FORECASTING CUSTOMER SERVICE DEMAND BY MACHINE LEARNING WITH REAL LIFE IMPLEMENTATION

by
SİMGE GÜÇLÜKOL

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfilment of
the requirements for the degree of Master of Science

Sabancı University
June 2020

FORECASTING CUSTOMER SERVICE DEMAND BY MACHINE LEARNING WITH REAL LIFE IMPLEMENTATION

Approved by:

Assoc. Prof. Dr. Kemal Kılıç ....
(Thesis Supervisor)

Asst. Prof. Dr. Mahmut Ali Gökçe Mahmut Ali Gökce ....
(Thesis co-Supervisor)

Prof.Dr. Abdullah Daşçı ....

Asst. Prof. Dr. Erdinç Öner ....

Asst. Prof. Dr. Murat Kaya ....

Date of Approval: June 25, 2020

ABSTRACT

FORECASTING CUSTOMER SERVICE DEMAND BY MACHINE LEARNING WITH REAL LIFE IMPLEMENTATION

SİMGE GÜÇLÜKOL

INDUSTRIAL ENGINEERING M.Sc. THESIS, JUNE 2020

Thesis Supervisors: Assoc. Prof. Kemal Kılıç
Asst. Prof. Mahmut Ali Gökçe

Keywords: failure ratio, machine learning, time series forecasting

The decision-making process has an important role in industry and customer service delivery is one of the essential parts of this process while it has a direct impact on customer satisfaction. In the first part of the study, the objective is forecasting monthly failure ratios for a line of the products for a Turkish multinational household appliances manufacturer in order to make better planning for spare parts and service personnel and, meet customer service demand on time. The real-life obtained data set from the company include the number of installed and failed products in a given month, and installation and failure dates of products individually. The data includes more than one time-series that is in multidimensional form and each one of time series has an impact on other time series. Machine learning-based approaches were applied in order to reveal this impact and achieve better forecasting results. As the second objective of the study, comparisons between statistical-based methods and machine learning-based approaches are made. The moving average for statistical-based methods and, artificial neural network and support vector regression methods for machine learning-based approaches are compared by the model performances.

ÖZET

MAKİNE ÖĞRENMESİ İLE MÜŞTERİ HİZMET TALEBİ TAHMİNLEMESİ

SİMGE GÜÇLÜKOL

ENDÜSTRİ MÜHENDİSLİĞİ YÜKSEK LİSANS TEZİ, HAZİRAN 2020

Tez Danışmanları: Doç. Dr. Kemal Kılıç
Dr. Öğr. Üyesi Mahmut Ali Gökçe

Anahtar Kelimeler: bozulma oranı,makine öğrenmesi, zaman serisi tahminlemesi

Karar verme süreci sektörde önemli bir role sahiptir ve müşteri memnuniyeti üzerinde doğrudan etkisi olması sebebiyle müşteri hizmetleri sunumu da bu sürecin önemli parçalarından biridir. Çalışmanın ilk bölümünde yedek parça ve servis personeli için daha iyi planlama yapmak ve müşteri hizmetleri talebini zamanında karşılamak amacıyla Türkiye'de kurulan ve bir çok ülkede bulunan şubesi ile sektöründe öncü ev aletleri üreticisi için ürünlerin aylık arıza oranlarının tahminlemesi yapılmaktadır. Şirketten elde edilen gerçek veri, kurulu, arızalı ürünlerin verilen ay içerisindeki sayısını ve ürünlerin bireysel kurulum, arıza tarihlerini içerir. Veriler çok boyutlu formda birden fazla zaman serisini içerir ve her bir zaman serisinin diğer zaman serileri üzerinde etkisi vardır. Bu etkiyi ortaya çıkarmak ve daha iyi tahminleme sonuçlarını başarabilmek için bu projede makine öğrenimine dayalı yaklaşımlar uygulanmıştır. Çalışmanın ikinci hedefi olarak istatistiksel temelli yöntemler ile makine öğrenim temelli yaklaşımlar arasında karşılaştırmalar yapılmıştır. İstatistiksel tabanlı yöntem olarak hareketli ortalama, makine öğrenimi tabanlı yaklaşımlar için yapay sinir ağı ve destek vektör regresyon yöntemleri model performansları açısından karşılaştırılmıştır.

ACKNOWLEDGEMENTS

First, I wish to express my sincere appreciation to my thesis supervisor Kemal Kılıç for the continuous support throughout my master's degree. I am grateful for his precious guidance and having the chance to work with him.

I also would like to pay my special regards to my thesis co-supervisor Mahmut Ali Gökçe for his continuous support and contributions in this research. I am grateful for his precious guidance and having the chance to work with him.

I am also grateful to the jury members Abdullah Daşçı, Erdinç Öner¸ and Murat Kaya for their valuable time.

I wish to extend my special thanks to my family for their endless support, love and effort. They always trust and encouraged me. I am grateful to my sister for always being there for me as a friend. I feel so lucky and I hope you always stay with me.

I would like to thank my all friends for their supports and wonderful moments that we shared. Especially, thank you to Berkan for his love and support. Thanks to their helpfulness and sincerity, my adaptation to the new environment become easier.

to my family

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

The performance of the customer service delivery process is among the key factors that influence firm-level competitiveness in various industries, especially in home appliances, machinery, automobile, etc. Therefore, in these industries improving the customer service delivery process receives a lot of attention both from the practitioners and from the academic research community. Accurate forecasting of the customer service demand (i.e., demand due to product failures) enables better planning and decision-making, thus has a central role for improved customer satisfaction levels. As a result of the high satisfaction levels, customer continuity and profit are ensured.

There are many forecasting techniques that have been developed in order to make better decisions. Time series forecasting is an important category for various industries. Traditional statistical methods for time-series forecasting are used to predict the future by using past time-series data such as moving average, autoregressive integrated moving average (ARIMA), and exponential smoothing. On the other hand, recent technological improvements and the availability of the data led to the increasing popularity of machine learning-based approaches.

Availability of big data helps to gain more insights about the demand for the products. Better forecasts can be achieved with good preprocessing of data, a well-designed architecture of the model, and use of suitable parameters. The preprocessing of the data, more generally the whole data preparation phase is often overlooked and does not receive enough attention during data analytics studies. However, data preparation is the phase where the model is determined since, in this phase, features are engineered and determined whether they would be included in the models or not. The way one incorporates a feature to the analysis heavily dictates the performance of the machine learning-based algorithms.

1.1 Motivation

In this thesis, we focus on the customer service demand forecasting problem of a major Turkish multinational household appliances manufacturer. The company, a market leader in Turkey's appliance sector with more than 50% market share and 4500 branches domestically, also operates in more than 100 countries. In addition, they are Europe's third-largest white good manufacturer in terms of total sales. The company has production and marketing for durable goods, small home appliances, white goods (refrigerators, washing machines, etc.), and electronic products. After-sale services have a considerable share in the operations of the company. Therefore, forecasting the number of failures of products is vital for the company in order to handle service requirements in time with high customer satisfaction.

The overall objective of the thesis is in two folds. First, we will introduce a forecasting problem that a major Turkey-based multinational household appliances manufacturer is facing and propose various machine learning-based approaches as a solution. Note that, due to the special structure of the problem, which will be introduced in Section 3.1, the proposed solutions can also be considered among the novel contributions of this study.

As the second objective, we will demonstrate how different modeling approaches perform and elaborate on the effect of the differences in these modeling approaches in terms of forecasting accuracy. Note that, in the context of time series forecasting often only the previous months' realizations are used as input features for machine learning-based approaches. Inclusion of the other relevant data is not a standardized process and usually differs depending on the subjective choice of the data analyst. Thus, we believe that these comparisons would assist and guide the researchers as well as the practitioners that consider implementing machine learning-based approaches in their forecasting problem.

1.2 Outline

The rest of the thesis is organized as follows. In Chapter 2, the review of the relevant literature will be covered. In Chapter 3, the problem statement will be introduced in detail. Furthermore, machine learning and statistical-based methods that we used in the analysis will be presented in Chapter 4. In addition, the computational analysis and the discussion of the results will be provided in Chapter 5. We will finalize the thesis in Chapter 6 with our concluding remarks and future research topic suggestions.

2. LITERATURE REVIEW

In this chapter, we will review the relevant literature. Firstly, we will cover the Time Series Forecasting which is the traditional approach addressing customer service demand forecasting, even though the problem at hand is significantly different from traditional time series in various aspects. In order to address these differences, we introduced machine learning-based approaches and compared the performance with the time-series methodologies. In Section 2.1, we have done a summary of the literature related to the time series forecasting for the manufacturing sector and factors that have importance for customer service demand which is the monthly failure ratio of refrigerators in our study. In Section 2.2, we summarize the history and performances of the different machine learning and statistical-based methods from the literature.

2.1 Time Series Forecasting

Researchers have been conducted studies on time series forecasting techniques for many years. As Box et al. (2015) mentioned that many statistical-based methods are proposed since the 1940s for linear time series forecasting models. Bontempi et al. (2013) stated that in the late 1970s and beginning of the 1980s, applying linear time series methods into real applications become harder, and nonlinear time series methods are proposed to cope with them. Additionally, it has been applied in many fields such as finance (Ariyo et al., 2014), electric power systems (Amini et al., 2016), spare parts forecasting (Kim et al., 2017) and etc. Forecasting of customer service requirements is an important problem for the decision-making processes of firms. According to the Agarwal and Jayant (2019) accurate forecasting is helpful to prevent excess and shortage of inventory and increase the profit of firms.

The focus of this research is forecasting the monthly failure ratio of refrigerators in order to meet customer service demand based on a set of multidimensional time-series data. The decision for the inventory level of spare parts is made by using the failure ratio of the products in order to cover the demand for service just-in-time. In order to achieve better forecasting for spare parts, many methods have been developed in the literature by using time series data (Bacchetti and Saccani, 2012; Cavalieri et al., 2008). However, they used one-dimensional time series data in their studies. This research takes into consideration multidimensional time-series data in order to achieve failure ratio forecasting. Even tough Boylan and Syntetos (2010) proposed methods for spare parts demand forecasting by using only past data, some external factors also have an impact on failure ratio forecasting which are explained in detail later in Section 3.1.

The first external factor for failure ratio is ”installed base”. It should be taken into considerations to make better forecasting for spare parts and failure rate forecasting as stated by Jalil et al. (2011). Therefore, we performed an excessive search about ”installed base” to find out the relation between installed base and failure ratio. Many definitions were done for the installed base. According to Kim et al. (2017), the installed base can be defined as the number of sold products that are used by the customer after sales. Dekker et al. (2013) reported that characteristics of the installed base such as age, size, and location of the product are helpful to make better decisions.

Seasonality is another factor for failure ratio in many sectors and shows recurrent patterns due to some factors like weather and holidays as indicated by Zhang and Qi (2005). For example, Cankurt and Subasi (2015) stated that tourism demand data most significantly have a seasonality factor and statistical-based methods such as Autoregressive Integrated Moving Average (ARIMA) and Seasonal ARIMA (SARIMA) used for this data. In the economy, seasonality should be the first factor to take into consideration as Hylleberg (1992) said that. In order to reveal the seasonal factor, the deseasonalization method has been used in the literature (Tseng et al., 2001). According to Benkachcha et al. (2015), most time-series data have seasonal effects, and prediction of this effect should be done properly and to achieve this deseasonalization can be used before creating the model or features that related with seasonality can be added.

