LUT University
School of Engineering Science
Industrial Engineering and Management

Warehouse stock optimization for Ramirent Finland Oy

Aleksi Toivanen

1. Supervisor: Jan Stoklasa
2. Supervisor: Pasi Luukka

TIIVISTELMÄ

Tekijä: Aleksi Toivanen

Työn nimi: Warehouse stock optimization for Ramirent Finland Oy

Vuosi: 2020 **Paikka:** Lappeenranta

Diplomityö. LUT-yliopisto, Tuotantotalous, Business Analytics

120 sivua, 15 taulukkoa, 48 kuvaa

Tarkastaja(t): Pasi Luukka, Jan Stoklasa

Hakusanat: Rakennuskoneet, vuokraus, kvantitatiivinen kysyntäennustaminen, jaksottaisen kysynnän ennustaminen, optimointi, data-analytiikka päätöksenteossa

Tämän diplomityön aiheena on kehittää työkalumalli päätöksenteon tueksi Ramirent Finland Oy:n vuokrakoneiden siirtoon palvelupisteiden välillä. Diplomityö pohjautuu kirjallisuuteen ja aikaisempiin tutkimuksiin aihealueesta, jotka käsittelevät kvantitatiivisia ennustusmenetelmiä ja eri optimointimalleja.

Työn aikana tarkasteltiin yrityksen viime vuosien vuokratransaktioiden dataa ja tämän pohjalta suoritettiin analyysi työkaluun sopivien kysynnänennustamis-, sekä palautusennustamismetodien valitsemiseen aikaisempaa kirjallisuuskatsausta hyväksikäyttäen. Kaikkiaan kahdeksaa eri kokoonpanoa valituista metodeista testattiin simuloimalla eri aikavälien transaktioskenaarioita, joista yksi valittiin sen parhaan suorituskyvyn perusteella. Valittua kokoonpanoa verrattiin kahteen muuhun eri toimintaskenaarioon: optimoimattomuuteen ja täydellisen ennustusmetodien omaavan työkalun käyttöön simuloimalla kahta eri varastosaldoskenaariota samalla transaktioskenaariolla. Tuloksista pystyttiin tekemään johtopäätöksiä työkalun hyödyllisyydestä ja mahdollisista kehitystarpeista.

Lopputulokseksi saatiin ehdotetun työkalun eri osien metodit ja niiden kokoonpano sekä data-arkkitehtuuri työkalun implementoimiseksi haluttuun, ERP-järjestelmän kanssa yhteensopivaan, ympäristöön. Lisäksi työkalun lisäkehityksen kannalta mahdollisia lisätutkimuksia käytiin läpi.

ABSTRACT

Author: Aleksi Toivanen

Title: Warehouse stock optimization for Ramirent Finland Oy

Year: 2020 **Place:** Lappeenranta

Master's thesis. LUT University, Industrial Engineering and Management, Business Analytics.

120 pages, 15 tables, 48 figures

Supervisor(s): Pasi Luukka, Jan Stoklasa

Keywords: Construction equipment, rental, quantitative forecasting, demand forecasting, intermittent demand forecasting, optimization, decision-making support, data analytics

The purpose of this master's thesis is to develop a tool model to support the decision-making of moving Ramirent Finland Oy's rental fleet between its branches. The thesis is based on a literature view and earlier studies of the research topic, which include quantitative forecasting and different optimization methods.

During the thesis the transactional data of rentals from past few years of the company were observed and analyses to select the most suitable demand and rental times forecasting methods were conducted together with the knowledge acquired from the literature review. A total of eight different compositions of the chosen methods were tested by a simulation analysis of different transaction scenarios and the best performing composition was chosen. The chosen composition was tested against two different operating scenarios: not doing optimization and using a tool which has no error in its forecasts with simulation analysis of one transactional scenario with two different total rental fleets. From the results useful knowledge about the usefulness and further development areas of the tool was extracted.

The final result included the composition of the methods for the tool, the framework for communication of these methods and the data architecture to implement the tool into an environment compatible with ERP-system. Further development of the tool and proposed studies connected to it were also discussed.

ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude towards Ramirent Finland Oy and especially to my superior Aleksi Kivistö for granting me the opportunity to write my thesis on such interesting topic, even though there were a lot hindering factors such as COVID-19 and the ERP transition process of the company.

I have been working as a master data specialist in Ramirent Finland Oy as a part time for a few years now, which together with my studies gave me and excellent starting point to work on this subject. I also would like to express my appreciation to my thesis supervisor Jan Stoklasa for always providing the help I needed and challenging my opinions.

It is also great to see that there exists a positive atmosphere in Ramirent Finland Oy towards the ever-increasing need of the usage of data-analytics when conducting business. I hope that the realized benefits of the usage of the warehouse stock optimization tool will further encourage the company to consider the usage of data-analytics in its operation.

Even though the problem at hand first seemed like an unclimbable mountain, through dedication, trial and error and countless hours of computational usage, that which was only a mere idea of the tool, was eventually able to be made into reality.

TABLE OF CONTENTS

ABBREVIATIONS AND SYMBOLS

Abbreviations

ERP	Enterprise resource planning system helping the business to operate by providing modules for different business functions under one software
SKU	Stock keeping unit
MA	Moving average
nmh	Number of months of history available at the moment
SES	Simple exponential smoothing
SBA	Syntetos-Boylan approximation
ANN	Artificial neural network
NN	Neural network
MLP	Multi-layered perceptron
BP	Back-propagation
ADI	Average inter-demand interval
APE	Absolute Percentage Error
MAPE	Mean Absolute Percentage Error
MSE	Mean Squared Error
SSE	Sum of Squared Errors
ME	Mean Error
MAE	Mean Absolute Error
PB	Percentage Better
PBt	Percentage Best
RGRMSE	Relative Geometric Root-Mean-Square Error
CDF	Cumulative Distribution Function
LOP	Linear optimization problem
NLP	Nonlinear program, nonlinear programming
SPP	Shortest Path optimization model
MCF	Minimum Cost Flow optimization model

TPR	True Positive Rate
AWS	Amazon Web Services

Roman symbols

t	Time period
a	Demand intercept
b	Demand slope
r	Seasonal demand ratio
d	Seasonal demand increment
a`(t)	Forecasted moving average at time *t*
N	Number of months for history demand on moving average
N`	Temporary number of months for history demand on moving average
$\bar{Y}$	Smoothed average (demand) of series
$\hat{y}(t)$	Estimate of demand at time *t*
y(t)	Actual demand at time *t*
D(t)	Observed value of nonzero demand at time *t*
Q(t)	Actual inter-arrival time of transactions at time *t*
P(t)	Estimated inter-arrival time of transactions at time *t*
p	Probability of scenario on a random walk
p'(t)	Probability of a demand occurrence at time *t*
Z(t)	Estimated value of non-zero demand at time *t*
U(t)	Forecasted demand per unit time
L	Lead time, L≥1
$\bar{L}$	Mean lead time duration
m	Total number of demand time periods contained in a time series
r	Number of periods that was used for initialization purposes, i.e, not considered for generating results
n	Number of demand time periods considered for the purpose of comparison; n=M-r-L

$CV^2(x)$	The squared coefficient of variation of demand sizes
E,e	Forecast error in demand period
$E(X_n)$	Expected value of a random walk
n(ctCDF)	The amount of transactions in the used CDF which have equal or greater value of relative rental time to the current transaction
n(CDF)	The total amount of transaction in the used CDF
p(ct)	The probability of the return of an item for current transaction
w, z	Objective function for optimization problem
]a, b[	Open interval between a and b, and a, b ∈ ℝ
[a, b]	Closed interval between a and b, and a, b ∈ ℝ
$\hat{S}$	Estimated stock level
S	Stock level
C_{ij}	Unit moving cost of a product from location to another
R	Daily rental price of a product

Greek symbols

μ	Average demand
γ	Random noise in demand series
α, β	Smoothing parameter
λ	Mean demand arrival rate
ℝ	A set of real numbers

1 INTRODUCTION

This chapter includes the background of the study, the objective and scope, structure of the thesis and the used methods of this study.

1.1 Background

To my best knowledge, there has not been too much published research about using data-analytics in order to optimize warehouse levels for rental items in rental businesses. This could be caused by the fact that depending on the business model it might be hard to predict the consumer behavior simply based on past data when it comes to for example the rental times and the demand of the rental product. What also makes this a hard subject is the fact that even though the customer behavior could be modeled accurately, there still would be a lot factors to consider in the stock level optimization as the products cannot simply be ordered, but they have to be transported from a service point, leaving a dent that service points stock level. Taking all of this into account, the problem presented in the thesis can be divided into three different areas: forecasting the demand for products, forecasting the rental times for the rented products and the stock level optimization based on these forecasts.

Management is continually facing the problem of knowing the future demand for their product, which means a forecast of some type should be used as a basis for solid decision-making when taking actions to meet this demand. The more reliable the forecasts are, the better the outcome for the planning and decisions made. The common goal is to have the least amount of inventory to satisfy the customers' needs for the product. Fortunately, this is not a new problem, as forecasting has been a common problem for businesses for centuries and the increasing computational power on our hands enable the usage of more advanced forecasting methods. The demand forecasts are typically monthly and updated as the new monthly demand entry is usable for the model. (Thompoulos, 2015, p. 1-11)

After successfully dealing with the daily demand forecasting, another problem presents itself: the estimation of rental times of the products. Unlike in forecasting the demand for consumption products in for example retail business where the consumer buys the product and consumes it,

in rental business the product is supposed to be returned after the usage. On paper, taking into account the return times of the tools sounds simple: the customer provides the company an estimate of the rental time, pays for that amount and returns the product after the time is up. However, this is system fairly flexible as the customer has the option to return the product before the set time and get a fair reimbursement of the non-used rental time, to apply for time extension for the rental or simply just keep the product as long as they want and pay when returning the product. However, there is an incentive for the customer not to extend the rental time without informing the company as the customer is required to pay 50% increased rental price for the time extension.

This leads to an assumption that there should exists different types of distributions for the rental times of different items, which might be of use in predicting the actual rental time for a unique rental. However, these distributions should be assumed to be non-normal as there are great number of real-world examples of skewed or heavy-tailed data. For instance, data on family income, CD4 count data from AIDS studies etc., meaning that there might be a need to develop a flexible model that can readily adapt to the non-normality behavior of the data. (Dávila et al., 2018, p. 11)

The last problem is how to optimize the stock levels of different items on different locations based on the demand forecasts and the rental time estimates of the unique rentals. First the model needs to be formulated so that an optimization algorithm can be used to find its solution. After this it comes to finding or building a suitable optimization algorithm as there is no universal solution for this, but rather a collection of algorithms, each of which being tailored to a particular type of optimization problem. (Nocedal & Wright, 2006, p. 2) However, it is suitable to assume that the optimization algorithm chosen will be of revenue optimization type as it is seen as the most suitable attribute to be optimized.

It is also good to acknowledge for the optimization method chosen for the tool that Ramirents fleet consists of several types of items, varying from handheld tools to large cranes, which could require different means of transportation and have different transportation times.

1.2 Objective and scope

The objective of this thesis is to build a warehouse stock optimization tool model for Ramirent Finland Oy's (regarded as Ramirent in some instances) rental fleet. The objective is not to build the tool in its working environment as this will be done in later by Ramirent, but to build a working model of the tool and develop a common data architecture to support the implementation. The goal of the tool is to offer prescriptive analytics to the user to support decision-making related to making product transportations between different branches of Ramirent. The fleet of Ramirent includes a product hierarchy from a unique item all the way to a product line of items. Between these two are other subgroups such as the make and model hierarchy and product group and "catclass" hierarchies, which include a larger set of same-like products such as all small vacuums or cordless drills.

The model will be built in such a way that it will operate on product group level meaning for example all same-like cordless drills will be handled as a same type item. As this product group hierarchy is already established by Ramirent no extra steps are needed in order to form one. Ramirent is currently in progress of switching their ERP system, meaning that the now in a daily usage product group hierarchy will be entirely replaced by a "catclass" -based hierarchy. This should not affect the model as the catclass hierarchy largely resembles the product group hierarchy, but it must be taken into account when designing the model in order to prevent any backlash caused by this soon occurring switch.

Also, as earlier stated that the rental fleet of the company varies from large tower cranes to small handheld tools, to not include the redundant products the model will be constrained so that it only takes into account products and locations which are provided to it. This is done by acquiring a list from each service point of their chosen optimal stock levels for a certain item type, which will enable to model to know what products should be available in which service points and if there are products that are not going to be part of this model at all. After knowing that only easily transportable suitable products will be included in the tool, it is determined that the tool will work in such a way that it attempts to balance the stock levels on a daily basis.

The purpose of the tool will be to deliver quick insight for the personnel responsible on moving items from a service point to another in order to make the best use of the available fleet to meet the customer demand and a successful implementation of the tool should help Ramirents personnel in decision-making related to the moving of products from a service point to another by providing optimized decisions based on maximizing the estimated revenue of available product transport options. This should also boost some of the fleet related KPIs used by Ramirent by increasing the utilization rate and yield of their fleet and generate more profit from the usage of its fleet to the company.

To build a successfully performing tool model all the three main components of the model must perform on a satisfactory level individually in order to form a model which can perform on a satisfactory level. This means that if even one component fails to uphold satisfactory results there is a risk for the whole model working poorly. Because of this, the following research questions and goals are set for this thesis:

- Which of the investigated forecasting methods are suitable for the daily demand of chosen rental product groups for an individual service location?
- Identify methods that are suitable to estimate rental time for a rental transaction.
- What combination of suitable demand forecasting methods and rental time estimation methods for the created tool model produces the best desired outcome?

For a method to be suitable it must be deemed to be applicable for the presented forecasting problem and if there exist many comparative methods, the performance of them on the given problem based on chosen performance metrics is to be investigated and based on this a subset of them can be chosen to represent suitable methods. In the subset of chosen suitable methods, there should not exist a method that is explicitly superior compared to other methods in the subset for the problem at hand. The best desired outcome can be defined by many ways for example based on the generated revenue or the service percent generated by the tool model. This means the performance of different combinations of suitable forecasting and rental time estimation methods will be measured with multiple variables and the combination which produces the best desired outcome is chosen based on the performance of all these variables.

1.3 Research methods and structure

This thesis is divided into three parts first of them being a literature review of the subjects related to the model built. These subjects include frameworks for intelligent data analysis, demand forecasting, estimation methods for rental times and different optimization methods for transportation of the products. Other subjects discussed in this part are measuring the accuracy of the used methods and the introduction of the case company. The literature used in this part are books and articles gathered from scientific literature databases by deeming them a trustable and relevant source of information for the topics of this thesis.

The second part is about the quantitative research for best methods to be used in the three major areas of the built model: forecasting method, rental times estimation method and the transportation between locations optimization method. The data is provided by Ramirent from their ERP system, which has many years of transactional data stored from different products and locations. The data related research is conducted as an observational study as the data generating process is not manipulated but is rather just data gathered of normal transactions. The methods to be used are picked into testing based on the literature review and chosen by their performance into the model.

The last part is about building a model for the tool and the data architecture needed in order to build the tool in its working environment. It also includes conclusions of the thesis subjects and discussion about further development for the model and other research possibilities related to this thesis.

2 INTELLIGENT DATA ANALYSIS

Every passing year brings more powerful computers for us to use with more computational power, cheaper storage for our data and higher bandwidth data connections. This means that it is easier and more tempting for us to gather and document data bringing us to the point where increasingly many enterprises, research centers and governments are creating huge data storages of all sorts. (Berthold et al., 2010, p. 1) For example NASA observation satellites generate a terabyte of data daily, many businesses maintain large Data Warehouses of customer transactions containing hundreds of millions transactions and there are huge amounts of data documented every day automatically by devices including data such as credit card transaction files and web logs. (Bramer, 2016, p. 1)

For centuries the lack of data has been a focal point of the problem for scientific and economic progress, which has led to a common belief that any problem we are facing can be solved, at least in theory, by having enough data on our hands. However, this is a false assumption as data alone, regardless of how voluminous it would be, is not enough for problem solving. This is because even as huge data archives enable the retrieval of several different points of information and to calculate different needed factors (for example average monthly sales in a service point), general patterns, structures and other regularities often stay unnoticed. The problem can be phrased by Berthold et al. (2010, p. 1) as "*we cannot see the wood (the patterns) for the trees (the individual data records)*" or "*We are drowning in information, but starving for knowledge.*" and by Bramer (2016, p. 2) as "*the world is becoming 'data rich but knowledge poor'.*".

Even though a lot of the actual data stored are only to be observed on a superficial level, if at all, there has been a growing realization that the stored data contains buried within it knowledge which can be the utmost important for a company to succeed. Suiting this need of knowledge extraction from the massive volumes of data we possess, a research area which has become known as data mining was developed. The objective in data mining is to meet the task of developing tools that help us to find useful patterns in the gathered data and solve the problems faced by better capturing the information which it carries. (Berthold et al., 2010, p. 1-2; Bramer, 2016, p. 2)

As of today, a lot of progress has been made in this area and numerous methods and implementations of techniques in software's to suit these needs have been developed. However, the existence of these implementations is not enough as Berthold et al. (2010, p. 2) clearly state that "*Still it is not the tools alone, but the intelligent composition of human intuition with the computational power, of sound background knowledge with computer-aided modeling, of critical reflection with convenient automatic model construction, that leads intelligent data analysis projects to success*.".

2.1 Data and knowledge

To understand the progress undergone in intelligent data analysis projects, it is important to determine the difference between data and knowledge. The essential property of data is that it refers to a single event, object, people, points in time etc. (Berthold et al., 2010, p. 2) The data points measured can be referred as objects, which can be described by several variables that represent the object's properties. In data mining these variables are often called attributes, but both terms can be used. Furthermore, a set of variables corresponding to each of the objects is often called a record or can also be referred as an instance. The complete data on hand is usually referred as a dataset.

In general, there exists several types of variables to be used in describing the properties of an object and not understanding the differences between these various types might cause problems when doing any form of data analysis. Bramer (2016, p. 10-12) distinguishes six main types of variables:

- **Nominal variables**: this variable is used categorize objects, for example a type of a car. A nominal variable can be expressed in numerical form, but these numbers would have no mathematical interpretation.

- **Binary variables**: a nominal variable which takes only two possible values for example true or false, 1 or 0, yes or no etc.

- **Ordinal variable**: like nominal variables, except these variables can be arranged in an order for example short, average, tall.

- **Integer variables**: genuine integers e.g. a number of demand or a number of days, which also means that arithmetic with integer variables is meaningful. Variables of this type are a subtype of cardinal variables.

- **Interval-scaled variables**: variables taking numerical values on a measurement with equal intervals from a zero point or origin are called interval-scaled variables. A well-known example of interval-scaled variable is Celsius temperature scale. One can definitely say that one Celsius temperature is greater than the other, but to say that one temperature in degrees Celsius is three times the other is meaningless since the zero value has been chosen arbitrarily and doesn't indicate that there would be a nonappearance of temperature. If the temperatures are transformed to an equivalent scale for example Fahrenheit, the same relationship is not valid anymore.

- **Ratio-scaled variables**: these are like the interval-scaled variables except the zero point now signifies the absence of the measured characteristic. An example of ratio-scaled variables would be Kelvin temperature or molecular weight.

Even though the distinction between different types of variable is sometimes important, usually practical data mining systems divide variables into only two types:

- **Categorical**, which corresponds to nominal, binary and ordinal variables
- **Continuous**, which corresponds to integer, interval-scaled and ratio-scaled variables. (Bramer, 2016, p. 12)

In contrast to data, knowledge implies documentations of occurrences which can be sets of objects, events, points in time etc. and unlike data, knowledge allows us to make predictions and forecasts about the future events. The descriptions of knowledge make it very clear that generally its value is greater than of raw data. Obviously not all kinds of knowledge are equally

valuable meaning that the knowledge acquired must be assessed. For this Berthold et al. (2010, p. 3) present the following list of a few of the most essential criteria in assessment of knowledge:

- **Correctness**: probability, success in trials
- **Generality**: validity and the conditions and domain of it
- **Usefulness**: applicability and predictive capability
- **Comprehensibility**: transparency, simplicity, clearness
- **Novelty**: not yet known, not expected

When it comes to science, the emphasis is on correctness, generality and comprehensibility of the knowledge. However, in economy and industry and thus also in this thesis, the focus is set on usefulness, comprehensibility and novelty as the main goal is to gain a competitive edge for revenue increase and other purposes.

A well-known historical example to illustrate the distinction between data and knowledge is a one from the sixteenth century study of the stars and planetary motions, a core area of research of its time. Tycho Brahe, a Danish astronomer collected fairly large amounts of data about our planetary system by observing and determining the positions of the planets and the sun with a precision of less than one angle minute. However, he failed to find a consistent scheme to combine his observations. This was caused partly by his intention to stick closely to the geocentric system (the planets revolve around the earth). It could be said that he had a “data analysis problem” as he had obtained the necessary data but could not extract the knowledge hidden in it.

This problem would later be solved by Johannes Kepler, a German astronomer, who worked as an assistant of Tycho Brahe. He started investigating the data Tycho Brahe had collected and unlike Tycho Brahe, he supported the Copernican planetary system (the planets revolve around the sun). After several fruitless trials, Kepler finally succeeded and managed to combine Tycho Brahe’s data into three laws, which are these days known as Kepler’s laws. By using the large amounts of data collected by Tycho Brahe and discovering the hidden knowledge in it, Kepler set up a place for himself as one of the most known “data miners” in the past.

Today the works of Tycho Brahe are almost forgotten, while Kepler's laws are treated in basically all astronomy and physics books. (Berthold et al., 2010, p. 4-6) This comes to show the distinction between data and knowledge and the possible repercussions of not being able to draw knowledge from the gathered data.

2.2 Intelligent data analysis process

There exist different suggestions on what the intelligent data analysis process should look like. However, Berthold et al. (2010, p. 8-9) mentions three separate frameworks for modeling this process: SEMMA (an acronym for sample, explore, modify, model, assess), CRISP-DM (an acronym for Cross Industry Standard Process for Data Mining) and the KDD process.

The three mentioned frameworks for intelligent data analysis project have different strengths each of their own. However, SEMMA and CRISP-DM can be seen as an implementation of the KDD process as KDD process is mostly used by researchers and data mining experts due to its more complete and accurate nature. SEMMA and CRISP-DM can be considered as more suitable frameworks when dealing with business problems not requiring expert level expertise. This can also be seen in the strong industry involvement on these models compared to KDD, which is partly caused by the fact that these two are more simple for the user whereas using KDD process requires a background in DM. (Kurgan & Musilek, 2006, p. 16; Azevedo & Santos, 2008, p. 5; Shafique & Qalser, 2014, p. 221) Based on these points, it is decided that KDD process will not be used as the intelligent data analysis framework in this thesis.

Now when comparing the two remaining frameworks CRISP-DM and SEMMA, Shafique & Qalser (2014, p. 221) argue that CRISP-DM is more complete as to compare to SEMMA. However, Azevedo & Santos (2008, p. 5) point out that at first sight it can seem that CRISP-DM is more complete than SEMMA, but analyzing it deeper, one can argue that they both achieve the standards concerning the overall process: SEMMA and CRISP-DM do guide people to know how DM can be applies in practice in real systems. That being said, SEMMA was still developed by only one enterprise whereas CRISP-DM was developed by the means of the effort of a consortium of large companies. CRISP-DM framework also seems to have more available and detailed literature to my best knowledge, which improves the understandability of its whole

process and the subprocesses connected to it. This also makes it easier to access more detailed explanation of the whole process of CRISP-DM in case questions about different subprocesses of it arise. Taking all of this into account, it is chosen that CRISP-DM framework will be used in the thesis.

The following chapter describes the CRISP-DM process in more detail and also the more detailed descriptions of SEMMA and KDD processes can be found in appendices A and B.

2.3 CRISP-DM methodology

CRISP-DM process was developed by a group of large enterprises including Daimler, SPSS and NCR. It seems to be the most widely used process model nowadays for intelligent data analysis projects. The CRISP-DM methodology is described with process model consisting of six phases, most of which are usually executed several times. (Berthold et al., 2010, p. 8; Chapman et al., 2000, p. 6) Chapman et al. (2000, pp. 10-11) describe the six phases as followed:

- **Business understanding**: this phase is about understanding the project objectives and needs from a viewpoint of a business, after which the knowledge of this is converted into a data mining problem definition. Also, a plan for achieving the set objectives is formed.
- **Data understanding**: the second phase begins with data gathering and continues with actions to understand the data, identify possible problems of its quality, form first insights of the data and possibly find subsets to form hypotheses concerning unseen information within the data.
- **Data preparation**: this phase includes all steps required to form the used dataset from the raw data to be used in the modeling tool(s). These activities are likely to be performed multiple times without any specified order. This includes the selection of tables, records and attributes, as well as cleaning and transforming the data for modeling tool(s).

- **Modeling**: different modeling methods are chosen and applied, and the parameters used in them are adjusted to ideal values. Going back to the data preparation phase might be needed as different methods might require different forms of the used data.
- **Evaluation**: at this phase, the model(s) that is qualified from a data analysis perspective has been built. Before moving forward, it is important to thoroughly evaluate the model and review the steps executed in the creation of it to be certain of the model's ability to meet the set business objectives. A key objective here is to confirm that all important business issues have been considered as a sufficient level. After these evaluations, a decision on the use of results is be done.
- **Deployment**: the development of the model does not always mean the end of the project as in most cases the gained knowledge needs to be arranged and presented in a such form that it can be used. The deployment phase can be as simple as producing a report or as complex as implementing a repeatable data mining process across the entire business. The usual case is that the customer handles the deployment, so it is important for the customer to know, in collaboration with the analyst, up front the actions needed to be performed in order to guarantee the successful usage of the built models.

Figure 1 shows the traditional CRISP-DM model presented by Chapman et al. (2000, p. 10) and figure 3 shows the more user-friendly version with some key notes of this presented by Berthold et al. (2010, p. 9). It is worth noting that the business understanding has been written as project understanding in the latter model as the term emphasizes that the faced problem might be purely technical in nature or a research project rather than financially motivated one.

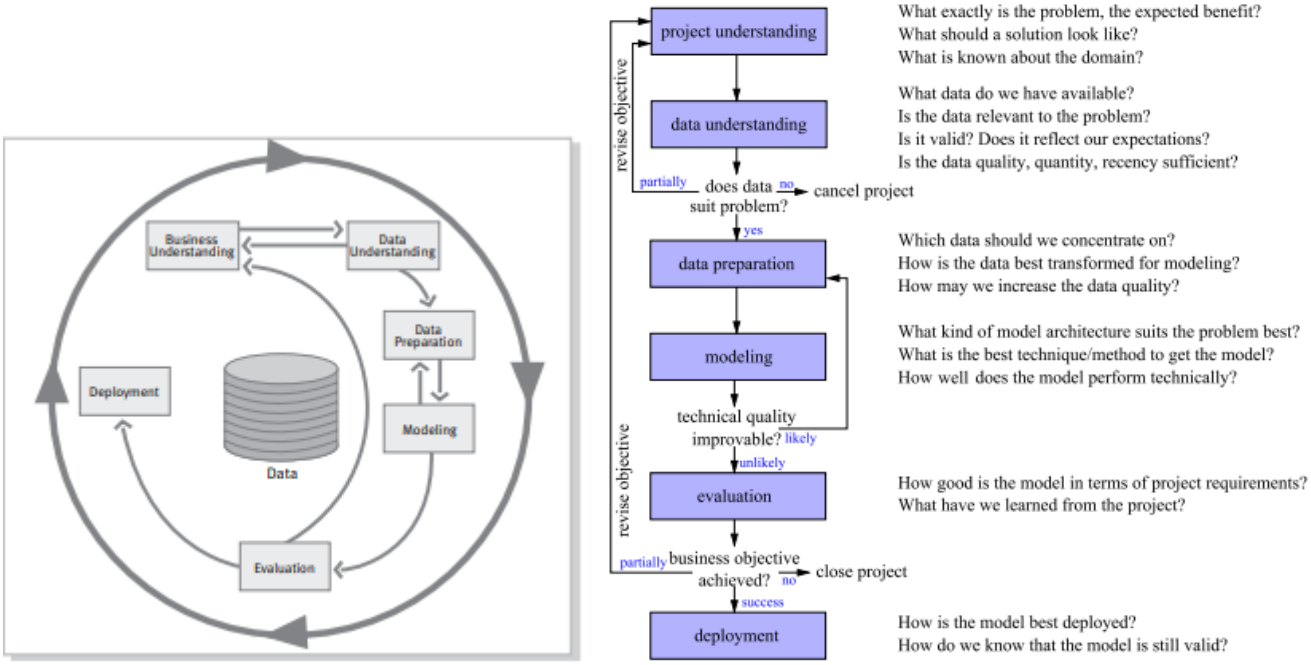

Figure 1: traditional CRISP-DM model (Chapman et al., 2000, p. 10).

