100.00	1.77	13.82
283	6.657	52.069	29.10	HUPSEL	0.00	0.00	0.05	0.00	0.00	100.00	0.00	0.00	12.51
249	4.979	52.644	-2.40	BERKHOUT	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	12.50
267	5.384	52.898	-1.30	STAVOREN	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	12.50
273	5.888	52.703	-3.30	MARKNESSE	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	12.50
277	6.200	53.413	2.90	LAUWERSOOG	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	12.50
278	6.259	52.435	3.60	HEINO	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	12.50
286	7.150	53.196	-0.20	NIEUW BEERTA	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	12.50
377	5.763	51.198	30.00	ELL	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	12.50
391	6.197	51.498	19.50	ARCEN	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	12.50
340	4.342	51.449	19.20	WOENSDRECHT	0.00	0.00	0.00	0.00	0.00	0.00	77.87	0.00	9.73
251	5.346	53.392	0.70	HOORN (TERSCHELLING)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.31	0.04
330	4.122	51.992	11.90	HOEK VAN HOLLAND	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.01
235	4.781	52.928	1.20	DE KOOY	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
240	4.790	52.318	-3.30	SCHIPHOL	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
260	5.180	52.100	1.90	DE BILT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
269	5.520	52.458	-3.70	LELYSTAD	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
270	5.752	53.224	1.20	LEEUWARDEN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
275	5.873	52.056	48.20	DEELEN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
279	6.574	52.750	15.80	HOOGEVEEN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
280	6.585	53.125	5.20	EELDE	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
290	6.891	52.274	34.80	TWENTHE	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
310	3.596	51.442	8.00	VLISSINGEN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
319	3.861	51.226	1.70	WESTDORPE	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
344	4.447	51.962	-4.30	ROTTERDAM	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
348	4.926	51.970	-0.70	CABAUW	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
350	4.936	51.566	14.90	GILZE-RIJEN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
356	5.146	51.859	0.70	HERWIJNEN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
370	5.377	51.451	22.60	EINDHOVEN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
375	5.707	51.659	22.00	VOLKEL	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
380	5.762	50.906	114.30	MAASTRICHT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

C. Autocorrelations of Demand Time Series

Autocorrelation Function (ACF) plot for the demand of replacement cars in the Netherlands



Partial Autocorrelation Function (PACF) plot for the demand of replacement cars in the Netherlands



D. INPUT FEATURES

	Features	Description	Type	Note
Output	Target	Demand of replacement car per day	Continuous	
Target lags	Target_t-1	Demand of replacement car 1 day before	Continuous	
	Target_t-2	Demand of replacement car 2 days before	Continuous	
	Target_t-3	Demand of replacement car 3 days before	Continuous	
	Target_t-4	Demand of replacement car 4 days before	Continuous	
	Target_t-5	Demand of replacement car 5 days before	Continuous	
	Target_t-6	Demand of replacement car 6 days before	Continuous	
	Target_t-7	Demand of replacement car 7 days before	Continuous	
	Rolling_mean	Average demand over the previous 7 days	Continuous	
	Rolling_std	Standard deviation of demand over the previous 7 days	Continuous	
Datetime attributes	Year	Year of the replacement car order	Continuous	
	Weekday_sin	Polar representation of Weekday feature	Continuous	
	Weekday_cos	Polar representation of Weekday feature	Continuous	
	Month_sin	Polar representation of Month feature	Continuous	
	Month_cos	Polar representation of Month feature	Continuous	
	Daylight	Time from sunrise to sunset	Continuous	
	Time_to_sunrise	Time from 00.00 to sunrise	Continuous	Highly correlated with Month_cos; removed
	Time_to_sunset	Time from 00.00 to sunset	Continuous	Highly correlated with Daylight; removed
	Weekday_0	Monday	Boolean	
	Weekday_1	Tuesday	Boolean	
	Weekday_2	Wednesday	Boolean	
	Weekday_3	Thursday	Boolean	
	Weekday_4	Friday	Boolean	
	Weekday_5	Saturday	Boolean	
	Weekday_6	Sunday	Boolean	
	Month_1	January	Boolean	
	Month_2	February	Boolean	
	Month_3	March	Boolean	
	Month_4	April	Boolean	
	Month_5	May	Boolean	
	Month_6	June	Boolean	
	Month_7	July	Boolean	
	Month_8	August	Boolean	
	Month_9	September	Boolean	
	Month_10	October	Boolean	
	Month_11	November	Boolean	
	Month_12	December	Boolean	

