Norwegian University
of Life Sciences

Master's Thesis 2021   M30-ØA
School of Economics and Business
Olvar Bergland and Marie Steen

Forecasting short-term electricity load in Norway – using a dynamic harmonic regression approach and artificial neural networks

Amanda Sophie Aronsen & Marius Liabø Gravem
Business Administration

Preface

This thesis wraps up our two-year master's degree in Business Administration with a specialization in Business Analytics at the Norwegian University of Life Sciences (NMBU).

The subject of forecasting electricity demand is inspired by the courses taken in machine learning, econometrics, finance, and programming, which has sparked the interest for the subject of time-series forecasting. Working on this thesis, we have learned a lot about the techniques and software available, and while we know there is much more to learn about the subject, we hope we can impart some new knowledge to the reader as well. For our thesis advisors Marie Steen and Olvar Bergland, a special thank you is due for their guidance and valuable input throughout the process. We also want to thank family and friends for their continuous support. All errors are of course our own.

Amanda Sophie Aronsen & Marius Liabø Gravem

Ås, August 2021

Summary

In this master thesis, the hourly electricity load in Norway for 2019 is forecasted a day-ahead, using past historical load from Nord Pool, weather temperature, and calendar effects for various holidays. Three models are constructed, one using a Dynamic Harmonic Regression (DHR) model with Autoregressive Integrated Moving Average (ARIMA) errors, and two artificial neural networks; one using Multi-Layer Perceptron (MLP) and another using Neural Network Autoregression (NNAR).

The forecast accuracy is evaluated in terms of Mean Absolute Percentage Error (MAPE) for the five different bidding zones of Norway and the aggregate. The predictions provided is then compared to a baseline seasonal naïve model and with the published forecasts by Nord Pool and ENTSO-E. For the resulting forecast models, the DHR outperforms the other models overall for the all the zones with an average MAPE of 2,73%, ranging from 1,84% for NO, to 3,53% for NO5. As for the NNAR, it performs slightly worse with the average MAPE of 3,38%, ranging from 1,91% to 4,66% for the same zones, but outperforms the DHR model between peak hours, some of the months and during weekdays for NO. Comparatively, the Seasonal naïve achieves an average MAPE of 6,40% across the zones, whereas the MLP fails to beat the baseline at 11,98%.

Sammendrag

I denne masteroppgaven prognostiseres den timesbaserte elektrisitets-etterspørselen i Norge for 2019 en dag i forveien, ved bruk av historisk etterspørsel hentet fra Nord Pool, temperatur, og kalendereffekter for de ulike helligdagene. Tre modeller blir konstruert, en som bruker en Dynamisk Harmonisk Regresjon (DHR) modell med Autoregressive Integrated Moving Average (ARIMA) feil, og to kunstig nevrale nettverk; en som bruker Multi-Layer Perceptron (MLP) og en annen med et autoregressivt nevralt nettverk (NNAR).

Treffsikkerheten til prognosene vurderes etter feilmålet Mean Absolute Percentage Error (MAPE) for de fem ulike prisområdene i Norge og samlet. Prognosene blir sammenlignet med en sesong-naiv referansemodell og mot de publiserte prognosene av Nord Pool og ENTSO-E. For de resulterende prognose-modellene slår DHR de andre modellene totalt sett for alle områdene med en gjennomsnittlig 2,73% MAPE, som varier fra 1,84% for NO, til 3,53% for NO5. NNAR modellen presterer litt lavere med en gjennomsnittlig MAPE på 3,38%, som varierer fra 1,91% til 4,66% for de samme sonene, men slår DHR-modellen mellom forbrukstoppene, enkelte måneder og på ukedager for NO. Sammenlignet med sesong-naiv modellen så scorer den en gjennomsnittlig MAPE på 6,40% på kryss av sonene, mens MLP ikke klarer å slå referansemodellen med 11,98% MAPE.

Table of contents

1. Introduction

Management of the electricity supply is increasingly important for sustaining critical infrastructure and everyday activities. Forecasting the day-ahead electric load is primarily of interest to the producers, market-participants, and Transmission System Operators (TSO) who are given more time and flexibility to plan their production, trade, as well as the maintenance and distribution of power along the electricity grid. By providing accurate forecasts, the costs of production can be lowered due to reduced surplus and deficits, and demand can be readily met with a more effective distribution, preventing the power system from failure.

In this thesis, the hourly electricity load series of Norway for 2019 is forecasted a day-ahead for the five different bidding zones and aggregated using traditional statistical methods and Artificial Intelligence (AI). The two most common methods used in load forecasting is the traditional Autoregressive Integrated Moving Average (ARIMA) and the AI-based Artificial Neural Network (ANN), according to Nti et al. (2020) as shown in Figure 1. As such, the goal of the thesis is to construct forecast models using the two approaches and compare the prediction accuracy obtained using the error metric Mean Absolute Percentage Error (MAPE). The models presented might be beneficial to others trying to predict electric load, whether it is under similar conditions or not.

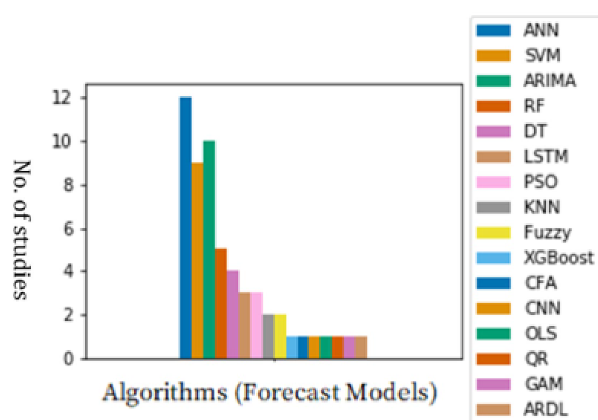

Figure 1: *Common methods for load forecasting, Figure 2 in the review by Nti et al. (2020).*

The forecast models presented is a Dynamic Harmonic Regression (DHR) model with ARIMA errors, which is compared to two ANN: One using the Multi-Layer Perceptron (MLP) architecture while the other uses an Autoregressive Neural Network (NNAR). The DHR and NNAR model are constructed using the R 'forecast' package by Hyndman et al. (2020), while the MLP is built in Python using the 'Sklearn' package. Lastly, the models are compared to a baseline seasonal naïve.

For market participants acting on the Nord Pool power market, knowing how much electricity is demanded a day ahead for specific zones and for each hour is important to ensure a stable supply. To understand the electricity load, it is often characterized by multiple seasonal fluctuations, such as the daily seasonality exhibiting higher consumption during daytime and for the peak hours, lower weekly demand during weekends as opposed to weekdays, and the yearly change in weather temperature affecting the demand for electric heating for instance. Other events such as holidays can also affect the demand, where a large part of the populations behaviour is affected. Therefore, the input variables used to forecast load consists primarily of past historical load, temperature, and calendar effects.

The rest of the thesis will be organized as follows: Chapter 2 outlines the Norwegian power system and its role going forward before discussing the factors associated with electric load. Chapter 3 reviews earlier literature within electricity forecasting and the common techniques utilized within statistics and AI. Chapter 4 describes the collected data, pre-processing steps, descriptive statistics, and error metrics used to evaluate forecasts. Chapter 5 details the theoretical framework for the statistical and AI-based approach, as well as some simpler methods, cross-validation, and data transformations. Chapter 6 considers the steps taken in constructing the models before chapter 7 presents the following results. Chapter 8 discusses the results and other considerations, while chapter 9 concludes the thesis contribution.

2. The Norwegian Electricity Market

Electricity is a key instrument to modern civilization, as a major part of society's communication relies on the internet and other electric devices. Not to mention the considerable volume of machine-made products expected to be readily available using automated processes. As many industries are taking steps to move from non-renewable to renewable energy sources in production, the demand on infrastructure and need for precise energy management will most likely increase in coming years. As a commodity, electricity is considered a non-storable good, where the few options available for storage, such as batteries are generally not yet viable. A key complication with electricity as a supplied good is the necessity to maintain the equilibrium of production and consumption at all times, where imbalances could lead to power failures and its subsequent costs. When discussing electric consumption and load, EnergifaktaNorge (2017) refers to consumption as the electricity used over a period of time, whereas load is the electricity used at a specific point in time. In this thesis however, the load and consumption are referred to interchangeably as the primary consideration is the hourly consumption.

The advantage of predicting the electric load provides market participants with an approximation of the future load, for instance a day or week ahead, and is important information when planning future production and transmissions. Accurate load forecasts facilitate producers and utility companies to reduce their risk and improve resource utilization by distributing more electricity in the local area to reduce transmission costs, enabling producers to generate electricity using the least expensive technology. Future investments can be planned based on economic and demographic growth in the area, and maintenance can be scheduled for periods with lower demand (Mill, 2016).

With increased focus on reducing carbon emissions, an efficient resource utilization in the market is essential to meet future power demand and expectations. In the market analysis by Statnett looking forward from 2020 to 2050, the European power system is heading towards zero emissions based on a higher usage of renewable energy sources; primarily wind and solar power (Statnett, 2020). If the European energy system transitions into more renewable energy sources, load forecasting will only become more valuable as the demand for coordination increases to facilitate a secure energy supply going forward.

Norway consists of 5 bidding zones, also referred to as price areas, going from NO1 to NO5 as illustrated in Figure 2. Each price area has its producers and distributors conducting trade with each other and with other European countries connected to the Nord Pool power market.

Figure 2: *Illustration of the Norwegian price areas (Nordpool, 2021)*

2.1 The Nord Pool power market

Nord Pool AS is owned by Euronext, and the Nordic and Baltic Transmission System Operators (TSOs). Through Nord Pool, electricity is traded from producers to distributors, and between connected bidding areas. The open and highly regulated market for electricity ensures equal terms and promotes more efficient use of resources through competition.

Nord Pool has one intra-day market and one spot market for the day-ahead, trading power for delivery within the same day or within the next 24 hours, respectively. For the day-ahead, the electricity price is determined for each hour based on the submitted bids and offers which make up the supply and demand curves the day before delivery. All orders are matched on the pan-European market using the market integration algorithm EUPHEMIA, matching the bids and orders across the European market while also taking the available transmission capacities provided by the TSO's into account (NordPool, 2020)

For each price area, electricity is supplied by the local producers until their marginal cost of production equals the price in market equilibrium, or according to their capacity. The market adaptation of the producers can be extended to provide electricity to the other zones as well, reducing price differences in the market and ensuring a more robust power grid. Therefore, market coordination results in electricity being produced more efficiently, at lower costs and benefits the consumers with overall lower prices and reduced volatility. Differences in price between the bidding areas is usually caused by congestion in the power grid when there is not enough transmission capacity to trade electricity. As such, Nord Pool calculates the theoretical 'System price' under the assumption of no congestion, where the whole European market has a uniform electricity price. For one bidding area, the local production can either be in balance, or deviate with a deficit or surplus depending on the market conditions. If one area has a surplus of production at a low cost, while another has a deficit with a high price, the price difference can be reduced through trade as low-priced electricity flows to the higher price areas.

The available production technologies have different CO_2 emissions associated with its respective energy source. According to an expert survey on climate change economics, the median estimate for an appropriate social cost of carbon was $50 per ton in 2009 (Howard & Sylvan, 2015), however, it is believed to be even higher. Thus, the carbon tax was implemented as a means to charge the producers of fossil power generation for the negative effects of carbon emissions and shed light on the social cost of CO_2-emmisions. This gives the producers of clean renewable energy a competitive advantage in the energy market, as well as owning to the fact that the polluter, the fossil fuel power plants, should pay for emissions. When calculating the Levelized Cost of Energy (LCOE), a measure in cost per produced electricity unit that determines the break-even price for the power generation, the carbon tax is a substantial part of the LCOE for fossil fuel power plants. IEA has found that LCOE for low-carbon generation technologies are declining. With the assumed moderate emission costs of USD 30/tCO_2, the cost of low-carbon generation is now competitive to fossil fuel-based electricity generation (IEA, 2020).

The carbon tax also incentivizes producers on existing power plants to generate electricity from renewable sources first, before relying on, carbon tax included, less cost-efficient fossil sources. In order for fossil sources to be profitable, the production costs and emission taxes have to be outweighed by a higher electricity price, whereas 'free' energy sources such as water, wind or solar power can be considered worth producing as long as the price is above zero.

Transitioning from thermal power sources to more renewable sources is expected to increase the volatility of the electricity price and production, as unregulated power sources such as wind and solar power are highly dependent on optimal weather conditions compared to the storable fossil energy sources.

2.2 Electricity production and transmission

Norway has a highly secure supply of renewable energy due to storable hydropower in reservoirs, while most other European countries are dependent on fossil sources for thermal power. According to EnergifaktaNorge (2017), Norway has an installed production capacity of 37 680 MW, supplying 154,2 TWh in 2020. Primarily, 88% of the Norwegian energy production is split between 1 681 hydro power plants, another 10% from 53 wind power plants, while the remaining (<2%) production comes from 30 thermal power plants. This flexible energy mix gives Norway an advantage to regulate its production with hydro power according to the demand, provided there is available water in the hydro reservoirs. The reservoirs enable for quick adjustments of production at a low cost, where the stored capacity can be stretched over longer periods even when there is little to no rain. Thermal power is often used combined with unregulated power sources as the last resort to fulfill the demand for power, often stationed close to large scale industries as an additional security of supply. While Norway enjoys a high flexibility in production, Statnett's prognosis for 2050 expects future production from wind and solar power to increase with 44 and 10 TWh respectively, while hydro power will increase from 139 to 152 TWh (Statnett, 2020, Figure 11-3). This will affect the earlier discussed flexibility to regulate production, especially if this increase includes run-of-river hydro power.

The transmission grid is operated by Statnett SF, the designated Norwegian TSO, being responsible for Norway's central grid below the Ministry of Petroleum and Energy. They are tasked to operate and develop the grid based on the needs of today and predicted requirements for the future, as well as calculating the available transmission capacities and adjusting any imbalance in the power equilibrium. The power grid is split into the transmission grid, regional grid, and the distribution grid. The transmission grid is used for long-distance transportation of high voltage electricity and is what carries power across price zones and across country-borders. The regional grid acts as a link between the transmission grid and the distribution grid, where the latter carries low voltage electricity to small consumers. For secure supply of electricity, Statnett is responsible for operating a grid capable of adequate transmission capacity to handle the varying peaks in consumption (EnergifaktaNorge, 2017).

Together with ENTSO-E, the European Network of Transmission System Operators for Electricity, Statnett is developing the cross-border interconnectors towards the goal of making the European market more integrated and flexible against price differences. Since 2017, the transmission capacity in and out of the Nordic region has been planned to increase from 6 200 MW to 9000 MW in 2021 (EnergifaktaNorge, 2017; Statnett, 2020).

2.3 Demand for electricity

The categories of important factors affecting the electricity load is related to the economy, calendar effects, weather, and random disturbances (Nti et al., 2020). When trying to predict the future, lagged values of the dependent variable tend to explain a large fraction of the movements, this should also hold true when predicting electricity load. The underlying relationships between the load series and other exogenous variables gives a fundamental understanding of the load patterns.

Economic factors are such as the industrial activities present in the area, population, income, size of homes and consumption habits. These elements represent factors that change slowly, affecting the long-term load. For the Norwegian consumers, Statistics Norway found the short-term price elasticity to be close to zero (Holstad & Pettersen, 2011).

Electricity prices in Norway are low-priced compared to other European countries, therefore, electric heating is very common. Low-cost electricity is also viewed as an important competitive advantage by Norwegian industries. From January 2020, fossil oil heating was banned in Norway, a law that was announced as early as 2012. While some consumers can substitute their consumption of electricity to oil or firewood, a small fraction of consumers have that opportunity. While 50% of households had a wood stove in 1993, only 39% did in 2009. Reducing the consumption of electricity in a household might be experienced as a significant decrease in the level of comfort. For consumers that can substitute their consumption of electricity to other energy sources, the price of oil, gas, and firewood for instance can have an impact on load. While the total consumption of electricity in Norway has increased in the period 1993-2017, the consumption compared to activity is more efficient, both for industrial and individual consumers (Aanensen & Holstad, 2018). The available variable reflecting short-term economic factors are price, which exists in spot, day-ahead, and forwards. The day-ahead and forward-prices contain its own predictions about load and risk-premiums.

Time affects the load pattern in several ways, there is variation in the load throughout the day, different routines on separate weekdays, yearly seasonality from summer to winter, and holiday effects. When consumers change their behaviour at specific times, it affects the total burden on the electricity grid, especially if the load peaks at certain times. Time features can typically be captured by constructing dummy variables, containing binary-encoded information for which hour, weekday, and month it is. Similarly, holidays or other special occurrences can be aggregated or constructed separately to capture the effect on load. Depending on how the dataset is constructed, this can amount to large numbers of variables, where for instance, a dummy variable for each hour will result in 24 features in the dataset, and so on.

Climatic factors also influences the load. As temperatures rise and fall, it affects our need for electric heating or air-conditioning. Wind, precipitation, humidity, and solar radiation can affect the perceived temperature and consumer behaviour. In Norway, 70% of indoor space is heated by electricity, where the electric consumption will increase as the outdoors reach sub-zero temperatures. Temperature affects the load in two ways: one is that the peak load depends on the minimum temperature. Secondly, the accumulated need for heating in a year depends on degree-days. Degree-days in Norway are defined as days with mean-temperatures below 11°C during fall, and a mean-temperature below 9°C during spring. This unsymmetrical heating requirement exists due to a stronger sun radiation in spring. Wind and cloudiness affect electricity consumption, but to a small extent (Wangensteen, 2012).

The effect of temperature on electricity load is often captured as a non-linear relationship, where the accumulated need for heating/cooling can be approximated using heating- and cooling degree-days (HDD and CDD). To reflect the temperature effect on load, HDD and CDD can be constructed as the number of degrees below or above a reference temperature. As for in-between, we have a comfort zone between the two reference temperatures where no adjustment is required. However, the comfort zone is subject to some research, where it is assumed to be different across geographical areas due to acclimation. In tropical areas the comfort zone will start and end at a higher temperature than in temperate areas. The comfort range is found to be approximately 7°C (Wang & Bielicki, 2018). The load can also be affected by the lagged effects of the temperature, as decreasing, or increasing temperature has a delayed effect on the indoors temperature.

Furthermore, random disturbances make load forecasting challenging. These might be operational difficulties in large industrial plants, outages, or unexpected behaviour by consumers, such as everyone charging their electric cars simultaneously. Random disturbances are hard to account for in technical analysis and is more likely to be picked up by qualitative models providing a more fundamental understanding of the system.

There are several challenges to forecasting electricity load. Time horizon, model complexity, seasonality, geographical and behavioral inconsistencies. Short-term predictions are more likely to be reliable than long-term ones, especially for load forecasts relying on the validity of future weather forecasts. Some forecasting models will require an understanding of the underlying factors to obtain reasonable predictions, as some exogenous variables will affect the load patterns differently across regions. For a specific country, national and religious holidays will differ, technology and preferences can vary greatly for instance if electric heaters are commonly used or not, and which size and isolation is standard for residential homes.

3 Existing Literature on Load Forecasting

Electricity demand forecasting is split into several categories depending on the time horizon of the analysis, categorized as short-, medium-, and long-term load forecasts (STLF, MTLF, and LTLF respectively). MTLF is usually used to forecast months ahead, whereas LTLF is used to forecast years ahead. In a review by Nti et al. (2020), STLF is found to be the most common due to its importance for the day-to-day operations and planning for the market participants. While shorter periods can be forecasted as well, it is mostly used for real-time applications, whereas MTLF and LTLF is used in long-term strategic planning such as scheduling maintenance and policy implementations (Mir et al., 2020).

In the review by Nti et al. (2020), they find artificial intelligence (AI) and statistical time series to be the most popular methods, with the top three being Artificial Neural Networks (ANN), Autoregressive Integrated Moving Average (ARIMA), and Support Vector Machine (SVM). Out of the top 10 most used algorithms, 9 were AI-based, with ARIMA as the exception. For statistic time series models, previous values of the load are often used along with exogenous variables to forecast, combining correlation and extrapolation techniques.

Correlation techniques utilizes measurable exogenous variables tied to the electric load to predict the future load and is often useful in investigating the relationship between the variables. A common technique is the Linear Regression (LR), where the variance between the dependent variable and the exogenous variable is minimized to estimate a best fit for the model parameters. Several techniques are based on the simple LR, such as the Multiple Linear Regression (MLR) for multiple exogenous variables, and the Dynamic Regression (DR) for time-varying parameter estimates. According to Jacob et al. (2020), simple LR proves to be a popular method in forecasting load despite often being outperformed by more complex models.

Extrapolation on the other hand uses the historical trend in the time series and assumes that the previously observed pattern will continue in the future. This is often the case when consumers behaviour has temporal dependency, where the previous value of the dependent variable is a good indicator for future values. According to Mir et al. (2020), the main advantage of extrapolation models is the ability to forecast reliably without exogenous variables. However, a disadvantage is that they can't be used to gain insight in the underlying determinants like correlation techniques can. The simplest example of an extrapolation technique is the autoregression based on Box-Jenkins ARIMA.

The ARIMA model consists of the autoregression (AR) and the moving average (MA). AR estimates the dependent variable using its previously observed values as exogenous inputs similar to the LR model, whereas MA uses past values of the forecast errors to predict the dependent variable, similar to AR. Combined we get the Autoregressive Moving Average (ARMA) model, where the time series is assumed to be stationary, often done by differencing to obtain the Integrated ARMA model (ARIMA). Additionally, the ARIMA can be extended to include exogenous variables in an ARIMAX model, or seasonality by using periodic differences or lags for a seasonal ARIMA (Weron, 2014). Another popular model is Exponential Smoothing, where past observations are weighted to decrease exponentially based on a smoothing parameter, giving recent observations a bigger weight than old observations. Adding additional smoothing parameters, the model can also account for trend and seasonality in what is known as the Holt-Winters Exponential Smoothing model (Jacob et al., 2020).

Seasonality can also be addressed using a similar-day approach, where days with similar load characteristics are chosen, such as hour of the day or weekday. This approach has been used by Weron and Misiorek (2005), where they find that specifying an Autoregression with exogenous variables (ARX) for each hour to outperform a single specified ARIMA, but performs slightly worse than a DR model.

Similarly, Fan and Hyndman (2011) develop a STLF model using a regression framework for each half-hourly load using temperature, calendar-effects, and lagged demand as predictors. They obtain an out-of-sample Mean Absolute Percentage Error (MAPE) of 1,88%. As for multiple seasonality, Hyndman and Athanasopoulos (2018) note that many methods are unable to account for more than one seasonality, but they can be included by using for instance external regressors in an ARIMA. In a paper by Elamin and Fukushige (2018), they use a SARIMAX model with dummy variables for the three seasonalities in the hourly load data. Additionally, they include dummy interactions between the seasonal dummies and other exogenous variables to further reduce the model errors. Alternatively, Yukseltan et al. (2020) use Fourier analysis with feedback to capture the seasonal variations in load data without any exogenous variables. They obtain a 2,9% MAPE for day-ahead predictions, and by applying feedback they can correct prior errors to obtain a 0,87% MAPE hour-ahead forecast.

Compared to the traditional statistical methods, Artificial Intelligence (AI) can be mistaken as newly emerged techniques, however, ANN dates back to 1943 as proposed by McCulloch and Pitts (1943).

In the paper by Kandananond (2011), three methods for forecasting electricity demand in Thailand is compared, using ANN, ARIMA, and MLR. Normally, ANN structure is based on the neural network Multi-Layer Perceptron (MLP) architecture. However, in this work MLP is also compared to a Radial Basis Function network (RBF), concluding that MLP was superior to RBF, ARIMA and MLR. Although the MAPE was better using MLP, the difference between the methods were not significant at $\alpha = 0.05$. Mordjaoui et al. (2017) achieved better results using a dynamic neural network compared to a Holt-Winters and ARIMA model, when predicting daily power consumption from a French transmission system operator.

