MASTERARBEIT | MASTER'S THESIS

Titel | Title

Comparison of Traditional and AI-Based Methods to Forecast Parts Demand

A Case Study in the Energy Infrastructure Supplier Sector

verfasst von | submitted by

Anika Schindlauer BSc MSc

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien | Vienna, 2026

Studienkennzahl lt. Studienblatt | Degree programme code as it appears on the student record sheet:	UA 066 914
Studienrichtung lt. Studienblatt | Degree programme as it appears on the student record sheet:	Masterstudium Internationale Betriebswirtschaft
Betreut von | Supervisor:	Univ.-Prof. Mag. Dr. Karl Franz Dörner Privatdoz.

Abstract

Diese Studie vergleicht klassische, statistische Bedarfsprognose-Methoden mit KI-basierten Vorhersagemodellen. Es wird analysiert, ob KI-basierte Ansätze den Bedarf genauer vorhersagen können als klassische, statistische Verfahren. Des weiteren wird untersucht, ob es spezifische Bedarfsmuster gibt, die dazu führen, dass für diese Muster KI-Modelle eine genauere Vorhersage erzeugen können als klassische Modelle. AuSSerdem wird ermittelt, ob es auch Teile gibt, deren Bedarf zu unregelmäSSig ist, um mit einem der betrachteten Modelle vernünftige Vorhersagen erzeugen zu können. Um diese Fragen zu beantworten wird die Bedarfshistorie von 30 verschiedenen Teilen, die vom Energieinfrastruktur-Zuliefererunternehmen Aqotec verwendet werden, analysiert. Es werden monatliche Vorhersagen und eine Jahresprognose auf Basis von Holt-Winters und SARIMA Modellen sowie auf Basis von zwei neuronalen Netzen erzeugt. Die Ergebnisse werden vorwiegend mittels Gewichtetem Mittleren Absoluten Fehler (Weighted Mean Absolute Percentage Error) evaluiert und mit der Vorhersage eines naiven Baseline-Modells verglichen. Die Ergebnisse zeigen, dass keines der Modelle universell für alle Teile und sowohl für die monatliche als auch für die Jahresprognose klar am besten funktioniert. Es gibt keine gesamte Gruppe an Teilen, für die KI-Modelle eine genauere Vorhersagequalität erreichen. Es gibt jedoch einzelne Teile, für die KI-basierte Modelle zu genaueren Vorhersagen führen. Über alle Teile hinweg jedoch können die genauesten monatlichen Prognosen mit Hilfe der SARIMA Modelle erzeugt werden. Keines der Modelle führt zu zufriedenstellenden monatlichen Vorhersagen für jene Teile, deren Bedarf besonders unregelmäSSig ist, wobei die Gewichteten Mittleren Absoluten Fehler teilweise bei circa 100% liegen. Für die Jahresmenge hingegen kann auch für die meisten dieser Teile eine zufriedenstellende Prognose erzeugt werden. Diese Studie vergleicht verschiedene Modelle zur Bedarfsvorhersage auf Basis von echten Industriedaten. Sie zeigt somit auch die Herausforderungen auf, genaue Bedarfsprognosen für verschiedenste Teile mit teilweise sehr unregelmäSSigen Bedarfsmustern zu erzeugen. AuSSerdem wird gezeigt, dass komplexere, KI-basierte Modelle nicht immer genauere Ergebnisse liefern als klassische, statistische Ansätze.

Abstract (English)

This study compares classical statistical demand forecasting approaches with AI-based time series prediction methods. It is analysed if AI-based models can forecast the parts demand more accurately than classical approaches and if there are specific demand patterns that lead to AI-approaches performing better than classical methods. Furthermore, it is investigated if there are parts with such irregular demand that no method can generate reasonable predictions. In order to answer these questions, the historical demand of 30 different assembly parts used by the energy infrastructure supplier company Aqotec is analysed and monthly and yearly demand predictions are generated fitting Holt-Winters and SARIMA models as well as two Neural Network based models. The results are evaluated mainly using the Weighted Mean Absolute Percentage Error and through comparing the forecasting quality to a naïve baseline approach. The results show that none of the models universally shows a superior performance for all parts and both for the monthly and the yearly forecasts. There is no group of parts, for which AI-based models generally perform better. There are, however, single parts for which AI-based approaches outperform the classical statistical models, but across all parts the SARIMA models tend to deliver the most promising results for the monthly forecasts. None of the models can produce reasonable monthly results for parts with a specifically high variation in the dataset with Weighted Mean Absolute Percentage Errors of around 100%, while yearly predictions of those parts achieve a significantly better accuracy than monthly forecasts. This study provides a comparison of different demand forecasting approaches on real-world data. It also emphasises the challenges of creating accurate predictions on datasets with partly considerably varying demands. It further points out that more complex AI-based methods do not always outperform classical statistical prediction approaches.

Contents

List of Figures

List of Tables

List of Acronyms

ACF Autocorrelation Function

AI Artificial Intelligence

AIC Akaiko Information Criterion

ARIMA Autoregressive Integrated Moving Average

ARMA Autoregressive Moving Average

BIC Bayesian Information Criterion

CV Variation Coefficient or Coefficient of Variation

ELM Extreme Learning Machine

ERP Enterprise Resource Planning

MAD Mean Abolute Deviation

MAE Mean Absolute Error

MAPE Mean Absolute Percentage Error

ML Machine Learning

MLP Multilayer Perceptron

MAPE Mean Absolute Percentage Error

NNAR Neural Network Autoregression

OEND Österreichische Nationalbank

PACF Partial Autocorrelation Function

PE Percentage

RNN	Recurrent Neural Network
RMSE	Root Mean Squared Error
SARIMA	Seasonal Autoregressive Integrated Moving Average
SCM	Supply Chain Management
SMAPE	Symmetric Mean Absolute Percentage Error
SME	Small and Medium-sized Enterprise
WMAPE	Weighted Mean Absolute Percentage Error

1. Introduction

1.1 Parts Demand Forecasting in supply chains

In recent years Artificial Intelligence (AI) has found its way into almost all economic sectors - from marketing to finance and production. One aspect which is highly relevant for production planning and punctual delivery is forecasting future demand both of final products and of assembly parts. Accurate and automatic demand forecasts are of great value within a supply chain, as they can lead to reduced inventory costs, allow a better planning of the required equipment and resources and therefore also might lead to reduced lead times of the end products (Heizer et al., 2024, p. 140 f.). Predictions can be generated for a short time range, which usually includes forecasts of up to one year, for example in order to optimise production levels or the purchasing and reordering process of required parts (Heizer et al., 2024, p. 140). Medium-term predictions are typically predictions from three months up to three years in advance, which are often used for general production planning as well as for the planning of financial resources and budgets, while long-term predictions are generally for time spans from three years onwards and are used to optimise strategic decisions, such as the location of the company (Heizer et al., 2024, p. 140).

To forecast parts demand, there exist several different techniques, ranging from classical statistical methods for time series forecasting such as the ARIMA approach first introduced in 1970 by Box and Jenkins (1970) or the Holt Winters method based on work from Holt (2004), originally published in 1957, and Winters (1960) to more advanced AI-techniques based on Neural Networks.

Recent developments show that more and more companies have already implemented automatic forecasting systems leveraging Artificial Intelligence techniques, as highlighted in a paper by Mediavilla et al. (2022) presenting an overview of AI methods used in the field of demand forecasting. Mediavilla et al. (2022, p. 1126) emphasise that even in 2021 there was already a trend towards using more complex AI-based methods instead of relying on classical statistical methods. Additionally, Douaioui et al. (2024) show different applications of AI-based prediction models in the field of Supply Chain Management, also briefly addressing that those models might have the power to improve the forecasting accuracy compared to traditional approaches such as SARIMA (Douaioui et al., 2024, p. 2).

1.2 Company overview: Aqotec

The company Aqotec provides the datasets for this study. The information presented in this section is based on internal company materials and presentations.

The company Aqotec GmbH is a Medium-Sized Austrian company and its headquarter is located in Weissenkirchen im Attergau in Upper Austria. Aqotec was founded in 1999 and in addition to the headquarter and production site in Austria, the company also operates a production facility in Hodonice in the Czech Republic. In total around 200 employees work for Aqotec, of which around 125 have their working place in Austria. The company has more than 15 sales markets, mainly countries in Europe including Austria itself as well as Germany, the Netherlands, Switzerland and Slovakia.

Aqotec's main business is to build and assembly heat transfer stations. Heat transfer stations are the connection between a central heating grid and the individual heating systems of buildings through the transfer of thermal energy from the heating water of the central heating grid to the heating water of the building. As it can be seen in Figure 1.1, to build such stations a variety of different parts and components is required, ranging from small parts such as screws, gaskets or small electrical parts to bigger parts such as the frame, the stations are built in (see Figure 1.1b).

The energy transfer itself takes place in the heat exchanger, which is the most important part of the heat transfer station and which is responsible to transfer the heat of the district grid's heating water to the heating system of the building. To control factors such as the water's flow rate and the temperature several valves are installed, which can be automatically controlled via electric actuators. The automatic control of the station as well as the collection of measurement data takes place in the controller, which can also be considered as the brain of the heat transfer station. It also determines the settings and positions of the valves in order to generate the desired heat output. The data collected is usually transmitted using data and network cables. The heat meter is another important component of the heat transfer station, especially for billing purposes. It measures how much energy is transferred to the heating system of the building in order to correctly charge for it. Furthermore, the stations also contain small parts, such as valves and gaskets as well as small electrical parts. Depending on the requirements of the customer additional pumps or flow meters might be installed in the heat transfer station. The entire heat transfer station is mostly installed within a frame, which is the support structure around it and can be seen in Figure 1.1b. The number of outgoing parts from the company's warehouse, which are used for assembling the heat transfer stations, is always recorded and stored in the ERP (Enterprise Resource Planning) system of the company, resulting in a time series dataset for each part with data several years into the past.

(a) The components of a heat transfer station

(b) A heat transfer station built within a frame

Figure 1.1: The components of heat transfer stations as sold by the company Aqotec. Pictures provided by the company Aqotec.

1.3 Objectives of this study

The goal of this study is to compare the quality of traditional time series forecasting approaches SARIMA and Holt-Winters with two Neural Network approaches, a Neural Network Autoregression (NNAR) approach as well as Multilayer Perceptron (MLP), to forecast the monthly and yearly demand of the required parts.

As the company requires more than 4000 different parts, the parts are firstly ordered regarding the purchase amount and the purchase value of the year 2024. Subsequently, the 15 parts with the highest purchase value and the 15 most purchased parts in 2024 are chosen to evaluate the models on.

In the following the 30 selected time series are split up into three groups based on an XYZ-analysis and the four different forecasting models are fitted to the datasets. Based on the monthly and yearly evaluation, it is analysed which approaches tend to work best on which class of parts. Furthermore, it is investigated if the application of forecasting methods makes sense for all parts or if there are also items with such an irregular demand that no reasonable predictions of business value can be generated.

In the thesis the following research questions are answered:

- Can AI-based time series models forecast the demand of the required parts more accurately than traditional time series forecasting approaches?
- Are there specific demand patterns that lead to the result that AI-based methods perform better on those parts than traditional statistical forecasting methods?
- Are there parts with especially irregular demand patterns where none of the models can provide reasonable results?

1.4 Methodology

This study is based on a quantitative approach applying different time series forecasting methods to real-world historical data.

For each part which is used by the company a dataset exists including the number of outgoing pieces per point in time. As the company works with more than 4000 different parts, at first the 15 most purchased parts and those with the highest purchase value in 2024 are chosen for further evaluation.

The prediction models evaluated are Holt-Winters, SARIMA, a Neural Network Autoregression approach as well as a Multilayer Perceptron. These methods were chosen, as they cover a wide range of complexity. While Holt-Winters is a relatively simple forecasting method based on exponential smoothing (Vogel, 2015, p. 70), SARIMA is a more complex statistical approach (Krispin, 2019, p. 329) . The Neural Network Autoregression approach uses a Neural Network with only one hidden layer (Hyndman et al., 2026, p. 111), whereas the Multilayer Perceptron allows for slightly deeper, more complex architectures (Venkateswaran and Ciaburro, 2017, p. 19).

The four different models are fitted to the time series producing a 12 months in advance forecast, which is evaluated based on the test period from July 2024 until June 2025. In addition to the in advance predictions rolling forecasts are generated, where always the demand of the following month is predicted. The forecasts are evaluated based on the Weighted Mean Absolute Percentage Error (WMAPE) as well as partly based on the Symmetric Mean Absolute Percentage Error (SMAPE). In addition to the monthly forecasts, a yearly forecast is generated based on summing up the monthly forecasted values and the predicted yearly demand is compared to the actual yearly demand based on the Percentage Error (PE).

As the company up to now does not use any forecasting tool, but rather only estimates the demand of the parts based on the expertise of the employees, the naïve approach of assuming the same demand for each month as in the same month of the preceding year is used as a baseline model. Each forecasting method is compared to this baseline approach in order to assess the added value of the method.

The practical study is carried out using the statistical software `R` (R Core Team, 2025) in the version 4.5.2. Packages used for the data preparation and modelling are, among others, `dplyr` (Wickham et al., 2026b), `forecast` (Hyndman et al., 2026; Hyndman and Khandakar, 2008) and `nnfor` (Kourentzes, 2023). The graphics in this study are produced using the package `ggplot2` (Wickham et al., 2026a; Wickham, 2016).

1.5 Outline of this thesis

The Chapter Related work and literature presents an overview of relevant studies and previous research in the field of demand forecasting, mainly focusing on the comparison of different forecasting methods, especially of classical statistical and AI-based approaches.

The subsequent chapter, Chapter 3, introduces the forecasting methods used in this study. It briefly discusses the theoretical backgrounds and mathematical formulations of the approaches and explains the software implementations in `R` of each of the approaches.

In the Chapter Data preparation the data collection and preparation steps are outlined and a first overview of the time series datasets is given. Additionally, the time series are grouped based on an XYZ-classification approach.

The Chapter Evaluation discusses the evaluation strategies, the baseline model and the evaluation metrics, which are used to discuss the results of the models.

The subsequent chapter, Chapter Results, illustrates and discusses the results of the different forecasting methods applied via displaying overview graphics as well as detailed plots for single items. Additionally, in the Section Possible improvements two possible ways of improving the forecasting results are discussed.

The Chapter Conclusion & possible future research provides a conclusion, discusses the limitations of this study and provides a brief outlook of possible future research opportunities in the field.

The Appendix contains the full tables of results, a table of hyperparameter configurations for the fitted functions in `R` as well as some additional graphics.

2. Related work and literature

This chapter presents a brief overview of existing work in the domain of demand forecasting as well as in the field of comparing classical time series prediction models with more complex AI models.

Time series forecasting is a thoroughly discussed topic in research, especially with new-emerging Machine Learning (ML) methods promising an improved quality even for more complex applications. Kontopoulou et al. (2023) present an overview of studies conducted in different fields comparing the traditional ARIMA approach with Machine Learning models mentioning studies in different fields, such as the forecast of financial data (see for example Rhanoui et al. (2019)), of health data (see for example ArunKumar et al. (2022)) or of environmental and weather data (see for example Liu et al. (2021) or Spyrou et al. (2022)). Another study comparing traditional forecasting approaches, such as Holt-Winters and ARIMA, with Neural Network approaches is Talkhi et al. (2021), predicting Covid-19 cases and deaths. In many of the considered studies in Kontopoulou et al. (2023) the more complex Machine Learning models, especially Neural Network approaches, outperform the traditional models. This is, however, not true for all investigated time series datasets. ArunKumar et al. (2022), for example, forecasts Covid-19 cases, recoveries and deaths in various countries including the United States of America, India and Russia, and observes that while the more advanced Neural Network approaches perform better for strongly non-linear data, there are also time series which can be better forecasted using the SARIMA or ARIMA approach (ArunKumar et al., 2022, p. 7600 p.).

While there are numerous studies on the comparison of traditional and AI-based methods in time series forecasting in general, there is less literature focused on comparing these methods for demand forecasting or more specifically for parts demand forecasting. Mediavilla et al. (2022) present an overview of applied methods in the field of demand forecasting, including studies dealing with data from the automotive industry (see for example Gonçalves et al. (2020)) or in manufacturing generally (see for example Nemati-Amirkolaii et al. (2017) or Francis and Kusiak (2017)). Mediavilla et al. (2022, p. 1130), however, also emphasise that most of the papers analysed focus on the demand prediction from the retailer's point of view instead of from the manufacturer's point of view. Additionally, a tendency towards using advanced AI-techniques such as Neural Network approaches is observed (Mediavilla et al., 2022, p. 1130). The trend towards more complex AI-methods is also observed in Douaioui et al. (2024) and Giannopoulos et al. (2025). Douaioui et al. (2024) analysed 119 papers from 2015 until 2024 dealing with Machine Learning techniques in the field of Supply Chain Management (SCM). It is not only emphasised that AI-based methods might produce more accurate results than classical approaches (Douaioui et al., 2024, p. 2), but also that the number of publications in the field of ML used in SCM has drastically increased since 2021 (Douaioui et al., 2024, p. 7). Giannopoulos et al. (2025, p. 5 ff.) specifically focus on 99 journals or conference papers investigating the forecast of intermittent or lumpy demand in different industries, including the automotive industry, the aviation and naval industry, electronics and electrics as well as heavy industries and retail. In all of the sectors mentioned, except for aviation and naval as well as for heavy industries, Neural Network approaches were the most commonly used methods for the forecast of demand (Giannopoulos et al., 2025, p. 18 f.). In aviation and naval and in heavy industries so-called synthetic methods were used including transfer learning, ensembles or hybrid approaches, where classical methods such as ARIMA are combined with AI techniques (Giannopoulos et al., 2025, p. 18 f.). Giannopoulos et al. (2025, p. 34 f.) conclude that there is still a gap in research regarding demand forecasting, especially because many of the investigated papers did not not work with real-world datasets and hence the actual complexity of demand prediction is hardly captured. Additionally, a wide variety of different evaluation metrics is used, some of which are also scale-dependent, which makes a valid comparison of the model performances impossible and the possible incorporation of forecast systems into the decision support system of a company is often not discussed in the studies, which is especially a challenge when applying very complex, resource-intensive models (Giannopoulos et al., 2025, p. 35).

There is a variety of literature on parts demand forecast. However, the focus is mostly on the prediction of spare parts demand, rather than on general parts demand to build and assembly. Çera Pinçe et al. (2021), for example, investigate 56 studies on the comparison of different forecasting methods to predict the intermittent demand of spare parts. Çera Pinçe et al. (2021, p. 23) also concludes that there is a trend towards more complex AI-based methods applied in demand forecasting.

To summarise, while there are numerous studies on time series and demand forecasting in general as well as some on the prediction of spare parts demand, hardly any study focuses on the forecast of parts demand in general. Additionally, the forecasting methods are hardly analysed on real-world datasets for a variety of different parts. This thesis investigates the forecast of demand of a diverse portfolio of assembly parts based on real-world datasets and focuses on answering the question, if AI-based methods outperform classical approaches on these time series. Additionally, not only forecasts for several periods in advance are evaluated, but also rolling forecasts are considered, which is another topic hardly covered in the literature yet.

3. Classical & AI-based time series forecasting methods

The following chapter provides an introduction to time series data in general and introduces the four forecasting approaches applied in this thesis, which are two statistical methods Holt-Winters and SARIMA and two AI-based Neural Network approaches.

3.1 Characteristics of time series data

Time series data consists of observations recorded at equally spaced points in time, often every hour, every day or every month (Krispin, 2019, p. 8). Real world examples of time series data include demographic data such as population size, environmental figures such as yearly rainfall (Konar and Bhattacharya, 2017, p. 2) or economic and financial metrics such as unemployment rates or daily stock prices (Shumway and Stoffer, 2025, p. 1).

Time series can be decomposed to different components (Krispin, 2019, p. 142). Krispin (2019, p. 142 ff.) mentions the following typical structural patterns in time series data:

- Trend: Time series can have an upwards or a downwards trend, if the values get constantly bigger or smaller as time progresses. Trends might be linear or exponential. (Krispin, 2019, p. 145)
- Seasonality: Time series can have seasonal fluctuations, for example typically higher values on the weekends than during the week or in the summer than in the winter. Time series might also have multiple seasonal patterns. (Krispin, 2019, p. 147)
- Cycles: Cycles are fluctuations besides the seasonal patterns. Unlike seasonal

patterns, they do not always happen at the same points in time and can have different cycle lengths, such as macroeconomic cycles. (Krispin, 2019, p. 143)

Additionally to the structural elements, time series usually also have a non-structural, irregular component that is not covered by the structural parts (Krispin, 2019, p. 142). Krispin (2019, p. 142 f.) uses the following equation to define a time series, where Y_t is the time series value at time point t and T, S, C and I stand for the structural parts trend, seasonality and cycle, and the irregular component receptively:

$$Y_t = T_t + S_t + C_t + I_t \tag{3.1}$$

Shumway and Stoffer (2025, p. 1) point out that most conventional statistical methods, such as Linear Regression, fail at forecasting time series data due to their requirement that the values need to be independent from each other and identically distributed. Time series data violates this assumption, as the observations at contiguous points in time are usually correlated with each other (Shumway and Stoffer, 2025, p. 1).

Therefore, numerous forecasting approaches explicitly dedicated to time series data have been developed in the last decades. While the statistical approaches Holt-Winters and SARIMA have already been introduced in the 1950s, 1960s and 1970s (see Holt (2004) originally published in 1957, Winters (1960) and Box and Jenkins (1970)) and have since then been commonly used techniques in time series forecasting, Neural Networks have only gained considerable attention in the 2010s with ever increasing computing resources, even though the original development of Neural Networks also dates back to the 1950s (Vandeput, 2021, p. 228 ff.). The advantage of such more complex approaches is that they sometimes tend to more accurately predict highly volatile time series with complex patterns (Khan et al., 2023, p. 2) (Vandeput, 2023, p. 144). The disadvantage of these more complex approaches is, however, that the produced forecasts are less explainable and less transparent than when classical statistical methods are used (Vandeput, 2023, p. 145). Additionally, the required amount of data for Neural Network methods to create accurate forecasting models tends to be higher than for statistical approaches (Vandeput, 2023, p. 145). Classical statistical approaches, on the other hand, tend to be better interpretable, often require less amount of data to produce reasonable results and tend to be easier to set up (Vandeput, 2023, p. 141). The latter point, however, is not completely valid when setting the models up with `R` as simple Neural Networks for time series forecasting can be fitted with the same amount of effort as for training classical statistical models (see Section 3.4).

The classical statistical approaches Holt-Winters and SARIMA as well as the two Neural Network approaches used for prediction in this study are more closely introduced in the following Sections 3.2, 3.3 and 3.4.

3.2 The Holt-Winters approach

The Holt-Winters approach was introduced by papers from Holt (2004), which originally dates back to the year 1957, and Winters (1960).

3.2.1 Mathematical formulation

The Holt-Winters model is based on the method of Simple Exponential Smoothing (Vogel, 2015, p. 70). In Simple Exponential Smoothing a weighted average is calculated using the previous observations of the time series, whereby observations in the more distant past have less influence on the prediction than values more recently observed (Krispin, 2019, p. 302). The method is called Exponential Smoothing because, as Krispin (2019, p. 302) points out, the weights of the values exponentially decay, the further they lay in the past. According to Krispin (2019, p. 302) the Exponential Smoothing method can be mathematically formulated as follows:

$$\hat{y}_{t+1} = \alpha \cdot y_t + (1 - \alpha) \cdot \hat{y}_t \tag{3.2}$$

In this equation (Equation 3.2) $\hat{y}_{t+1}$ is the value predicted for the point in time $t+1$, y_t is the most recent observation, $\hat{y}_t$ is the value which was forecasted for t in the previous forecasting step and α is the smoothing parameter, determining how sharply the weights decay (Krispin, 2019, p. 302).