2.2 Machine Learning-Based Time Series Forecasting

In time series forecasting problem machine learning methods, particularly artificial neural networks have been used since the 1980s (Ahmed et al., 2010). As explained by Ahmed et al. (2010), remarkable developments have been available in the past years for understandings of models and comparisons of the performances as well. Thanks to advances in this field, in many sectors, machine learning methods were used to forecast the future. For instance, Cankurt and Subasi (2015) applied for tourism demand forecasting by using multilayer perceptron regression (MLP) and support vector regression (SVR). On the other hand, Agarwal and Jayant (2019) applied support vector machine (SVM) for the automobile industry, and Shabani et al. (2017) used SVM in the context of water distribution systems.

Many types of research conducted by using different machine learning methods and statistical-based methods. For example, Sharda and Patil (1992) has made comparisons between the neural network and the Box-Jenkins method by using M-competition time series data. Callen et al. (1996) have applied linear time series models in order to compare with neural network models. Although the majority of them have been done in order to compare classification models (Caruana and Niculescu-Mizil (2006)), there are a limited number of studies related to the regression models (Ahmed et al., 2010). These studies conducted with one-dimensional time series data. However, in this research, we have done comparisons for multi-dimensional time-series data.

There is an ongoing debate on whether machine learning methods outperform the others in time series forecasting. Makridakis et al. (2018b) indicates that the aims of machine learning and statistical methods are the same and they both try to forecast the future by minimizing the loss function but machine learning methods required more computational power. Besides, Carbonneau et al. (2008) found that although machine learning-based approaches perform better, in their analysis the difference in terms of the forecasting accuracy was not statistically significant. They conclude that such a difference in terms of forecasting performance does not change the fact that the simplicity of the implementation of traditional models makes them still the more attractive choice.

Even though, the machine learning-based approaches are becoming to receive more attention from the practitioners as well as the scientific community, in a study

(Makridakis et al., 2018a), which compares the performance of machine learning approaches with traditional forecasting techniques in a data set that consists of 1045 time series cases, the traditional forecasting techniques outperform the machine learning-based approaches. According to these studies, the machine learning-based approaches also suffer from the computational requirements when compared to the traditional time series approaches. Yet, in the same study, the authors still consider machine learning-based approaches as a viable alternative for forecasting. However they suggest that the learning process should be improved, the data might be preprocessed for the learning phase (such as using deseasonalized data instead of the raw data), or hybrid usage of the techniques might be possible ways to improve the performance of the machine learning-based approaches in forecasting.

Qiu et al. (2014) argued that artificial neural networks (ANN), support vector regression (SVR), and other machine learning methods can be used for time series forecasting, regression, and classification instead of statistics-based methods. In addition to this, support vector regression is a successful method in order to find a globally optimum solution. According to the Bontempi et al. (2013), models that are used historical data to predict dependency of past and future are called non-parametric nonlinear models and machine learning models are examples of them. Besides, Lapedes and Farber (1987) showed that artificial neural networks have better performances than statistical ones with increasing attention to the machine learning methods. Furthermore, Hill et al. (1996) reported that neural network models have better performance than traditional methods, they explained the reason for this situation as the ability of neural networks to handle with discontinuity.

There is a gap in the literature for the implementation of machine learning-based approaches. Artificial neural networks implemented by using past data as inputs at many studies (Khashei and Bijari, 2011; Zhang, 2001). However, few of them included external factors (seasonality, installed base, etc.) (Hamzaçebi, 2008). Therefore, the usages of these factors might increase the model performances. In this study, we have preprocessed data in order to include these factors instead of using past data as input that are explained in detail in Section 4.3.

3. PROBLEM DEFINITION

In this chapter, we will present the details of the research problems we considered in this thesis. Firstly, we will discuss the failure ratio forecasting problem that the company faces. Various difficulties of customer service demand forecasting for refrigerators and external factors that affect the monthly failure ratio of products will be introduced in the next section. On the other hand, since determining the role of modeling in the performance of machine learning applications in the context of forecasting is also an important objective of our research, we will discuss various problems that should be taken into consideration while implementing machine learning methods in the context of forecasting in the second part of this chapter.

3.1 Failure Ratio Forecasting

In this thesis, one of the objectives is achieving an accurate forecast of customer service demand (i.e., monthly service requirements) for a major multinational home appliances manufacturer headquartered in Turkey. The company needs to predict the monthly service requirements in order to plan the workforce, spare parts, and space requirements for the upcoming months. They also use the forecasts (actually, variations from the forecasts) as an indicator (i.e., flag) for various quality problems, thus use the model as part of their early warning system as well. Each product has a production (i.e., assembly) date. The products are not sold right away and kept in the inventory of the manufacturer or the retail stores before the customers buy them. This takes usually a few months but in some cases, they might stay in the stocks over a year. Most of the products that are sold to the customers are installed within a few days. Nevertheless, some of the items are kept by the customers (even in some exceptional cases for 10+ years) before they are installed. The sales date and the installation dates for the products are also available. Finally, the service requirements dates due to the failure of the products are also available.

There are various factors that influence the monthly service requirements. First of all, due to the continuous improvement programs in the company, there are some changes in the production lines related to the production methods. Also, the parts used during the assembly of the end-product change over time. As a result, the production batch (i.e., which month the appliance is produced) influences the likeliness and timing of future failures. That is why "more than expected" failures of a particular batch might serve as an early warning for the quality department, When a flag is raised due to such cases, quality department jumps in the boat and addresses the root cause of the problem and eliminates it in order to avoid future failures before it spreads to more customers.

Secondly, the age of the appliance in the installed base also influences the service requirement. It is expected that when the appliance gets older, it has a higher tendency to break. However, the relationship between the failures and the age is more complex. Mechanical parts tend to fail later than the electronic parts, which fail (if they ever fail!) within a few months after installation. The age of the appliance is considered to be the total time that passed since the appliance is installed (i.e., set up) because it is the actual duration that the appliance is working. Note that the age with respect to the production date is already considered (indirectly) as the first factor (production month) explained above. As Figure 3.1 shows that there can be a positive correlation between the age of the product and service requirement quantity. According to the figure, initial months also have higher failure ratios since there can be production-related failures and it is noticed in the first months. However, age-related failures increase, when the products become older.

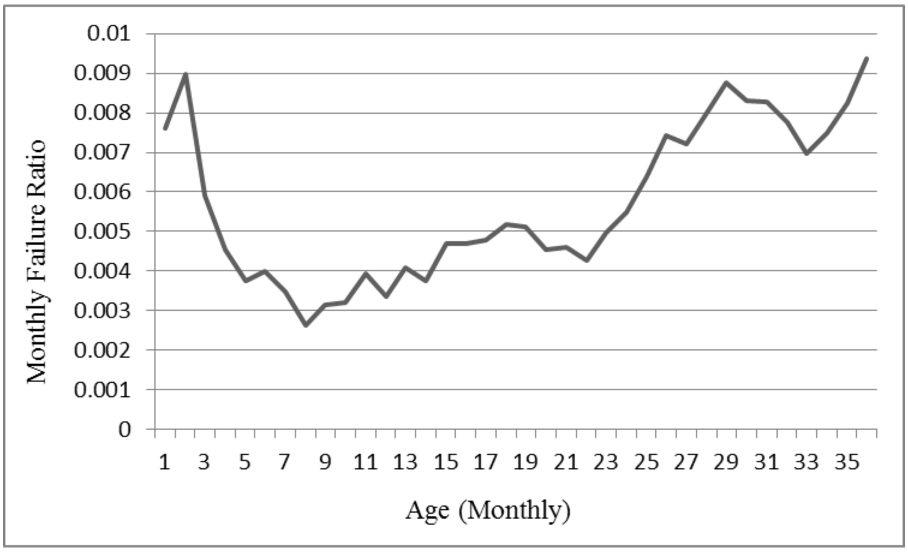


Figure 3.1 Monthly failure ratio for 36 months that start from July 2013

Thirdly, seasonality has an impact on failure rates. For example, refrigerators have an increased tendency to fail during the summer months due to the higher difference between the internal and the external temperature and the additional load this difference creates on the appliance. Likewise, air-conditioning units also utilized more during the summer months. Therefore they have a higher failure rate during the summer months. On the other hand, for the washing machines and dishwashers, the cold-water temperatures create extra problems thus the colder winter months, their failure rates increase. Furthermore, appliances can be purchased by the customers for their summer houses. In this situation, although the appliance fails during any time of the year, the customers do not realize the failure before they return to their summer house during summers. Figure 3.2 demonstrates increased failure rate at the summer months.

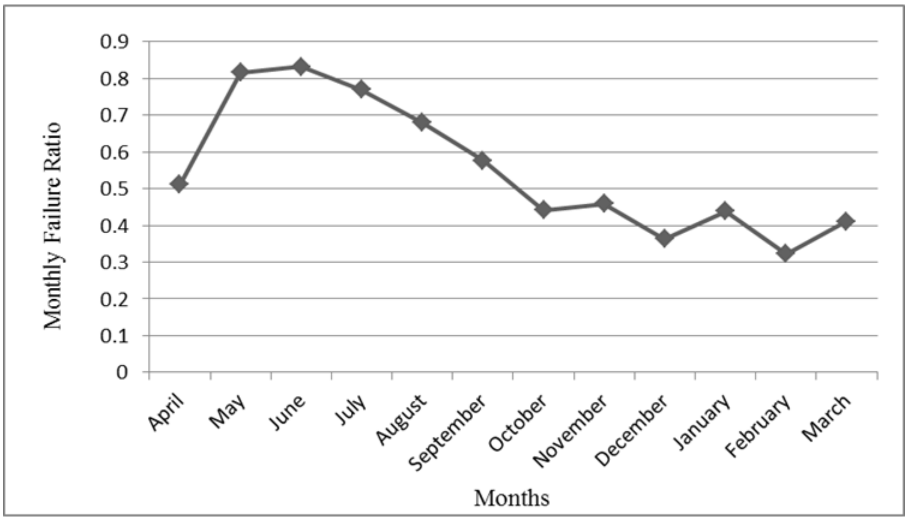


Figure 3.2 Monthly failure ratio for 12 months that start from April 2013

Finally, the monthly production quantities can also have a high impact on the failure ratio of the appliance as Porteus (1986) stated that. When the lot sizes increase, the possibility of the defect increases as well. Therefore we believe that higher production quantities can be led to higher failure ratios of refrigerators. When the production quantities increase for a month, there can be more chances of errors due to pace. As a result, increasing production quantities can be another reason for higher failure rates for the products.

3.2 Machine Learning Methods

Machine learning-based approaches such as an artificial neural network (ANN), and support vector regression (SVR) are used for many forecasting models. However, as we mentioned earlier in Section 2.2, there is a debate on whether statistical methods or machine learning methods give better results. There are three main issues that affect the performance of machine learning-based approaches.