The MLP is one of the most popular and successful methods used for predicting energy production and consumption according to Koprinska et al. (2018). They wanted to see if Convolutional Neural Networks (CNN) would perform better or likewise, creating models forecasting solar power and electricity load for the day-ahead on four different time series collected from three different countries. The CNN was compared to MLP, Long Short-Term Memory (LSTM) recurrent neural networks and a baseline. They found that CNN and MLP performed similarly and with more precision than LSTM and the baseline.

On the other hand, Kychkin and Chasparis (2021) find that their MLP model performed better than a Holt-Winters, SARIMA and Persistence-based Auto-Regressive (PAR) model, but inferior to the Seasonal Persistence-based Regressive (SPR), in terms of the Root Mean Squared Error (RMSE), when predicting day-ahead load for a group of residential buildings on a 15 minute basis.

Chow and Leung (1996) successfully improved STLF using a nonlinear autoregressive integrated neural network, a hybrid model between ARIMA and NN. In 2019, Yazici et al. (2019) used a non-linear Autoregressive Neural Network (NARXNet) to predict short term load in Istanbul, achieving a MAPE of 1,35% over a period of three months. Lass et al. (2020) used a non-linear Autoregressive Neural Network with exogenous inputs and Genetic Algorithm (NARX-GA) to forecast monthly electric load on an automobile assembly plant, accomplishing a MAPE of 0,56%.

Neural networks can also be used to extract features from historical electricity load that can help predict future load. He (2017) created a Deep Neural Network (DNN) with variables constructed with CNN components to extract rich features from historical load series and used recurrent components to model the dynamics of the series. Dense layers were used to transform other types of features. Likewise, El-Hendawi and Wang (2020) uses wavelet transformations of historical load and other features to train a neural network applied on the electric market of Ontario, Canada.

Haben et al. (2019) found that temperature is not an important factor in short-term forecasts on low voltage grids, on the contrary it sometimes had a detrimental effect on accuracy. While on high voltage grids, temperature does have explanatory power. One reason can be the strong correlation between temperature and annual seasonality. But the finding was not consistent among all their test subjects. This can have an impact on how to forecast large areas, where both low and high voltage grids are included, with both individual and industrial consumers.

While a variety of methods have been tried to STLF, each come with their own strengths and weaknesses where there are no clear-cut winners. In the findings of Suganthi and Samuel (2012), models are often developed for a specific country or purpose, and may therefore lack comparability.

4 Data and descriptive statistics

In this section, the collected time series are described, inspecting some of their properties before discussing the variables constructed as model features. Some of the challenges and pre-processing steps are addressed, before showing the descriptive statistics in Table 2.

4.1. Electricity load data

The electricity load data is collected from Nord Pool in the period 01/01/2013 to 31/12/2019 for hourly observations.

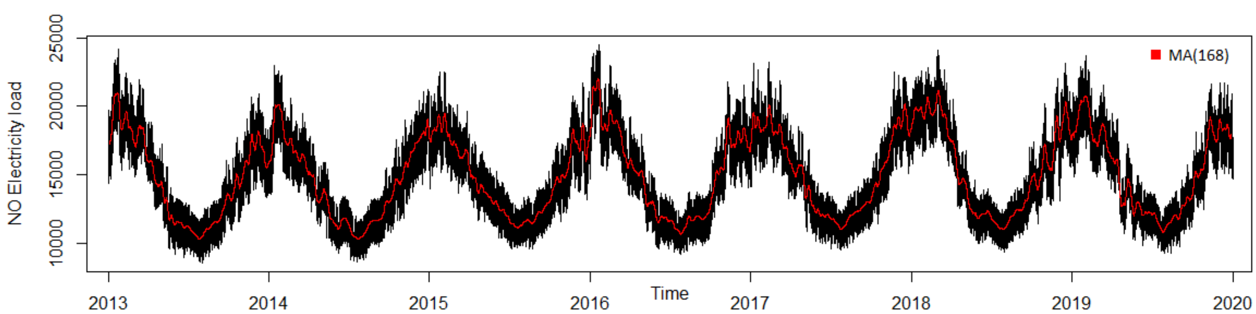

***Figure 3**: Hourly NO electric load in MWh for 01.01.2013 to 31.12.2019.*

The aggregated Norwegian consumption is plotted in Figure 3, the annual changes can be observed throughout the season, ranging from 8 000 MWh in the summer, to approximately 24 000 MWh in winter. It does not appear to reveal any upgoing or downward trend patterns. The graph displays a clear yearly seasonality, with higher load during winter and lower consumption during summer months. From the overall consumption, it is difficult to discern other patterns due to the strong effect of the yearly seasonality.

Averaging the load for each hour, Figure 4 displays the daily seasonality. Here, the consumption changes throughout the day, generally starting with a sharp increase in consumption for the morning hours, before declining at 11:00-12:00 and showing another slight increase around 16:00, often categorized as the peak hours. After 20:00 the consumption gradually declines throughout the night before starting anew in the morning.

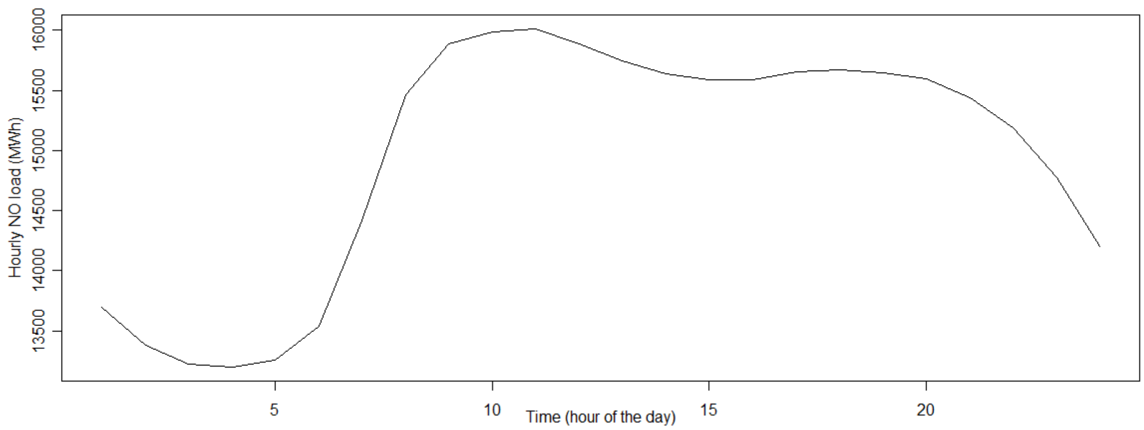

Figure 4: *Average NO electric load in MWh for each hour of the day.*

To obtain the weekly seasonality, the average of each hour across the week is plotted in Figure 5, displaying how the weekly pattern normally behaves. The start of the week behaves quite similar through Monday to mid-day Friday, while Saturday and Sunday exhibit a different shape and a lower load profile. It seems apparent that the data contains multiple seasonal patterns, changing annually, weekly, and hourly.

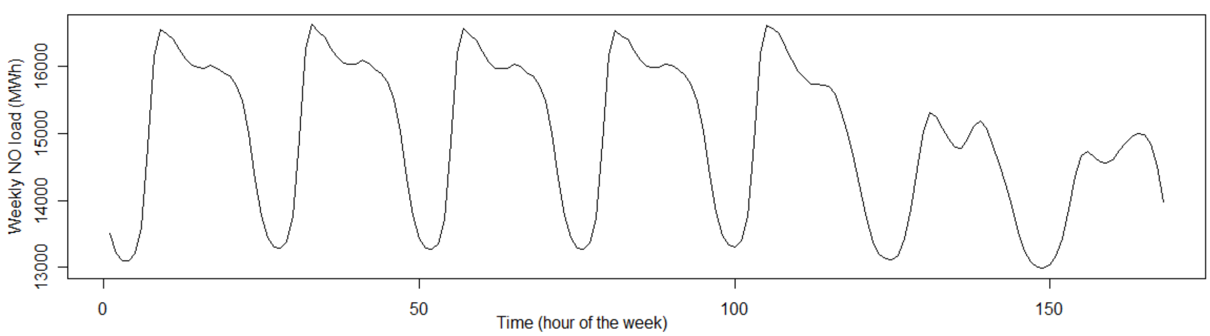

Figure 5*: Average NO electric load in MWh for each hour through the week.*

As proposed by Weron and Misiorek (2005), a model option is to create 24 separate models for each hour to ease the model estimation with less disturbance from the other hours, which could improve predictive power. Daily electric load for

09:00 for 2018-2019, is shown in Figure 6. It appears to have a more consistent load pattern during summer, whereas winter shows more varied consumption.

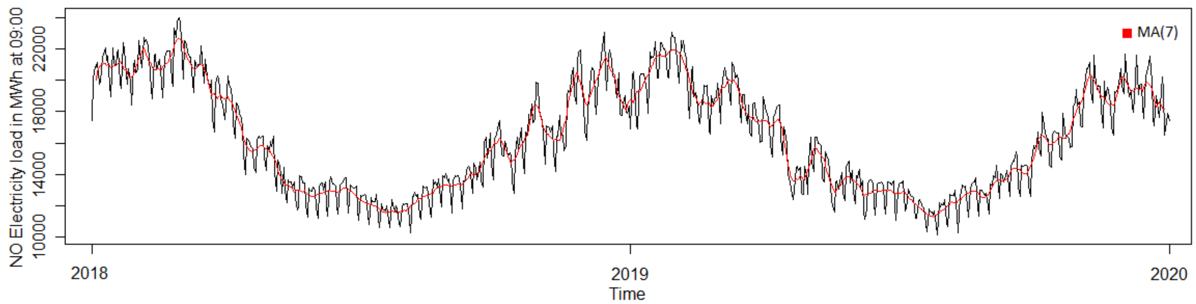

Figure 6: *Daily NO Electric load in MWh at 09:00 for 01.01.2018 to 31.12.2019*

4.2. Weather data

Historical weather data is collected using the Frost API from the Norwegian meteorological institute, acting as a proxy variable for actual weather forecasts, collecting the hourly observations for air temperature, precipitation amount and wind speed.

Table 1: *Weather stations used for each bidding zone*

Zone	Source station	Location
NO1	SN18700	Oslo
NO2	SN44640	Stavanger
NO3	SN68860	Trondheim
NO4	SN90450	Tromsø
NO5	SN50540	Bergen
NO		Average NO

For each bidding zone, a representative weather station has been chosen without too many compromising gaps in the observations (see Table 1). Choosing only one representative weather station for the zone simplifies the model inputs, where a more complex input could have been constructed using multiple locations and applying weights based on population densities for instance. However, using one representative weather station for each zone should be able to capture the overall variation well enough. To obtain the weather inputs for the aggregated NO, an average of the chosen weather stations has been constructed.

Plotting the temperature for the period reveals the negative correlation against the electric load, with warm temperatures combined with less consumption in summer and vice versa for cold winter temperatures (Figure 7). Plotting the daily temperature at 09:00 for 2018-2019 in Figure 8 displays the periods with cold or warm weather more clearly.

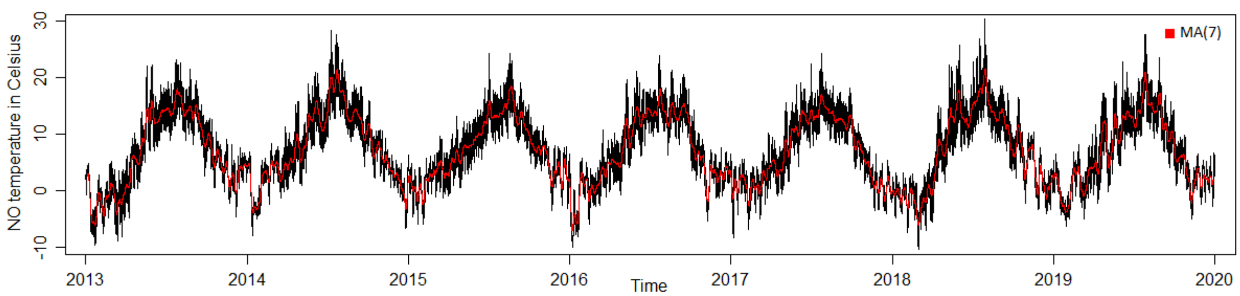

Figure 7: *Hourly NO Temperature for 01.01.2013 to 31.12.2019.*

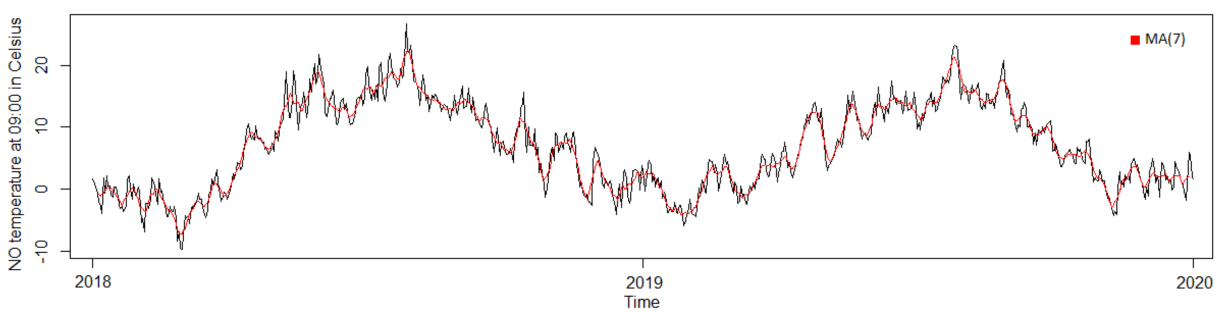

Figure 8: *Daily NO Temperature at 09:00 for 01.01.2018 to 31.12.2019*

Temperature has a non-linear influence on electricity consumption. To show their relationship, temperature and electric load is plotted in Figure 9. The average negative correlation is 82%, where the shape resembles a wave representing the non-linearity. The steepest part of the scatterplot is around 5-10 degrees Celsius, but as it gets colder the electricity demanded seems to diminish. When it gets warmer, there is a declining reduction in consumption, with the lowest point around 15-18 degrees Celsius before it increases slightly. This might show how

Norwegian electricity is mainly used for heating, with air-conditioning being used to a lesser extent compared to countries with warmer climates.

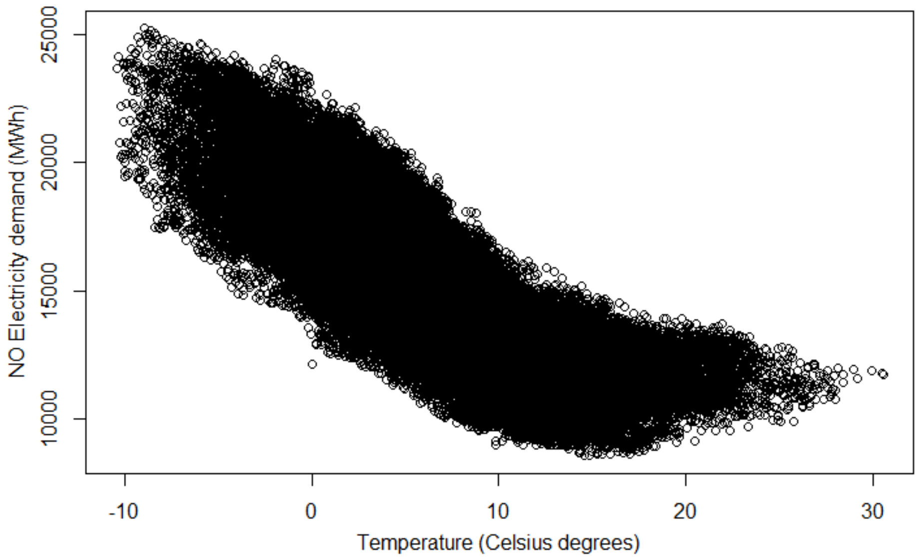

Figure 9*: The relationship between NO Electricity load and temperature (average NO1-NO5).*

As an attempt to capture the non-linearity, variables for heating- and cooling degree-days (HDD and CDD) were constructed as discussed in section 2.3, where a lower limit of 15,5 degrees and an upper limit of 22 degrees were used as reference temperature. Variables for the minimum-, maximum-, and daily average temperature were created as well, in addition to a dummy capturing "degree-days", where daily average was below 9°C during the first six months of the year and below 11°C during the last six months of the year according to the definition outlined by Wangensteen (2012).

4.3. Calendar effects

As discussed in section 4.1, one issue is accounting for the various dates and events affecting the load over time, specifically for the reoccurring variance. In the load data, there are three types of seasonality giving varying levels of consumption, features that should be captured in the model.

Power consumption is dependent on the type of day, whether it is a workday, weekend or holiday, and which hour, day, and month it is. Norway has multiple public holidays where the effect can be captured using a binary encoded dummy. Using this feature, the model can control the holiday effect for those days relative to the regular consumption (Figure 10). Alternatively, separate features can be used to capture types of holidays, for instance, one for Christmas, Easter and so on, as these holidays might have different implications on the load.

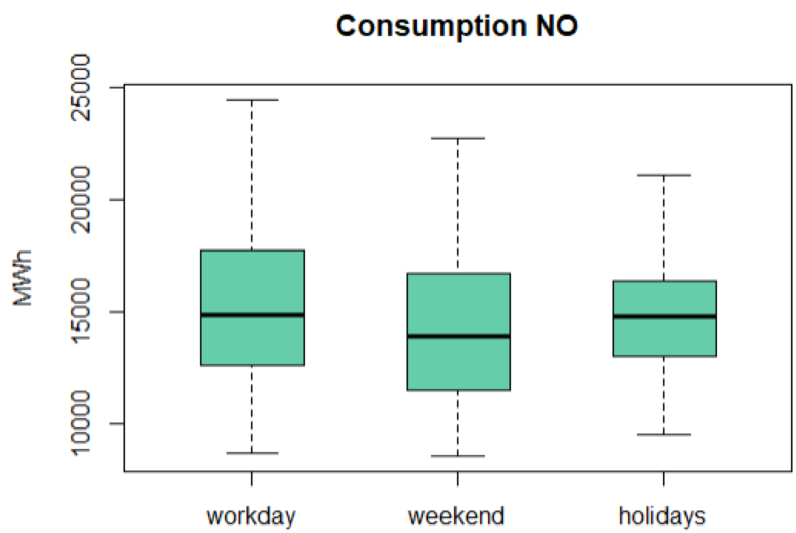

***Figure 10**: Different load for weekdays, weekends, and holidays*

To account for the calendar effects discussed, dummy variables that captures the multiple seasonalities in the data, weekdays, and months, has been constructed, along with a holiday dummy for selected Norwegian holidays. See Appendix 1: Norwegian holidays for an overview.

4.4. Data pre-processing

The collected time series data has been pre-processed to deal with missing values and outlier observations that introduce unnecessary noise to the data. Outlier observations in the electricity load may be caused by outages or measurement errors from the TSOs. To identify the outliers, the first difference of the hourly load series was taken and sorted from large to small. By examining the largest differences found in the level series, observations with abnormal deviations were removed and replaced by an average of the subsequent observations. As for missing values, these can be imputed unless there are gaps spanning for several hours or longer, which might compromise the data. When choosing the weather stations, only the ones with few separate missing values were selected and imputed in a similar manner as the outliers.

4.4.1. Daylight saving time and leap years

One frequent issue for time series is the inconsistency caused by daylight saving time (DST). In the spring, the clock is forwarded by one hour as the daylight lasts longer and is reversed by one hour in autumn as the days grow shorter. In the collected data, this is often observed as missing values in the spring, and a double post in autumn. To correct this, missing values were imputed in spring, and the twin observations in autumn were removed.

Another inconsistency in long-spanning datasets is the presence of leap years, for instance for 2016 with the addition of February 29th adding an additional 24 hours in the otherwise yearly 8 760 hours. With the additional day, forecasting over longer periods can result in a de-synchronization over time in the presence of leap years. One approach could be to remove it altogether to keep a consistent 8 760 hours a year, where the excluded day could be modeled separately (Hyndman & Athanasopoulos, 2018, Ch. 2.1). Due to forecasting only for 2019, we leave the leap year in the training data as the de-synchronization should not affect the model results in a significant manner.

4.5. Descriptive statistics

The descriptive statistics for the collected load and temperature across the bidding zones are displayed in Table 2.

Table 2: *Descriptive statistics. Load in MWh and temperature in Celsius.*

Variable	Mean	St.dev.	Median	Max	Min	Kurtosis	Skewness
NO_load	14 943,97	3 230,61	14 585	24 485	8 558	-0,83	0,34
NO1_load	4 116,90	1 346,96	3 923	8 565	1 674	-0,75	0,40
NO2_load	4 009,79	758,81	3 913	6 896	2 411	-0,70	0,39
NO3_load	2 764,58	498,26	2 713	4 338	1 583	-0,52	0,31
NO4_load	2 153,34	370,38	2 137	3 330	1 108	-0,83	0,16
NO5_load	1 899,64	389,81	1 882	3 327	829	-0,49	0,30
NO_temp	7,00	6,44	6,39	30,56	-10,44	-0,63	0,14
NO1_temp	7,45	8,28	6,90	34,40	-16,80	-0,62	0,09
NO2_temp	8,74	5,82	8,40	33,70	-10,80	-0,16	0,18
NO3_temp	6,24	7,24	6,00	31,50	-18,70	-0,11	0,06
NO4_temp	3,75	6,70	3,30	29,70	-16,80	-0,29	0,23
NO5_temp	8,80	6,08	8,40	32,30	-11,50	-0,10	0,18

N=61344

4.6. Evaluating forecasts: Error metrics

To evaluate the forecasting performance of a model, error metrics are commonly used on the forecasted period of interest. The primary goal is to measure the forecast errors between the model prediction and the actual value, which then can be averaged for the given period. A simple error metric is the Mean Absolute Error (MAE) that measures the absolute error for each point in time and averages it, returning forecast errors of a similar scale as the level data. If n is the forecasted period of interest, the MAE can be given by:

$$MAE = \sum_{t=1}^{n} \frac{|y_t - \hat{y}_t|}{n} \qquad \textit{Eq. 1}$$

By using absolute values, positive and negative errors are prevented from canceling each other out.

The Mean Squared Error (MSE) is similar to MAE but uses squared errors instead of absolute errors. Due to being squared, the metric penalizes large errors more than small ones, which might be useful fir cases when large forecasting errors are less desirable than small ones.

Applying the root to the MSE gives the Root Mean Squared Error (RMSE) and returns MSE to the same data scale as MAE, but still with a bigger weight penalizing the large errors. MSE and RMSE are given by:

$$MSE = \sum_{t=1}^{n} \frac{(y_t - \hat{y}_t)^2}{n} \quad , \quad RMSE = \sqrt{\sum_{t=1}^{n} \frac{(y_t - \hat{y}_t)^2}{n}} \qquad \textit{Eq. 2 \& 3}$$

The Mean Absolute Percentage Error (MAPE) presents the MAE in percentage form by dividing the absolute error with the actual value. Using the percentage, forecast errors of different scales can be compared, and it gives a simple interpretation. MAPE is usually given by:

$$MAPE = \sum_{t=1}^{n} \left| \frac{y_t - \hat{y}_t}{y_t} \right| \times \frac{100}{n} \qquad \textit{Eq. 4}$$

According to Nti et al (2020), the two most common error metrics are the RMSE (38%) and MAPE (35%), followed by MSE and MAE. The model results of this thesis will primarily be displayed using the MAPE due to its straightforward interpretation and comparability with other models, while the equivalent RMSE results will be displayed in Appendix 3: RMSE results.