While the Simple Exponential Smoothing method is only reasonable to apply on time series with no trend, as the model basically only captures the series level (Krispin, 2019, p. 301), Charles C. Holt extended the Simple Exponential Smoothing approach to also incorporate a trend component (Vogel, 2015, p. 70). This model is called the Holt method or Double Exponential Smoothing as it is dependent on two smoothing parameters α and β (Krispin, 2019, p. 311).

Mathematically, according to Vogel (2015, p. 71), the Holt method can be formulated in the following way:

m_t symbolises the level of the time series and can be calculated via the formula:

$$\hat{m}_t = \alpha \cdot y_t + (1 - \alpha) \cdot (\hat{m}_{t-1} + \hat{b}_{t-1}) \tag{3.3}$$

b_t stands for the trend increase $m_t - m_{t-1}$ and is calculated as follows:

$$\hat{b}_t = \beta \cdot (\hat{m}_t - \hat{m}_{t-1}) + (1 - \beta) \cdot \hat{b}_{t-1} \tag{3.4}$$

Both values have to be calculated recursively for all $t = 3, 4, ..., n$, while $\hat{m}_2 = y_2$ and $\hat{b}_2 = y_2 - y_1$ are the starting values. Using $\hat{m}_n$ and $\hat{b}_n$ the following equation can be used to predict h future values of the time series:

$$\hat{y}_{n+t} = \hat{m}_n + \hat{b}_n \cdot t \tag{3.5}$$

In equantion (3.5) t can take the values $t = 1, 2, ..., h$.

Vogel (2015, p. 72) furthermore points out, that the optimal values for α and β can be chosen based on the sum of squared errors given by the following formula:

$$\sum_{t=m}^{n} (y_t - \hat{y}_t)^2 \tag{3.6}$$

where $\hat{y}_t$ is calculated as follows:

$$\hat{y}_t = \hat{m}_{t-1} + \hat{b}_{t-1} \tag{3.7}$$

Vogel (2015, p. 72) mentions, that m can be any value, but it should not be too small, as otherwise the starting values have too much influence.

The Holt method allows the prediction of a time series with trend, but not yet with seasonality. In 1960 Peter Winters, a student of Holt, added a seasonal component to Holt's approach, which turned it into the Holt-Winters model (Vogel, 2015, p. 72 f.). The Holt-Winters approach is considered to be the most advanced Exponential Smoothing model (Krispin, 2019, p. 318).

Vogel (2015, p. 73) mathematically defines the Holt-Winters model as follows:

Like in equation (3.3) $\hat{m}_t$ represents the level of the time series and it can be mathematically expressed in the Holt-Winters model using the equation:

$$\hat{m}_t = \alpha \cdot (y_t - \hat{s}_{t-1}) + (1 - \alpha) \cdot (\hat{m}_{t-1} + \hat{b}_{t-1}) \tag{3.8}$$

Similarly to equation (3.4) the trend is given as:

$$\hat{b}_t = \beta \cdot (\hat{m}_t - \hat{m}_{t-1}) + (1 - \beta) \cdot \hat{b}_{t-1} \tag{3.9}$$

Additionally, the seasonal component with a period length of l is calculated using the formula:

$$\hat{s}_t = \gamma \cdot (y_t - \hat{m}_t) + (1 - \gamma) \cdot \hat{s}_{t-1} \tag{3.10}$$

The smoothing parameters α, β, γ all need to have values between 0 and 1 and due to the more complex nature of the model are usually optimised computationally (Vogel, 2015, p. 73).

Furthermore, Vogel (2015, p. 73) mentions that while these equations are used from the second seasonal period onwards, the starting values for $t = 1, ..., l$ in the Holt-Winters method are given as follows:

$$\hat{m}_l = \frac{1}{l} \sum_{i=1}^{l} y_i \tag{3.11}$$

$$\hat{b}_l = \frac{1}{l^2} \sum_{i=1}^{l} (y_{i+l} - y_i) \tag{3.12}$$

$$\hat{s}_t = y_t - \hat{m}_t \tag{3.13}$$

According to Vogel (2015, p. 73) values in future periods can then be predicted using:

$$\hat{y}_{n+t} = (\hat{m}_n + t \cdot \hat{b}_n) + \hat{s}_{n+t-l} \tag{3.14}$$

with $t = 1, ..., l$.

While the previous equations describe an additive Holt-Winters model, there are also variations of the approach with multiplicative seasonal components, with a multiplicative trend or with both components being of multiplicative nature (Krispin, 2019, p. 319). The multiplicative model is not explained in detail, as it is not used in this study.

3.2.2 Software implementation

The Holt-Winters approach is, among others, integrated in the `R` built-in package `stats` through a function called `HoltWinters` (R Core Team, 2026a). According to its documentation the optimal smoothing parameters α, β and γ are established by the minimization of the prediction error and it can be chosen between an additive and a multiplicative seasonal component, whereby the default setting is additive (R Core Team, 2026a). The fitted Holt-Winters model can then be used to predict future values using the built-in function `predict` (R Core Team, 2026b).

3.3 The SARIMA approach

The ARIMA approach was introduced in 1970 by Box and Jenkins (1970) and builds on the assumption that it is possible to explain the current observation of a time series using a subset of past values of this series (Shumway and Stoffer, 2025, p. 85). Krispin (2019, p. 329) mentions that the ARIMA method is the most robust traditional time series forecasting approach.

3.3.1 Mathematical formulation

The idea of ARIMA models is based on the ARMA (Autoregressive Moving Average) method introduced by Whittle in 1951, which assumes the underlying time series to be stationary (Shumway and Stoffer, 2025, p. 85). Shumway and Stoffer (2025, p. 21) mathematically defines the stationarity of a time series in the following way:

$$\{x_{t_1}, x_{t_2}, ..., x_{t_k}\} \overset{d}{=} \{x_{t_{1+h}}, x_{t_{2+h}}, ..., x_{t_{k+h}}\} \tag{3.15}$$

The equation needs to be valid for all $k = 1, 2, ...,$, for all points in time $t_1, t_2, ...t_k$ and for all offsets in time $h = 0, \pm 1, \pm 2, ...$ and $\overset{d}{=}$ means that both sets need to follow an equal distribution (Shumway and Stoffer, 2025, p. 21).

In a less mathematical way, Krispin (2019, p. 330) defines stationary time series as data series where the mean, the variance and the correlation structure stay the same over time. While the class of ARMA models only works with naturally stationary time series, ARIMA models are also applicable to non-stationary time series, if they become stationary by differencing them a finite number of times (Brockwell and

Davis, 2016, p. 158).

In contrast to classical linear regression models, where the value of the predicted variable only depends on the current values of the regressors, for autoregressive models it is assumed that the current value is explainable by the past values of the time series (Shumway and Stoffer, 2025, p. 85). Shumway and Stoffer (2025, p. 86) mathematically defines a so-called Autoregressive Model (AR) in the following way:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + ... + \phi_p x_{t-p} + \omega_t \tag{3.16}$$

In this equation p is the order of the autoregressive model, x_t is a stationary time series with mean zero, the irregular component ω_t is normally distributed with mean zero and $\phi_1, \phi_2, ..., \phi_p$ are constants, whereby ϕ_p is not allowed to be zero (Shumway and Stoffer, 2025, p. 86). Shumway and Stoffer (2025, p. 86) additionally mentions, that if the mean of the time series is not zero, the term x_t can be replaced with $x_t - \mu$ in Equation (3.16).

ARIMA models do not only consist of the autoregressive part, but also of the moving average part, which can be, according to Shumway and Stoffer (2025, p. 92), mathematically expressed in the following way:

$$x_t = w_t + \theta_1 w_{t-1} + \theta_2 w_{t-2} + ... + \theta_q w_{t-q} \tag{3.17}$$

In this equation q is the order of the moving average model, $\theta_1, \theta_2, ...\theta_q$ are parameters with θ_q not allowed to be zero (Shumway and Stoffer, 2025, p. 92) and w_t is a so-called White Noise, which means that the values are uncorrelated, have a mean of zero and a finite variance (Shumway and Stoffer, 2025, p. 11). In the case of an moving average model, w_t is also normally distributed (Shumway and Stoffer, 2025, p. 92).

Putting together the autoregressive and the moving average part for a stationary time series with mean μ, Shumway and Stoffer (2025, p. 95) defines the following mathematical formulation of an ARMA model with autoregressive order p and moving average order q:

$$x_t = \mu + \phi_1(x_{t-1} - \mu) + ... + \phi_p(x_{t-p} - \mu) + w_t + \theta_1 w_{t-1} + ... + \theta_q w_{t-q} \tag{3.18}$$

As seasonality, for example yearly, monthly or quarterly, naturally plays a huge role in time series forecasting, the ARIMA model was extended to the SARIMA model also taking into account seasonal fluctuations (Shumway and Stoffer, 2025, p. 157). In a similar manner like the ARMA model defined in Equation (3.18) Shumway and Stoffer (2025, p. 157) defines the seasonal $ARMA(P, Q)_S$ model using the following equation:

$$\Phi_P(B^S)x_t = \Theta_Q(B^S)w_t \tag{3.19}$$

Additionally, Shumway and Stoffer (2025, p. 157 f.) defines the operators $\Phi_P(B^S)$ and $\Theta_Q(B^S)$ as follows:

$$\Phi_P(B^S) = 1 - \Phi_1 B^S - \Phi_2 B^{2S} - ... - \Phi_P B^{PS} \tag{3.20}$$

$$\Theta_Q(B^S) = 1 + \Theta_1 B^S + \Theta_2 B^{2S} + ... + \Theta_P B^{PS} \tag{3.21}$$

B is the so-called Backshift-Operator and applying it to a variable means that the variable is shifted back by one point in time or by multiple points in time if the operator is applied multiple times, such as given by B^S (Vogel, 2015, p.83).

The seasonal $ARMA(P, Q)_S$ model can be rewritten in a similar form as in Equation (3.18) by putting together the operators from Equation (3.20) and from Equation (3.21) and slightly reformulating Equation (3.19):

$$x_t = \Phi_1 x_{t-S} + \Phi_2 x_{t-2S} + ... + \Phi_P x_{t-PS} + w_t + \Theta_1 w_{t-S} + \Theta_2 w_{t-2S} + ... + \Theta_Q w_{t-QS} \tag{3.22}$$

In a compact form, including the ordinary and the seasonal autoregressive and moving average parts using the backshift operator, Shumway and Stoffer (2025, p. 162) defines the Seasonal Autoregressive Integrated Moving Average Model (SARIMA) model in the following way:

$$\Phi_P(B^s)\phi_p(B)\nabla_S^D\nabla^d x_t = \delta + \Theta_Q(B^s)\theta_q(B)w_t \tag{3.23}$$

In Equation (3.23) p and q are the autoregressive and moving average orders, while P and Q are the seasonal autoregressive and moving average orders. The polynomials of the autoregressive and the moving average processes are accounted for by $\phi(B)$ and $\theta(B)$, whereas $\Phi(B)$ and $\Theta(B)$ represent the polynomials of the seasonal autoregressive and moving average parts of the model. The difference components of the models, which stand for the process of differencing the time series to make it stationary, are represented by the terms $\nabla^d x_t$ and ∇_S^D. (Shumway and Stoffer, 2025, p. 162)

3.3.2 Order selection

The quality of a SARIMA model largely depends on the selection of the order parameters p and q, as well as, if the model includes seasonal components, the seasonal order parameters P and Q. There are different ways of determining the best values for these hyperparameters. Atwan (2022, p. 338) mentions that the best hyperparameters can be found via a grid search, where different hyperparameter values are tried out and those which lead to the best result are chosen. A more common way in time series forecasting of determining the best hyperparameter values is, however, using the so-called Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) (Atwan, 2022, p. 339).

Atwan (2022, p. 339 ff.) explains the process of identifying the best-fitting hyperparameters based on the ACF and PACF as follows:

At first, it needs to be ensured that the time series is stationary. If the time series is not yet stationary, it needs to be transformed to a stationary series by differencing it. If there are seasonal effects in the time series, seasonal differencing needs to be applied additionally, which means that the previous season's value always has to be subtracted from the value of the current season. Subsequently, the ACF and PACF value can be plotted, showing the correlation values between -1 and 1 on the y-axis and the lags on the x-axis. Additionally, the 95% confidence interval is usually shown in the plot, in order to make it possible to identify lags with significant correlations. The last lag with a significant PACF value before they cut off is the optimal p value and the last lag with a significant ACF value before cutting of should be chosen as the q value. Choosing the optimal seasonal order hyperparameters P and Q works similarly, only that they are chosen based on the last significant lags after S, 2S and so forth. (Atwan, 2022, p. 339 ff.)

Another way of identifying the model with the best hyperparameter settings is via so-called penalty factors (Brockwell and Davis, 2016, p. 149). According to Brockwell and Davis (2016, p. 149), the idea behind this evaluation technique is to penalise models with higher orders, as the higher the autoregressive and moving average orders, the more likely the model is to overfit. One of those criteria that can be used for the model selection process is the AIC criterion introduced by Akaike in

1973 (Brockwell and Davis, 2016, p.151). The Akaike Information Criterion (AIC) tries to find a model which fits the data well, but is not too complex and therefore does not overfit the training data (Atwan, 2022, p. 367). The Bayesian Information Criterion (BIC) is similar to the AIC, but its focus is even more on preferring less complex models with lower orders (Atwan, 2022, p. 368). Atwan (2022, p. 361) points out, that these information criteria do not have a meaning on their own in terms of quality of the model, but can only be used to compare different models with each other.

3.3.3 Software implementation

The SARIMA approach is, among others, implemented in the `R`-package `forecast` (Hyndman et al., 2026; Hyndman and Khandakar, 2008). In this study the `auto.arima` function is used to automatically determine a SARIMA model with optimal hyperparameters.

In the article published in addition to the package itself, Hyndman and Khandakar (2008, p. 10) elaborate on how the best-fitting hyperparameters are determined in their function:

The differencing hyperparameters are established using statistical tests. How many times the time series needs to be differenced to become stationary is determined using the KPSS test originally introduced by Kwiatkowski et al. (1992) and the seasonal differencing hyperparameter is chosen based on the Canova-Hansen test presented in Canova and Hansen (1995). The optimal values of p, q, P and Q are established through choosing the configuration with a minimal AIC. (Hyndman and Khandakar, 2008, p. 10)

As evaluating and comparing a large number of models with different hyperparameter settings can be computationally costly and therefore slow, the `auto.arima` function also offers an option for a stepwise model selection, which is, however, not used in this study (Hyndman and Khandakar, 2008, p. 10 f.) (Hyndman et al., 2026, p. 16). As computational time is not the main focus of this study and a non-stepwise model selection still only takes around 6 minutes for all 30 time series at hand, the parameter `stepwise` is set to `FALSE`. Future forecasts can be generated applying the `forecast` function on the model (Hyndman et al., 2026, p. 18).

3.4 The Neural Network approaches

While traditional approaches such Holt-Winters or SARIMA sometimes have difficulties predicting volatile, non-stationary time series, more advanced Machine Learning techniques, such as Neural Networks, have emerged which might have the power to capture the complex relations in such data structures more adequately (Khan et al., 2023, p. 2). Chollet et al. (2022, Chapter 1.1.3) describes Machine Learning as the search for reasonable schemas and rules considering the input data and a certain range of possible options as well as a feedback signal. Machine Learning is considered to be a subfield of the broader field of Artificial Intelligence (AI) (Nandi and Pal, 2022, p. 1). While Neural Networks are to some extent based on the idea of simulating the biological mechanism in human's brains (Aggarwal, 2023, p. 1), Chollet et al. (2022, Chapter 1.1.4) points out that modern Neural Networks do not have many similarities with how a human's brain works. According to a different perspective, the inspiration to Neural Networks rather has come from simpler models such as Linear and Logistic Regression and that Neural Networks are simply a higher-level abstraction of those (Aggarwal, 2023, p. 2).

3.4.1 The concept of Neural Networks

Aggarwal (2023, p. 1) describes the concept of Neural Networks in simple terms as transforming the inputs to the output using a function, while each input is multiplied with a certain weight. These weights serve as the parameters, which can be optimised (Aggarwal, 2023, p. 1). Neural Networks are usually trained in iterations and the weights are adapted in each iteration in order to minimise the prediction error (Aggarwal, 2023, p. 1 f.).

Neural networks can have different architectures. The simplest type of Neural Network is called a Perceptron and it only consists of one input layer, which is directly connected to the output node (Aggarwal, 2023, p. 5). This type of Neural Network is also displayed in Figure 3.2, which has been reproduced from Hyndman and Athanasopulos (2021, p. 12.4). This kind of Neural Network corresponds to a classical Linear Regression with four predictor variables, which are combined linearly and the weights of the Neural Network are equal to the regression's estimated coefficients (Hyndman and Athanasopulos, 2021, Chapter 12.4). Additionally, a constant can be added to the sum of the weights multiplied by the input variables, which is called bias and can be compared to an intercept in a classical linear regression

(Venkateswaran and Ciaburro, 2017, p. 12).

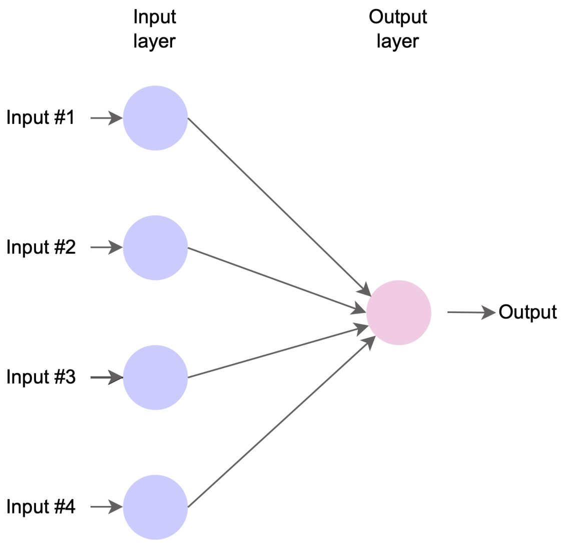


Figure 3.1: The concept of Neural Networks with no hidden layer - figure reproduced from Hyndman and Athanasopulos (2021, Chapter 12.4, Figure 12.15), with draw.io (`https://app.diagrams.net/`)

Neural Networks, however, usually do not only consist of one single input layer and the output node, but there is at least one layer between them. These layers are called hidden layers and a Neural Network with at least one hidden layer is called a Multilayer Neural Network (Hyndman and Athanasopulos, 2021, Chapter 12.4). Such a Neural Network is illustrated in Figure 3.2, which is adapted from Hyndman and Athanasopulos (2021, Chapter 12.4).

The process of how the input is transformed to an output in such a Multilayer Neural Network is described by Hyndman and Athanasopulos (2021, Chapter 12.4) as follows:

Each node of the hidden layer receives its input from the nodes in the preceding layer. Subsequently, the nodes in the hidden layer combine their input values using a linear equation of the following form:

$$z_j = b_j + \sum_{i=1}^{p} w_{ij} x_i \tag{3.24}$$

In Equation 3.24 b_j denotes the bias of node j in the hidden layer, p is the number of nodes in the preceding layer, w_{ij} are the estimated weights and x_i are the input values from the nodes of the foregoing layer. (Hyndman and Athanasopulos, 2021, Chapter 12.4)

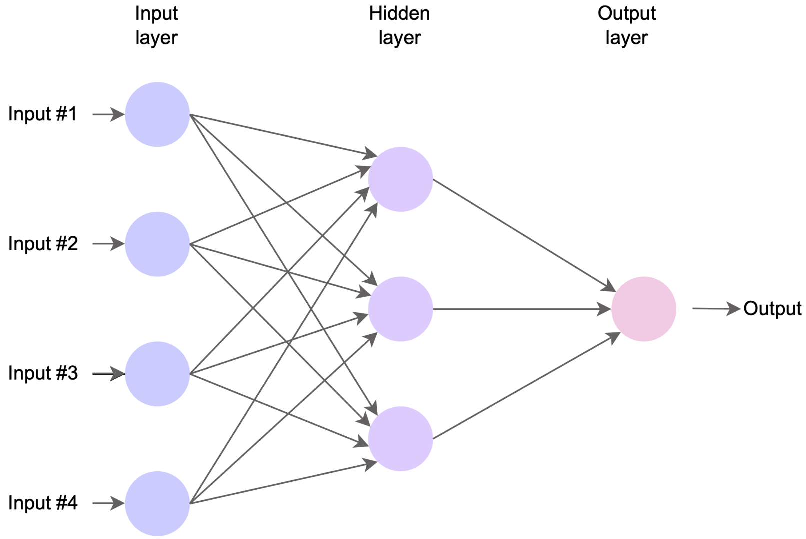


Figure 3.2: The concept of Neural Networks with one hidden layer - figure reproduced from Hyndman and Athanasopulos (2021, Chapter 12.4, Figure 12.16), with draw.io (`https://app.diagrams.net/`)

A Neural Network which only combines inputs according to Equation 3.24 is, however, still linear. The non-linearity is introduced via a non-linear function which is applied on z_i (Hyndman and Athanasopulos, 2021, Chapter 12.4). There are different so-called activation functions, such as the sign function, the tanh function or the sigmoid function (Aggarwal, 2023, p. 10). According to Aggarwal (2023, p. 10) and Hyndman and Athanasopulos (2021, Chapter 12.4), the latter is given as:

$$s(z) = \frac{1}{1 + e^{-z}} \tag{3.25}$$

In Equation 3.25 e stands for Euler's number.

The parameter values for b_j and the weights are determined based on the data, while the initial starting values are usually chosen randomly (Hyndman and Athanasopulos, 2021, Chapter 12.4). Due to this randomness, Hyndman and Athanasopulos (2021, Chapter 12.4) point out that Neural Networks are typically trained multiple times and the results are then combined using the average. Neural Networks are usually trained in multiple steps, where the weights are optimised using the so-called backpropagation method, which means that the weights are adapted based on the error between the calculated output and the real output at each iteration (Venkateswaran and Ciaburro, 2017, p. 53 ff.).

Some Neural Networks only consist of one hidden layer as displayed in Figure 3.2, especially if they are applied to less complex tasks. Most Neural Networks, however, consist of more than one hidden layer arranged in a sequence after each other, as displayed in Figure 3.3. A Neural Network with multiple hidden layers can also be called a Multilayer Perceptron (MLP) (Venkateswaran and Ciaburro, 2017, p. 19). At each layer the input values from the previous layer are transformed through Equation 3.24 and after applying the activation function the values are passed on to the next layer (Venkateswaran and Ciaburro, 2017, p. 20).