Public holiday	holiday_nieuwjaar	New Year's day	Boolean	
	holiday_1e paasdag	First Easter day	Boolean	
	holiday_2e paasdag	Second Easter day	Boolean	
	holiday_koningsdag	King's day	Boolean	
	holiday_bevrijdingsdag	Liberation day	Boolean	
	holiday_bevrijdingsdag_off	Liberation day as official (paid) holiday (every 5 years)	Boolean	Low variance/occurences; removed
	holiday_hemelvaartsdag	Ascension day	Boolean	
	holiday_1e pinksterdag	First Pentecost day (Whit Sunday)	Boolean	
	holiday_2e pinksterdag	Second Pentecost day (Whit Monday)	Boolean	
	holiday_1e kerstdag	First Christmas day	Boolean	
	holiday_2e kerstdag	Second Christmas day	Boolean	
School holiday	holiday_schoolvakantie_noord	School holiday in The Northern Region	Boolean	
	holiday_schoolvakantie_midden	School holiday in The Central Region	Boolean	
	holiday_schoolvakantie_zuid	School holiday in The Southern Region	Boolean	
Event and other non-official holiday	holiday_goede_vrijdag	Good Friday	Boolean	
	holiday_dodenherdenking	National remembrance day	Boolean	
	holiday_oudjaar	New Year's eve	Boolean	
	holiday_sinterklaas	Saint Nicholas' eve	Boolean	
	holiday_carnaval	Carnival	Boolean	
Day before holiday	holiday_goede_vrijdag-1		Boolean	
	holiday_1e paasdag-1		Boolean	
	holiday_koningsdag-1		Boolean	
	holiday_bevrijdingsdag_off-1		Boolean	Low variance/occurences; removed
	holiday_hemelvaartsdag-1		Boolean	
	holiday_1e pinksterdag-1		Boolean	
	holiday_1e kerstdag-1		Boolean	
	holiday_oudjaar-1		Boolean	
Day after holiday	holiday_nieuwjaar+1		Boolean	
	holiday_2e paasdag+1		Boolean	
	holiday_koningsdag+1		Boolean	
	holiday_bevrijdingsdag_off+1		Boolean	Low variance/occurences; removed
	holiday_hemelvaartsdag+1		Boolean	
	holiday_2e pinksterdag+1		Boolean	
	holiday_2e kerstdag+1		Boolean	
	holiday_1e werkmaandag		Boolean	
Holiday attributes	isHoliday	If the day is a public holiday, school	Boolean	
	First_day_long_holiday	The first day of ≥ 3 days consecutive holidays, including weekend	Boolean	
	Last_day_long_holiday	The last day of ≥ 3 days consecutive holidays, including weekend	Boolean	
	Long_weekend	One of the day in ≥ 3 days consecutive holidays, including weekend	Boolean	
	Day_before_long_holiday	The day before ≥ 3 days consecutive holidays, including weekend	Boolean	

Weather	FG	Daily mean windspeed (in 0.1 m/s)	Continuous	
	TG	Daily mean temperature (in 0.1°C)	Continuous	Highly correlated with TN and TX; removed
	TN	Minimum temperature (in 0.1°C)	Continuous	
	TX	Maximum temperature (in 0.1°C)	Continuous	
	RH	Daily precipitation amount (in 0.1 mm; -1 for < 0.05 mm)	Continuous	
	PG	Daily mean sea level pressure (in 0.1 hPa)	Continuous	
	UG	Daily mean relative atmospheric humidity (%)	Continuous	
Weather attributes	Consecutive_cold_days	Number of consecutive days where the	Continuous	
	Consecutive_warm_days	Number of consecutive days where the maximum temperature ≥ 25°C	Continuous	
	Cold_days_prev_week	Total number of days in the previous 7 days where the minimum temperature ≤ -2°C	Continuous	
	Warm_days_prev_week	Total number of days in the previous 7 days where the minimum temperature ≥ 25°C	Continuous	
	temp_diff_1_days_ago_TN	Difference between minimum temperature today and the day before	Continuous	
	temp_diff_1_days_ago_TX	Difference between maximum temperature today and the day before	Continuous	