Initially, choosing a suitable method for the forecasting model is important. There are many machine learning-based approaches and by classifying them according to their usages, we can find a proper method for the model. To achieve this, understanding the problem is the most important part of the projects. There are supervised and unsupervised machine learning methods that they divided according to their outputs. If output values are known, supervised machine learning methods such as linear regression, decision tree, support vector regression, etc. should be used. Otherwise unsupervised learning methods such as hierarchical clustering, principal component analysis (PCA), etc. can be used. After deciding on the type of method, according to the structure of the problem, we should decide whether regression or classification methods should be used. For the cases when the output is nominal or ordinal scaled the problem becomes a classification problem, when the output is interval or ratio scaled the problem becomes a regression problem. Thus, generally speaking, forecasting is a regression problem in the context of machine learning terminology where the output in our research is a ratio scale failure rate.

Another important issue that arises from machine learning methods is deciding on the architecture of the method that will be used. Parameter tuning is defined as usage of different parameters that may affect the performance of the method (in this case, kernel functions and architecture). By using different parameters, kernel functions, or architecture, we can get better performances with the machine learning methods. Parameter tuning is an important part of the modeling and if conducted with proper training and test data splitting prevents the model from overfitting or underfitting. Additionally, suitable combinations of parameters help to increase quality of results. For example, we need to determine the learning rate, number of layers, and the number of input nodes for an artificial neural network. We can create a set of numbers for these parameters and we compare the performance measure results of models. While creating a model, cross-validation methods can be used to keep the model under control.

In addition, data preparation has a significant role in the model performances. We obtain useful information from data with the preparation which encompasses data visualization, feature engineering as well as data cleaning. Therefore, exploring the data and understanding the nature of the problem are significant issues in order to attain a model that performs well. Generally speaking, machine learning techniques determine patterns i.e., relations between the input features and output that is hidden inside the data set. If the features that constitute such relations are missing, none of the machine learning alternatives can recognize a pattern. On the other hand, incorporating irrelevant features together with the relevant ones creates noisy data in which pattern recognition becomes an extremely difficult task. That is to say, not only missing data but also an abundance of data leads to uncertainty, and the hidden relations in the data cannot be identified. At this stage, the domain expertise becomes extremely crucial to improve the modeling performance of the machine learning methodology. One needs to choose carefully the input features that will be included in the modeling stage in the data preparation stage. Sometimes rather than the readily available features, some features can be constructed from them (e.g., squares, powers, ratios, differences, etc.) should be used in the modeling phase. This process is known as feature engineering and at the end of the date, directly dictates the performance of the machine learning method.

Data cleaning is another process that should be conducted with care during the data preparation stage. There can be some missing values in the data set. One approach for such cases would be the elimination of the data row vector, or the feature column vector altogether. However, elimination may result in the loss of significant information, i.e., the relations hidden inside the data. An alternative to elimination would be the replacement of the missing values with substituted values. This process is referred to as the imputation of the missing data. Various imputation techniques are available in the literature. One simple example is replacing the missing values with the average of the data. Generally speaking, there are two categories of imputation, namely, single and multiple imputations. In a multiple imputation approach, multiple numbers of data sets are obtained and the analysis is conducted on these multiple data sets. Other than substituting the missing value with the mean, one can also substitute it with the most frequent value or determine similar records with the record where there is missing value and replace the missing value with the input from those similar records. The similarity among the records can be measured based on simple techniques such as k-nearest neighbors. Again the handling of the missing values would directly influence the model performance.

When there is a string or categorical values among the input features, in order to conduct the machine learning process, it might be preferred to convert them to numerical values by means of various encoding methods. In the forecasting problem that we focus on this research, most of the features are already numerical. However, as we have stated earlier, seasonality is a factor that influences the failure rates. One can leave the seasons as binary categorical values (i.e., four seasons and values ϵ 0,1) or can allow degradation among the values so that they can be real numbers from [0,1]. These two different encoding alternatives, as well as some others, might influence the modeling performance.

In addition, there should be sufficient data in order to prevent overfitting and underfitting of the model. If necessary, by using oversampling methods, the size of data can be increased, or by using feature transformation methods, inputs of the model increase as well. The number of training and testing data changes the performance of the model. If there are few data for training, the learning process of the model becomes harder, because of this, model performances decrease.

To sum up, the data preparation stage which consists of the feature engineering process, handling the missing values, encoding, and training set-test set splitting all are crucial steps and directly influences the modeling process. Different preferences yield different modeling performance. However, generally speaking, this stage of data analytics usually overlooked, and not enough attention is given. One major reason for this is the lack of domain expertise in the data analytics team. In this research, we also addressed the effect of various data preparation alternatives in the context of forecasting problems and provide some insights for the researchers and practitioners for the future.

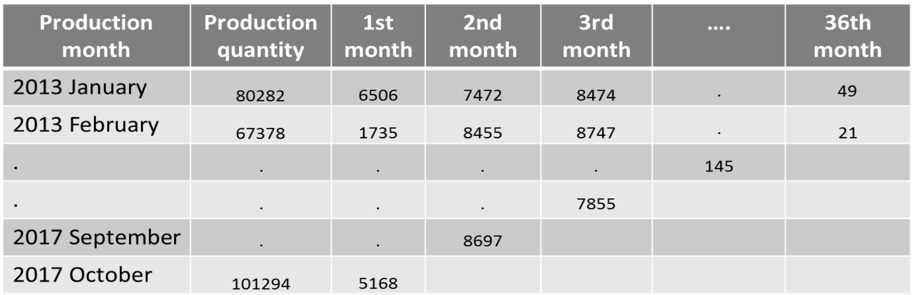
4. METHODS and EXPERIMENTAL ANALYSIS

The household appliances manufacturer produces various different products such as refrigerators, washing machines, dishwasher machines, ovens, air conditioning units, TV sets, etc. In this thesis, we focus only on the forecasting of the customer service demand for refrigerators. However, the proposed techniques can be applied to other types of appliances as well without much change.

One of the main objectives of the research is to compare the performance of the statistical-based approaches with the machine learning-based approaches in the context of forecasting. In an earlier study, the manufacturer worked together with a management-consulting firm in order to determine an appropriate time series analysis technique that they can use to forecast the monthly customer service demands. As a result of that extensive analysis, they have decided to use moving averages for their forecasts, not only because it achieved one of the most accurate forecasts but also because of its simplicity. That is to say, the "as-is" methodology in the company for the forecasting problem is moving average. Therefore, we limited our attention to the moving average method as among the statistical-based approaches for comparison purposes. For machine learning-based approaches, we used an artificial neural network and support vector regression method with different feature sets because these methods do not result in bad performances (Lin et al., 2007; Zhang et al., 1998).

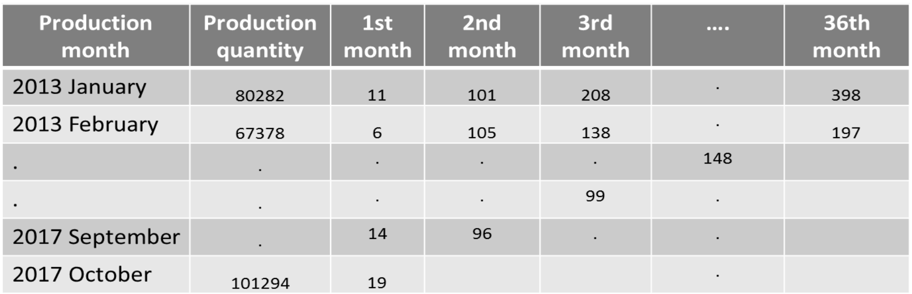
4.1 Data Set

The raw data set includes the production date, sales date, installation date, failure date of each refrigerator as well as various other details regarding the installation and services provided during the failures. Note that, the focus of the company is limited to the warranty period, which starts after the sales of the product and typically only 3 years (i.e., 36 months). By using the raw data set corresponding to the individual products, the company creates two data matrices that depict how many refrigerators produced at each month is installed (i.e., installation data matrix) and failed (i.e., failure rate data matrix) during the time horizon. Initially, we were provided with the two data matrices the company has created from January 2013 to April 2015 (i.e., each row in the data matrices corresponds to one of the months between this time horizon). However, the raw data set for individual products (i.e., each refrigerator produced) is available from January 2013 to October 2017. The raw data corresponds to approximately 4.4 million products installed during this period. We recreated the two matrices (i.e., installation and failure rate) from January 2013 to October 2017 by using the raw data set. Again each row corresponds to the time series for the production that month. In the installation matrix, the entries of the rows correspond to the increase in the installed base at consecutive months, on the other hand in the failure matrix the entries correspond to the total number of failures for that month. That is to say, the installation matrix depicts the cumulative amounts for the installed base but failure matrix does not contain cumulative amounts. Representative figures of matrices are shown at Figure 4.1 and 4.2. However, please note that the figures are presented merely for illustrative purposes and the data provided in the figure is fictitious and created by the authors.



Production month	Production quantity	1st month	2nd month	3rd month	....	36th month
2013 January	80282	6506	7472	8474	.	49
2013 February	67378	1735	8455	8747	.	21
.	.	.	.	.	145	
.	.	.	.	7855		
2017 September	.	.	8697			
2017 October	101294	5168				

Figure 4.1 Installation Matrix



Production month	Production quantity	1st month	2nd month	3rd month	....	36th month
2013 January	80282	11	101	208	.	398
2013 February	67378	6	105	138	.	197
.	.	.	.	.	148	
.	.	.	.	99	.	
2017 September	.	14	96	.	.	
2017 October	101294	19			.	

Figure 4.2 Failure Matrix

Note that, the data is not a typical time-series but rather a time-matrix data. Each row in the data set corresponds to an individual time series. On the other hand, the rows are not independent time series because they refer to refrigerators that are produced in different months but face the same demand and failure environment. That is to say, the demand, as well as a failure across the rows, are interrelated with each other. The rows correspond to the production batch, the columns in the matrices provide the information corresponding to the installed base as well as the age of the products. On the other hand, seasonality information is hidden in the diagonals. A typical time series analysis would treat each row as an independent vector, i.e., carrying out the time series analysis individually, and combine the results later to obtain the final model. However, in this case, the whole contains much more than the sum of the individual parts. Therefore, typical conventional time-series approaches fail to incorporate the whole information provided by the data set. That's one of the major reasons why the earlier analysis conducted by the management consulting company yielded a relatively very simple moving average technique as the best choice among the time series methods.

After the installation and failure rate matrices are constructed from the raw data set, we found a monthly failure ratio by dividing the monthly number of failed products into cumulative installed products. Figure 4.3 depicts the steps of data preparation.