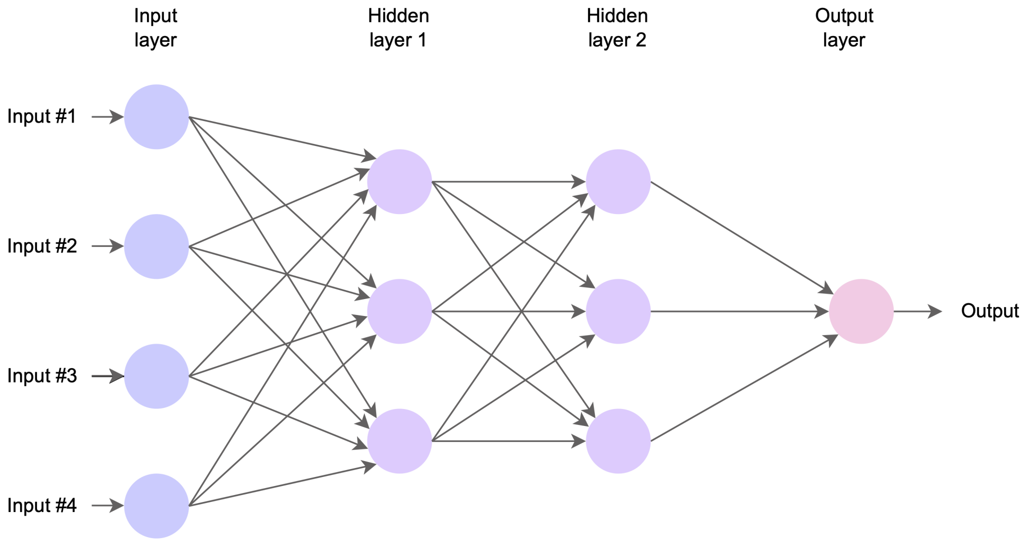


Figure 3.3: The concept of Neural Networks with two hidden layers - figure reproduced from Hyndman and Athanasopulos (2021, Chapter 12.4, Figure 12.16), and adapted to two hidden layers with draw.io (`https://app.diagrams.net/`)

Neural networks with a single hidden layer as well as Multilayer Perceptrons are also frequently applied in the literature. Talkhi et al. (2021), Khan et al. (2023) and Khadka et al. (2025), for example, use the `R`-function `nnetar`, which is explained in more detail in Subsection 3.4.2, for fitting neural networks with a single hidden layer for various different time series forecasting tasks in the field of health, finance and agricultural economics. Multilayer Perceptrons are applied in numerous studies too, as outlined in Mediavilla et al. (2022, p. 1128). In Talkhi et al. (2021, p. 7), for example, the Multilayer Perceptron outperformed traditional approaches, such as Holt-Winters and SARIMA.

3.4.2 The software implementation: nnetar of package `forecast`

In the `R`-package `forecast` a function called `nnetar` is included, which was written by Rob J. Hyndman and Gabriel Caceres and uses a Feed-Forward Neural Network with one hidden layer (see Figure 3.2) for prediction (Hyndman et al., 2026, p. 111 ff.). According to the documentation the implemented algorithm is especially tailored to predicting univariate time series via lagging the values to transform the time series data to a format which can serve as an input for Neural Networks (Hyndman et al., 2026, p. 111 ff.). Such a model is called Neural Network Autoregression (NNAR) model (Hyndman and Athanasopulos, 2021, Chapter 12.4).

In their book Hyndman and Athanasopulos (2021, Chapter 12.4) mention that the `nnetar` function in `R` fits $NNAR(p, P, k)_m$ models, where p denotes the lagged inputs, P represents the seasonal lagged inputs, k stands for the number of nodes used in the hidden layer of the Neural Network and m specifies the seasonal pattern, for example 12 for monthly data. If the k is set to 0, the model corresponds to an $ARIMA(p, 0, 0)(P, 0, 0)_m$ model without the stationarity requirement (Hyndman and Athanasopulos, 2021, Chapter 12.4). According to Hyndman and Athanasopulos (2021, Chapter 12.4) the default hyperparameter for seasonal time series is $P = 1$, while p is optimised by fitting a linear model to the data after removing seasonal effects and choosing the best hyperparameter value based on the AIC criterion. The default value for the number of nodes in the hidden layer k is given as the rounded value of $(p + P + 1)/2$ (Hyndman and Athanasopulos, 2021, Chapter 12.4). In the function of the package `forecast` the hyperparameter k is denoted as `size` (Hyndman et al., 2026, p. 112).

Forecasting future values using the fitted `nnetar` model can be done using the function `forecast.nnetar` (Hyndman et al., 2026, p. 70 ff.). Just like for the SARIMA models, the forecasting process for the `nnetar` function works recursively, meaning that for predicting one future value only the historic training data is used, but the prediction of $t = 2, ..., h$ is also based on the previously forecasted values (Hyndman and Athanasopulos, 2021, Chapter 12.4).

3.4.3 The software implementation: mlp of package `nnfor`

`nnfor` is a specialised `R`-package for time series prediction using Neural Networks created by Nikolaos Kourentzes including a function for fitting Multilayer Perceptron

(MLP) models (Kourentzes, 2023, p. 12 ff.). In contrast to the `nnetar` implementation of the `forecast` package, which combines the results of the multiple trained networks using the average (Hyndman et al., 2026, p. 112), the `mlp` implementation of the `nnfor` package combines the results using the median by default (Kourentzes, 2023, p. 12 f.). The `hd` argument allows the specification of the number of hidden layers as well as of the number of nodes in each hidden layer (Kourentzes, 2023, p. 13). While the function `mlp` allows an automatic optimisation of the hyperparameter `hd` through setting the parameter `hd.auto.type` either to `valid` using a 20% validation set, or to `cv` using 5-fold cross-validation (Kourentzes, 2023, p. 14), in order to have more control over the hyperparameter tuning process the hyperparameter `hd` is manually optimised in this study. This is done via using the last 12 values of the training dataset as a validation dataset and choosing the value of `hd`, which minimises the Weighted Mean Absolute Percentage Error (WMAPE) on this validation dataset. Additionally, through the `sel.lag` parameter, the lags can be automatically chosen (Kourentzes, 2023, p. 13), which is also deactivated for this study, as all of the time series at hand consist of monthly data and therefore the frequency of each time series is 12.

Similarly to the `nnetar` function, future values of the time series can be predicted using the function `forecast.mlp`, and the argument `h` represents the forecasting horizon, which is 12 to predict one year (Kourentzes, 2023, p. 10).

4. Data preparation

4.1 Data Preprocessing

As the company works with more than 4000 different parts, at first a number of parts has to be chosen to carry out the analysis on. On the 19th of August 2025 4910 different parts were registered in the ERP system of the company. To choose the most relevant parts for the analysis, the parts were sorted directly in the ERP software of the company based on purchase value and purchase amount in 2024 and the 15 parts with the highest purchase value in 2024 and the 15 parts most purchased in 2024 were chosen to be further worked with. Ordering the articles in the ERP system based on the number of outgoing items was, as of August 2025, not possible.

For each of these parts a dataset could be exported from the ERP system looking like in Table 4.1. For anonymisation purposes, the order numbers displayed in Table 4.1 do not correspond to the real order numbers in the data.

Date	**Quantity**	**Order number**	**Booking type**
2025-05-26	3	11111	Delivered
2025-05-26	1	22222	Delivered
2025-05-26	40		Receipt
2025-05-23	2	33333	Cancelled
2025-05-23	2	33333	Delivered
2025-05-23	1	44444	Delivered

Table 4.1: Extract of the raw data

It should be noted that the raw article names are not published in this study due to confidentiality requirements by the company. As it is, however, important to understand if the analysed article is a built-in component in all stations, such as a heat exchanger or an actuator, or if it is a small fastener, such as a screw or a nut, the articles at hand are grouped into six categories. The groups of parts are as follows: .

- **Category 1:** Main components of heat transfer stations - heat exchangers, meters, regulators and actuators
- **Category 2:** Valves
- **Category 3:** Gaskets and seals
- **Category 4:** Small electrical parts
- **Category 5:** Fasteners - screws, nuts, nails and accessories
- **Category 6:** Network data cables

The datasets have different time spans. While they are all cut off at the 30th of June 2025, they start at different points in time. None of the datasets, however, starts earlier than January 2018 as the company introduced the current ERP system only at the beginning of 2018.

Table 4.2 shows the 15 parts with the highest purchase values in 2024, their article categories and the date of their first data entry. The purchase value of each article is defined as the purchased quantity multiplied by the purchase price. Due to confidentiality requirements raw prices of parts are not published. The last column shows the item ID which is simply the abbreviation "pv" for "purchase value" and the number in the ranking according to it. This ID is later used in the tables and results in Section 4.2 and in Chapter 6 as an identifier.

Ranking value	Article category	Start date	Item ID
1	Category 1 - heat exchanger	2021-09-21	pv1
2	Category 1 - meter	2018-01-10	pv2
3	Category 1 - meter	2018-01-12	pv3
4	Category 1 - regulator	2018-01-09	pv4
5	Category 1 - meter	2018-01-09	pv5
6	Category 1 - actuator	2018-01-15	pv6
7	Category 2 - valve	2018-01-09	pv7
8	Category 1 - actuator	2018-01-12	pv8
9	Category 2 - valve	2022-01-19	pv9
10	Category 2 - valve	2018-01-12	pv10
11	Category 1 - regulator	2018-01-09	pv11
12	Category 2 - valve	2018-01-15	pv12
13	Category 2 - valve	2018-01-12	pv13
14	Category 1 - heat exchanger	2018-01-22	pv14
15	Category 1 - meter	2020-09-01	pv15

Table 4.2: The 15 parts with the highest purchase value in 2024 and the date of the first data entry

Table 4.3 shows the 15 most purchased articles by the company during the year 2024, their article categories, the date of their first data entry as well as the item ID starting with "pa" for "purchase amount".

Ranking amount	Article category	Start date	Item ID
1	Category 3 - gaskets / seals	2020-05-06	pa1
2	Category 4 - small electrical parts	2022-02-01	pa2
3	Category 3 - gaskets / seals	2020-05-05	pa3
4	Category 4 - small electrical parts	2018-06-25	pa4
5	Category 6 - network data cable	2018-02-08	pa5
6	Category 4 - small electrical parts	2020-05-14	pa6
7	Category 5 - screw accessories	2020-05-04	pa7
8	Category 5 - screws	2018-06-27	pa8
9	Category 5 - screw accessories	2020-05-11	pa9
10	Category 5 - screw accessories	2020-05-14	pa10
11	Category 3 - gaskets / seals	2020-05-07	pa11
12	Category 5 - screw accessories	2018-03-22	pa12
13	Category 4 - small electrical parts	2020-05-29	pa13
14	Category 4 - small electrical parts	2020-07-23	pa14
15	Category 4 - small electrical parts	2018-03-22	pa15

Table 4.3: The 15 most purchased parts in 2024 and the date of the first data entry

In order to ensure a fair comparison between the models, articles are only considered if they have a data history from at least July 2022 until June 2025. Some methods such as Holt-Winters and the Multilayer Perceptron in `R` also need at least two full years of monthly data to fit a model and one year of data is used as a test set. For the group of articles with the highest purchase value, therefore twelve articles had to be filtered out, including five articles which would belong to a seventh group of parts representing the big frames where the stations are built in. For the group of articles with the highest purchase amount one article had to be filtered out due to not enough training data and three articles were excluded, because their last data entry was in June 2024, December 2024 and February 2025 respectively and forecasting the demand of parts, which are not in use any more, is not of relevance.

Each article's dataset is preprocessed using the statistical programming language `R` (R Core Team, 2025). The majority of preprocessing steps is performed using `R` built-in functions and the functionalities of the package `dplyr` (Wickham et al., 2026b). Other packages used to prepare the data for the modelling step are `readxl` (Wickham and Bryan, 2025), `padr` (Thoen, 2024) and `lubridate` (Spinu et al., 2026; Grolemund and Wickham, 2011).

As displayed in Table 4.1 the datasets also include purchases, cancellations as well as transfers between warehouses. Therefore, the dataset is filtered to only include bookings of the type "delivered" (in the original dataset in German language "Ausgeliefert").

In the next step all rows which have either a missing value in the date column, in the quantity column or in the order number column are dropped from the dataset as these three values are required for further processing and the forecast.

As there can be multiple delivery bookings per day for the same item, these are aggregated to a daily level. This means that the quantities of all delivered entries on the same day are summed up, which leads to a table only containing at most one entry per day and article.

Subsequently, the dataset is corrected for cancellations. As each cancellation belongs to a specific order, this is done order by order. Cancellations do not always take place on the same date as the original entry they belong to, which makes mapping via the order number and date impossible. Therefore, the correction for cancellations must be done row by row and works the following way:

The entries of each order are processed from the entry with the latest date to the entry with the earliest date. If an order includes a cancellation booking, the quantity of the respective cancellation booking is subtracted from the quantity of the last delivery booking before the cancellation. If the cancellation quantity is higher than the quantity of the last delivery booking, the remaining items are subtracted from the quantity of the penultimate delivery entry and the subsequent entries if necessary. Table 4.4 and 4.5 show examples for this process.

Table 4.4 shows cancellation bookings assigned to two different orders. Due to anonymisation purposes, the displayed order numbers do not correspond to the real order numbers in the data.

Date	Quantity	Order number	Booking type
2024-11-12	2	55555	Cancelled
2024-10-28	6	66666	Cancelled
2024-10-28	1	66666	Cancelled

Table 4.4: Cancellation bookings

Table 4.5 shows the original bookings of the delivered parts belonging to the same orders as well as the new quantities corrected for the cancellation entries.

Date	Quantity	Corrected quantity	Order number	Booking type
2024-11-21	2	2	55555	Delivered
2024-11-12	3	1	55555	Delivered
2024-10-28	6	0	66666	Delivered
2024-10-25	1	0	66666	Delivered
2024-10-24	6	6	66666	Delivered

Table 4.5: Bookings of delivered parts corrected for cancellations

After the correction of the entries for cancellations the data is prepared for the modelling step. To transform the datasets into the format of a time series days without an entry are added with the quantity of zero. This is done using the function `pad` from the `R`-package `padr` (Thoen, 2024).

To allow a monthly forecast, the data is subsequently transformed to a monthly level via grouping the data by month and year and summing up the quantities of the respective entries. As for some articles there can be months without a single entry, the data is padded once more using the `pad` function from the package `padr` (Thoen, 2024). That numerous parts have considerably sparse demand is also the reason why the forecasting is carried out on a monthly level and not on a weekly or even daily level.

In the following the data is transformed to a time series object using the function `ts` of the built-in `R`-package `stats`. According to its documentation the function takes the frequency of the time series, which is for example 12 for monthly data, as well as the start date of the time series, such as September of the year 2021, as arguments (R Core Team, 2026d). The conversion of the data into a `ts`-object is important, as some modelling implementations such as `mlp` do not accept a classical vector, matrix or data frame as an input (Kourentzes, 2023, p. 13).

4.2 XYZ Analysis

Before fitting any models to the data an XYZ-analysis is performed on the time series, in order to get an overview of their variability and their demand patterns. The analysis of demand patterns also gives an insight into how well prediction models are expected to perform on the respective time series as the accuracy of prediction methods depends on the properties of the demand data (Kostenko and Hyndman, 2006, p. 1256).

One business method, which can be used for the classification of demand patterns is the XYZ-analysis. While the ABC-analysis groups the parts according to their importance represented through the turnover, the XYZ-analysis is used to categorise them based on their fluctuations in usage (Scholz-Reiter et al., 2012, p. 445 f.). X-parts are those with relatively low fluctuations, for Y-parts the demand is less regular and Z-parts are characterised through even higher demand fluctuations (Vahrenkamp and Kotzab, 2012, p. 52 f.). For the XYZ-analysis the demand fluctuations are usually quantified using the standard deviation of the time series divided through its average (Scholz-Reiter et al., 2012, p. 446) (Vahrenkamp and Kotzab, 2012, p. 52), which is called the variation coefficient (Scholz-Reiter et al., 2012, p. 446). Mathematically the variation coefficient is therefore given as:

$$CV = \frac{s}{\bar{x}} \tag{4.1}$$

For this study the coefficient of variation is calculated using the `R` base function `mean` and the function `sd` of the built-in package `stats`. According to R Core Team (2026c) in the function `sd` the sample standard deviation is calculated using the denominator of $n-1$ and is therefore given as:

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2} \tag{4.2}$$

There is no unique definition on what the critical values for each group are. Vahrenkamp and Kotzab (2012, p. 52 f.), for example, mention that all items with a variation coefficient of under 0.1 are of class X, parts with a variation coefficient between 0.1 and 0.5 are of group Y and items with a variation coefficient of above 0.5 belong to class Z. In contrast to that Scholz-Reiter et al. (2012, p. 447) defines X-parts as items with a variation coefficient of under 0.5, Y-parts have a variation coefficient between 0.5 and 1 and Z-items have a variation coefficient of above 1. For this study an approach is chosen where the critical values are selected in such a way that an equal number of ten parts is in each of the three groups.

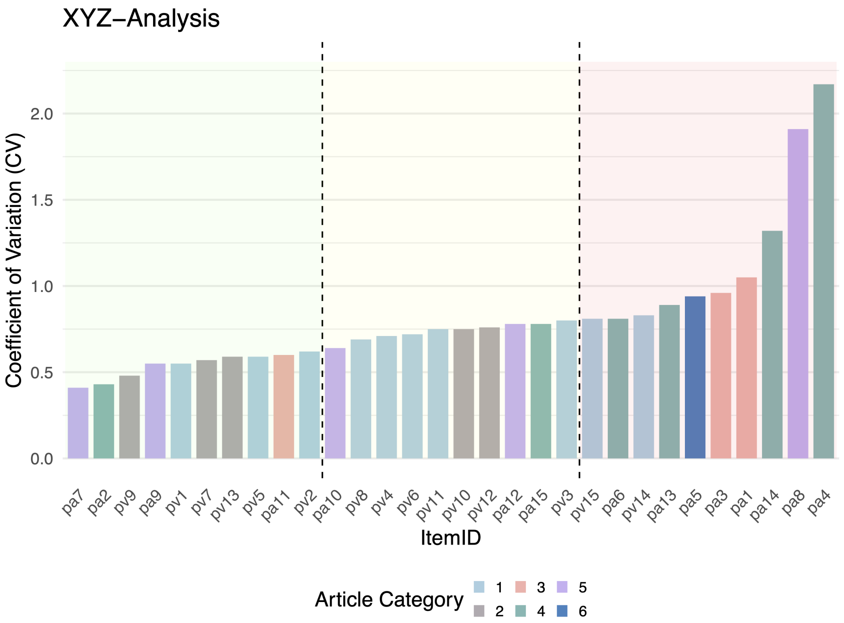


Figure 4.1: Bar plot of the variation coefficient of each part, their article category and the group according to the XYZ-analysis

Figure 4.1 shows the calculated variation coefficients for each part and is sorted in an increasing manner. The colours of the bars show the article category and the groups of the XYZ-analysis are visible via the background colours. Table 4.6 shows the parts and their calculated variation coefficient as well as their group based on the XYZ-analysis.

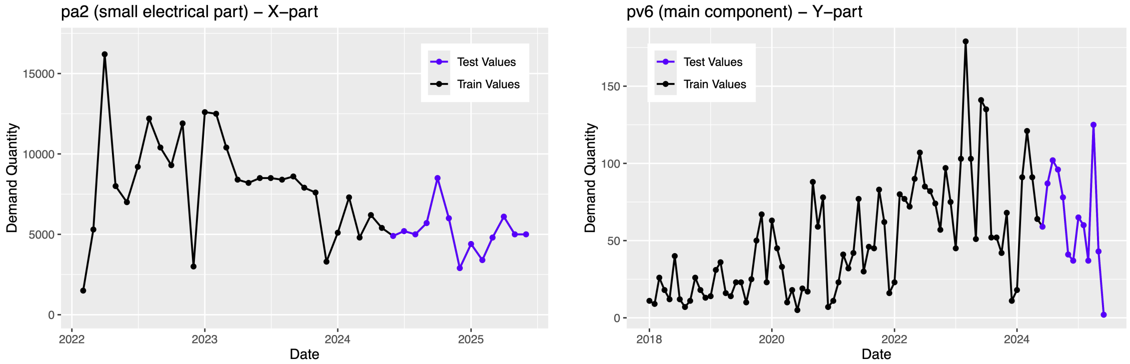
Part ID	**Article category**	**CV**	**XYZ Class**
pa7	Fasteners (5)	0.41	X
pa2	Small electrical parts (4)	0.43	X
pv9	Valves (2)	0.48	X
pa9	Fasteners (5)	0.55	X
pv1	Main components (1)	0.55	X
pv7	Valves (2)	0.57	X
pv13	Valves (2)	0.59	X
pv5	Main components (1)	0.59	X
pa11	Gaskets & seals (3)	0.6	X
pv2	Main components (1)	0.62	X
pa10	Fasteners (5)	0.64	Y
pv8	Main components (1)	0.69	Y
pv4	Main components (1)	0.71	Y
pv6	Main components (1)	0.72	Y
pv11	Main components (1)	0.75	Y
pv10	Valves (2)	0.75	Y
pv12	Valves (2)	0.76	Y
pa12	Fasteners (5)	0.78	Y
pa15	Small electrical parts (4)	0.78	Y
pv3	Main components (1)	0.8	Y
pv15	Main components (1)	0.81	Z
pa6	Small electrical parts (4)	0.81	Z
pv14	Main components (1)	0.83	Z
pa13	Small electrical parts (4)	0.89	Z
pa5	Network data cables (6)	0.94	Z
pa3	Gaskets & seals (3)	0.96	Z
pa1	Gaskets & seals (3)	1.05	Z
pa14	Small electrical parts (4)	1.32	Z
pa8	Fasteners (5)	1.91	Z
pa4	Small electrical parts (4)	2.17	Z

Table 4.6: The parts, their variation coefficient and the group according to the XYZ-analysis

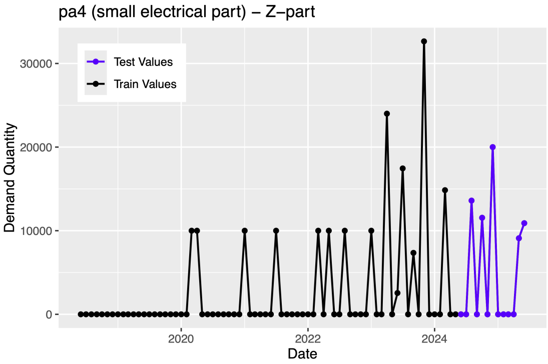
Looking at Figure 4.1 and Table 4.2 assuming that each group should be of equal size it can be inferred that the critical values for the XYZ-analysis are that all parts with a variation coefficient below 0.63 are of group X, all parts with a variation coefficient between 0.63 and 0.8 are of group Y and all articles with a variation coefficient above 0.8 are of group Z.

The XYZ-analysis shows that the majority of parts has a calculated variation coefficient of under 1, while four parts have a variation coefficient higher than 1, of which two parts belong to the article category of Small electrical parts. Looking at the article categories more closely, most parts of category 2 (Valves) belong to group X or group Y, while most main components (category 1) belong to group Y. For group Z, most of the parts are of the categories 3 (Gaskets & seals) and 4 (Small electrical parts). Furthermore, most of the parts in group Z belong to the ranking by purchase amount and not by purchase value. These results make sense, as auxiliary parts such as small electrical parts or gaskets and seals are often retrieved from the warehouse in bigger bundles and are then stored directly at their point of use at the workstations. Therefore, for such parts there are often higher demand spikes, which are followed by months with no demand at all. Such a demand pattern can also be seen in Figure 4.2c. On the contrary, main components or valves are only retrieved from the warehouse when directly needed for assembly, which leads to a more regular demand pattern. Furthermore, these kinds of parts are used in almost every heat transfer station configuration, which might also smooth the demand history. Generally, however, it can be concluded that the variation coefficient is considerably high for all parts with the lowest value being 0.41, which emphasises that despite having a more regular pattern than other parts, even items in the X-class do not have a completely regular demand.