To take a look at some previous forecasts by ENTSO-E and Nord Pool for 2019, we find forecasts published by ENTSO-E to have the aggregated NO as well as the individual zones, whereas Nord Pool only has forecast for NO published (see Table 3). For the NO forecasts, Nord Pool has a MAPE of 2,27%, whereas ENTSO-E has a lower MAPE of 0,91%. As for the individual zones, the ENTSO-E forecasts range between 3,05% for NO1 to 8,33% for NO5, giving an average of 5,33% across all the zones. For all the series, 'N/A' observations were removed to provide a clearer picture, as some data gaps were present in in the ENTSO-E data.

Table 3: *Mean Absolute Percentage Error (MAPE) for ENTSO-E and Nord Pool forecasts in 2019.*

MAPE for 2019	NO	NO1	NO2	NO3	NO4	NO5	Average
ENTSO-E	0,91%	3,05%	6,60%	6,00%	7,10%	8,33%	5,33%
Nord Pool	2,27%	-	-	-	-	-	2,27%

Plotting the absolute percentage errors for 2019 in Figure 11, one can see how they compare, with the biggest errors observed in April. Looking at the ENTSO-E errors, the model performs well in the first months of 2019, but with a sharp increase mid-March until May. In the summer months, the errors are low, and increases slightly at the end of august till the end of the year.

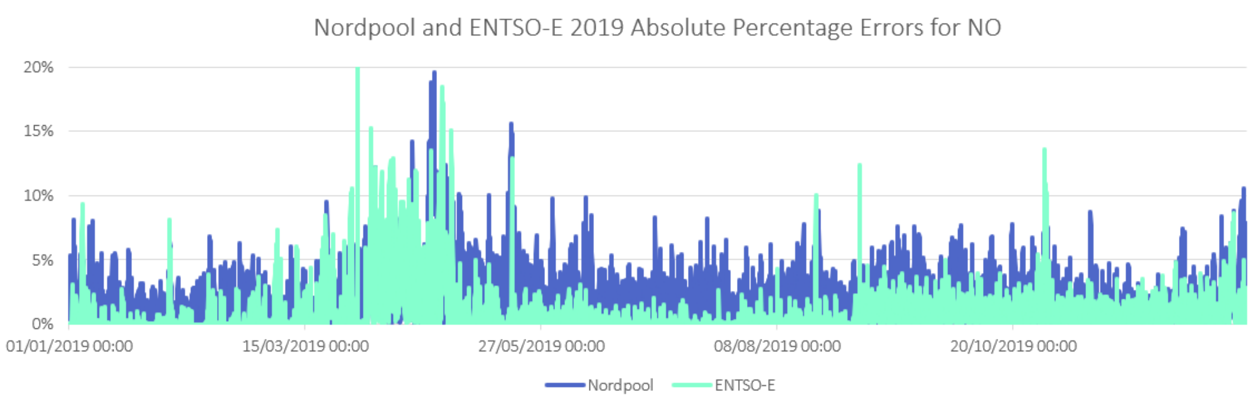

Figure 11: *Absolute percentage error for ENTSO-E and Nordpool in 2019.*

To provide these forecasts in practice, the bids and offers for the day-ahead are submitted by the market participants the day before, between 08:00 to 12:00 according to EnergifaktaNorge (2017, p. 43). As such, to be of use to the market participants' decision making, the day-ahead forecasts would have to be delivered in this timeframe using the earliest data available, being the historical data 24 hours prior and forecast 48 hours ahead. For a delivery at 08:00 for instance, the prediction can be dynamically forecast in two steps as illustrated in Figure 12: one using the available information to forecast the current time, and a second step utilizing the prediction for the current time to forecast the next step, becoming the day-ahead forecast.

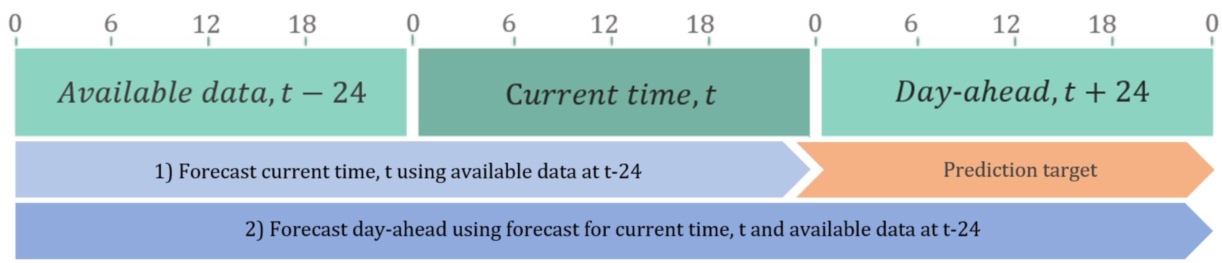

Figure 12: *How day-ahead forecasts are made using data 48 hours prior.*

5. Theoretical Framework

To develop the short-term load forecasting (STLF) models, one model is created using statistical methods within traditional econometrics and another using machine learning to predict the electricity load. This chapter starts by explaining the preliminary approach such as cross-validation and data transformations before expanding on the forecasting methods that has been applied. Simple forecasting methods available are described, such as naïve models, which will serve as a simple baseline against the more complex models. Finally, the concepts of stationarity, Box-Jenkins ARIMA methodology, Dynamic Regression and Neural Networks are outlined.

5.1. Cross-validation

To better evaluate the forecasting models proposed, normal practice is to split the available data into a training and validation set, so that the model can be tested on the 'unseen' validation data. This is to reduce the risk of overfitting the model on the training sample and resulting in a better reference of fit. The data is split into a training set ranging from 01.01.2013 to 31.12.2018, whereas the validation set ranges from 01.01.2019 to 31.12.2019.

When training models, there are several options to utilize the available data in a good manner, such as expanding window, sliding window or k-fold cross-validation. The expanding window approach works by updating the training set with the newest information available as each forecast is computed, illustrated in Figure 13. The expanding window makes it possible to use all the available observations in the data, whereas the sliding window operates without a fixed point of origin to keep the training set at the same size as it updates itself with new information and leaves out older observations. Not using all the data can be advantageous if there is for instance a regime change introducing noise but can also ease computation time for the model training.

K-fold cross-validation divides the data into k equal-sized subsamples called folds, where $k-1$ subsamples are used as the training set, while the last subsample left out is used as the validation set. This process is repeated k times, and then averaged for all the folds. By using k-folds cross-validation one can use all the available data and extract as much information as possible. However, it is more appropriate for small sample sizes where data is harder to come by and requires longer computation times due to the iterative process. Another issue with k-fold cross-validation is applying it on time-series data as it contains sequential information. Using the k-folds allow the validation sets to be tested using models trained on future observations, creating a potential future information-bias when forecasting as it is no longer “unseen” data (Hastie et al., 2009).

For cross-validation, the expanding window approach in Figure 13 was applied as it keeps the time sequence intact while allowing the use of most of the data, being more appropriate for time series models.

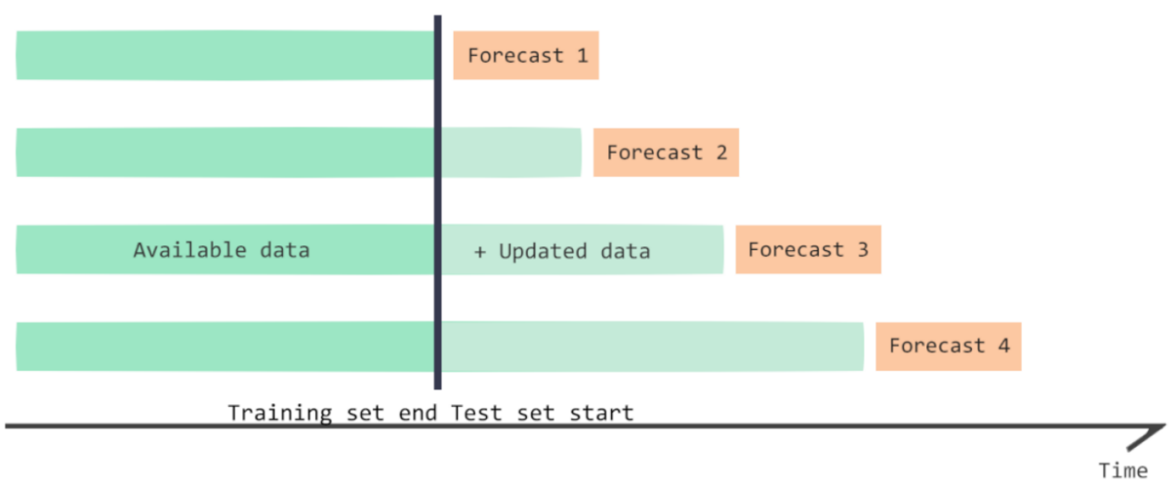

***Figure 13**: Expanding window approach.*

5.2. Data transformations

Data transformations can be used for time series where the variance changes over time, where common transformations are the logarithmic or Box-Cox (1964) transformation. Using the logarithmic transformation for instance can help rescale the data, giving a more constant variance and a more normal distribution. This can help improve the model fit and reduce the likelihood of the residual assumptions breaking. Box-Cox is another popular transformation, which uses a lambda value in the exponent and in the denominator to transform the data into a more normal distribution. It uses a power transformation if $\lambda \neq 0$. Otherwise, a natural log transformation is applied, giving the formula:

$$\text{Box Cox transformation:} \quad y_t(\lambda) = \begin{cases} \dfrac{y_t^{\lambda} - 1}{\lambda} & , \; if \; \lambda \neq 0; \\ \log(y_t) & , \; if \; \lambda = 0. \end{cases} \qquad \textit{Eq. 5}$$

5.3. Simple forecasting methods

While forecasting can be complex, it can also be done using simple methods that are intuitive and powerful. Common techniques are the average-, the exponential smoothing-, and the naïve method. While straightforward, they still provide a valid baseline when compared with more complex models.

In the average model, the future consumption $\hat{y}_t$ is predicted to be the average load of the historical data T, and can be adjusted in terms of window size and point of origin. One adjustment is the simple Moving Average (MA) which uses a window of more recent observations, where $\hat{y}_t$ is the mean of the previous observations k, from the historical data, giving MA of order k. The usefulness of $MA(k)$ lies in the smoothing of the random variance in the observations and can for instance be used to decompose the trend component.

In the Exponential Smoothing model, the prediction depends on an exponentially decreasing weight added to previous observations of the dependent variable, so that the most recent observation weighs the most. The weight parameter, α, can then be used to tune the model. Additionally, trend and seasonality can also be incorporated into the equation (Hyndman & Athanasopoulos, 2018).

A simpler method is the naïve method, predicting future values to be equal to the value found in the last observation, or alternatively the last value a season ago, such as one day, week, or a year prior for a seasonal naïve. The methods are summarized in Table 4, where the formulas can be expressed as:

Table 4: Simple forecasting methods

Average method	$\hat{y}_t = \sum_{t=1}^{T} \frac{y_t}{T}$	Eq. 6
Moving average	$\hat{y}_t = \sum_{t=T-k+1}^{T} \frac{y_t}{k}$	Eq. 7
Naïve method	$\hat{y}_t = y_{t-1}$	Eq. 8
Seasonal naïve	$\hat{y}_t = y_{t-s}$	Eq. 9
Exponential smoothing	$\hat{y}_t = \alpha y_t + \alpha(1-\alpha)y_{t-1} + \alpha(1-\alpha)^2 y_{t-2} + \cdots$, where $0 \leq \alpha \leq 1$	Eq. 10

$s = season, year, month, or\ day$

5.4. Traditional Model Approach

Among the traditional approaches for predicting load is the Autoregressive Integrated Moving Average (ARIMA) model, where the inclusion of exogenous variables and seasonality is referred to as SARIMAX. An ARIMA model with exogenous variables can also be called Linear Regression with ARIMA errors or Dynamic Regression. According to Weron (2014), using the various names interchangeably is often a source of confusion, for which we provide a quick overview in Table 5.

Table 5: *Overview of forecasting model names*

Model name:	Alternative name:	Abbr.
Autoregressive Integrated Moving Average		ARIMA
w/ seasonality		SARIMA
w/ exogenous variables	Linear regression with ARIMA errors	ARIMAX
w/ seasonality and exogenous variables	Linear regression with seasonal ARIMA errors	SARIMAX
Linear regression with ARIMA errors	Dynamic regression	DR
w/ seasonality using Fourier Terms	Dynamic Harmonic regression	DHR

According to Hyndman and Athanasopoulos (2018), the SARIMA model is a powerful forecasting tool for short-term forecasts, but is designed for short seasonal periods such as quarterly or monthly data with an annual frequency of 4 and 12 respectively. Adjusting the model to hourly data, the seasonal period can be set to a yearly frequency of 8760, 168 for weekly, or 24 for daily. Thus, using SARIMA with a daily seasonality seems most appropriate in this case, but renders the model unable to include the weekly and yearly seasonality. One solution is to include the remaining seasonality using dummy variables, or one can use a Dynamic Regression with Fourier terms to handle the seasonality, also known as Dynamic Harmonic Regression (DHR). As proposed by Hyndman, DHR handles the short-term dynamics using ARIMA errors, whereas the seasonality is assumed fixed by the Fourier series.

5.4.1. Stationarity

Stationarity is a requirement when using the ARIMA model , where the expected mean, variance and autocovariance in the time series must not depend on time. The stationarity assumptions for time series can be expressed with the formulas found in Brooks (2019, p. 252) given by:

1. Constant mean: $E(y_t) = \mu$ Eq. 11
2. Constant variance $E(y_t - \mu)(y_t - \mu) = \sigma^2 < \infty$ Eq. 12
3. Constant autocovariance $E(y_{t1} - \mu)(y_{t2} - \mu) = y_{t2-t1} \ \forall \ t_1 , \ t_2$ Eq. 13

For non-stationary data, stationarity can often be obtained by applying a first or second difference to the level series to get an integrated AR process. The first difference is calculated using the change between the current and previous observation ($\Delta y_t = y_t - y_{t-1}$) but can also be seasonally differenced using the previous seasons observation to the current ($\Delta y_t = y_t - y_{t-m}$). Using the seasonal difference is often used when applying a seasonal ARIMA.

Achieving a stationary series through differencing, the time series have a mean-reverting process where no trend, cycle or seasonality will be present. Trends are generally found in time series where the level series is increasing or decreasing in the long-term, for instance in the per capita consumption or the industrial sector and could be present in countries where more and more people are connecting to the electricity grid. Plotting the annual changes, Norway appears to have a slight increase over the period as displayed in Figure 14, where most years show an increase compared to the year prior. This indicates a small positive trend and might be balanced due to increasing consumption along with more efficient energy usage. Seasonality can also be present in a time series, where there is a fixed recurring pattern for certain periods, which is present in the annual, weekly, and daily pattern of the load series. Lastly, cycles are similar to seasonality but without a fixed period, a common example being business cycles.

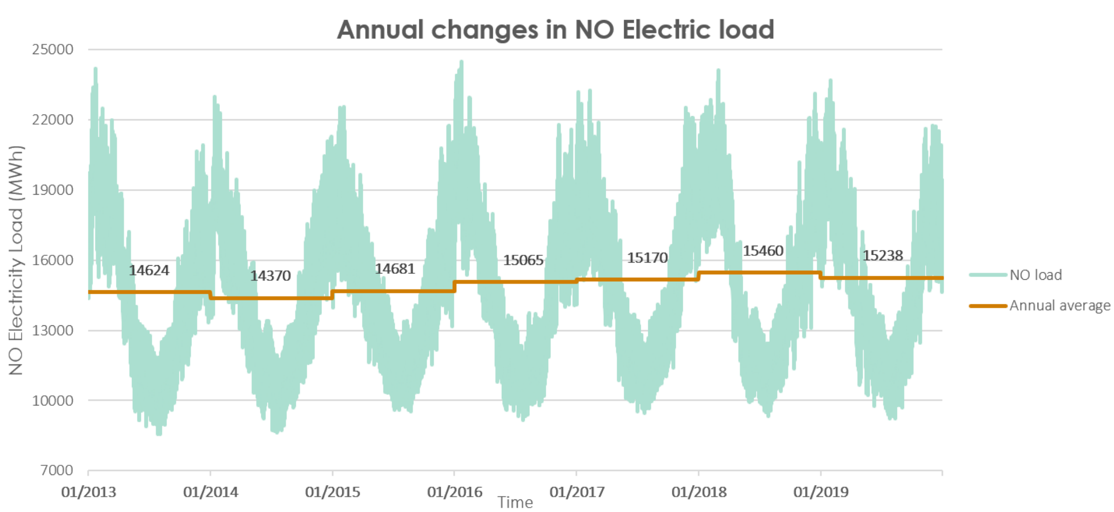

Figure 14: *Annual changes in NO load indicating a slight positive trend.*

5.4.2. Autoregressive Integrated Moving Average

The ARIMA model is generally split into three parts: The Autoregressive process, the integrated component, and the Moving Average process. The Autoregressive process uses the previous values, also referred to as lags, of the dependent variable to forecast, denoted as an $AR(p)$ model depending on the lag length (p). The Moving Average on the other hand uses the previous forecast errors or residuals to forecast the load, denoted $MA(q)$ with the lag order (q). The autoregressive and moving average process can be expressed as:

$$AR(p): \quad y_t = c + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \cdots + \alpha_p y_{t-p} \qquad \text{Eq. 14}$$

$$MA(q): \quad y_t = c + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} \qquad \text{Eq. 15}$$

The $AR(p)$ model uses a multiple linear regression on the dependent variables previous values to obtain the current value. The produced model errors can then be used as inputs for the combined $ARMA(p, q)$ model where the current value depends on its previous values and errors. In the case of non-stationary data, the series can be differenced to obtain an integrated autoregressive process, resulting in the $ARIMA(p, d, q)$ model given by (Brooks, 2019):

$$\mathrm{ARIMA(p, d, q)}: \quad y_t = c + \sum_{i=1}^{p} \alpha_i y_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t \qquad \text{Eq. 16}$$

Extending the model with seasonality results in a seasonal ARIMA, known as SARIMA by adding a seasonal $AR(P)$ term, the seasonal difference (D), and a $MA(Q)$ term. The frequency m denotes which period the seasonality accounts for, for instance 24 for daily seasonality using hourly observations. A seasonal $AR(1)$ term for instance will use the lagged value of the dependent variable one season ago to predict the current value. In the case of electricity load, using y_{t-1}

represents the load one hour ago, whereas y_{t-24} is the load for the same hour the day before, which could a better input in some cases.

In order to select an appropriate ARIMA lag order, a common method is to minimize an information criterion such the Akaike Information Criteria (AIC) or a Bayesian Information Criteria (BIC). The information criteria can be expressed as:

$$\text{AIC}: -2\, log(L) + 2k \qquad \text{Eq. 17}$$

$$\text{BIC}: -\, 2Log(L) + 2k + k[\log(T) - 2] \qquad \text{Eq. 18}$$

$\text{Log}(L)$ is the log-likelihood function, representing the model fit, where a higher value indicates a better fit relative to a lower value. k is the number of parameters estimated, and T is the sample size (Hyndman & Athanasopoulos, 2018). Including more parameters is penalized by $\Delta k = 2$ for the AIC, or stricter in the BIC with $\Delta k = \log(T)$ for sample sizes $T > 100$. Minimizing the information criteria will therefore be a good candidate model in terms of the ARIMA lag order, and also for the variable selection. The next step is to expand the ARIMA model by including exogenous variables in the ARIMAX or Dynamic Regression model.

5.4.3. Dynamic Regression

A Dynamic Regression (DR) is a state-space, or transfer function model based on the general linear regression, where the parameters are able to change independently over time instead of remaining static (Ferreira & Gamerman, 2000). With parameters being able to change due to a state indicator, it is better suited to handle abrupt or continuous changes, for instance as the seasonal indicator changes from winter to summer or when a holiday occurs. A static linear regression model can be expressed as $y_t = F_t\beta + u_t$, where F_t is a $(n \times 1)$ vector of the external inputs and β is a vector of the model parameters. Replacing β with the current state equation β_t results in the regression parameters being able to express change with time, where the current load can be expressed as a function of the current state and the external variables:

$$\text{Current load:} \quad y_t = F_t\beta_t + u_t, \qquad u_t \sim N(0, \sigma^2) \qquad \text{Eq. 19}$$

$$\text{Current state:} \quad \beta_t = G_t\beta_{t-1} + w_t, \qquad w_t \sim N(0, W_t) \qquad \text{Eq. 20}$$

Where G_t and W_t are known $(n \times n)$ matrices called the state transition matrix and the correlation matrix respectively according to West et al. (1985). By using DR, one can allow for the inclusion of autocorrelation in the regression residuals u_t under the condition of having white noise in the ARIMA model errors, ε_t. For instance, a linear regression with $ARIMA(1,0,1)$ errors can express the DR model as:

$$y_t = F_t\beta_t + u_t, \qquad u_t = \alpha_1 u_{t-1} + \theta_1\varepsilon_{t-1} + \varepsilon_t, \qquad \text{Eq. 21}$$

Where the ARIMA errors are given by the regression residuals. For general regression models, the residual assumptions for a Best Linear Unbiased Estimator (BLUE) are given by (Brooks, 2019, p. 91):

1) $E(u_t) = 0$, Zero-mean residuals. Eq. 22
2) $var(u_t) = \sigma^2 < \infty$, Constant and finite variance in residuals. Eq. 23
3) $cov(u_i, u_j) = 0$, No autocorrelation between residuals. Eq. 24
4) $cov(u_t, x_t) = 0$, No correlation between the residual and regressors. Eq. 25
5) $u_t \sim N(0, \sigma^2)$, Normally distributed residuals. Eq. 26

By minimizing the sum of squares error (SSE) for u_t, the normal residual assumptions would likely fail due to autocorrelation, whereas minimizing SSE for ε_t allows for estimation (Hyndman & Athanasopoulos, 2018).

To capture the seasonality, one can use Fourier series to obtain a Dynamic Harmonic Regression (DHR). Including Fourier series as a variable for periodic seasonality works similar to how sounds can be represented as wavelengths of different frequencies and amplitudes. According to the Fourier theorem developed by Joseph Fourier in 1822, a Fourier series can be used to approximate any periodic function using the sum of sine and cosine terms. Fourier series seasonality can be expressed as:

$$\text{Seasonality:} \quad S_t = \sum_{k=1}^{K} \left[\alpha_k \, sin\left(\frac{2\pi kt}{m}\right) + \beta_k \, cos\left(\frac{2\pi kt}{m}\right) \right] \qquad \text{Eq. 27}$$

Using the Fourier terms one can include multiple seasonalities of any length by using different frequencies, where the seasonal periods are approximated by choosing the order of Fourier terms, K, and minimizing an information criterion. A small K will result in a smooth seasonal pattern, whereas larger values will give a more complex pattern (Hyndman, 2010).

5.5. Machine Learning

While computers can be very good at solving mathematical problems when used correctly, they are not anywhere near the human brain when it comes to spontaneous pattern recognition and image identification. The field of Artificial Intelligence (AI) is trying to create algorithms able to imitate human skills. When predicting electricity load, the aspiration is to combine the computational power of computers with the ability to see patterns in a set of features, to improve the automatically generated prediction of load.

Machine learning is a subgroup of AI models that adapt their internal structure to a set of data used for training to predict the value of an output variable outside of the training data. This is done without assumptions about the input variable parameters. The training can be supervised or unsupervised, where 'supervised' refers to the training data consisting of examples where the solution is known.