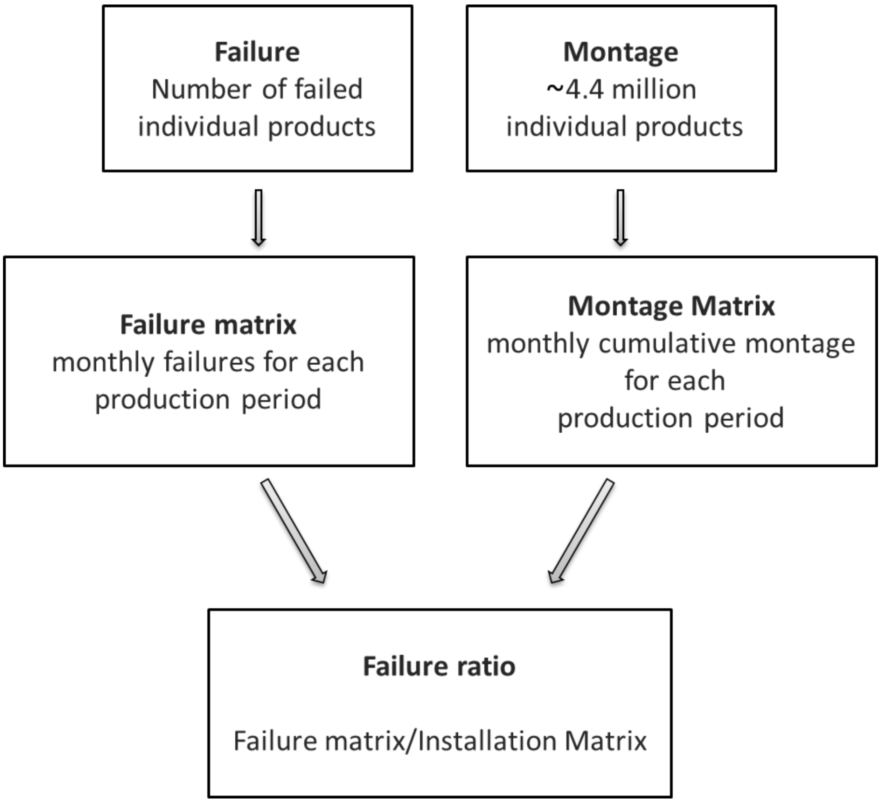
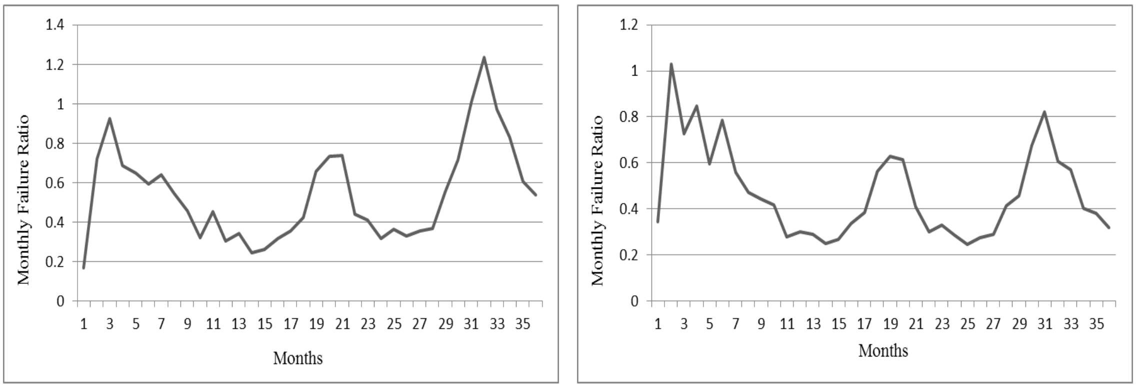

Failure
Number of failed individual products
Montage
~4.4 million individual products
Failure matrix
monthly failures for each production period
Montage Matrix
monthly cumulative montage for each production period
Failure ratio
Failure matrix/Installation Matrix

Figure 4.3 Time Series Failure Ratio Calculation Steps

4.2 Moving Average

Moving average is a statistical method used for stationary data and it takes the averages of the last specific number of observations and finds the predicted value of the future. This specific number can be is referred to as the time window. If the time window is 3, the average of the last 3 observations is used as a forecasted number. It uses time series data and generally results in good performances (Hansun, 2013) so that it is a popular technique and widely used in practice.

Recall that we have matrix data with 58 rows. The first 23 rows of them have 36 columns for the monthly failure ratio. After 23 rows, columns decrease one by one up to remain one column in the last row. As we previously mentioned, each row represents time-series data. We know that stationary time series data is necessary for the moving average method. On the other hand, Figure 4.4 shows that each time series has a seasonality factor. Therefore, we have nonstationary data. In order to remove the seasonal factor, we calculated averages of 12 months. In addition, we divided each data into corresponding averages and obtained the deseasonalized data.



(a) Time series start with January 2013 (b) Time series start with February 2013

Figure 4.4 Monthly failure ratio for 36 months for 2 different time series

We implemented the moving average method to deseasonalized matrix data. From each time series, we took the last 7 months of each time series and we have done moving average method into this data. Furthermore, we implemented this method with 3, 5,7,10,13,15, and 20 periods. Results are explained in Chapter 5.

4.3 Artificial Neural Network

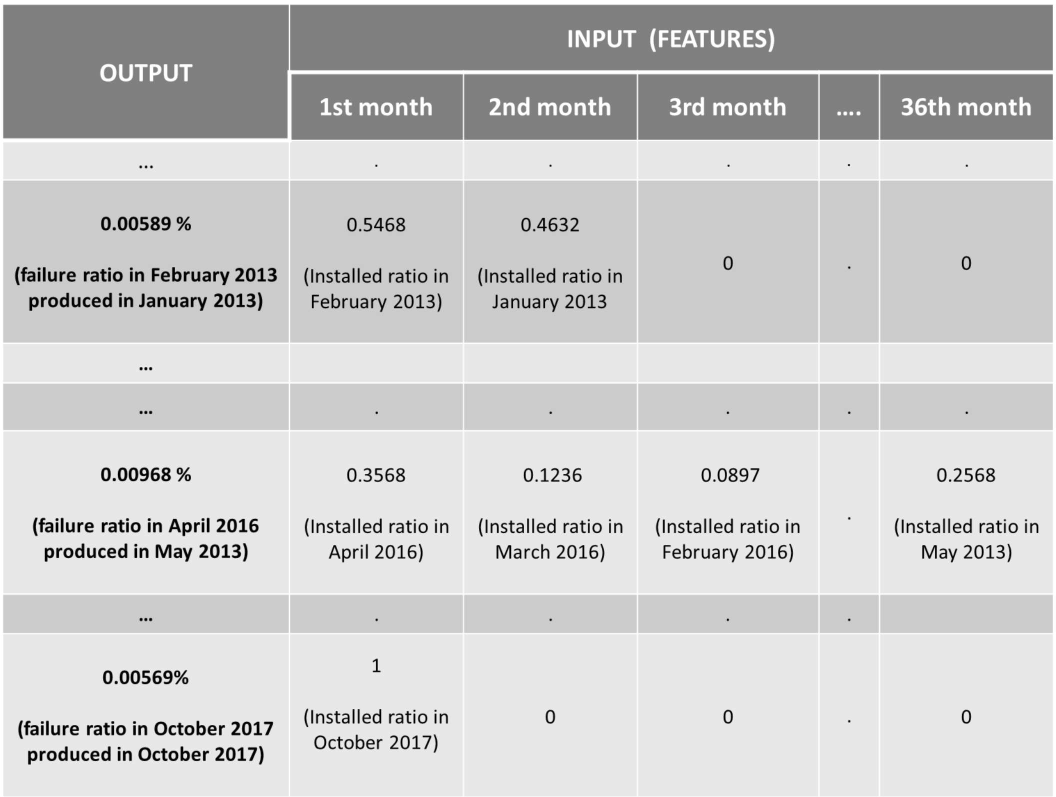
An artificial neural network is a system that imitates the work of the human brain. It has neurons as input and by using kernel functions it takes input from the input layer and converts them into usable knowledge at the hidden layer. The output layer uses this information and kernel function later creates an output of the model. As a result of this process, ANN takes the input feature values and maps them to an output. The universal approximation theorem (Csáji et al., 2001) states that ANNs with single hidden layers can approximate arbitrary continuous functions given appropriate parameters. There can also be more than one hidden layer at the model. Thanks to this multilayer structure, ANNs are determined to have a high predictive capability for many complex problems (Tseng et al., 2002). In order to make good predictions, the kernel function for each layer should be chosen properly.

As we mentioned earlier in Section 3.1, age, production quantity, and seasonality affects the failure rates in the data set. Recall that as the number of months after installation increases, the number of failures generally increases. However, other factors such as installed base recording and seasonality factors also influence the failure rates. Secondly, the production quantity has an impact on the data. When the production quantity increase, the possibility of the failure can be increase because of the high production at a limited time. Lastly, the seasonal factor has an impact on the data and especially, failure ratios of products are increasing in summer months. Therefore, we required to take into consideration these effects in the neural network model as well.

The output of the model is the percentage of monthly failure ratio of products that are produced in a specific production month that is included in the failure ratio matrix. That is to say, during the data preparation stage, for each pair (production month, the month of interest), one feature vector and output are constructed. For instance, in March 2013 (i.e., the month of interest), the failure ratio of products produced in February 2013 is 1.03% (i.e., the output of the corresponding feature vector). This means that 1.03 percent of the installed products from February 2013 to March 2013 from the batch of refrigerators produced in February 2013 broke down.

We created three groups of feature (i.e., input) that reveals external factors for the model. The first group is created for installed base and age effects. The preparation process of these features is explained below in detail:

- The cumulative installation data (i.e., installed base) for each production month (batch) for the following 36 months is available.
- A failure ratio is calculated (i.e., the output is determined) for each pair of production month and failure months (i.e., the month of interest). They are given in Figure 4.5 in output column.
- As an example, we take the February 2013 failure ratio of products produced in January 2013. Second row of the Figure 4.5 represents the example.
- For each output, we create a total of 36 inputs, because the warranty period is 36 months and each input corresponds to the one month of the last 36 months up to failure month.
- From production months to failure month, we count the number of months (we have 2 months in this example).
- As we have two months, we create two ratios as features and the remaining 34 features are 0.
- We divided the number of installed products in February 2013 into cumulative installed products from production month that is January 2013 to failed month that is February 2013. This ratio is our first input that given at second row of Figure 4.5 and ”1st month” column.
- As the first feature, we found the second feature with the same logic, but we divided the number of installed products from January 2013 into cumulative installed products up to February 2013.
- For each output, we found all input with this algorithm.



OUTPUT	INPUT (FEATURES)				
	1st month	2nd month	3rd month	….	36th month
…	.	.	.	.	.
0.00589 % **(failure ratio in February 2013 produced in January 2013)**	0.5468 (Installed ratio in February 2013)	0.4632 (Installed ratio in January 2013	0	.	0
…					
…	.	.	.	.	.
0.00968 % **(failure ratio in April 2016 produced in May 2013)**	0.3568 (Installed ratio in April 2016)	0.1236 (Installed ratio in March 2016)	0.0897 (Installed ratio in February 2016)	.	0.2568 (Installed ratio in May 2013)
…	.	.	.	.	
0.00569% **(failure ratio in October 2017 produced in October 2017)**	1 (Installed ratio in October 2017)	0	0	.	0

Figure 4.5 Example for the Installed-base features of the neural network models

The second group of features is created in order to show the effects of monthly production quantities in our data. To include this effect, we converted quantities into categorical features by using encoding methods which are explained in Section 5. If the monthly number of quantities is less than 70000 products that is equal to lower quartile of available production quantity data, we labelled them as "Low", if they are more than 100000 that is upper quartile, we labelled them as "High" and otherwise, they are labelled as "Normal". Therefore we have 3 categories for production quantity in our models.

Besides, as we mentioned in Section 3, we had a seasonal factor. To include this factor, we tried two different methods. As the first method, we used deseasonalized data for NN1 and SVR1 models. We found monthly averages of outputs for each month (12 months) and deduced the monthly averages from the whole data set in order to deseasonalize the data. The monthly failure rate averages are provided in Table 4.1. These averages are the seasonal factor of the data. We applied our models by using deseasonalized output. We predicted deseasonalized output with machine learning models and we converted them seasonalized form by multiplying representative month averages.