It is also worth noting that in addition to these simple steps, both literatures have reserved entire chapters to give out a detailed description of all the possible sub-processes contained in these steps. The sub-processes are not discussed in this thesis as it is not the main subject, but the information about them can be found in these two literatures.

3 DEMAND FORECASTING IN RENTAL BUSINESS

As earlier mentioned, management is continually faced with the problem of knowing the future demand, leaving a need for forecasting methods to be used as a basis for decision-making for the actions needed to meet this demand. This chapter discusses the benefits of demand forecasting and the default patterns of demand while examining which of these patterns might be present in the sales data acquired from Ramirents ERP. In the chapter qualitative and quantitative forecasting methods are also discussed and a great emphasis is placed for methods that are suitable for the possible demand patterns found in the data.

3.1 Introduction to demand forecasting

"*If a man gives no through about what is distant, he will find sorrow near at hand.*", these are the words of a Chinese philosopher Confucius, described by Armstrong (2001, p. 1) as a statement to describe the importance of forecasting. This importance can be seen even in our everyday lives since as individuals, we try to predict the success of numerous different events such as commuting, occupations and investments.

On a higher level, organizations depend on forecasts when investing for new products and factories and government agencies require forecasts of the economy, environmental impacts and effects of planned actions. On the other hand, devastating decisions can be made when depending on poor forecasts, for example the construction of convention centers by U.S. cities based on wishful forecasts of demand. (Armstrong, 2001, pp. 1-2)

The term demand forecasting in all its simplicity is defined as the forecasts of demands for items in stock. Consumers who want to purchase the item for their immediate use are the ones generating this demand. The different items can be identified by various ways such as a part number, style number, product number etc. and the demand is for the item at a specific stocking location. The term stock-keeping-unit (SKU) can be used for and item at a particular location. (Thompoulos, 2015, p. 2) This can be reflected into the model to be built in this thesis as the items are identified with their product group and the demand is for the item at a specific service point location.

There exist various ways to interpret the process for forecasting whether it be for demand or something else, one of them being presented in figure 2. This framework will be used in order to select the best at hand forecasting method to be used in the model.

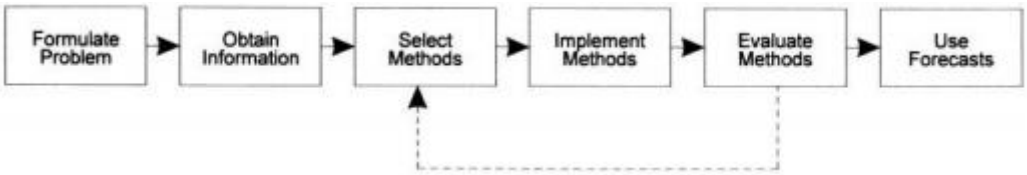

Figure 2: the forecasting process (Armstrong, 2001, p. 1).

3.2 Principles for demand forecasting

There exist different points to account for when performing successful and necessary forecasting. For example, decision makers need forecasts only when there is significant uncertainty about the future, meaning there is no need to forecast certain events. Forecasting may often also be confused with planning, which should not happen as planning concerns what the world should look like, whereas forecasting is about what it will look like. (Armstrong, 2001, p. 2)

To summarize knowledge of forecasting, Armstrong (2001, p. 3) uses a set of principles representing advices, guidelines, prescriptions, conditions-action statements and rules. Armstrong (2001, p. 4) argues that having such principles is of importance as they affect our behavior in a supportive way. Selecting and applying forecasting methods are often performed poorly in organizations, sometimes because managers have too much confidence in their intuition for which these principles should offer help for (Armstrong, 2001, p. 5). The following principles of Armstrong's can be seen as applicable or useful to the demand forecasting in this thesis:

- **Correct for biases in judgmental forecasts**: Judgmental forecasts have been greatly affected based on biases such as forecasting the wanted outcome (optimism bias).
- **Use longest time series available**: Forecasters should use the longest possible time series. This principle might conflict with the principle of using the most relevant data,

as it usually refers to the most recent data. Into this principle, it could be added that the accuracy of the demand forecasts is based on the accuracy of the demand history, as Thompoulos (2015, p. 11) states. If there has been changes to the underlying framework which causes the patterns of the time series one should consider what is the longest time series possessing similar underlying framework to todays.

- **Econometric forecasting models should be fairly simple**: Armstrong (2001, p. 8) presents studies which support the simplicity of econometric forecasting models. However, taken into account that these principles have been made in 2001 and they're based on even older literature, it should be carefully measured what kind of models can nowadays already be considered as fairly simple ones.
- **Theory should precede analysis of data in developing econometric models**: It has been concluded and showed that simple practices have been sufficient for combining earlier theory with regression estimates. (Armstrong, 2001, pp. 7-8)
- **Use quantitative methods rather than judgmental methods, if enough data exist:** Quantitative methods are expected to be more accurate than qualitative if enough data exists. To determines if there is enough data one must consider the data's source, amount, reliability, validity, relevance and variability. (Armstrong, 2001, p. 373)

Also, Armstrong's golden rule be conservative is seen as a helpful rule to increase the demand forecasting accuracy in the model. The long form is "*to be conservative by adhering to cumulative knowledge about the situation and about forecasting methods*" and it refers to set of rules which apply especially to forecasting problems when situation is uncertain and complex and bias can be expected. (Armstrong et al., 2015, pp. 1723-1725) Soyer & Hogarth (2015, p. 1702-1703) illustrated an excellent example while observing Armstrong's article, capturing the essentiality of these rules. The example tells about a turkey which is remarkably accurate in predicting its daily outcome of being fed, but fails to predict that it will be eaten at Thanksgiving dinner: the constant success on its predictions ultimately lead to its failure of predicting its own demise. Two key points can be derived from this: highly improbable events do occur, and it must be accepted that they cannot be predicted, and instead of just forecasting, protective actions for negative outcomes should be taken.

3.3 Time series and forecasting

As earlier discussed, demand forecasting is a term for forecasting the demands for items in stock. When these demand estimates have a time stamp, we can refer to this data as a time series, consisting of observations recorded at specific times. In a discrete-time time series, observations made are a discrete set: for example, when observations are made at fixed time intervals. In a continuous-time time series the observations are recorded continuously over a chosen time interval. (Brockwell & Davis, 2016, p. 1) This means that as demand forecasts are usually done by daily, monthly or yearly basis, the data included in them is a discrete-time time series.

When dealing with time series, unwanted components of the time series can be also removed. This includes for example unwanted seasonal components or noise. By doing this the user can gain more precise information in order to for example predict future sales trend or more accurate sales estimates for certain month. An important part of the time series analysis is the selection of suitable model for the data. (Brockwell & Davis, 2016, p. 5-6)

In order to fit the model to a given time series, the first step should be to plot the observations against a specific time. This will result into a graph, called time plot, which will show the main properties of the dataset, providing instantly possible insights about the data to the user. (Chatfield, 2005, p. 132) By doing this, it can be examined whether it contains a trend, a seasonal component, any apparent sharp changes in behavior or any outliers (Brockwell & Davis, 2016, p. 12).

3.4 Demand patterns

When it comes to demand patterns, there exist countless different kinds and the categorization of different demand patterns is important in the selection of a forecasting method and is an essential element of many software packages for inventory control (Syntetos et al., 2005, p. 501). Thompoulos (2015, p. 16) suggests that there are three basic ones: horizontal, trend and seasonal. According to literature, there however still exists one more basic type, which is referred as intermittent or as lumpy demand, in which the demand occurs in lumps and zero

demand intervals are present. (Armstrong, 2001, p. 222; Teunter et al., 2011; Syntetos et al., 2005; Gutierrez et al., 2005).

Perhaps the most basic of the demand patterns is the horizontal where the demands fluctuate above and below a path (called the level) without any trend or seasonal influence. This means that the average of the demand stays relatively same. Low volume parts demand is often of the horizontal type. Figure 3 illustrates an example of horizontal demand and the average demand for this type of a demand pattern at period *t* can mathematically be defined as follows:

$$\mu(t) = a$$

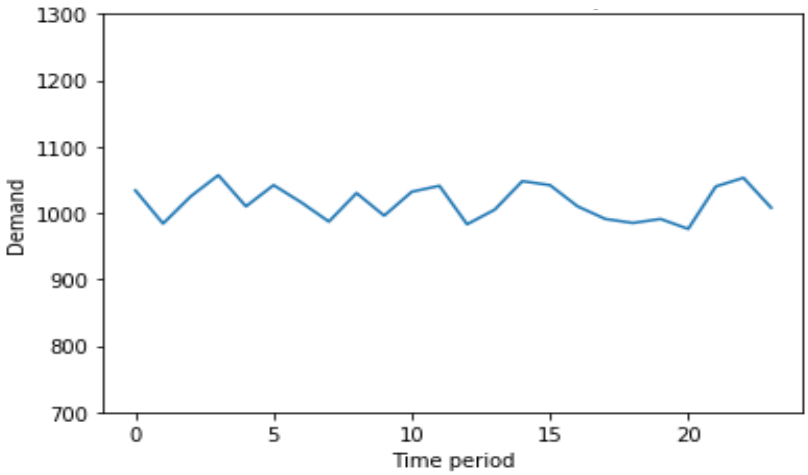

Figure 3: Horizontal demand example (artificial data).

Forecasting models existing to forecast horizontal demand are for example horizontal moving average forecast, horizontal discount forecast and horizontal smoothing forecast (Thompoulos, 2015, pp. 16; 23)

Items in the inventory can have demand patterns where the periodical demand level is gradually increasing or decreasing. When this happens, the demand is referred as trending. Figure 4 shows an example of trending demand of which average demand at time *t* can be defined as follows:

$$\mu(t) = a + bt$$

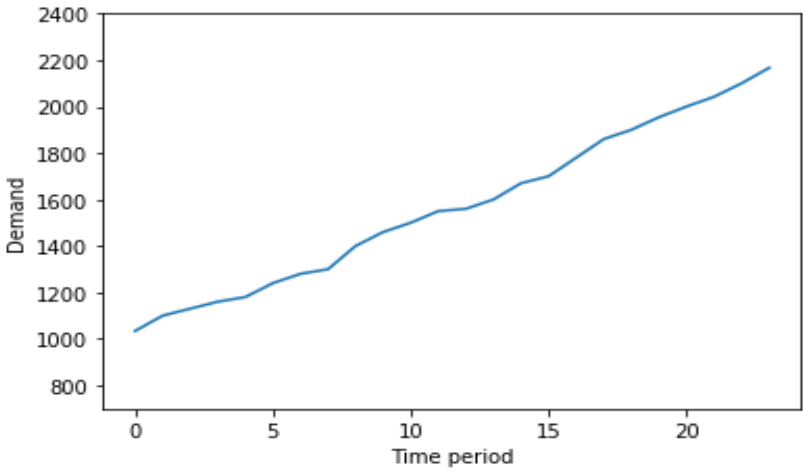

Figure 4: Increasing demand trend example (artificial data).

The model for trending demand has two coefficients, a and b, and such models include trend regression forecasts, trend discount forecasts and trend smoothing forecasts. (Thompoulos, 2015, p. 41).

Seasonal demand occurs when the demands vary monthly and the pattern repeats every year. Examples of seasonal demand are the increased demand for light clothes during the summers, school equipment in late summers, snow shovels during winters etc. Two varieties of the seasonal pattern occur: seasonal multiplicative and seasonal additive. Figure 5 demonstrates an example of seasonal demand, of which average demand at time t generally can be defined as follows:

Seasonal multiplicative:

$$\mu(t) = (a + bt)r(t)$$

Seasonal additive

$$\mu(t) = (a + bt) + d(t)$$

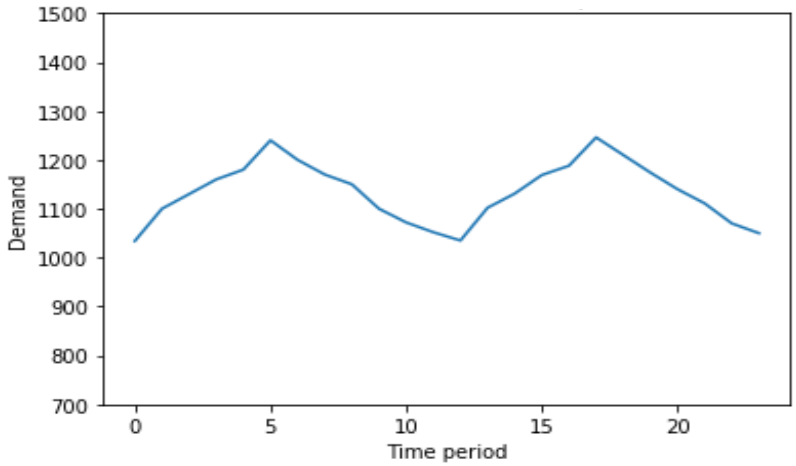

Figure 5: Seasonal demand example (artificial data).

Majority of the Ramirents rental products are affected by seasonal demand as the seasonal weather, vacations and other related factors make it so that the certain fleet is rented out more during certain time periods. This can be seen in figure 6, which shows the two-year daily demand of 10 rental products which are available in most of the service locations withing the service network of Ramirent during working days. These items include for example small grinders, vacuums and concrete equipment.

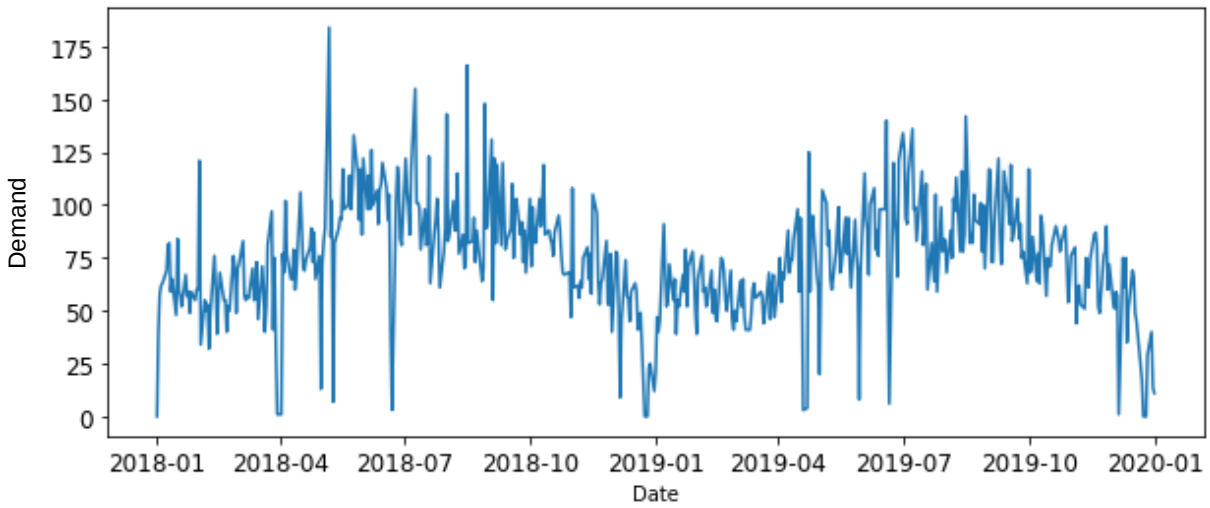

Figure 6: Ramirents 10 fairly available products demand in the service network on working days 1/1/2018-1/1/2020.

Some suitable forecast models for seasonal demand are seasonal smoothing multiplicative forecast model and the seasonal smoothing additive forecast model. (Thompoulos, 2015, pp. 16;59)

There exist many cases where the items in an inventory are demanded infrequently, resulting in a sporadic or an intermittent demand (Kourentzes, 2013, p. 198). An intermittent series (also called interrupted series, intermittent demand or irregular demand) is a non-negative series where orders occur in lumps and it contains one or more periods of zero demand. Intermittent demand patterns can be found in demand of heavy machinery and respective spare parts, expensive capital goods, seasonal goods such as grass seed or snow shovels, aircraft service parts, electronics and automotive spare parts. (Armstrong, 2001, p. 222; Kourentzes, 2013, p. 198)

The demand patterns for these products include difficulties for their demand forecasting and inventory control due to their nature of having demand arrivals that are linked with a demand size distribution. This means that forecasters are facing two forecasting problems: When will the next demand happen what will be the size of it? (Xu & Wang, 2012, p. 1468) Figure 7 illustrates an example of intermittent demand.

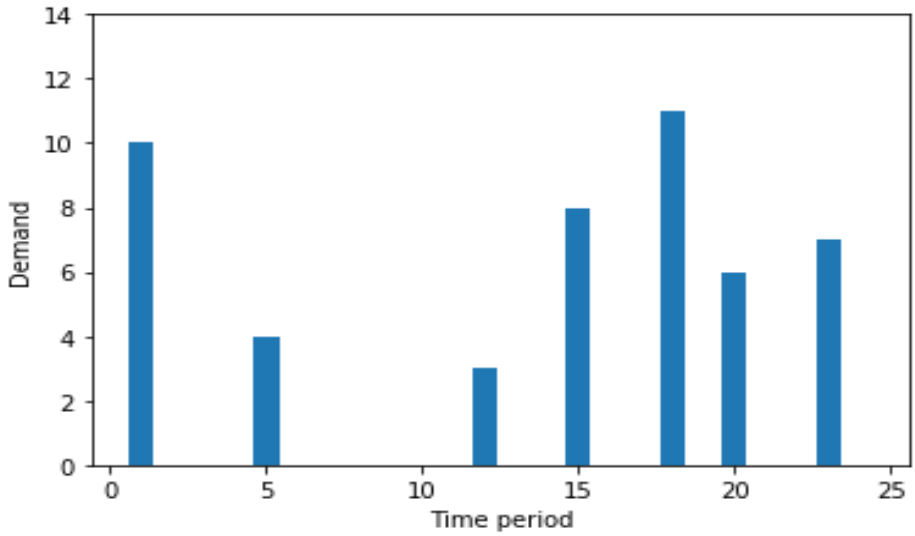

Figure7: Intermittent demand example (artificial data).

As demand forecasting methods can encounter difficulties because of the zero demand in certain periods, various methods can be used to adjust the series. For example, one way is to aggregate the interval of the time series so that it is long enough to rule out the intermittency of the demand series. This means that instead of using daily intervals containing zeroes, an aggregation to weekly, monthly or even quarterly intervals could be made efficiently eliminating zero valued periods from the time series. This however might be disadvantageous as the length of the interval might become longer than desired for decision making.

In addition to aggregation across time, aggregating across space is also possible. An example of this is to instead of looking at country data to look at region data or instead of region data to use national data. This however can create problems when the decisions are made on a country level. One way of also dealing with the problems of intermittent demand could be to change the structure of items for which the forecasts are made. For example, rather than forecasting for a particular size of a spare part, the data could be aggregated across lengths of the spare parts. (Armstrong, 2001, p. 223)

When it comes to the demand on Ramirents rental products, it can be seen from figure 8 that a single SKU demand for a service location on working days resembles intermittent demand pattern.

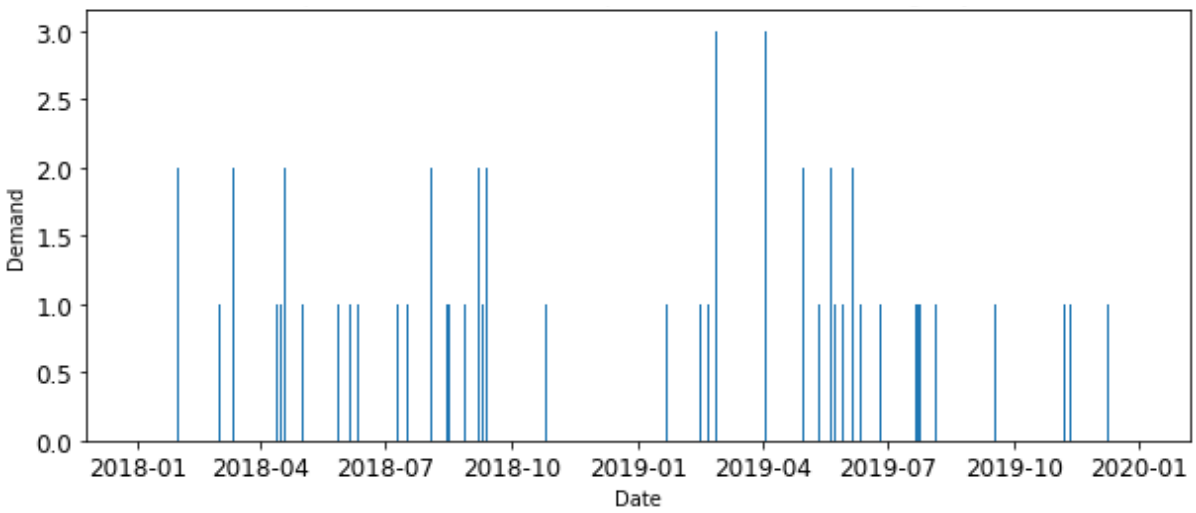

Figure 8: SKU based demand for single service point 1/1/2018-1/1/2020.

Moreover, figure 9 and 10 illustrate the weekly and monthly aggregated series for the same SKU.

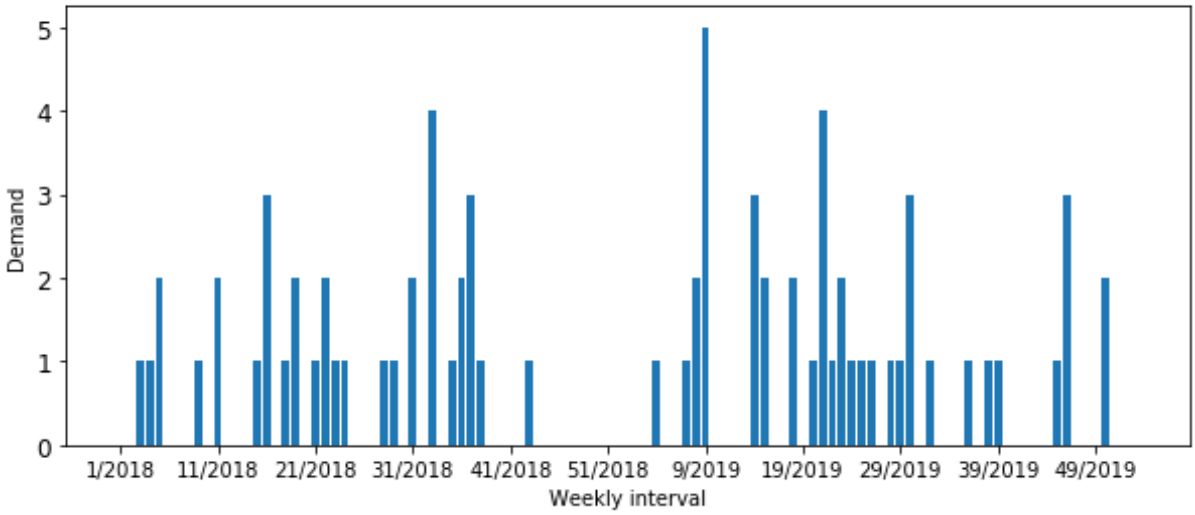

Figure 9: SKU based demand with weekly aggregation.

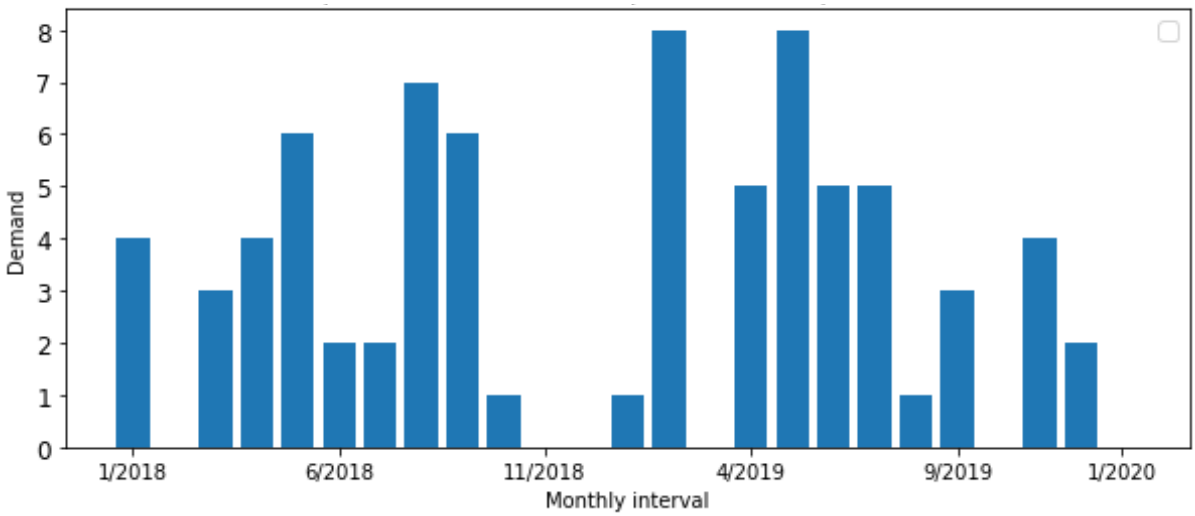

Figure 10: SKU based demand with monthly aggregation.

As it can be seen, even the monthly interval aggregation is not enough to eliminate the zero demand periods leaving the demand still fairly intermittent. One way to possibly get rid of the intermittency could be to aggregate the time series to be for wider area than one service location or group the product differently. These options however would most likely, if even if successfully eliminating the intermittency, be unsuitable or require too complex changes for the

operating ways of the company that they are outright discarded. Because of this, this thesis focuses on the forecasting methods suitable for intermittent forecasting.

Intermittent demand can be examined by looking at the times between demand spikes and the demand sizes which occur. Hence the time between the demand spikes explains how intermittent the demand is while the variability of the demand sizes points to the lumpiness of the demand. In 1984, a method of categorization of demand patterns was suggested based on variance partition. In the method, variance of the demand is split during lead time into its constituent parts. With this categorization it was possible to make subsets of available forecasting and inventory methods based on the suitability for different categories.

The items are categorized according to the matrix shown in figure 11 and the cutoff values are based on managerial decision.

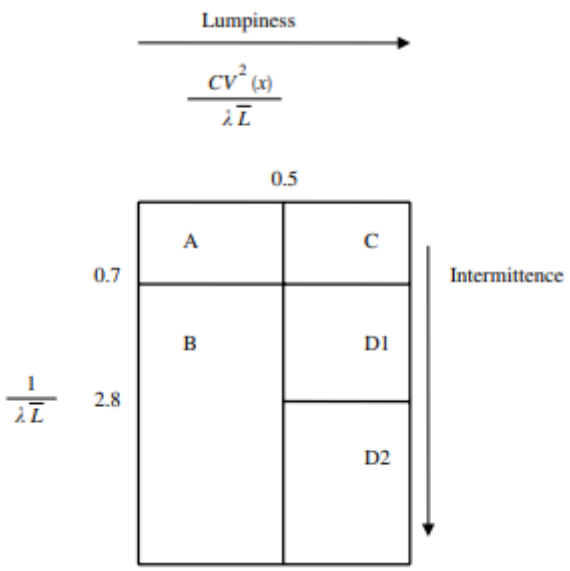

Figure 11: Intermittent demand categorization scheme (Syntetos et al., 2005, p. 496).