5.5.1. Artificial Neural Networks

Artificial Neural Networks (ANN) is a type of machine learning inspired by the structures of neurons and their connections in the brain. The concept revolves around the imitation of the neurons in the human brain, where the touch of a warm object will send an electric signal from the hand to the neurons which processes the signal through one another before reaching the conclusion that it is warm. The neurons in an ANN algorithm are referred to as nodes, being structured in an input layer consisting of the exogenous variables, one or more hidden layers processing the inputs, and an output layer where the results are received (see Figure 15).

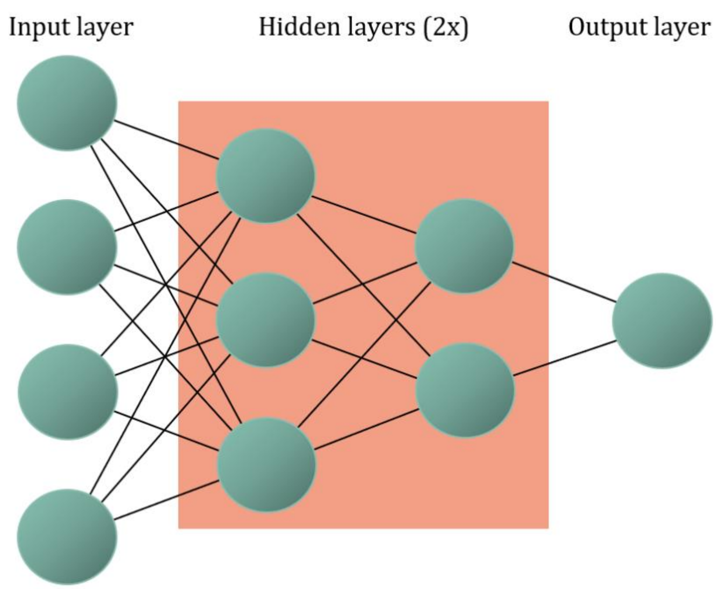

Figure 15*: Neural network architecture using two hidden layers, called Multi-Layer Perceptron.*

In ANN algorithms, each input variable is viewed as an individual input node acting as the neuron. The input layer does not process the data but sends a weighted sum of the input directly to the first hidden layer. The last layer is called the output layer, which in most cases has one node. Between the input and output layer there can be one or more hidden layers, where each layer contains a number of nodes. These nodes adjust themselves to the training data so that the connection between each node is given a weight and a bias value, which when put into an activation function decides the importance of that node in the first layer to the connected node in the next layer. It is the ability to change the weights in the neurons through each epoch of its training stage, called backpropagation, that makes us say it is mimicking the long-term memory seen in brains.

In neural networks, different activation functions can be used. Within each node is a number derived from an activation function. It is also possible to build networks using different activation functions in each layer. The activation function is important as it returns whether the inputted data in the node is activated for any positive values or be output as a zero for negative ones as given by Eq. 28. In the 1990's, the default activation functions were the Sigmoid (logistic) and Hyperbolic Tangent (tanh), but today the modern default activation function used for deep neural networks with multiple hidden layers is the Rectified Linear Unit Function (ReLu) as shown in Figure 16:

$$f(x) = \max(0, x) \qquad \text{Eq. 28}$$

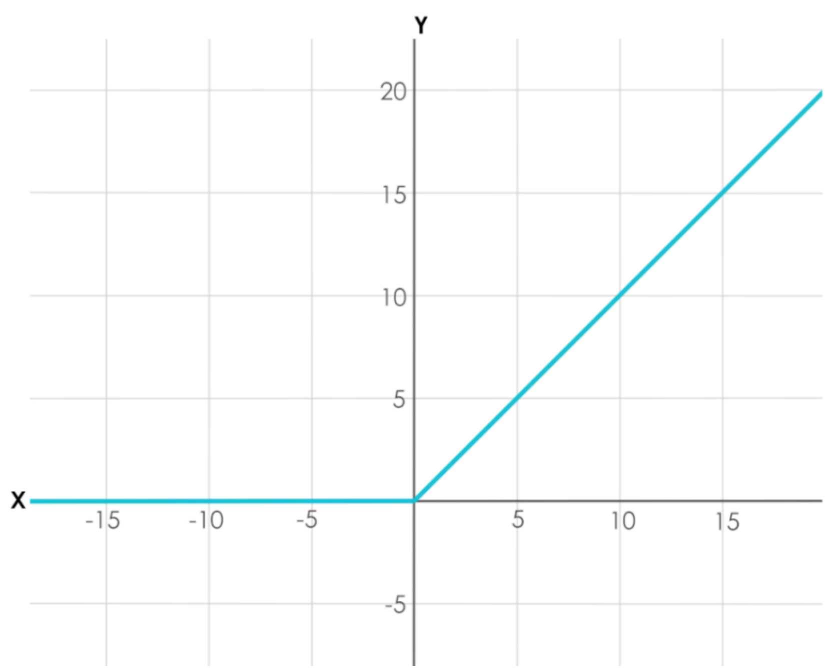

Figure 16: *Plot of the ReLu-function for different values of x.*

5.5.2. Different types of neural networks

There are three main groups of neural networks. ANN is mainly used for solving classification and regression problems. Convolutional neural networks (CNN) are used mainly for computer vision problems. Recurrent neural networks (RNN) are often used for time series analysis problems. Yet there are no fixed rules that one type of neural network cannot be used for a different type of problem. All of them can be used for supervised machine learning problems. A deep neural network (DNN) is any neural network with multiple layers between the input and output layers. Being able to process the data using multiple nodes and weights, DNN's work well for modelling complex non-linear relationships.

RNN feeds information back to the input to help predicting the outcome of the layer. The first layer is normally a feed-forward neural network, followed by a recurrent neural network where some information it had in the previous time-step is remembered by a memory function, storing only information that is required for future use. If the prediction is wrong, the learning rate is used to make small changes. Long-Short-Term Memory networks (LSTM) is an improved type of RNN, including a memory cell that can keep information for long periods with gates that control when information enters the memory.

Multi-Layer Perceptron (MLP) is a type of ANN. The terminology is used ambiguously, and many refer to MLP as simply ANN. This is a fully connected feed-forward neural network, with bi-directional propagation, that is forward propagation where inputs are multiplied with weights and fed into the activation function, and backward propagation where the weights are adjusted to fit the value of the dependent variable in the training data. It is used for deep learning; due to its dense fully connected layers and non-linear activation it can differentiate data that are not linearly separable.

CNN has a three-dimensional arrangement of nodes, instead of the standard two-dimension. The first layer is called convolutional layer. Each neuron in this layer only processes information from a small part of the visual field. Variables are taken in batches, like a filter.

Neural Network Autoregression(NNAR) is a hybrid model which combines the non-linear functions and hidden layer of a neural network, and the use of lagged values and application of seasonality from the ARIMA-model. Contrary to ARIMA-models, the non-linearity of the model can be more accurate when cycles are non-symmetric, and there are no restrictions on parameters to ensure stationarity. Adding autoregression to the neural network adds information to the model that the data is time series, as well as the frequency of observations.

6. Methodological Approach

This chapter explains the approach used for the four different load forecasting models employed on the Norwegian power market. Seasonal naïve, Dynamic Harmonic Regression (DHR), Neural Network Autoregression (NNAR), and Multi-Layer Perceptron (MLP).

6.1. Baseline model: Seasonal naïve

The baseline model is a seasonal naïve, using the electric load 48 hours prior to predict load. The 48-hour lag is used in favor of the 168-hour lag, as it resulted in a lower forecasting error overall for the sample data. However, the 168-hour lag model might perform better at other aspects. The 48-hour lag model is simple but intuitive as it should capture the hourly dynamics well but will be unable to capture the effect of the separate weekdays and holidays. The yearly seasonality should also be captured fairly well, due to the lag being only 2 days. The model can be expressed as:

$$Seasonal\ naive: \quad \hat{y}_t = y_{t-48} \qquad \text{Eq. 29}$$

6.2. Dynamic Harmonic Regression with ARIMA errors

Two options were proposed, modelling the electric load using one model for all hours, or split each hour into separate models, where the latter is chosen. For the modelling procedure, the Box and Jenkins (1970) ARIMA approach was used to construct the Dynamic Harmonic Regression (DHR) model where 3 steps are used: 1) Identification, 2) Estimation, and 3) Diagnostic checking. The model construction was conducted using the R 'forecast' package by Hyndman et al. (2020) and based on the daily NO load data at 09:00 as a generalization for the other models.

Model identification

Before considering the ARIMA lag order of the model, the possibility of reducing the variance using a transformation is investigated, and a stationarity test is conducted. To check the need for a transformation, the '`BoxCox.lambda`' function in R selects the lambda coefficient that minimizes the variance and returns the lambda of -0.7599 for the NO series. When forecasting using the Box-Cox function in R, it will normally return the median point forecasts whereas we want to use the mean point forecasts. To avoid this, the '`biasadj=TRUE`' argument is used to get forecasts using the mean. The Box-Cox transformed NO series is plotted in Figure 17, where the transformed series does not appear to be very different compared to the original data from Figure 3.

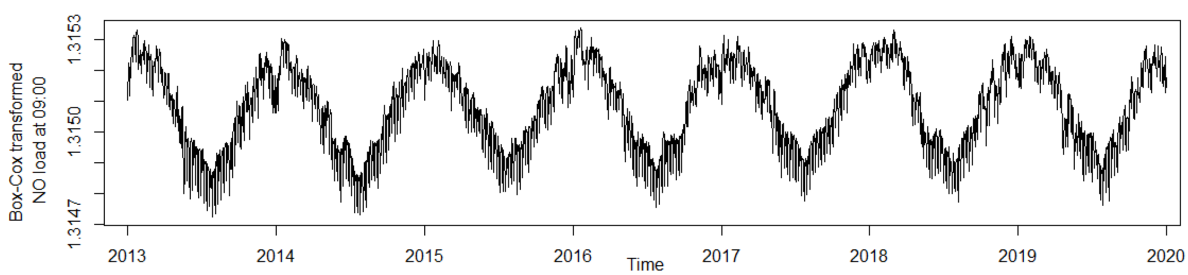

Figure 17: *Box-Cox transformed 09:00 electric load series for NO*

Concerning the stationarity assumption, the load series may be non-stationary due to the seasonal patterns, which could be argued to also have a mean-reverting effect. According to Hyndman and Athanasopoulos (2018), a DR model requires the dependent variable and the exogenous variables to be stationary to obtain consistent estimates. The first and seasonal difference can be applied to see if it looks more stationary. Figure 18 shows the first differenced load while Figure 19 shows the weekly seasonal difference ($m = 7$), displaying that both series are more centered around the zero-mean, but the with changes in variance when switching between summer and winter.

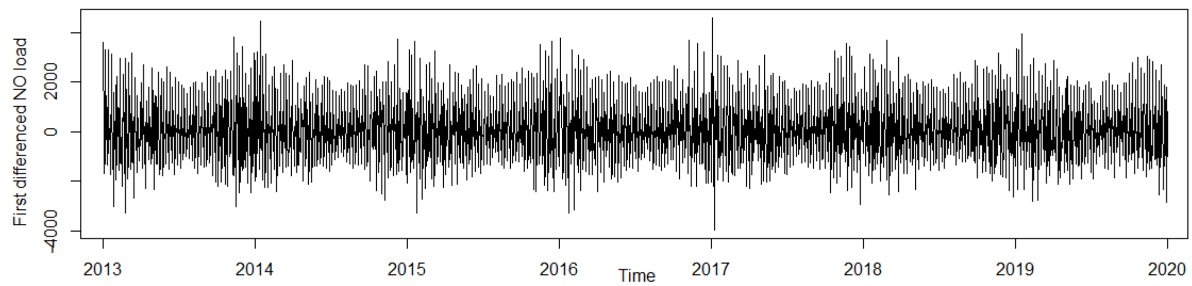

Figure 18: *First differenced NO daily load at 09:00*

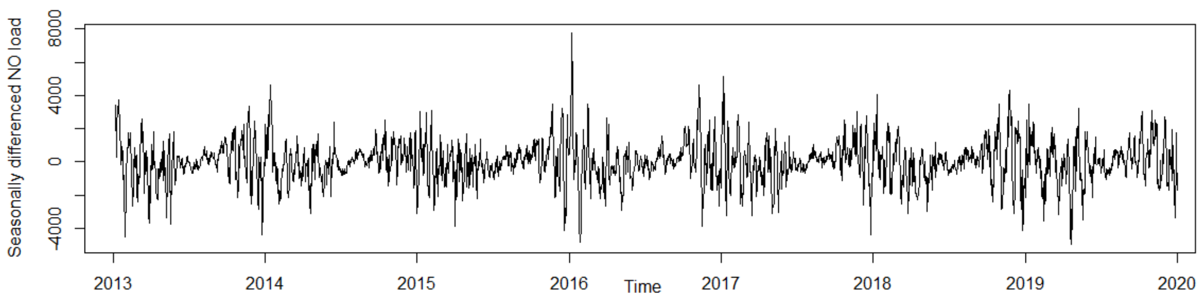

Figure 19*: Seasonally differenced* $(m = 7)$ *NO daily load at 09:00.*

Formally, a unit root test can be employed to check for stationarity using the Augmented Dickey-Fuller (ADF) test and the Kwiatkowski, Phillips, Schmidt, and Shin (KPSS) test (Kwiatkowski et al., 1992). The ADF tests the null hypothesis that the time series is non-stationary against the alternative of having a stationary series. The ADF returns a 5% critical value of -2.87, where a lower test statistic is required to reject the null. The KPSS test on the other hand tests the null hypothesis that the time-series is trend-stationary, with the alternative being a non-stationary series. Trend-stationary means that the time series can be stationary around the deterministic trend, where decomposing and removing the trend from the data would result in a stationary series instead of having to apply a difference. The KPSS 5% critical value is 0.463, where a higher test statistic is required to reject the null of stationarity, and the same lag length as for the ADF test was used. Due to the tests being slightly different, both were applied for the formal testing.

The optimal lag length for the ADF and KPSS test is selected using the EViews Unit Root Test, automatically selecting the number of lags according to an information criterion. Doing so, the maximum number of lags EViews can use is specified according to the formula by Schwert (1989) in Eq. 30, where the length of the time series for the individual hours, $T = \frac{61344}{24\ hours} = 2556$ gives a maximum lag length of 27.

$$p_{max} = \left[12 \times \left(\frac{T}{100}\right)^{\frac{1}{4}}\right] \qquad \text{Eq. 30}$$

Selecting the lag length based on the AIC, the results of the unit root tests is given in Table 6 for the daily NO load series, revealing a stationary series for the level data at the 5% significance level using the ADF, and at the 1% level using the KPSS test.

Table 6: *ADF test and KPSS test, critical value for 1% significance.*

Test	Level	First diff.	Seasonal diff.
ADF	-3,05*	-8,65**	-8,72**
KPSS	0,29**	0,01**	0,04**

*Significant at the 5% level
**Significant at the 1% level

As the load series are stationary at level, the ARIMA lag order can then be determined by inspecting the Autocorrelation- and partial autocorrelation functions (ACF and PACF) to look for the appropriate autoregressive or moving average process. Alternatively, the lag order can be chosen using automated ARIMA algorithms such as ‘`auto.arima`’ in R. Normally, the ACF and PACF are plotted using the stationary time series, however, in a DR model the regression residuals must be obtained first before determining the lag order for the ARIMA errors.

As a starting point, a dynamic regression with $ARIMA(0,0,0)$ errors for load were run using just temperature, obtaining a regression and ARIMA error residuals as shown in Figure 20:

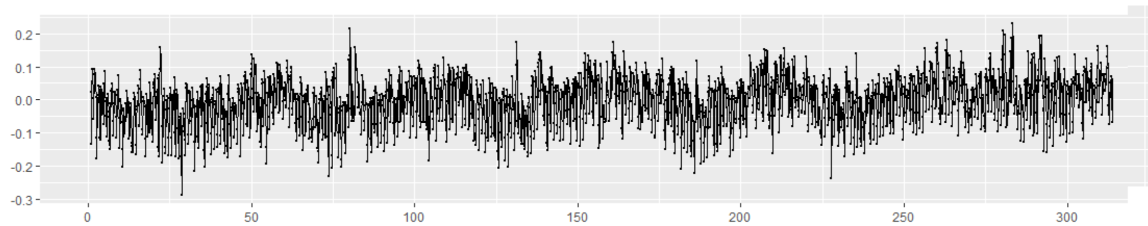

Figure 20: *Dynamic Regression residuals for daily NO load at 09:00.*

The ACF measures the correlation of y_t for previous values y_{t-k}, where a slowly decaying series indicates that several values correlate with each other over time. On the other hand, the PACF measures the correlation of y_t for previous values of y_{t-k} but without factoring in the correlation of lags smaller than k on y_t. The ACF plot was used to identify non-seasonal and seasonal MA terms by the significant autocorrelations shown, whereas the PACF was used for identifying the non-seasonal and seasonal AR processes. The residuals are plotted in Figure 21, where the dashed blue line indicates whether the autocorrelated lag is significant or not.

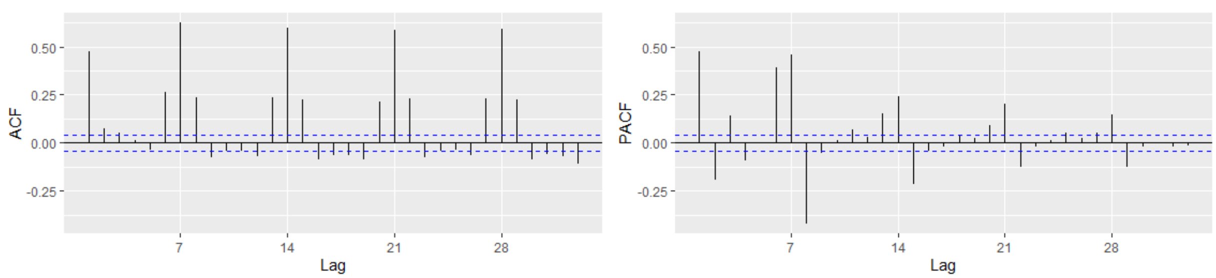

Figure 21*: ACF- and PACF plots of the Dynamic Regression residuals.*

The ACF plot shows a significant autocorrelation in lag 1 to 3 for the non-seasonal terms, and then significance for the seasonal lags at the weekly interval along with the subsequent lags. This might suggest a non-seasonal $MA(3)$ with a seasonal $MA(Q)$ component. For the PACF, it starts out with 4 significant lags in addition to the seasonal lags, suggesting a non-seasonal $AR(4)$ and a seasonal $AR(P)$.

Additionally, we add Fourier terms to the model, where the terms are iterated using the '`fourier`' function for different values until the AIC is no longer reduced. Here, the weekly seasonality is captured by $k = 3$, and the yearly seasonality with $k = 5$. Using an additional argument, the Fourier series generated using the training data is extended a year ahead to serve as the input for the test data.

Estimation and diagnostic checking

Based on Figure 21, an $ARIMA(4,0,3)(1,0,1)_7$ model can be tested, which will be referred to as Model 1. The resulting output for the model is displayed in Table 7, with the in-sample residuals, ACF-plot and error distribution displayed in Figure 22.

Table 7: *Model 1 – ARIMA(4,0,3)(1,0,1)₇ estimation output*

Model 1

Regression with ARIMA(4,0,3)(1,0,1)[7] errors	Log(L)	4943,47	MAPE	2,0443
Dependent variable: NOload at 09:00	Sigma^2	0,00065	MPE	-0,0574
Box-Cox transformation: lambda = -0,01139128	AIC	-9830,93	RMSE	459,46
	AICc	-9830,18	MAE	326,09
	BIC	-9671,55	MASE	0,3617

Variables			**Yearly Fourier series: k=5**			**Weekly Fourier series: k=3**		
	Coefficient	s.e.		Coefficient	s.e.		Coefficient	s.e.
Intercept	9,2143	0,0047	S1-365	0,0509	0,0039	S1-7	0,0515	0,0016
AR1	0,8761	0,1662	C1-365	0,1471	0,0045	C1-7	-0,0186	0,0016
AR2	-0,5952	0,2033	S2-365	-0,0009	0,0033	S2-7	0,0268	0,0013
AR3	0,8964	0,1944	C2-365	-0,0072	0,0032	C2-7	0,0303	0,0013
AR4	-0,3449	0,0917	S3-365	0,0009	0,0031	S3-7	-0,0042	0,0008
MA1	-0,2430	0,1661	C3-365	-0,0061	0,0031	C3-7	0,0174	0,0008
MA2	0,4511	0,1660	S4-365	-0,0031	0,0030			
MA3	-0,5055	0,1289	C4-365	-0,0053	0,0029			
SAR1	0,9484	0,0597	S5-365	0,0032	0,0028			
SMA1	-0,9047	0,0864	C5-365	-0,0080	0,0028			
Temp	-0,0102	0,0003						

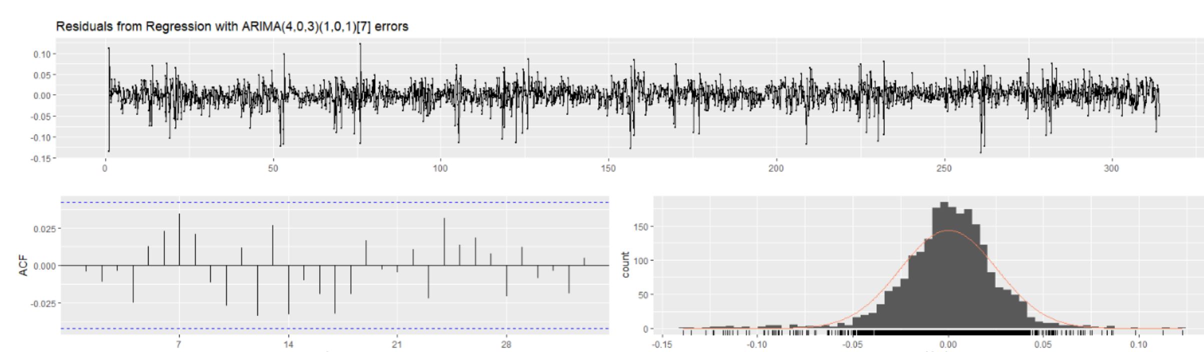

Figure 22: *Model 1 – Residual-, ACF- and distribution plot*

The ARIMA errors follow a white noise process, where any significant lags shown in the ACF plot would indicate that the model fails to account for the autocorrelation present. Looking at Figure 22, all the lags shown exhibit no significant autocorrelation, meaning the residuals contain less information compared to Figure 21. To formally test the autocorrelation of the ARIMA residuals, the Box-Pierce- and Ljung-Box test statistic were employed. Box-Pierce tests the null hypothesis that the residuals are uncorrelated over time, with the alternative being serial correlation residuals. Ljung-Box on the other hand, is a slight modification which handles small sample sizes better. While the sample collected is not small, both tests are conducted and can be given as (Brooks, 2019, p. 254):

$$\text{Box-Pierce: } Q^* = T\sum_{k=1}^{h}\hat{\tau}_k^2, \quad \text{Ljung-Box: } Q^* = T(T+2)\sum_{k=1}^{h}\frac{\hat{\tau}_k^2}{T-k} \qquad \text{Eq. 31 \& 32}$$

Where T is the sample size, h is the lag, and $\hat{\tau}_k^2$ is the squared residuals at lag k. The null hypothesis of having no serial correlation can be rejected if $Q^* > \chi^2_{1-\alpha,h}$, or if the p-value is less than 0,05. A rule of thumb used by Hyndman and Athanasopoulos (2018) is to use lag $h = 10$ for non-seasonal data, and $h = 2m$ for seasonal ones as it allows to capture at least two seasonal periods.