Month	January	February	March	April	May	June	July	August	September	October	November	December
Average	0.3723	0.3500	0.3761	0.3758	0.4294	0.5173	0.6366	0.6715	0.5327	0.4670	0.4537	0.3761

Table 4.1 Averages of percentage of monthly failure ratio

The second method that we used for seasonal factors is adding seasonality features into the model. Instead of using deseasonalization, we used output directly, but we added four new features as seasonality features that correspond to four seasons. We used fuzziness to create seasonality features. We considered that each month can be included in more than one season (i.e., in two subsequent seasons) to a degree. Thus, we did not assign each month to one season but rather assigned a degree of membership for each month to a season, i.e. represented the seasons with fuzzy sets. Figure 4.6 illustrates how we constructed the seasonality feature of the model. For instance, when we look at the March which means that we predict monthly failure ratio of March, for winter feature it takes value 0.25 and for Spring it takes 0.75. On the other hand, its membership degree to summer and fall is 0. When we sum all seasons for each month, it is going to be 1. We added four new features that represent each season in our models.

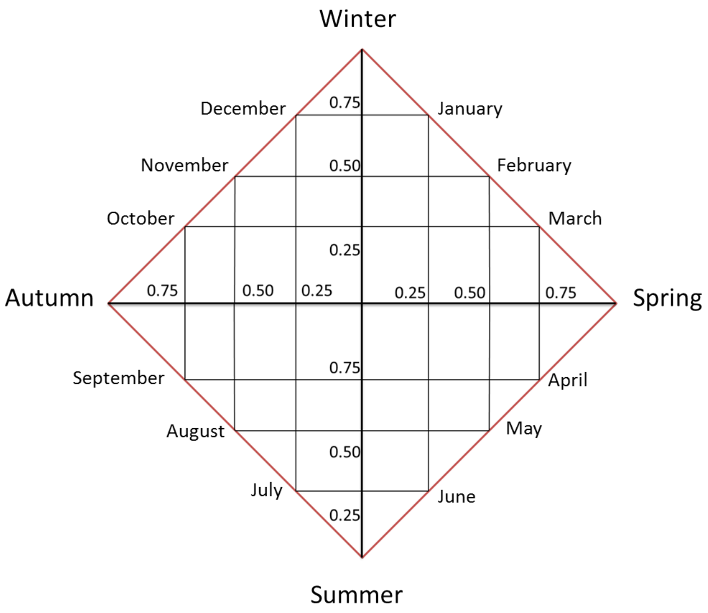


Figure 4.6 Illustration of Seasonality Feature

The neural network model was created after the data was pre-processed. We have two artificial neural network models which are called NN1 and NN2 models. As we mentioned previously, the only difference between them is the implementation of seasonality factors. Details of these models are given in Section 4.3.1 and 4.3.2.

4.3.1 NN1 model

NN1 is the model that considers the deseasonalized data. Thus seasonality is not incorporated directly into the data. As a result, there are 36 features in order to reveal the age effect from data and 3 features that represents the number of production quantities. As a result, a total of 39 features are available for the NN1 model. We created the NN1 model with two options. The first NN1 model created by using 39 features by adding production quantity features. Second NN1 model includes only 36 features represents the age. We had 3 layers at both models which consist of one input layer, one hidden layer, and one output layer. We had 39 inputs that each feature represented by one input for the first NN1 model and 36 inputs for the second NN1 model. Figure 4.7 illustrates the architecture of the first NN1 model. After that, we had 10 nodes at the hidden layer for both of them. In addition, activation functions for both layers were linear. More information will be given in Section 5.

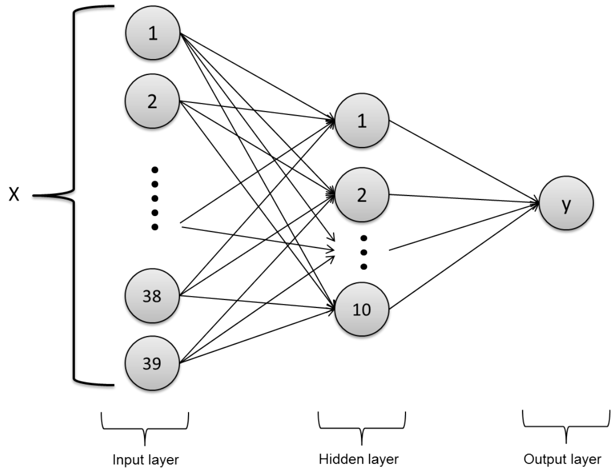


Figure 4.7 Neural Network Architecture for NN1 model with 39 features

4.3.2 NN2 model

NN2 model created by using 36 features for age effect, 4 features for seasonality effect which are given at 4.2. and 3 features for the production quantity effect. As a result, there are 43 features for NN2 model. As we explained for the NN1 model, we applied this model for two options. For the first NN2 model, we included production quantity features and for the second we did not add these features and we compared model performances for these options. It has the same neural network architecture with the NN1 model which is explained in Section 4.3.1. The only difference is the number of inputs for all models. The number of inputs for each model is the same as the number of features used for each model.

Month	Winter	Spring	Summer	Autumn
January	0.75	0.25	0	0
February	0.50	0.50	0	0
March	0.25	0.75	0	0
April	0	0.75	0.25	0
May	0	0.50	0.50	0
June	0	0.25	0.75	0
July	0	0	0.75	0.25
August	0	0	0.50	0.50
September	0	0	0.25	0.75
October	0.25	0	0	0.75
November	0.50	0	0	0.50
December	0.75	0	0	0.25

Table 4.2 Seasonaliy feature values for each month

4.4 Support Vector Regression

Support vector regression is a supervised learning technique used to predict regression models. It has similar logic with the support vector machine used for classification methods. There are four key concepts for support vector regression. These are the kernel, the hyper-plane, the boundary line, and the support vectors. Kernel functions are used to convert low dimensional data into higher-dimensional data. A hyper-plane is a line that helps in order to predict continuous outputs. Boundary lines are used to find a margin. A hyper-plane is available in the middle of two boundary lines. Aim of support vector regression is minimizing the margin, therefore they are important terms for the model. Support vectors are the closest data points to the boundary lines.

In our support vector model, we again used two different approaches regarding the inclusion of the seasonality to the model, in which the first approach corresponds to the utilization of the deseasonalized data and the second approach directly incorporates the seasonality as an input feature to the model. As a result, two different feature sets were created for the two models, namely, SVR1 and SVR2, respectively. These models are explained in Section 4.4.1 and 4.4.2 in more detail.

4.4.1 SVR1 model

Input and output data that used for SVR1 are the same as the NN1 model. The only difference is instead of an artificial neural network, we applied support vector regression. In order to decide the parameters of SVR1 model, we applied Grid Search cross-validation by using `GridSearchCV()` function of `Sckitlearn` library for Python. This function takes a set of parameters and applies the grid search algorithm. By comparing the results of the scoring functions such as MSE (mean squared error), MAPE (mean absolute percentage error), etc. returns the best model with the best parameters. Set of parameters that were considered in the analysis after some test iterations are provided in Table 4.3. We used 5 fold cross-validation for grid search and MAPE as the scoring function.

Parameters	Values			
kernel	rbf		polynomial	
degree	4		5	
C	0.001	0.1	1	10
epsilon	0.00001	0.0001	0.001	0.01
gamma	scale	auto	0.1	0.5

Table 4.3 Parameters for parameter tuning of SVR models

As a result of grid search cross validation best parameters for SVR1 model with all 39 features were determined as the polynomial kernel function, regularization parameter "C" as 0.1, epsilon as 0.01, gamma as the scale which equals 1/(number of features * variance of inputs) and the degree for polynomial kernel as 5. If we did not add production quantity features into the model, the best model was found as rbf kernel function with C is 0.01, epsilon is 0.00001, gamma is a scale and since that is rbf kernel, we do not need degree value.

4.4.2 SVR2 model

SVR2 model was implemented with the same input and output data with the NN2 model. In order to create support vector regression model, by using same set of parameters with SVR1 model that are given at Table 4.3, we applied `GridSearchCV()` function of `Sckitlearn` library for Python. A similar parameter identification process was applied as discussed in the previous section. Table 4.4 shows the best model parameters for the SVR2 model with 40 features and with 43 features. As Table indicates that only kernel functions of models are different. Details of model performance measurements are given in Section 5.3.

Model	Kernel	C	epsilon	gamma	degree
SVR2_1	polynomial	0.1	0.001	scale	5
SVR2_2	rbf	0.1	0.001	scale	4

Table 4.4 Parameters that used for SVR2 models

5. RESULTS and DISCUSSIONS

In this chapter, we present computational results of the model and we make a comparison of models by using selected performance measures. Firstly, results of the moving average method is presented in Section 5.1 and neural network model and support vector regression model are explained in Section 5.2 and 5.3.

We conducted our experiments on the computer which is running on Windows 8.1 with Intel(R) Core(TM) i7-4700HQ CPU @ 2.4 GHz and 16 GB RAM. We apply Python programming language and software version 3.7.4 in Spyder IDE with version 3.3.6. First of all, the moving average method was applied via Python and we implemented code for moving average by defining new function ”Moving Average” instead of using developed packages. Moreover, we used `sckitlearn` package of the Python (Pedregosa et al., 2011) in order to use machine learning method functions. To create neural network models, we imported `keras` library (Chollet et al., 2015) . Neural network models were run by using `Tensorflow` backend (Abadi et al., 2015). In addition, while doing preprocessing of data, `LabelEncoder and Onehotencoder` function of the `sckitlearn.preprocessing` package was used since the ”Production Quantity” feature of the model was converted into 3 features that each of them represents one category. Lastly, for Support Vector Regression model, same steps were done for preprocessing and to create model `SVR` function from `sckitlearn.svm` (Chang and Lin, 2011) was used.

Performance measures that we used for our models are ”Mean Absolute Percentage Error” (MAPE), ”Root Mean Square Error” (RMSE), ”Mean Absolute Error” (MAE), ”R squared" and ”Adjusted R squared”. Each performance measure calculations are explained. For each formula, we used following abbreviations;

y_i=i th actual output value

$\hat{y_i}$=i th predicted output value

$\bar{y}$=average of actual output value

n=number of output value

Mean Absolute Percentage Error (MAPE): MAPE calculates the ratio that differences of actual and predicted output values and divided into actual values. It takes the average of these ratios for each output value and by multiplying with 100, it finds the percentage of the error. It is the most common measurement for forecasting problems especially if there is no outlier. A higher value indicates a worse model.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{(y_i - \hat{y_i})}{y_i}\right| * 100$$

Mean Absolute Error (MAE): Calculation of MAE gives that average of the absolute value of differences between predicted and actual output values. The sensitivity for outliers is low for MAE. If MAE is high, the model can be worse due to high error.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y_i}|$$

Root Mean Square Error (RMSE): It is the square root of the mean square error which calculates the average of the squared error of each output with prediction values. It cannot take a negative value because of the square root. Higher RMSE means higher error and worse model.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y_i} - y_i)^2}$$

R Squared (R^2): R^2 value is calculated as the ratio of the MSE value of two models. We divided the MSE of our model into the MSE of the baseline model. The baseline model means that we forecast every output value as the average of the actual value which is the simplest prediction method. Moreover, we subtract this ratio from 1. R^2 can take a value between $-\infty$ and 1. If it is 1 or closest to 1, the model is created in a good way. If R^2 is negative, the model is worse.