Figure 4.2 shows the demand pattern of three representative parts, one of each group of the XYZ-analysis. In Figure 4.2a it can be seen that the demand varies, but not as much as for the parts of group Y (Figure 4.2b) and group Z (Figure 4.2c). The example part for the Y-group belongs to the category of main components and in Figure 4.2b it can be observed that the variance of the time series varies significantly. It is rather low from 2018 until 2022 and it increases considerably from 2022 onwards until the middle of 2025. Therefore, this part is classified as an Y-part by the XYZ-analysis. Figure 4.2c highlights the typical behaviour of auxiliary parts, such as small electrical parts. Irregular demand spikes can be observed at certain points in time, while for other months the demand is zero. This behaviour occurs, because the parts are retrieved jointly and are then stored directly at the work stations. As soon as there are no such items at the work station any more, another bundle of this item is retrieved from the warehouse. As it can be seen, however, looking at the example for group X in Figure 4.2a, not all small electrical parts behave that way.



(a) X-part: Small electrical part - pa2 (b) Y-parts: Main component - pv6



(c) Z-part: Small electrical part - pa4

Figure 4.2: Plots of example parts for each group of the XYZ-analysis

The XYZ-analysis provides an overview of the different demand patterns present in the data. It might also already indicate, for which parts rather good forecasting results can be expected and for which parts the prediction of the future demand might be more challenging.

At the end of this section, it should also be mentioned that there are also other, more sophisticated, approaches presented in the literature of how to group time series based on their patterns, such as an approach by Syntetos et al. (2005) classifying the datasets based on their average inter-demand intervals as well as based on the squared coefficient of variation. For this thesis, however, the rather business-driven segmentation was chosen, focusing more on interpretability than on methodological complexity.

5. Evaluation

In order to test the different models, the time series are split into a training and test set. For each article's time series the test set is the data from July 2024 until June 2025, which means that the different models are evaluated on the data of a one year period.

5.1 Evaluation strategies

Each of the models fitted to the data, including the two classical approaches Holt-Winters and SARIMA and the two AI-based Neural Network approaches, is evaluated in three ways:

- Evaluation based on a 12-months in advance forecast
- Evaluation based on a 12-months rolling forecast, always predicting the following month
- Evaluation based on a yearly forecast, through the 12-months in advance forecast summed up to a year

For the evaluation based on the 12-months in advance forecast, each of the fitted models predicts the following 12 months in one single step. This means that at the end of June 2024, where the training data ends, the demand values for the upcoming 12 months from July 2024 until June 2025 are predicted and then compared with the real values of the test dataset. Predictions in advance are important for the company to adequately plan their resources and to order items in advance which have longer lead times. The company, however, as of the beginning of 2026 does not have any reliable and continuously updated data on the lead times of the parts. How much a certain item has to be ordered in advance is currently decided solely based on the expertise of the employees.

The 12-months rolling forecast always predicts the following month based on all the available data. This means that to forecast the demand value of July 2024 the training data until June 2024 is used, while to forecast the demand of August 2024 all the data until July 2024 is used for training. To forecast the value of June 2025, therefore 11 more values are used for training than for the 12-months in advance forecast. The logic of the rolling forecast is also visualised in Figure 5.1. This forecasting scheme is particularly useful for parts with short lead times and if there is inventory available for these items. Its advantage is that it also allows the company to adapt its resource planning based on most recent developments, which is specifically useful if the demand pattern is rather volatile. A rolling forecast also makes sense from a mathematical point of view, as due to a higher level of uncertainty predictions tend to become less accurate the more in advance they are generated (Hyndman and Athanasopulos, 2021, Chapter 5.5).

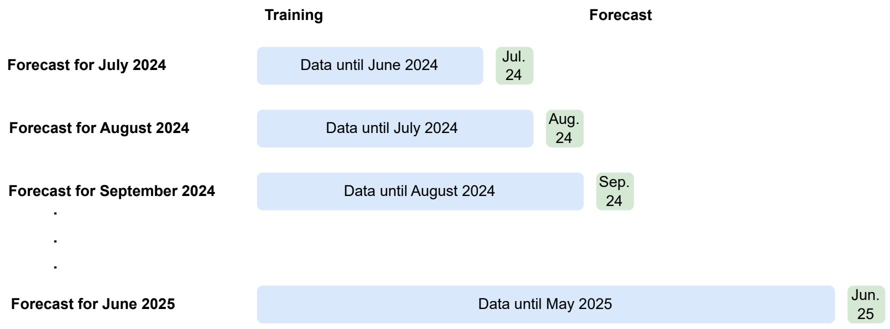


Figure 5.1: Visualisation of the logic of the monthly rolling forecast, created with draw.io (`https://app.diagrams.net/`)

Additionally, a yearly forecast is generated, which is based on the monthly in advance predictions. A direct forecast of the yearly sum using the data of the previous years is not possible due to the short time frame of the training data. The longest training data ranges back to the beginning of 2018. Therefore the models would need to be trained only on the basis of 7 values from 2018 until 2024 at maximum, which is too little to produce reasonable results. Therefore, to create a forecast of a whole year the monthly forecasted values from July 2024 until June 2025 are summed up. The accuracy of this forecast is specifically useful for the company, as deals with suppliers are often agreed on for longer periods of time and the supplier might want to receive an estimation of how much items are required during the period of a whole year.

5.2 Baseline model

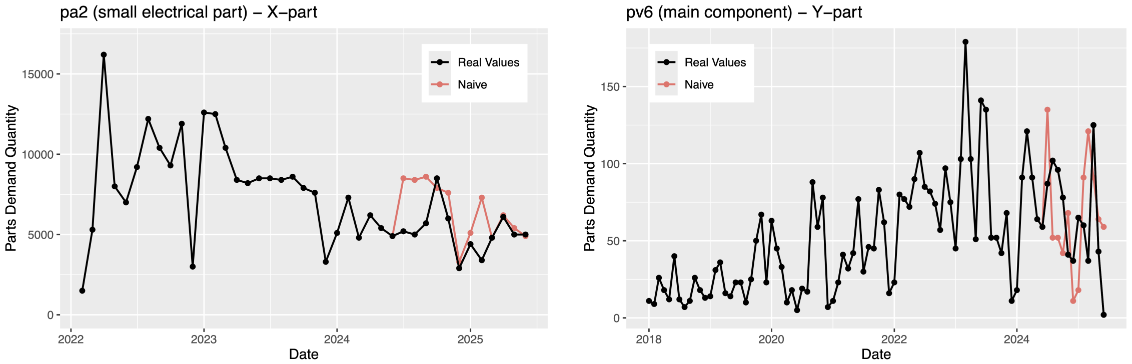
An important aspect considered by a company when introducing a new forecasting tool might be that it performs better than the currently used tool. The company Aqotec, however, does currently as of the beginning of 2026 not use any demand forecasting tools. Resource planning and the placement of orders is carried out simply based on the expertise of the responsible employees. In unfavourable circumstances orders might even only be placed when a certain item runs out, which might delay the completion of products.

In order to still be able to compare the results of the applied forecasting models with a certain baseline approach, which is important to assess the added value for the company, it is assumed that the employees of the company base their demand estimations on the demand of the same month in the preceding year. In the literature this kind of time series prediction model is called the Seasonal Naïve method (Hyndman and Athanasopulos, 2021, Chapter 5.2). According to Hyndman and Athanasopulos (2021, Chapter 5.2) and slightly adapted to match monthly forecasts and to match the notation in this study, this model can be mathematically expressed as follows:

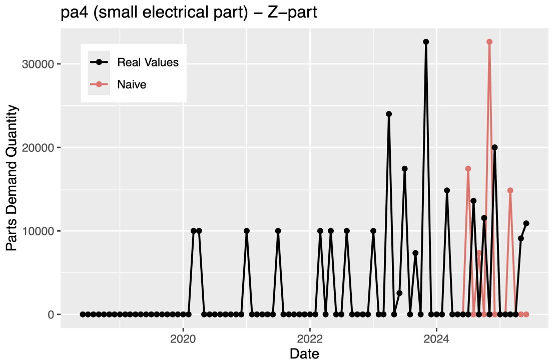
$$\hat{y}_{n+t} = y_{n+t-12}, \qquad t = 1, \ldots, 12. \tag{5.1}$$

In Equation 5.1 n is the number of training observations and t represents which future value is predicted, which is 1 if the first month after the training period is predicted and 12 if the twelfth month after the training period is predicted.

Figure 5.2 shows the real values and the predicted values by the baseline model for the three example parts also shown in Figure 4.2. Looking at the forecasts it becomes obvious that while the demand for some months can be accurately predicted using this simple baseline approach, such as the demand for the months October and December 2024 for the part `pa2`, the actual demand of other months, such as July or August 2024, does not match the value of the previous year (see Figure 5.2a). Therefore, resource planning and reordering of the parts based on this simple baseline approach might lead to massively underestimating or overestimating the monthly parts demand.



(a) X-part: Small electrical part - pa2 (b) Y-part: Main component - pv6



(c) Z-part: Small electrical part - pa4

Figure 5.2: Plots of example parts for each group of the XYZ-analysis including the naïve seasonal forecast

5.3 Evaluation metrics

The quality of the monthly forecasted values, both for the in-advance predictions as well as for the rolling forecasts, is evaluated using the Weighted Mean Absolute Percentage Error (WMAPE). These errors are partly also compared with the Symmetric Mean Absolute Percentage Errors (SMAPE) of the predictions.

The WMAPE was introduced by Kolassa and Schütz (2007) and was originally called MAD/Mean ratio (Hyndman, 2025), which stands for the Mean Absolute Deviation or Mean Absolute Error divided through the mean of the actual values (Kim and Kim, 2016, p. 670). Its formula according to Hyndman (2025) when multiplied with 100 to be shown as a percentage value is therefore given as:

$$100 \cdot \frac{\sum_t |y_t - \hat{y}_t|}{\sum_t |y_t|} \tag{5.2}$$

Compared to the Mean Absolute Percentage Error (MAPE), which cannot be calculated if any of the actual values is zero, the WMAPE is defined as the mean absolute error divided by the sum of the actual values, which also allows single actual values to be zero (Kim and Kim, 2016, p. 670) (Hyndman, 2025). This property is necessary to be able to compare the errors across all analysed time series, because as shown in Figure 4.2c, especially series of the Z-group tend to have numerous months with zero demand. An additional characteristic of the WMAPE is that the forecast errors of months with a higher actual demand have a larger influence on the calculated WMAPE (Kolassa and Schütz, 2016, p. 213). The division through the sum of the actual values is the reason that months with higher actual values have a higher associated weight within the error metric than months with lower actual values (Kolassa and Schütz, 2016, p. 213). That the WMAPE puts more weight on months with higher actual demand makes sense from a business-perspective as forecasting a demand of 50 instead of the actual 100 required parts has a much higher negative impact than forecasting a demand of one instead of actually two required parts. A similar example is also discussed in (Kolassa and Schütz, 2016, p. 213).

The WMAPE has been used in a number of recent studies, such as in Ghribi et al. (2025) or in El-Meehy et al. (2025). Studies such as Giannopoulos et al. (2025, p. 22) or Mediavilla et al. (2022, p. 1128), which analyse a variety of papers in different industries dealing with demand forecasting, however suggest that scale-dependent measures such as the Root Mean Squared Error (RMSE) and the Mean Absolute Error (MAE) tend to be the preferred metrics. Nevertheless, in order to keep the results comparable across different parts with demands on different scales in this study percentage-based metrics are favoured over scale-dependent ones.

Due to the point of criticism on the WMAPE, that it puts more weight on months with a higher actual demand, some results are additionally evaluated using the Symmetric Mean Absolute Percentage Error (SMAPE). The SMAPE is based on the work by Armstrong (1978) and Makridakis (1993), as mentioned by Hyndman (2025, Chapter 5.8) and Kim and Kim (2016, p. 670). Hyndman and Athanasopulos (2021, Chapter 5.8) defines the SMAPE as follows:

$$\frac{1}{n}\sum_{t=1}^{n}\frac{200\,|y_t-\hat{y}_t|}{y_t+\hat{y}_t} \tag{5.3}$$

which can also be rewritten as follows:

$$\frac{100}{n}\sum_{t=1}^{n}\frac{|y_t-\hat{y}_t|}{(y_t+\hat{y}_t)/2} \tag{5.4}$$

The SMAPE is also a modification of the MAPE, where each absolute forecast error is divided by the average of the actual and the predicted values summed up and it is therefore also defined if either the actual or the forecasted value is zero (Kim and Kim, 2016, p. 670). It should be noted that in this study, if for single observations both the predicted value and the actual value are zero, an error of zero is assumed, even though mathematically a division through zero is not defined. The SMAPE has the characteristic that it weighs all the errors, independent of underestimation and overestimation, equally (Hyndman and Athanasopulos, 2021, Chapter 5.8). The measure, however, becomes unstable when a lot of actual and predicted values are close to zero (Hyndman and Athanasopulos, 2021, Chapter 5.8).

Due to the advantages and disadvantages of both measures, some results are evaluated based on both the WMAPE and the SMAPE in order to provide a more robust analysis.

The yearly forecast values are compared based on the classic Percentage Error (PE) as always only two values, the actual and the forecasted value, are compared with each other. As the predicted value is compared to the summed up yearly demand the problem regarding the division through zero cannot occur, because if there was no demand for the item in the last year, it would not be included in this analysis (see Chapter 4). Mathematically, this PE can be expressed the following way:

$$100 \cdot \frac{\sum_{t=1}^{n} y_t - \sum_{t=1}^{n} \hat{y}_t}{\sum_{t=1}^{n} y_t} \tag{5.5}$$

According to Equation 5.5, negative PEs mean that the actual demand was less than the predicted demand and positive PEs mean that the actual demand was more than the forecasted demand. It should also be noted that when comparing Equation 5.5 and Equation 5.2, it becomes clear that when only comparing one actual and one predicted value with each other, the PE exactly corresponds to the WMAPE with the only difference that the PE can also be negative.

6. Results

6.1 Forecasting results

The four time series forecasting methods Holt-Winters, SARIMA, the Neural Network Autoregression as well as the Multilayer Perceptron are applied on the training sets of the 30 available time series using the software `R`. In order to ensure reproducibility of the results of the Neural Network approaches, a fixed seed is set in the code. Table 6.1 shows the applied `R`-functions, how the hyperparameters are chosen for each method and the seeds which have been set. The applied functions are described in more detail in the Sections 3.2.2, 3.3.3, 3.4.2 and 3.4.3. The actual hyperparameters chosen during the training process can be found in Table 8.5 in the Appendix.

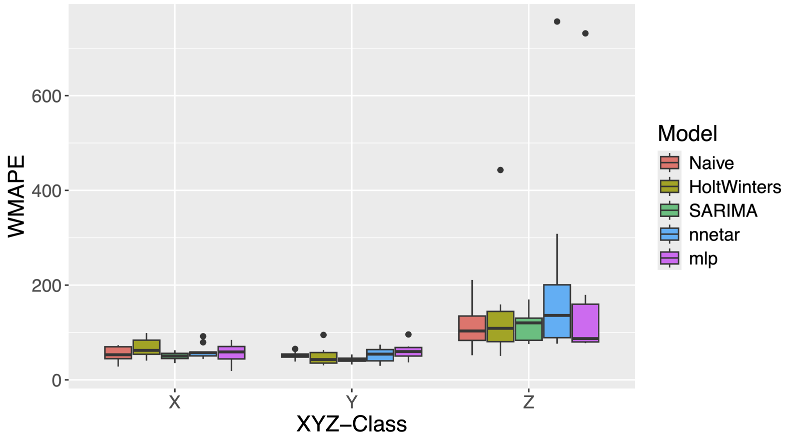
Method	**R function**	**(Hyper)parameter selection**	**Seed**
Holt-Winters	`HoltWinters` (`stats`)	Automatic selection of α, β, γ	-
SARIMA	`auto.arima` (`forecast`)	Automatic selection of p, q, P, Q, `stepwise = FALSE`	-
`Neural Network Autoregression`	`nnetar` (`forecast`)	Automatic selection of p, P and `size`	555
`Multilayer Perceptron`	`mlp` (`nnfor`)	`sel.lag = FALSE`, Selection of `hd` based on the WMPAE out of `c(6)`, `c(12)`, `c(8, 4)`, `c(12, 6)`	555

Table 6.1: Overview of the applied forecasting methods in `R`

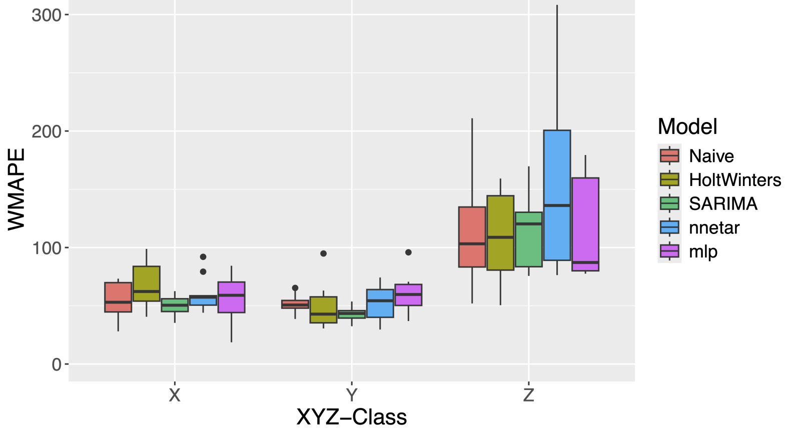
After applying the forecasting methods to the time series, the predicted values are rounded up, because only whole numbers make sense as quantities of parts. Furthermore, if negative amounts are predicted by some methods due to, for example, negative trends present in the time series, the predicted values are set to zero, as negative demand quantities do not make sense.

Even though the main focus of this study is rather on the accuracy of the models and not on the optimisation of computational resources and computation times, it should be briefly mentioned that while fitting a Holt-Winters model to all 30 time series including the optimisation of the parameters only took around 0.08 seconds, fitting a SARIMA model using the `auto.arima` function to all 30 datasets took 5.57 minutes. Fitting and optimising the nnetar model for 30 time series took around 0.9 seconds and fitting the MLP models including the manual optimisation of the number of layers and the number of neurons for 30 datasets took around 54.63 seconds. That fitting the SARIMA models takes the longest time is expectable as here the most hyperparameters need to be optimised. This is done via fitting several models and comparing their quality (see Section 3.3.3) which takes up a considerable amount of time. For a monthly forecast, however, even when fitting models to more than 4000 parts slightly longer computation times for the SARIMA models should not pose a major obstacle. There are also several ways of how to speed up the model fitting process using `auto.arima`, such as allowing a stepwise model selection process via the argument `stepwise` or running the search in parallel via setting the argument `parallel` to `TRUE` (Hyndman et al., 2026, p. 16 f.).

Figure 6.1 shows the distribution of the calculated WMAPEs of the in advance forecasted monthly values per model and per XYZ-classification of the items. To display the boxplots in this figure, as well as in all other figures containing boxplots, the function `geom_boxplot` is used, where the upper hinge of the box represents the 75% quartile and the lower hinge shows the 25% quartile of the distribution (Wickham et al., 2026a, p. 85). Single points outside of the whiskers of the boxplot represent outliers, which are either larger than the 75% quartile plus 1.5 times the inter-quartile range or smaller than the 25% quartile minus 1.5 times the inter-quartile range (Wickham et al., 2026a, p. 85). Figure 6.1a shows the full plots, while the coordination system is cut at 300% in Figure 6.1b in order to make differences between models in the X-class and the Y-class better visible. In the Appendix in Table 8.1 the WMAPEs for all parts and all models individually listed can be found.



(a) Boxplots of WMAPEs per model and per XYZ-group



(b) Boxplots of WMAPEs per model and per XYZ-group cut at 300%

Figure 6.1: Boxplots of WMAPEs per model and per XYZ-group of the monthly forecast

Looking at Figure 6.1, it is visible that while the WMAPEs for the items in the Z-group tend to be the highest, the forecast quality for the items in the X-class and the Y-class is on a similar level. For the X-class the medium WMAPE for the SARIMA approach is the lowest being around 50%, whereas the highest medium WMAPE can be seen for the Holt-Winters approach being around 60%. The naïve method, the Holt-Winters approach and the MLP show higher variations in the forecast quality than the SARIMA and the NNAR models. For the naïve baseline method and the MLP WMAPEs as low as 28% and 19% respectively could be achieved, while there are also WMAPEs of up to 73% and 84% respectively. For the items in the Y-class the traditional mehtods Holt-Winters and SARIMA outperform the Neural Network approaches, while the quality of the baseline naïve approach is in between. Both in the X-class and the Y-class the SARIMA shows the best prediction quality based on the WMAPE. For the Z-group the forecast quality varies largely between different parts, with some errors of the Neural Network approaches being higher than 600%. The lowest median, however, occurs for the MLP approach, while the baseline model has the second lowest median in this class of items. Generally, however, with the median of the WMAPEs in this class being higher than 75% for every model, it is questionable if a forecast on a monthly level even makes sense for this group of items. It should also be noted that while for numerous parts the baseline approach performs slightly worse than some of the dedicated forecasting algorithms, in many cases it also outperforms especially the Neural Network methods. The results, however, also show that creating an extremely accurate monthly forecast is not possible for most of the parts, which highlights that despite some regular components in the time series, real industrial data still incorporates a variety of irregular and unpredictable factors.

Figure 6.2, similarly to Figure 6.1, shows boxplots per model and per XYZ-group, but it displays the SMAPE on the y-axis instead of the WMAPE. The list containing the values for all items and all models can be found in Table 8.2 in the Appendix.

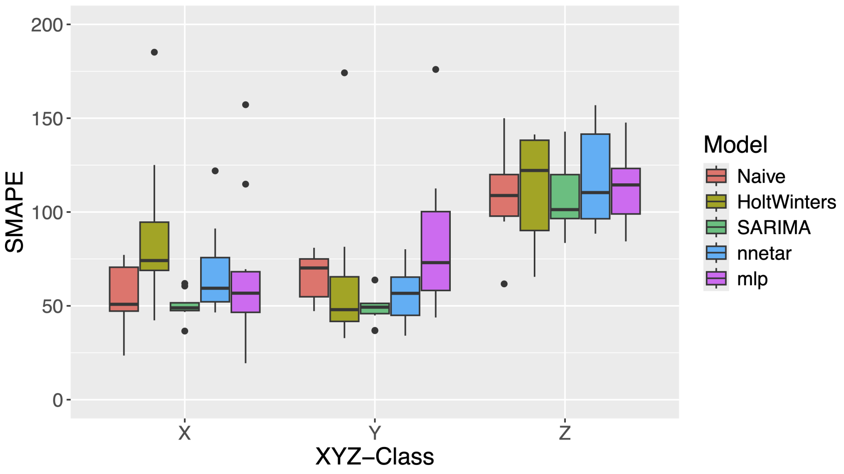
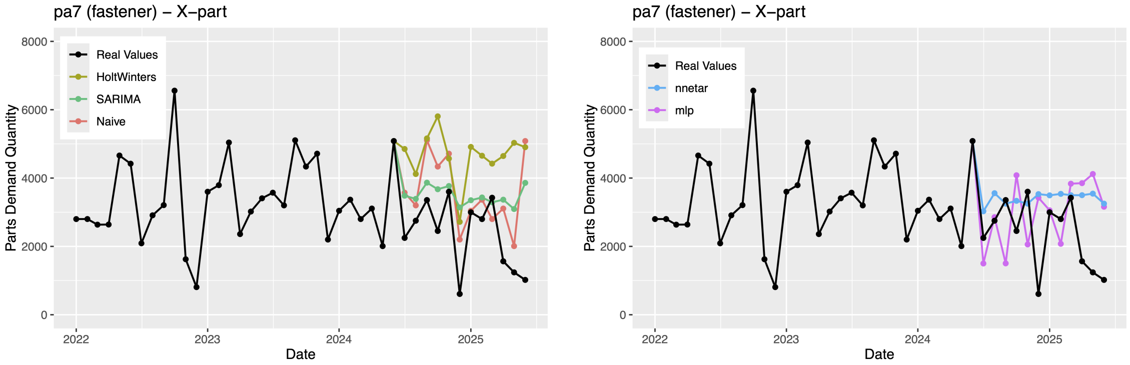


Figure 6.2: Boxplots of SMAPEs per model and per XYZ-group of the monthly forecast

Comparing Figure 6.2 and Figure 6.1 it can be seen that the results based on the WMAPE and based on the SMAPE are similar. Both based on the WMAPE and based on the SMAPE for the X-class of items the SARIMA model performs best, while the Holt-Winters approach delivers the worst results. For the Y-class both of the traditional approaches outperform the Neural Network approaches. Despite this, one difference between the analysis based on the WMAPE and based on the SMAPE is that based on the SMAPE the NNAR models tend to outperform the naïve baseline approach for the Y-parts, which is the other way around considering the WMAPE. The biggest differences between the results based on the two metrics occur for the Z-class. One difference is that according to the WMAPE errors can be far larger than 200%, while the SMAPE is by definition restricted to 200% maximum (see Equation 5.3). The second difference is that according to the median WMAPE the MLP performs best with the baseline approach being in second, whereas according to the median SMAPE the SARIMA approach is the best-performing approach followed by the NNAR method. Generally, however, it can seen that the results do not vary considerably between the two error metrics.