E. SELECTED FEATURES PER MODEL

Features	Total Models	Linear Regression	Lasso	Ridge	SVR Linear	Random Forest	Gradient Boosting	XGBoost
Consecutive_cold_days	7	1	1	1	1	1	1	1
Consecutive_warm_days	7	1	1	1	1	1	1	1
Month_5	7	1	1	1	1	1	1	1
RH	7	1	1	1	1	1	1	1
Rolling_mean	7	1	1	1	1	1	1	1
Target_t-1	7	1	1	1	1	1	1	1
Target_t-2	7	1	1	1	1	1	1	1
temp_diff_1_days_ago_TX	7	1	1	1	1	1	1	1
TN	7	1	1	1	1	1	1	1
TX	7	1	1	1	1	1	1	1
Weekday_1	7	1	1	1	1	1	1	1
Weekday_4	7	1	1	1	1	1	1	1
Weekday_5	7	1	1	1	1	1	1	1
Weekday_6	7	1	1	1	1	1	1	1
Year	7	1	1	1	1	1	1	1
Daylight	6	1	1	1	0	1	1	1
Rolling_std	6	0	1	1	1	1	1	1
UG	6	0	1	1	1	1	1	1
Warm_days_prev_week	6	0	1	1	1	1	1	1
Weekday_2	6	1	0	1	1	1	1	1
Day_before_long_holiday	5	0	1	1	0	1	1	1
First_day_long_holiday	5	0	1	1	1	1	1	0
holiday_1e kerstdag-1	5	1	1	1	1	0	1	0
holiday_2e kerstdag+1	5	1	1	1	1	0	1	0
holiday_2e pinksterdag	5	1	1	1	1	0	1	0
holiday_2e pinksterdag+1	5	1	1	1	1	0	1	0
holiday_goede_vrijdag-1	5	1	1	1	1	0	1	0
holiday_hemelvaartsdag	5	1	1	1	1	0	1	0
holiday_hemelvaartsdag+1	5	1	1	1	1	0	1	0
holiday_hemelvaartsdag-1	5	1	1	1	1	0	1	0
holiday_nieuwjaar	5	1	1	1	1	0	1	0
Month_11	5	1	1	0	1	0	1	1
Month_6	5	1	1	1	0	0	1	1
Target_t-3	5	0	1	0	1	1	1	1
Target_t-4	5	0	0	1	1	1	1	1
Target_t-6	5	0	0	1	1	1	1	1
Target_t-7	5	0	0	1	1	1	1	1
Cold_days_prev_week	4	0	0	0	1	1	1	1
holiday_1e pinksterdag	4	0	1	1	1	0	1	0
holiday_koningsdag	4	1	1	1	1	0	0	0
holiday_schoolvakantie_zuid	4	0	1	0	0	1	1	1
isHoliday	4	0	0	0	1	1	1	1
Month_7	4	1	0	0	0	1	1	1

Features	Total Models	Linear Regression	Lasso	Ridge	SVR Linear	Random Forest	Gradient Boosting	XGBoost
PG	4	0	1	0	0	1	1	1
Target_t-5	4	0	0	1	1	1	1	0
Weekday_3	4	1	1	0	0	1	1	0
FG	3	0	0	0	0	1	1	1
holiday_1e paasdag-1	3	0	0	1	1	0	1	0
holiday_2e kerstdag	3	0	0	1	1	0	1	0
holiday_oudjaar	3	1	1	1	0	0	0	0
holiday_schoolvakantie_midden	3	0	0	0	0	1	1	1
Month_12	3	1	1	0	1	0	0	0
Month_3	3	1	0	0	1	1	0	0
Month_4	3	1	0	0	1	0	1	0
temp_diff_1_days_ago_TN	3	0	0	0	0	1	1	1
holiday_1e kerstdag	2	0	0	0	1	0	1	0
holiday_1e pinksterdag-1	2	0	0	1	1	0	0	0
holiday_1e werkmaandag	2	0	0	1	1	0	0	0
holiday_koningsdag+1	2	0	0	1	1	0	0	0
holiday_schoolvakantie_noord	2	0	0	0	0	1	1	0
holiday_sinterklaas	2	0	0	0	1	0	1	0
Last_day_long_holiday	2	0	1	0	1	0	0	0
Month_1	2	1	0	0	0	0	0	1
Month_10	2	1	0	0	1	0	0	0
Month_2	2	1	0	0	0	1	0	0
Weekday_0	2	1	0	0	0	0	1	0
holiday_1e paasdag	1	0	0	1	0	0	0	0
holiday_2e paasdag	1	0	0	1	0	0	0	0
holiday_2e paasdag+1	1	0	0	0	0	0	1	0
holiday_dodenherdenking	1	0	0	0	1	0	0	0
holiday_nieuwjaar+1	1	0	0	0	1	0	0	0
holiday_oudjaar-1	1	0	0	0	1	0	0	0
Month_8	1	1	0	0	0	0	0	0
Month_9	1	1	0	0	0	0	0	0
holiday_bevrijdingsdag	0	0	0	0	0	0	0	0
holiday_carnaval	0	0	0	0	0	0	0	0
holiday_goede_vrijdag	0	0	0	0	0	0	0	0
holiday_koningsdag-1	0	0	0	0	0	0	0	0
Long_weekend	0	0	0	0	0	0	0	0