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

Adjusted R Squared (R^2): Adjusted R squared is calculated by using R squared value. Adjusted R squared value consider the correlation of each feature with the output value, although R squared only look into the proportion of variation in output value. Therefore the increasing number of features can be increase R squared value but not adjusted R squared value. Adjusted R square value is always less than R squared value and a higher value means that better prediction performance for the model.

$$R^2_{adjusted} = 1 - \frac{(1 - R^2)(n-1)}{(n-p-1)}$$

where n=number of data points and m=number of features

5.1 Moving Average

In this section, we experimented with moving average with 3,5,7,10,13,15 and 20 window lengths. We have 1494 data from 58 time- series that 23 of them have 36 months and remaining are decreasing one by one. We used the last 7 observations for each time series. Therefore we cannot use all time series data in order to make predictions. For 3 window lengths, each time series should have at least 10 months of data. According to this logic, when we change the window length, the number of predicted data decreased.

Results of performance measures are given in Table 5.1. According to the results, at time window 10, all performance measurement values reach their optimal. Therefore, we took a moving average with time length 10 in order to make comparisons with machine learning-based approaches. Besides, graphs for performance measures are available in the Appendix.

Window Length	MAPE(%)	MAE	RMSE	R^2
3	26.2198	0.1476	0.2377	0.1028
5	23.1975	0.1280	0.1931	0.4277
7	21.7259	0.1201	0.1849	0.4945
10	20.1020	0.1137	0.1837	0.5219
13	20.8678	0.1196	0.1943	0.4907
15	22.9085	0.1320	0.2106	0.4228
20	23.1851	0.1460	0.2350	0.3459

Table 5.1 Performance Measures for Moving Average Method with Different Window Lengths.

There are 287 predicted values that are taken from matrix time-series data. We took the last 7 observations as an actual value from each one of the 41 different time-series and we sorted them in ascending order. Figure 5.1 illustrates the actual and predicted values of these observations that are predicted with time window length 10. Actual values with blue color and predicted are in orange colors.

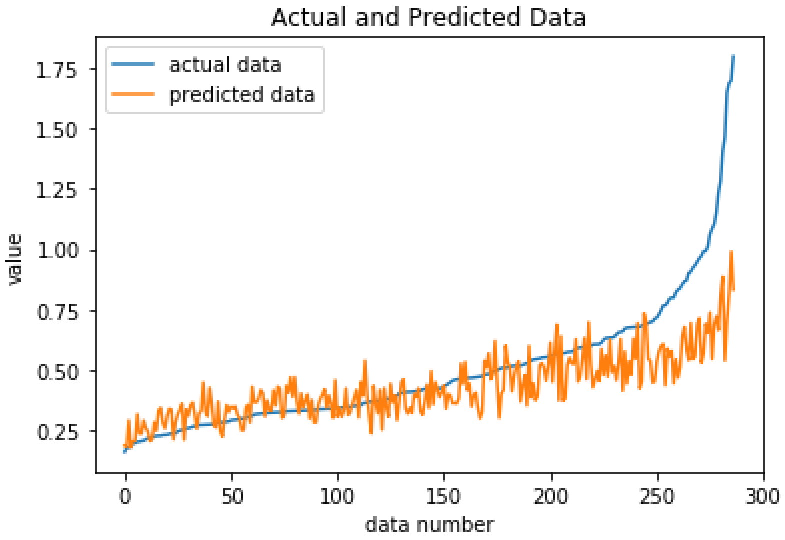


Figure 5.1 Actual and Predicted Data for Moving Average

5.2 Artificial Neural Network

We have done our experiments with the artificial neural network by using different combinations of features. As we mentioned in Section 4.3, there are two types of models that handle with the seasonality. The first model constructed by using deseasonalized output data which is called an NN1 model and we did not have seasonality feature in this model. On the other hand, the second model which is called as NN2 model has four seasonality features that correspond to four seasons of the year. Each model includes 36 months of features that were explained in Section 4.3. Furthermore, we have a production quantity feature and we created two more models by using these features for both NN1 and NN2 models. We compared the performance measures of these four models. Figure 5.2 shows all models created by artificial neural networks.

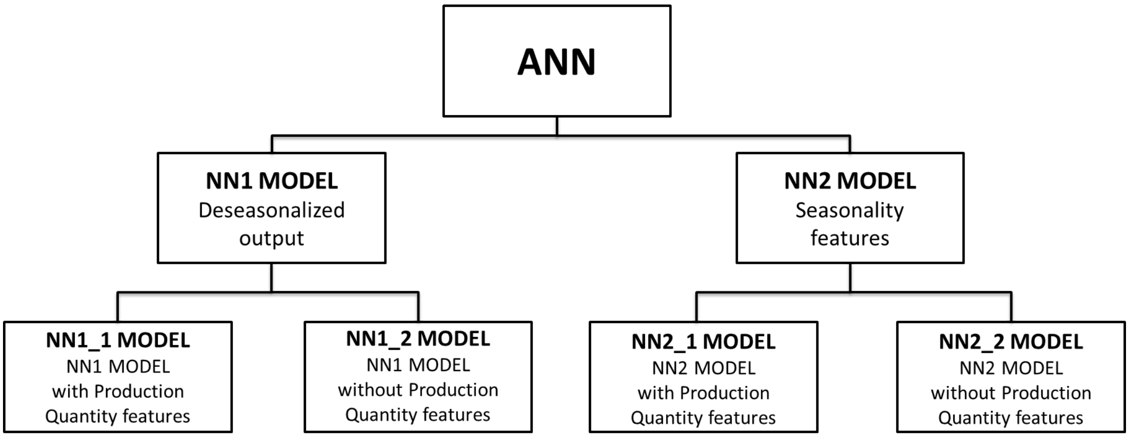

ANN
NN1 MODEL
Deseasonalized output
NN2 MODEL
Seasonality features
NN1_1 MODEL
NN1 MODEL with Production Quantity features
NN1_2 MODEL
NN1 MODEL without Production Quantity features
NN2_1 MODEL
NN2 MODEL with Production Quantity features
NN2_2 MODEL
NN2 MODEL without Production Quantity features

Figure 5.2 Artificial Neural Network Models

First of all, the NN1 model had done by using deseasonalized data. We had 1458 output data that are taken from the failure ratio matrix. We assumed that we are in January 2017, therefore we tried to predict the monthly number of failure ratios from January 2017 to October 2017. As a result of this situation, we had 1098 training and 360 test data. Moreover, we choose Adaptive Moment Estimation (`Adam`) optimizer to change weights from `keras` library, because it is computationally efficient and gives better results. It takes three parameters which are learning rate, beta1, and beta2. We tried different combinations of these parameters and found that learning rate 0.001, beta1 as 0.99 and beta2 as 0.999 is suitable to construct the model.

The loss function was based on ”Mean Absolute Percentage Error” and metrics with ”MAPE” used as well. Besides, we used L1 regularization to prevent the model from overfitting. While constructing architecture, we inserted the kernel_regularizers parameter by using regularizers.l1() function of `keras` library with parameter 0.01. In addition, we put a kernel constraint for the weights matrix. We applied `MinMaxNorm` class and we restricted norm for incoming weights between 0 and 1.

We applied K fold cross-validation into training data with 7 splits and cross-validation scores were found closer to each other. We understood that there is no overfitting at the model. Besides, we fitted model by using training data and the number of epochs decided as 300. As Figure 5.3 demonstrates that there is no overfitting at the model. In addition, we predicted output value for the test data, and actual and predicted test data are illustrated in Figure 5.4. We implemented the same methods by using only 36 features that represent age effects as well. The model did not overfit and worked properly. Performance measures of these models are presented in Table 5.2.

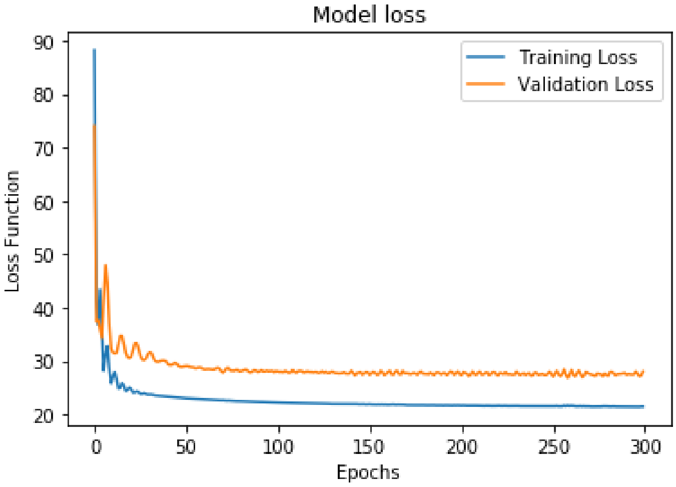


Figure 5.3 Validation and training data loss function of NN1_1 model

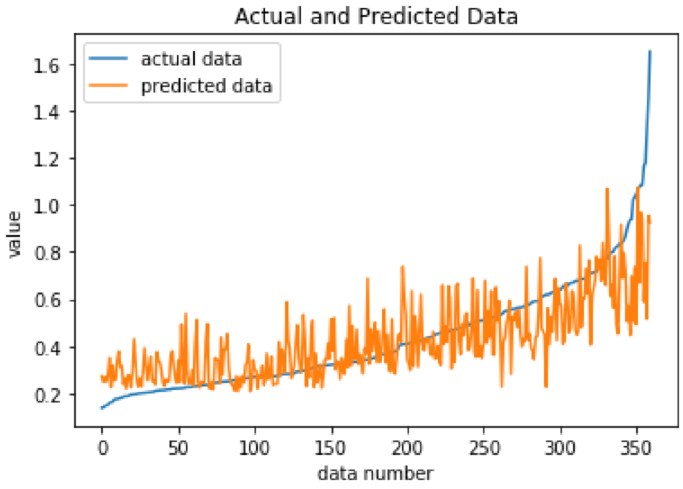


Figure 5.4 Actual and Predicted Data for Neural Network with Deseasonalized Data

Secondly, at the NN2 model, we constructed the model by using four more features which represent four seasons of the year instead of using deseasonalized data. We used the same methods with the NN1 model in order to split training and test data. Optimizer, learning rate, and beta values were also the same as the NN1 model. We implemented NN2 models with two different sets of features as NN2_1 and NN2_2 models. We changed only input nodes as 43 instead of 39 for model architecture since we had 43 features for the NN2_1 model and 39 for NN2_2 model. Figure that shows validation and training loss function for the NN2_1 model can be found in the Appendix, also actual and predicted value of the test data are available in the Appendix.

The performance measure values of these two models are given in Table 5.2.

	MAPE(%)	MAE	RMSE	R^2	Adjusted R^2
NN1_1 Model	27.4160	0.1088	0.1537	0.5610	0.5075
NN1_2 Model	29.7149	0.0943	0.1353	0.6936	0.6715
NN2_1 Model	27.2834	0.1070	0.1529	0.5657	0.5066
NN2_2 Model	37.0311	0.0974	0.1354	0.6930	0.6682

Table 5.2 Performance Measures for Neural Network Method with Different Features.