In the figure, $\bar{L}$ is the mean lead time duration, λ the mean (Poisson) demand arrival rate and $CV^2(x)$ the squared coefficient of variation of occurring demand sizes. $\frac{1}{\lambda\bar{L}}$ is the amount of lead times between successive demands (how often demand occurs meaning how intermittent the demand is) and the higher it is, the more intermittent the demand. $\frac{CV2(x)}{\lambda\bar{L}}$ implies the lumpiness of demand and both the intermittence and the variability of the demand size on occurring

demand affect it. The higher this ratio is, the lumpier the demand is. (Syntetos et al., 2005, p. 496)

Suitable forecasting methods for intermittent demand include simple exponential smoothing and Croston's method, which are both discussed in chapter 3.7. However, it has been found that Croston's method, while requiring more restrictive assumptions, is more robust and the superior forecaster compared to simple exponential smoothing. (Armstrong, 2001, p. 200) Other newer methods include a modification of the Croston's method called Syntetos-Boylan approximation and applying artificial neural networks in the forecasting (Gutierrez, 2008, p. 410). These are also further discussed in chapter 3.7.

3.5 Accuracy metrics for intermittent demand forecasting

Before going deeper into the methods used for demand forecasting, it is important to go through how the performance of these different methods can be measured. Understanding different methods of measuring forecasting accuracy can be important especially when forecasting intermittent demand. As Altay et al. (2012, p. 283) highlighted in their paper the used accuracy metrics in their research were not consistent in some cases and that the search for a good error measure for intermittent demand forecasting should be continued in order to find robust error measures for this. Additionally, the chosen measure of the forecast errors impacts the observed performance of the forecasting method (Xu & Wang, 2019, p. 1471).

The existence of some zero-demand time periods in intermittent demand and the very nature of it create major difficulties in selecting an appropriate accuracy measure. In addition, these special properties seem to have been even underestimated or completely ignored by both academicians and practitioners. This being said, before selecting accuracy measures for the purpose of comparing alternative methods in an intermittent demand forecasting, the methods which cannot be computed due to zero-demand time periods should be left out. These methods include all relative-to-the-series accuracy measures (e.g MAPE and MdAPE). The symmetric MAPE could be used but is left out as "*it is known to suffer from asymmetry problems*" as Syntetos & Boylan (2005, p. 307) point out. Also, all relative-to-a-base accuracy measures are left out as the forecast error in a specific time period on these methods is linked to some sort of

benchmark which usually results in a poor performance of the metric. (Syntetos & Boylan, 2005, pp. 306-307)

In this chapter two kind of accuracy metrics are described: Absolute accuracy measures and accuracy measures relative to another method.

3.5.1 Absolute accuracy measures

Absolute error measures are computed as a function of the forecast errors only, meaning there are no ratio calculations w.r.t the series itself or to other forecasting methods. (Syntetos & Boylan, 2005, p. 307) One of the most popular error metrics of this type is mean squared error (MSE). With a time series y_t at time t, the MSE is written as:

$$MSE_n = n^{-1} \sum_{t=1}^{n} (y_t - \hat{y}_t)^2$$

Where n is the number of in-sample observations. For the case of exponential smoothing forecasting method, MSE is the most widely used optimization criterium. (Kourentzes, 2014, pp. 182-183) However, as it is based on squared errors, it is more sensitive to errors and outliers bigger than one (Wallström & Segerstedt, 2010, p. 628). It is also prone to extreme values and as a result the mean absolute error (MAE) has been discussed by Kourentzes (2014, p. 183) as an alternative:

$$MAE_n = n^{-1} \sum_{t=1}^{n} |y_t - \hat{y}_t|$$

Although these measures can be calculated for intermittent demand when averaged across many time series, they don’t take into account the scale differences between them. An exception to the exclusion of absolute measures may be the mean error (ME), which is also known as mean signed error and is defined as:

$$ME = \frac{\sum_{t=1}^{n} |y_{t+L} - \hat{y}_{t,L}|}{n} = \frac{\sum_{t=1}^{n} e_{t+L}}{n}$$

Where $\hat{y}_{t,L}$ is the estimate (made in period t) of demand period $t+L$, obtained by any of the forecasting methods considered, y_{t+L} the actual demand in period $t+L$, and e_{t+L} the forecast error in period $t+L$. ME is fairly easy to compute and the straightforward explanation of it is that the difference between the absolute average ME yielded by any two methods tells how more or less biased is one of the methods compared to the other. As ME accounts for the sign of the error, it is considered less scale dependent than other absolute accuracy measures. (Syntetos & Boylan, 2005, p. 307)

3.5.2 Relative accuracy measures

Relative accuracy measures are not calculated as a function of forecast errors only as they are accuracy measures relative to another forecasting method. Methods like this include for example the percentage of times better (PB) which summarizes the performance of a method relative to another method across a set of time series in which the forecast error of the methods for each series is examined with absolute accuracy measure. Simply put, it calculates the percentage of times (observations) that the method outperforms the other. By this, it tells how many times a method outperforms the other, but it does not tell by how much. However, PB is great to be used for intermittent demand data as all series and all data periods within each series are contemplated to produce results.

In order to know by how much a method outperforms another one, a descriptive accuracy measure has to be used. The relative geometric root-mean-square error (RGRMSE) is a fairly used example of this type of a measure and it is defined for methods A and B in a specific time series as:

$$RGRMSE = \frac{\left(\prod_{t=1}^{n}(Y_{t+L} - Y'_{A,t,L})^2\right)^{\frac{1}{2n}}}{\left(\prod_{t=1}^{n}(Y_{t+L} - Y'_{B,t,L})^2\right)^{\frac{1}{2n}}}$$

On the case of having more than two methods which to compare, it can be useful to report the times when one method performs better than all other methods. For this, Percentage Best (PBt), which calculates the percentage of times a certain method outperforms all the other methods, can be used. (Syntetos & Boylan, 2005, p. 308)

3.6 Qualitative forecasting methods

In many situations, the first step of the process is to consult the experts. Forecasting done by this way can be referred as qualitative forecasting or judgmental forecasting. Sometimes this is enough as experts may have excellent insights thus make outstanding forecasts. However, expert opinion is subject to biases and shortcomings. As discussed earlier, judgmental forecasts should be done when there is not enough data to perform quantitative forecasting. Situations like these might include for example a new product launch, where there is no previous useful data to perform quantitative forecasts.

Judgmental methods can be divided into two categories: those that predict one's own behavior and those in which experts predict how others will behave. Such methods that predict one's own behavior are for example intention methods and conjoint analysis. The methods using experts predictions on how others will behave include expert opinions and judgmental bootstrapping. Both of these methods will lead in the usage of expert systems, which represent the rules that experts use.

There can exist number of stages in the forecasting process, such as defining the forecasting problem, method choosing, method application, forecast comparison, estimation of the forecasting uncertainty, forecasts adjustment and evaluation of forecasts. There is a risk that some of these stages are performed in a non-optimal way involving judgment in some level. To reduce the inconsistency and bias of the forecasts, one should prefer using forecast records for feedback, prefer graphical displays over tabular data displays, use multiple methods in the uncertainty assessment of the forecasts, use different people in assessing the chances of a plan's success and developing and implementing it and respect the importance of establishing agreed criteria for selecting forecast methods. (Armstrong, 2001, pp. 9;58-59;616)

In addition of just performing qualitative forecasting, the integration of qualitative and quantitative methods for demand forecasting has been explored. The findings on this area have shown that judgmental forecasting can support quantitative forecasting and vice versa. (Salehzadeh et al., 2020, p. 1719)

3.7 Quantitative forecasting methods

Whereas qualitative forecasting methods are based on human opinions, quantitative forecasting methods are based on data. These methods have kept on adapting ever since the data storing has become more efficient and available to everyone making it easier to store demand data from the past time periods and develop more powerful demand forecasting methods. (Thompoulos, 2015, p. 4) These conservative extrapolation methods for forecasting can be used with time series data or cross-sectional data.

Extrapolation is no longer conservative if knowledge about the situation that is not included in the time series or cross-sectional data is at odds with the extrapolation. Because of this, there have been tries to incorporate judgments into extrapolation, resulting in different approaches in combining more knowledge into extrapolations. Such approaches include for example using the longest time series of valid and relevant data, modifying trends to incorporate more knowledge, decomposing by casual forces and modifying trends or seasonal factors if the series has too few observations or it is variable or unstable. (Armstrong et al., 2015, pp. 1723-1725)

Even though more powerful models might be developed, some of the oldest models created are still in common use today. (Thompoulos, 2015, p. 3) This chapter presents some of the most used forecasting methods especially when dealing with intermittent demand patterns.

3.7.1 Naïve models

Models known as Naïve 1 and Naïve 2 can be categorized into time series forecasting models and they are of the simplest kind forecasting methods for time series. The Naïve 1 or no change model forecasts the value of demand as the value of last observed demand for example $\hat{y}_{t+1} = y_t$ on annual data, which simply implies that the forecast for the next year should be the actual

value of this year. This can be applied to any timeframe of observations, be it quarterly, monthly or daily. The Naïve 1 model is sometimes present in forecasting studies as it can be a good base comparison for other used models. Surprisingly, there have been suggestions that it performs better than the more formal forecasting methods in many cases.

The Naïve 2 model is a bit more sophisticated as it uses the growth rate of previous period to forecast the current period. The model is the following for annual data:

$$\hat{y}_{t+1} = y_t \left[1 + \frac{y_t - y_{t-1}}{y_{t-1}}\right]$$

For example, if y_{2019}=80 and y_{2018} = 60, then the quantity $\left[\frac{y_t - y_{t-1}}{y_{t-1}}\right]$ = 0.33, implying a growth rate of 33% from 2018 to 2019 meaning that the forecast for 2020 would be 80*[1+0,33]. As for the Naïve 1, also this method can be used on any timeframe of observations. The observation recorded one time period ago is called a lag of 1 time period. (Aljandali, 2017, p. 88)

3.7.2 Simple moving average

Simple moving average uses a parameter N that specifies the number of time periods of history demand to use in calculating the average demand for the current time period. The average consists of the most recent N demand entries, x(1), …, x(N), where N is the index of most recent demand. This can be modeled as:

$$a`(N+1) = [x(1) + \cdots + x(N)]/N$$

A difficulty in this model occurs for new parts where the number of months of history (nmh) is smaller than the parameter N. This issue can be solved by using the following temporary parameter N` (Thompoulos, 2015, p. 27):

$$N` = \min(N, nmh)$$

Table 1 illustrates the usage of temporary parameter N`. Notice how it after reaching the 10th period, the used parameter is equal to N.

Table 1: Calculating moving average by using temporary parameter N` (Thompoulos, 2015, p. 28).

t	x(t)	N`	a`(N+1)
1	10	1	10.00
2	12	2	11.00
3	6	3	9.33
4	9	4	9.25
5	4	5	8.20
6	10	6	8.50
7	6	7	8.14
8	7	8	8.00
9	9	9	8.11
10	8	10	8.10
11	13	10	8.40
12	11	10	8.30

3.7.3 Simple exponential smoothing (SES)

Simple exponential smoothing (SES) is considered as a reasonable performing method for forecasting intermittent demand and is also widely used for that purpose. (Armstrong, 2001, p. 200; Syntetos et al., 2015, p. 1747). SES forecasts based on a moving average of past values, the smoothed value at the present time being used as the forecast of the next value (Chatfield, 2005, p. 21). Exponential smoothing using Brown's (1962) formulation, presented by Armstrong (2001, p. 228):

$$\bar{Y}(t) = \alpha Y(t) + (1 - \alpha)\bar{Y}(t - 1)$$

Here Y(t) denotes the latest value of the series at time *t* and $\bar{Y}(t)$ is the smoothed average of the series. The α [0,1] explains how much weight to place on the most recent data, meaning the

lower the α, the smaller the weight. For example, an α of 0.3 means that 30 percent of the forecasted value comes from the latest smoothed average and the remaining 70 percent comes from the previous smoothed average. This means that the drop off of the weights of the periods is geometrical as the latest period is weighted by α, data from the period by α(1- α), the data from two periods ago by $\alpha(1-\alpha)^2$ and so forth. (Armstrong, 2001, p. 228) To be noted, the parameter α should be within bounds of [0;1] and a typical value for α would be 0.1 (Thompoulos, 2015, p. 32). Figure 11 shows an exponentially smoothed strike data with α = 0.4.

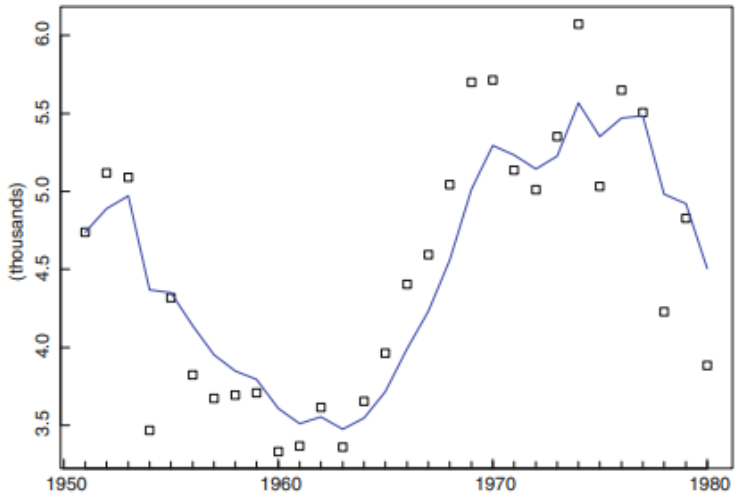

Figure 11: Exponentially smoothed strike data with α = 0.4 (Chatfield, 2005, p. 24).

One of the earliest studies supports the principle to weight recent data more heavily. A study analyzed monthly sales forecasts for 23 sewing machine products in 5 countries using seven and a half years of data in which the forecast horizon varied from 1 to 6 months. For a six-month horizon, the moving average and exponential smoothing methods were comparable in accuracy, but as the forecast horizon was shortened, the exponential smoothing became slightly more accurate.

Other studies have also supported these findings as for example in 68 monthly series the Median absolute percentage error averaged over forecast horizons of 1 to 18 months, was 13,4% for the untrended moving average and 9% for the untrended exponential smoothing. SES was also

more accurate based on all time horizons, but the improvements of it compared to the moving average were greater for short-term horizons. (Armstrong, 2001, p. 229)

However, it has been found that with more random error in data, moving averages outperform SES. This is consistent with the fact that recent errors can transmit shocks to an exponentially smoothed forecast. (Armstrong, 2001, p. 229)

3.7.4 Croston's method

In 1972, J.D. Croston suggested that traditional forecasting methods such as MA and SES may not be efficient enough for intermittent demand forecasting. He demonstrated that these methods could lead to sub-optimal stocking decisions and proposed and alternative forecasting procedure, nowadays called Croston's method. (Teunter & Duncan, 2008, p. 321)

In addition to SES, Croston's method is widely used to forecast intermittent demand in industry and it is incorporated in best-selling forecasting (and stock control) software packages (Syntetos et al., 2005, p. 498). While SES forecasts the mean level of demand for both zero and non-zero demand periods, Croston's method makes separate forecasts of the mean level of non-zero demand and the mean inter-arrival time (time between demand occurrences). It assumes that the distribution of non-zero demand sizes is normal, the distribution of inter-arrival times is geometric and that demand sizes and inter-arrival times are mutually independent. (Syntetos & Boylan, 2001, p. 459)

When denoting the observed value of non-zero demand as D(t) and the inter-arrival time of transactions as Q(t), the smoothed estimates are denoted by Z(t) and P(t) respectively:

$$Z(t) = \alpha D(t) + (1 - \alpha)Z(t - 1)$$

and

$$P(t) = \beta Q(t) + (1 - \beta)P(t - 1)$$

It is assumed that the value of the smoothing parameters α and β is the same in both equations, resulting into the estimate of demand per unit time i.e the forecast for next period U(t) being:

$$U(t) = Z(t)/P(t)$$

Moreover, in addition to presenting the formula, Syntetos & Boylan (2001, p. 464) suggest that in Croston's method it is recommended to only use low values for the smoothing parameter α. In all of their simulation runs, the model becomes pronouncedly biased for α values above 0.15. If there is no demand in a period, *Z(t)* and *P(t)* are unchanged. Now knowing this, the forecasted demand produced by Croston's method for the next period can be summarized by the following way:

$$\text{If D(t)} > 0 \text{ then} \begin{cases} Z(t) = \alpha D(t) + (1-\alpha)Z(t-1) \\ P(t) = \beta Q(t) + (1-\beta)P(t-1) \\ U(t) = Z(t)/P(t) \end{cases}$$

$$\text{If D(t)} = 0 \text{ then} \begin{cases} Z(t) = Z(t-1) \\ P(t) = P(t-1) \\ U(t) = U(t-1) \end{cases}$$

Note that when demand occurs on every period, Croston's method gives the same forecasts as normal SES, meaning it can be used for both intermittent and non-intermittent demands. (Syntetos et al., 2015, p. 1747-1748)

On a study where Croston's method was tested against exponential smoothing on an actual and artificial dataset, the findings showed that Croston's method produced substantially more accurate forecasts of demand per period than exponential smoothing. (Armstrong, 2001, p. 223) This comparison was also revised in 1996 and it was found, by using simulation analysis, that when the average inter-demand interval is greater than 1.25 forecast revision periods, the Croston's method performs better than exponential smoothing (Gutierrez, 2008, p. 410; Xu & Wang, 2012, p. 1469).

3.7.5 Syntetos-Boylan approximation (SBA)

Croston's concept for forecasting future demand estimates on intermittent demand has been claimed to have great value for corporations that deal with intermittent demand. However, in 2001, Syntetos & Boylan (2001, pp. 459-464) showed that despite the theoretical superiority of the method, empirical evidence suggests limited gains in performance when compared to simpler forecasting techniques and some evidence even suggests losses in performance. They found that Croston's separate estimations of the demand size and the inter-demand interval are indeed correct, but if fused as a ratio they do not produce accurate estimates of demand per time period. Moreover, Croston's method was shown to over-forecast the *U(t)*.

Syntetos & Boylan (2001, pp. 460-465) presented their improved model of Croston's method, called Revised Croston's method and later Syntetos & Boylan (2005, p. 304) developed a further method referred as the Syntetos-Boylan approximation. The method was a modified version of Croston's equation for *U(t)*, which was said to be approximately unbiased:

$$U(t) = \left(1 - \frac{\beta}{2}\right)\left(\frac{Z(t)}{P(t)}\right)$$

To tackle the issue on using either Croston's method or SBA method, Syntetos, Boylan and Croston together studied the Mean Squared Error (MSE) produced by the methods on different types of intermittent demand patterns. What they found out was that the different methods perform better in different intermittent demand patterns based on MSE illustrated by cutoff values in figure 12.

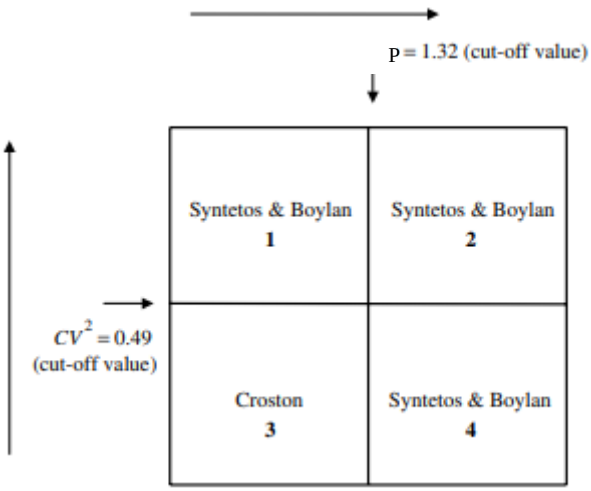

Figure 12: Cutoff values for Croston's and SBA method (Syntetos et al., 2005, p. 499).

Syntetos et al. (2005, p. 499) determined that for the estimated inter-arrival time of transactions $P > 1.32$, the SBA method produces more accurate results, and when $1 < P \leq 1.32$ and $CV^2 \leq 0.48$, the Croston's method performs better. These above results were valid for $\alpha = 0.15$ and approximately true for other realistic α values. Table 2 illustrates the different cutoff values based on smoothing value α.

Table 2: MSE Croston's method - MSE SBA method (Syntetos et al., 2005, p. 499).

α Smoothing constant value	*P -Cutoff value*	*CV^2 cutoff value*
0.05	1.32	0.49
0.10	1.32	0.49
0.15	1.32	0.48
0.20	1.31	0.47

It can also be expected that the SBA model performs better when $(P - 1)/P$ gets closer to one meaning as the $1/P$ of positive demand in a period gets smaller. Teunter & Sani (2006, p. 4) illustrated this as the Croston's original method had smaller bias when $1/P$ was large (few demands were zero), and the SBA model had a smaller bias when $1/P$ was small (many demands were zero).

3.7.6 Teunter, Syntetos & Babai model (TSB)

Even though widely used and proven to work for intermittent demand forecasting, the traditional Croston's method and even its improved method SBA and its variations are fairly old. Moreover, even though both of these methods take into account the separate inputs of demand size and demand interval for intermittent demand, neither of them updates the forecast after periods with zero demand. This results in so that the forecasts become outdated after many periods with zero demand and provide inaccurate estimates in terms of the risk of obsolescence. This can be an important factor when dealing with highly slow-moving intermittent SKUs. (Teunter et al., 2011, p. 606)

In order to solve the issue of these traditional models, Teunter et al. (2011, p. 608) proposed a new method modified from the SBA method, which is coined in the thesis as Teunter, Syntetos and Babai (TSB) model. The model calculates the forecasted non-zero demand *P(t)* in the same way as in Croston's and SBA method, but it use a separate SES estimate of the demand probability instead of the traditional interval of non-zero demand as in Croston's and SBA method. When using the same notations as in Croston's and SBA method and denoting β as a smoothing constant for the probability of a demand occurrence *p'*, the TSB model can be summarized the following way:

$$\text{If D(t)} > 0 \begin{cases} Z(t) = \alpha D(t) + (1-\alpha)Z(t-1) \\ p'(t) = \beta + (1-\beta)p'(t-1) \\ U(t) = Z(t)p'(t) \end{cases}$$

$$\text{If D(t)} = 0 \begin{cases} Z(t) = Z(t-1) \\ p'(t) = (1-\beta)p'(t) \\ U(t) = Z(t)p'(t) \end{cases}$$

The new method uses two smoothing parameters: α and β and it is identical to the earlier discussed naïve method when $\alpha = \beta = 1$. As Croston's and SBA method only use the α smoothing parameter which is used only when updating the forecast for a period with preceding non-zero demand period, resulting in so that the methods automatically use longer demand history (in time periods) for slow moving SKUs when $\alpha = \beta$. Therefore it was proposed and

numerically confirmed by Teunter et al. (2011, pp. 608-611) that the parameters should be set in such a way that $\alpha > \beta$.

Moreover, the choosing of the smoothing parameters depends on various factors, one of which is the stationary of the demand process. If the demand is suspected to be largely non-stationary, the method adapts quickly to these changes when sufficiently large smoothing parameters are used and vice versa. A good way of setting these parameters is by empirical optimization based on the demand history. This however might be hard when dealing with a lack of data on an intermittent demand. Another option is to use the earlier discussed suitable smoothing parameter values for intermittent forecasting methods. It is important to keep in mind though that for the TSB method, the smoothing parameter values should depend on the period length. This means that for a demand series with monthly periods a 0.05 to 0.2 parameter range might be suitable, but for a series with daily periods much smaller parameter values are needed. (Teunter et al., 2011, pp. 608-611)

It was found by Teunter et al. (2011, pp. 609-614) that TSB performed generally well compared to SES, Croston and SBA methods in terms of both bias and variance. However, it is utmost important to use the right smoothing parameters to achieve optimal results. On the other hand, this is also one of the strengths of the model as it can be tuned depending on the demand series. All in all, this topic certainly needs more research and by doing so the right parameter values can be found for different datasets and it is also generally advocated to use situation-dependent smoothing parameters. It is also worth noting that although the TSB model is generally unbiased when all points in time are considered, the method suffers from statistical bias for the estimates isolated in demand occurring periods only. This bias reminds characteristics of SES when applied directly on intermittent data. (Teunter et al., 2011, p. 609-614)

Even though TSB model has been shown to provide great theoretical performance for SKUs with linear and sudden obsolescence, as years have passed by the model has also been shown to be empirically outperformed by the SBA method in some scenarios with obsolescence. This was one of the reasons why in a recent study, Babai et al. (2019, pp. 30-41) investigated both the TSB and the SBA models and how they could be improved.

They noted that the SBA model had flaws when dealing with obsolescence issues of a dataset and that the TSB model was meant to fix this issue. Knowing this, it would make sense to not use TSB model whenever forecasting a period with preceding zero-demand as the SBA model was not flawed in that sense. This resulted in such a model which would generally work like a SBA model, but if the actual demand interval became higher than the most recent estimated demand interval (might occur with obsolescence), the demand interval was updated with the probability of occurrence like in the TSB model. Hence, this model uses the perks of TSB by capturing the risk of obsolescence and the perks of SBA by using the same estimator as a standard.

This method is referred as the modified SBA and using the previous notations, the resulting model can be summarized by the following way:

$$\text{If D(t)} > 0 \begin{cases} Z(t) = \alpha D(t) + (1-\alpha)Z(t-1) \\ P(t) = \beta Q(t) + (1-\beta)P(t-1) \end{cases}$$

$$\text{Else} \begin{cases} P(t) = \begin{cases} \beta Q(t) + (1-\beta)P(t-1) & If\ Q(t) > P(t-1) \\ P(t-1) & If\ Q(t) \leq P(t-1) \end{cases} \\ Z(t) = Z(t-1) \end{cases}$$

And

$$U(t) = \left(1 - \frac{\beta}{2}\right)\left(\frac{Z(t)}{P(t)}\right)$$

Now it should be noted that Babai et al. (2019, p. 32) propose the usage of two smoothing parameters in the demand interval and demand size forecasts derived from Croston's method. They explain that even though Croston originally suggested using only one smoothing parameter for both, they wanted to test the performance of this method under better conditions.

To test the performance of this model Babai et al. (2019, p. 32-33) generated demand dataset for a 10-year demand history with monthly periods by assuming the demand sizes to be stationary and generating them from a logarithmic distribution and the demand occurrences were generated from the Bernoulli distribution. These both methods were used as they have a

considerable empirical evidence to back up them as suitable methods for this. The generated demand sizes had a mean and standard deviation of $\mu = 1$, $\sigma = 0$ and $\mu = 4$, $\sigma = 5$, respectively and the resulting demand occurrences were a linearly decreasing demand and a demand with a sudden obsolescence.

The model was tested against Croston's, SBA, TSB and SES models by observing the resulting MSE and bias and the numerical results showed the modified SBA to perform the best, followed by TSB. It was shown that the modified SBA method can lead to a significant bias reduction that can reach 80% and a MSE reduction up to 2% when compared to TSB model. Babai et al. (2019, p. 33-37)

3.7.7 Neural networks

It has been discussed that conventional statistical time series methods are prone to misjudging the functional form relating the dependent and independent attributes. Methods like these might also fail in making the required data alterations and observed outliers in the data can lead to biased estimates of the model's parameters. In addition, they might not be able to capture non-linear patterns and they also must be recalibrated on all previous data.

To overcome these limitations of the traditional models, ANN (artificial neural network), or more commonly referred as just NN (neural network), based models have been considered as they can provide good approximations to variety of functional relationships. Even as NN is not considered as a traditional method for this, successful applications have been well documented as early as in the 1980s. Flexibility and nonlinearity are the two most important aspects of NN modeling. (Gutierrez et al., 2008, p. 410) It has also been argued that NNs do not require experts to prescribe rigid model structures and thus there is no need for expert input (Kourentzes, 2013, p. 199)

However, NN models are not always suited for many applications when used for prediction in extrapolation unless these models are designed to fit the application phenomenon as the gradient search technique may find a local minimum instead of the global minimum in the least mean

squared cost function. The NN approach also requires somewhat a large amount of data compared to the traditional methods. (Gutierrez et al., 2008, p. 410)

Gutierrez et al. (2008, p. 411) used the most widely used NN method, a multi-layered perceptron (MLP) trained by a back-propagation (BP) algorithm. Detailed explanation of how MLP network work for time series forecasting and BP training algorithm can be found in Appendix C. They followed the guidelines proposed by a recent study of architecture selection of MLP and used three layers of MLP: one for input variables, one hidden unit layer and one output layer. The study also suggested that one should start with the minimum number of hidden units required to approximate the target function for a minimal architecture leading to the usage of three nodes (n= 3) in the hidden layer. Also, a learning rate value of 0.1 and a momentum factor of 0.9 (see Appendix C) were used in line with past research.