F.1. Model Performance with Time Structured Dataset

		MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
Linear Regression	Training	11.0675	9.0152	205.566	14.3376	8.85352	0.672707
	Testing	14.1083	9.80982	353.395	18.7988	11.4258	0.504017
Lasso	Training	12.0703	9.88644	250.319	15.8215	9.84373	0.601454
	Testing	15.167	10.3459	411.08	20.2751	12.2842	0.423057
Ridge	Training	11.5101	9.40743	226.548	15.0515	9.23137	0.639301
	Testing	14.6449	10.1152	384.949	19.6201	11.913	0.459731
Random Forest	Training	8.50518	6.98514	116.036	10.772	7.03245	0.815253
	Testing	16.9345	10.8908	521.575	22.838	12.8449	0.267979
Gradient Boosting	Training	10.3913	8.55885	176.245	13.2757	8.53941	0.719391
	Testing	15.7201	10.2446	445.146	21.0985	12.7781	0.375246
XGBoost	Training	10.2114	8.41627	170.99	13.0763	8.46264	0.727758
	Testing	16.7681	10.7704	506.486	22.5053	12.8608	0.289156
SVR (Linear)	Training	11.603	9.41152	238.623	15.4474	8.9286	0.620076
	Testing	15.0387	10.3556	404.058	20.1012	12.0534	0.432912
SVR (RBF)	Training	10.7351	8.64665	213.092	14.5977	8.19995	0.660725
	Testing	14.8243	10.1424	391.446	19.785	12.0018	0.450613

F.2. Model Performance with Randomized Dataset

		MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
Linear Regression	Training	11.7942	9.10182	237.555	15.4128	9.53125	0.692933
	Testing	12.3535	9.68606	258.898	16.0903	9.8125	0.670209
Lasso	Training	11.9978	9.24619	245.178	15.6582	9.64758	0.683079
	Testing	12.3308	9.71742	257.207	16.0377	9.78918	0.672364
Ridge	Training	11.8042	9.1029	239.071	15.4619	9.59531	0.690972
	Testing	12.1955	9.5713	252.68	15.8959	9.76322	0.67813
Random Forest	Training	8.82478	6.87921	126.813	11.2611	7.40805	0.83608
	Testing	13.0045	10.2575	283.94	16.8505	10.2666	0.63831
Gradient Boosting	Training	9.45201	7.41139	144.53	12.0221	7.84885	0.813178
	Testing	12.1378	9.58108	244.092	15.6234	9.72451	0.68907
XGBoost	Training	11.067	8.62665	198.766	14.0984	9.14	0.743072
	Testing	12.1754	9.64069	246.139	15.6888	10.1648	0.686462
SVR (Linear)	Training	11.65	8.91016	248.74	15.7715	9.10414	0.678475
	Testing	12.352	9.65136	261.649	16.1756	9.83929	0.666705
SVR (RBF)	Training	10.4214	7.93812	221.432	14.8806	7.89844	0.713774
	Testing	12.706	9.88174	273.035	16.5238	10.3543	0.652201

F.3. Model Performance after RFE

		MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
Linear Regression	Training	12.1391	9.36855	251.078	15.8455	9.68293	0.675452
	Testing	12.2346	9.67313	250.608	15.8306	9.80661	0.68077
Lasso	Training	11.938	9.20645	242.81	15.5823	9.78793	0.68614
	Testing	12.308	9.68269	254.854	15.9642	9.97939	0.675361
Ridge	Training	11.8995	9.18505	242.511	15.5728	9.66369	0.686526
	Testing	12.305	9.67334	255.151	15.9735	9.8918	0.674982
Random Forest	Training	9.9609	7.74652	167.736	12.9513	8.2911	0.783181
	Testing	13.0943	10.3242	288.397	16.9823	10.1547	0.632632
Gradient Boosting	Training	9.3988	7.36741	143.095	11.9622	7.68901	0.815033
	Testing	12.2624	9.67205	251.737	15.8662	9.6451	0.679331
XGBoost	Training	9.65745	7.55937	156.009	12.4904	7.74156	0.79834
	Testing	12.0608	9.49054	241.797	15.5498	10.1423	0.691993
SVR (Linear)	Training	11.6848	8.93205	249.902	15.8083	9.17245	0.676972
	Testing	12.3439	9.64133	262.694	16.2078	9.87814	0.665373