Table 8: *Autocorrelation tests for model 1*

Autocorrelation test		$(h = 14)$
$ARIMA(4,0,3)(1,0,1)_7$	Test statistic	P-value
Ljung-Box	15,32	0,36
Box-Pierce	15,24	0,36

Running the tests in Table 8, the null hypothesis of no autocorrelation could not be rejected at any significance level, meaning the ARIMA specification for Model 1 exhibit residuals resembling a white noise process.

To see if there is a better lag specification, the '`auto.arima`' function by Hyndman et al. (2020) was also applied, it iterates through the possible ARIMA specifications and returns the best model according to the AIC. An $ARIMA(3,1,3)(1,0,0)_7$ specification was returned, giving both a lower AIC and MAPE compared to the first model (result not shown). Additionally, to make the model more complex, more exogenous variables were added such as the various calendar effects and lags of temperature and load. Using the AIC as information criteria for the model fit, the variables which lowered the AIC was incorporated. The model includes daily load lags from 48 hours to 168 hours, and daily temperature lags from 24 to 96 hours, in addition to 1 hour temperature lag to capture the lagged effect on consumption. The daily minimum temperature and HDD were also included, whereas maximum temperature and CDD variables were found to lower the AIC. Lastly, the public holiday dummy was split into separate dummies for Christmas-, and easter holidays, whereas the remaining holidays were constructed as a separate dummy. The resulting output from this DHR, referred to as Model 2 with additional exogenous variables is shown in Table 9, with residuals displayed in Figure 23.

Table 9: *Model 2 – ARIMA(3,1,3)(1,0,0)$_7$ estimation output*

Model 2				
Regression with ARIMA(3,1,3)(1,0,0)[7] errors	Log(L)	5519,59	MAPE	1,6338
Dependent variable: NOload at 09:00	Sigma^2	0,00038	MPE	0,0322
Box-Cox transformation: lambda = -0,01139128	AIC	-10957,38	RMSE	343,02
	AICc	-10955,77	MAE	257,98
	BIC	-10724,15	MASE	0,2861

Variables

	Coefficient	s.e.		Coefficient	s.e.		Coefficient	s.e.
Intercept			Mintemp	-0,0018	0,0005	C2-365	-0,0038	0,0015
AR1	0,1114	0,6407	Holiday	-0,0587	0,0030	S3-365	0,0020	0,0014
AR2	0,3751	0,1990	Easter	-0,0678	0,0055	C3-365	0,0022	0,0014
AR3	-0,0835	0,1485	Christmas	-0,0788	0,0048	S4-365	-0,0047	0,0013
MA1	-0,6930	0,6385	Load_48	0	NaN	C4-365	-0,0023	0,0013
MA2	-0,4958	0,4636	Load_72	0	NaN	S5-365	0,0025	0,0013
MA3	0,2001	0,2884	Load_96	0	NaN	C5-365	-0,0029	0,0013
SAR1	0,2125	0,0206	Load_120	0	NaN			
Temp	0,0040	0,0012	Load_144	0	NaN	**Weekly Fourier series: k=3**		
Temp_1	-0,0037	0,0012	Load_168	0	NaN	S1-7	0,0581	0,0013
Temp_24	-0,0044	0,0003				C1-7	-0,0182	0,0015
Temp_48	-0,0009	0,0003	**Yearly Fourier series: k=5**			S2-7	0,0275	0,0011
Temp_72	0,0002	0,0003	S1-365	0,0216	0,0020	C2-7	0,0289	0,0009
Temp_96	-0,0001	0,0003	C1-365	0,0771	0,0037	S3-7	-0,0038	0,0009
HDD	0,0087	0,0007	S2-365	-0,0040	0,0014	C3-7	0,0177	0,0007

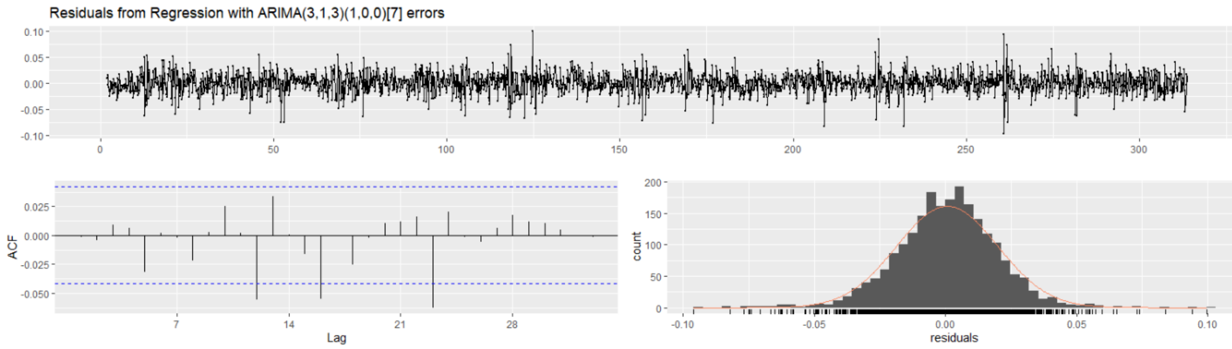

Figure 23: *Model 2 – Residual-, ACF- and distribution plot*

In Model 2, the AIC has reduced to -10 957 from -9 831, with an in-training MAPE of 1,63%. The autocorrelation tests result in a fail to reject the null hypothesis with the p-value 0,42 for the Ljung-Box test (see Table 10), and the ACF plot shows low to no correlation between the lags up to lag 11. However, some lags do become significant past 12, but should not compromise the model performance.

Table 10: *Autocorrelation tests for model 2*

Autocorrelation test		$(h = 14)$
$ARIMA(3,1,3)(1,0,0)_7$	Test statistic	P-value
Ljung-Box	14,34	0,42
Box-Pierce	14,26	0,43

Using the '`forecast`' function on the resulting model along with a test set of the exogenous variables, a dynamic forecast for the whole of the year 2019 was created. The results are plotted together with the actual load in Figure 24. For the predictions presented later in the results, they will be made using the expanding window approach to update the data for each conducted hour forecast, and will be done in two steps as mentioned in section 4.6 (see Figure 12): One forecasting the current day using 24-hour old data, and a second using the current day forecast in addition to the available data to forecast the day-ahead. This approach is used for all the models with the seasonal naïve as the exception, where the separate model results for each hour is later joined together to create the aggregated hour forecast of 2019.

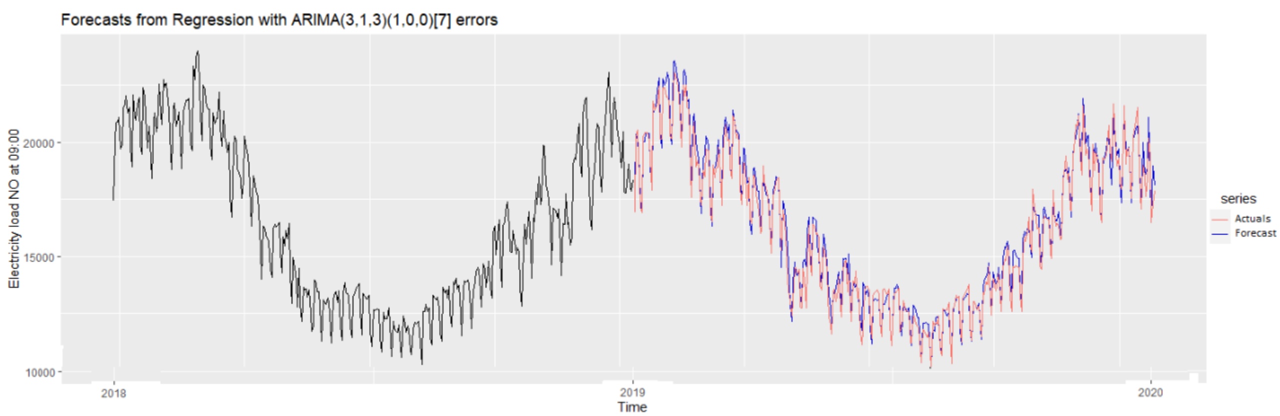

Figure 24: *Dynamic forecast of 2019 for 09:00, fit versus actual consumption.*

6.3 Multilayer Perceptron Regression Neural Network

The '`MLPRegressor`' from 'Sklearn' was used to create a supervised deep learning model for load forecasting. The first step was importing the data into Python and declaring the datatype for each column. Input must be numerical, so categorical variables were transformed into separate Boolean columns.

The data were first split into a training set from 2013 to 2018 as described in 5.1, along with the test set for 2019, but later the training period was reduced to 01.01.2017 to 31.12.2018 as this improved the final results. Features used in the models were selected using a combination of previous research, brute force, and a function for Recursive Feature Selection (RFE) from Sklearn. RFE is a wrapper-type feature selection algorithm, it can be fed with a machine-learning algorithm, and the best features for that algorithm are selected. Unfortunately, it does not support the '`MLPRegressor`' algorithm. Another drawback is that it cannot find the optimal number of features. It is fed with all variables, and the number of features must be set manually, then the algorithm selects this number of features with the most explanatory power to the dependent variable. In this thesis, the RFE function were run with both a linear regression and decision tree as estimators, and some experimenting was done using the training data to find the best features. A set of features was selected and used across all hours and zones for both the MLP and NNAR forecasting models as shown in Table 11.

***Table 11**: Inputs used for the Multi-Layer Perceptron and Neural Network Autoregression*

$\boldsymbol{X_{load,\ t-48}}$	Load lagged 48 hours
$\boldsymbol{X_{MaximumDayLoad,\ t-48}}$	The highest hourly load the last 48 hours
$\boldsymbol{X_{MinimumDayLoad,\ t-48}}$	The lowest hourly load the last 48 hours
$\boldsymbol{X_{Temperature,\ t}}$	Weather temperature at time t
$\boldsymbol{X_{TemperatureDailyAverage,\ t}}$	Daily average temperature
$\boldsymbol{X_{DegreeDay,\ t}}$	Binary variable $\begin{cases} 1, if\ temp < 9°\text{C for month 1-6 or 11°C for month 7-11} \\ 0, if\ temp > 9°\text{C for month 1-6 or 11°C for month 7-11} \end{cases}$
$\boldsymbol{X_{MinimumDailyTemperature,\ t}}$	Daily minimum temperature
$\boldsymbol{X_{HeatingDegrees,\ t}}$	Number of degrees below 15,5°C
$\boldsymbol{X_{WeekdayDummy,\ t} \times 7}$	One Boolean variable per weekday
$\boldsymbol{X_{MonthDummy,\ t} \times 12}$	One Boolean variable per month
$\boldsymbol{X_{PublicHolidays,\ t}}$	Binary variable $\begin{cases} 1, if\ holiday \\ 0, if\ non-holiday \end{cases}$

A total of 28 input variables was used in the models for zones NO1-NO5. However, in zone NO a total of 33 variables was employed, as adding both temperature variables for zone NO1 and an average of temperature for all zones yielded better results.

Most neural networks will achieve a higher precision if the data, both input variables and target variables are normalized, so that they have a value ranging between 0 and 1. If the distribution of the quantity is normal, the data should be standardized, otherwise the data should be normalized. The data preprocessing must be done separately on the training data and the test data, so that no information leaks from the test data into the training data. To do this the 'StandardScaler' is used for standardization or the 'MinMaxScaler' for normalization, both functions from Sklearn. The process is done separately on the training data and test data, and the prediction is then inverted using the 'fit_transform' when the computation is done. Scaling was tested, but did not result in any improvement of result, and was not included in the ultimate model.

To optimize the neural network parameters, the '`GridSearchCV`' function from 'Sklearn' was run. This is an exhaustive search over specified values of the parameters. It tests all the different combinations of the parameters set and find which combination that fits the training data best. For MLPRegressor these parameters are the number of hidden layers, and their sizes, activation function, solver, alpha, learning rate and random state. The combinations searched are displayed in Appendix 2: Grid search for Multi-Layer Perceptron model, where the following combination in Table 12 yielded the best fit:

Table 12: *Grid search for best MLP parameter settings.*

Parameter	best fit
Hidden layer size	(28,14,7)
Activation function	relu
Solver	adam
Alpha	0,0001
Learning rate	constant
Random state	60
Max iteration	1000
Warm start	FALSE

There are two possibilities for training, training one model on the training data, and running the model on the entire test set. This a very fast process. The other option is an expanding window, training a new model for each day, so that the training data include information that is only 48 hours old. This yielded better results and was included in the MPL model, but the difference was marginal, only approximately 0.5 %.

The model was tested both on hourly observations, and on daily observations. When each hour of the day was modelled separately, using 24 individual models for each zone, results improved with approximately 3%. This was included in the final models.

6.4 Autoregressive Neural Network

The '`nnetar`' function in R 'forecast' package was used for the Autoregressive Neural Network (NNAR). The data was read into R and declared as a time series along with the frequency of observations before being split into the training and test set.

'`nnetar`' fits a feed-forward $NNAR(p, P, K)_m$ model. Where p denotes the number of lagged inputs from the last available observations. P is the number of lags from the same season, defined by the frequency of the datapoints m. And k is the number of nodes in the hidden layer. Using the default settings for the parameters, p is set to the optimal number of lags according to AIC for a linear $AR(p)$ model. P is set to 1, and $K = \frac{p+P+1}{2}$. This resulted in a $NNAR(28,1,22)_7$ model. Since 24 models were created and individually optimized by the NNAR, the parameters may vary for each model. Additionally, the model uses the Box-Cox transformation and scales the inputs afterwards, which makes it run more efficiently.

The variables were the same as those used for the MLP-model, as mentioned in Table 11. Fourier transformations, as used in the DHR-model, was also applied, but did not improve results and was excluded from the final NNAR model.

7. Results

In this section, the results from the Seasonal Naïve, Dynamic Harmonic Regression (DHR), Neural Network Autoregression (NNAR), and Multi-Layer Perceptron (MLP) are presented in Table 7 for the Mean Absolute Percentage Error (MAPE). For all resulting tables, an equivalent table using the Root Mean Squared Error (RMSE) penalizing larger errors is displayed in Appendix 3: RMSE results.

Table 13: *Mean Absolute Percentage Error (MAPE) for the models across the bidding zones.*

Model MAPE	NO	NO1	NO2	NO3	NO4	NO5	Average
Baseline: Seasonal naïve	5,83%	9,57%	5,78%	5,19%	5,38%	6,64%	6,40%
Dynamic Harmonic Regression	1,84%	2,99%	2,62%	2,72%	2,65%	3,53%	2,73%
Neural Network Autoregression	1,91%	3,39%	3,12%	3,88%	3,34%	4,66%	3,38%
Multi-layer Perceptron Regressor	12,31%	18,59%	10,43%	9,51%	9,35%	11,71%	11,98%
Absolute change from baseline							
DHR	3,99%	6,58%	3,16%	2,47%	2,73%	3,11%	3,67%
NNAR	3,92%	6,18%	2,66%	1,31%	2,04%	1,98%	3,02%
MLPRegressor	-6,48%	-9,02%	-4,65%	-4,32%	-3,97%	-5,07%	-5,59%
Relative change from baseline							
DHR	68,4%	68,8%	54,6%	47,5%	50,7%	46,9%	56,1%
NNAR	67,2%	64,6%	46,0%	25,2%	37,9%	29,8%	45,1%
MLPRegressor	-111,1%	-94,3%	-80,4%	-83,2%	-73,8%	-76,4%	-86,5%

The seasonal naïve model delivers forecasts with an average MAPE of 6,40% across the Norwegian zones, providing good results despite its simplicity. The model provides the lowest forecasting errors for NO3 with a MAPE of 5,19%, whereas NO1 deviates compared to the rest with a much higher error of 9,57% MAPE.

The DHR model provides an average MAPE of 2,73% across the zones, being lower by 3,67% compared to the baseline with a relative model improvement of 56,1%. The NNAR model on the other hand has an average MAPE of 3,38%, performing worse by 0,65% compared to the DHR model and better by 3,02% against the baseline, being equivalent to a 47,12% relative improvement. For the individual zones NO and NO1 has the biggest model improvements of 68,43% and 68,77% relative to the baseline, but the NO1's improvement can be attributed to the seasonal naïve predicting much worse compared to the other zones. If NO1 is considered as a deviant, it seems model predictions for larger aggregates perform better compared to lower aggregates such as NO5 with the highest MAPE of the zones of 3,54% for the DHR model.

Lastly, while they may not be directly comparable to our knowledge, another comparison is against the forecasts published by ENTSO-E and Nord Pool from Table 3. Here, ENTSO-E average MAPE of 5,33% is beaten by both the DHR and NNAR model but fails to beat the MAPE for the aggregated NO, being 0,91%. The DHR and NNAR also beat the Nord Pool prognosis of 2,27% for NO.

7.1. Model error breakdown

In order to discover more about the model accuracy, the errors for each hour, weekday, and month can be used to pinpoint any flaws. Table 14 to Table 19 displays the MAPE for the aggregated NO and the individual zones, where the bold values highlight the model with the best accuracy.

Table 14: *Breakdown of Mean Absolute Percentage Error (MAPE) for NO.*

	S_naive	DHR	NNAR	MLP
NO	5,83%	**1,84%**	1,91%	12,31%

Weekday	S_naive	DHR	NNAR	MLP
Mon	6,86%	2,11%	**1,91%**	11,79%
Tue	7,71%	1,80%	**1,79%**	12,20%
Wed	3,81%	1,89%	**1,88%**	12,82%
Thu	3,78%	1,85%	**1,76%**	11,75%
Fri	4,11%	**1,88%**	2,13%	11,27%
Sat	7,09%	**1,73%**	1,94%	12,69%
Sun	7,40%	**1,65%**	1,96%	13,66%

Month	S_naive	DHR	NNAR	MLP
Jan	5,56%	**1,58%**	1,78%	9,67%
Feb	6,07%	1,81%	**1,80%**	11,87%
Mar	5,25%	**1,56%**	1,65%	10,12%
Apr	6,47%	2,51%	**2,50%**	9,77%
May	6,98%	**2,23%**	2,68%	12,77%
Jun	5,95%	**2,33%**	2,35%	13,60%
Jul	4,86%	1,70%	**1,69%**	14,90%
Aug	5,77%	**1,46%**	1,57%	18,78%
Sep	5,78%	**1,65%**	1,72%	14,70%
Oct	5,74%	**1,78%**	1,99%	10,63%
Nov	4,95%	**1,33%**	1,36%	10,67%
Dec	6,57%	2,17%	**1,86%**	10,18%

*Lowest MAPE in bold

Hour	S_naive	DHR	NNAR	MLP
0	3,65%	**1,73%**	1,94%	11,14%
1	3,62%	**1,72%**	1,99%	11,37%
2	3,65%	**1,74%**	1,95%	11,39%
3	3,86%	**1,80%**	1,96%	11,70%
4	4,14%	**1,80%**	1,92%	12,16%
5	5,25%	**1,80%**	2,09%	12,57%
6	8,56%	**1,99%**	2,06%	13,23%
7	11,58%	**2,15%**	2,18%	14,56%
8	10,66%	**1,99%**	2,00%	14,09%
9	8,34%	1,87%	**1,83%**	13,20%
10	6,90%	1,80%	**1,78%**	12,80%
11	6,21%	1,80%	**1,79%**	12,52%
12	6,18%	1,94%	**1,89%**	12,34%
13	6,07%	1,96%	**1,88%**	12,16%
14	6,24%	1,97%	**1,90%**	12,45%
15	6,17%	1,95%	**1,81%**	12,56%
16	6,02%	1,96%	**1,92%**	12,60%
17	5,50%	**1,91%**	2,01%	12,62%
18	4,84%	**1,79%**	1,97%	12,26%
19	4,68%	**1,76%**	1,89%	11,94%
20	4,66%	**1,68%**	1,89%	11,78%
21	4,69%	**1,68%**	1,75%	11,71%
22	4,48%	**1,73%**	1,80%	11,33%
23	3,88%	**1,65%**	1,72%	10,99%

***Table 15**: Breakdown of Mean Absolute Percentage Error (MAPE) for NO1.*

	S_naive	DHR	NNAR	MLP
NO1	9,57%	**2,99%**	3,39%	18,57%

Weekday	S_naive	DHR	NNAR	MLP
Mon	10,19%	**3,35%**	3,42%	18,17%
Tue	11,96%	**2,86%**	2,87%	18,37%
Wed	6,59%	**3,18%**	3,25%	18,32%
Thu	6,12%	**3,04%**	3,52%	17,74%
Fri	7,16%	**2,97%**	3,82%	18,25%
Sat	12,34%	**2,78%**	3,53%	18,85%
Sun	12,60%	**2,74%**	3,35%	20,30%

Month	S_naive	DHR	NNAR	MLP
Jan	7,20%	**2,29%**	2,70%	14,31%
Feb	9,05%	**2,18%**	2,70%	15,94%
Mar	8,08%	**2,30%**	2,45%	12,81%
Apr	9,92%	**3,70%**	4,35%	13,52%
May	12,91%	**4,68%**	5,63%	20,98%
Jun	11,39%	**4,08%**	4,77%	24,52%
Jul	8,31%	**2,69%**	2,72%	26,97%
Aug	9,92%	3,48%	**3,26%**	32,26%
Sep	9,72%	**2,73%**	3,19%	20,21%
Oct	9,73%	**2,36%**	3,54%	14,48%
Nov	8,02%	2,14%	**2,12%**	12,76%
Dec	10,60%	**3,18%**	3,23%	13,75%

***Lowest MAPE in bold**

Hour	S_naive	DHR	NNAR	MLP
0	6,06%	**3,19%**	3,68%	18,17%
1	6,27%	**3,14%**	3,70%	18,76%
2	6,59%	**3,09%**	3,68%	18,89%
3	6,79%	**3,22%**	3,74%	19,14%
4	7,09%	**3,34%**	3,56%	19,80%
5	8,75%	**3,35%**	3,65%	20,16%
6	14,00%	**3,45%**	3,50%	19,95%
7	18,62%	3,74%	3,74%	19,92%
8	16,96%	**3,23%**	3,35%	18,66%
9	13,34%	**2,91%**	3,11%	18,02%
10	10,94%	**2,63%**	2,89%	17,15%
11	9,91%	**2,70%**	3,00%	17,81%
12	9,88%	**2,75%**	3,31%	17,36%
13	9,72%	**2,72%**	3,26%	17,41%
14	10,04%	**2,82%**	3,21%	17,26%
15	10,01%	**2,79%**	3,13%	17,82%
16	9,89%	**2,87%**	3,27%	18,99%
17	9,28%	**2,87%**	3,55%	19,31%
18	8,31%	**2,72%**	3,48%	19,43%
19	7,88%	**2,71%**	3,29%	19,35%
20	7,94%	**2,76%**	3,32%	18,54%
21	7,86%	**2,84%**	3,33%	18,31%
22	7,35%	**2,96%**	3,39%	17,83%
23	6,30%	**2,94%**	3,28%	17,67%

***Table 16**: Breakdown of Mean Absolute Percentage Error (MAPE) for NO2.*

	S_naive	DHR	NNAR	MLP
NO2	5,78%	**2,62%**	3,12%	10,43%

Weekday	S_naive	DHR	NNAR	MLP
Mon	6,54%	**2,66%**	2,94%	9,84%
Tue	7,24%	**2,60%**	3,12%	10,56%
Wed	3,97%	**2,88%**	3,47%	10,55%
Thu	4,16%	**2,91%**	3,27%	10,45%
Fri	4,72%	**2,68%**	3,21%	10,14%
Sat	6,70%	**2,34%**	2,91%	10,29%
Sun	7,12%	**2,31%**	2,90%	11,14%