Figure 6.3 and 6.4 show the predicted monthly values for two parts of class X, where the predicted values for the classical methods as well as for the baseline naïve approach are displayed in the Figures 6.3a and 6.4a and the forecasted values for the Neural Network approaches are shown in 6.3b and 6.4b. For both parts only the values from 2022 onwards are displayed in order to make the predicted values better visible.

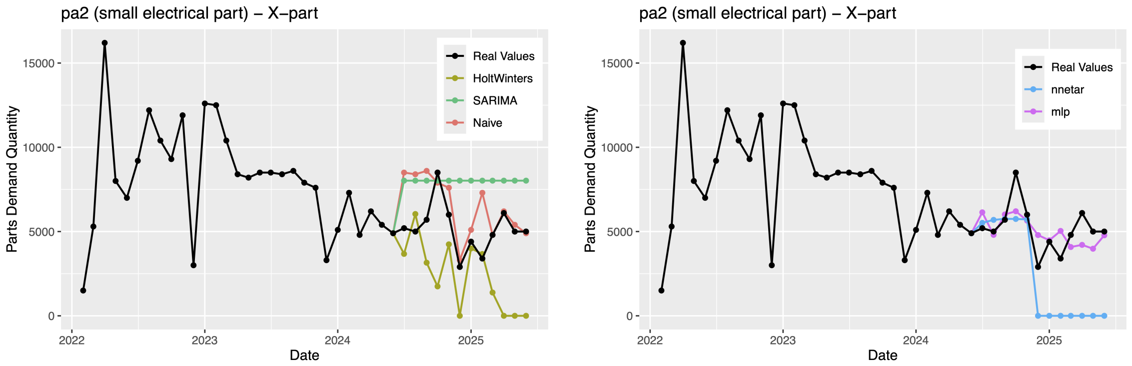


(a) Forecasts based on the classical approaches Holt-Winters and SARIMA

(b) Forecasts based on the Neural Network approaches

Figure 6.3: Forecast results for part pa7 (fastener) - X-part

The results of the item `pa7`, which belongs to the group of fasteners, shown in Figure 6.3 tend to be in line with the aggregated results shown in the Figures 6.1 and 6.2. The SARIMA model clearly outperforms the Holt-Winters model and performs slightly better than the Neural Network approaches too. The predictions displayed in Figure 6.3 also show that SARIMA outperforms the more complex approaches via predicting approximately the same level, while especially the predictions of the MLP approach vary intensively. Therfore the MLP often predicts demand spikes exactly in the opposite months to when the demand spikes actuall occured. It appears like the Neural Network overfits the data. Overfitting means that the model bases its predictions too much on the random and noisy components of the training time series, which are however irregular and non-recurring patterns and therefore lead to inaccurate predictions (Vandeput, 2021, p. 66). Both the Holt-Winters and the NNAR approach estimate the level of the time series too high. For the Holt-Winters method it is especially obvious that the slight downwards trend in the series could not be recognised by the model through looking at the `beta`-parameter being zero (see Table 8.5).



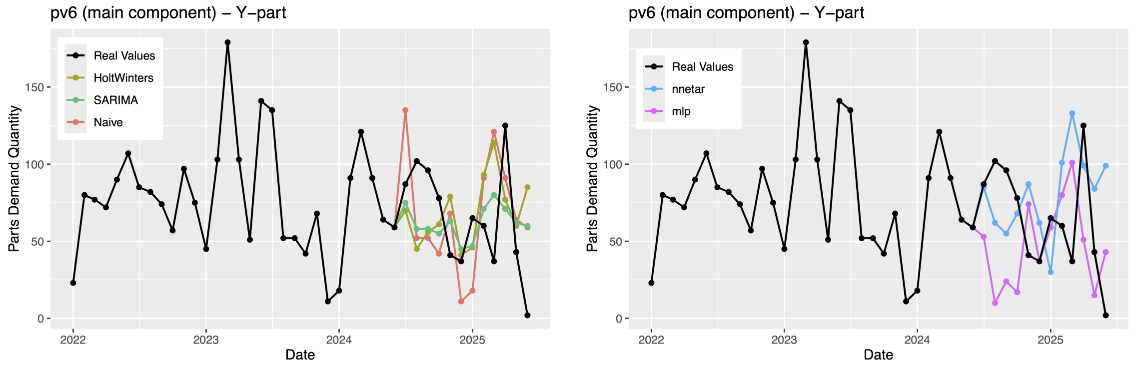
(a) Forecasts based on the classical approaches Holt-Winters and SARIMA

(b) Forecasts based on the Neural Network approaches

Figure 6.4: Forecast results for part pa2 (small electrical part) - X-part

While the results for `pa7` follow the pattern visible in the aggregated results in the Figures 6.1 and 6.2, the results of `pa2`, which is a small electrical part, are completely different. The only model capturing the pattern of this time series well and producing a reasonable forecast is the MLP model delivering a WMAPE of 18.65% and a SMAPE of 19.43% (see Tables 8.1 and 8.2 in the Appendix). The SARIMA approach does not recognise any patterns in the time series at all and therefore only predicts a straight line, whereas both the Holt-Winters and the NNAR (nnetar) model overestimate the downwards trend of the time series. The NNAR model even predicts negative values, which were then changed to predicted zeros. This time series is an example for a dataset on which the more complex Neural Network approach was able to recognise the patterns in the series, which have not been recognised by the classical approaches and the simpler NNAR method.

Figure 6.5 shows the prediction results for the `pv6` part, which is a main component and belongs to the Y-class. The results for this item also emphasise how the SARIMA model producing a smoother forecast outperforms other models, which predict demand spikes exactly in the wrong months. Especially for Y-parts, which tend to be more irregular than X-parts, SARIMA also significantly outperforms the naïve method as patterns and demand spikes from the previous year do not seem to repeat in the exact same way in the following year.

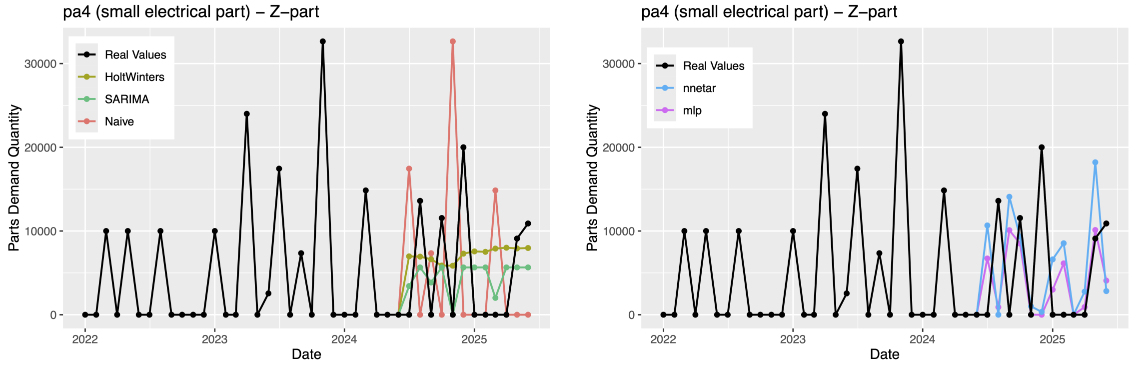


(a) Forecasts based on the classical approaches Holt-Winters and SARIMA

(b) Forecasts based on the Neural Network approaches

Figure 6.5: Forecast results for part pv6 (main component) - Y-part

Figure 6.6 shows the predictions for `pa4`, which is a small electrical part and belongs to the Z-class. The demand spikes occur very irregularly and there are also numerous months without a demand. As there are hardly any structural patterns occurring in this time series, no model can produce reasonable monthly forecasting results, with WMAPEs and SMAPEs being larger than 100%. The question, however, if a monthly forecast makes sense, does not only arise for this specific part, but for the whole group of Z-items, as according to the Figures 6.1a and 6.2 for the majority of parts no better results than WMAPEs and SMAPES of over 75% can be achieved.



(a) Forecasts based on the classical approaches Holt-Winters and SARIMA

(b) Forecasts based on the Neural Network approaches

Figure 6.6: Forecast results for part pa4 (small electrical part) - Z-part

Additionally to the monthly in advance forecasts, rolling forecasts are produced where always the demand of the following month is predicted. The overview of the results based on the rolling forecast evaluated using the WMAPE is displayed in Figure 6.7. The detailed results can be found in Table 8.3 in the Appendix.

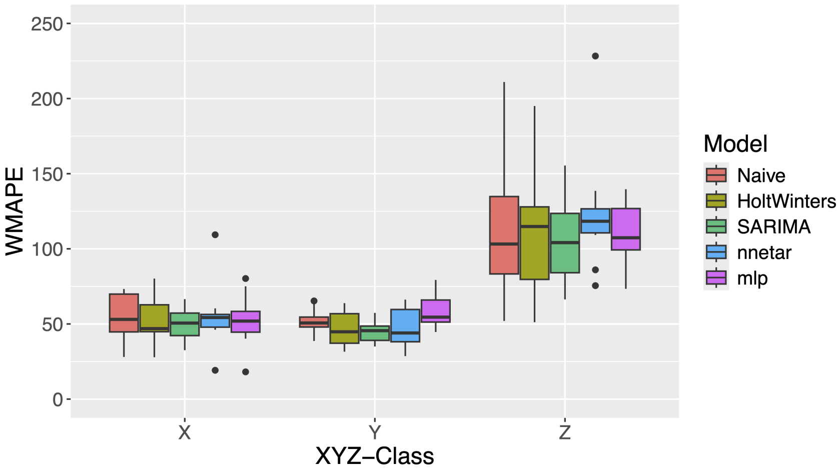
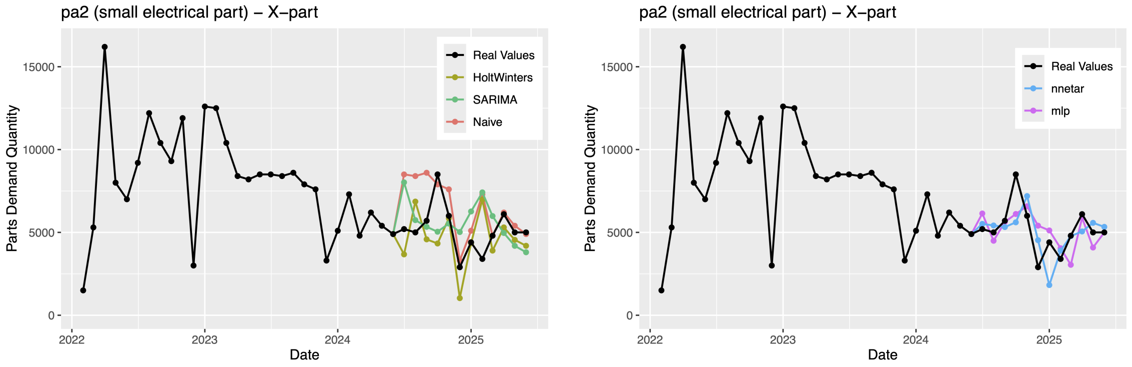


Figure 6.7: Boxplots of WMAPESs per model and per XYZ-group of the rolling monthly forecast

As for the rolling forecast, always the demand for the following month is predicted and all data points until that month are used for training the model, it is expected that the forecast quality improves compared to the in advance prediction. For the naïve method, the results for the in advance and the rolling predictions are the same, because the demand of the previous year stays the same regardless of how many values are considered for training. Comparing the Figures 6.1 and 6.7, in line with the expectations, a slight improvement of the prediction quality, especially for the X-class and the Y-class, can be observed. For the X-class the most improvement occurred for the Holt-Winters method and the MLP. For the Y-class mainly the two Neural Network approaches improved, while the quality of the SARIMA models got slightly worse compared to the in advance forecast. Despite still showing considerably high errors, the forecast quality in the Z-class significantly improved, especially for the SARIMA and the NNAR approach. For the MLP the median WMAPE slightly increased from under 100% to a bit above 100%, but on the other hand the 75% quartile decreased from above 150% to about 130%. Generally, however, for the Z-class also for the rolling forecasts the median WMAPEs are around 100% and therefore also the business value of a monthly rolling forecast for this group of items is questionable. Similar results as for the analysis based on the WMAPE can be seen when evaluating based on the SMAPE. This graphic can be found in Figure 8.1 in the Appendix, as it is not discussed in detail.

Figure 6.8 shows the rolling monthly forecasts for `pa2` and compared to the in advance predictions illustrated in Figure 6.4 there are various differences. While for the in advance forecast the Holt-Winters approach massively overestimated the downwards trend of the time series and the SARIMA method only predicted the same value for all 12 months, both models show reasonable rolling forecasts with WMAPEs of 27.86% and 32.55% respectively (see Table 8.3). While for the forecasts in advance the NNAR approach did not produce any reasonable predictions, its performance for the rolling forecast almost matches the quality of the MLP model with a WMAPE of 19.17% compared to 18.12% (see Table 8.3). The results of this item show that the distinction between an in advance forecast and the rolling prediction strategy can have a massive influence on the prediction quality.



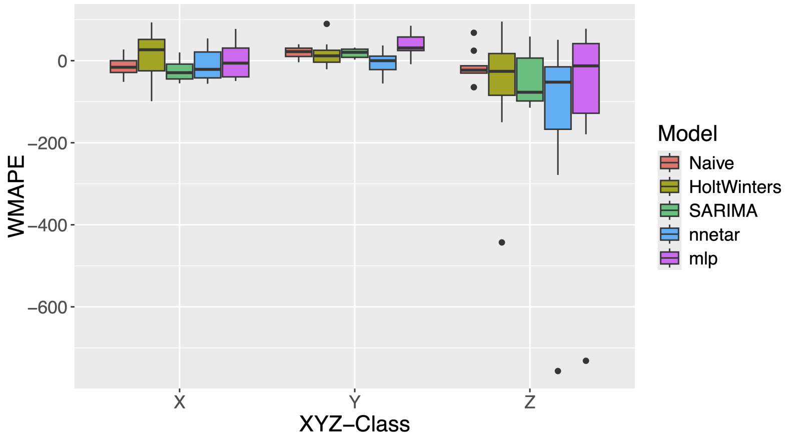
(a) Rolling forecasts based on the classical approaches Holt-Winters and SARIMA

(b) Rolling forecasts based on the Neural Network approaches

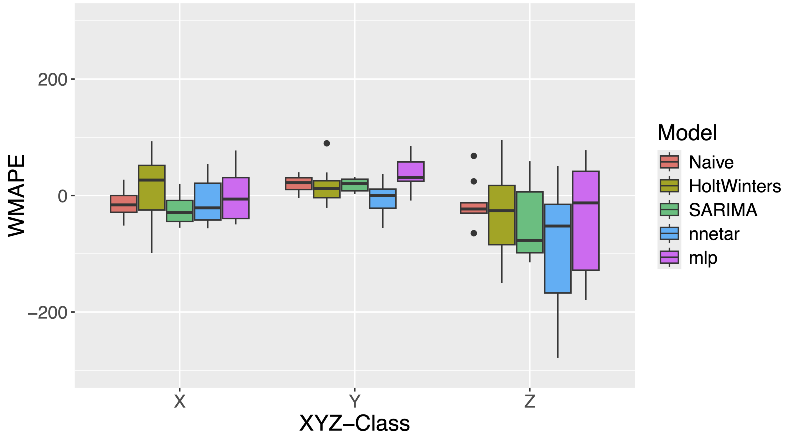
Figure 6.8: Rolling forecast results for part pa2 (small electrical part) - X-part

Additionally to the monthly forecasts, yearly forecasts were generated based on summing up the monthly in advance forecasts. As in some months the demand might be underestimated and in some months it might be overestimated, which balance each other out, it is expected that the quality of the yearly forecasts is better than the quality of the monthly forecasts. The detailed WMAPEs for all parts can be found in Table 8.4 in the Appendix.

Figure 6.9 shows the distribution of the Percentage Errors (PEs) per model and per XYZ-class. The upper graphic (Figure 6.9a) displays the whole distribution including the outliers with massively negative PEs, while Figure 6.9b is cut at -300% and at 300% to make differences between the models and the groups better visible. If only two values, the actual and the predicted demand, are compared, the PE corresponds exactly to the WMAPE in magnitude. The only difference between the two error metrics when applied on only one pair of values is that the PE can be negative or positive, while the WMAPE is calculated based on the absolute error. Due to the fact that if only two values are compared the PE and the WMAPE are calculated in the same way, the monthly results and the yearly forecasting quality can be directly compared when ignoring the sign of the PE.



(a) Boxplots of PEs per model and per XYZ-group



(b) Boxplots of PEs per model and per XYZ-group cut at 300%

Figure 6.9: Boxplots of PEs per model and per XYZ-group of the yearly forecast

As defined in Section 5.3, the PE is negative if the actual yearly demand was less than the predicted amount and the PE is positive if the actual yearly demand was more than the forecasted amount. Looking at the error distributions in Figure 6.9 it can be seen that most models, especially in the X-class and in the Z-class, overestimate the yearly parts demand. Part of this effect might, however, also occur due to the fact that the forecasted demand quantity is always rounded up if the predictions is no integer number, which might accumulate due to summing up. Especially the demand for Z-items is overestimated, which might also be caused by the behaviour of time series models hardly predicting zeros, even though the actual time series contains numerous zeros. Such a behaviour can also be seen for `ps4` in Figure 6.6. Looking, for example, at the mathematical formulas of the classical approaches, as illustrated in Section 3.2 and 3.3, it also makes sense that zero is very rarely predicted, because the forecasts are basically a weighted combination of the past values. Even though a time series has very sparse demand, there have usually also been a few positive demand spikes in the preceding months, which prevent the forecasts from being completely zero. From a business perspective, however, a slight overestimation of the yearly demand might be more desirable than an underestimation. This is also the reason, why the monthly demands are rounded up, if no integer number was predicted by the models.

Comparing the absolute PEs with the monthly WMAPEs in Figure 6.1, it can be seen that the forecasting quality for the yearly forecasts is, as expected, significantly better than for the monthly predictions. The naïve approach performs particularly well across all three classes with rather little quality variations within the groups. While for the X-class the MLP approach works well with a medium PE of only -6.04%, for the Y-class the NNAR approach shows a better quality with a medium PE of only -0.03%. This means that when comparing the Neural Network approaches with each other, the more complex MLP seems to predict the yearly demand more accurately for the group of smoother parts of the X-class, while the network with only one hidden layer performs better on the slightly less smooth group of Y-parts. A similar behaviour could also be seen in Figure 6.2 when evaluating the monthly forecast based on the SMAPE. The reason for this might be that the more complex MLP overfits for the Y-parts and therefore bases its predictions too much on the irregular noise, which can be better filtered out by the simpler NNAR networks. In general, it can be observed that the ranking which model performs best on which groups is similar to the monthly results. This is the expected behaviour as the yearly forecast is based on the monthly predictions.

Table 6.2 shows the median of the WMAPEs per model and per XYZ-class. The abbreviation `m.` stands for the monthly in advance forecast, while `r.` stands for the rolling forecast and `y.` is the abbreviation for the yearly forecast. For a better visual orientation, the columns are coloured like in Figure 4.1, where green represents the X-class, yellow is the colour for the Y-parts and red is associated with the Z-parts. While Table 6.2 only shows the median of the WMAPEs and therefore does not give an insight into the complete distribution of the errors, it gives an overview of the results. Like it can be seen in Figure 6.1, the Holt-Winters approach does not work well for the X-parts, while it produces much more competitive results for the Y-items. SARIMA works especially well for the X-items and the Y-items, where it outperforms the more complex Neural Network approaches. The naïve approach works especially well for the yearly forecasts. It can also be seen that the yearly medium errors are significantly lower than the medium errors for the monthly forecasts and that the medium WMAPEs for the rolling forecasts tend to be smaller than for the in advance forecasts, at least for the X-group and the Y-group. For the Z-group, the rolling forecasts are partly even worse than the in advance predictions. This also emphasises that hardly any pattern can be identified within these time series, even when updating the forecasts monthly. Generally, the medium monthly errors of above 100% for almost every model show, that monthly predictions are not reasonable for the Z-group of items. Regarding the yearly forecasts, however, especially the naïve method as well as the Holt-Winters and MLP approach produce results, which are even competitive with the other groups of items. The fact that the naïve approach, where simply the amount of the last year is taken as a forecast, performs that well on this group of parts shows that while monthly peaks can hardly be predicted, the yearly demand of these items often stays in a similar range than in the preceding year.

Model	X-m.	X-r.	X-y.	Y-m.	Y-r.	Y-y.	Z-m.	Z-r.	Z-y.
Naïve	53	53	-16.1	50.7	50.7	22	103	103	-23.1
Holt-Winters	62.2	46.9	26.6	42.8	44.8	11.8	109	115	-26.1
SARIMA	50.5	50.6	-29.3	43.4	45.5	20.4	120	104	-77
nnetar	57.4	54.3	-21.2	54.3	44	-0.03	136	118	-52.3
mlp	59	51.9	-6.04	59.7	54.5	31.1	87.2	107	-12.7

Table 6.2: Median WMAPEs per model and XYZ-group for the monthly, rolling and yearly forecast

To summarise, none of the approaches applied shows a uniformly superior performance on all groups of parts and for all forecasting modi. The naïve baseline approach also delivers results, which are not far worse and sometimes even better than the more advanced methods. Looking at the Figures 6.1 and 5.1, however, it can be seen that forecasting using dedicated prediction approaches does deliver some added value compared to the baseline approach. For the monthly in advance forecast of the X-parts and the Y-parts especially the SARIMA approach tends to lead to better results than the baseline method and it also tends to outperform the more complex Neural Network methods. The monthly forecasting results can be improved when using a rolling forecasting scheme, where always the following month is predicted using all data available until that respective month. For that forecasting scheme, the differences in performance between the different methods is smaller than for the in advance prediction, while Holt-Winters achieves the best median WMAPE for the X-class of parts and the NNAR approach for the Y-items, even though their results vary more than those of the SARIMA approach. No reasonable monthly forecasts can be generated for the Z-class of parts, neither in advance nor using the rolling approach, as the WMAPEs for numerous parts are above 100%. As expected, the quality for the yearly forecast is significantly better than for the monthly predictions. While for the X-parts the MLP method and for the Y-class the NNAR method achieve the lowest median WMAPE, the baseline approach also shows considerably promising results for the yearly prediction, even for the Z-class of parts. The results, however, also show that the data at hand does not allow an extremely accurate prediction of the monthly demand due to its varying and partly highly irregular nature.