F.4. Model Performance for B2C Segment

		MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
Linear Regression	Training	10.2070	12.7307	173.1700	13.1594	8.3354	0.5520
	Testing	10.4109	13.6021	183.2190	13.5359	8.8185	0.5401
Lasso	Training	8.8308	11.0611	130.8700	11.4399	7.0197	0.6614
	Testing	9.8215	12.7211	156.5600	12.5124	8.4838	0.6070
Ridge	Training	8.7534	10.9551	129.5460	11.3818	7.0146	0.6648
	Testing	9.8380	12.7097	155.4410	12.4676	8.3329	0.6099
Random Forest	Training	7.3177	9.2145	88.5450	9.4098	6.0997	0.7709
	Testing	9.9276	12.7629	164.5260	12.8268	8.3224	0.5870
Gradient Boosting	Training	6.8947	8.7399	75.1773	8.6705	5.8997	0.8055
	Testing	9.6592	12.4034	151.5030	12.3086	7.9630	0.6197
XGBoost	Training	7.9877	10.0994	105.5790	10.2752	6.5570	0.7268
	Testing	9.7015	12.5307	152.9030	12.3654	8.4889	0.6162
SVR (Linear)	Training	8.7279	10.8874	133.0760	11.5359	6.7745	0.6557
	Testing	9.9458	12.7792	159.2220	12.6183	8.4226	0.6004
SVR (RBF)	Training	6.9539	8.6858	97.0805	9.8530	5.1605	0.7488
	Testing	9.9572	12.7221	158.2500	12.5798	8.6607	0.6028

F.5. MODEL PERFORMANCE FOR B2B SEGMENT

		MAE	MAPE	MSE	RMSE	MEDIAN_AE	R^2
Linear Regression	**Training**	8.1508	18.1194	108.3800	10.4106	6.9703	0.4276
	Testing	8.0948	18.0444	105.2700	10.2601	6.7425	0.4243
Lasso	**Training**	6.8213	14.7743	80.1383	8.9520	5.4192	0.5767
	Testing	6.8949	15.1608	80.6627	8.9812	5.3371	0.5588
Ridge	**Training**	6.8125	14.7522	80.0281	8.9458	5.4173	0.5773
	Testing	6.9132	15.1636	80.7130	8.9840	5.4153	0.5586
Random Forest	**Training**	5.6101	12.0539	53.0894	7.2863	4.6577	0.7196
	Testing	7.1360	15.6676	85.6179	9.2530	5.7273	0.5317
Gradient Boosting	**Training**	5.5520	12.1045	49.3432	7.0245	4.5484	0.7394
	Testing	7.1470	15.6392	84.0733	9.1692	5.9011	0.5402
XGBoost	**Training**	5.6858	12.4329	53.1104	7.2877	4.6036	0.7195
	Testing	7.1644	15.6546	83.1704	9.1198	5.7477	0.5451
SVR (Linear)	**Training**	6.7090	14.4271	80.8781	8.9932	5.2800	0.5728
	Testing	6.9435	15.1708	81.6669	9.0370	5.7921	0.5534
SVR (RBF)	**Training**	5.4343	11.4787	64.0996	8.0062	3.8120	0.6614
	Testing	6.8411	14.7507	79.5612	8.9197	5.2753	0.5649