5.3 Support Vector Regression

In our experiment with ”Support Vector Regression”, we implemented the same logic with ANN models. Therefore we created four models for support vector regression as well. We called them as SVR1_1,SVR1_2,SVR2_1 and SVR2_2. Additionally, We divided training and test as 1098 and 360 with respectively that we pointed out in Section 5.2. After deciding the parameters, we draw the learning curve of the model by using training data to understand whether the model overfitted or not. The learning curve for each model shows that there is no overfitting or underfitting. Curve for SVR1_2 is shown at Figure 5.6. Given vertical values are mean absolute percentage errors and the horizontal line indicates training sizes. After ensuring correctly worked model, by using `SVR` function of `sckitlearn` with corresponding kernel functions and parameters for each SVR models that are explained in Section 4.4.1 and 4.4.2, we fitted our model with training data. We predicted test output with our model and actual and predicted data for SVR1_1 are given in Figure 5.5. Performance measure values are given in Table 5.3.

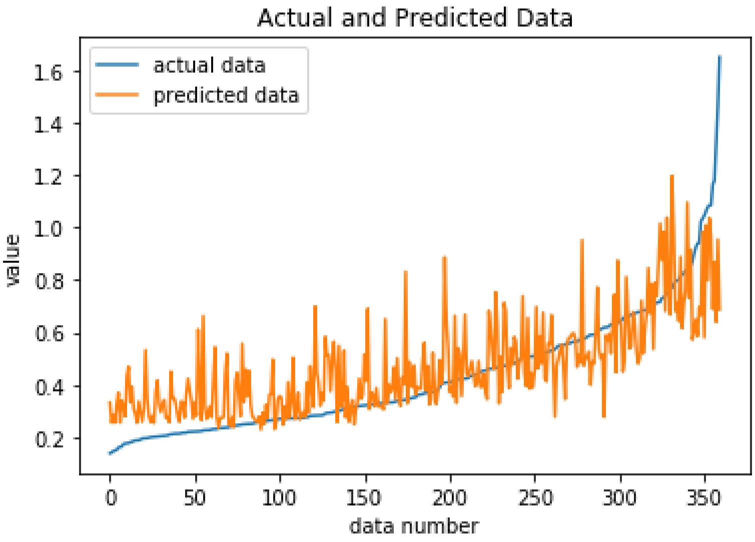


Figure 5.5 Actual and Predicted Data for SVR1_1 model

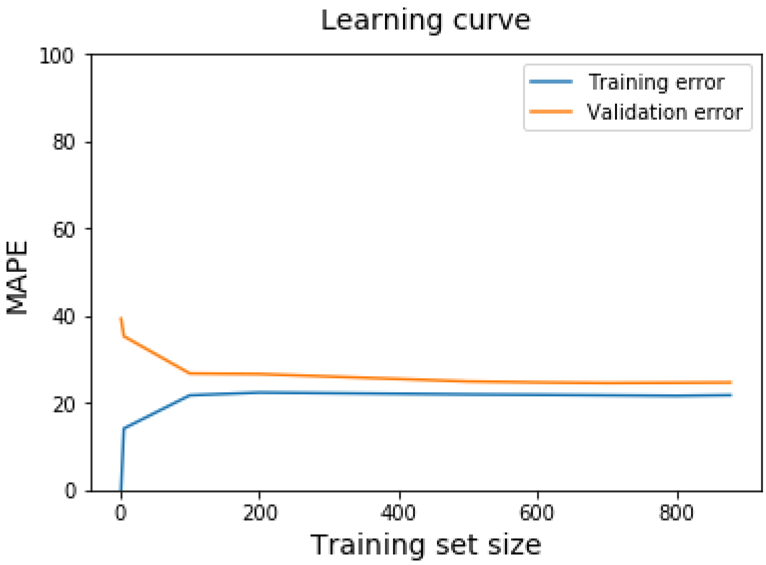


Figure 5.6 Learning curve for SVR1_2 model

	MAPE(%)	MAE	RMSE	R^2	Adjusted R^2
MA with 10 window	20.1020	0.1137	0.1837	0.5219	0.5203
NN1_1 model	27.4160	0.1088	0.1537	0.5610	0.5075
NN1_2 model	29.7149	0.0943	0.1353	0.6936	0.6715
NN2_1 model	27.2834	0.1070	0.1529	0.5657	0.5066
NN2_2 model	37.0311	0.0974	0.1354	0.6930	0.6682
SVR1_1 model	32.0997	0.11478	0.1615	0.5155	0.4564
SVR1_2 model	36.0575	0.1297	0.1692	0.4680	0.4087
SVR2_1 model	37.9597	0.1281	0.1759	0.4251	0.3468
SVR2_2 model	36.9367	0.1267	0.1669	0.4820	0.4170

Table 5.3 Performance Measures for All Models.

When we compared the performance measure results of all models at given Table 5.3, the moving average method gives better results for the mean absolute percentage error (MAPE) from all models. Besides, except for root mean square error (RMSE), for all performance metrics, the moving average gives a better result than SVR models. On the other hand, MAE, RMSE, R squared, and adjusted R squared values are worse than neural network models.

Chai and Draxler (2014) states that generally RMSE is applied for model sensitivity studies, and RMSE penalizes variance of the model since higher errors take higher weights with the RMSE. As a result of this, RMSE can minimize the maximum error of the model by minimizing the average error. In this study, by using forecasted failure ratios of the products, we are going to have the capacity and workforce planning, thus we try to minimize the maximum error of our models and we chose RMSE in order to compare our models.

According to RMSE values, we had 26.34% performance improvements for the artificial neural network and 12.08% improvements for the support vector regression model. Therefore we can infer from these results that if we have used suitable methods and parameters machine learning-based approaches can give better results than statistical-based methods. On the other hand, when we compared computational times of the models, as Table 5.4 shows that machine learning-based approaches need more computational powers.

	Computational Time (sec)
MA with 10 window	0.81
NN1_1 model	45.76
NN1_2 model	48.24
NN2_1 model	75.01
NN2_2 model	84.34
SVR1_1 model	77.45
SVR1_2 model	78.71
SVR2_1 model	63.81
SVR2_2 model	66.14

Table 5.4 Computational Times for All Models.

In this thesis, a higher computational time for machine learning methods did not have a high impact because of the size of the data. However, when the data size increase, computational time will increase. Therefore we can make an inference that even though machine learning methods give better performance results, statistical methods are more efficient in terms of computational results.

6. CONCLUDING REMARKS AND FUTURE WORK

In this thesis, we focus on the forecast of monthly failure ratios of refrigerators for a major multinational home appliances manufacturer headquartered in Turkey. Due to the special structure of the data set, the problem is not a typical time series forecasting problem, since the data set is multidimensional and in a matrix form. Each row is indeed a time series but they cannot be treated as independent time series since they represent the very same conditions with a single difference, i.e., the production month(batch). The rest of the conditions (trends, seasonality, etc.) are all same but there is no way in traditional time series analysis to transfer the information among the time series. Therefore, machine learning-based approaches might be considered more appropriate for the problem. However, the machine learning-based approaches should be tailored to the problem and should not be used without incorporating the domain expertise as part of the analysis. In this study, we aim to develop a machine learning-based approach that outperforms the current technique that is used by the company, at the same time evaluates the influence of different modeling approaches in terms of the forecast accuracy.

We applied artificial neural networks and support vector regression as machine learning-based approaches and we used our matrix data as the output of the model. By using domain expertise, we created features (inputs) as age, seasonal, and production quantity effect. According to the performance measure results that used for comparisons of models (RMSE, MAE, etc.), even though the moving average method gives better results than support vector regression, neural network models overcome moving average methods for almost all performance measures. On the other hand, there is a high difference between statistical and machine learning methods in terms of computational powers. Although, computational results for this study can be allowable, increasing data size can be an essential issue for the computational time.

In general, proposed methods for refrigerators can be adapted to other products i.e., washing machine, dishwasher, etc. As a future work of this study, the effects of the product model can be taken into considerations. In addition, forecasting methods can be implemented in order to predict the number of monthly sales for future months. By using estimated monthly sales and estimated monthly failure ratios, the number of failed products for each month can be predicted.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Agarwal, A. and Jayant, D. A. (2019). Support vector machine model for demand forecasting in an automobile parts industry: A case study. *Research Journal of Applied Sciences, Engineering and Technology*, 9:33–49.

Ahmed, N. K., Atiya, A. F., Gayar, N. E., and El-Shishiny, H. (2010). An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews*, 29(5-6):594–621.

Amini, M. H., Kargarian, A., and Karabasoglu, O. (2016). Arima-based decoupled time series forecasting of electric vehicle charging demand for stochastic power system operation. *Electric Power Systems Research*, 140:378–390.

Ariyo, A. A., Adewumi, A. O., and Ayo, C. K. (2014). Stock price prediction using the arima model. In *2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, pages 106–112. IEEE.

Bacchetti, A. and Saccani, N. (2012). Spare parts classification and demand forecasting for stock control: Investigating the gap between research and practice. *Omega*, 40(6):722–737.

Benkachcha, S., Benhra, J., and El Hassani, H. (2015). Seasonal time series forecasting models based on artificial neural network. *International Journal of Computer Applications*, 116(20).

Bontempi, G., Ben Taieb, S., and Le Borgne, Y.-A. (2013). *Machine Learning Strategies for Time Series Forecasting*, pages 62–77. Springer Berlin Heidelberg, Berlin, Heidelberg.

Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.

Boylan, J. E. and Syntetos, A. A. (2010). Spare parts management: a review of forecasting research and extensions. *IMA journal of management mathematics*, 21(3):227–237.

Callen, J. L., Kwan, C. C., Yip, P. C., and Yuan, Y. (1996). Neural network forecasting of quarterly accounting earnings. *International Journal of Forecasting*, 12(4):475–482.

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

Carbonneau, R., Laframboise, K., and Vahidov, R. (2008). Application of machine learning techniques for supply chain demand forecasting. *European Journal of Operational Research*, 184(3):1140–1154.

Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, pages 161–168.

Cavalieri, S., Garetti, M., Macchi, M., and Pinto, R. (2008). A decision-making framework for managing maintenance spare parts. *Production planning & control*, 19(4):379–396.

Chai, T. and Draxler, R. R. (2014). Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature. *Geoscientific model development*, 7(3):1247–1250.

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27.

Chollet, F. et al. (2015). Keras. `https://keras.io`.

Csáji, B. C. et al. (2001). Approximation with artificial neural networks. *Faculty of Sciences, Etvs Lornd University, Hungary*, 24(48):7.

Dekker, R., Pinçe, Ç., Zuidwijk, R., and Jalil, M. N. (2013). On the use of installed base information for spare parts logistics: A review of ideas and industry practice. *International Journal of Production Economics*, 143(2):536–545.

Hamzaçebi, C. (2008). Improving artificial neural networks' performance in seasonal time series forecasting. *Information Sciences*, 178(23):4550–4559.

Hansun, S. (2013). A new approach of moving average method in time series analysis. In *2013 conference on new media studies (CoNMedia)*, pages 1–4. IEEE.

Hill, T., O'Connor, M., and Remus, W. (1996). Neural network models for time series forecasts. *Management science*, 42(7):1082–1092.

Hylleberg, S. (1992). *Modelling seasonality.* Oxford University Press.