6.2 Possible improvements

In the following section, two possible ways of improving the results of the forecasting methods are presented. In Section 6.2.1, a closer look is taken at the prediction results per part category and if the results can be improved if the demand of each category is predicted as a whole and in Section 6.2.2 the possibility of the inclusion of exogenous variables into the models is evaluated.

6.2.1 Prediction per category

Figure 6.10 shows the distribution of the WMAPEs per article category. The graphic shows that the quality of the forecasts varies largely between the categories. While the prediction quality tends to be better for the category of main components, for the valves and for the fasteners, it tends to be worse and more varying for the gaskets and seals as well as for the small electrical parts. This difference in quality is expected when looking at Table 4.6 which shows the parts, the articles categories and the variation coefficients, as class Z with the highest variation coefficients mainly consists of gaskets and seals and of small electrical parts. As the results presented in Section 6.1 show that the prediction of Z-items works significantly worse, it is also presumable that the prediction quality for these article categories tends to be worse.

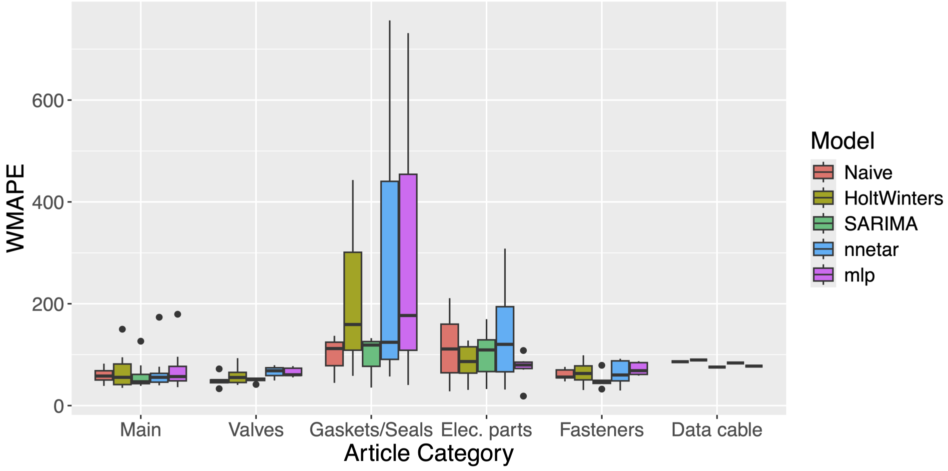


Figure 6.10: Boxplots of WMAPEs per model and per article category of the monthly forecast

As especially the prediction of auxiliary parts is a challenge, the question arises, if instead of predicting the demand of individual parts, the prediction of the sum of the demand of the whole article categories could significantly improve the performance, because it might smooth the demand. This could be especially helpful and reasonable for parts such as gaskets and seals or small electrical parts, where resource planning is carried less order driven because these parts are required for a variety of different products. Therefore, it could make sense to not only plan in advance for specific parts and part configurations, but for a whole part category. For the sake of completeness, however, the analysis is carried out for all article categories. As article category 1, which is the category for the main components, covers a number of significantly diverse parts including heat exchangers, meters, actuators and regulators (see Table 4.3), this article category is split up into exactly these four subcategories for the analysis. Furthermore, for category 5 the part `pa8` is excluded from the analysis as it is the only part in category 5 which is actually a screw and does not belong to the subgroup of screw accessories and item `pa5` is not considered as it is the only part which belongs to the category of network data cables (see Table 4.2).

Table 6.3 gives an overview of the groups and which parts they contain.

Group	**Article category**	**Part ID**
1	Category 1 - heat exchangers	pv1, pv14
2	Category 1 - meters	pv2, pv3, pv5, pv15
3	Category 1 - actuators	pv6, pv8
4	Category 1 - regulators	pv4, pv11
5	Category 2 - valves	pv7, pv9, pv10, pv12, pv13
6	Category 3 - gaskets / seals	pa1, pa3, pa11
7	Category 4 - small electrical parts	pa2, pa4, pa6, pa13, pa14, pa15
8	Category 5 - screw accessories	pa7, pa9, pa10, pa12

Table 6.3: Overview of the grouping for the analysis per article category and subcategories

After preprocessing the datasets through summing up the monthly demand values of the individual parts to one time series per category, all four models and the baseline model are fitted to the datasets. After combining all the individual time series, each of the analysed datasets has 78 data points available for training ranging from January 2018 until June 2024, while, similarly to the per part analysis, the test dataset contains 12 values from July 2024 until August 2025.

As the focus is mainly on improving the monthly demand forecast, because the yearly prediction already delivers reasonable results when carried out on single parts, Table 6.4 shows the WMAPEs for the monthly in advance and the monthly rolling forecasts. A complete table including the quality for the yearly forecasts can be found in Table 8.6 in the Appendix. For category 4 (small electrical parts) and for category 5 (screw accessories) no rolling forecast could be generated based on the Holt-Winters model due to an optimization failure, as the `HoltWinters` function of the `R`-package `stats` determines the parameter values of α, β and γ based on numerical optimization (R Core Team, 2026a), which failed in these particular cases.

Comparing the prediction qualities for the different article categories it can be seen that the most accurate forecasts can be achieved for category 1, category 2 and category 5, while the prediction quality for category 3 and category 4 is considerably worse across all models. This is expected as Figure 6.10 also shows that forecasting single parts in category 3 and category 4 tends to work remarkably worse than for the other item categories.

	Cat1-ex.		Cat1-met.		Cat1-act.		Cat1-reg..	
Model	m.	r.	m.	r.	m.	r.	m.	r.
Naïve	50	50	47.8	47.8	37.1	37.1	47.7	47.7
Holt-Winters	40.3	41.5			30.8	27.4	34.1	39
SARIMA	45.7	48.5	45.5	33.1	40.8	30	41.1	44.1
nnetar	51.9	68	32	33.5	29.7	31.4	34.4	37.4
mlp	98.3	77.2	37.7	38.9	59.4	38.3	43	55

	Cat2		Cat3		Cat4		Cat5	
Model	m.	r.	m.	r.	m.	r.	m.	r.
Naïve	19.6	19.6	63.8	63.8	94.5	94.5	33.3	33.3
Holt-Winters	32.8	29.1	65.7	60.2	57.1		35.3	
SARIMA	37.3	27.4	83.4	57.8	49.6	58.2	34	31.5
nnetar	29.8	26.8	58.7	58.4	69.2	42.1	26.1	23.3
mlp	21.7	27.2	53.1	66.3	53.1	45.6	40.3	32.6

Table 6.4: WMAPEs per model for the monthly and rolling forecast per parts group

Comparing not the categories with each other, but the quality of the models it can be seen in Figure 6.11 and Figure 6.12 that based on the median WMAPE the Neural Network Autoregression approach works best. Despite this, for the in advance forecast the quality of the NNAR models varies more and its 75% quartile of the WMAPE distribution is higher than those of the SARIMA and the Holt-Winters approach. For the rolling forecasts, especially Holt-Winters shows a good performance with only one WMAPE being significantly larger than 40%. Holt-Winters, however, also has the disadvantage that for two time series (for category 4 and category 5) the parameter optimization did not converge and no results were generated. The more complex MLP approach shows the worst performance and in some cases even produces worse results than the naïve baseline approach.

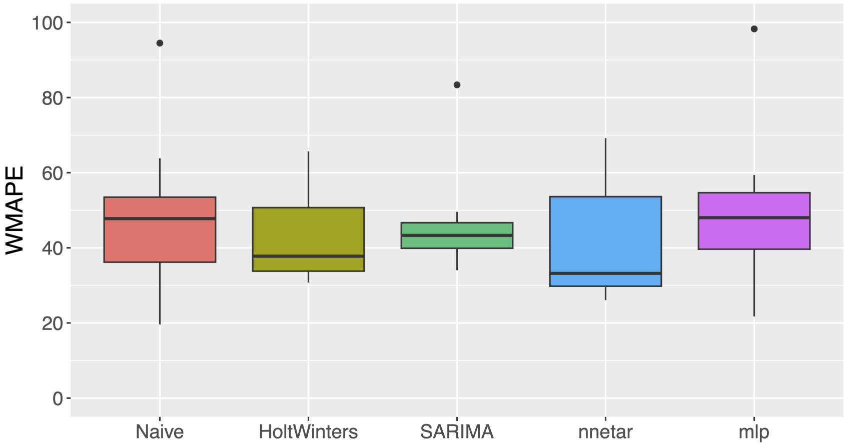


Figure 6.11: Boxplots of WMAPESs per model for the grouped monthly forecast

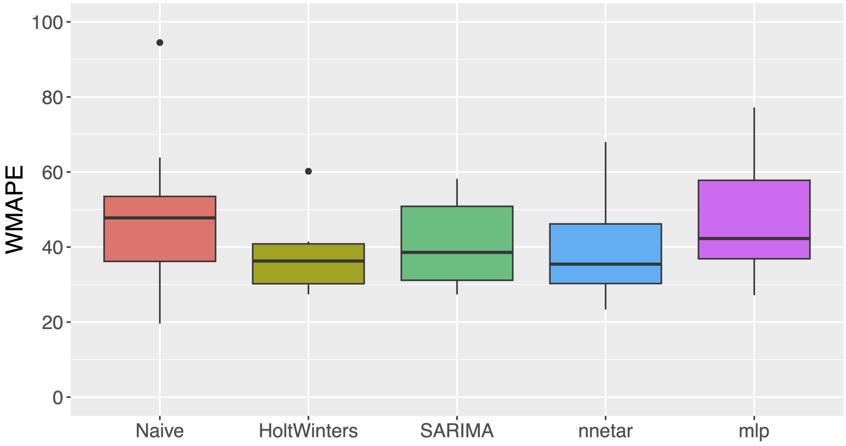


Figure 6.12: Boxplots of WMAPESs per model for the grouped rolling forecast

As the goal of the grouped analysis is not only to compare the forecast qualities between the categories and between the models, but also to compare it to the predictions per parts, 6.13 compares both results. It shows the WMAPESs per part split up by model and item category, similarly to Figure 6.10, and it additionally illustrates the WMAPEs of the grouped forecasts via the grey x-shaped markers. In order to make differences better visible, the graphic is cut at a WMAPE of 200%. The full graphic can be found in Figure 8.2 in the Appendix. For the category of main components, which was split up into four subcategories, the grey markers show the mean WMAPE of all four subcategories. In the figure it can be seen that for most categories and for most models the WMAPEs of the grouped prediction are remarkably lower than the median WMAPEs of the per part forecasts. In several cases, they are even lower than the 25% quartile of the WMAPEs of the per item analysis. Quality improvements could also be achieved for the category of gaskets and seals (category 4) and for the category of small electrical parts (category 5). Nevertheless, their WMAPEs are still considerably higher than those of the other categories.

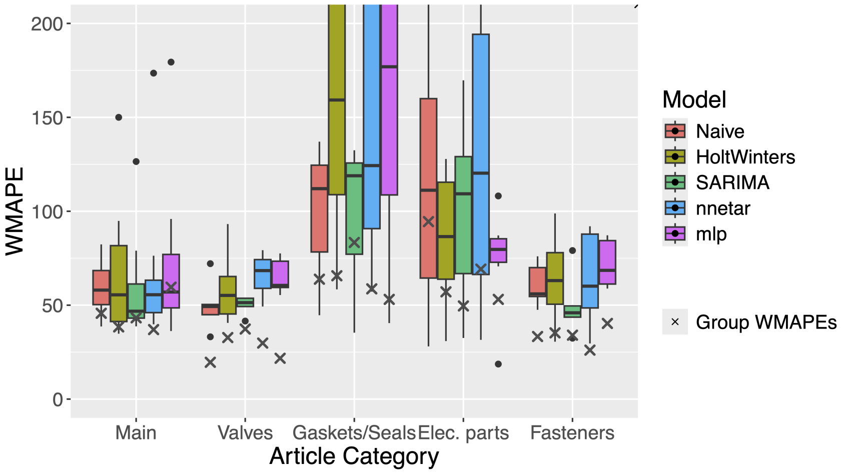


Figure 6.13: Boxplots of WMAPEs per model and per article category of the monthly forecast including the WMAPEs of the grouped monthly forecast

Even though, as expected, the predictions per group lead to an improvement in the quality of the forecasts, this approach has numerous significant limitations. First of all, a prediction for the whole group of items does not always add business value. While it can help to plan and optimise resources and budget it is not of great value for the optimisation of the order process itself, as in many cases items within one category are not interchangeable. There are, however, exceptions, especially considering auxiliary parts such as gaskets or certain screw accessories where slightly different items from different suppliers might be used interchangeably. Another considerable drawback regarding this grouped analysis is that in this study only 30 parts are taken into account, while in reality all those categories consist of a vast amount of more parts. These parts also need to be manually classified into the categories, which would require a considerable amount of manual effort.

6.2.2 Inclusion of exogenous variables

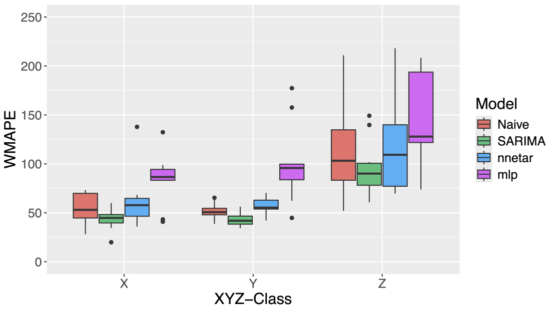
While the future values of time series are heavily influenced by the past values of the dataset, also additional variables, so-called exogenous variables, might have an effect. The integration of such exogenous variables into the forecasting models can significantly improve the forecasting performance (Giannopoulos et al., 2025, p. 34).

Exogenous variables can be easily integrated in `R` into three out of the four analysed methods. While for the Holt-Winters approach it is not possible to integrate exogenous variables, for SARIMA using the `auto.arima` function it can be done using the argument `xreg`, which can be either a numerical vector or a matrix (Hyndman et al., 2026, p. 16). The same can be done for the function `nnetar` fitting the Neural Network Autoregression model (Hyndman et al., 2026, p. 112) and for the function `mlp` fitting the Multilayer Perceptron (Kourentzes, 2023, p. 13).

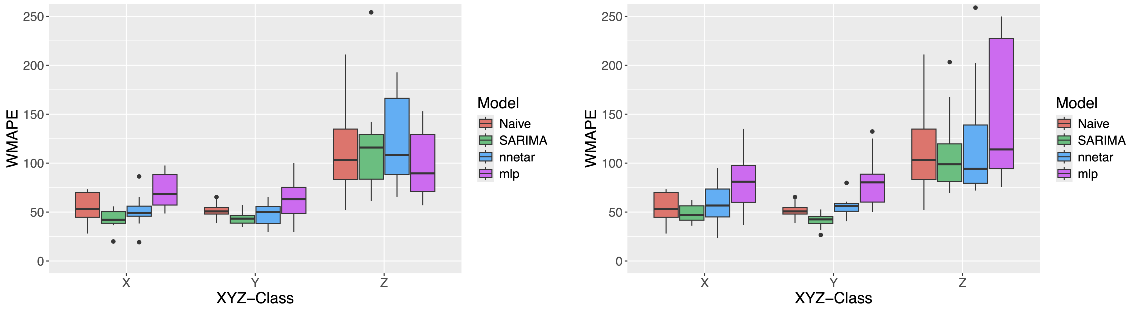
The biggest drawback of the inclusion of external variables is that the values for these variables do not only have to be available for the whole history of the time series, but also future values of this variable need to exist in order to generate forecasts. This, however, presents a significant challenge as, for example, most macroeconomic factors do not exist in advance for the future or at least not in the desired granularity and for the whole forecasting time span. Additionally to being available, the data should also be constantly updated, accessible and ideally automatically downloadable by the forecasting tool without any additional manual work.

Nevertheless, in the course of the research for variables which could possibly have an influence on the parts demand and which are also available for the future, two variables were identified. The variables, the expected price development in the upcoming 12 months and if bigger investments are expected to be made in the upcoming 12 months, are published by the OENB (Österreichische Nationalbank) and are based on a monthly telephone survey carried out by the market research institute Ipsos among 1500 men and women in Austria Oesterreichische Nationalbank (2026). Due to the fact that the variables actually represent expectations for the future, their effect can be seen as lagged and they can also be used to generate future predictions. It, however, has to be noted that both variables also have severe disadvantages. First of all, they are only based on a survey and on an expectation Therefore, they might not always reflect the actual economic development of the upcoming months. Secondly, they are only estimates for Austria. While Aqotec's main sales market, with a sales share between 30% and 40%, is Austria, also the economic situation in other sales markets might have an influence on the parts demand.

Research was also carried out towards a possible integration of the amount of subventions from the Austrian state for a heating change towards a more environmentally friendly heating system. Informations about federal subventions can be found on the official portal of the Austrian Federal Ministry for Climate Protection, Environment, Energy, Mobility, Innovation, and Technology, for example for the year 2025 (Bundesministerium fuer Klimaschutz, Umwelt, Energie, Mobilitaet, Innovation und Technologie, 2025) and for the year 2024 (Bundesministerium fuer Klimaschutz, Umwelt, Energie, Mobilitaet, Innovation und Technologie, 2024). The provided information, however, only includes the federal subventions and no state-specific subventions. The biggest disadvantage is that the maximum amount of subvention is always constant for at least several months or even for one or more years. This means, the information is not usable to improve the prediction accuracy of the monthly demand spikes. For this reason, only the two monthly variables provided by the OENB are included in the following analysis.



(a) Models including the exogenous variable for the expected price development and the expected bigger investments



(b) Models including the exogenous variable for the expected price development

(c) Models including the exogenous variable for the expected bigger investments

Figure 6.14: Boxplots of monthly WMAPEs per model and per XYZ-group for the model including the exogenous variable for the expected price development and for the expected bigger investments

Figure 6.14 shows the results for the models with both exogenous variables integrated (Figure 6.14a) as well as the results for the models with only one of the two variables included (Figure 6.14b and Figure 6.14c). Naturally, the naïve approach does not allow the inclusion of exogenous variables into the predictions. However, in order to compare the qualities of the other models with the baseline the WMAPEs for the naïve method are also included in the figures. All three figures are cut at the WMAPE value of 250% excluding single outlier points for the Z-class, in order to be able to also identify subtle quality differences between the models. The figures including the outlier points can be found in the Appendix in Figure 8.3. Additionally, the full results can be found in the Tables 8.7, 8.8 and 8.9 in the Appendix.

Comparing the three sub-figures in 6.14, it can be seen that the inclusion of both exogenous variables leads to the worse results. The best results were achieved including only the expected price development as an exogenous variable. For the X-class and for the Y-class the SARIMA approach delivers the best results, while the MLP approach leads to the worst results. Compared with the results without the integration of exogenous variable (see Figure 6.1), it can be noticed that the results for the SARIMA model and for the NNAR approach are approximately of equal quality or could be improved. The quality of the MLP model, however, significantly worsened. The reason for this might be that the more complex MLP approach even more overfits when the exogenous variable is included.

While the inclusion of the exogenous variable regarding the expected price development leads to a minimal improvement of the SARIMA and NNAR models, the improvement is not significantly large. This is also expected, as while the general economic development might influence the demand for the parts, there would naturally be variables which might have a larger predictive power, such as the company internal variables regarding current marketing campaigns or discounts. These variables are, however, not available in a format which could be used for the prediction tool.

Regarding the decision, if a macroeconomic indicator such as the expected price development provided by the OENB should be included into the forecasting tool, one also needs to consider that while a small improvement of the forecasting quality might be achieved the inclusion of external variables also adds sources of dependence. If the required data is not published by the OENB any longer or it is published in a different form or granularity, the models need to be adapted. Furthermore, in a productive environment the querying and processing of output from an external source might lead to errors if there are only the slightest changes of the format in which the data is provided. To summarise, while the inclusion of external variables might lead to an improvement in the forecasting accuracy, it also leads to an increased maintenance effort, especially as soon as the tool runs productively.

7. Conclusion & possible future research

The following chapter presents a final summary and conclusion of the findings, highlights the limitations of the study and discusses possible further research topics regarding the implementation of a parts demand forecasting tool.

7.1 Conclusion

In this study two classical statistical approaches, Holt-Winters and SARIMA, and two Neural Network approaches, NNAR and MLP, were applied to forecast the demand of 30 parts regularly used by the company Aqotec. Which 30 parts out of the more than 4000 parts were included in the analysis was decided based on the purchase amount and the purchase value in 2024. Only parts, which have a history from at least July 2022 are considered in this study, as otherwise no reasonable model training can be carried out on the data. The training dataset includes data up to June 2024, and the data from July 2024 until June 2025 is used as a test dataset for evaluation purposes.

As one of the goals of this study was to find out for which kinds of parts which models work best, the data was first split into three groups through a XYZ-classification approach based on the variation coefficient. Each of the three groups should have an equal size. Therefore, all parts with a variation coefficient below 0.63 are considered to belong to the X-group, parts with a variation coefficient between 0.63 and 0.8 were assigned to the Y-group and all other parts are of group Z.

The models were evaluated using three different schemas including a 12 months in advance forecast, a 12 months rolling forecast always predicting the following month, and a yearly forecast generated via summing up the monthly in advance forecast values. As the company has not yet used any forecasting tool, the naïve approach of simply assuming the same demand as in the same month of the previous year is used as the baseline model. The main evaluation metric is the Weighted Mean Absolute Percentage Error (WMAPE) due to its characteristic of also being defined if the actual demand is zero. For some analysis also the Symmetric Mean Absolute Percentage Error (SMAPE) and the classical Percentage Error (PE) were used.

None of the models performs universally better on all parts and regarding all prediction strategies. There is no group of items, for which AI-based methods generally outperform the classical approaches. There are, however, single parts for which the more complex Neural Network approaches recognise patterns, which could not be identified by the statistical methods, and therefore deliver a better performance. Despite this, for the majority of parts the AI-based methods seem to overfit, which leads to a prediction quality partly even worse than the performance of the naïve baseline approach. The classical statistical SARIMA method tends to produce the most stable and most accurate monthly predictions across all parts both for the in advance and the rolling forecasts.

The prediction quality varies largely between the groups of parts. While the forecasting quality is similar for the parts in the X-group and in the Y-group, it is significantly worse for the Z-parts. This is expected, as the Z-parts are these parts with the highest irregularity according to the variation coefficient. Monthly predictions do not lead to reasonable results for the Z-parts with WMAPEs of around 100%.

For most parts the monthly in advance predictions could be slightly improved using a rolling strategy. This is expected, as the rolling forecasting scheme uses more recent data points for training. For many parts, however, the WMAPEs both for the in advance and the rolling forecast are around 40%. This emphasises the enormous challenge of forecasting highly-varying real-world industrial data. As the company has not yet used any kind of forecasting tool, an automatic prediction is still of business value, even though the predictions are not highly accurate on a monthly basis.

The yearly amount can be forecasted more accurately. For the X-parts all models achieve a median WMAPE of under 30%, while for the Y-parts only the median of the MLP approach is slightly above 30%. Even for the Z-parts Holt-Winters and the MLP approach lead to median WMAPEs of under 30%. For the yearly prediction, however, the baseline model is one of the best-performing approaches too, which indicates that the yearly parts demand does not vary vastly.