F.6. Model Performance for Logicx Rental Locations

Rental location		Selected Model	MAE	MAPE	MSE	RMSE	MED_AE	R^2	Average demand	Highest demand	Max capacity
604	Kuzee Vlissingen	SVR (RBF)	0.38	95.60%	0.24	0.49	0.26	-0.0139	0.26	3	150
613	Bergnet Blaricum	Lasso	1.07	49.57%	1.92	1.39	0.99	0.0143	2.16	8	25
615	Logicx Rotterdam	Ridge	1.68	49.35%	4.39	2.10	1.42	0.0396	5.03	17	45
620	Autohulpdienst Broekmans (Venlo)	XGBoost	1.02	55.97%	1.72	1.31	0.86	0.0937	1.66	8	20
622	Autohulpdienst Broekmans (Roermond)	Random Forest	0.89	54.58%	1.14	1.07	0.65	0.0051	1.40	6	40
623	Wolves Autoberging BV (Zwolle)	Random Forest	0.95	55.79%	1.28	1.13	0.69	-0.0005	1.41	7	9
628	Wolves Autoberging BV (Wierden)	SVR (linear)	0.71	67.54%	0.84	0.91	0.85	-0.0027	0.87	5	15
629	Delta Berging en Transport (Ermelo)	Ridge	1.13	55.94%	2.01	1.42	0.94	0.0284	1.78	10	15
630	Delta Berging en Transport (Emmeloord)	SVR (linear)	0.77	62.30%	1.01	1.00	0.88	-0.0124	1.07	8	8
631	Logicx Eindhoven	Random Forest	1.58	64.11%	3.81	1.95	1.45	0.0553	3.59	13	26
632	Houterman Wijchen BV	Ridge	1.20	62.73%	2.29	1.51	1.07	0.0523	1.93	9	10
633	Van Egeraat Berging en Transport CV (Rilland)	SVR (RBF)	0.72	67.75%	0.83	0.91	0.88	-0.0034	0.93	6	50
635	Bergnet Amersfoort	Ridge	1.53	57.42%	3.67	1.92	1.27	0.0123	3.50	13	25
639	Logicx Den Bosch	Random Forest	1.63	69.74%	3.94	1.98	1.51	0.0082	3.72	12	25
644	BCD (Dordrecht)	LinearReg	1.14	56.31%	2.17	1.47	0.97	0.0122	1.82	7	25
668	BCF Heerenveen	Random Forest	0.72	61.74%	0.88	0.94	0.83	0.0175	0.87	6	6
671	Roos Autoberging	Lasso	1.22	59.50%	2.32	1.52	1.14	0.0284	2.32	9	6
675	Takel- en Bergingsbedrijf Gerritse	SVR (RBF)	0.91	51.75%	1.47	1.21	0.80	0.0017	1.21	6	5
679	Vorgers Autoberging BV (Borne)	Ridge	0.77	69.06%	0.95	0.98	0.74	0.0217	0.80	5	12
680	Van Egeraat Berging en Transport CV (Oosterhout)	Lasso	1.10	58.06%	1.89	1.37	0.88	0.0238	1.63	8	15
682	Vermaat Hellevoetsluis	SVR (RBF)	1.13	55.50%	2.03	1.42	0.53	-0.0258	1.58	7	30
802	Haulo Alkmaar	Ridge	1.05	55.62%	1.65	1.28	0.74	0.0155	1.60	7	8
808	Auto- en bergingsbedrijf Besems BV	Ridge	1.28	58.25%	2.51	1.58	1.32	0.0216	2.53	9	10
811	Barendregt Rhoon	Ridge	1.07	53.62%	1.95	1.40	0.88	0.0045	2.26	8	10
814	Smits Diemen	Ridge	1.18	61.62%	2.17	1.47	1.10	0.0234	2.18	7	15
818	Autoberging Dallinga	SVR (RBF)	0.33	95.87%	0.16	0.40	0.24	-0.0203	0.23	2	10
819	Kooijman Takel & Berging (Spijk)	Ridge	1.05	58.03%	1.72	1.31	0.65	0.0058	1.48	7	20
820	VaRaMo Breskens	Ridge	0.35	94.54%	0.19	0.43	0.26	0.0080	0.24	2	4