One output node was used as an output layer and all the input nodes were fully connected to all the hidden nodes. These hidden input nodes were then connected to the said output node. The input nodes represented two variables: *the demand at the end of the immediately preceding period* and *the number of periods separating the last two non-zero demand transactions as of the end of the immediately preceding period*. The architecture of the NN is illustrated in figure 13.

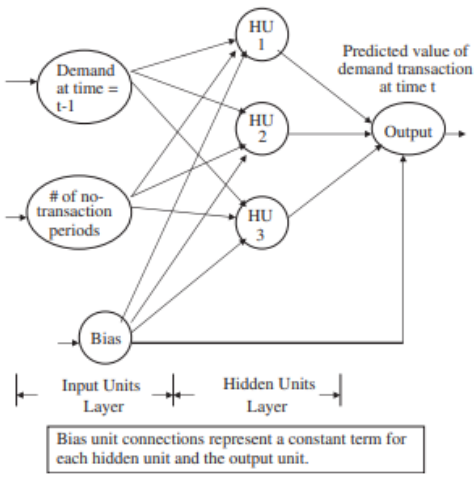

Figure 13: NN architecture used by Gutierrez et al. (2008, p. 412).

The data used by Gutierrez et al. (2008, pp. 412-418) was 24 time series of industrial data consisting of 967 daily demand observations for SKUs carried by an electronics distributor operating in Monterrey, Mexico. This is generally a lot of data points for a NN model to process compared to a normal intermittent dataset for a monthly demand periods with a few years of data. The datasets were classified as lumpy demand patterns as the squared coefficient of variation (CV^2) was greater than 0.49 and the average inter-demand interval (ADI) was greater than 1.32. Furthermore, the mean nonzero demand transaction sizes in the datasets were roughly between 700-1500, meaning that they cannot be reflected to for example earlier described SKU based demand for single service point at Ramirent. The detailed information about the datasets can be found in appendices D and E.

Gutierrez et al. (2008, pp. 412-418) used an alternative method of MAPE, RGRMSE and PBt as an error measurement tools in their research. RGRMSE was argued by Syntetos & Boylan (2005, p. 313) to be a very well-behaved accuracy measure for intermittent demand and PBt was also argued to be a consistent method for this. As discussed earlier, intermittent demand involves periods with zero demand, leading to the failure of traditional MAPE as an error measurement. The alternative specification of MAPE was the following:

$$MAPE = \frac{\sum_{t=1}^{n} |E_t|}{\sum_{t=1}^{n} |D_t|}$$

By using these error metrics, Gutierrez et al. (2008, pp. 412-418) compared the performance of Croston's method, SBA, SES and NN models when using a low smoothing constant α of 0.05, 0.1, 0.15 and 0.2 as suggested by literature. As a result, they found out that the NN models generally outperform the three traditional demand forecasting methods except for demand series 24 and apart from series 22, 23 and 24 when α = 0.05 based on MAPE.

As SBA and NN methods showed superiority among the chosen models, Gutierrez et al. (2008, p. 416) conducted t-tests of the paired differences between absolute errors for NN vs. SBA when α = 0.05 and found out that the difference in performance was significant for all the 24 time series. The paired differences favored NN model performance in 20 of 24 time series.

Similar results were found with the usage of RGRMSE and PBt as NN method showed dominance in the series 1-22 on both cases.

Later, Kourentzes (2013, pp. 198-206) revised the usage of NNs for intermittent demand forecasting and used the study conducted by Gutierrez et al. (2008, pp. 409-420) as one of the examples on research conducted previously on this topic. He examined the potential of Gutierrez et al. (2008, pp. 409-420) NN model to overfit and instead of using a traditional cost function of mean squared error loss, an improved cost function F was used:

$$F = \gamma \frac{\sum_{i=1}^{N} e(i)^2}{N} + (1-\gamma) \frac{\sum_{j=1}^{n} (w(j)^2)}{n}$$

The first part of the cost function is the standard mean squared error loss of *N* one-step ahead *e(i)* errors and the second part is keeping the weights of the network *w(j)* small by penalizing large weights, effectively making the network response smoother and reducing the chances of overfitting. The γ is a performance ratio that controls the size of regularization and it was determined by the data automatically. The advantage of regularization was that a validation sample was no longer needed to prevent overfitting to the training data, therefore the implementation of NNs for intermittent demand problems became feasible for even small samples.

Kourentzes (2013, pp. 198-206) formed two different NN models, one which gave two outputs: the forecasted demand interval and the forecasted non-zero demand, called NN-Dual and another which yielded a demand rate as an output, referred as NN-Rate. More details about these models can be found in appendices F and G.

These models were then compared to other forecasting methods such as Naïve, MA, SES and variations of Croston's method with the same smoothing values as used by Gutierrez et al. (2008, pp. 409-420). Kourentzes (2013, pp. 198-206) also used the network proposed by Gutierrez et al. (2008, pp. 409-420) as a benchmark, referred as NN-GSM. However, the NN-GMS had one significant modification: instead of using the standard BP with momentum,

regularized loss and the Levenberg-Marquardt training algorithm were used to allow training the networks in small samples. All of the NNs were regularized and a γ of 0.9 was used.

To measure the accuracy of different methods, Kourentzes (2013, pp. 198-206) used ME and MAE. A service level estimator was also used with a lead time of 3 days, forcing the forecast horizon to be three periods as well. Target service levels were set for 0.8, 0.9, 0.95 and 0.99 in order to see how different models would behave. In each period, the realized demand for e ach item was subtracted from the holding stock H. If the stock fell below order-up-to level S, then an order an order S-H was placed with the chosen lead time. If the order couldn't be serviced, an out-of-stock even was measured.

The dataset used by Kourentzes (2013, pp. 198-206) was the same as used by Syntetos & Boylan (2005, pp. 303-314), which consisted of the intermittent monthly demand histories of automotive spare parts over a 2-year period of 3000 SKUs. The average inter-demand interval of this data varied from 1.04 to 2 months and the average demand per unit time period from 0.5 to 120 units, which was significantly lower compared to the dataset used by Gutierrez et al. (2008, pp. 409-420). (Syntetos & Boylan, 2005, p. 305).

Based on this dataset Kourentzes (2013, pp. 198-206) formed new monthly intermittent time series, each of them having 236 observations, out of which 36 observations (3 years of history), were used as training data, 100 were used as testing data to estimate the performance of the models and 100 were used as burn-in period for a simulation. This burn-in period stared with a full stock and ended in a realistic scenario to guarantee a reliable starting point for the simulation to get accurate service level results.

The findings of Kourentzes (2013, pp. 203-205) suggest that NNs are good contesters for intermittent demand forecasting problems. Interestingly, simple models such as Naïve model, performed well when measured by service level by overstocking, but the NN-Dual model was found to be the best performing model, reaching substantially higher service levels with minimal small increase in the holding volumes. However, contrary on findings of Gutierrez et al. (2008, pp. 409-420), the NN models showed poor forecasting performance on accuracy metrics used. This might be the causation of the usage of short monthly intermittent time series

instead of daily long lumpy demand series, as in the study conducted by Gutierrez et al. (2008, pp. 409-420). Furthermore, Kourentzes (2013, p. 204-205) suggests that the accuracy metrics provide misleading findings and in fact are erroneous for intermittent demand data.

After the literature review about the intermittent demand forecasting methods, it is chosen that the focus on the thesis will be on all of the above models excluding the neural network ones. This is simply because the NN models did not seem to outshine the traditional models by a large scale and they’re essentially based on the same intermittency and demand lumpiness idea as the other methods. Also given the fact that there is a large expected amount of demand series to be dealt with in the tool and the iterative nature of the NN models, which may require a lot time and computational power, gives less incentive of actually using NN models in the tool.

4 RENTAL TIMES ESTIMATION

As mentioned earlier, the renting process of Ramirent is heavily connected to an estimated rental time on a rental set by the customer and the employee. As it is just an estimate, it means that sometimes the realized rental time is shorter than estimated or longer than estimated. A causation of this is that in order to build a system where the stock levels can be optimized based on some criteria, a model that attempts to predict the rental times needs to be developed. For this, statistical methods and their usage for this problem are investigated in this chapter.

4.1 Framework for rental times estimation method

As the rental time for an equipment can vary based on many factors and is an unpredictable event as the customer does not have to provide an exact date or time for the rental, the outcome for this process can be seen as somewhat of a random walk. The concept of random walk can be explained with a case of a random walk on Z. Let $(Z_n)_{n\in N}$ be an infinite collection of independent and identically distributed variables taking values in [-1,1] with $P(Z_n = 1) = p$, and X_0 a random variable taking values in Z, independent of the Z_n's. For $n \geq 1$ it can be put that:

$$X_n = X_{n-1} + Z_n$$

Or more explicitly,

$$X_n := X_0 + Z_1 + Z_2 + \cdots + Z_n$$

Hence the process $(X_n)_{n\in N}$ is referred as a random walk on parameter p on Z. The expected value of X_n is by linearly:

$$E(X_n) = E(X_0) + n(2p - 1)$$

Based on this it can be said that as n increases the $E(X_n)$ approaches infinity if $p > 0.5$ and $E(X_n)$ approaches minus infinity if $p < 0.5$. With $X_0 = 0$, this leads to that 0 will not be revisited much unless $p = 0.5$, in which case $E(X_n) = 0$ meaning the random walk is symmetric and the process is recurrent. (Collet, 2010, pp. 103-104)

Even though multiple factors such as current season, customer type, rental location and estimated rental time can affect the realized rental time of an item, the data which would be deemed recent enough in order to be used to predict the rental times by taking into account all these factors wouldn't necessary have enough volume when dealing with the item based rental data for a single service location. In addition to this, the used forecasting method will most likely not classify the customer type for the forecasted demand, giving less incentive on using it as a variable for rental times prediction.

When it comes to choosing the right variables for this, it is an empirically perceived phenomenon at Ramirent that the rental equipment rental times can be explained at some level based on the service location. This phenomenon can be clearly seen for example on the standard of location based rental times for an item group 611, which is a common certain type of an industrial vacuum cleaner. On locations with 50 or more rentals during 2018-2019, the product had a mean rental time of roughly 19 days and a standard deviation of 16 days, meaning the location certainly explains some of the realized rental time of an item.

It is also known that a part of Ramirents future strategy includes the reinforcing of the usage of the estimated rental time. This means that it should be considered to be used as one of the explaining variables when it comes to rental times estimation. Also, the fact that it essentially (if used correctly) should best capture the effects of seasonality and time the item is needed by the customer and thus lead to correct rental time prediction. Of course this is considering that the customer actually knows how long they are going to need the product, has no intent on giving a false estimation (which can happen for example if the daily price correlates negatively with the time estimate) and that no other factors come in play which would decrease or increase the rental time of the item after the estimation.

Based on the points presented, it is decided that rental times estimation models which use the location and the estimated rental time should be examined and considered for the usage of this tool.

4.2 Data analysis on rental times

As discussed before, the actual rental time for a rental is not necessarily the same as the estimated time for the rental at the time of the renting transaction. Because of this it is wise to take a look on how the actual rental time compares to the estimated rental time on the data on hand. Figure 14 illustrates the minimum, maximum and mean rental times as a function of estimated rental time on the transactional data for 2018-2019 of item/location combinations which will be included in the tool.

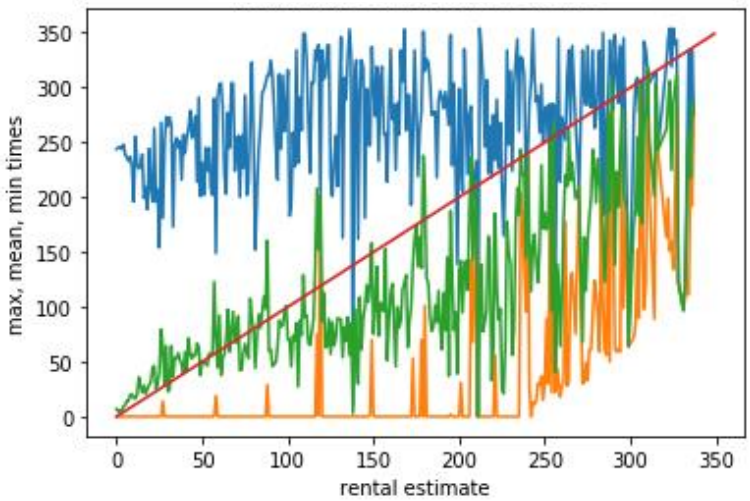

Figure 14: Max (blue), mean (green) and minimum (orange) rental times as a function of rental estimate on item/location combinations for the tool in transactional rental data for 2018-2019.

From the figure it can be seen that the green mean rental time of items somewhat follows the red line which is the indicator of having the same rental time as the rental estimation. However, it is fairly noisy and begins to get even noisier the bigger the rental estimate gets. This is of course natural as it most likely is harder to estimate the exact time the rental item is needed the longer it is estimated to be needed. The maximum and minimum times do not really explain much more than that there seems to be some kind of a distribution existing between the rental time and the rental estimate. However, it can be seen that on some rental estimate times there has not been a rental that has had a rental time of 0, meaning the item was returned the same day it was rented out, which is why the occasional spikes occur before actually arriving to an eventual breaking point where the item is of such nature that it most likely is always needed for more than just one day.

After observing the relationship between rental times and estimates, volume of different rental times, estimates and the errors in the data were observed. Figure 15 shows a collection of different plots which include the rental amounts based on rental estimate, rental times, both and rental time errors.

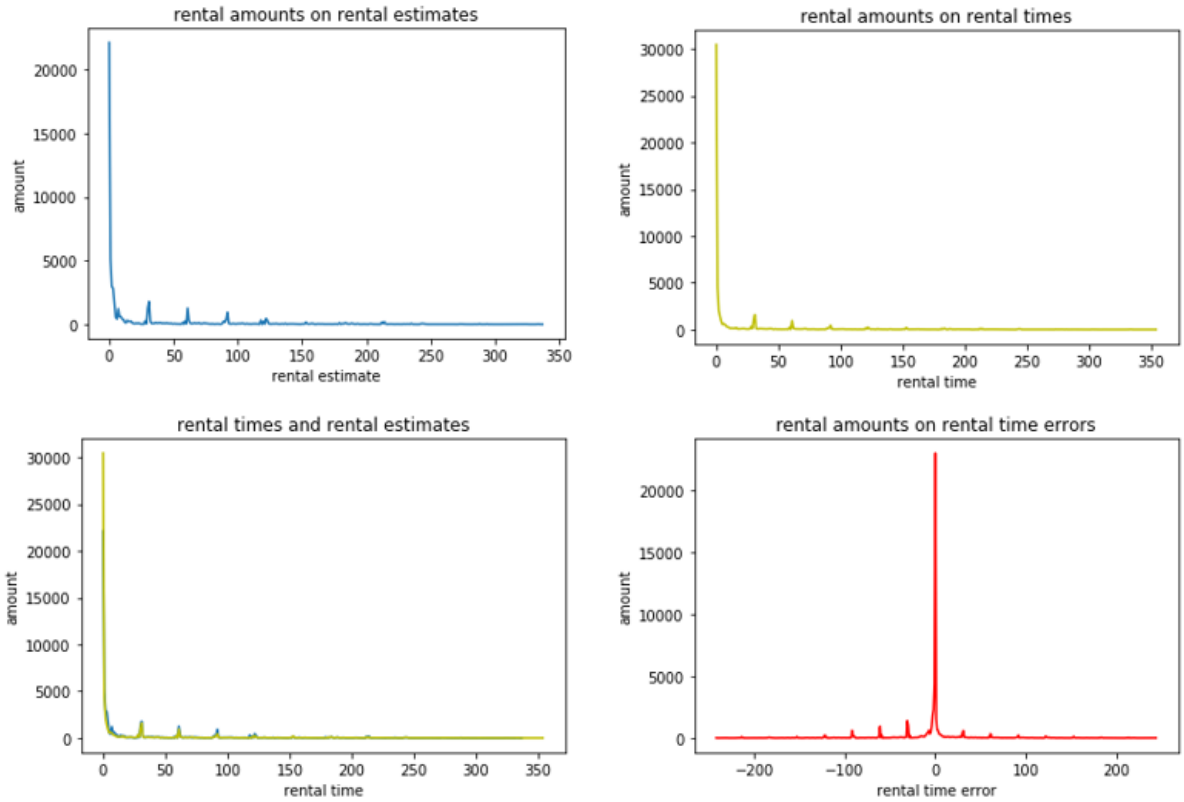

Figure 15: Plots on amounts in data based on rental estimate, rental times, both and rental time errors.

From the plots it can be seen that typically the rents are estimated to be short 0-1 days and they also seem to be largely returned within those days. However, there are still some spikes for certain times on the estimation and realized rental time and the estimations are set for longer time than the actual rental time. This can be seen in the plot on rental time amounts, which illustrates that these spikes have a negative rental time error, meaning they were rented out shorter than estimated.

4.2.1 CDF-method

As there seems to be no reliable way in determining the actual rental time solely based on the rental estimate, a statistical approach to this problem is proposed. In order to do this, the rental times are converted into percentages of the estimated rental time so that the noise between individual rental estimates is avoided. The relativity is calculated with a simple formula of

$$Relative\ rental\ time = \frac{Rental\ time}{Estimated\ rental\ time}$$

and this relativity can be seen in figure 16.

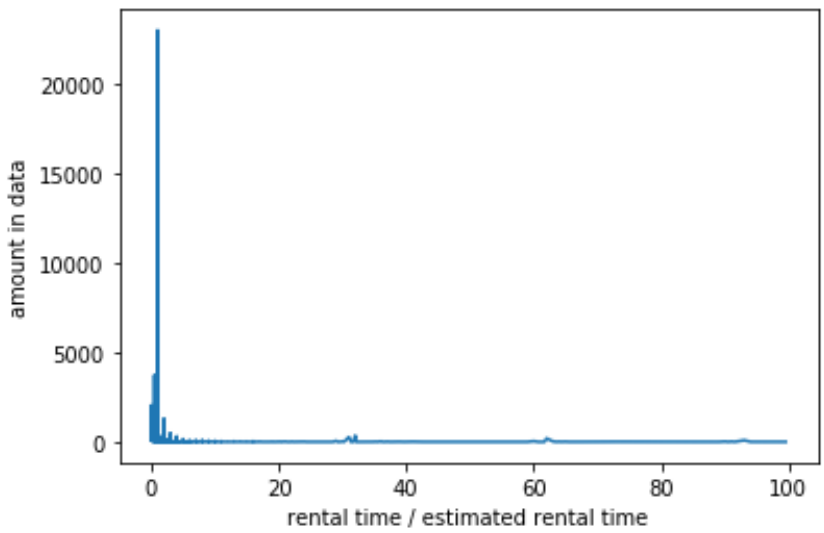

Figure 16: Relative rental time amounts in transactional rental data for 2018-2019.

Now it can be seen that the relative rental times are heavily concentrated on small values and figure 17 captures the emerging pattern:

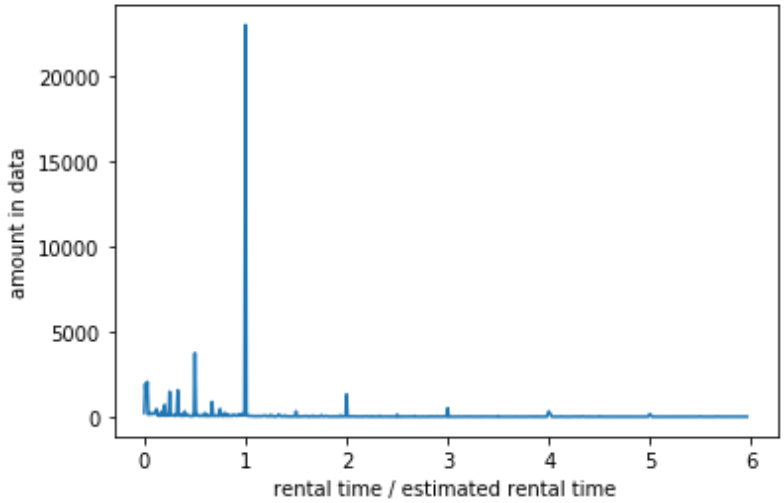

Figure 17: Zoomed relative rental times amounts in transactional rental data for 2018-2019.

Now it can be seen even clearer the pattern that emerges from the relative rental times: the items are returned largely on the day they're estimated to be returned and the amount of single day estimate rentals make it so that the pattern has ticks on integer numbers.

After doing this, in order to determine the expected return of a product per a rental day, a cumulative distribution function (CDF) can be formed of a chosen set of transactions to describe the probability of an item returning based on the time it has been rented out relative to its estimated rental amount. Figure 18 illustrates 2 plots one containing the rental CDF for year 2018 transactions and the other containing this same for the year 2019 transactions. Both of these are already zoomed in as it was shown above that the pattern is largely focused on small integer numbers.

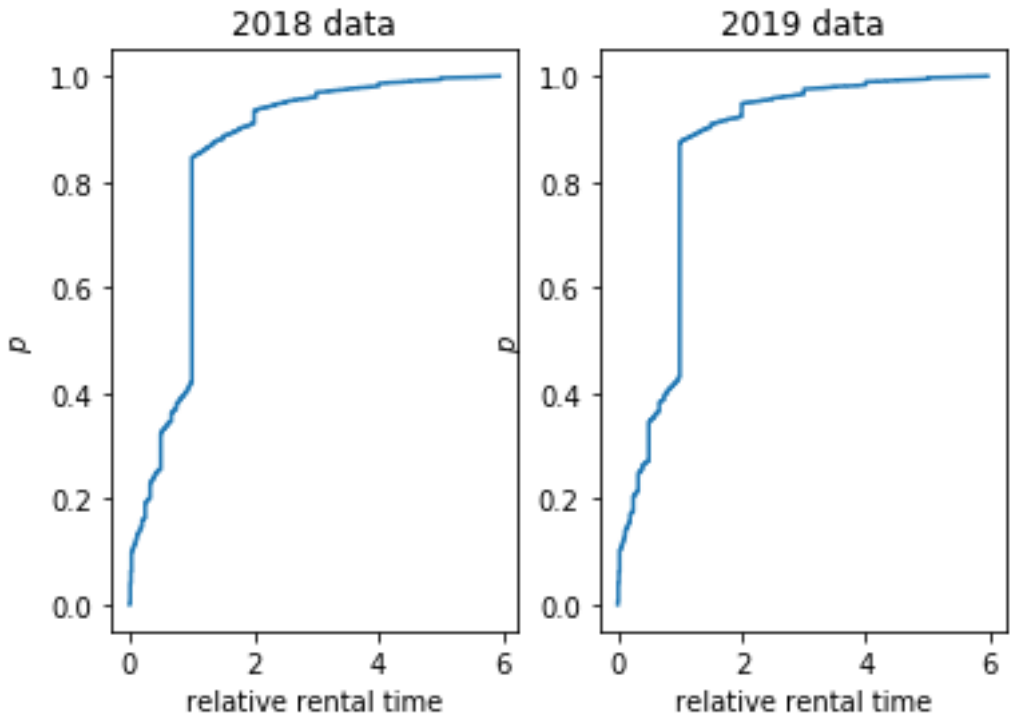

Figure 18: Zoomed in CDF of relative rental times of 2018 and 2019 rental transactions.

From the plots it can be seen that both of these years have somewhat same kind of a pattern to them and noticeably the relative point of 1 where the rental time equals the rental estimate gives a huge increase to the probability of an item returning, which was also to be seen from the earlier figure. Using this approach for the location and item-based CDF's will be one of the rental times estimation methods for the tool. The method will work in such a way that it compares the relative rental time of the current transaction to the CDF of similar transactions and the probability for the item to return is calculated as:

$$p(ct) = \frac{n(ctCDF)}{n(CDF)}$$

Where *n(ctCDF)* is the amount of transactions contained in the CDF which have equal or greater value of relative rental time to the current transaction and *n(CDF)* is the total amount of transactions contained in the CDF.

4.2.2 Triangle “activation function” -method

Even though the CDF-method for rental times estimation should reflect in what time compared to the estimated time the products are returned, the nature of it bounds it to almost never predict the rental time fully correct. Knowing this and the fact that many rentals are returned exactly when they are estimated, a comparative method which would tend to predict the exact return date is sought for.

For this, a so-called triangle “activation function” -method is proposed. This method calculates the probability of the return of a rental based on the rental estimated time and the rental time elapsed for the rental. The probability is calculated by the formula of:

$$p = \min(1, \frac{\frac{Rental\ time\ elapsed}{Estimated\ rental\ time} * \frac{2 * Rental\ time\ elapsed}{Estimated\ rental\ time}}{2})$$

This also gives the method its name, as the probability is calculated as an area of the triangle formed from the rental estimated time and the rental time elapsed and the function itself resembles the type of activation functions seen in neural networks. Contrary to the CDF-method, this method should be able to perform especially well when there are a lot of rentals with a zero error in the rental time compared to the estimated time and if it is crucial for the method to be able to predict exact rental times. However, it also assumes the probability of a rental return being 100% after the estimated return date is passed, meaning it ultimately overestimates the stock a location has to meet the future demand which might lead to missed rental transactions.

5 OPTIMIZATION METHODS

After demand and rental times forecasting the tool will optimize the stock of each service point for each product group based on the forecasts made in previous modules. Optimization can be thought as to finding the best way to perform in a situation that requires decision-making or figuring the best-case scenario among different choices. More precisely, optimization is concerned with maximization and minimization with the purpose of finding the values of variables that achieve this for the value of a given function (Cottle & Thapa, 2017, p. xxiii). As discussed earlier, this optimization will be based on the revenue generated by the product transfers made between the service locations in the service network provided for the tool and this optimization will be constrained by the available products within the network. These qualities of the optimization are referred as the objective function and the constraints of optimization.

The concept of optimization has been lingering around in mathematics for many centuries: in early as the seventeenth century, mathematicians used to speak about calculating the maxima and minima. Leonard Euler asserted in 1744 that "*For since the fabric of the universe is most perfect, and is the work of a most wise Creator, nothing whatsoever takes place in the universe in which some relation of maximum and minimum does not appear. Wherefore there is absolutely no doubt that every effect in the universe can be explained as satisfactorily from final causes, by the aid of the method of maxima and minima, as it can from the effective causes themselves.*". (Aragón et al., 2019, pp. 1-2) The steady growth of computing power since the mid-twentieth century has strongly aided the development of optimization as it has enabled even more complex models to be solved. This has resulted in so that nowadays the formulated optimization model for the problem is typically solved by using specialized computational methods referred as algorithms. (Cottle & Thapa, 2017, p. xxviii).

This chapter discusses different optimization methods and their suitability or probability to be used in this model. Such methods include linear optimization, nonlinear optimization and Network models in optimization.

5.1 Linear and nonlinear optimization

Linear optimization, which can also be referred as Linear Programming, is named according to the nature of functions it uses for its objective and constraints. This means that all the functions used in the optimization model must be linear. (Cottle & Thapa, 2017, p. xxvi) This also means that it can be only applied to simple linear optimization problems, but is a fairly powerful tool to be used for problems of this type. These type problems can include many short-term facets of business such as supply and distribution planning, refinery planning, product blending or process control, which represent closed problems meaning that the single criteria of for example profit maximization may be adequate. (Garg & Tadj, 2017, p. 5)

As simple optimization is not the main subject of this thesis and linear optimization is even less so, this thesis will not go deeper into the subject. However, for those who are interested, appendices H and I further explain the concepts connected to linear optimization focusing on such subjects as the diet problem and strong duality theorem.

Contrary to linear optimization problem, where the objective function and constraints are all linear and there exists at least one linear inequality constraint, an optimization problem with continuous variables where one or more of these functions is not linear is called a nonlinear optimization problem. This can also be referred as nonlinear program (NLP) and if only the objective function of it is nonlinear and the constraints are linear, then it is referred as linearly constrained. Some normal features of NLP include:

- Having at least one nonlinear function
- One or more continuous variables
- Having inequality constraints, equality constraints or is unconstrained
- Properties such as continuity, differentiability, or convexity
- Occasionally complex optimality criteria
- Usually not finite, but convergent solution algorithms with connected rates of convergence e.g linear, superlinear, and quadratic.

When it comes to Convex functions, they can be very useful in optimization as their local minima is also global minima. These functions can either be univariate or multivariate and there exists different classes of convex functions such as strictly convex and the strongly convex functions. (Aragón et al., 2019, pp. 55)

The last two features in the list play a key role when it comes to explaining how nonlinear optimization problems are being solved. They are connected with the iterative nature of NLP algorithms, which iteratively attempt to find the optimal value for the objective function. Ideally, this results into feasible solution that yields a locally optimal objective function value. Due to the iterative nature of the solution method, it is usually of interest to know the rate of convergence and how much computation is involved in each iteration. (Cottle & Thapa, 2017, p. 233-235).