Jalil, M. N., Zuidwijk, R. A., Fleischmann, M., and Van Nunen, J. A. (2011). Spare parts logistics and installed base information. *Journal of the operational Research Society*, 62(3):442–457.

Khashei, M. and Bijari, M. (2011). A novel hybridization of artificial neural networks and arima models for time series forecasting. *Applied Soft Computing*, 11(2):2664–2675.

Kim, T. Y., Dekker, R., and Heij, C. (2017). Spare part demand forecasting for consumer goods using installed base information. *Computers & Industrial Engineering*, 103:201–215.

Lapedes, A. and Farber, R. (1987). Nonlinear signal processing using neural networks: Prediction and system modelling. Technical report.

Lin, K., Lin, Q., Zhou, C., and Yao, J. (2007). Time series prediction based on linear regression and svr. In *Third International Conference on Natural Computation (ICNC 2007)*, volume 1, pages 688–691. IEEE.

Makridakis, S., Spiliotis, E., and Assimakopoulos, V. (2018a). The m4 competition: Results, findings, conclusion and way forward. *International Journal of Forecasting*, 34(4):802–808.

Makridakis, S., Spiliotis, E., and Assimakopoulos, V. (2018b). Statistical and machine learning forecasting methods: Concerns and ways forward. *PloS one*, 13(3).

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

Porteus, E. L. (1986). Optimal lot sizing, process quality improvement and setup cost reduction. *Operations research*, 34(1):137–144.

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

Shabani, S., Yousefi, P., and Naser, G. (2017). Support vector machines in urban water demand forecasting using phase space reconstruction. *Procedia Engineering*, 186:537–543.

Sharda, R. and Patil, R. B. (1992). Connectionist approach to time series prediction: an empirical test. *Journal of Intelligent Manufacturing*, 3(5):317–323.

Tseng, F.-M., Yu, H.-C., and Tzeng, G.-H. (2001). Applied hybrid grey model to forecast seasonal time series. *Technological Forecasting and Social Change*, 67(2-3):291–302.

Tseng, F.-M., Yu, H.-C., and Tzeng, G.-H. (2002). Combining neural network model with seasonal time series arima model. *Technological Forecasting and Social Change*, 69(1):71–87.

Zhang, G., Patuwo, B. E., and Hu, M. Y. (1998). Forecasting with artificial neural networks:: The state of the art. *International journal of forecasting*, 14(1):35–62.

Zhang, G. P. (2001). An investigation of neural networks for linear time-series forecasting. *Computers & Operations Research*, 28(12):1183–1202.

Zhang, G. P. and Qi, M. (2005). Neural network forecasting for seasonal and trend time series. *European journal of operational research*, 160(2):501–514.

APPENDIX A

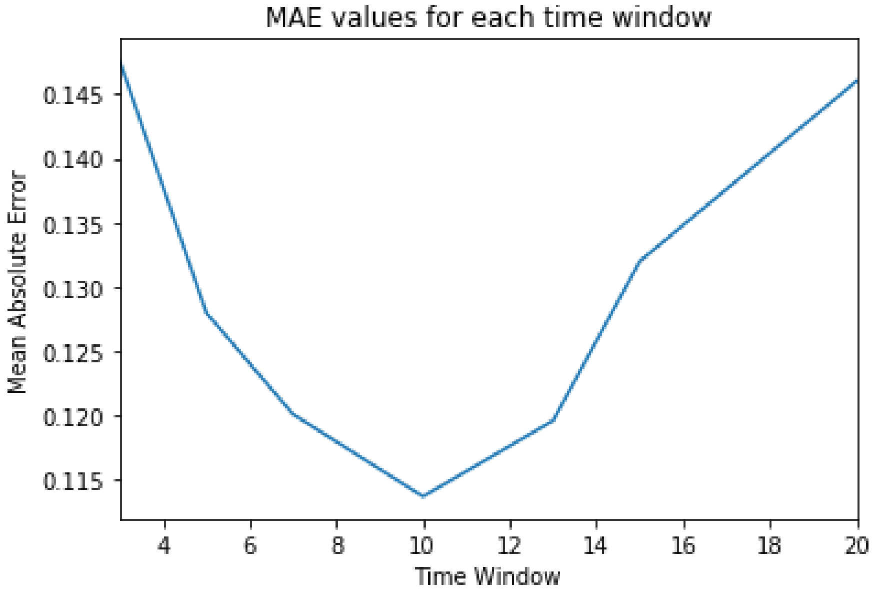


Figure A.1 Mean Absolute Percentage Error values for different time window values of Moving Average method

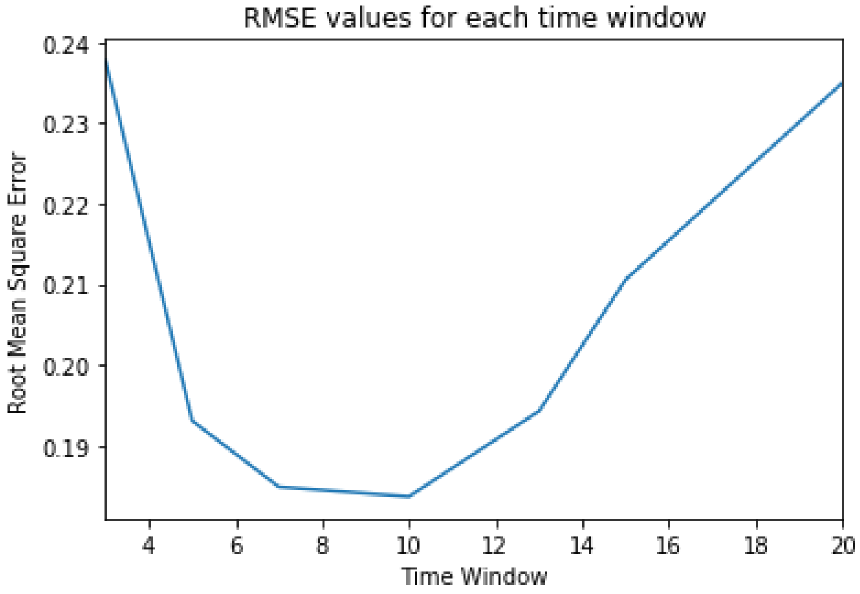


Figure A.2 Root Mean Square Error values for different time window values of Moving Average method

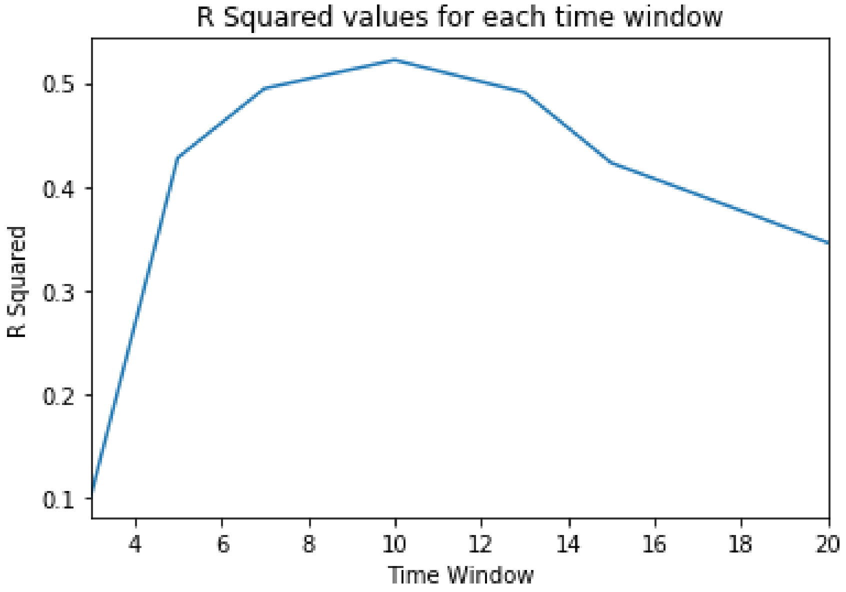


Figure A.3 R Squared values for different time window values of Moving Average method

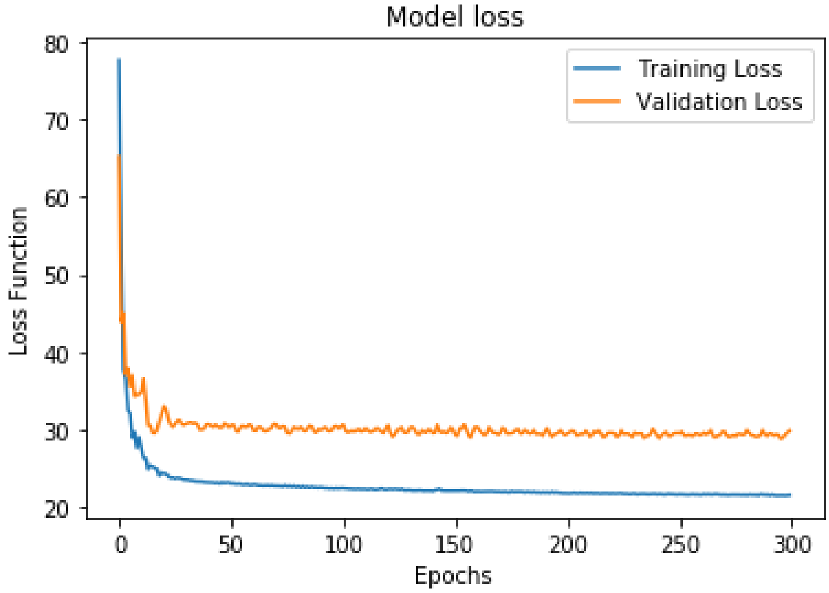


Figure A.4 Validation and Training Data Loss Function for NN2_1 model

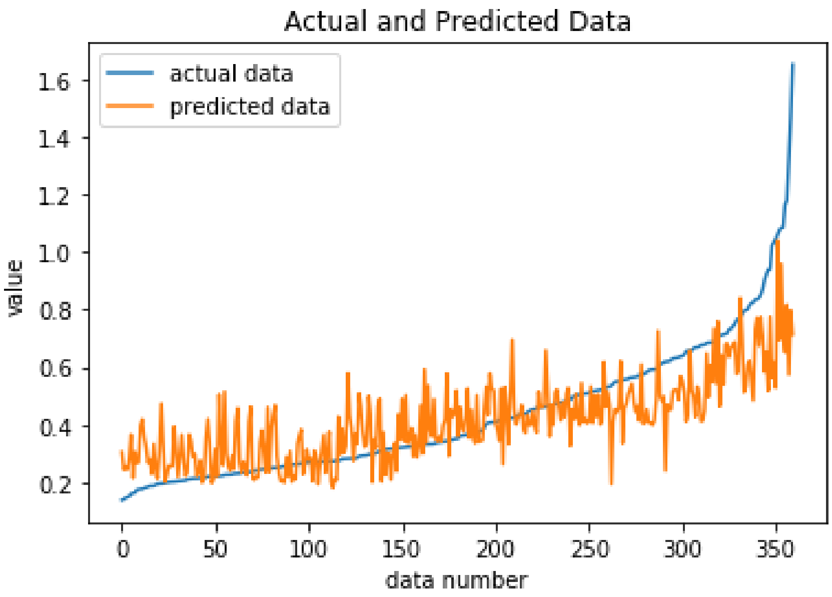


Figure A.5 Actual and Predicted Data for Neural Network with Seasonality Features