Month	S_naive	DHR	NNAR	MLP
Jan	6,04%	**2,18%**	2,89%	9,53%
Feb	5,99%	**3,07%**	4,02%	10,73%
Mar	5,44%	**2,42%**	3,54%	8,97%
Apr	6,79%	**3,24%**	3,49%	8,29%
May	6,35%	**3,29%**	3,73%	10,86%
Jun	5,48%	3,16%	**2,86%**	11,16%
Jul	4,72%	**2,12%**	2,46%	11,87%
Aug	5,04%	**2,00%**	2,26%	14,88%
Sep	5,75%	**2,35%**	2,85%	11,79%
Oct	5,78%	**2,37%**	3,06%	8,92%
Nov	5,05%	**2,08%**	2,51%	9,02%
Dec	6,97%	**3,27%**	3,79%	9,08%

***Lowest MAPE in bold**

Hour	S_naive	DHR	NNAR	MLP
0	4,00%	**2,62%**	3,35%	9,84%
1	4,00%	**2,67%**	3,39%	9,97%
2	3,97%	**2,61%**	3,34%	10,02%
3	3,99%	**2,70%**	3,36%	10,18%
4	4,23%	**2,64%**	3,28%	10,38%
5	5,11%	**2,55%**	3,16%	10,50%
6	8,40%	**2,72%**	3,11%	11,13%
7	10,93%	**2,80%**	3,13%	11,40%
8	9,78%	**2,57%**	2,91%	11,24%
9	7,54%	**2,42%**	2,80%	10,64%
10	6,44%	**2,45%**	2,89%	10,17%
11	6,04%	**2,70%**	3,11%	10,53%
12	6,01%	**2,66%**	2,93%	10,43%
13	6,07%	**2,67%**	2,97%	10,35%
14	6,35%	**2,76%**	3,14%	10,17%
15	6,41%	**2,80%**	3,23%	10,47%
16	6,29%	**2,76%**	3,21%	10,83%
17	5,67%	**2,66%**	3,07%	10,91%
18	4,92%	**2,65%**	3,13%	10,84%
19	4,72%	**2,54%**	3,08%	10,75%
20	4,71%	**2,47%**	3,01%	10,28%
21	4,69%	**2,49%**	2,91%	9,89%
22	4,38%	**2,49%**	3,01%	9,66%
23	4,12%	**2,57%**	3,28%	9,63%

Table 17: *Breakdown of Mean Absolute Percentage Error (MAPE) for NO3.*

	S_naive	DHR	NNAR	MLP
NO3	5,19%	**2,72%**	3,88%	9,51%

Weekday	S_naive	DHR	NNAR	MLP
Mon	5,81%	**2,94%**	4,05%	10,07%
Tue	5,97%	**2,55%**	3,77%	9,89%
Wed	4,51%	**2,80%**	3,95%	10,03%
Thu	4,24%	**2,63%**	3,81%	8,89%
Fri	4,25%	**2,79%**	4,15%	9,27%
Sat	5,50%	**2,67%**	3,61%	9,07%
Sun	6,04%	**2,69%**	3,81%	9,33%

Month	S_naive	DHR	NNAR	MLP
Jan	5,82%	**2,44%**	3,61%	7,47%
Feb	5,42%	**2,57%**	3,41%	10,72%
Mar	4,69%	**2,34%**	3,73%	8,44%
Apr	5,67%	**2,74%**	4,06%	7,51%
May	5,61%	**2,84%**	4,10%	8,11%
Jun	5,62%	**3,01%**	3,90%	10,63%
Jul	4,58%	**2,66%**	3,56%	10,06%
Aug	4,44%	**2,25%**	4,29%	14,01%
Sep	5,38%	**3,35%**	5,78%	10,85%
Oct	5,03%	**3,16%**	3,86%	8,32%
Nov	4,26%	**2,23%**	2,67%	8,87%
Dec	5,80%	**3,08%**	3,54%	9,22%

***Lowest MAPE in bold**

Hour	S_naive	DHR	NNAR	MLP
0	4,26%	**2,92%**	3,39%	9,19%
1	4,15%	**2,89%**	3,24%	9,29%
2	4,10%	**2,75%**	3,19%	9,42%
3	4,46%	**3,02%**	3,36%	9,68%
4	4,59%	**3,03%**	18,65%	9,85%
5	4,65%	**2,78%**	3,24%	9,70%
6	6,02%	**2,82%**	3,32%	9,86%
7	8,46%	**3,04%**	3,37%	10,13%
8	8,32%	**2,98%**	3,39%	9,93%
9	6,84%	**2,76%**	3,18%	9,76%
10	5,80%	**2,61%**	3,02%	9,45%
11	5,41%	**2,60%**	3,09%	9,22%
12	5,45%	**2,56%**	3,04%	9,17%
13	5,46%	**2,61%**	3,13%	9,32%
14	5,51%	**2,56%**	3,23%	9,17%
15	5,28%	**2,50%**	3,03%	9,05%
16	4,86%	**2,45%**	3,04%	9,32%
17	4,55%	**2,51%**	3,17%	9,71%
18	4,25%	**2,50%**	3,22%	9,75%
19	4,26%	**2,52%**	3,29%	9,45%
20	4,47%	**2,54%**	3,25%	9,66%
21	4,59%	**2,70%**	3,32%	9,51%
22	4,65%	**2,85%**	3,47%	9,42%
23	4,20%	**2,87%**	3,44%	9,21%

Table 18: *Breakdown of Mean Absolute Percentage Error (MAPE) for NO4.*

	S_naive	DHR	NNAR	MLP
NO4	5,38%	2,65%	3,34%	9,35%

Weekday	S_naive	DHR	NNAR	MLP
Mon	6,37%	**2,62%**	3,58%	9,07%
Tue	6,55%	**2,94%**	3,69%	9,44%
Wed	3,97%	**2,45%**	3,25%	10,16%
Thu	4,41%	**2,65%**	3,14%	8,97%
Fri	4,33%	**2,62%**	3,38%	8,93%
Sat	5,85%	**2,57%**	3,02%	9,65%
Sun	6,13%	**2,72%**	3,33%	9,23%

Month	S_naive	DHR	NNAR	MLP
Jan	5,5%	**2,6%**	3,3%	7,1%
Feb	6,5%	**2,5%**	3,3%	10,5%
Mar	4,9%	**1,9%**	2,8%	7,9%
Apr	5,7%	**2,8%**	3,1%	7,1%
May	6,1%	**3,1%**	3,7%	8,9%
Jun	5,3%	**2,9%**	3,6%	10,4%
Jul	5,0%	**3,1%**	3,8%	11,0%
Aug	6,1%	**3,0%**	4,2%	14,0%
Sep	5,5%	**2,5%**	3,0%	11,0%
Oct	4,4%	**2,4%**	3,0%	7,7%
Nov	4,3%	**2,2%**	2,9%	8,9%
Dec	5,3%	**2,8%**	3,4%	7,7%

***Lowest MAPE in bold**

Hour	S_naive	DHR	NNAR	MLP
0	4,50%	**2,76%**	3,39%	9,06%
1	4,51%	**2,78%**	3,38%	8,99%
2	4,41%	**2,75%**	3,48%	9,00%
3	4,40%	**2,74%**	3,44%	9,19%
4	4,91%	**2,93%**	3,42%	9,53%
5	5,55%	**2,92%**	3,47%	9,59%
6	7,03%	**2,76%**	3,41%	9,56%
7	8,69%	**2,87%**	3,51%	9,66%
8	8,31%	**2,73%**	3,43%	9,53%
9	6,96%	**2,52%**	3,26%	9,46%
10	6,16%	**2,62%**	3,44%	9,46%
11	5,72%	**2,56%**	3,21%	9,25%
12	5,83%	**2,57%**	3,44%	9,36%
13	5,46%	**2,61%**	3,23%	9,41%
14	5,50%	**2,59%**	3,10%	9,71%
15	5,17%	**2,57%**	3,03%	9,60%
16	4,83%	**2,60%**	3,15%	9,74%
17	4,39%	**2,44%**	3,22%	9,66%
18	4,48%	**2,56%**	3,33%	9,67%
19	4,35%	**2,52%**	3,33%	9,29%
20	4,34%	**2,40%**	3,30%	9,05%
21	4,51%	**2,53%**	3,16%	8,97%
22	4,57%	**2,66%**	3,48%	8,74%
23	4,46%	**2,68%**	3,58%	8,96%

***Table 19**: Breakdown of Mean Absolute Percentage Error (MAPE) for NO5.*

	S_naive	DHR	NNAR	MLP
NO5	6,64%	**3,53%**	4,66%	11,71%

Weekday	S_naive	DHR	NNAR	MLP
Mon	6,77%	**3,44%**	4,06%	10,81%
Tue	7,41%	**3,68%**	4,74%	11,92%
Wed	5,66%	**3,61%**	4,74%	12,73%
Thu	4,87%	**3,12%**	4,53%	11,39%
Fri	6,59%	**3,90%**	5,16%	10,74%
Sat	7,66%	**3,60%**	4,74%	11,77%
Sun	7,51%	**3,33%**	4,63%	12,59%

Month	S_naive	DHR	NNAR	MLP
Jan	6,56%	**2,38%**	3,08%	9,37%
Feb	5,57%	**2,22%**	2,89%	11,56%
Mar	4,87%	**2,31%**	3,29%	8,88%
Apr	6,62%	**3,61%**	4,14%	7,87%
May	7,75%	**4,10%**	6,01%	11,44%
Jun	6,64%	**3,67%**	4,06%	10,09%
Jul	7,16%	**4,77%**	5,76%	12,54%
Aug	9,45%	**5,83%**	8,36%	18,62%
Sep	6,72%	**3,89%**	5,49%	18,73%
Oct	7,25%	**4,37%**	6,34%	12,54%
Nov	5,51%	**2,93%**	3,48%	10,24%
Dec	5,50%	**2,12%**	2,79%	8,60%

***Lowest MAPE in bold**

Hour	S_naive	DHR	NNAR	MLP
0	5,04%	**3,24%**	4,46%	11,24%
1	5,14%	**3,38%**	4,64%	11,63%
2	5,25%	**3,42%**	4,61%	11,57%
3	5,48%	**3,44%**	4,92%	11,75%
4	5,68%	**3,60%**	4,91%	11,96%
5	6,46%	**3,85%**	4,86%	12,05%
6	8,93%	**3,96%**	4,98%	12,19%
7	11,04%	**3,82%**	5,03%	12,28%
8	10,27%	**3,49%**	4,58%	12,14%
9	8,33%	**3,59%**	4,55%	11,54%
10	7,07%	**3,57%**	4,45%	11,29%
11	6,68%	**3,45%**	4,29%	11,33%
12	6,67%	**3,58%**	4,48%	11,57%
13	6,57%	**3,86%**	4,59%	11,64%
14	6,89%	**3,75%**	4,93%	11,75%
15	6,73%	**3,59%**	4,82%	11,73%
16	6,73%	**3,51%**	4,62%	11,65%
17	6,52%	**3,54%**	4,99%	12,23%
18	5,97%	**3,44%**	4,77%	12,16%
19	5,83%	**3,40%**	4,72%	12,00%
20	5,90%	**3,36%**	4,45%	11,64%
21	5,52%	**3,28%**	4,26%	11,19%
22	5,50%	**3,25%**	4,50%	11,24%
23	5,23%	**3,29%**	4,42%	11,21%

Overall, the DHR errors are the most accurate of the models, but the breakdown highlights where the NNAR model performs better, namely for workdays, a few months, and between the hours 09:00-16:00 in the peak hours for the NO zone. As for the NO1-NO5 zones, the DHR model outperforms the NNAR consistently, with a few exceptions. To summarize the results, we take the average MAPE across the zones and display the findings in Figure 25 to Figure 27.

Looking at the separate weekdays in Figure 25, the seasonal naïve suffers as expected when using a 48-hour lag, as predicting a weekday using data from the weekend causes the consumption to miss somewhat for the affected days. As for the DHR and NNAR model, the model accuracy between each weekday is mostly consistent, with a difference of less than 1% between the max and minimum observation for NO-NO4. The MLP model also exhibit consistent errors across the weekdays.

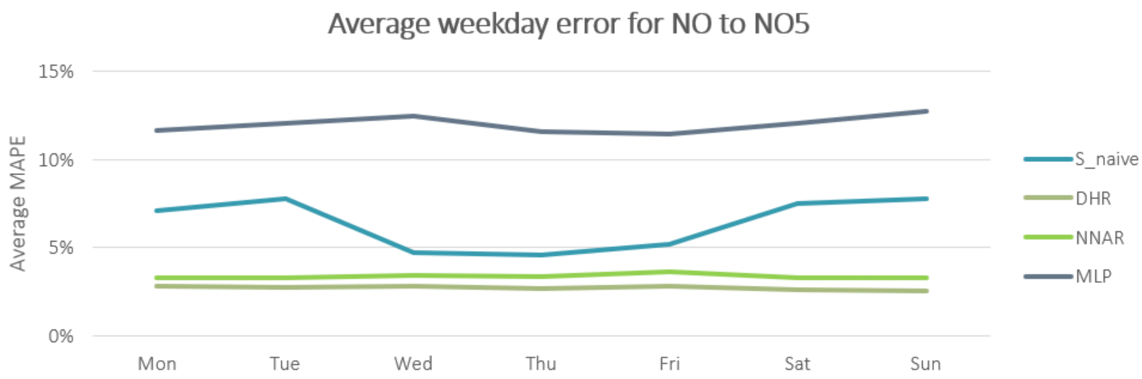

Figure 25: *Average MAPE across the zones for each weekday.*

For the separate months in Figure 26, we find the models to perform slightly worse for some of the summer months, with May peaking for the seasonal naïve, DHR and NNAR. This is perhaps due to more public holidays being present in these months, failing to capture the effect on consumption in a good manner. For the MLP model, the average MAPE peaks for August with about 18%, and shows a large variance in forecast accuracy.

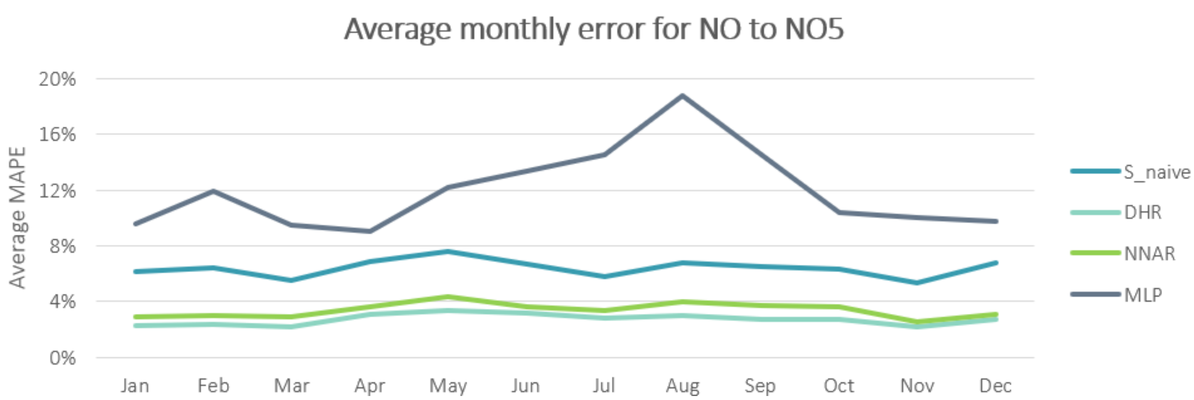

Figure 26: *Average MAPE across the zones for each month.*

Thirdly, we display the errors for the separate hours in Figure 27, where the DHR and MLP model has a relative consistent behavior for each hour, but the seasonal naïve and NNAR model has some deviation. The errors for the seasonal naïve start increasing from 04:00 and peaks at 07:00 during the first peak hour with about 12% error before decreasing but remains about 7% until 17:00 before another reduction for the night. The NNAR model on the other hand only seems to exhibit a higher error at 04:00 in the night, with the rest of the day having a consistent forecasting error.

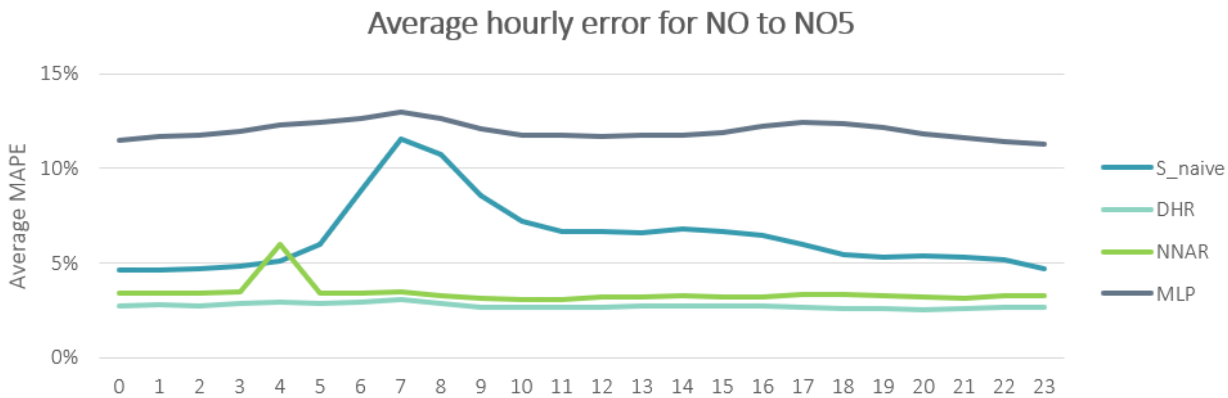

Figure 27: *Average MAPE across the zones for each hour.*

We also display the prediction errors for the separate public holidays in Table 20. Here, the errors are not consistent across the zones, perhaps indicating that the effect or participation rate is varied. As the test set only has one of each holiday present for every zone, conclusions should be drawn with care.

Table 20: *Mean Absolute Percentage Error (MAPE) for the holidays in 2019.*

Zone	Model	01/01	14/04	18/04	19/04	20/04	21/04	22/04	01/05	17/05	30/05	09/06	10/06	24/12	25/12	26/12	31/12
NO	S_naive	3,9%	11,8%	14,5%	13,9%	6,5%	4,9%	**1,5%**	6,7%	20,7%	4,8%	11,2%	**2,5%**	4,6%	3,2%	3,5%	6,1%
	DHR	3,9%	3,1%	7,0%	**3,1%**	3,3%	**1,2%**	3,6%	5,7%	**7,7%**	**1,9%**	**1,4%**	5,1%	**2,6%**	3,6%	**1,7%**	3,0%
	NNAR	**3,0%**	**1,0%**	**4,9%**	6,9%	**2,2%**	2,4%	2,4%	**3,8%**	12,4%	2,6%	5,5%	8,0%	3,6%	**1,6%**	2,5%	**1,5%**
	MLP	5,6%	12,0%	14,2%	12,5%	8,7%	8,2%	6,3%	50,0%	17,1%	6,4%	13,3%	31,0%	4,4%	4,6%	5,5%	3,5%
NO1	S_naive	11,4%	19,3%	23,2%	20,7%	11,1%	9,1%	4,8%	20,3%	37,2%	6,2%	23,8%	**3,0%**	7,0%	5,8%	7,1%	7,7%
	DHR	3,8%	4,4%	10,8%	**7,5%**	**5,4%**	3,3%	7,3%	11,2%	**17,6%**	**3,2%**	**6,1%**	9,1%	**3,8%**	5,9%	**3,1%**	**1,8%**
	NNAR	**2,6%**	**4,0%**	**8,0%**	11,9%	6,1%	**3,1%**	**2,4%**	**7,6%**	25,2%	7,7%	15,2%	19,0%	4,8%	**4,2%**	4,2%	2,6%
	MLP	10,4%	9,8%	10,3%	10,2%	7,3%	6,3%	4,1%	84,6%	22,8%	5,0%	45,1%	68,0%	5,0%	4,5%	5,0%	2,8%
NO2	S_naive	**1,9%**	10,4%	12,2%	12,8%	4,0%	2,2%	2,1%	**9,8%**	16,9%	5,8%	6,7%	**2,7%**	5,0%	2,5%	4,9%	4,4%
	DHR	2,8%	2,2%	4,6%	**4,7%**	3,3%	2,0%	1,8%	8,2%	**7,6%**	**3,0%**	2,5%	3,3%	**2,7%**	3,3%	**2,2%**	**2,8%**
	NNAR	6,7%	**1,5%**	**3,2%**	6,3%	**1,7%**	**1,9%**	**1,4%**	11,2%	9,8%	3,8%	**2,1%**	4,8%	4,9%	3,1%	5,6%	3,0%
	MLP	3,7%	5,9%	7,8%	9,9%	5,2%	4,4%	4,0%	49,2%	11,2%	4,0%	12,0%	25,1%	4,1%	**2,0%**	5,5%	3,2%
NO3	S_naive	**2,2%**	10,0%	13,0%	12,8%	**3,8%**	4,0%	**2,8%**	4,4%	16,3%	3,6%	14,3%	5,0%	2,7%	3,4%	3,6%	5,5%
	DHR	3,9%	**2,3%**	8,2%	**2,2%**	4,5%	**2,4%**	4,7%	**3,2%**	**5,5%**	2,7%	**2,4%**	5,8%	2,6%	2,5%	**1,8%**	3,4%
	NNAR	4,3%	3,0%	**5,4%**	8,1%	4,1%	6,1%	6,2%	5,4%	9,0%	2,0%	2,9%	**3,4%**	2,3%	**1,9%**	2,0%	**2,7%**
	MLP	3,6%	3,0%	7,0%	6,8%	5,8%	5,9%	2,9%	19,1%	9,1%	**1,9%**	18,8%	25,8%	**2,0%**	2,8%	3,0%	3,2%
NO4	S_naive	3,0%	7,2%	9,3%	6,0%	3,9%	5,8%	4,0%	2,9%	12,7%	9,3%	6,4%	8,3%	3,3%	3,3%	2,8%	8,2%
	DHR	4,8%	6,4%	7,0%	4,0%	3,5%	**2,5%**	**2,4%**	2,4%	**2,1%**	2,3%	**1,5%**	**2,2%**	3,4%	1,6%	**2,0%**	3,3%
	NNAR	**2,8%**	3,6%	5,4%	4,2%	**3,3%**	5,7%	2,7%	**2,3%**	4,8%	**2,2%**	1,9%	3,8%	3,9%	2,3%	2,5%	**3,2%**
	MLP	4,4%	**2,1%**	**3,5%**	**3,8%**	5,0%	2,9%	3,8%	18,8%	4,9%	4,1%	12,3%	11,9%	**2,3%**	**1,4%**	3,2%	4,9%
NO5	S_naive	7,5%	8,3%	11,2%	17,2%	12,8%	8,3%	4,2%	9,1%	19,2%	3,6%	4,6%	3,6%	7,4%	**1,9%**	3,4%	5,4%
	DHR	8,8%	2,4%	6,1%	**7,7%**	8,9%	**2,7%**	8,5%	**6,3%**	**10,7%**	2,7%	**2,0%**	3,8%	**3,5%**	2,2%	**1,3%**	**2,0%**
	NNAR	**5,4%**	**1,5%**	5,3%	8,0%	7,9%	5,3%	4,9%	8,5%	18,7%	4,7%	2,8%	**3,6%**	5,6%	4,5%	4,8%	2,5%
	MLP	10,4%	3,9%	**5,1%**	9,3%	**4,5%**	4,1%	**2,5%**	33,9%	12,3%	**2,4%**	11,7%	10,3%	5,1%	2,3%	2,8%	2,6%

***Lowest MAPE in bold**

7.2. Forecast showcase

To showcase what the output from the DHR model looks like, Figure 28 displays the actual demand compared to the forecasted values for a random selection of weeks in 2019: 6, 18, 24, and 44 for the NO zone.