Rental location		Selected Model	MAE	MAPE	MSE	RMSE	MED_AE	R^2	Average demand	Highest demand	Max capacity
822	Haulo Schagen	Ridge	0.46	93.27%	0.36	0.60	0.31	0.0066	0.30	3	10
825	Logicx Assen	Lasso	0.80	52.02%	1.11	1.05	0.91	0.0498	1.04	6	25
826	Schoenmaker & Zonen BV	SVR (RBF)	0.76	74.47%	0.94	0.97	0.87	-0.0149	0.85	5	10
831	Autohulpdienst Ben Heiltjes	Random Forest	0.88	57.02%	1.37	1.17	0.94	0.0009	1.04	6	25
832	Haulo Den Oever	Ridge	0.58	82.99%	0.45	0.67	0.49	0.0124	0.47	4	30
836	Koolen Garage en Bergingsbedrijf	Ridge	0.69	72.60%	0.70	0.84	0.63	0.0195	0.68	6	58
846	Delta Berging en Transport (Lelystad)	Lasso	0.97	55.98%	1.33	1.15	0.67	0.0080	1.46	6	15
848	Kraan- en Bergingsbedrijf Stouwdam (Oldebroek)	Ridge	0.92	56.83%	1.31	1.14	0.85	0.0206	1.26	8	10
849	Autax VOF	Lasso	0.50	90.10%	0.37	0.61	0.42	0.0195	0.36	3	4
851	BCD (Papendrecht)	Ridge	0.84	55.95%	1.13	1.06	0.92	0.0258	1.13	5	10
853	ANWB Servicecentrum Rhoon	Ridge	1.56	58.94%	3.58	1.89	1.34	0.0118	4.00	12	4
857	BCF Sneek	Lasso	0.56	88.50%	0.48	0.70	0.44	-0.0178	0.40	3	25
858	Bergingsbedrijf Willem Keizer	Ridge	0.46	90.48%	0.29	0.54	0.37	0.0042	0.32	4	5
861	ANWB Servicecentrum	Random Forest	2.08	47.03%	7.03	2.65	1.73	0.0295	5.93	16	8
870	Logicx Valkenburg	Ridge	1.31	62.98%	2.99	1.73	1.09	0.0265	2.34	11	40
873	Bergnet Amsterdam	SVR (RBF)	1.78	55.14%	5.00	2.24	1.45	0.0344	4.74	13	35
879	Haulo Den Helder	Ridge	0.59	79.87%	0.43	0.66	0.51	0.0270	0.52	4	2
891	ABS Autoherstel ASG Autoschade Gelderland	Random Forest	0.50	91.38%	0.37	0.61	0.37	-0.0024	0.36	3	6
900	Iliohan Garagebedrijf BV	Ridge	0.60	85.38%	0.47	0.69	0.50	-0.0030	0.48	4	10
914	Takel- en Bergingsbedrijf Grootveld VOF Brunssum	Ridge	1.06	55.28%	1.80	1.34	0.87	0.0294	1.62	8	10
917	Logicx Amsterdam	Ridge	1.74	51.40%	4.89	2.21	1.48	0.0013	5.20	13	25
919	24-Seven Berging B.V. (Staphorst)	SVR (linear)	0.78	63.46%	1.09	1.04	0.90	-0.0006	1.02	6	15
920	Smits Haarlem	Random Forest	1.50	62.02%	3.57	1.89	1.13	0.0199	3.07	10	30
923	Theo Rood B.V. (Zwaag)	SVR (RBF)	0.78	54.91%	1.03	1.02	0.99	0.0040	0.99	5	15
926	BRL B.V. (Leiden)	SVR (RBF)	1.56	61.38%	3.78	1.94	1.38	-0.0008	3.66	14	100
927	Van Egeraat Berging en Transport CV (Roosendaal)	Ridge	0.97	56.65%	1.53	1.24	0.74	0.0211	1.33	6	30
928	Hoogwout Berging BV (Oostzaan)	Random Forest	1.64	54.50%	4.06	2.01	1.39	0.0728	4.41	14	20
930	Wolves Autoberging BV (Apeldoorn)	Ridge	1.42	57.58%	3.09	1.76	1.22	0.0552	3.07	12	15