Two additional strategies were investigated to achieve possible improvements of the monthly forecasts through grouping the datasets and through the inclusion of exogenous variables. Both approaches lead to a slight improvement in forecasting quality, but they are also associated with limitations. Grouping the parts according to their category does not always make sense from a business perspective and the inclusion of variables from external providers leads to an increased maintenance effort and is a source of dependence from another institution.

While this study presents an overview of the performance of several forecasting methods on 30 real-world datasets, it also has its limitations, which are pointed out in Section 7.2. Section 7.3 illustrates various possibilities of further research in the field of demand forecasting and also discusses how such a tool can actually be used in a productive business environment.

7.2 Limitations of the study

In this section four main limitations of this study are pointed out and explained. These include that for the optimisation of the whole reordering process lead times would need to be included, that only a subset of the whole bandwidth of parts is analysed, that the parts demand patterns of each company are unique as well as that the models were tested on a limited period of time.

7.2.1 No inclusion of lead times

The timespan that passes between issuing an order and receiving the ordered items is called lead time (Heizer et al., 2024, p. 537). For Aqotec the lead times of the required parts can vary vastly ranging from a few days to multiple months. While auxiliary parts such as gaskets and seals often only have lead times of about a week, special configurations of main components, for example of heat exchangers, can have lead times of up to multiple months. In order to fully optimise the order process and determine the optimal reorder points for each part, not only the demand must be known, but also the lead time for each part (Heizer et al., 2024, p. 537). Currently, however, there exists no reliable record or database within the company that includes the information about the lead times for each item, as the reordering of parts only takes place based on the expertise of the employees. Additionally, it would need to be considered that lead times may vary and are not constant over time. In order to fully optimise the reorder points of each item, however, reliable and up-to-date records of lead times would be necessary. Further projects could therefore also deal with how to acquire those lead times, for example automatically directly from the suppliers' website via crawling algorithms. For many parts and especially for individualised product configurations, this information might however not be publicly available and would therefore need to be updated manually.

7.2.2 Only a subset of data considered

Another limitation of this study is that only a subset of the parts registered in the ERP system of the company was used for forecasting and testing. While through choosing the parts with the highest purchase value or the highest purchase amount in 2024 those items with considerably high business value were selected for this study, the results might still not be fully generalisable to all of the parts. As of the beginning of 2026, however, the export of datasets out of the ERP system of the company in a usable format is rather time-intensive taking up to a few minutes to export the file for one part only, while direct querying is not possible at all. Nevertheless, the company is planning to replace its current ERP system with a more performant one at the beginning of 2027. The introduction of this new ERP system might also make the export of datasets faster and therefore might make it possible to further expand this study to more parts. This could especially be helpful regarding the grouped forecast discussed in Section 6.2. Moreover, the new ERP system might also allow a direct connection between its database and the `R`-script, which would make it possible to carry out the training and forecasting steps completely automatically. The predictions could then be sent back to the ERP system and additionally saved in accessible files, which could later be visualised in a dashboard. The possible deployment of the forecasting tool is discussed in more detail in Section 7.3.1.

7.2.3 Generalisability to other companies

Additionally to the fact that only a subset of parts is considered in this study, it has to be mentioned that while some conclusions can be generalised to the challenge of demand forecasting in other companies, such as that the prediction of sparse demand with a very high coefficient of variation usually delivers less accurate results than for parts with a lower variation coefficient, parts demand patterns are highly individual for each company. It might be possible to generalise some insights to other companies in the exact same field, but the sales market as well as the company's size might still highly influence the demand patterns. Therefore, if some models work well on some item categories for Aqotec, it does not automatically mean they also deliver the same performance for the same group of parts for other companies. What can be, however, generally said is that more complex Neural Network methods do not always outperform classical approaches and therefore, classical statistical methods should always also be taken into account when a parts demand forecasting tool is set up.

7.2.4 Time validity

While the models were evaluated based on different forecasting schemes, including a monthly forecast, a rolling forecast and a yearly forecast, all of the models were evaluated on one testing period ranging from July 2024 until June 2025. This means that while the general statements that the models tend to perform better on parts with lower CVs and that AI-based methods do not always perform better than classical approaches, the exact performance values cannot be transferred to other time periods. It can be, therefore, not confidently assumed that all models that performed well on the testing period from 2024 until 2025 will also show the same performance when forecasting future demand in 2026 or later. Demand patterns might not only change over time due to economic fluctuations, but they might also be influenced by strategic changes in the company. Parts could be gradually phased out or the company might put more focus on different end products than it previously did. Therefore, if an automatic forecasting system is rolled out within the company, it is important to also integrate some kind of quality tracking system, for example in the form of a dashboard, in order to make sure that the quality of the predictions does not suddenly diminish significantly.

Another aspect to consider is that new parts might be used by the company in the future. As explained in Chapter 4, for some parts not a long enough history was available to train a model on. If a forecasting tool for all parts is introduced in the company at some point, also a solution for new items has to be found, whether it is simply using the naïve baseline approach if at least one year of history is available or manual action might be required for this group of parts.

Additionally, it has to be pointed out that the forecasts evaluated in this study are on a monthly basis. This means, that the results can not be generalised to weekly or daily forecasts. A direct prediction of weekly or even daily demand might pose a major challenge as many parts are not used daily or weekly, which would cause the demand patterns to be considerably sparse. There are, however, possibilities to convert the monthly forecasted demand to a finer granularity, for example through simply splitting it into equal parts or based on a specific, individually adaptable business rule.

7.3 Possible further research

In this section various further research possibilities are presented. These include the deployment of the forecasting script, the comparison with additional models as well as the optimisation of the complete reordering process.

7.3.1 Deployment of the forecasting system

This study is carried out based on files which were manually exported once from the ERP system of the company. Subsequently the script in `R` read in all those Excel files after each other and generated the forecasts via saving the results to a csv-file. While the automatic loop over all files is already one aspect of automatisation, the current set-up is not enough to actually be used in a productive environment within the company.

The first step, which is necessary to fully use the forecasting system within a productive environment is to automatise the querying of the data from the ERP system. As pointed out in Section 7.2.2, the current ERP system of the company does not allow to set up such a functionality. The company, however, plans the change from this ERP system to a newer, more modern one from 2027 onwards, which might also offer the possibility for automatic and efficient data-querying. For confidentiality reasons, the names of the current and of the ERP system which will be used in future are not explicitly mentioned in this study. Despite this, it can be concluded that without the setup of an automatic data export or a direct querying possibility no automatisation of forecasting will be possible, as a regular manual export of more than 4000 data files is not efficient and not feasible.

Subsequently, the forecasting application itself needs to be deployed. One option to deploy such a forecasting tool is via `Databricks`. `Databricks` is a so-called open analytics platform, which is cloud-based and which, among other functionalities, allows the deployment of data science tools (Databricks, 2026c). `Databricks` offers the possibility to deploy `R`-code within a job that can either be manually triggered if required or run on a fixed schedule (Databricks, 2026b). On `Databricks` code can also be parallelized if it is slightly restructured to use the `sparklyr` interface, which makes sense in this use-case as more than 4000 models would need to be trained to include all items (Databricks, 2026b).

In the final step, the company might not only want to analyse the forecasts within files, but also visualise the results for specifically important parts in order to get a better overview. There exist numerous different possibilities to create dashboards in a business environment ranging from Microsoft's Power BI (Microsoft Corporation, 2026), to the `R`-package `Shiny` (Chang et al., 2026) and the dashboard functionality directly built into `Databricks` (Databricks, 2026a). When deciding for one option not only the toolbox of functionalities for each of these options has to be considered, but also the cost-benefit trade-off should to be taken into account.

7.3.2 Additional approaches

This study covered two classical statistical approaches, Holt-Winters and SARIMA, and two Neural Network approaches, a Neural Network Autoregression model and a Multilayer Perceptron. Naturally, there are many more possible methods, which can be used for time series forecasting. Studies like from Mediavilla et al. (2022, p. 1128) and from Giannopoulos et al. (2025, p. 18 f.) point out that other forecasting techniques often used in demand forecasting are Machine Learning techniques such as tree-based algorithms as well as more complex approaches such as Support Vector Machines. Giannopoulos et al. (2025, p. 18 f.) also point out that in a number of forecasting studies in different sectors Recurrent Neural Networks (RNN) are used for the prediction of the demand. RNNs are more complex Neural Networks, but they are specifically tailored to sequential data, such as texts and time series data (Aggarwal, 2023, p. 265 ff.). RNNs could also be built in `R` using the package `keras`, which allows to build Neural Network models with a more complex and customised architecture (Allaire and Chollet, 2026). The drawback regarding this approach is, however, that the set-up and the data preparation might be more complex compared to directly using functions such as `nnetar` from the `forecast` package or `mlp` from the `nnfor` package.

Giannopoulos et al. (2025, p, 18 f.) also point out that hybrid approaches are commonly used in demand forecasting, which can combine classical approaches such as SARIMA with more complex AI-based methods. Such models can be easily set up in `R` using the package `forecastHybrid` (Shaub and Ellis, 2026). This package, for example, allows to fit an ensemble of models that combines both the `auto.arima` function and the `nnetar` function (Shaub and Ellis, 2026, p. 13 f.). As no special data preparation is required to fit these models, trying out such hybrid approaches would be an easily implementable and reasonable next step in the research.

7.3.3 Optimisation of complete reordering process

As pointed out in Section 7.2.1, in order to optimise the whole reordering process within the company, the information about the lead times per product is required, because the basic formula to calculate the reorder point multiplies the daily demand with the lead time in days (Heizer et al., 2024, p. 537). While this formula can also be converted to a weekly or monthly granularity, it also points out that the current forecasts on a monthly basis might still be of a too large granularity for being the basis for a detailed optimisation of the reordering process.

Further research could therefore deal with the challenge of how to create reasonable weekly or even daily forecasts and with the hurdle of acquiring detailed and up-to-date informations about lead times. While the lead times for some auxiliary items might stay fairly the same over time, the lead times of main components might vary significantly. Acquiring the lead times of all items and keeping the information up-to-date would therefore require the set-up of a crawling system if delivery times are visible on the supplier's websites to avoid an unfeasible amount of manual work. An exploratory analysis of lead times in the company could already give an important insight into how much the lead times of the articles really vary and what the most efficient and suitable option would be to optimise the gathering of this information.

Bibliography

Aggarwal, C. C. (2023). *Neural Networks and Deep Learning.* Springer, 2nd edition.

Allaire, J. and Chollet, F. (2026). *keras: R Interface to 'Keras'.* R package version 2.16.2. `https://cran.r-project.org/web/packages/keras/keras.pdf` (access: 28th February 2026).

Armstrong, J. S. (1978). *Long-range forecasting: From crystal ball to computer.* John Wiley & Sons.

ArunKumar, K., Kalaga, D. V., Kumar, C. M. S., Kawaji, M., and Brenza, T. M. (2022). Comparative analysis of gated recurrent units (gru), long short-term memory (lstm) cells, autoregressive integrated moving average (arima), seasonal autoregressive integrated moving average (sarima) for forecasting covid-19 trends. *Alexandria Engineering Journal*, 61:7585–7603.

Atwan, T. A. (2022). *Time Series Analysis with Python Cookbook: Practical recipes for exploratory data analysis, data preparation, forecasting, and model evaluation.* Packt Publishing. pdf-version accessed via University of Vienna Online Library.

Box, G. E. and Jenkins, G. M. (1970). *Time Series Analysis. Forecasting and Control.* Holden-Day.

Brockwell, P. J. and Davis, R. A. (2016). *Introduction to Time Series and Forecasting.* Springer, 3rd edition.

Bundesministerium fuer Klimaschutz, Umwelt, Energie, Mobilitaet, Innovation und Technologie (2024). Sauber heizen für alle 2024 heizungstausch für einkommensschwache haushalte. Portal Umweltförderung operated by Kommunalkredit Public Consulting GmbH. `https://www.umweltfoerderung.at/privatpersonen/sauber-heizen-fuer-alle-2024` (access: 20th January 2026).

Bundesministerium fuer Klimaschutz, Umwelt, Energie, Mobilitaet, Innovation und Technologie (2025). Sauber heizen für alle 2025 heizungstausch für einkommensschwache haushalte. Portal Umweltförderung operated by Kommunalkredit Public

Consulting GmbH. `https://www.umweltfoerderung.at/privatpersonen/sauber-heizen-fuer-alle-2025` (access: 20th January 2026).

Canova, F. and Hansen, B. E. (1995). Are seasonal patterns constant over time? a test for seasonal stability. *Journal of Business & Economic Statistics*, 13(3):237–252.

Chang, W., Cheng, J., Allaire, J., Sievert, C., Schloerke, B., Aden-Buie, G., Xie, Y., Allen, J., McPherson, J., Dipert, A., and Borges, B. (2026). *shiny: Web Application Framework for R*. R package version 1.13.0. `https://cran.r-project.org/web/packages/shiny/shiny.pdf` (access: 28th February 2026).

Chollet, F., Kalinowski, T., and Allaire, J. J. (2022). *Deep Learning with R*. Manning, 2nd edition. Kindle version.

Databricks (2026a). Dashboards. `https://docs.databricks.com/aws/en/dashboards/` (access: 28th February 2026).

Databricks (2026b). Databricks for r developers. `https://docs.databricks.com/aws/en/sparkr/` (access: 28th February 2026).

Databricks (2026c). What is databricks? `https://docs.databricks.com/aws/en/introduction/` (access: 28th February 2026).

Douaioui, K., Oucheikh, R., Benmoussa, O., and Mabrouki, C. (2024). Machine learning and deep learning models for demand forecasting in supply chain management: A critical review. *Applied System Innovation*, 7(5).

El-Meehy, A. O., El-Kharbotly, A. K., and El-Beheiry, M. M. (2025). Systematic hyperparameter analysis of gru and lstm across demand pattern types: a demand-characteristic-driven meta-learning framework for rapid optimization. *Scientific Reports*, 15.

Francis, H. and Kusiak, A. (2017). Prediction of engine demand with a data-driven approach. *Procedia Computer Science*, 103:28–35.

Ghribi, Y., Graha, E., and Wicaksono, H. (2025). Comparative analysis of statistical and machine learning models for enhancing demand forecasting accuracy in the medical device industry. *Proceedings of the 58th CIRP Conference on Manufacturing Systems 2025*, 134:849–854.

Giannopoulos, P. G., Dasaklis, T. K., Tsantilis, I., and Patsakis, C. (2025). Machine learning algorithms in intermittent demand forecasting: a review. *International Journal of Production Research*, pages 1–43.

Gonçalves, J., Cortez, P., Carvalho, M. S., and Frazao, N. (2020). A multivariate approach for multi-step demand forecasting in assembly industries: Empirical evidence from an automotive supply chain. *Decision Support Systems*, 142(1). 113452.

Grolemund, G. and Wickham, H. (2011). Dates and times made easy with lubridate. *Journal of Statistical Software*, 40(3):1–25.

Heizer, J., Render, B., and Munson, C. (2024). *Operations Management.* Pearson, 14th edition. Global Edition.

Holt, C. C. (2004). Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, 20(1):5–10.

Hyndman, R. J. (2025). Wape: Weighted absolute percentage error. `https://robjhyndman.com/hyndsight/wape.html` (access: 31$^{\text{st}}$ January 2026).

Hyndman, R. J., Athanasopoulos, G., Bergmeir, C., Caceres, G., Chhay, L., O'Hara-Wild, M., Petropoulos, F., Razbash, S., Wang, E., and Yasmeen, F. (2026). *forecast: Forecasting functions for time series and linear models.* R package version 9.0.0. `https://cran.r-project.org/web/packages/forecast/forecast.pdf` (access: 9$^{\text{th}}$ February 2026).

Hyndman, R. J. and Athanasopulos, G. (2021). *Forecasting: principles and practice.* Springer, 3rd edition. Online version. `https://otexts.com/fpp3/` (access: 10$^{\text{th}}$ October 2025).

Hyndman, R. J. and Khandakar, Y. (2008). Automatic time series forecasting: the forecast package for R. *Journal of Statistical Software*, 27(3):1–22.

Khadka, R., Sanwa, R. L., Khadka, N., and Chi, Y. N. (2025). Forecasting the global price of corn using neural network and hybrid approach. *IEEE Access*, 13:167424–167438.

Khan, F. U., Khan, F., and Shaikh, P. A. (2023). Forecasting returns volatility of cryptocurrency by applying various deep learning algorithms. *Future Business Journal*, 9(1).

Kim, S. and Kim, H. (2016). A new metric of absolute percentage error for intermittent demand forecasts. *International Journal of Forecasting*, 32:669–679.

Kolassa, S. and Schütz, W. (2007). Advantages of the mad/mean ratio over the mape. *Foresight: The International Journal of Applied Forecasting*, 6:40–43.

Kolassa, S. and Schütz, W. (2016). Advantages of the MAD/Mean Ratio over the MAPE. In Gilliland, M., Tashman, L., and Sglavo, U., editors, *Business Forecasting: Practical Problems and Solutions*. John Wiley & Sons.

Konar, A. and Bhattacharya, D. (2017). *Time-Series Prediction and Applications. A Machine Intelligence Approach.* Springer. pdf-version accessed via University of Vienna Online Library.

Kontopoulou, V. I., Panagopoulos, A. D., Kakkos, I., and Matsopoulos, G. K. (2023). A review of arima vs. machine learning approaches for time series forecasting in data driven networks. *Future Internet*, 15(8). 255.

Kostenko, A. V. and Hyndman, R. J. (2006). A note on the categorization of demand patterns. *Journal of the Operational Research Society*, 57(10):1256–1257.

Kourentzes, N. (2023). *nnfor: Time Series Forecasting with Neural Networks.* R package version 0.9.9. `https://cran.r-project.org/web/packages/nnfor/nnfor.pdf` (access: 9th February 2026).

Krispin, R. (2019). *Hands-On Time Series Analysis with R: Perform Time Series Analysis and Forecasting Using R.* Packt Publishing.

Kwiatkowski, D., Phillips, P. C., Schmidt, P., and Shin, Y. (1992). Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? *Journal of Econometrics*, 54(1):159–178.

Liu, X., Lin, Z., and Feng, Z. (2021). Short-term offshore wind speed forecast by seasonal arima - a comparison against gru and lstm. *Energy*, 227:120492.

Makridakis, S. (1993). Accuracy measures: theoretical and practical concerns. *International Journal of Forecasting*, 9(4):527–529.

Mediavilla, M. A., Dietrich, F., and Palm, D. (2022). Review and analysis of artificial intelligence methods for demand forecasting in supply chain management. *Procedia 55th CIRP Conference on Manufacturing Systems*, 107:1126–1131.

Microsoft Corporation (2026). What is power bi. `https://learn.microsoft.com/en-us/power-bi/fundamentals/power-bi-overview` (access: 28th February 2026).

Nandi, A. and Pal, A. K. (2022). *Interpreting Machine Learning Models. Learn Model Interpretability and Explainability Methods.* Apress.

Nemati-Amirkolaii, K., Baboli, A., Shahzad, M., and Tonadre, R. (2017). Demand forecasting for irregular demands in business aircraft spare parts supply chains by using artificial intelligence (ai). *IFAC-PapersOnLine*, 50(1):15221–15226.

Oesterreichische Nationalbank (2026). Konsumentenvertrauen – realwirtschaftliche indikatoren. Telephne survey by Ipsos Austria. `https://www.oenb.at/Statistik/Standardisierte-Tabellen/Realwirtschaftliche-Indikatoren/konjunkturindikatoren/Konsumentenvertrauen.html` (access: 28th February 2026).

R Core Team (2025). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. R version 4.5.2. `https://www.R-project.org/` (access: 9th February 2026).

R Core Team (2026a). *R documentation: HoltWinters {stats}.* R Foundation for Statistical Computing. Part of base R version 4.5.2. `https://stat.ethz.ch/R-manual/R-devel/library/stats/html/HoltWinters.html` (access: 9th February 2026).

R Core Team (2026b). *R documentation: predict.HoltWinters {stats}.* R Foundation for Statistical Computing. Part of base R version 4.5.2. `https://stat.ethz.ch/R-manual/R-devel/library/stats/html/predict.HoltWinters.html` (access: 9th February 2026).

R Core Team (2026c). *R documentation: sd {stats}.* R Foundation for Statistical Computing. Part of base R version 4.5.2. `https://stat.ethz.ch/R-manual/R-devel/library/stats/html/sd.html` (access: 9th February 2026).

R Core Team (2026d). *R documentation: ts {stats}.* R Foundation for Statistical Computing. Part of base R version 4.5.2. `https://stat.ethz.ch/R-manual/R-devel/library/stats/html/ts.html` (access: 9th February 2026).

Rhanoui, M., Yousfi, S., Mikram, M., and Merizak, H. (2019). Forecasting financial budget time series: Arima random walk vs lstm neural network. *IAES International Journal of Artificial Intelligence (IJ-AI)*, 8:317.

Scholz-Reiter, B., Heger, J., Meinecke, C., and Bergmann, J. (2012). Integration of demand forecasts in abc-xyz analysis: Practical investigation at an industrial company. *International Journal of Productivity and Performance Management*, 61(4):445 – 451.

Shaub, D. and Ellis, P. (2026). *forecastHybrid: Convenient Functions for Ensemble Time Series Forecasts*. R package version 5.1.21. `https://cran.r-project.org/web/packages/forecastHybrid/forecastHybrid.pdf` (access: 28th February 2026).

Shumway, R. H. and Stoffer, D. S. (2025). *Time Series Analysis and Its Applications*. Springer, 5th edition.

Spinu, V., Grolemund, G., and Wickham, H. (2026). *lubridate: Make Dealing with Dates a Little Easier*. R package version 1.9.5. `https://cran.r-project.org/web/packages/lubridate/lubridate.pdf` (access: 9th February 2026).

Spyrou, E., Tsoulos, I., and Stylios, C. (2022). Applying and comparing lstm and arima to predict co levels for a time-series measurements in a port area. *Signals*, 3:235–248.

Syntetos, A., Boylan, J., and Croston, J. (2005). On the categorization of demand patterns. *Journal of the Operational Research Society*, 56:495–503.

Talkhi, N., Fatemi, N. A., Ataei, Z., and Nooghabi, M. J. (2021). Modeling and forecasting number of confirmed and death caused covid-19 in iran: A comparison of time series forecasting methods. *Biomedical Signal Processing and Control*, 66. 102494.

Thoen, E. (2024). *padr: Quickly Get Datetime Data Ready for Analysis*. R package version 0.6.3. `https://cran.r-project.org/web/packages/padr/padr.pdf` (access: 9th February 2026).

Vahrenkamp, R. and Kotzab, H. (2012). *Logistik Management und Strategien*. De Gruyter Oldenbourg, 7th edition. pdf-version accessed via University of Vienna Online Library.

Vandeput, N. (2021). *Data Science for Supply Chain Forecasting*. De Gruyter, 2nd edition.

Vandeput, N. (2023). *Demand Forecasting Best Practices*. Manning.

Venkateswaran, B. and Ciaburro, G. (2017). *Neural Networks with R. Smart models using CNN, RNN, deep learning, and artificial intelligence principles.* Packt Publishing. pdf-version accessed via University of Vienna Online Library.

Vogel, J. (2015). *Prognose von Zeitreihen. Eine Einführung für Wirtschaftswissenschaftler.* Springer Gabler.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York.

Wickham, H. and Bryan, J. (2025). *readxl: Read Excel Files.* R package version 1.4.5. `https://cran.r-project.org/web/packages/readxl/readxl.pdf` (access: 9th February 2026).

Wickham, H., Chang, W., Henry, L., Pedersen, T. L., Takahashi, K., Wilke, C., Woo, K., Yutani, H., Dunnington, D., and van den Brand, T. (2026a). *ggplot2.* R package version 4.0.2. `https://cran.r-project.org/web/packages/ggplot2/ggplot2.pdf` (access: 10th February 2026).