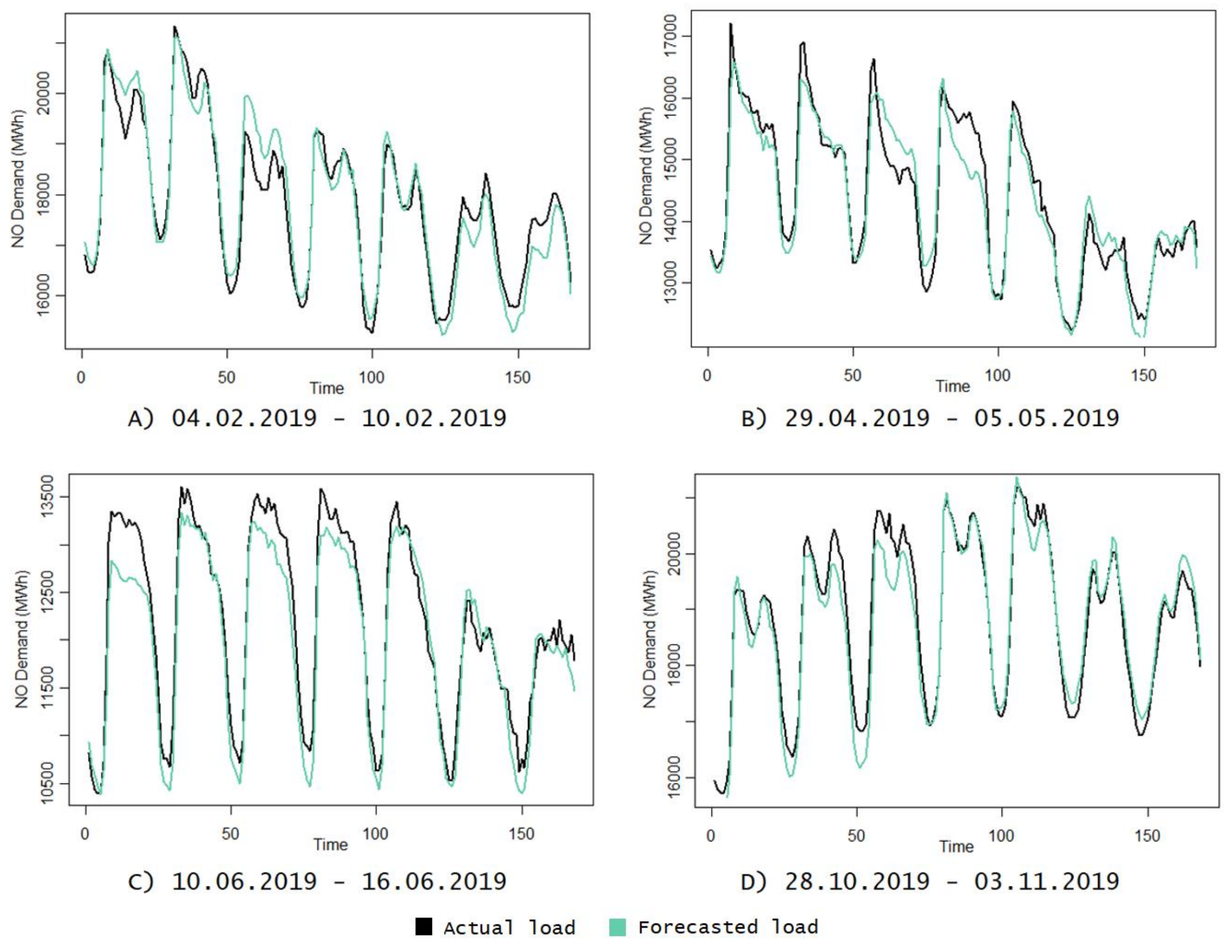

Figure 28: *Actual vs. forecasted NO demand for randomly selected weeks (6, 18, 24, and 44).*

For the different weeks displayed in A) to D), one can see the differences in load pattern, where each week exhibits a different shape of consumption. In A) One can clearly see the peak hours going through the week, as well as a drop-off in the load mid-week, which might be caused by a temperature increase persisting throughout the rest of the week. As for the load pattern in B), being the start of May, one can see a different shape for the peak hours, with a prevalent peak in the morning but almost no peak later during the day. The peak hours in C) are harder to distinguish in comparison, except for Saturday, with a seemingly more stable consumption throughout the day for June. Lastly, D) showcases the more pronounced peak hours as the season heads toward winter again, increasing as the week progresses. As for the forecasted values, the model seems to adjust well to changes, but clearly misses somewhat when sudden changes in the consumption occurs. However, the forecasts for the next day seem to adjust well in most cases.

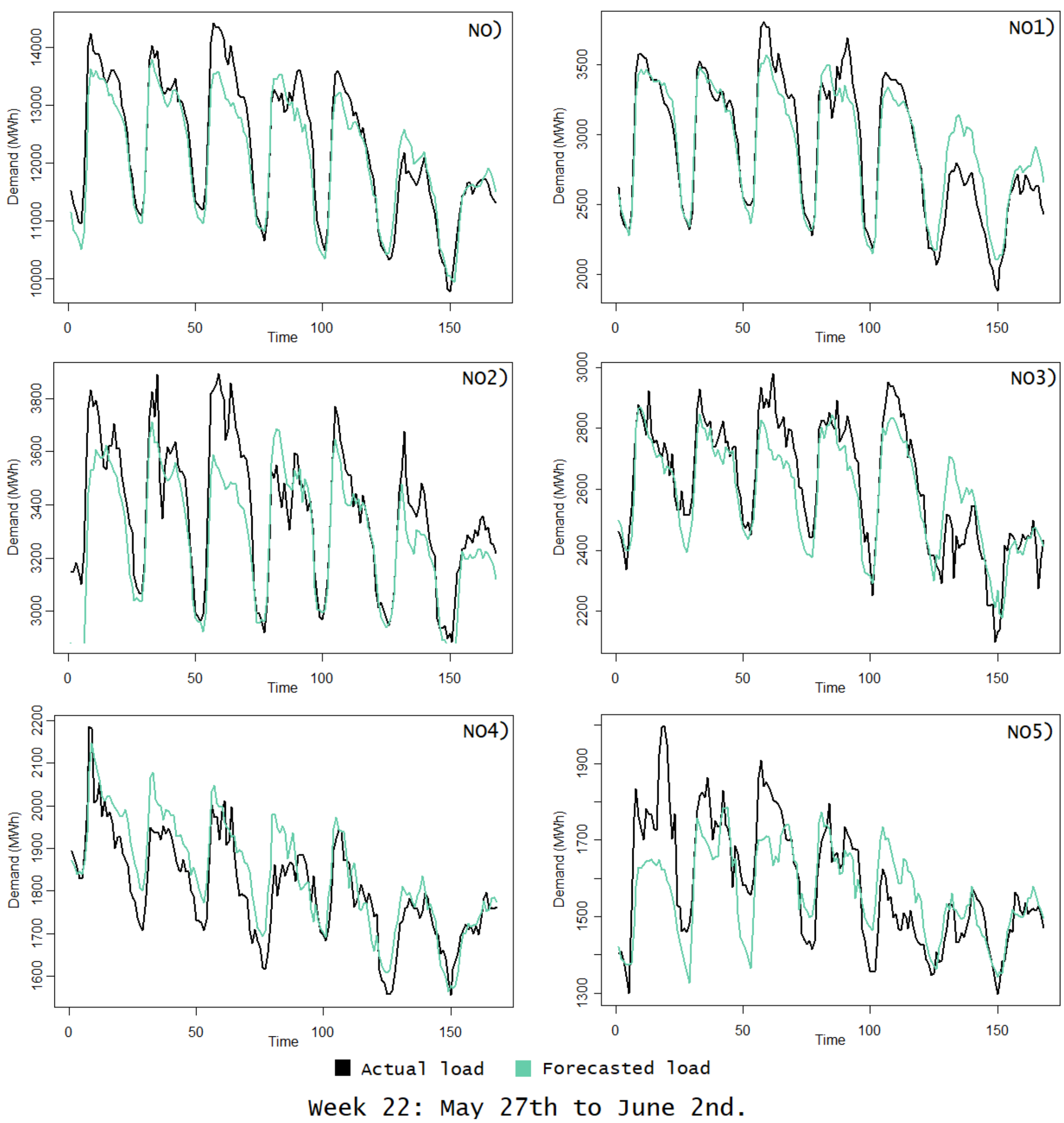

Figure 29: *Actual vs. forecasted demand of the different zones for week 22.*

As another showcase in Figure 29, the forecasts of the different zones are displayed for week 22. Going from NO to NO5, one can see how the accuracy of the forecast values is reduced as the total demand goes down and the data become less aggregated. Comparing NO5 to NO, the changes in consumption during the day are more volatile, likely as individual behavior has a bigger effect on the load here, whereas larger aggregations will smooth out their presence. Smaller aggregations also increase the samples magnitude of the peak hours

8. Discussion

Comparing the models, one important thing to note is the differences in the exogenous variables included. While the baseline only contains the dependent variable, the DHR and the ANN models are constructed using various lags and a different approach to modelling seasonality. Since the models contain different variables, one might argue they are not directly comparable. On the other hand, since the variables constructed are derived from the same data, it might be a fair comparison given their individual optimization and prerequisites for handling data. Leaving out fairness, one can also say that only the results should matter in the end.

Considering the fit of the hourly models, the DHR model was optimized using the 09:00 load as a generalization for the rest of the hours, while the neural network models were optimized using the 00:00 load. As such, conducting individual optimizations for each separate hour could have yielded better results. Judging from section 7.1 however, the observed model accuracy across the different hours seems consistent with a variation of less than 1% in the DHR model, indicating that the general fit perhaps could work well for all hours.

We have also seen from the results how the monthly and weekday errors are consistent for the DHR and NNAR model. Given that the errors are persistent enough, they could potentially be adjusted manually by an analyst to overcome its shortcomings to some extent, although uncertainty would still be present. Like in the words of Box and Draper (1987): "Essentially, all models are wrong, but some are useful" (p. 424). As the forecasting models only are simplified approximations of reality, we can only accept that they are flawed and do the best out of the information they provide.

Looking at the performance of the individual months, we also know that they fluctuate slightly more during the spring and summertime, where historical data might not be as valuable due to differences in climate from year to year. One way to approach this could be to make individual models for each month of the year. At the same time, the models in this thesis are working on actual historical weather data, so the errors cannot be due to bad weather forecasts. Therefore, it might be that the effect of the monthly binary variables has a negative effect on the model for some months of the year, even though the effect on the whole year is relatively good. During feature selection for the neural networks, some of the monthly dummy variables had more explanatory power than others, and some experimenting was done. However, it seemed strange not to give the model knowledge of the time of the year. It would be interesting to experiment more, maybe by leaving out just the spring months, or replace them with one dummy for spring. Another possibility is that the weather in spring is not the cause of the prediction errors, but that the errors are large in certain months because of the holiday effect.

For the collected data, there are many available options for feature engineering, some including the construction of variables using neural networks, or wavelets on the history of the dependent variable or other exogenous variables. For the DHR model, seasonality is included using two Fourier-series, whereas the ANN models include a weekly- and monthly dummy variable. For this modelling challenge, other options could have been explored further, such as decomposing the time series for trend and seasonality before estimation and adding the effect back to produce the complete forecast. Instead, simpler transformations were conducted using the hourly temperature and load, where the inclusion of the daily average, maximum, minimum, and degree-days could improve the model fit in some cases. Another unexplored option is to include squared variables to better capture non-linear relationships, an option explored by Elamin and Fukushige (2018) who included the squared temperature in their SARIMAX model.

Additionally, the comfortable temperature where no heating or cooling is demanded has a range of 7°C according to Wang and Bielicki (2018) as discussed in section 2.3. However, the range of the comfort zone is dependent on the geographical location due to acclimation, and partly due to personal preference. This makes the construction of the heating- and cooling-degrees another challenge, as constraining all the price areas to the same variable might be a gross simplification. Exploring the comfort range could have been further examined for the price areas by testing and including the range with the best model fit.

As for the forecasting accuracy of the different holidays, we generally find the presence of a holiday to add uncertainty to the load forecasts, where large errors can occur in some cases. Modelling the public holidays, two options were explored: using a single aggregated variable or separate them into several to control for the individual effects on load. The single aggregation was used for the ANN models, whereas the DHR arranged a separate variable for Christmas and Easter as well, while the remaining holidays were kept aggregated. From the results, the forecasting accuracy of the holidays are varied, and could be due to failing to control the individual effects. As such, separating the individual holidays further could have been explored to see if it improved the model fit, but would also make for a less parsimonious model. Another option we did not explore was an aggregated holiday variable including Sundays, as the effect on the load pattern could be similar.

Lastly, although MLP and other types of neural networks have proven themselves at load forecasting in other papers, Nti et al. (2020) comments that they are better suited for markets where relationships between exogenous and endogenous variables are complex. A further dive into complicated methods of feature engineering could have benefited the neural networks. This can also be the reason why the NNAR only manages to outperform the DHR at certain times in the NO zone. Due to the strength of being able to handle more complex data, it could perhaps have a bigger potential on less aggregated STLF. For our models, only a few variables are included, whereas lesser aggregations could have included more micro-level data, such as information about households and industries, painting a fuller and more complex picture of reality.

9. Conclusion

Throughout this thesis we have investigated how to construct forecasting models for the short-term hourly electric load in Norway, and presented our predictions for the aggregated, and individual price areas. In our findings, a linear Dynamic Harmonic Regression (DHR) model performs better than the non-linear Neural Network Autoregression (NNAR) with an average MAPE across the predicted zones of 2,73% and 3,38% for the models respectively. A Multi-Layer Perceptron (MLP) neural network was also created but fails to deliver an adequate forecast accuracy with a MAPE of 11,98%, performing worse than the baseline seasonal naïve, with 6,40% MAPE.

Exploring the different methods available, a question that often arises is whether artificial intelligence will revolutionize the forecasting field in place of the more traditional statistics. From our results, this is not the case as the DHR provides a lower MAPE of 0,65% compared to the NNAR but might hypothetically have a similar potential for short-term load forecasting depending on the model construction and optimization. For the model performance, the DHR forecasts consistently better than the other models across all the zones, weekdays, months, and hours, with just a few exceptions for NO. This might indicate that the Norwegian market has mostly linear relationships with underlying determinants, and that the DHR captures these patterns better than the more complicated MLP, and the hybrid NNAR.

As for potential model improvements, collecting the weather temperature for the price areas could have been more thorough, for instance constructing an average for the zone depending on population size and density to better capture the effect. Similarly, this could also have been done for precipitation and wind, which was found to reduce model fit in our case. More calendar effects could also have been controlled for, perhaps by examining the model errors to identify further areas of improvement.

References

Aanensen, T., & Holstad, M. (2018). Tilgang og anvendelse av elektrisitet i perioden 1993-2017. *Statistics Norway Report 2018/16*.

Box, G., & Jenkins, G. (1970). *Time series analysis: forecasting and control*. Holden-Day.

Box, G. E., & Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological)*, *26*(2), 211-243.

Box, G. E., & Draper, N. R. (1987). *Empirical model-building and response surfaces*. John Wiley & Sons.

Brooks, C. (2019). *Introductory econometrics for finance (3rd ed.)*. Cambridge university press.

Chow, T., & Leung, C.-T. (1996). Nonlinear autoregressive integrated neural network model for short-term load forecasting. *IEE Proceedings-Generation, Transmission and Distribution*, *142*(5), 500-506.

El-Hendawi, M., & Wang, Z. (2020). An ensemble method of full wavelet packet transform and neural network for short term electrical load forecasting. *Electric Power Systems Research*, *182*, 106265.

Elamin, N., & Fukushige, M. (2018). Modeling and forecasting hourly electricity demand by SARIMAX with interactions. *Energy*, *165*, 257-268.

EnergifaktaNorge. (2017). *Everything you need to know about the Norwegian energy sector*. Energifaktanorge. https://energifaktanorge.no/en/norsk-energiforsyning/kraftmarkedet/

Fan, S., & Hyndman, R. J. (2011). Short-term load forecasting based on a semi-parametric additive model. *IEEE Transactions on Power Systems*, *27*(1), 134-141.

Ferreira, M. A., & Gamerman, D. (2000). Dynamic Generalized Linear. In D. K. Dey, S. K. Ghosh, & B. K. Mallick (Eds.), *Generalized linear models: a Bayesian perspective* (pp. 57-72). CRC Press.

Haben, S., Giasemidis, G., Ziel, F., & Arora, S. (2019). Short term load forecasting and the effect of temperature at the low voltage level. *International Journal of Forecasting*, *35*, 1469-1484.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media.

He, W. (2017). Load forecasting via deep neural networks. *Procedia Computer Science*, *122*, 308-314.

Holstad, M., & Pettersen, F. E. L. (2011). Hvordan reagerer strømforbruket i alminnelig forsyning på endringer i spotpris? *Økonomiske analyser 2/2011*

Howard, P. H., & Sylvan, D. (2015). The Economic Climate: Establishing Expert Consensus on the Economics of Climate Change. *Institute for Policy Integrity*, 438-441.

Hyndman, R. (2010). *Forecasting with long seasonal periods*. Robjhyndman. robjhyndman.com/hyndsight/longseasonality/

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: principles and practice* (2nd ed.). OTexts. https://otexts.com/fpp2/

Hyndman, R. J., Athanasopoulos, G., Bergmeir, C., Caceres, G., Chhay, L., O'Hara-Wild, M., Petropoulos, F., Razbash, S., & Wang, E. (2020). Package 'forecast'. https://cran.r-project.org/web/packages/forecast/forecast.pdf

IEA. (2020). *Projected Costs of Generating Electricity 2020*. IEA. https://www.iea.org/reports/projected-costs-of-generating-electricity-2020

Jacob, M., Neves, C., & Vukadinović Greetham, D. (2020). Forecasting and Assessing Risk of Individual Electricity Peaks. https://doi.org/10.1007/978-3-030-28669-9

Kandananond, K. (2011). Forecasting electricity demand in Thailand with an artificial neural network approach. *Energies*, *4*(8), 1246-1257.

Koprinska, I., Wu, D., & Wang, Z. (2018). Convolutional neural networks for energy time series forecasting. 2018 international joint conference on neural networks (IJCNN),

Kwiatkowski, D., Phillips, P. C., Schmidt, P., & 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*(1-3), 159-178.

Kychkin, A. V., & Chasparis, G. C. (2021). Feature and model selection for day-ahead electricity-load forecasting in residential buildings. *Energy and Buildings*, *249*, 111200.

Lass, B. T., Ayuba, P., Damuut, L. P., Zachariah, B., & Amos, P. (2020). Forecasting Electric Load Demand Using Hybrid Nonlinear Autoregressive Neural Network with Exogenous inputs and Genetic Algorithm. *KASU Journal of Mathematical Science*, *1*(2), 90-103.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, *5*(4), 115-133.

Mill, S. (2016). *Electric load forecasting: advantages and challenges*. Electrical Equipment. https://engineering.electrical-equipment.org/electrical-distribution/electric-load-forecasting-advantages-challenges.html

Mir, A. A., Alghassab, M., Ullah, K., Khan, Z. A., Lu, Y., & Imran, M. (2020). A review of electricity demand forecasting in low and middle income countries: The demand determinants and horizons. *Sustainability*, *12*(15), 5931.

Mordjaoui, M., Haddad, S., Medoued, A., & Laouafi, A. (2017). Electric load forecasting by using dynamic neural network. *International journal of hydrogen energy*, *42*(28), 17655-17663.

NordPool. (2020). *Price Coupling of Regions (PCR)*. Nordpoolgroup. https://www.nordpoolgroup.com/the-power-market/Day-ahead-market/Price-coupling-of-regions/

Nordpool. (2021). Illustration of the Norwegian price areas. *Interactive map*. https://www.nordpoolgroup.com/Market-data1/#/nordic/map

Nti, I. K., Teimeh, M., Nyarko-Boateng, O., & Adekoya, A. F. (2020). Electricity load forecasting: a systematic reivew. *Journal of Electrical Systems and Information Technology*, *7*(1).

Schwert, G. W. (1989). Why does stock market volatility change over time? *The journal of finance*, *44*(5), 1115-1153.

Statnett. (2020). *Langsiktig markedsanalyse: Norden og Europa 2020-2050*. Statnett. https://www.statnett.no/globalassets/for-aktorer-i-kraftsystemet/planer-og-analyser/lma/langsiktig-markedsanalyse-norden-og-europa-2020-50_revidert.pdf

Suganthi, L., & Samuel, A. A. (2012). Energy models for demand forecasting—A review. *Renewable and sustainable energy reviews*, *16*(2), 1223-1240.

Wang, Y., & Bielicki, J. M. (2018). Acclimation and the response of hourly electricity loads to meteorological variables. *Energy*, *142*, 473-485.

Wangensteen, I. (2012). *Power system economics: the Nordic electricity market (2nd ed.)*. Tapir academic press.

Weron, R. (2014). Electricity price forecasting: A review of the state-of-the-art with a look into the future. *International Journal of Forecasting*, *30*(4), 1030-1081.

Weron, R., & Misiorek, A. (2005, May 10-12, 2005). *Forecasting spot electricity prices with time series models* Proceedings of the European electricity market EEM-05 conference, Lodz, Poland.

West, M., Harrison, P. J., & Migon, H. S. (1985). Dynamic generalized linear models and Bayesian forecasting. *Journal of the American Statistical Association*, *80*(389), 73-83.

Yazici, I., Temizer, L., & Beyca, O. F. (2019). Short term electricity load forecasting with a nonlinear autoregressive neural network with exogenous variables

(NarxNet). In *Industrial Engineering in the Big Data Era* (pp. 259-270). Springer.

Yukseltan, E., Yucekaya, A., & Bilge, A. H. (2020). Hourly electricity demand forecasting using Fourier analysis with feedback. *Energy Strategy Reviews*, *31*, 100524.

Appendix 1: Norwegian holidays

The data for Norwegian holidays is retrieved from norskkalender.no, taking into account the holidays which fall on different dates shown in.

Table 21: *Norwegian holidays.*

Holiday / year	2013	2014	2015	2016	2017	2018	2019
				Date			
1. Nyttårsdag	01/01	01/01	01/01	01/01	01/01	01/01	01/01
Palmesøndag	24/03	13/04	29/03	20/03	09/04	25/03	14/04
Skjærtorsdag	28/03	17/04	02/04	24/03	13/04	29/03	18/04
Langfredag	29/03	18/04	03/04	25/03	14/04	30/03	19/04
Dag før 1. påskedag	30/03	19/04	04/04	26/03	15/04	31/03	20/04
1. påskedag	31/03	20/04	05/04	27/03	16/04	01/04	21/04
2. påskedag	01/04	21/04	06/04	28/03	17/04	02/04	22/04
Den internasjonale arbeiderdagen	01/05	01/05	01/05	01/05	01/05	01/05	01/05
Kristi Himmelfartsdag	09/05	29/05	14/05	05/05	25/05	10/05	30/05
Grunnlovsdag	17/05	17/05	17/05	17/05	17/05	17/05	17/05
1. pinsedag	19/05	08/06	24/05	15/05	04/06	20/05	09/06
2. pinsedag	20/05	09/06	25/05	16/05	05/06	21/05	10/06
Julaften	24/12	24/12	24/12	24/12	24/12	24/12	24/12
1. juledag	25/12	25/12	25/12	25/12	25/12	25/12	25/12
2. juledag	26/12	26/12	26/12	26/12	26/12	26/12	26/12
Nyttårsaften	31/12	31/12	31/12	31/12	31/12	31/12	31/12

Appendix 2: Grid search for Multi-Layer Perceptron model

```
# GridSearch
from sklearn.model_selection import GridSearchCV

mlp = MLPRegressor(max_iter=10000)

parameter_space = {
    'hidden_layer_sizes': [(100,50,50), (50,100,50), (100),(28), (28,14,7),(28,100)],
    'activation': ['tanh', 'relu','logistic'],
    'solver': ['sgd', 'adam'],
    'alpha': [0.0001, 0.05, 0.1, 0.2],
    'learning_rate': ['constant','adaptive'],
    'random_state': [0, 1, 12, 42, 57, 60, 100]
}

clf = GridSearchCV(mlp, parameter_space, n_jobs=-1, cv=3)
clf.fit(X_train, y_train.values.ravel())

# Best parameter set
print('Best parameters found:\n', clf.best_params_)

Bestparametersfound = clf.best_params_

import json

a_file = open("GridSearchMLP.json", "w")
json.dump(Bestparametersfound, a_file)
a_file.close()
```

Appendix 3: RMSE results

Results displayed using Root Mean Squared Error (RMSE) instead of the Mean Absolute Percentage Error (MAPE) as displayed in Table 13 to Table 20.