Rental location		Selected Model	MAE	MAPE	MSE	RMSE	MED_AE	R^2	Average demand	Highest demand	Max capacity
931	Logicx Nieuwegein	Lasso	1.48	61.48%	3.49	1.87	1.15	0.0101	3.01	12	40
937	ANWB Servicecentrum Naarden	Ridge	1.16	56.77%	2.00	1.42	0.90	0.0044	2.44	11	9
938	ANWB Servicecentrum Breda	Lasso	1.27	55.93%	2.46	1.57	1.13	0.0288	2.59	9	16
940	ANWB Servicecentrum Ypenburg	Ridge	1.65	69.44%	3.99	2.00	1.51	0.0194	3.58	10	5
941	ANWB Servicecentrum Wolfheze	SVR (linear)	1.09	59.52%	1.86	1.36	0.88	0.0327	1.73	7	10
942	Poort Takel- en Berging BV	SVR (linear)	0.74	64.96%	0.93	0.96	0.85	0.0054	0.88	5	8
943	ANWB Servicecentrum Groningen	Random Forest	0.93	56.18%	1.20	1.10	0.58	-0.0113	1.43	6	5
944	ANWB Servicecentrum Geldrop	Lasso	1.00	55.57%	1.52	1.23	0.59	-0.0018	1.58	9	6
945	Autosleepbedrijf Sprankenis v.o.f. Leende	LinearReg	0.80	66.11%	1.09	1.05	0.83	0.0412	0.94	7	2
946	Delta Berging en Transport (Almere)	Random Forest	1.14	55.35%	2.09	1.45	1.02	0.0189	2.18	8	25
947	BRL B.V. (Rijswijk)	Lasso	1.93	54.03%	5.55	2.36	1.64	0.0760	5.01	13	15
948	De Jonge Auto en Bergingsbedrijf vof (Hoogeveen)	Lasso	0.75	53.30%	0.94	0.97	0.98	0.0096	1.02	6	10
950	Autoberging Rutjes	Lasso	0.76	55.25%	0.99	1.00	0.92	0.0017	1.00	5	20
952	BCF Leeuwarden BV	LinearReg	0.72	62.88%	0.84	0.92	0.77	0.0189	0.89	4	15
953	van der Vliet BV	SVR (linear)	1.09	58.48%	2.25	1.50	0.98	0.0057	2.29	9	100
954	Kooijman Takel & Berging (Vianen)	LinearReg	1.24	64.57%	2.66	1.63	1.05	0.0010	2.08	9	10
955	Kuzee Goes	Random Forest	0.75	64.91%	0.86	0.93	0.69	0.0581	0.83	9	10
956	Kuzee Terneuzen	Ridge	0.50	89.58%	0.34	0.58	0.41	0.0016	0.37	3	10
957	Roy van Rijswijk VAS	SVR (linear)	1.29	98.16%	2.80	1.67	1.09	0.0284	2.50	10	6
958	Houterman Arnhem BV	SVR (linear)	1.15	60.00%	2.26	1.50	1.00	-0.0046	1.98	9	12
959	Garage Speerstra Workum BV	Ridge	0.57	84.93%	0.41	0.64	0.51	-0.0466	0.48	4	20
961	Autoberging Twente Weerselo	SVR (linear)	0.68	63.70%	0.79	0.89	0.79	0.0163	0.81	5	8
962	Houterman Nijmegen BV	SVR (RBF)	1.02	59.37%	1.47	1.21	0.58	-0.0048	1.38	6	20
963	Houterman Veenendaal	Lasso	1.31	61.49%	2.67	1.63	1.06	0.0119	2.04	8	15
964	BCF Beetsterzwaag	SVR (linear)	0.72	60.14%	0.88	0.94	0.90	0.0037	0.97	5	6
965	Vermaat Oude Tonge	Lasso	0.39	94.20%	0.28	0.53	0.25	0.0080	0.26	3	15
966	Fruitema	SVR (RBF)	0.38	95.60%	0.24	0.49	0.26	-0.0139	0.26	3	10

F.7. Performance of Outlier Classification Model

	Precision	Recall	F1-score	Support
Non-outlier	0.77	0.89	0.83	440
Outlier	0.3	0.16	0.2	135
Accuracy			0.71	575
Macro avg	0.53	0.52	0.52	575
Weighted avg	0.66	0.71	0.68	575

G.1. Classes of Cars

Voertuigen per klasse

A	Hyundai I10, Volkwagen Up
B	Fiat 500, Ford Fiesta, Opel Corsa, Suzuki Swift, Suzuki Ignis, Audi A1, VW Polo, Skoda Fabia combi, Citroën C3, Renault Clio Estate, Renault Captur
S	Fiat 500 X, Peugeot 2008
C	Mini Clubman, Fiat Tipo, Ford Focus wagon, Opel Astra tourer, Opel Astra 5 drs, Toyota Auris, Suzuki S-Cross, Seat Leon, Renault Megane estate, VW Golf
D	BMW 118, BMW 218, Mini Countryman, Jeep Renegate, Alfa Romeo Guilietta, Mercedes A, Audi A3, Skoda Octavia combi
X	Opel Zafira, Opel Mokka, Nissan Quashqai, Renault Scenic, VW Touran
E	Volkswagen Passat, Volvo V40
F	BMW X1, BMW 320i Touring, Mercedes GLA, Mercedes CLA, Volvo V60, Audi A4
M	Renault Grand Scenic
I	Volvo XC60, Volkswagen Tiguan
G	BMW 520D, Volvo S90, Volvo V90
P	VW Transporter 9 persoons
Q	Volvo XC90
VC1	VW Transporter enkele cabine
VC2	VW Transporter dubbele cabine
VC3	VW Crafter enkele cabine
VC4	VW Crafter dubbele cabine
N1	BMW F 800 GT, BMW 800 R
N2	BMW R 1200 RT, BMW R 1200 GS
Y1, Y2	Aanhanger zonder huif, aanhanger met huif

G.2. Segments of Classes of Cars

Klein (Small)	A, B, C and S
Midden (Medium)	D, E and X
Hoog (High)	F, M, I, G and Q
Berdijfswagens (Company vans/buses)	VC1, VC2, VC3 and VC4
Specials	Y1, Y2, N1, N2 and P

H. CAR RENTAL FLEET MANAGEMENT FRAMEWORK

Framework for car rental fleet management problem (Oliveira et al., 2017)