As the subject of nonlinear optimization is extremely broad and it is unlikely that the tool built in this thesis will see the need for the usage of NLP models, this chapter is not going to elaborate deeper on NLP models that arise in NLP field. However, it is worth mentioning that the nonlinear optimization problems, and thus the NLP models used can be divided into 3 categories: Unconstrained NLP, linear constrained NLP and nonlinear constrained NLP. Concepts and problems affiliated with these categories are for example the least-squares problem, entropy maximization problem, quadratic programming and optimization over an ellipsoid. (Cottle & Thapa, 2017, p. 237-265).

However, there has been applications where NLP models have been used for warehouse optimization. For example, Qin et al. (2013, pp. 1-6) used genetic algorithm in order to optimize a logistic warehouse problem on an automated warehouse system. The goal was to decrease the cost time of getting or putting goods on shelf and the distance of the same kind of goods. The implementation of NLP with genetic algorithm model successfully reduced both of these and as a conclusion the model was said to improve the efficiency of the storage.

5.2 Network models in optimization

There exist different types of network model notations, but in this thesis the basis of network model is defined as a network consisting of nodes and arcs between those nodes. There can also be other attributes within the network such as cost of an arc or different constraints connected to these arcs. These concepts are further explained in the following parts of the thesis when relevant by using the notations presented in their corresponding literatures.

Network design is a crucial and also frequently encountered class of optimization problems. These type of optimization problems can be found in different types of everyday practices in engineering and management including the service point network of Ramirent. When speaking about network models the usual terminology is that the different locations of network are referred as nodes and the pathways connecting these nodes are called arcs.

Network optimization problems may include such topics as determining the shortest or most reliable paths between nodes, maximal or compatible flows, coordinating projects and solving supply and demand problems. The level of these problems can vary a lot as some require only the simplest linear optimization solving whereas some might not even be solvable by modern algorithms within a reasonable time. Figure 19 illustrates the different network design models with the particularly useful models for this thesis marked with red, with further explanation of these provided in the following chapters.

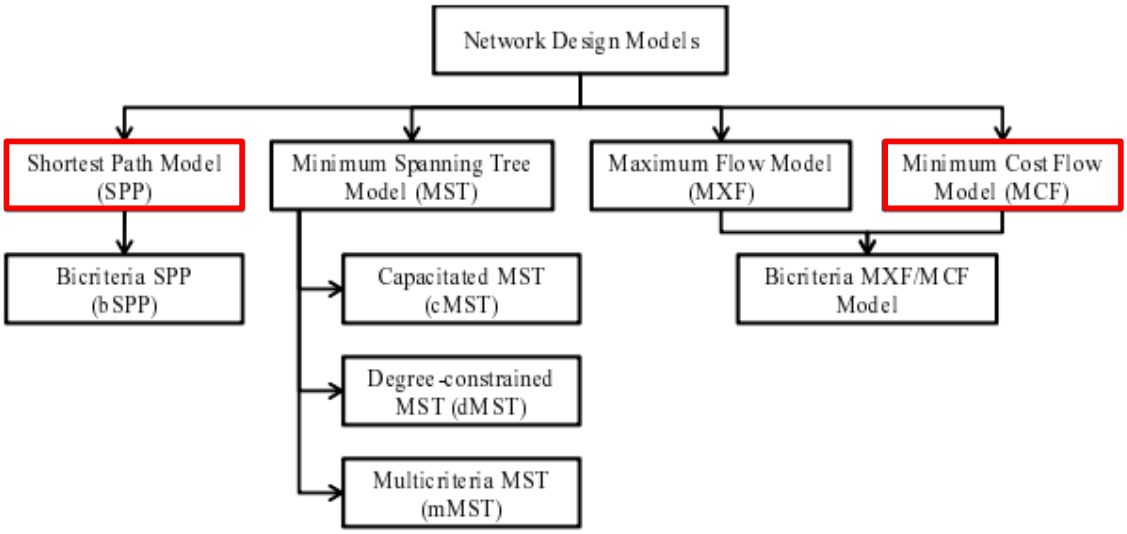

Figure 19: network design models (Gen et al., 2008, p. 50).

5.2.1 Shortest Path Model (SPP)

The shortest path model (SPP) answers to the question "What is the shortest path between two specified nodes in the network?" and it is the heart of network design optimization. This is because firstly it can be used on variety of different areas in order to compute the fastest or cheapest way to move something between network nodes, and secondly it captures many of the most core ingredients of network design problems providing a benchmark for studying more complex network models. SPP problems are usually relatively easy to solve, but the design and analysis of most effective algorithms for solving them involves a lot of work. (Gen et al., 2008, p. 49-51)

A simple SPP problem can be illustrated by using the network shown in figure 20.

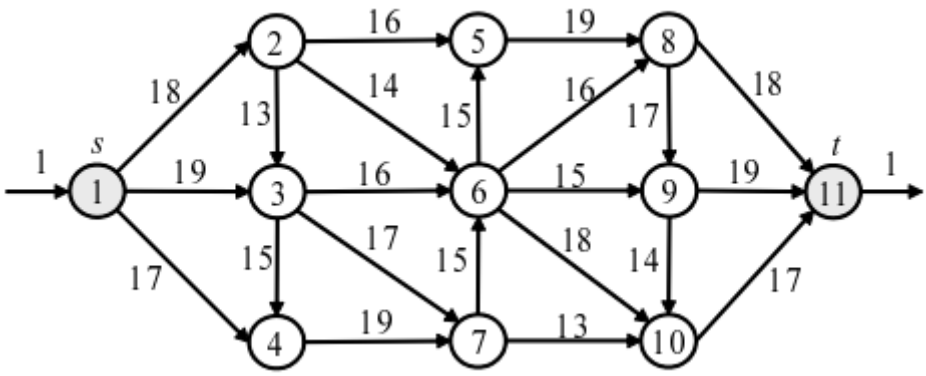

Figure 20: a simple network model (Gen et al., 2008, p. 58).

In the figure it can be seen how the network has nodes and arcs and how each arch between nodes has a cost associated with it. It can also be noted that in this network it is possible to only move to a certain direction between two nodes, meaning it is a directed network. In other networks this would not necessary be the case and there also might exist longer arcs for example an arc reaching from node 1 to node 10. To solve a SPP problem in a simple network like this, simply put, one must compare all the available arc paths from the starting node to the ending node and choose the way which has the lowest cost associated to it.

This network can also be turned into table for easier mathematical formulation for its SPP model. This conversion can be seen in table 3.

Table 3: Network model illustrated with a table (Gen et al., 2008, p. 57).

i	1	1	1	2	2	2	3	3	3	4	5	6	6	6	6	7	7	8	8	9	9	10
j	2	3	4	3	5	6	4	6	7	7	8	5	8	9	10	6	10	9	11	10	11	11
c_{ij}	18	19	17	13	16	14	15	16	17	19	19	15	16	15	18	15	13	17	18	14	19	17

Now to mathematically formulate the SPP model for this network, let $G = (N, A)$ be a directed network consisting of a set of nodes $N = \{1, 2, 3\ldots, n\}$ and a set of directed arcs A = $\{(i, j), \quad (k, l), \ldots, (s, t)\}$ connecting m pairs of nodes in N. Each arc has a cost value c_{ij} assigned to it. The SPP can be defined by the following assumptions:

A1. The network is directed and if it is not, the assumption can be fulfilled by transforming any undirected network into a directed one.
A2. No transmission delay or arc cost is negative
A3. The network does not have parallel arcs, meaning there does not exist two or more arcs which have the same starting and ending node.

With these assumptions and notations, the SPP from a specified source node 1 to another specified sink node n can be formulated and solved by:

$$\min \ z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

Subject to

$$\sum_{j=1}^{n} x_{ij} - \sum_{k=1}^{n} x_{ki} = \begin{cases} 1 \text{ for } i = 1 \\ 0 \text{ for } i = 2,3,\ldots,n-1 \\ -1 \text{ for } i = n \end{cases}$$

(Gen et al., 2008, pp. 58-59)

5.2.2 Minimum Cost Flow Model (MCF)

The minimum cost flow model is the most fundamental one of the network optimization models. Whereas the SPP considers only the arc flow costs, the MCF model considers also the capacities which can be sent through an arc. The MCF model thus attempts to find the cheapest way possible of sending a certain amount of flow through the network. Figure 21 illustrates a simple network for MCF model and table 4 illustrates the respective dataset.

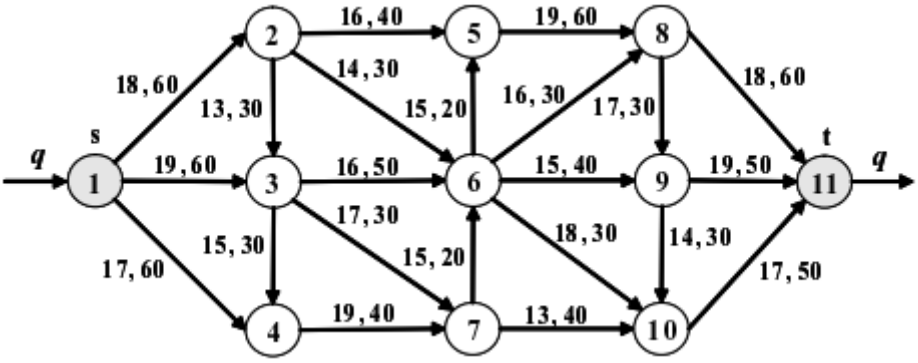

Figure 21: A simple example of MCF model, where the left number of an arc explains the cost of it and the right number explains the capacity of an arc. (Gen et al., 2008, p. 108)

Table 4: The dataset of illustrated MCF model

i	j	c_{ij}	u_{ij}	i	j	c_{ij}	u_{ij}
1	2	18	60	6	5	15	20
1	3	19	60	6	8	16	30
1	4	17	60	6	9	15	40
2	3	13	30	6	10	18	30
2	5	16	40	7	6	15	20
2	6	14	30	7	10	13	40
3	4	15	30	8	9	17	30
3	6	16	50	8	11	18	60
3	7	17	30	9	10	14	30
4	7	19	40	9	11	19	50
5	8	19	60	10	11	17	50

The way to solve the MCF problem of this network would be similar to the solution of SPP network, but instead of only looking at the costs associated with moving from node to node, also the maximum flow constraints on an arc should be taken into account. The notations for mathematical formulation for solving MCF problem for this network are the same as in SPP model, but there is an added variable u_{ij}, which denotes the capacity of (i, j) and q represents the supply/demand for the network. Also, the assumptions for MCF model are slightly different:

A1. The network is directed and if it is not, the assumption can be fulfilled by transforming any undirected network into a directed one.

A2. No capacity, arc cost or supply/demand q are negative.

A3. The network contains none directed path from the start node to the end node with only infinite capacity arcs. Whenever all arcs on a directed path P from the start node to end node have infinite capacity, infinite amount of flow can be sent.
A4. The network does not have parallel arcs, meaning there does not exist two or more arca which have the same starting and ending node.

With these assumptions and notations, the MCF from a specified source node 1 to another specified sink node n can be formulated and solved by:

$$\min \ z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

Subject to

$$\sum_{j=1}^{n} x_{ij} - \sum_{k=1}^{n} x_{ki} = \begin{cases} q \text{ for } i = 1 \\ 0 \text{ for } i = 2,3,\ldots,n-1 \\ -q \text{ for } i = n \end{cases}$$

With flow capacity of

$$0 \leq x_{ij} \leq u_{ij} \quad (i,j = 1,2,\ldots,n)$$

Any set of numbers $x = (x_{ij})$ which satisfy the equations are referred as a feasible flow and q as its value. (Gen et al., 2008, pp. 108-110)

6 CASE COMPANY

6.1 Ramirent Finland Oy as part of Ramirent Group

Ramirent Finland Oy is a construction equipment rental company operating all over Finland. The company was founded in 1955 and is the oldest and founding member of the Ramirent Group, which nowadays consists of different construction equipment rental companies from all over Scandinavia and middle to eastern-Europe. In 2018 the group itself was the market leader in 8 countries, has around 2900 employees in 294 service locations, roughly 25 000 different

product types and a turnover of 712 million euros. In 2019, Ramirent became a part of a pan-European equipment rental leader as it was acquired by a French construction equipment rental company Loxam.

In the same year, Ramirent Finland Oy had a 194 million euros turnover, which was an increase of 3 million euros from the previous year and amounting to 27% of the group's total turnover. It had total of 564 employees in their 58 service locations. The groups turnover consisted of 18% for industry, 57% for construction and 25% to other areas and the sales consisted of 63% on construction equipment renting, 29% on services, 5% on merchandise and 3% on sales of old equipment. (Ramirent, 2019, p. 15-16) These key numbers can be seen in figure 22 and they can somewhat be reflected to Ramirent Finland Oy by saying that most of its turnover comes from renting construction equipment for construction business.

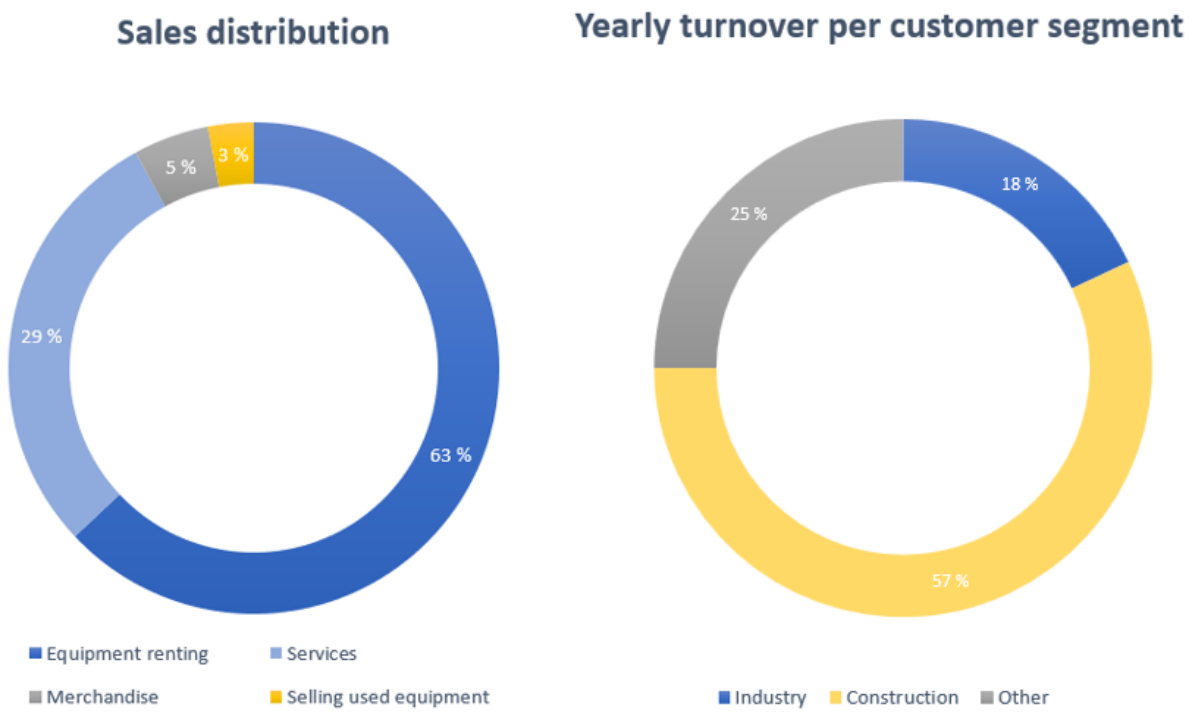

Figure 22: Turnover and sales distribution of Ramirent Group (Ramirent, 2019 p. 15).

6.2 Rental items of Ramirent Finland Oy

When it comes to rental equipment, as of now Ramirent Finland Oy has around 1500 different product groups in 13 product lines. These product groups include serialized items and bulk items and the distribution of products into product lines can be seen in the figure 23.

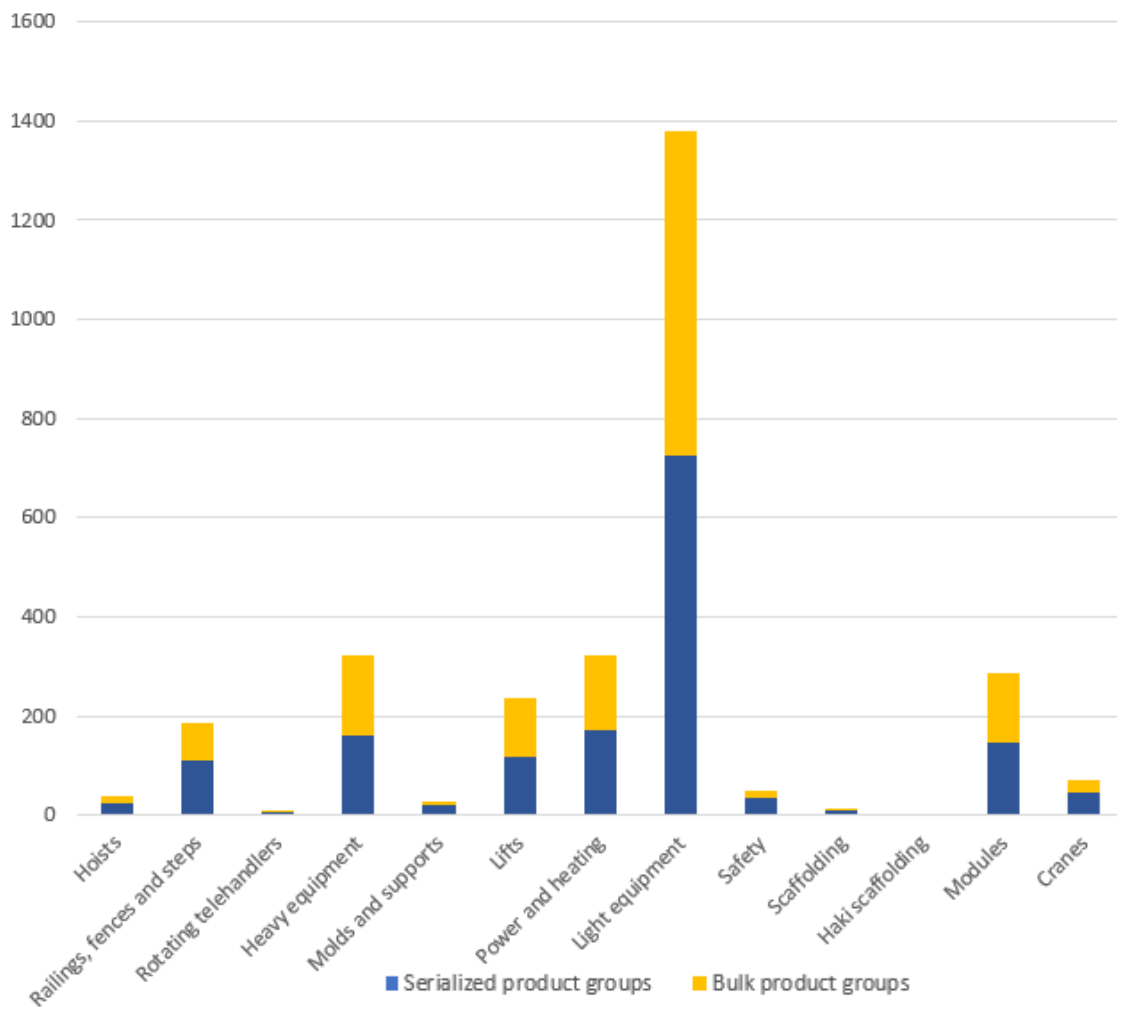

Figure 23: Ramirent Finland Oy product group distribution over product lines

In the figure, the blue indicates the product groups that have serialized items in them and yellow contains only bulk items. As it can be seen, some product lines have more product groups than the others and product groups from such product lines are most likely chosen for the tool. Below are some example products of these larger product lines.

Rotating excavator on wheels from heavy product line

Rotary hammer from light product line

Simple module from Modules product line

Power generator from Power and heating product line

Fence from the railings, fences and steps category

Scissor lift from lifts category

Figure 24: Example products of different product lines.

6.3 Strategic goals of Ramirent

6.3.1 Values, vision & mission

In business, it is crucial to meet the customer demands in a suitable way in an ease. The strategy of Ramirent is focused on capital efficient and sustainable growth in their core business, which is equipment rental and the services connected to it. They are aiming to continually grow especially amongst small and mid-sized businesses in all their operating countries.

These strategic goals can be met by having the right values, vision and mission. Ramirent values smooth operating and the vision of the company is to offer an excellent experience when renting

products from them. This vision is being executed by their mission of providing smooth rental services with positive attitude. (Ramirent, 2019, p. 8)

6.3.2 Optimization as a part of strategic goals

As Ramirent Finland Oy is a part of Ramirent Group it also naturally follows the guidelines of the group's strategic goals. In order to support the execution of the smooth operating strategy goal, it is important to have a smooth and robust system for having rental products on hand where they are needed by the customers. This is where the warehouse stock levels optimization tool being developed in this thesis is aiming to help out.

By optimizing the rental items transfers, Ramirent Finland Oy not only improves the service level for rental equipment customers, but optimizing these transfers should also lower the emissions caused compared to non-optimized transfers even though this is not the main objective of the created tool. However, as a large amount of Ramirent environmental stress comes from transferring products, it is an impact worth mentioning. By optimizing the warehouse levels and the respective transfers for products, Ramirent Finland Oy can also aim for higher revenues whilst maintaining the same amount of product capacity leading to earlier mentioned capital efficiency. (Ramirent, 2019, p. 46)

6.4 Service locations of Ramirent Finland Oy

Ramirent Finland Oy operates across the whole Finland and it has over 60 different service locations across the country. However, the outlets are more concentrated on heavier populated areas, such as southern, eastern and western Finland. The different service locations can be seen from figure 25.

Figure 25: Service locations of Ramirent Finland Oy 2020 (Ramirent Finland Oy service locations. 2020)

These locations are ready to service the customers on weekdays, meaning that there are no rentals done on the weekends or on public holidays.

6.5 Optimization tool as a part of Ramirent Finland Oy's operation

In order to successfully implement the tool as a part of Ramirent Finland's operation, certain boundaries for the tool are set. First of all, the tool is supposed to be on a daily usage for personnel responsible of managing the product transfers between locations. This means that the forecasting will be done on a daily basis and that it is sufficient enough for the company that the tool will forecast only as far as the next operating day.

Secondly, as Ramirent Finland has launched a project to define optimal stock levels for certain service locations for certain rental product groups, this tool will aim to work within the support of these optimal stock levels. Because of this and the fact that Ramirent Finland's fleet consists of variety of different items, some of which would never be included in such tool as this, the

locations and products that the tool will be handling are constrained to the locations and serialized product groups for which optimal stock levels have been defined. This also gives constraints on what item groups transactional data should be used when performing analysis on what methods should be used for different parts of the tool. Based on the given stock levels for the serialized product groups, the model is left with 40 service points around Finland. The geographical distribution of these service points can be seen in figure 26.

Figure 26: The geographical distribution of service points for the tool

This also means that out of roughly 1400 serialized product groups, around 270 will be used in the tool which means that around 20% of the serialized product groups of Ramirents fleet will be represented at all in the tool, which can be seen from the figure 27.

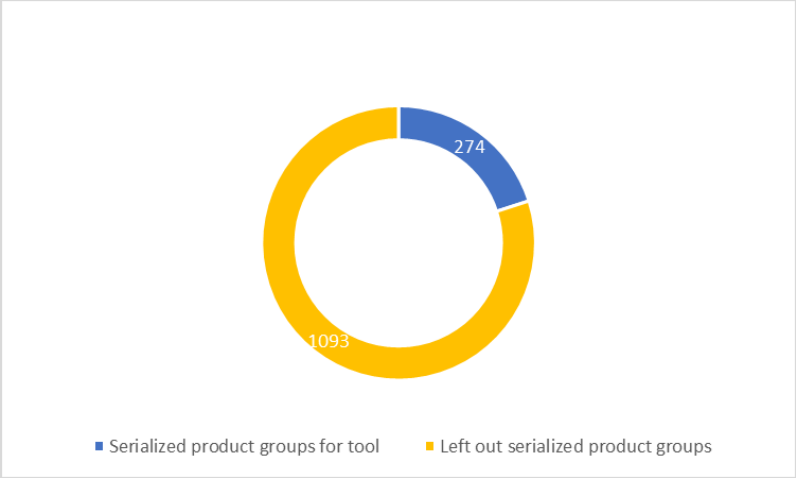

Figure 27: Serialized product groups fraction for tool

The distribution of these included product groups between the different product lines can be seen in figure 28.

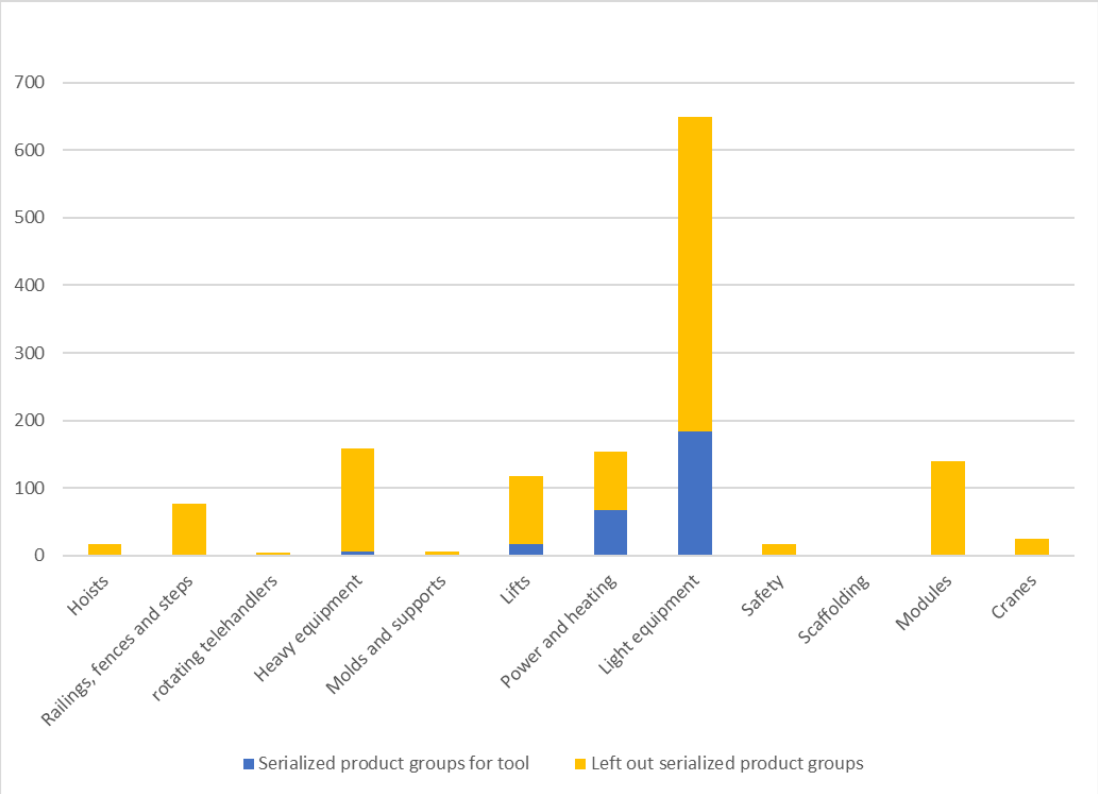

Figure 28: Serialized product group distribution on product lines

It can be noted that the focus is heavily on light equipment and power and heating product lines, which is most likely caused by the fact that they both include a lot of smaller equipment which

are widely available within the service network. Other than that, there might be numerous other reasons why these product groups are chosen, but that is the task left to the Ramirents staff's expert opinion.