***Table 22**: Root Mean Squared Error (RMSE) for the models across the bidding zones*

Model RMSE	**NO**	**NO1**	**NO2**	**NO3**	**NO4**	**NO5**	**Average**
Baseline: Seasonal naïve	888,5	384,4	236,9	156,7	118,2	116,7	316,9
Dynamic Harmonic Regression	276,3	115,5	105,5	80,9	57,4	59,8	115,9
Neural Network Autoregression	285,1	131,6	127,3	113,6	72,5	78,1	134,7
Multi-layer Perceptron Regressor	1825,4	708,0	417,3	285,8	203,6	202,8	607,1
Absolute change from baseline							
DHR	612,3	268,9	131,4	75,8	60,8	56,9	201,0
NNAR	603,4	252,7	109,6	43,1	45,6	38,6	182,2
MLPRegressor	-936,9	-323,6	-180,4	-129,1	-85,4	-86,0	-290,2
Relative change from baseline							
DHR	68,9%	69,9%	55,5%	48,4%	51,4%	48,8%	57,2%
NNAR	67,9%	65,8%	46,2%	27,5%	38,6%	33,1%	46,5%
MLPRegressor	-105,4%	-84,2%	-76,1%	-82,4%	-72,3%	-73,7%	-82,3%

***Table 23**: Model breakdown of Root Mean Squared Error for NO.*

	S_naive	**DHR**	**NNAR**	**MLP**
NO	888,54	**276,27**	285,14	1825,44

Weekday	**S_naive**	**DHR**	**NNAR**	**MLP**
Mon	1102,5	321,6	**287,6**	1740,4
Tue	1235,9	**280,6**	284,2	1888,6
Wed	598,2	**289,7**	290,3	1954,1
Thu	597,6	279,8	**270,8**	1794,9
Fri	628,0	**279,3**	309,9	1711,6
Sat	1015,9	**247,9**	276,2	1825,8
Sun	1035,0	**234,8**	276,8	1861,4

Month	**S_naive**	**DHR**	**NNAR**	**MLP**
Jan	1081,9	**307,1**	344,8	1829,3
Feb	1109,9	330,5	**327,1**	2281,8
Mar	927,9	**272,5**	288,8	1829,0
Apr	938,0	351,7	**351,0**	1475,2
May	927,3	**291,3**	349,4	1724,1
Jun	725,2	288,0	**285,3**	1666,5
Jul	558,2	**191,9**	193,7	1766,9
Aug	670,4	**169,5**	182,0	2109,8
Sep	747,5	**212,2**	218,8	1835,8
Oct	894,0	**272,7**	301,1	1591,9
Nov	903,9	**241,8**	245,9	1978,7
Dec	1192,0	390,9	**336,7**	1849,4

***Lowest RMSE in bold**

Hour	**S_naive**	**DHR**	**NNAR**	**MLP**
0	518,9	**237,8**	269,3	1520,5
1	504,9	**232,3**	267,4	1513,1
2	503,4	**232,2**	258,1	1501,1
3	529,5	**239,1**	261,0	1532,9
4	570,6	**241,5**	257,2	1596,8
5	729,7	**245,7**	280,8	1679,3
6	1248,2	**286,3**	294,8	1881,9
7	1781,2	**327,1**	334,0	2196,9
8	1677,5	**314,3**	315,4	2186,1
9	1326,1	298,5	**288,3**	2068,9
10	1101,3	289,2	**281,9**	2012,5
11	980,4	285,5	**282,3**	1957,5
12	965,9	305,9	**293,8**	1912,0
13	944,7	308,4	**292,1**	1876,4
14	970,0	308,0	**296,5**	1919,7
15	962,0	308,0	**287,5**	1943,3
16	945,8	309,2	**304,1**	1960,7
17	866,2	**300,0**	317,4	1966,0
18	764,9	**280,0**	309,0	1907,6
19	739,1	**274,6**	291,6	1850,1
20	729,8	**260,1**	289,5	1809,2
21	726,5	**255,8**	264,5	1772,2
22	675,0	**254,6**	263,2	1678,4
23	563,5	**236,4**	243,8	1567,5

***Table 24**: Model breakdown of Root Mean Squared Error for NO1.*

	S_naive	DHR	NNAR	MLP
NO1	384,37	**115,52**	131,64	708,01

Weekday	S_naive	DHR	NNAR	MLP
Mon	442,8	**133,9**	135,3	699,5
Tue	518,4	**116,5**	119,9	754,8
Wed	280,5	**126,1**	135,5	741,9
Thu	265,8	**120,7**	142,3	710,8
Fri	290,4	**113,7**	143,8	709,6
Sat	449,6	**98,9**	125,5	673,8
Sun	440,5	**98,8**	119,5	664,7

Month	S_naive	DHR	NNAR	MLP
Jan	427,2	**135,3**	158,8	804,5
Feb	487,5	**114,7**	145,0	931,6
Mar	397,7	**111,4**	121,5	665,4
Apr	364,7	**130,4**	156,5	541,1
May	416,8	**146,1**	178,4	688,9
Jun	325,6	**117,1**	133,9	699,9
Jul	206,7	**66,6**	66,8	711,3
Aug	273,4	97,7	**88,8**	803,6
Sep	321,8	**89,1**	103,9	637,7
Oct	412,6	**98,7**	145,0	589,7
Nov	419,5	112,3	**109,5**	695,2
Dec	565,6	**166,3**	172,1	740,7

*Lowest RMSE in bold

Hour	S_naive	DHR	NNAR	MLP
0	222,9	**109,4**	127,6	615,4
1	221,5	**104,3**	120,4	610,2
2	228,3	**100,1**	120,0	601,6
3	235,0	**104,1**	121,4	603,8
4	244,9	**107,9**	115,1	621,4
5	304,8	**111,1**	119,2	648,9
6	518,2	125,5	**125,2**	693,3
7	741,8	150,0	**148,7**	762,1
8	698,7	**135,5**	139,7	750,5
9	563,3	**124,5**	132,5	746,7
10	466,7	**113,4**	124,1	720,5
11	420,2	**114,8**	129,4	747,7
12	415,5	**114,9**	139,4	723,1
13	405,6	**112,9**	135,9	717,1
14	416,8	**115,4**	135,6	708,2
15	418,8	**115,8**	133,0	731,8
16	418,5	**120,0**	139,1	786,9
17	395,1	**120,1**	150,6	802,9
18	354,1	**111,2**	145,1	803,2
19	335,1	**112,3**	136,6	792,3
20	335,1	**114,3**	135,4	748,3
21	326,0	**114,6**	134,6	724,5
22	295,6	**114,3**	131,5	685,9
23	242,5	**106,0**	119,7	645,9

***Table 25**: Model breakdown of Root Mean Squared Error for NO2.*

	S_naive	DHR	NNAR	MLP
NO2	236,92	**105,51**	127,34	417,32

Weekday	S_naive	DHR	NNAR	MLP
Mon	281,6	**109,0**	121,9	398,7
Tue	310,6	**107,8**	132,4	440,6
Wed	165,7	**118,4**	145,5	435,1
Thu	172,3	**118,4**	134,8	427,0
Fri	193,0	**107,8**	130,8	417,6
Sat	260,3	**89,1**	112,4	394,1
Sun	273,6	**88,0**	113,6	407,6

Month	S_naive	DHR	NNAR	MLP
Jan	310,1	**111,3**	148,1	481,3
Feb	283,0	**141,5**	187,3	532,3
Mar	250,1	**110,7**	163,2	419,7
Apr	266,4	**122,2**	134,4	333,4
May	224,6	**115,3**	131,0	388,5
Jun	185,1	107,9	**96,6**	373,2
Jul	152,8	**67,7**	79,2	393,8
Aug	165,6	**66,3**	74,8	470,6
Sep	205,9	**83,3**	101,1	406,2
Oct	240,3	**97,2**	125,3	360,7
Nov	238,8	**97,8**	118,4	435,1
Dec	323,1	**148,0**	172,6	420,2

*Lowest RMSE in bold

Hour	S_naive	DHR	NNAR	MLP
0	152,0	**97,7**	128,4	365,7
1	148,6	**97,7**	126,1	362,9
2	146,5	**93,9**	122,4	361,2
3	147,4	**97,1**	123,0	366,7
4	157,4	**95,6**	120,5	375,7
5	193,1	**94,3**	119,1	388,5
6	331,2	**106,2**	124,1	431,7
7	452,9	**114,6**	131,6	463,6
8	414,5	**107,2**	124,0	467,3
9	322,4	**103,4**	121,2	447,2
10	275,5	**104,6**	125,8	427,7
11	257,2	**115,8**	135,2	443,0
12	253,6	**112,3**	125,9	434,3
13	254,8	**112,3**	126,0	428,8
14	265,0	**115,4**	133,6	422,2
15	268,7	**117,7**	137,4	436,0
16	265,1	**116,6**	137,7	452,5
17	240,3	**111,9**	131,5	457,6
18	209,4	**110,5**	133,3	453,7
19	200,7	**105,8**	130,8	447,0
20	197,9	**102,3**	126,2	424,1
21	193,8	**101,3**	120,8	403,0
22	176,4	**99,0**	122,5	384,6
23	161,5	**98,9**	129,0	370,7

***Table 26**: Model breakdown of Root Mean Squared Error for NO3.*

	S_naive	DHR	NNAR	MLP
NO3	156,72	**80,92**	113,62	285,78

Weekday	S_naive	DHR	NNAR	MLP
Mon	182,9	**88,3**	120,1	302,0
Tue	188,7	**76,9**	111,9	306,9
Wed	136,0	**82,3**	115,5	303,1
Thu	127,4	**78,2**	112,2	268,8
Fri	127,9	**84,3**	122,0	285,2
Sat	160,8	**78,7**	104,9	270,1
Sun	172,6	**77,9**	108,8	264,0

Month	S_naive	DHR	NNAR	MLP
Jan	204,5	**84,9**	127,5	260,3
Feb	184,6	**87,5**	115,0	383,6
Mar	161,0	**79,6**	126,0	298,2
Apr	163,4	**78,1**	115,8	227,0
May	159,4	**79,8**	115,5	231,9
Jun	142,9	**76,9**	100,4	273,8
Jul	113,4	**64,3**	84,7	255,7
Aug	108,8	**54,6**	101,6	333,4
Sep	141,7	**86,3**	145,8	278,9
Oct	154,2	**96,3**	116,5	249,5
Nov	151,6	**79,0**	94,2	322,0
Dec	196,9	**104,0**	120,7	323,2

Hour	S_naive	DHR	NNAR	MLP
0	121,4	**81,6**	95,2	259,9
1	117,3	**79,5**	90,4	259,2
2	114,2	**75,3**	88,0	260,5
3	123,4	**82,6**	92,3	266,9
4	127,3	**82,5**	501,8	272,1
5	130,3	**76,6**	89,2	272,1
6	176,7	**81,4**	95,8	288,0
7	259,9	**92,3**	102,5	309,2
8	261,3	**93,5**	105,7	311,9
9	215,7	**86,9**	99,2	307,5
10	182,4	**82,1**	93,8	297,7
11	168,1	**80,9**	95,4	288,3
12	167,7	**79,1**	93,7	286,5
13	168,0	**80,4**	95,8	289,1
14	169,3	**79,1**	99,6	284,8
15	163,6	**77,8**	94,2	281,8
16	150,3	**75,2**	94,2	291,2
17	141,3	**77,0**	97,3	300,6
18	132,0	**77,0**	99,7	302,0
19	132,0	**77,2**	101,2	292,7
20	137,6	**77,0**	98,7	296,5
21	140,2	**81,0**	100,0	289,6
22	139,5	**84,1**	102,9	282,4
23	122,1	**82,1**	100,0	268,1

***Lowest RMSE in bold**

***Table 27**: Model breakdown of Root Mean Squared Error for NO4.*

	S_naive	DHR	NNAR	MLP
NO4	118,17	**57,39**	72,55	203,55

Weekday	S_naive	DHR	NNAR	MLP
Mon	145,7	**58,3**	78,8	197,8
Tue	150,5	**64,3**	81,3	213,0
Wed	88,6	**54,2**	71,3	224,5
Thu	96,3	**57,2**	68,2	196,5
Fri	94,5	**57,7**	75,3	201,4
Sat	123,9	**53,7**	64,5	205,8
Sun	127,1	**56,1**	68,3	185,6

Month	S_naive	DHR	NNAR	MLP
Jan	144,3	**67,4**	83,5	180,1
Feb	162,8	**63,7**	84,8	279,8
Mar	124,7	**48,8**	69,3	205,4
Apr	124,8	**61,9**	67,3	165,1
May	124,8	**63,8**	76,8	184,4
Jun	99,0	**52,9**	66,3	195,4
Jul	88,5	**53,7**	66,3	201,7
Aug	107,4	**53,1**	73,4	244,0
Sep	106,2	**47,2**	57,5	204,2
Oct	99,6	**53,6**	67,6	166,6
Nov	108,1	**55,3**	74,5	229,7
Dec	131,1	**67,5**	83,8	193,1

Hour	S_naive	DHR	NNAR	MLP
0	92,9	**56,7**	69,8	187,2
1	91,8	**56,0**	68,1	183,0
2	89,1	**54,8**	69,8	181,9
3	88,6	**54,3**	68,3	184,9
4	99,8	**58,6**	68,6	192,4
5	114,7	**59,3**	71,3	197,5
6	151,0	**58,0**	72,2	203,7
7	195,2	**63,3**	78,3	214,2
8	189,3	**61,0**	77,5	214,4
9	159,2	**56,8**	74,4	214,2
10	141,2	**59,5**	77,7	214,7
11	130,5	**57,9**	73,2	209,2
12	130,8	**57,8**	76,8	209,0
13	123,4	**58,8**	72,4	211,0
14	125,6	**58,7**	70,3	218,4
15	116,9	**58,1**	68,2	215,1
16	108,3	**58,1**	69,7	217,9
17	98,9	**54,6**	71,6	216,1
18	100,5	**56,6**	73,8	215,8
19	97,9	**56,3**	74,4	207,4
20	97,8	**53,6**	73,9	201,5
21	99,9	**55,6**	69,9	197,6
22	99,2	**57,1**	75,4	189,3
23	93,4	**56,0**	75,3	188,9

***Lowest RMSE in bold**

***Table 28**: Model breakdown of Root Mean Squared Error for NO5.*

	S_naive	DHR	NNAR	MLP
NO5	116,73	**59,78**	78,12	202,75

Weekday	S_naive	DHR	NNAR	MLP
Mon	124,6	**59,1**	69,3	185,6
Tue	138,4	**65,0**	83,0	210,2
Wed	102,6	**62,8**	81,7	223,5
Thu	88,7	**54,0**	77,8	204,1
Fri	114,4	**66,5**	86,4	195,9
Sat	126,0	**57,7**	76,2	197,8
Sun	122,0	**53,1**	72,3	202,0

Month	S_naive	DHR	NNAR	MLP
Jan	147,7	**52,8**	69,9	209,0
Feb	117,5	**46,3**	61,1	255,0
Mar	102,6	**48,6**	69,4	190,9
Apr	113,8	**60,5**	68,8	142,3
May	125,3	**66,3**	94,8	192,4
Jun	100,9	**56,5**	62,4	153,0
Jul	104,8	**68,8**	81,9	185,8
Aug	127,3	**77,7**	109,8	252,8
Sep	97,2	**54,9**	77,1	249,0
Oct	127,2	**75,8**	105,2	206,6
Nov	116,7	**61,6**	73,3	214,8
Dec	118,6	**46,0**	61,0	184,8

***Lowest RMSE in bold**

Hour	S_naive	DHR	NNAR	MLP
0	81,6	**50,9**	69,4	180,8
1	81,0	**51,7**	69,8	181,9
2	82,0	**51,7**	68,8	179,9
3	84,9	**51,7**	72,2	181,6
4	89,4	**55,2**	73,0	186,0
5	103,9	**60,2**	74,2	191,0
6	152,8	**64,8**	80,8	203,7
7	199,7	**66,5**	87,3	216,2
8	190,6	**62,9**	81,6	220,0
9	156,2	**65,0**	81,2	212,7
10	131,6	**64,5**	80,2	207,3
11	122,6	**62,2**	77,0	207,2
12	120,9	**63,3**	79,1	209,1
13	117,6	**67,8**	79,6	207,5
14	123,7	**65,3**	85,4	209,6
15	121,2	**63,4**	83,4	209,9
16	123,1	**63,0**	81,8	211,3
17	119,7	**63,0**	88,9	222,6
18	108,4	**60,1**	83,7	220,8
19	106,0	**59,6**	82,5	217,8
20	105,7	**58,2**	77,1	208,8
21	97,8	**56,0**	72,6	198,7
22	94,7	**54,5**	75,0	194,6
23	86,5	**53,3**	70,1	186,9

***Table 29:** Root Mean Squared Error (RMSE) for the holidays present in 2019.*

Zone	Model	01/01	14/04	18/04	19/04	20/04	21/04	22/04	01/05	17/05	30/05	09/06	10/06	24/12	25/12	26/12	31/12
NO	S_naive	665,2	1751,8	1911,7	1778,7	808,0	587,5	178,6	797,0	612,0	2313,5	1240,1	289,8	824,3	563,3	634,5	1075,5
	DHR	659,3	466,2	920,5	**394,2**	402,3	**138,6**	449,4	698,3	**245,5**	**861,1**	**156,2**	**587,1**	**457,5**	622,1	**314,8**	533,1
	NNAR	**525,3**	**143,8**	**649,6**	877,3	**267,0**	286,9	**303,0**	**464,7**	331,0	1382,9	615,2	922,2	650,3	**266,6**	463,8	**248,0**
	MLP	952,1	1785,7	1881,7	1599,7	1084,8	994,8	787,0	6218,7	833,5	1909,7	1476,4	3543,7	766,4	812,6	1007,9	582,3
NO1	S_naive	548,7	724,5	747,1	634,4	322,7	251,0	140,8	536,4	190,1	904,6	572,2	**72,3**	349,9	288,3	376,5	390,5
	DHR	178,2	164,1	349,5	**226,8**	**152,3**	88,8	212,0	298,4	**101,6**	**426,4**	**144,1**	232,7	**185,6**	286,0	**164,3**	**86,7**
	NNAR	**127,7**	**148,2**	**251,1**	358,3	173,2	**84,3**	**69,9**	**197,0**	251,7	611,9	366,7	479,9	237,3	**200,6**	228,1	133,9
	MLP	498,9	361,9	329,0	307,3	212,4	175,9	122,6	2282,9	155,7	545,8	1060,9	1698,2	241,5	223,5	270,5	135,2
NO2	S_naive	**86,7**	415,5	450,5	452,5	140,3	74,1	71,1	318,0	195,8	521,0	210,2	**90,0**	236,9	113,7	242,2	199,2
	DHR	130,1	89,5	170,1	**164,6**	118,2	68,6	64,4	**276,1**	**101,7**	**232,9**	79,0	104,9	**127,1**	148,7	**108,2**	**123,8**
	NNAR	319,8	**57,6**	**117,9**	219,3	**61,0**	**64,8**	**50,3**	368,7	132,9	299,3	**65,8**	156,9	235,3	139,8	266,4	138,9
	MLP	170,9	235,6	287,2	347,6	185,4	152,2	139,6	1651,4	136,9	342,5	379,7	808,4	189,3	**89,7**	266,4	144,1
NO3	S_naive	**72,0**	298,1	339,8	321,4	**94,4**	97,3	**68,2**	116,8	102,4	394,8	337,0	116,0	93,8	114,4	121,2	175,7
	DHR	128,5	**68,5**	212,9	**54,6**	115,0	**57,9**	114,7	**84,2**	77,2	**132,5**	**55,6**	136,8	84,0	83,8	**59,9**	111,8
	NNAR	144,0	89,4	**139,9**	204,0	102,0	148,0	153,4	147,0	55,2	217,0	66,9	**81,3**	76,5	**63,6**	63,5	**88,2**
	MLP	117,5	88,7	181,8	170,1	150,6	149,6	70,8	504,5	**54,2**	219,1	437,9	612,4	**70,1**	93,1	97,5	102,4
NO4	S_naive	71,5	159,8	190,2	123,8	76,4	112,7	81,1	59,6	173,6	242,2	107,4	156,0	76,1	81,3	67,3	203,7
	DHR	112,1	144,0	142,1	83,7	68,4	**49,7**	**47,9**	50,7	43,2	**40,5**	**25,6**	**40,8**	78,6	38,5	**48,9**	79,5
	NNAR	**65,0**	82,5	110,1	87,9	**64,2**	111,4	55,4	**48,4**	**41,2**	90,7	33,1	71,5	90,3	56,4	60,2	**74,5**
	MLP	106,5	**46,6**	**70,9**	**80,6**	98,0	56,6	76,7	392,4	78,2	92,9	208,1	215,7	**54,7**	**33,5**	76,5	120,2
NO5	S_naive	143,7	154,1	184,2	266,1	185,0	118,2	58,3	157,8	59,6	250,5	65,7	55,5	164,8	**39,4**	77,9	118,8
	DHR	167,0	43,7	100,3	**118,8**	127,8	**38,1**	119,2	**109,8**	45,7	**139,1**	**29,0**	55,8	**78,6**	48,2	**30,8**	**44,7**
	NNAR	**99,6**	**26,8**	88,0	122,7	115,1	75,0	68,4	134,5	78,7	243,0	40,5	**52,8**	123,1	99,2	110,4	53,1
	MLP	201,7	71,3	**82,9**	142,7	**64,8**	59,0	**35,3**	542,9	**40,3**	159,4	171,3	151,1	110,8	47,9	62,6	56,9

***Lowest RMSE in bold**

Appendix 4: Abbreviations

ACF – Autocorrelation function

ANN – Artificial Neural Network

ARIMA – Autoregressive Integrated Moving Average

ARIMAX – Autoregressive Integrated Moving Average with exogenous variables

CNN – Convolutional Neural Network

DHR – Dynamic Harmonic Regression

DR – Dynamic Regression

Electricity consumption – Amount of electricity used over time

Electricity load – Electricity consumption at a specific moment in time

LCOE – Levelized Cost of Energy

LR – Linear regression

LSTM – Long Short-term Memory Recurrent Neural Network

MA – Moving average

MAE – Mean Absolute Error

MAPE – Mean Absolute Percentage Error

MLP – Multi-Layer Perceptron

MSE – Mean Square Error

PACF – Partial autocorrelation function

RNN – Recurrent Neural Network

RMSE – Root Mean Squared Error

SARIMA – Seasonal Autoregressive Integrated Moving Average

SARIMAX – Seasonal Autoregressive Integrated Moving Average w/ Exogenous variables

SSE – Sum of squares error

STLF – Short-term load forecasting

Appendix 5: List of figures

Appendix 6: List of tables

Norges miljø- og biovitenskapelige universitet
Noregs miljø- og biovitskapelege universitet
Norwegian University of Life Sciences

Postboks 5003
NO-1432 Ås
Norway