Wickham, H., François, R., Henry, L., Müller, K., and Vaughan, D. (2026b). *dplyr: A Grammar of Data Manipulation.* R package version 1.2.0. `https://cran.r-project.org/web/packages/dplyr/dplyr.pdf` (access: 9th February 2026).

Winters, P. R. (1960). Forecasting sales by exponentially weighted moving averages. *Management Science*, 6(3):324–342.

Çera Pinçe, Turrini, L., and Meissner, J. (2021). Intermittent demand forecasting for spare parts: A critical review. *Omega*, 105(1):102513. 102513.

8. Appendix

Part ID	**Naive**	**Holt-Winters**	**SARIMA**	**nnetar**	**MLP**
pa7	56.02	98.83	49.60	48.53	61.26
pa2	28.06	59.22	56.79	57.58	18.65
pv9	72.1	65.26	53.76	79.27	77.56
pa9	70	77.96	43.64	92.01	84.37
pv1	73.27	85.78	57.47	54.43	55.41
pv7	33.16	93.18	51.34	58.96	73.4
pa13	44.93	40.58	49.28	49.28	59.4
pv5	69.47	52.54	62.54	57.54	36.23
pa11	44.65	58.42	35.36	57.22	40.46
pv2	50.06	44.06	42.4	44.06	58.6
pa10	47.54	63.08	45.95	60.13	58.84
pv8	38.67	40.37	42.81	40.94	48.58
pv4	45.1	37.01	38.73	51.96	36.89
pv6	65.33	58.47	45.41	65.2	67.66
pv11	50.96	34.81	44.02	39.83	48.56
pv10	50.38	55.2	53.68	68.4	60.53
pv12	49.28	45.32	41.55	74.28	55.4
pa12	54.62	30.57	32.41	29.63	68.54
pa15	54.46	30.89	32.54	31.55	70.65
pv3	64.05	94.86	48.18	56.68	95.89
pv15	82.35	150	126.47	173.53	179.41
pa6	170.59	95.36	131.54	209.64	79.39
pv14	52.01	69.42	79.02	76.34	80.13
pa13	94.37	77.65	121.69	92.7	87.13
pa5	86.15	89.6	75.64	83.71	77.55
pa3	112.02	442.99	118.9	756.43	731.48
pa1	137.04	159.23	132.46	124.31	176.92
pa14	128	127.84	169.7	308.3	79.98
pa8	76	50.49	79.09	87.82	87.19
pa4	210.97	122.16	96.9	147.86	108.16

Table 8.1: WMAPEs for the 12-months in advance forecasts

Part ID	Naive	Holt-Winters	SARIMA	nnetar	MLP
pa7	48.67	71.86	47.33	46.5	58.42
pa2	23.52	100.39	46.73	121.94	19.43
pv9	72.12	68.56	50.47	58.86	64.15
pa9	77.15	125.06	47.89	77.1	157.19
pv1	70.71	68.01	52.01	59.92	44.11
pv7	46.71	185.17	60.56	71.59	114.86
pa13	50.56	42.29	49.81	49.94	55.09
pv5	70.23	76.41	61.94	57	53.87
pa11	51.1	77.24	36.57	91.21	41.64
pv2	44.9	69.92	48.06	50.54	69.54
pa10	50.74	81.48	51.34	80.16	72.98
pv8	47.2	44.64	49.2	43.83	57
pv4	66.62	43.57	44.89	64.88	43.8
pv6	75.31	58.55	51.33	63.56	83.65
pv11	73.76	41.09	49.39	48.33	52.83
pv10	64.36	67.8	63.78	67.2	73.09
pv12	51.67	51.3	48.74	65.46	61.88
pa12	80.95	32.82	36.79	34.11	105.74
pa15	80.79	33.04	36.92	35.99	112.56
pv3	74.08	174.19	51.48	49.8	175.99
pv15	94.95	102.55	96.28	88.47	109.08
pa6	136.26	122.89	101.78	116.34	116.07
pv14	61.71	85.36	92.81	91.15	95.6
pa13	107.84	139.91	100.79	109.05	84.35
pa5	95.34	86.02	83.53	92.2	85.07
pa3	109.75	141.33	97.26	156.93	143.89
pa1	121.29	121.39	112.98	110.22	114.7
pa14	105.24	136.08	142.85	150.31	114.17
pa8	116.03	65.47	122.25	110.51	147.65
pa4	150	138.99	131.09	149.91	125.6

Table 8.2: SMAPEs for the 12-months in advance forecasts

Part ID	**Naive**	**Holt-Winters**	**SARIMA**	**nnetar**	**MLP**
pa7	56.02	44.83	41.56	50.3	40.24
pa2	28.06	27.86	32.55	19.17	18.12
pv9	72.1	80.18	66.51	60.36	50.23
pa9	70	71.55	59.37	56.45	75.05
pv1	73.27	67.31	55.77	109.36	58.2
pv7	33.16	47.99	44.39	47.06	58.42
pa13	44.93	39.13	51.45	52.9	53.62
pv5	69.47	44.91	57.72	55.96	48.07
pa11	44.65	45.79	41.35	55.65	43.38
pv2	50.06	49.06	49.72	46.17	80.24
pa10	47.54	63.84	45.85	65.64	77.11
pv8	38.67	38.26	52.48	43.46	54.18
pv4	45.1	40.69	45.22	37.75	50.49
pv6	65.33	60.03	45.15	59.38	68.82
pv11	50.96	36.84	46.77	39.35	54.9
pv10	50.38	55.33	57.36	44.54	57.23
pv12	49.28	48.92	37.05	59.71	53.6
pa12	54.62	31.55	35.01	30.61	44.67
pa15	54.46	31.68	35.1	28.6	48.55
pv3	64.05	57.33	49.21	66.2	79.27
pv15	82.35	123.53	111.76	114.71	100
pa6	170.59	108.66	104.97	125.31	139.66
pv14	52.01	76.34	81.7	120.54	86.16
pa13	94.37	79.05	127.49	116.14	100.85
pa5	86.15	81.44	86.2	75.5	99.06
pa3	112.02	195.04	83.39	138.58	130.88
pa1	137.04	157.15	128.56	109.24	129.59
pa14	128	129.37	155.43	228.3	118.41
pa8	76	51.2	66.33	86.02	73.38
pa4	210.97	121.01	103.27	127.02	113.876

Table 8.3: WMAPEs for the 12-months rolling forecasts

Part ID	**Naive**	**Holt-Winters**	**SARIMA**	**nnetar**	**MLP**
pa7	-51.57	-98.83	-48.65	-45.31	-26.59
pa2	-25.81	54.99	-55.25	54.14	2.93
pv9	29.73	-20.84	-44.42	-56.26	-43.85
pa9	1.43	61.01	6.32	-53.44	77.34
pv1	-32.93	-65.86	-39.98	-26.73	-49.57
pv7	27.27	93.18	20.05	33.02	73.4
pa13	-4.35	-26.09	-13.04	-14.49	-46.38
pv5	-18.95	29.74	-44.65	-32.11	-15
pa11	20.5	42.11	-6.89	44.54	7.03
pv2	-13.21	23.42	-18.65	-15.65	38.62
pa10	21.83	39.68	31.87	37.11	49.5
pv8	27.29	27.54	25.91	13.48	36.23
pv4	32.84	18.87	20.34	-7.84	24.88
pv6	-4.01	-7.5	2.46	-25.61	26
pv11	39.95	8.49	30.86	7.78	-8.61
pv10	31.6	15.1	20.43	-46.83	24.49
pv12	3.96	7.55	5.22	-55.58	2.16
pa12	22.03	-21	8.05	11.31	60.38
pa15	22.03	-20.8	8.24	10.12	64.19
pv3	6.35	89.64	28.76	-10.18	85.06
pv15	-64.71	-150	-114.71	-150	-179.41
pa6	-64.71	95.36	-114.68	-172.99	-4.05
pv14	24.33	23.44	33.48	37.5	47.99
pa13	-16.9	17.39	-89.97	-47.99	-79.42
pa5	-25.22	-63.12	-23.82	-17.84	-21.34
pa3	-23.83	-442.99	-101.07	-756.43	-731.48
pa1	-22.34	-91.52	-66.56	-56.69	-144.4
pa14	-32	-19.47	-87.39	-278.52	77.83
pa8	68	17.18	58.8	50.78	72.77
pa4	-10.97	-32.7	16.33	-14.17	22.51

Table 8.4: PEs for the forecasts of the yearly demand

Part ID	**Holt-Winters:** α, β, γ	**SARIMA:** p, q, P, Q, S, d, D	**nnetar:** $p, P, size$	**MLP:** hd
pa7	0, 0, 0.2	0, 1, 1, 0, 12, 1, 0	1, 1, 2	12
pa2	0.06, 0.81, 0	0, 0, 0, 0, 12, 0, 0	1, 1, 2	12
pv9	0, 0, 1	0, 1, 0, 0, 12, 0, 0	4, 1, 3	12;6
pa9	0.09, 0.8, 0.45	1, 2, 0, 0, 12, 0, 0	3, 1, 2	12
pv1	0.03, 0, 0.56	0, 0, 0, 0, 12, 0, 0	1, 1, 2	8;4
pv7	0.3, 0.12, 0.57	1, 3, 0, 0, 12, 1, 0	2, 1, 2	6
pa13	0.01, 0.38, 0.33	1, 1, 0, 0, 12, 1, 0	1, 1, 2	12;6
pv5	0.19, 0, 0.52	1, 0, 1, 0, 12, 0, 0	6, 1, 4	8;4
pa11	0.27, 0, 0.63	1, 3, 0, 0, 12, 0, 0	3, 1, 2	8;4
pv2	0.26, 0, 0.26	1, 0, 1, 0, 12, 0, 0	1, 1, 2	8;4
pa10	0.21, 0, 0.58	4, 0, 1, 0, 12, 0, 0	12, 1, 6	8;4
pv8	0.25, 0, 0.23	2, 0, 0, 2, 12, 1, 0	3, 1, 2	8;4
pv4	0.3, 0.04, 0.72	1, 3, 1, 0, 12, 0, 0	8, 1, 5	12;6
pv6	0.28, 0, 0.37	0, 2, 1, 0, 12, 1, 0	7, 1, 4	6
pv11	0.32, 0.04, 0.54	3, 0, 1, 0, 12, 0, 0	10, 1, 6	8;4
pv10	0.21, 0, 0.17	3, 0, 0, 0, 12, 1, 0	8, 1, 5	12
pv12	0.27, 0, 0.26	0, 3, 0, 0, 12, 1, 0	3, 1, 2	8;4
pa12	0.46, 0, 0.18	1, 0, 0, 0, 12, 1, 0	2, 1, 2	8;4
pa15	0.47, 0, 0.18	1, 0, 0, 0, 12, 1, 0	2, 1, 2	8;4
pv3	0.04, 0.8, 0.2	0, 2, 0, 0, 12, 1, 0	5, 1, 4	8;4
pv15	0, 0, 0.52	0, 3, 0, 1, 12, 1, 0	2, 1, 2	12
pa6	0.21, 0.1, 0	0, 4, 0, 0, 12, 0, 0	12, 1, 6	8;4
pv14	0.15, 0, 0.11	4, 1, 0, 0, 12, 1, 0	5, 1, 4	12
pa13	0.04, 0, 0.1	0, 0, 0, 0, 12, 0, 0	12, 1, 6	8;4
pa5	0.01, 0.21, 0.15	0, 1, 1, 0, 12, 1, 0	5, 1, 4	6
pa3	0.34, 0, 0.49	0, 1, 0, 0, 12, 1, 0	12, 1, 6	8;4
pa1	0, 0, 0	0, 1, 0, 0, 12, 0, 0	1, 1, 2	8;4
pa14	0, 0, 0.58	0, 1, 0, 0, 12, 1, 0	1, 1, 2	6
pa8	0.46, 0, 0	1, 3, 1, 0, 12, 1, 0	3, 1, 2	6
pa4	0, 0, 0.01	0, 2, 1, 0, 12, 1, 0	12, 1, 6	12;6

Table 8.5: The hyperparameters of the fitted models

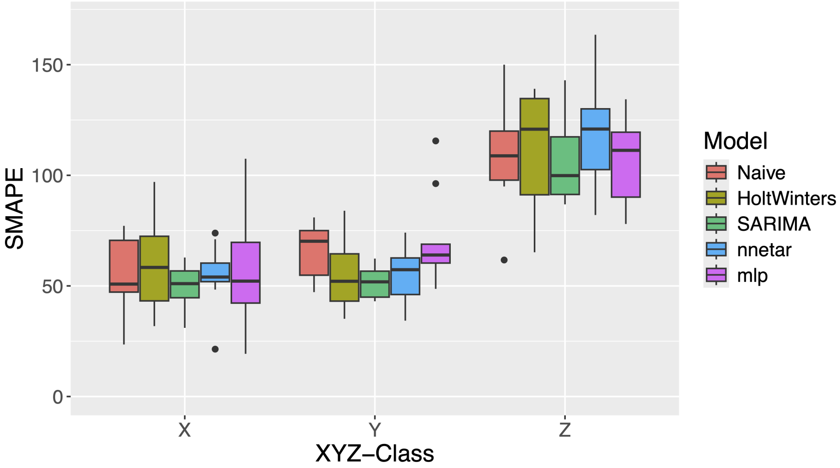


Figure 8.1: Boxplots of SMAPESs per model and per XYZ-group of the rolling monthly forecast

	Cat1-exchangers			Cat1-meters			Cat1-actuators		
Model	m.	r.	y.	m.	r.	y.	m.	r.	y.
Naïve	50	50	-12.8	47.8	47.8	-9.2	37.1	37.1	-9.4
Holt-Winters	40.3	41.5	-15.4				30.8	27.4	14.6
SARIMA	45.7	48.5	-15.5	45.5	33.1	42.9	40.8	30	37
nnetar	51.9	68	1.7	32	33.5	-3.5	29.7	31.4	11.6
mlp	98.3	77.2	-62.4	37.7	38.9	-10.1	59.4	38.3	52.4

	Cat1-regulators			Cat2-valves			Cat3-gaskets/seals		
Model	m.	r.	y.	m.	r.	y.	m.	r.	y.
Naïve	47.7	47.7	36.4	19.6	19.6	2	63.8	63.8	-1.7
Holt-Winters	34.1	39	15	32.8	29.1	24.1	88	74.3	-74.6
SARIMA	41.1	44.1	25.4	37.3	27.4	33.3	94.2	90.8	-84.3
nnetar	34.4	37.4	8.4	29.8	26.8	22.3	151.3	279.8	-140.1
mlp	43	55	38.3	21.7	27.2	8.5	317	95.8	-317

	Cat4-small electrical parts			Cat5-screw accessories		
Model	m.	r.	y.	m.	r.	y.
Naïve	94.5	49.5	-14.9	33.3	33.3	-2.7
Holt-Winters	59.1	60.2	-17.1	38.1	33.4	-34.8
SARIMA	52.8	55.3	-9.1	28.5	28.1	-17.1
nnetar	73	43.3	73	31.5	51.2	-31.5
mlp	82.6	43.8	-35.1	108	38.4	-107.1

Table 8.6: WMAPEs per model for the monthly, rolling and yearly forecast per parts group

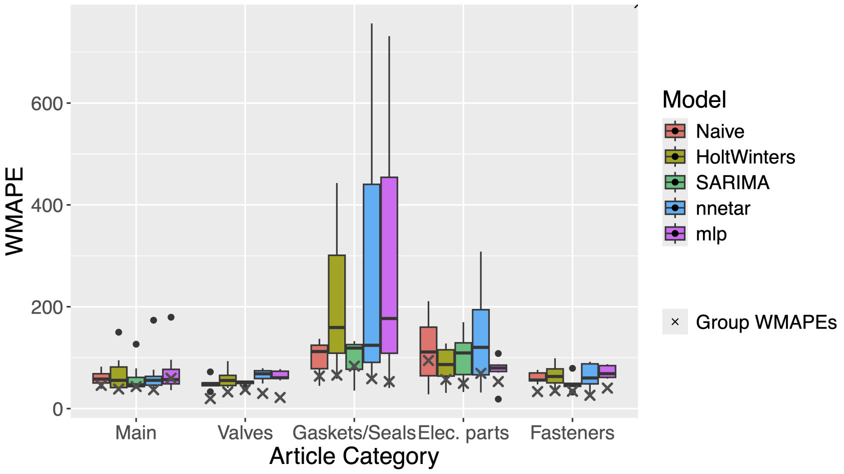
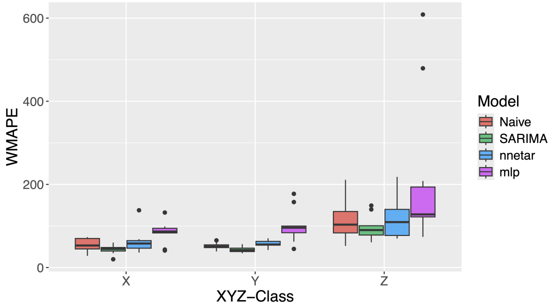
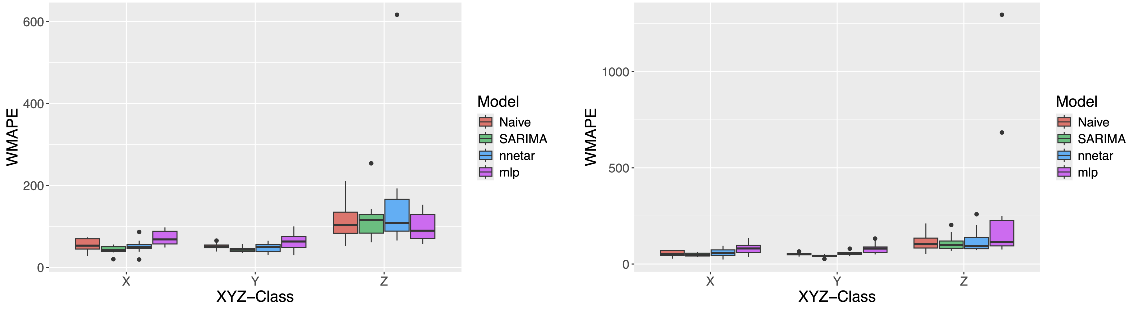


Figure 8.2: Boxplots of WMAPEs per model and per article category of the monthly forecast including the WMAPEs of the grouped monthly forecast without being cut at a WMAPE of 200%



(a) Models including the exogenous variable for the expected price development and the expected bigger investments



(b) Models including the exogenous variable for the expected price development

(c) Models including the exogenous variable for the expected bigger investments

Figure 8.3: Boxplots of monthly WMAPES per model and per XYZ-group for the model including the exogenous variable for the expected price development and for the expected bigger investments without being cut at a WMAPE of 200%

Part ID	**Naive**	**SARIMA**	**nnetar**	**MLP**
pa7	56.02	38.83	46.95	132.32
pa2	28.06	19.92	35.97	82.98
pv9	72.1	53.19	63.33	40.89
pa9	70	42.18	58.96	90.65
pv1	73.27	47.63	137.79	43.26
pv7	33.16	60.03	68.32	95.59
pa13	44.93	43.48	46.38	86.23
pv5	69.47	46.05	44.82	84.04
pa11	44.65	34.48	65.2	98.87
pv2	50.06	48.39	56.6	87.01
pa10	47.54	47.01	66.47	93.09
pv8	38.67	40.05	42.16	80.67
pv4	45.1	37.87	53.68	177.33
pv6	65.33	46.83	65.2	62.23
pv11	50.96	46.05	54.43	157.54
pv10	50.38	56.35	55.96	98.86
pv12	49.28	41.91	55.58	44.78
pa12	54.62	34.35	70.37	95.89
pa15	54.46	34.34	55.14	95.64
pv3	64.05	41.92	43.42	100
pv15	82.35	97.06	135.29	150
pa6	170.59	78.02	141.46	123.34
pv14	52.01	78.57	69.64	130.58
pa13	94.37	87.73	69.99	73.75
pa5	86.15	68.97	79.54	96.53
pa3	112.02	139.72	218.09	608.71
pa1	137.04	101.79	119.47	121.32
pa14	128	149.14	196.14	479.35
pa8	76	60.74	76.34	208.29
pa4	210.97	92.41	99.09	125.15

Table 8.7: WMAPEs for the 12-months in advance forecasts including the exogenous variable for the expected price development and the expected bigger investments

Part ID	**Naive**	**SARIMA**	**nnetar**	**MLP**
pa7	56.02	36.51	38.31	48.53
pa2	28.06	19.95	19.2	62.11
pv9	72.1	50.34	65.26	49.77
pa9	70	40.62	48.39	94.46
pv1	73.27	53.22	86.39	59.17
pv7	33.16	50.4	50.27	75.8
pa13	44.93	42.03	44.93	56.52
pv5	69.47	55.79	57.98	74.47
pa11	44.65	37.87	49.92	92.33
pv2	50.06	42.29	48.39	97.56
pa10	47.54	45.02	65.17	98.85
pv8	38.67	40.78	42.16	61.74
pv4	45.1	37.99	49.63	44
pv6	65.33	46.57	56.53	75.42
pv11	50.96	46.05	56.22	41.63
pv10	50.38	57.36	53.93	74.87
pv12	49.28	41.55	36.87	29.68
pa12	54.62	34.9	29.83	64.31
pa15	54.46	35.22	35.15	61.96
pv3	64.05	47.9	50.23	100
pv15	82.35	120.59	97.06	152.94
pa6	170.59	129.66	192.71	91.55
pv14	52.01	79.91	65.62	71.88
pa13	94.37	111.24	90.99	70.58
pa5	86.15	70.01	80.99	59.19
pa3	112.02	142.2	616.65	136.43
pa1	137.04	127.63	119.73	147.5
pa14	128	254.02	177.69	87.49
pa8	76	61.2	87.65	56.91
pa4	210.97	94.7	132.01	108.27

Table 8.8: WMAPEs for the 12-months in advance forecasts including the exogenous variable for the expected price development

Part ID	**Naive**	**SARIMA**	**nnetar**	**MLP**
pa7	56.02	46.12	65.56	135.06
pa2	28.06	43.62	23.56	36.72
pv9	72.1	57.4	74.6	59.91
pa9	70	41.07	44.77	72.99
pv1	73.27	55.53	95.14	49.7
pv7	33.16	62.43	90.64	89.04
pa13	44.93	47.83	47.83	108.7
pv5	69.47	56.58	45.79	60.26
pa11	44.65	36.09	70.33	93.31
pv2	50.06	38.07	43.06	98.89
pa10	47.54	45.31	79.84	76.71
pv8	38.67	42.57	40.62	49.96
pv4	45.1	37.01	46.81	125.12
pv6	65.33	45.8	60.67	70.25
pv11	50.96	45.81	55.5	132.3
pv10	50.38	52.66	58.88	56.85
pv12	49.28	42.45	57.01	53.42
pa12	54.62	26.6	50.44	85.13
pa15	54.46	31.59	52.08	89.9
pv3	64.05	41.92	58.54	83.94
pv15	82.35	120.59	126.47	120.59
pa6	170.59	98.95	86.89	88.27
pv14	52.01	78.79	84.38	91.29
pa13	94.37	87.39	75.83	75.53
pa5	86.15	69.26	71.96	103.35
pa3	112.02	203.17	258.83	1295.78
pa1	137.04	116.81	101.56	159
pa14	128	167.54	202.29	683.78
pa8	76	79.06	77.8	249.81
pa4	210.97	98.67	143.1	107.36

Table 8.9: WMAPEs for the 12-months in advance forecasts including the exogenous variable for the expected bigger investments