After defining the use related boundaries for the tool, there are also some acknowledgements about the development of the tool and its modules:

- The tool aims to provide only a feasible solution in order to support Ramirents strategy of conducting business of certain volume with less rental fleet
- According to existing literature, intermittent demand forecasting is fairly hard to do efficiently, which is why that is one of the major topics studied in this thesis and also comprehensively examined for when it comes to selecting the right forecasting method for the tool. However, simplicity is kept in mind as it is only one of the three major parts of the tool.
- The demand series which the tool uses in order to generate its demand forecasts consist only of the documented demand by Ramirent. This means that there is no knowing how large the actual demand is and how the difference of using the documented demand instead of the real demand affects to the performance of the tool.
- Rental times estimation is a very hard subject with little to no feasible literature to use to my best knowledge. This in addition to the fact that personnel of Ramirent AB have told that this is a very hard subject for the tool and that Ramirent AB has performed a fairly comprehensive study in this on their markets for a similar tool to this one, gives less incentive for consuming time on this subject. This means that the rental times estimation model is implemented keeping simplicity in mind and that there is room to improve this part of the tool.
- There is no constraint regarding the delivery times of items between locations. It is assumed by the company that the item will arrive to the desired location to meet the next day's demand. If this assumption does not sufficiently hold true, an additional constraint for this must be added for the optimization algorithm or the whole logic of the tool must be changed depending on the nature of the actual delivery times.
- As the tool is supposed to help with decision-making without causing any potential drawbacks, the optimization part of the tool will not aim to be purely analytical for when

it comes to moving products, but instead it will be constrained so that there will be no products moved from a location if it causes a shortage at that place even if this shortage would be smaller than in the location where the product would be moved. This should help to prevent possible negative feedbacks on the tool as the service point personnel would not necessarily understand the logic behind the purely analytical approach leading to frustration when their last product was moved away and a for the product occurs.

- The tool will be implemented on python programming language, but the data storing method and programming language may vary. The tool is planned to be implemented into Amazon Web Services (AWS) cloud environment, but the layout for the data architecture of the tool is built in such a way that it can be implemented into a different cloud environment too.

7 CHOOSING MODELS FOR THE TOOL

After a literature review on the areas connected to making this tool, the next step is to choose the used models for each part of the tool. This is done by using the earlier mentioned CRISP-DM framework and taking a look at each model individually before forming a functioning tool.

7.1 Data understanding

As the purpose of this project has been established, it is time to take a deeper look at the data available. The dataset acquired for this project is the rental data of years 2018 and 2019 for the product groups to be included in the tool, which should give a good overall basis for choosing the best methods to be used for each module of the tool. This data is acquired from Ramirents ERP and has the following attributes of which the ones of key interest in terms of building the tool are boldened:

- Product id
- **Product group id**
- **Product group name**
- Amount rented
- Rental order id
- Order row
- Contract id
- **Warehouse id**
- **Warehouse name**
- Business partner id
- Business partner name
- Business partner class
- **Rental date**
- **Return date**
- **Estimated return date**
- State of order
- **State of row**

- **Area id**
- **Area name**

7.2 Initial data preparation

After understanding the data on hand, it is initially prepared. This means that as the data now comprehends the transactions of the product groups desired for all the warehouses of Ramirent, it must be filtered so that it only comprehends the desired product group / service location combinations. After filtering the dataset so that only the transactional data for the desired product and location combinations is present, roughly 101 000 transactions remain.

As this is just an initial data preparation, further data preparation is expected to be made based on the module examined.

7.3 Demand forecasting

As stated earlier, it is one of the key interests to know the structure of the intermittent data pattern before considering applying predictive models for it. To do this, the transactional data of the year 2019 was transformed into a form of having the demand for a certain product group for each date in the data. This data comprehended 2869 different demand series. The weekend dates were also removed from these demand series as these dates have a natural 0 demand since Ramirent does not rent products on weekends. This each demand series having 261 business days. After this, the lumpiness and intermittence values for each demand series were calculated resulting into following descriptive values:

Table 5: Descriptive values of investigated demand series.

	P	CV^2
Mean	73.09	0.07
Median	43.50	0.00
Standard deviation	80.20	0.21
Minimum	2.18	0.00
Maximum	261.00	4.37

These results indicate that an average product of this dataset has 261/43.5 = 6 demand occurrences per year per location and as the squared coefficient of variation of demand sizes is 0 and the most present demand size in the demand series is 1, it means that these average products have a demand of 1 when demand occurs. However, the mean values for P and CV^2 are larger, which indicates that on average there is some fluctuation between demand sizes and that roughly 261/73.09 = 3.57 demands occur. It can also be noted that the standard deviation between the yearly demand occurrences is around 261/80.2 = 3.25 times and the demand series include product location combinations having only 261/261 = 1 demand occurrences per year to having 261/2.175 ≈ 120 demand occurrences per year.

The descriptive P and CV^2 values for each demand series can be illustrated with the same cutoff values as presented by Syntetos et. al (2005, p. 499) in their research of suitable models for intermittent demand. This illustration can be seen in the figure 29.

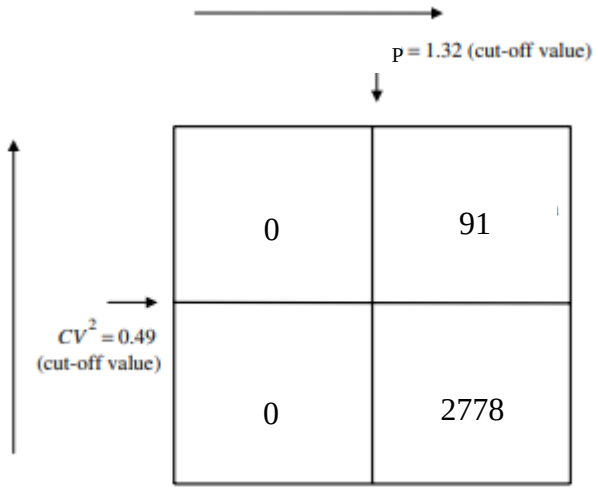

Figure 29: Descriptive P and CV^2 values for investigated demand series.

Based on this, two hypothesis can be made with first of them being that the Croston's method should have realized benefits over exponential smoothing method on these demand series and the second of them being that the SBA method should outperform Croston's method with realistic α smoothing constant values.

After examining the structure of the demand series on hand, 5 different demand forecasting models were implemented on the data: Croston's method, SBA method, TSB method, improved SBA method and the improved SBA method where smoothing constant α is used in the U(t) formula. These methods were tested on each demand series with the α and β smoothing constant values of 0.01, 0.0.5, 0.15, 0.5, 0.85, 0.95 and 0.99. The performance of each model was measured by the absolute accuracy measures of MSE and ME and after this a sensitivity analysis was performed as averaged accuracy measure values were plotted for specific α and β smoothing constants resulting into the following kind 3d-plots:

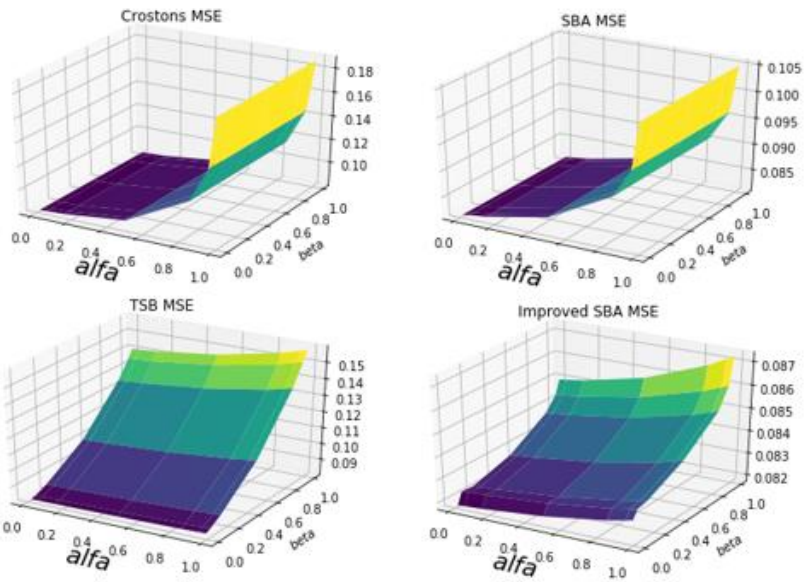

Figure 30: Tested forecasting methods α, β specific averaged series MSE accuracies.

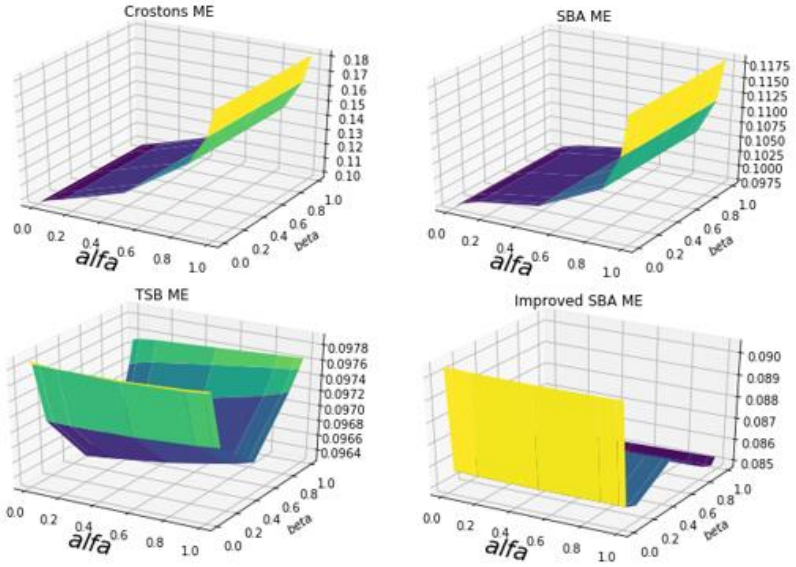

Figure 31: Tested forecasting methods α, β specific averaged series ME accuracies.

Now it can be noted that SBA method seems to indeed outperform Croston's method based on both ME and MSE on every α, β combination tested and that small smoothing constant values tend to commonly yield the best MSE and ME performance. However, in the case of ME accuracy measure on TSB model it seems that the β value of 0.5 yields the best accuracy.

To better compare the relative performance of these models, the best performance of each model was plotted against each of the models with a PB comparison resulting into two plots: one based on the best MSE performance and the other based on the best ME performance. These plots are illustrated in the figure 32 and the accuracy measures of each method in table 6. Note that in the performance axis are swapped between the plots as this gives a better view.

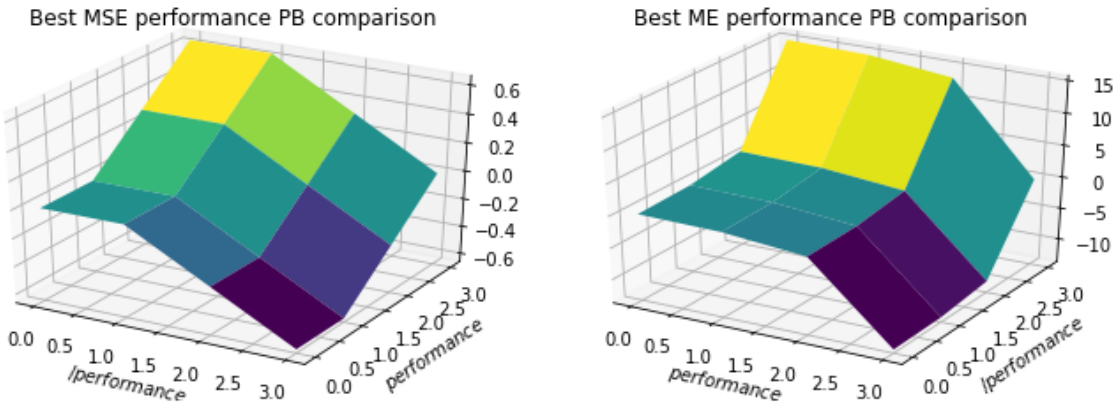

Figure 32: Best MSE and ME performance PB comparison of tested models, with 0,1,2,3 meaning Croston's, SBA, TSB and improved SBA in corresponding order.

Table 6: Best MSE and ME performance values and smoothing parameters

Method	**Best MSE performance**	**α-value MSE**	**β-value MSE**	**Best ME performance**	**α-value ME**	**β-value ME**
Croston's	0.08130284	0.01	-	0.09763593	0.01	-
SBA	0.08130176	0.01	-	0.09742266	0.01	-
TSB	0.08156057	0.01	0.01	0.09628067	0.01	0.5
Improved SBA	0.08182082	0.01	0.01	0.08479387	0.01	0.95

From these results it can be seen that there is no single best performing model or most beneficial individual smoothing parameter scenario as the improved SBA model performs the best on smoothing values of $\alpha = 0.01$ and $\beta = 0.95$ based on ME and the original SBA model performs the best on smoothing values $\alpha = 0.01$ and $\beta = 0.01$ based on ME. This indicates that the Improved SBA model performs better on average than the normal SBA model but is more prone to individual higher errors of which the MSE accuracy metric punishes a lot.

To better understand how these best performing models would forecast based on a demand scenario, both of their forecasts for one of the demand series is plotted. This demand series is one of the used ones in the analysis and it is of an item 214, which is a light engine, at a warehouse 67, which is located in the southern Finland. This demand series had a total of 25 demand occurrences and a total demand of 29 units.

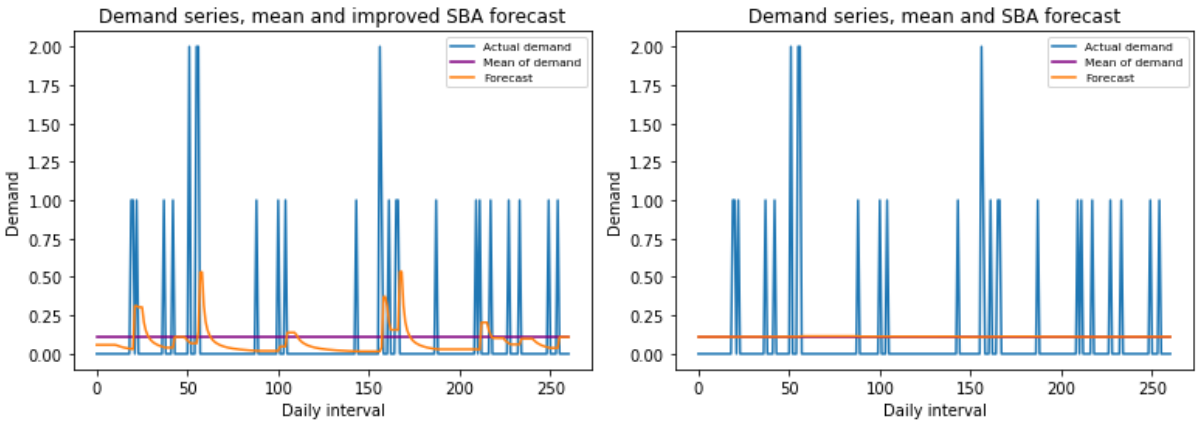

Figure 33: Demand series, mean, SBA and improved SBA forecast.

From the plots it can be seen that the improved SBA method is somehow trying to forecast the occurring demand spikes whereas the SBA method forecasts are nearly just the plain mean of the demand series values. Needless to say, both of them seem to do somewhat poor job on forecasting the actual demand spikes. However, this might not be the worst way to forecast the demands as it was examined that an average series had a total of 6 demand occurrences with the average demand size of being one with 0 variation meaning that a series with 25 occurrences is a fairly active demand series.

This being said, the difference becomes notably different when these results are aggregated for every service location present in the data:

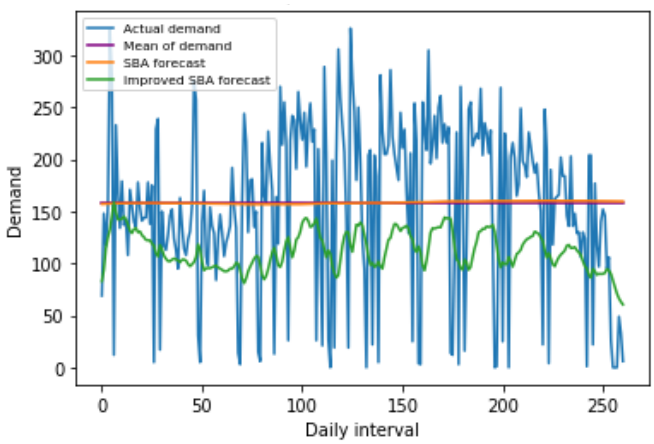

Figure 34: SBA and improved SBA backwards forecasts aggregated across all warehouses.

From figure 24 it can now be seen that the improved SBA model reminds of the pattern of the actual demand, but it tends to underestimate it. This underestimation is most likely caused by the fact that the average demand present in the data series is very intermittent as discussed before and thus setting the forecast lower increases the performance when ME accuracy metric is used. SBA method on the other hand just forecasts very close to the mean, causing it to perform good on MSE accuracy metric. This being said, the analysis calls for new accuracy metrics and most likely new examined methods too.

As an answer to this, a different approach is taken: the demand of these items in 2018 should somewhat represent the demand of these items in 2019. Based on this assumption, a plot of demand series above and the last year's corresponding demand series is made:

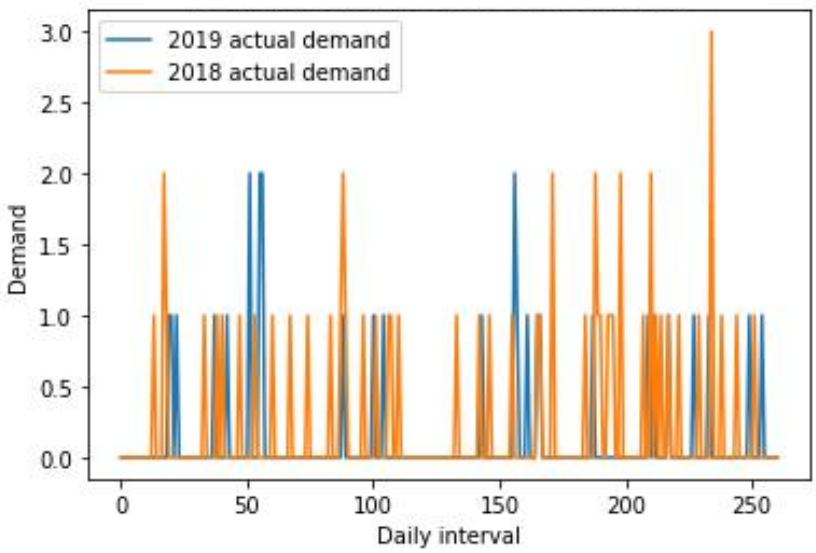

Figure 35: 2018 and 2019 item location specific demand series on business days.

From figure 35 it can be seen that even though the demand series don't correspond to each other fully, there are still notable similarities between them. As the demand of the exact days does not seem to match, but there certainly seems to be a noticeably pattern between the longer demand trends on the series, the weekly and monthly mean for the 2018 series demand are calculated and plotted against the 2019 demand series daily demand:

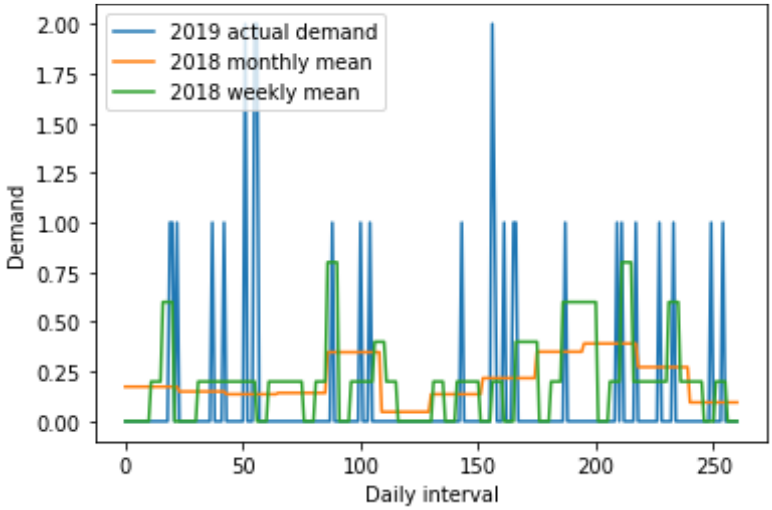

Figure 36: 2019 demand series daily demand and 2018 periodic means.

Now from figure 36 it can be seen that especially the weekly mean, even though being fairly conservative and sometimes mispredicting, is able to capture the pattern of the 2019 demand series at some level. If these results are aggregated to all warehouses, the ability of these periodic means to capture the demand trend compared to the tested ones is fairly obvious:

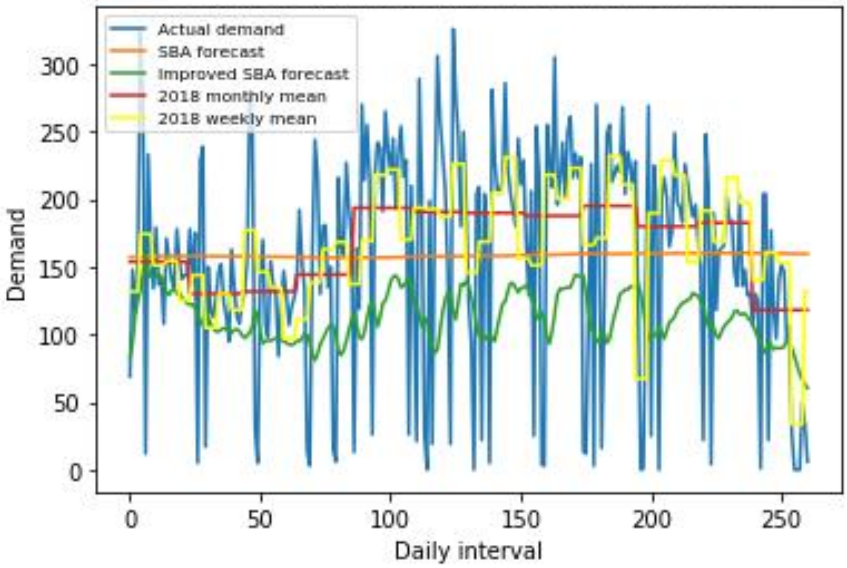

Figure 37: 2019 demand series and different forecasting methods.

Based on these findings, another round of data-analysis on the demand series is conducted, with this time testing the daily, weekly and monthly mean methods based on last year's data and in addition to this, also the true positive rate (TPR) per series of the demand forecasting method are calculated. This is because using a plain error rate seems to favor very conservative models and might lead to erroneous conclusions when selecting the forecasting method for the model.

After conducting the analysis, the following kind of smoothing parameter sensitivity plots were made of different models' true positive rates and the non -smoothing parameter sensitive past year periodic mean models results are shown in table 7.

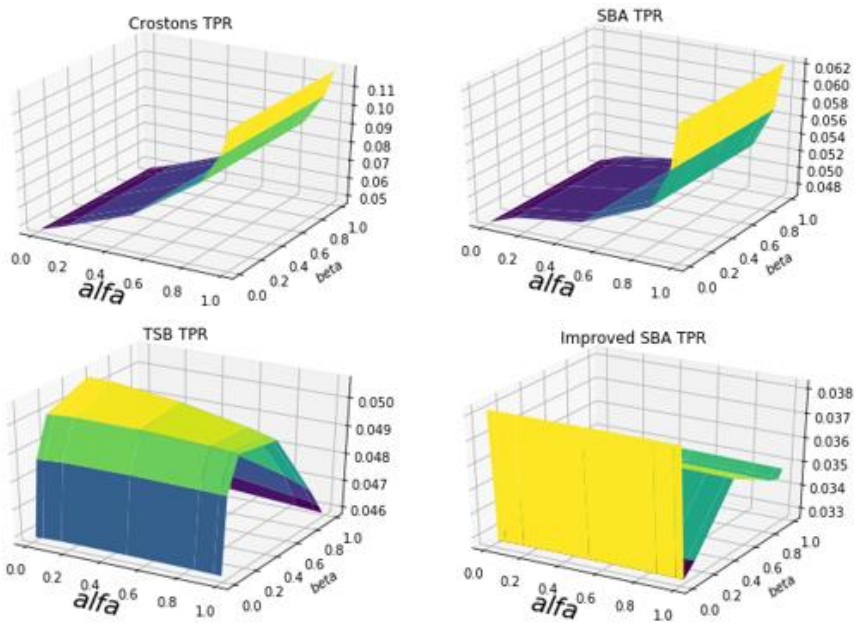

Figure 38: Averaged true positive rate of tested models

Table 7: Periodic mean model's descriptive accuracies

	Monthly mean	**Weekly mean**	**Daily "naïve"**
MSE	0.08713516	0.10284879	0.18536751
ME	0.099299	0.09875354	0.09912592
TPR	0.05314605	0.05575825	0.05209793

From these results it can be seen that there seems to be a tradeoff between the models TPR and MSE/ME: a model which has relatively low MSE or ME, performs poorly in terms of TPR and vice versa. A good example of this is the Croston's method: on α value of 0.99 it performs

extremely bad based on MSE and ME accuracy metrics but performs extremely good based on TPR accuracy metric. One exception to this however is the TSB model, which seems to have its best results with the same smoothing parameter values based on TPR and ME. It can also be seen that from the periodic mean models, the weekly mean seems to perform the best out of the 3 based on the average TPR and ME.

If now the best performing based on different accuracy metrics models are plotted on a single location/item-based demand series and on a single item-based demand series, the results look the following:

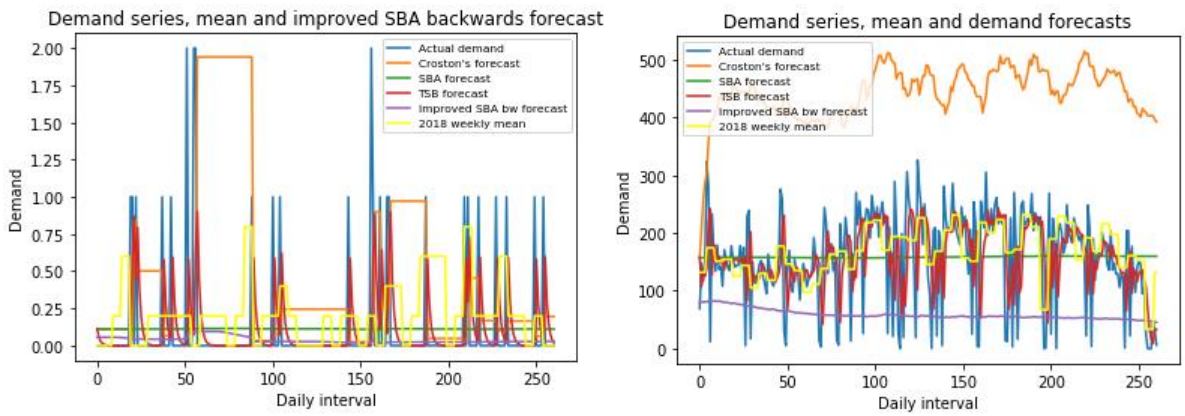

Figure 39: Forecasts and actual demand on demand series

From figure 39 it can be seen that Croston's method indeed should produce high TPR values, but at the cost of forecasting too high values.

Based on all of this analysis, as there seems to be no single method to outperform all of the other methods based on the accuracy metrics used, a few different style representative methods are chosen to be further tested on a simulation loop of the actual model of the tool and based on results obtained from that, it can be determined which of the methods should be used in the final tool and if other methods tested here should be considered to be tested in the simulation also. These methods are: SBA method with $\alpha = 0.01$, which should represent a pure mean type approach, TSB method with $\alpha = 0.01$ and $\beta = 0.5$, which should represent a somewhat demand pattern following approach and the 2018 weekly mean method, which should represent a fairly conservative demand pattern following approach.

7.4 Rental times estimation

As the core idea of rental times estimation was already explained before, there is not much of a need to do method selection for the rental times estimation in this chapter anymore. However, it is important to focus on the performance of the selected method and to explain thoroughly how it will be working as a part of the tool.

7.4.1 Data preparation

As there are many different possible error factors connected to the documentation of the rental dates, return dates and the estimated return dates, the data on hand needs to be prepared in such a way that the erroneously documented observations are either removed or fixed. The documentation errors found in the data were mostly such that the month and the day were mixed up or the year of the rental was mixed (this happened mostly near the change of the year). As trying to determine the real causation of the erroneous documentation of the observations attributes is hard, it was decided that these observations were to be removed. This happened in two parts: first the observations with a return date before the actual rental date or an estimated return date before the return date were removed and after that 1% of outliers of rental length, rental estimate length and rental length error were removed.

After removing the surely falsely documented observations, roughly 83 000 observations remained out of 101 000 and after removing the outliers roughly 81 000 out of 83 000 observations remained.

7.4.2 Model performance testing

After the required data preparation for the rental times estimation models, both of the earlier proposed methods were tested by two different ways. The first way was to calculate the error of expected return amount across the whole rental time. This would represent the overall correctness of the rental estimation of the model. The second way was to calculate the error of expected return amount on the actual return date of the transaction. This would tell how capable the model is to estimate the exact rental time of a transaction.

The transactional data used for these performance tests was the 2019 transactional data of the prepared dataset, which included roughly 39 000 transactions. For the triangle method, there was no need to use other than the 2019 transactional data, but for CDF-model, a CDF consisting of 2018 rental transactions was used for each transaction. The hierarchy of choosing the right CDF of relative rental times for the CDF-model was the following:

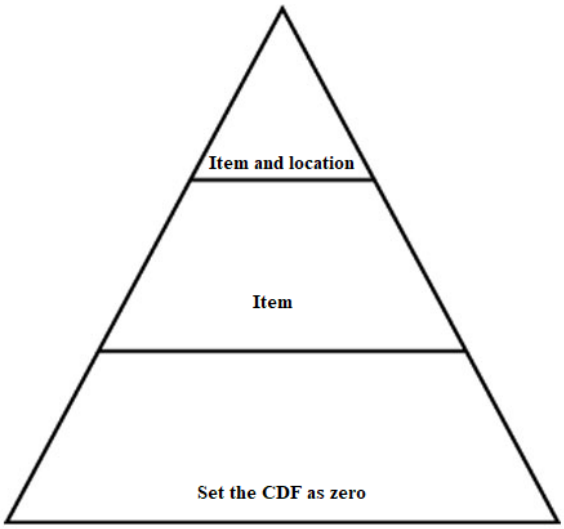

Figure 40: The CDF hierarchy of the CDF-model.

This ensures that there are no errors if the past data has missing items, locations or the combinations of these two which are present in the 2019 data. If the item was not present in the 2018 data there is no reliable estimate of the rental times of the item and it was assumed to be returned right away so that at least it would not try to underestimate the estimated available stock when implemented as a part of the built tool. After executing the mentioned performance tests on both models, a few error metrics of the expected number of products to return per transaction were gathered and are illustrated in table 8 below.

Table 8: rental times estimation methods performance tests results.

	CDF-method	Triangle-method
ME on return day	0.375	0.316
MSE on return day	0.258	0.275
ME across rental days	0.449	0.514
MSE across rental days	0.333	0.439

From these results it can be seen that the hypotheses made in chapter 4 seem to be correct: the CDF-method performs overall better than the triangle method, but the triangle-method is on average more capable of predicting the exact rental time of the transaction. However, even though the CDF-method has a higher mean error of estimated products returned on the return day, it has a lower mean squared error of this than the triangle method. This means that even though the triangle method is on average more correct on estimating the exact return date on a transaction, it is more prone to making larger individual errors of the expected returns than the CDF-method.

As there is neither of these methods can be deemed as being the superior one based on these results, both of them will be tested with the chosen rental times estimation methods.

7.5 Optimization method

After the tool has successfully forecasted the occurring demand and the estimated returns for the products, it still needs to optimize the stock levels for tomorrow based on these forecasts and the current situation stock situation. In order to do this product/location stock optimization the optimization algorithm of the tool needs to know:

- S_t = The current stock
- $\hat{S}_{t+1}$ = The estimated stock for tomorrow
- $\hat{S}_t$ = The estimated stock for today
- C_{ij} = The unit cost of moving a product from location to another
- R = The daily renting price of a product

When these features are fed to the optimization algorithm the first thing it does is it calculates the node values (which represent the stock level for a product in a warehouse) with the formula:

$$nodevalue = \min\ (S_t, \hat{S}_{t+1}, \hat{S}_t)$$

By this it knows if the node has products available to be sent or if the node needs products. These values are also floored to the nearest integer number as fractions of products cannot be sent and fractions of products cannot either be received. However, the negative stock values of nodes are stored, as they are possibly needed in the revenue calculation for the transports made. After this, the following assumptions are given for the network:

1. Products can only be moved from a stock positive node to a stock negative node
2. The only and always most efficient way of moving a product between locations is a straight node to node connection

This transforms the network into a directed one by only allowing transports from a negative node to a positive node. Figure 41 and 42 illustrate the plain and transformed networks for a fictional optimization scenario.

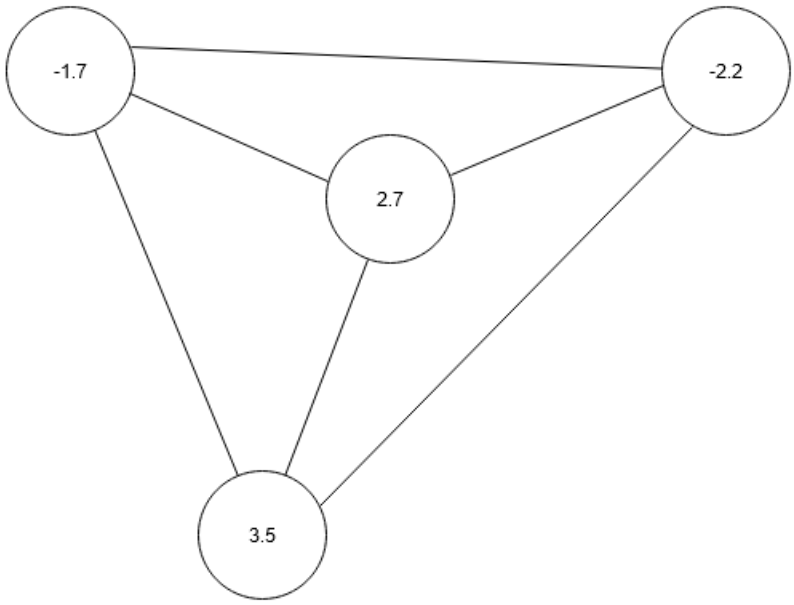

Figure 41: Plain warehouse network of fictional scenario.

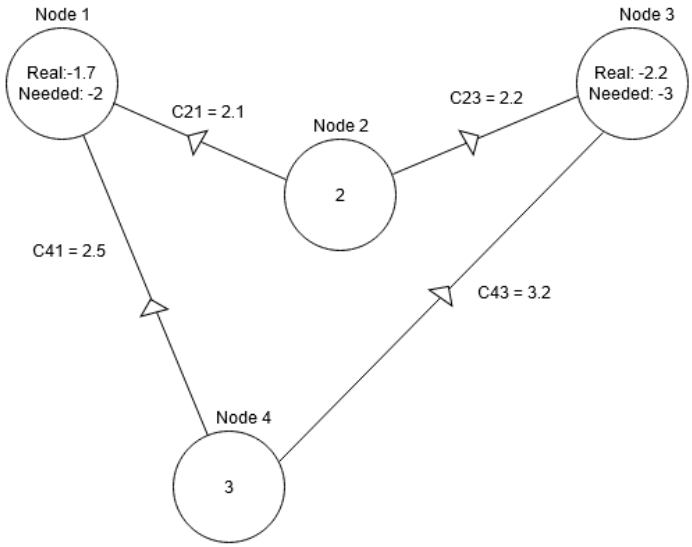

Figure 42: warehouse network of fictional scenario transformed for optimization.

Now it can be seen that the transformed network only has paths available from negative nodes to positive nodes and those paths are also directed. The stock levels have also been transformed in the way described earlier. After the transformation has been made the optimization algorithm performs the following kind of network optimization:

Objective function

- Maximize profit from moving products

Constraints

- Can only move integer number of products from a node to a node
- Maximum number of products which can be moved from a node is its stock level
- Maximum number of products which can be moved to a node is the stock deficiency

With a daily product rental price of R = 4 the optimized product transports within the network would look like the following:

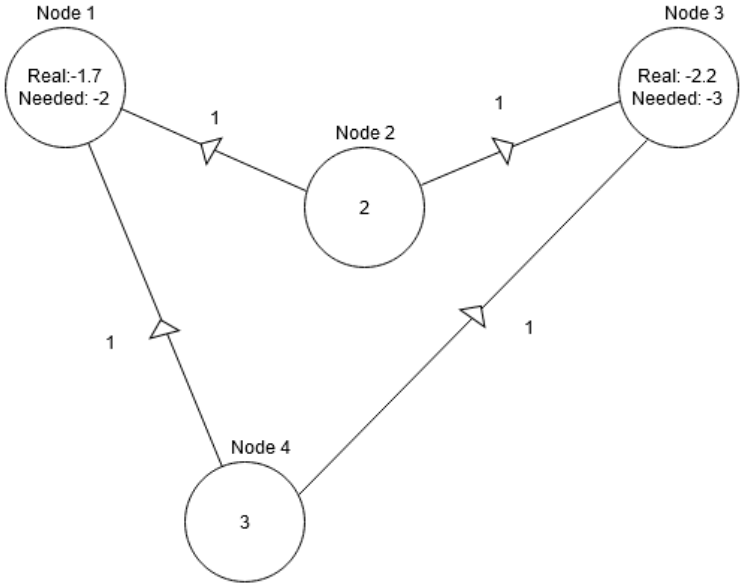

Figure 43: Optimized product transports in the network

Now it can be seen that 4 transports are made, but 1 is left out even though the total demand for the nodes is 5 products. This is because the fraction which remains in the end on node 3 is 0.2 products, which would amount to 4*0.2 = 0.8 worth of revenue, but the lowest cost of transportation was 2.1 to begin with, so there is no incentive of moving a product to meet that remaining demand of 0.2 products as it would not be cost efficient.

The optimization algorithm does the network optimization similarly to this process explained to every product present in the tool and after that has the optimized networks for each product.

8 BUILDING THE TOOL MODEL

In this chapter the framework for the tool is built in order to reflect what inputs each module of the tool needs and what outputs they should give. After this, the different forecasting methods chosen before are tested as a part of a functioning system with a simulation analysis. Finally, when the used models for the tool are confirmed the functionality of the tool is validated and the final data architecture for the tool is built in order to be able to implement the tool in its working environment.

8.1 Framework of the tool

This chapter presents the framework and the inputs needed in order for the tool to work. The inputs presented here might change for the final version of the tool, but the essential information presented by these inputs will most likely be needed in some format.

8.1.1 Initial inputs

The initial inputs which the tool needs to work are:

- The currently rented out equipment. This will be automatically filtered based on product and location combinations given.
- Optimal stocks for product and location combinations. Only these product and location combinations are used in the tool.
- The daily rental prices for products. Only products which are present in this listing are used in the tool.
- The current stock for the products and locations. This will be automatically filtered based on product and locations given.
- The prices of moving an item per kilometer
- The latitude and longitude values of warehouses present in the tool
- Inputs such as median rental times for location/product from the past data

Initially, the following outputs are formed:

- The used product location combinations
- The Distances between service locations

8.1.2 Demand forecasting module inputs and outputs

The demand forecasting method of the tool needs the following inputs to work:

- Past data attributes. This depends on the used forecasting method. For example, on Croston's method it would require the Z and P values for today.
- The product and location combinations received from the initial inputs

After this, the demand forecasting will be made for product location combinations and the following outputs are given:

- The forecasted demand for today
- The forecasted demand for tomorrow

8.1.3 Rental times estimation module inputs and outputs

After demand forecasting module, the rental times estimation module needs the following inputs to work:

- Inputs such as median rental times for location/product from the past data
- The forecasted demand for today
- The forecasted demand for tomorrow
- The current stock levels

After this, the rental times estimation will be made for product location combinations and the following outputs are given:

- Expected return for currently rented out equipment for today
- Expected return for currently rented out equipment for tomorrow

- Expected return for todays forecasted demand for today
- Expected return for today forecasted demand for tomorrow
- Expected return for tomorrows forecasted demand for tomorrow

After this, in this module also some values which are needed in the optimization algorithm are calculated:

- **Todays estimated stock** = max(0, current stock + expected return for currently rented out equipment for today – forecasted demand for today + expected return for todays forecasted demand for today)
- **Tomorrows estimated stock** = Todays estimated stock + expected return for currently rented out equipment for tomorrow – forecasted demand for tomorrow + expected return for todays forecasted demand for tomorrow + expected return for tomorrows forecasted demand for tomorrow

8.1.4 Optimization module inputs and outputs

After preceding modules, the optimization module needs the following inputs to work:

- The daily rental prices for products from the initial inputs
- The current stock
- Todays estimated stock
- Tomorrows estimated stock
- The costs of moving a single product per kilometer
- The distances between service locations

After the optimization, the module gives the following outputs:

- Suggested product transportations to be made based on optimized stock scenario
- Additional information of the suggested product transportations

8.2 Evaluation of different compositions of the tool

After building the framework for the tool, it was implemented by using Python and options to simulate the operation of the tool with different demand forecasting methods, rental times estimation methods and different time periods were also implemented. The purpose of the simulation option was to simulate the rental operation of Ramirent on a daily basis on working days and the process of it can be determined as follows:

1. Determine starting stocks for each product group for each service location
2. Determine the used forecasting methods
3. Determine the time period of which the simulation takes place on
4. Run the simulation day by day
    a. Add incoming transported products to stocks
    b. Add served rental transactions which are returned on this day to stocks
    c. Determine needed product transports based on forecasts and execute them
    d. Serve rentals that took place on the given day within the number of current stocks
    e. Deduct served rental amounts from stocks
    f. Collect wanted KPIs on a daily basis

After this, combinations of different pre-chosen demand forecasting method, rental times estimation method and time period were simulated and a few KPIs were collected from each of the simulation. Table 9 illustrates the mean and table 10 the standard deviation of collected KPIs for each demand forecasting and rental time estimation model of 6 different time periods consisting of 2 months of transaction data of 2019 of a subset which resembles the properties of the selected item/location combinations. The stock level used for the simulations were the rounded mean values of given desired minimum and maximum stock levels for different warehouses on different product groups. These desired stock levels had been earlier gathered from the staff of different service locations by Ramirent and they should be able to reasonably represent the distribution of different product groups within different warehouses.

The combination of these simulations serves as a robustness analysis for each combination of these models. Table 11 illustrates the same KPIs for each model combination for a whole year of these 2019 transactions compared to a scenario where no warehouse stock optimization is performed at all. These KPIs include the transactions served, not served, utilization percent, revenue generated, service percent and amount of shipments and the costs of them. Utilization is a KPI used by Ramirent for their fleet and the formula of it can be found in appendix J.

Table 9: Mean values of robustness analysis.

Rental times estimation method	CDF				Triangle "activation function"			
Demand forecasting method	TSB	SBA	Improved SBA	Weekly mean	TSB	SBA	Improved SBA	Weekly mean
Served	1359.66	1352.33	1357.67	1366.5	1352.83	1347.67	1349.67	1357.5
Not served	167.5	174.833	169.5	160.668	174.334	179.5	177.5	169.667
Utilization	26.627	26.177	26.397	26.651	26.479	26.11	26.367	26.93
Revenue generated (€)	300 004	294 763	297 286	300 334	298 242	293 987	297 161	304 487
Served-%	89.43	88.848	89.273	89.885	88.935	88.553	88.727	89.24
Shipments	47.667	14.5	45	52	42.5	14.167	47.833	46.833
Shipment costs (€)	228.519	97.87	287.051	244.046	229.762	100.908	358.872	240.649

Table 10: Standard deviation of robustness analysis.

Rental times estimation method	CDF				Triangle "activation function"			
Demand forecasting method	TSB	SBA	Improved SBA	Weekly mean	TSB	SBA	Improved SBA	Weekly mean
Served	287.905	293.058	288.507	287.312	288.653	290.649	286.891	287.750
Not served	62.042	57.044	62.076	61.278	62.568	59.52	62.815	61.227
Utilization	1.711	2.318	1.784	2.152	1.881	2.478	2.200	2.654
Revenue generated (€)	22031	29927	23465	27769	24389	31579	28333	34291
Served-%	1.718	1.24	1.643	1.679	1.634	1.342	1.527	1.54
Shipments	18.446	5.115	3.899	11.96	16.74	4.587	4.49	10.763
Shipment costs (€)	111.695	40.351	40.716	119.552	128.392	28.483	51.089	121.702

Table 11: 2019 transactions scenario values relative to the not forecasting scenario.

Rental times estimation method	CDF				Triangle "activation function"				Not forecasting
Demand forecasting method	TSB	SBA	Improved SBA	Weekly mean	TSB	SBA	Improved SBA	Weekly mean	Not forecasting
Served	64	-6	51	165	-121	0	-105	-16	7655
Not served	-64	-6	-51	-165	121	0	105	16	1508
Utilization	0.582	0.064	0.326	1.244	0.118	0.12	-0.036	1.137	23.667
Revenue generated (€)	43 655	4 529	23 771	93 221	10 153	8 356	-4 958	87 897	1 584 156
Served-%	0.698	-0.066	0.557	1.801	-1.321	0	-1.146	-0.175	83.543
Shipments	189	41	184	249	167	39	157	202	0
Shipment costs (€)	948.67	296.73	1072.34	1271.63	880.67	244.22	989.99	1202.14	0

Based on the gathered mean and standard deviation values of different KPIs, it can be seen that the weekly mean forecasting method combined with the CDF rental times estimation method outshines the other ones: it outperforms the other methods on average and it is fairly consistent for when it comes to for example the served rentals. It also has a fairly high standard deviation between the shipments made, which is a good indicator that the model is working as intended as the demand of the products is fairly seasonal meaning that in some time intervals there should be more shipments needed and in some time intervals less shipments.

It can also be noted though that it has fairly high standard deviation when it comes to the service percent, but it seems that the on average poorly performing models on this KPI have a low standard deviation in this, meaning that the achievable service percent between time series might also differ and that the poor models perform poorly on every time series which results into low standard deviation.

The weekly mean forecasting method combined with CDF rental times estimation method superiority can ultimately be seen on the whole year simulation results table though as the model seems to outshine other models by far. The results of other models are also fairly correlating the expectations: the improved SBA model with the triangle rental times estimation method overshoots the forecasts and thus results in a very poor performance and the SBA

method just commonly doesn't really do much as the demand it forecasts is always fairly low and it can be seen in the very similar KPI values as in the scenario of not doing any optimization.

8.3 Validation of the chosen composition of the tool

After determining the superior combination of demand forecasting and rental times estimation methods, this combination was further tested with a dataset containing all the products to be included in the tool, meaning that a total of 40 warehouses and 274 product groups were included. The performance of this tool was compared to two other scenarios: the scenario where no optimization is done and also to a tool which is able to predict the future demand and returns by taking its forecasts directly from the forecasted day of the dataset of past data used. This model should serve as a benchmark of how well the tool could perform with exact forecasts and it is referred as the "Perfect tool" in the thesis.

In addition to this, the comparison between the tools was done in two different stock scenarios: one of which had the rounded mean values of the desired stock levels as the total fleet and the other in which the stock levels were adjusted as a minimum of the desired stock levels. The mean scenario had a total fleet of 1171 products and the minimum stock scenario had a total fleet of 890 products. As the demand within the used dataset is the demand only documented by Ramirent, meaning that there might exist a larger demand for the products if enough fleet was available, the sensitivity analysis between stock levels should give some information about the outcome when the demand/fleet ratio is higher than what it is perceived to be.

Table 12 illustrates the KPIs gathered from a half a year simulation of both stock scenarios for each model.

Table 12: Simulation results on different stocks and tool scenarios.

Total fleet	890			1171		
Tool scenario	Not optimizing	The final tool	The perfect tool	Not optimizing	The final tool	The perfect tool
Served	12 846	13 992	17 457	15 207	15 747	17 977
Not served	5 728	4 582	1 117	3 367	2 827	597
Utilization	32.350	36.057	44.227	27.793	29.074	34.11
Revenue generated (€)	14 087 042	15 819 509	20 397 643	16 982 970	17 806 859	21 179 640
Served-%	69.16	75.33	93.99	81.87	84.78	96.79
Shipments	0	1 853	4 236	0	1 038	2 570
Shipment costs (€)	0	49 750	118 049	0	23 558	74 874

Now it can be seen that even with the perfect forecasting models, the service percent was not a full 100%, which means that there might be too little stock for the demand or that to meet the demand, the cost of moving the product would have surpassed the estimated profit made from the transfer. The service percent for each model scenario was also more sensitive to go down the more suboptimal the model scenario was.

Now what this means is that even with a suboptimal model usage, the achieved performance can be most likely increased by having the correct demand stock ratio for the products. However, this also means that the better the model performs, the harder the performance is to increase meaning that if the service percent is already high as a baseline, it will be harder to increase even if the forecasts of the used tool were on point. All in all, the chosen model seems to perform reasonably well compared to not doing optimization in both stock scenarios as the generated revenue grew by roughly 12% and 5% in the different stock scenarios and the service percent also increased by roughly 6 and 3 percent in different stock scenarios respectively. The results also clearly indicate that a certain demand can be met with lesser fleet with the usage of the tool. This being said, there is still a lot room for the model to improve especially on its demand forecasting and rental times estimation.

As the evaluated model of the tool seems to be performing well enough, offer reasonable solution to the business problem at hand and also be implementable to an environment chosen by the company, it is chosen as the model suggested to be deployed.

8.4 Data architecture of the final tool

As there are a lot factors which are hindering the implementation of the tool in its working environment, such as the ongoing switch of the ERP system and the resulting product hierarchy changes, the deployment of the final tool in its working environment will not be part of this thesis. However, the general data architecture for the tool in its working environment is presented and illustrated in figure 44.

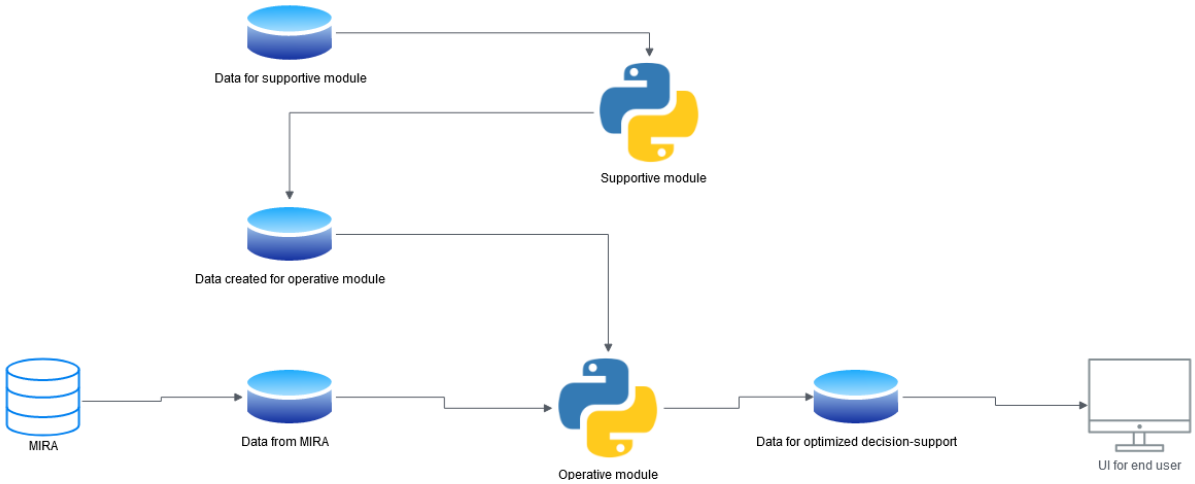

Figure 44: Proposed data architecture for the tool in its working environment.

The usage of the tool will be twofold: supportive module will be run rarely, on an yearly basis or so, and the operative module will be run operatively, meaning it will be run daily or even hourly if wanted. What is also known is that the tool is planned to be implemented in the AWS cloud environment, which supports an architecture of this kind.

8.4.1 Supportive module

The supportive module, which is rarely run, is the one which is responsible of creating the tables for the operational module to use in its forecasts. This means that the module should be run

whenever there is need to update these tables. Such reasons may include the need to update the time frame of past data from which the demand forecasts are generated or the addition of new locations or items to the tool. The module will take the necessary input tables from a database and insert generated tables to a database for the operative module to use. The usage of the supportive module could be thought as the training phase of the tool.

8.4.2 Operative module

After the supportive module has been run, the operative module can be run. For this the module needs the generated tables of the supportive module and also data from the ERP, MIRA. The data needed from MIRA includes for example the current stock levels, ongoing transfers, currently active transactions and possibly occurred transactions on the date which the module is ran. As there is a high possibility that this information will be only available on a daily basis from a query run overnight, it is most likely possible that the module will only be run once a day. If this is the case, the module can be set to be run automatically for example before the branches open and the results of it can be viewed during the day at any given point.

After the operative module has calculated its optimized decisions, it stores variety of information regarding this. This can include for example the optimized transfers, costs related to them and estimated profits connected to them. After this the UI built for the tool can access this data from the database and present and informative output for the end user for decision-making.

9 CONCLUSIONS

In conclusion, the subject handled in this thesis seems fairly alienated to the construction rental business industry. To my best knowledge there does not seem to exist easily accessible knowledge about demand forecasting and the rental times estimation of different rental products in different branches of a specific company. However, the demand patterns found resemble the type of intermittent demand, of which there is plenty of helpful literature available. It is still somewhat astonishing to see that the models which were developed specifically to this type of demand pattern were ultimately outshined by a very simplified model. Given the high intermittency and low demand amount occurrences though, this result is not that surprising.

Based on this there is also no incentive to believe that the intermittency and lumpiness value based neural network models presented in this thesis would be superior to the chosen weekly mean method, but to the non-linear forecasting capability of them might set them to be the superior option. Also, an alternative method of using neural networks could be considered for this. However, that would be a suggestion for another research. Also, as the demand series used to generate the demand forecasts consist only of the documented demand, the forecasts made are bound to be erroneous at some level. To find out the impact of this and to possibly document the real demand in order to generate more accurate forecasts could also be considered as another research topic. However, the usage of this tool might in time lead to a situation where the documented demand gets closer to the actual demand and the demand forecasting accuracy increases.

The rental times estimation methods presented in the thesis are perceived to work in an acceptable level and given the fact that there already exists a research conducted by one of the Ramirent Groups member countries, there was no incentive to pursue the development of these models further. However, to implement a more sophisticated model based on the research already made, the performance of the tool could be most likely improved.

Fortunately, there was plenty of literature available of the optimization models and it was somewhat clear that what kind of an optimization method should be used. However, the logic which the optimization algorithm uses to determine the required and available resources within

the network could be modified based on different opinions. This also somewhat applies to the objective function of the algorithm as the estimated profits from the implemented transports depend on the estimated daily rental prices, rental times and the shipment costs, which were determined fairly superficially.

All in all, a successfully working stock optimization model for Ramirent Finland Oy's chosen rental fleet was developed and can be implemented in a form of a tool in an environment of their choice. There exist many focal points on which the model could be improved on and improving these points could be a good subject for future research.

REFERENCES

Aljandali, A. 2017. Multivariate Methods and Forecasting with IBM® SPSS® Statistics. Springer International Publishing. 178 p.

Altay, N., Litteral, L. & Rudisill F. 2012. Effects of correlation on intermittent demand forecasting and stock control. Int. J. Production Economics. Vol. 135, pp. 275–283

Aragón, F., Goberna, M., López, M. & Rodríguez, M. 2019. Nonlinear Optimization. Springer, Cham, Switzerland. 350 p.

Armstrong, J. 2001. Principles of forecasting: a handbook for researchers and practitioners. Kluwer Academic Publishers. 849 p.

Armstrong, J., Green, K. & Graefe, A. 2015. Golden rule of forecasting: Be conservative. Journal of Business Research. Vol. 68, pp. 1717-1731

Axler, S. & Ribet, K.A. 2010. Linear Optimization The Simplex Workbook. Springer, New York, NY. 272 p.

Azevedo, A. & Santos, M.F. 2008. KDD, SEMMA and CRISP-DM: a parallel overview. *ISCAP - Informática - Comunicações em eventos científicos.*

Babai, M., Dallery Y., Boubaker, S. & Kalai, R. 2019. A new method to forecast intermittent demand in the presence of inventory obsolescence. Int. J. Production Economics. Vol. 209, pp. 30-41

Berthold, M.R., Borgelt, C., Höppner, F., Klawonn, F. 2010. Guide to Intelligent Data Analysis: How to Intelligently Make Sense of Real Data. Springer, London. 394 p.

Bramer, M. 2016. Principles of Data Mining, Third Edition. Springer, London. 526 p.

Brockwell, P. & Davis, A. 2016. Introduction to Time Series and Forecasting, Third Edition. Springer International Publishing Switzerland. 425 p.
Chatfield, C. Time-series forecasting. 2005. Significance. Vol. 2, No. 3, pp. 131-133

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. [WWW-document]. [referred 10.7.2020]. Available: ftp://ftp.software.ibm.com/software/analytics/spss/support/Modeler/Documentation/14/UserManual/CRISP-DM.pdf

Collet, J-F. 2010. Discrete Stochastic Processes and Applications. Springer International Publishing. 220 p.

Cottle, R. & Thapa, M. 2017. Linear and Nonlinear Optimization. Springer, New York, NY. 614 p.

Dávila, V., Cabral C. & Zeller C. 2018. Finite Mixture of Skewed Distributions. Springer, Cham. 101 p.

Fayaad, U., Piatetsky-Shapiro, G. & Smyth P. 1996. From Data Mining to Knowledge Discovery in Databases. AI Magazine. Vol. 17, No. 3, pp. 37-54

Fernández, A., García, S., Galar, M., Prati, R.C., Krawczyk, B. & Herrera F. 2018. Learning from Imbalanced Data Sets. Springer International Publishing. 377 p.

Garg, A.K. & Tadj L. 2017. Emerging Challenges in Business, Optimization, Technology, and Industry Proceedings of the Third International Conference on Business Management and Technology, Vancouver, BC, Canada 2017. Springer International Publishing. 232 p.

Gen, M., Cheng, R. & Lin, L. 2008. Network Models and Optimization, Multiobjective Genetic Algorithm Approach. Springer-Verlag. London. 692 p.

Gutierrez, R., Solis, A. & Mukhopadhyay S. 2008. Lumpy demand forecasting using neural networks. Int. J. Production Economics. Vol. 111, pp. 409–420

Kourentzes, N. Intermittent demand forecasts with neural networks. 2013. Int. J. Production Economics. Vol. 143, pp. 198-206

Kourentzes, N. 2014. On intermittent demand model optimisation and selection. International Journal of Production Economics. Vol. 156. pp. 180-190

Kurgan, L. A. & Musilek P. 2006. A survey of Knowledge Discovery and Data Mining process models. The Knowledge Engineering Review. Vol. 21, No. 1, pp. 1–24

Nocedal, J. & Wright, S.J. 2006. Numerical Optimization, Second Edition. Springer, New York, NY. 664 p.

Qin, G., Li, J., Jian, N., Li, Q & Wang, L. 2013. Warehouse Optimization Model Based on Genetic Algorithm. Mathematical Problems in Engineering. Vol. 2013, pp. 1-6

Ramirent. 2019. Ramirent Annual Report 2018.

Ramirent Finland Oy. 2020. Ramirent tuoteluettelo.

Ramirent Finland Oy service locations. 2020. Available: https://www.ramirent.fi/?gclid=EAIaIQobChMItsz867Kx6wIVEZiyCh2kpQ_3EAAYASAAEgJac_D_BwE

Salehzadeh, R., Tabaeeian, R.A. & Farahnaz E. 2020. Exploring the consequences of judgmental and quantitative forecasting on firms' competitive performance in supply chains. Benchmarking: An International Journal. Vol. 27, No. 5, pp. 1717-1737

SAS Institute Inc. 30.8.2017. Introduction to SEMMA. [WWW-document]. [referred 9.7.2020]. Available:

https://documentation.sas.com/?docsetId=emref&docsetTarget=n061bzurmej4j3n1jnj8bbjjm1a2.htm&docsetVersion=14.3&locale=en

Shafique, U. & Qalser, H. 2014. A Comparative Study of Data Mining Process Models (KDD, CRISP-DM and SEMMA). International Journal of Innovation and Scientific Research. Vol. 12, No. 1, pp. 217-222

Soyer, E. & Hogarth, R. 2015. The golden rule of forecasting: Objections, refinements, and enhancements. Journal of Business Research. Vol. 68, pp. 1702-1704

Syntetos, A.A. & Boylan J.E. 2001. On the bias of intermittent demand estimates. Int. J. Production Economics. Vol. 71, pp. 457-466

Syntetos, A.A. & Boylan J.E. 2005. The accuracy of intermittent demand estimates. International Journal of Forecasting. Vol, 21, pp. 303 – 314

Syntetos, A.A., Boylan J.E. & Croston J. 2005. On the categorization of demand patterns. Journal of the Operational Research Society. Vol. 56, pp. 495–503

Syntetos, A.A, Zied Babai, M. & Gardner Jr., E. 2015. Journal of Business Research. Vol. 68, pp. 1746-1752

Teunter, R. & Sani, B. 2006. On the bias of Croston's forecasting method. Lancaster University Management School, Working Paper. Pp. 1-11

Teunter, R.H. & Duncan L. 2008. Forecasting intermittent demand: a comparative study. Journal of the Operational Research Society. Vol. 60, pp. 321-329

Teunter, R., Syntetos, A. & Zied Babai, M. 2011. Intermittent demand: Linking forecasting to inventory obsolescence. European Journal of Operational Research. Vol. 214, No. 3, pp. 606-615

Thompoulos, N.T. 2015. Demand Forecasting for Inventory Control. Springer, Cham. 183 p.

Wallström, P. & Segerstedt, A. 2010. Evaluation of forecasting error measurements and techniques for intermittent demand. Int. J. Production Economics. Vol. 128, pp. 625–636

Xu, Q. & Wang, N. 2012. A Review of Croston's Method for Intermittent Demand Forecasting. 9th International Conference on Fuzzy Systems and Knowledge Discovery. Pp. 1468-1472.

Zhang, G., Patuwo, B.E. & Hu, M.Y. 1998. Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting. Vol 14, pp. 35–62

Appendix A.

The SEMMA process, developed and used by SAS Institute, is applicable across a variety of industries and provides methodologies for business problems such as fraud detection, householding, customer retention and attrition, market segmentation, risk analysis, customer satisfaction etc. SEMMA considers a cycle with 5 stages for the data mining process: (SAS Institute Inc. 2017)

1. Sample: This is an optional stage which consists of sampling the data by extracting a portion of a large data enough to contain the significant information, yet small enough to be manipulated quickly.
2. Explore: This stage consists of the exploration of the data by searching for trends and anomalies in order to gain significant insights on the examined data.
3. Modify: This stage consists of the modification process of the data by creating, selecting and transforming the variables in it. This may include for example outlier detection.
4. Model: Stage consisting of modeling the data by allowing the software for this to automatically search for a combination of data that reliable predicts a desired outcome. SAS offers for example neural networks, tree-based models, logistic models and other stat models for this, but when thought on a framework level, the used model can be of the framework's users choosing.
5. Assess: The final stage which consists of assessment of the used model by measuring its usefulness, reliability and performance. This can be done by various methods.

Although the SEMMA process is independent from the used tools in the process, it is heavily linked to the SAS Enterprise Miner software and aims to guide the user on following the software's ways. However, SEMMA offers an easy to understand process, allowing an organized and adequate development and maintenance of DM projects. (Azevedo & Santos, 2008, p. 3; Shafique & Qalser, 2014, p. 220) Figure 45 shows a diagram which illustrates the stages of the SEMMA process.

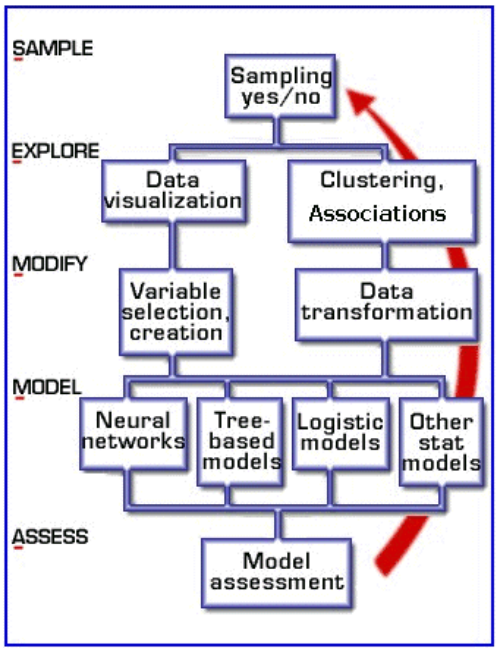

Figure 45: the SEMMA process (SAS Institute Inc. 2017).

Appendix B.

In the late 1980s, the explosive data growth generated the emergence of a new field of research called KDD (Knowledge Discovery Databases). Within these acronyms is hidden "*the non-trivial process of discovering valid, new patterns, potentially useful and understandable in large volumes of data*". The KDD process has served to unite researchers from different areas of principle in the search for useful techniques to find the potential knowledge that is engaged in the large volumes of data stored by organizations. (Fernández et al., 2018, p. 2-3)

The concept of a KDD process model was first discussed during the first workshop on KDD in 1989, the main driving factor being the acknowledgement of the fact that knowledge is the end product of a data-driven discovery process. In 1996, the foundation of the process model was defined and did not address particular DM techniques, but rather provided support for the complex and highly iterative process of generating knowledge while also emphasizing the close involvement of a human analyst in the process. (Kurgan & Musilek, 2006, p. 3-4)

The KDD process uses DM methods to extract knowledge using a database along with any required preprocessing, sub sampling and transformation of the database. Fayaad et al. (1996, p. 42) describes the KDD process with nine basic steps:

1. **Developing an understanding of the application domain** and the relevant prior knowledge while also identifying the aim of the KDD process from the customer's point of view.
2. **Creating a target dataset** by selecting a dataset, or concentrating on a subset of variables or data samples, on which discovery is to be executed
3. **Cleaning and preprocessing the data** by removing noise, collecting information to model or account for noise, deciding on strategies to handle missing data fields and accounting for time sequence information and possible known changes.
4. **Data reduction and projection**: finding useful features to represent the data depending on the aim of the task. Using dimensionality reduction or transformation methods is an effective way for the reduction of variables.
5. **Matching the goals of the KDD process to a data-mining method**. These may include summarization, classification, regression, clustering etc.
6. **Exploratory analysis and model hypothesis selection**: selecting the data-mining algorithms(s) and selecting the method(s) to be used for searching the data for patterns. This process contains deciding which models and parameters might be suitable and matching a particular data-mining method with the overall criteria of the KDD process (for example the end user might be more concerned in knowing the models predictive capabilities than understanding the model itself).
7. **The data mining itself**: searching for patterns of interesting. These may include classification rules or trees, regression and clustering. To significantly help the data-mining method, the user should perform the preceding steps correctly.
8. **Interpreting the mined patterns**, if no patterns are found steps 1 through 7 can be repeated for further iteration. This step can also include visualization of the mined patterns and models or visualization of the data given the mined models.
9. **Acting on the discovered knowledge**: it can be used directly, integrated into another system for further actions or simply documented and reported to interested parties. This process also involves checking for and solving potential disputes with previously assumed knowledge

In addition to the defined steps, the KDD process can involve significant iteration and contain loops between any steps. This and the whole KDD process is illustrated with a diagram in figure 46.

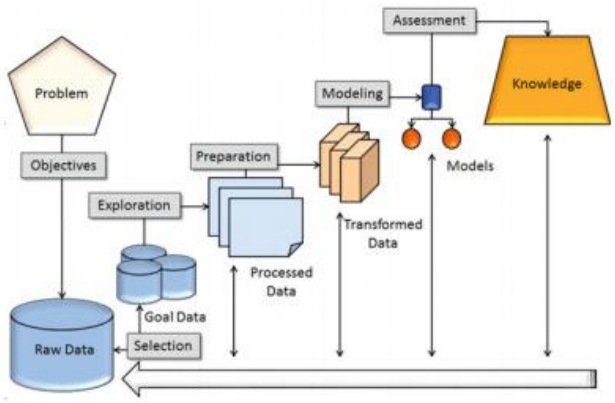

Figure 46: The KDD-process (Fernández et al., 2018, p. 3).

Appendix C.

Artificial neural networks are biologically motivated learning systems developed to solve problems in all areas. Human brains are composed of a number of connected processing elements called neurons, or as referred in neural networks, nodes. NNs include perceptrons, which are brain-like single layer networks compromising neurons receiving input information to learn a pattern or relationship between input and output. (Gutierrez et al., 2008, pp. 409-420) The functional relationship estimated by the NN can be written as:

$$y = f(x_1, x_2 \dots, x_p)$$

Where x_1, x_2…, x_p are *p* independent variables and *y* is a dependent variable. In addition, for time series forecasting problem, the inputs are typically the past observations of the data series and the output is the future value, meaning the NN performs the following function mapping:

$$y_{t+1} = f(y_t, y_{t-1} \dots, y_{t-p})$$

Where y_t is the observation at time t. (Zhang et al., 1998, p. 38) Moreover, a layer of units (referred as hidden layer) is required to mathematically approximate any relationships between input and output. Because of this, multi-layered perceptron (MLP) became a popular NN tool for computation. The MLP network consists of a layer of input nodes, one or more layers of hidden nodes and of a layer of output nodes. All the nodes of certain layer are connected to all the nodes of the next layer and vice versa. The values connections have assigned values to them called connection strengths or weights. The output of each node in an MLP network is a function of the inputs from the connecting nodes of the previous layer to itself and the assigned weights. These outputs are sometimes referred as activation values and the function assigning these outputs is called an activation function resulting in so that the outputs of the input layer nodes have just the values of the input variables. The values of the weights are used in MLP to learn the mathematical relationship between the input and output variables of which equivalent nonlinear regression model form of one feed forward (information flow from input to output) NN is the following:

$$\log \hat{y}_{t+h} = \beta\hat{}_{j,h} + \sum_{j=1}^{n} (\beta\hat{}_{j,h}\, \mathrm{f}(I, \hat{w}_{h,j})$$

Where I_t is the input vector of current time period value and lagged values of $\log \hat{y}_{t+h}$. The parameter h is the forecast horizon and $\hat{w}_{h,j}$ is the network weight vector corresponding to the forecast horizon h and jth hidden node. The $f(I,\hat{w}_{h,j})$ is the activation function of the model used in research conducted by Gutierrez et al. (2008, pp. 409-420) and defined as follows:

$$f(I_t, \hat{w}_{h,j}) = (1 + e^{-z})^{-1}$$

Where

$$z = \hat{w}_{h,j,\phi} + \hat{w}_{h,j} t + \sum_{i=1}^{l} (\hat{w}_{h,j,i} \log y_{t+h-i})$$

MLP needs a training algorithm to learn this nonlinear mathematical relationship, for which the back propagation is the most widely used method. It is a supervised learning technique where the values of the independent variables along with the values of the dependent variables are fed to an MLP network input layer. The task of the training algorithm is to extract the functional relationship between the input and the dependent variable, called “target”, through proper assignment of weights. The values of the target variables are

represented by the output layer units. MLP uses the information available from the independent variables for each observation in the training data to compute an output value, which is then compared with the target value to generate error signals for all units. If there is no difference between the two values, there will be no error signal. The training involves a backward pass through the network during which error signals are sent to all units in the network and the weights in the network are changed proportionally to the error signals. The constant proportionality in this procedure is called the learning rate. The larger this learning rate is the larger are the changes in the weights in each step. The MLP network learning process can be fast with a high learning rate, but a too high one can lead to oscillation of the learning process. One way of dealing with this is to use a momentum factor in the weight change formula, which is a constant that determines the effect of past weight changes on the current direction of movement in the weight space effectively filtering out high-frequency variations of the error surface in that space. The learning is completed when an error function based on the output node errors is minimized. (Gutierrez et al., 2008, pp. 409-420) As finding the global minima can be hard within the constraints of time for a general nonlinear optimization problem, the most we can do is to use the available optimization method which can give the "best" local optima if true global solution is not available. For time series forecasting problem, a training pattern consists of a fixed number of lagged observations of the series. If we have N observations $y_1, y_2, \ldots, y_N$ in the training set and we need 1-step-ahead forecasting, then using ANN with *n* input nodes, we have *N*-n training patterns. The first training pattern is composed of $y_1, y_2, \ldots, y_n$ as inputs and y_{n+1} as the target output. The second one is $y_2, y_3, \ldots, y_{n+1}$ and y_{n+2} and the last training pattern will be $y_{N-n}, y_{N-n+1}, \ldots, y_{N-1}$ for inputs and y_N for the target. (Zhang et al., 1998, p. 39) For example, Kourentzes (2013, p. 200) computed the one step ahead forecast Y'_t using inputs that are lagged observations of the time series where *I* denotes the number of input p_i of the NN with the functional form:

$$Y'_t = \beta_O + \sum_{h=1}^{H} \beta_0 g\,(\gamma_{0i} + \sum_{i=1}^{I} \gamma_{hi} p_i)$$

Where w = (β,γ) are the network weights with $\beta = [\beta_1, \ldots, \beta_H]$ and $\gamma = [\gamma_{11}, \ldots, \gamma_{HI}]$ first being for the output and the second being for the hidden layers. The β_0 and γ_{0i} are the biases of each neuron, which function as the intercept in a regression for each. H is the number of hidden nodes in the NN and g(·) is a non-linear transfer or activation function providing the non-linear capabilities to the model. Typically, an SSE (sum of squared errors) based objective

function (also known as cost function or error function) to be minimized during the training process is:

$$E = \frac{1}{2} \sum_{i=n+1}^{N} (y_t - a_t)^2$$

Where a_t is the actual output of the network and ½ is there to simplify the expression of derivatives computed in the training algorithm. (Zhang et al., 1998, p. 39)

Appendix D.

Table 13: Mean demand, standard deviation, CV^2 and ADI of NN study datasets by Gutierrez et al. (2008, p. 413).

Series	1	2	3	4	5	6	7	8
Mean	251.02	262.08	271.60	274.43	278.01	324.84	237.09	274.31
S.D.	1078.80	985.19	1305.36	1221.31	1191.04	1387.20	743.88	1134.55
CV^2	18.47	14.13	23.10	19.81	18.35	18.24	9.84	17.11
ADI	4.51	4.25	4.78	3.97	3.77	3.73	5.21	4.73
Series	9	10	11	12	13	14	15	16
Mean	253.77	346.04	303.11	321.61	299.15	296.07	288.78	305.81
S.D	959.19	1710.19	1229.80	1149.70	1425.87	1321.28	1090.65	1257.98
CV^2	14.29	24.43	16.46	12.78	22.72	19.92	14.26	16.92
ADI	4.03	4.83	5.14	4.83	5.44	4.68	4.39	4.41
Series	17	18	19	20	21	22	23	24
Mean	228.74	352.32	322.98	355.48	328.70	394.84	314.33	410.00
S.D	889.07	1480.69	1054.75	1609.05	1390.67	2675.95	1438.57	1929.56
CV^2	15.11	17.66	10.66	20.49	17.90	45.93	20.95	22.15
ADI	4.30	4.09	3.90	4.86	4.09	4.37	3.38	3.39

Appendix E.

Table 14: Average of nonzero demand sizes of time series in NN study by Gutierrez et al. (2008, p. 414).

Series	Average of nonzero demand sizes		
	All ($n = 967$)	Training ($n = 624$)	Test ($n \leqslant 343$)
1	826	732	937
2	799	497	1132
3	831	699	979
4	804	635	956
5	779	480	1118
6	897	530	1225
7	732	566	888
8	824	548	1064
9	737	469	970
10	1023	662	1311
11	867	709	993
12	915	617	1147
13	890	786	979
14	868	626	1064
15	821	535	1082
16	875	542	1160
17	676	387	913
18	971	877	1052
19	849	662	1018
20	1023	826	1209
21	919	905	934
22	1197	1423	936
23	881	963	785
24	1257	1467	991

Appendix F.

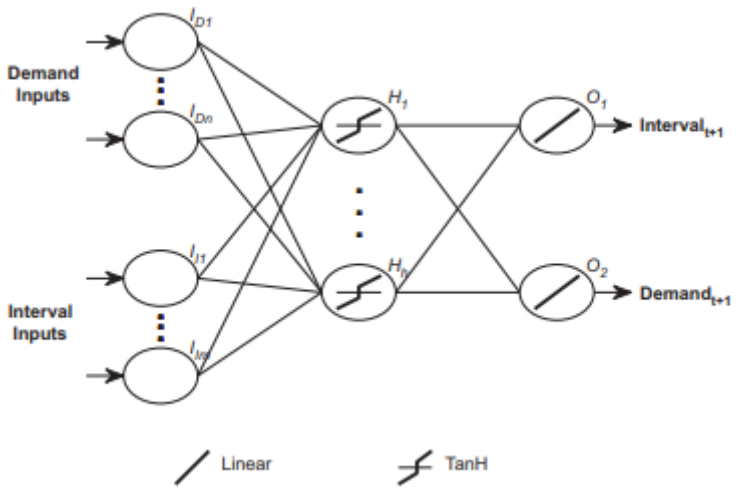

Figure 47: The NN-Dual architecture, with the variable number of demand I_{DN} and interval I_{IM} lagged inputs and hidden nodes H_h, which use the TanH activation function. Two linear output nodes provide the demand and interval forecasts. (Kourentzes, 2013, p. 201).

Appendix G.

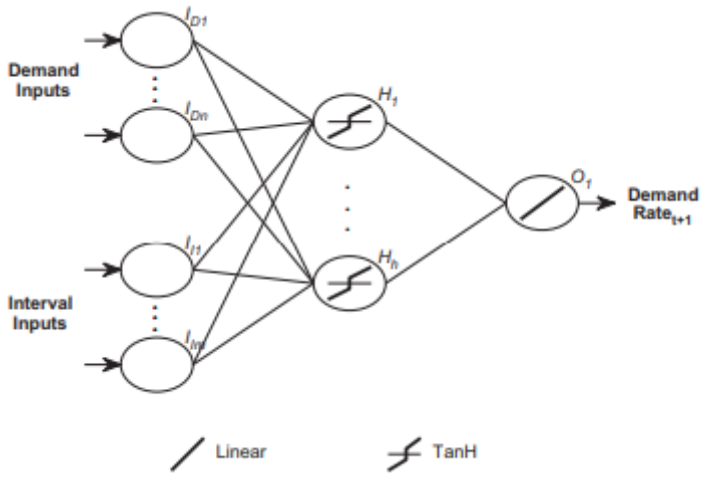

Figure 48: The NN-Rate architecture, with the variable number of demand I_{Dn} and interval I_{Im} lagged inputs and hidden nodes H_h, which use the TanH activation function. A single linear output is used to provide the demand rate forecast. (Kourentzes, 2013, p. 201).

Appendix H.

The framework of linear optimization can be illustrated for example with a diet problem. In this problem, you are allowed to eat from the following menu, which is illustrated in table 3, and you must satisfy a certain amount of supplements and vitamins from your meal with the least amount of calories.

Table 15**:** meal supplements and calories (Axler & Ribet, 2010, p. 2)

	$\%A$	$\%C$	$\%Calc$	$\%Iron$	$Calories$
Hamburger	4	4	10	15	250
Chicken	8	15	15	8	400
Fish	2	0	15	10	370
Cheeseburger	15	6	30	20	490
Requirements	10	10	15	15	

This means that for example if one eats fish and cheeseburger from the menu, it will amount to respective supplement percentages of 17, 6, 45 and 30 along with 860 calories. To mathematically formulate this, we can set y_1 through y_4 to be the amounts consumed from the

menu from hamburger to cheeseburger respectively and come to the following observation that the total percentage of Vitamin C eaten is $4y_1 + 15y_2 + 0y_3 + 6y_4$. We can also notice that the requirement for this is $4y_1 + 15y_2 + 0y_3 + 6y_4 \geq 10$, as the requirement for Vitamin C was 10 percent. These type of requirements for the problem can be referred as problem constraints and they are all linear inequalities as we are dealing with linear optimization problem. (Axler & Ribet, 2010, pp. 1-2). Moreover, for a problem to be considered as a LOP, it must contain at least one linear inequality.

In order to better understand LOPs and the constraints connected to it, one must grasp such concepts as linear functions, linear equations and linear inequalities. In simplicity and for the concept of LOPs. Linear function can be defined as

$$f(x_1, x_2, \dots, x_n) = \mathrm{c}_1 x_1 + \mathrm{c}_2 x_2 + \cdots + \mathrm{c}_n x_n$$

where c_1, c_2, … c_n are constants and x_1, x_2, … x_n are variables. In this thesis all of these can be assumed to be real-valued. Knowing this, linear equation can be defined as

$$f(x_1, x_2, \dots, x_n) = \mathrm{K}$$

and linear inequalities as

$$f(x_1, x_2, \dots, x_n) \geq \mathrm{K} \qquad \text{and} \qquad f(x_1, x_2, \dots, x_n) \leq \mathrm{K}$$

Where K is a constant and $f(x_1, x_2, \dots, x_n)$ is a linear function. When the right-hand side K is zero, these functions are said to be homogeneous. (Cottle & Thapa, 2017, pp. 5-6).

After defining the mathematical equations for the problem constraints, there still needs to be an equation for the problem which being solved. In this case, we would like to minimize the calories intake from the meal, meaning that we want to minimize $250y_1 + 400y_2 + 370y_3 + 490y_4$ as those were the calories from the menu's meals from hamburger to cheeseburger, respectively. This is the final building block of this (and every) linear optimization problem (LOP) and it is referred as the objective function of the problem. Usually, a separate variable is reserved for the usage of this function, for example w.

Now that we have laid the foundations for constraints and the objective function of the optimization problem, the problem can be written as:

Minimize

$$w = 250y_1 + 400y_2 + 370y_3 + 490y_4$$

Subject to

$$4y_1 + 8y_2 + 2y_3 + 15y_4 \geq 10$$
$$4y_1 + 15y_2 + 0y_3 + 6y_4 \geq 10$$
$$10y_1 + 15y_2 + 15y_3 + 30y_4 \geq 15$$
$$15y_1 + 8y_2 + 10y_3 + 20y_4 \geq 15$$

And

$$y_1, y_2, y_3, y_4 \geq 0$$

It is worth noting that the last constraint is a nonnegativity constraint, which should always be included when the problem does not allow negative amounts, for example in this case one wouldn't be able to consume a negative amount of cheeseburgers from the menu.

Now to solve this, one does not have to do it by hand as there exists different algorithms and computational methods to find the optimal values for the problem. For example, by using the star superscript, the following optimal values can be connoted: The winner would consume roughly 424.156 calories (exact amount of w^* = 766450/1807) by eating 535/1807 of the hamburger (y^*_1), 810/1807 of the chicken sandwich (y^*_2), 0 amount of the fish sandwich (y^*_3) and 630/1807 of the cheeseburger (y^*_4). (Axler & Ribet, 2010, pp. 3-4)

Appendix I.

Now as the optimization solving progress might be a black box type, meaning the exact progress of the solving is not explained by the used algorithm, one might want some validation about

whether or not the optimal solution provided is correct. To tackle this issue, a so called dual of the original LOP can be constructed:

Maximize

$$z = 10x_1 + 10x_2 + 15x_3 + 15x_4$$

Subject to

$$4x_1 + 84 + 10x_3 + 15x_4 \leq 250$$
$$8x_1 + 15x_2 + 15x_3 + 8x_4 \leq 400$$
$$2x_1 + 0x_2 + 15x_3 + 10x_4 \leq 370$$
$$15x_1 + 6x_2 + 30x_3 + 20x_4 \leq 490$$

And

$$x_1, x_2, x_3, x_4 \geq 0$$

Now if z* is the highest z with these conditions and w* points out the minimum w subject to its constraints, then z* = w*. This is what is referred as the Strong Duality Theorem and it has powerful implications in topics such as Game Theory, Linear Algebra, Geometry and Economics.

Now knowing this, one of the qualities of LOP problem is that we can be easily convinced of the minimality of the given practical solution without knowing the details which lead to the solution. This can be done by simply multiplying the first constraint by x^*_1 = 27570/1807, the second by x^*_2 = 24880/1807, the third one by x^*_3 = 0 and the final one by x^*_4 = 16130/1807 and add them up. More information about how these figures can be obtained is found in the book: Linear Optimization The Simplex Workbook. (Axler & Ribet, 2010, pp. 3-4)

Appendix J.

$$Utilization\ = \frac{Corrected\ rental\ days}{Calendar\ days} * 100\%$$

In which

$$Corrected\ rental\ days = \frac{Rental\ days * Calendar\ days}{Max\ rental\ days} * 100$$