Case Studies 2024

Forecasting the Electricity Load
Approach using Linear Models and Seasonal Filtering

Professors

Prof. Dr. Matei Demetrescu, Prof. Dr. Paul Navas

Author: Fedi Ghanmi

Group Members

Alicia Hemmersbach, Ketevan Gurtskaia, Lev Luskin

24^{th} Mai 2024

Table of Contents

1 Introduction

Since the greatest discovery of the electron in the late 19^{th} century, electricity has evolved to become a complicated network of infrastructures. In Germany specifically, its role in economic growth has been significant. Presently, four major transmission system operators (TSOs) take care of the distribution of electricity across the country. However, fluctuating demand poses a challenge to maintaining a stable production, hence additional costs endured by them. This study aims to address this challenge by understanding consumption patterns and forecasting future electricity demands. First of all, a data gathering step on electricity usage is made. Then, time series techniques will be employed to understand the correlations between terms of time and uncover the behavior of the data. Using additional variables like temperature data, future gas prices, CO_2 prices, electricity trading data, renewable energy forecasts and output per generation type data, a new linear model will be created, each time fitting it to a single variable and testing it to see its ability in representing real-world patterns. Using the most promising predictors, another model will be built. Eventually, the presence of seasonal patterns will be examined, data will be deseasoned and final forecasts will be generated. As a result of this process, it was found that electricity prices can be represented with an autoregressive model of order 3. In addition, some of the unit generation feature alongside the temperature and co_2 data proved to be of an added value when included in our base model. By further eliminating the yearly seasonality from them even better forecasts were produced to finally reach a mean squared forecast error of 682 after starting with 1317. In the following sections of this report, the initial chapter will explains more why a good forecast for electricity demand is needed and describe in more details the data used to solve it. Following that, the methods employed will be briefly explained and a theoretical background on them will be provided. Finally, the empirical findings that were briefly discussed in this introduction will be detailed in its corresponding section and conclusions from this study will be drawn.

2 Problem and Data Description

In this section, firstly the project objective are outlined with the respective research questions needed to be answered by this study and secondly the data is described in details.

2.1 Project Objective

The primary objectives of this study is to have a good forecast of the electricity demand. TSO's need this information in order to reduce the costs of overproducing electricity which when it happens, can lead them to sell it at a negative price. That is why the first research question to be asked is: Could the power consumption be predicted across time? To answer this, load correlations across different hours will first be considered. If the answer is yes, then is it also possible to state that other variables like temperature, CO_2 and gas prices, wind and solar energies production or electricity trading data have an effect on this consumption over time? Even if it is the case, any recurrent data across time bear the risk of including seasonality. So will it yield better results if the seasonal component is removed before forecasting? The primary measure that will be used to be guided in the right direction is the Mean Squared Forecast Error [1]. Its minimization is the key objective while also taking into consideration the risk of overfitting using an information criterion.

2.2 Data Description

Both the actual electricity load and the generation output per unit data are obtained from (ENTSO-e, 2008) which stands for European Network of Transmission System Operators for Electricity. They contain information respectively of the hourly recorded load and the output of energy per each generation unit in Germany. The date frame of interest spans from 1^{st} of January 2015 until 15^{th} of March 2024. Since this sample have an hourly frequency it contains $N = 80.688$ observation. Other features like the CO_2 and gas futures data come from the European Energy Exchange AG (EEX, 2002). The data contains dates following daylight savings time adjustments according to the CET zone [2]. Since time series model cannot deal with such gaps, both cases caused by the DST [3] will be addressed. When clocks are set forward, a duplicate value of 1 a.m will be made to also represent 2 a.m and when clocks are set backward the second instance of 2 a.m will be removed to avoid redundancy.

[1] Abbreviation: MSFE

[2] Central European Time Zone

[3] Daylight Savings Time

The total load data acquired is the sum of the information collected across all 4 TSO's that control the 4 parts of Germany's electricity grids covering the regions shown in table 2.1.

TSO	**Control Area**
Amprion	western and southern Germany
50Hertz	north-eastern Germany
TransnetBW	southern and central regions of Germany
TenneT DE	northern and eastern parts of Germany

Table 2.1: TSO's and their Respective Covered Regions in Germany

In table 2.2 the variables used, their contextual meaning, units and number of missing values encountered are recapitulated.

Name	**Interpretation**	**Units**	**Missing**
datetime_clean	date and time index	Hourly freq.	0
total_load_clean	Electricity demand	Megawatt (MW)	0
co2_prices	CO2 price	Euro (€)	0
gas_prices	Gas price	Euro (€)	778
temp	Temperature information	Celsius (°C)	2
wind	Day ahead wind energy forecast	Megawatt (MW)	13
solar	Day ahead solar energy forecast	Megawatt (MW)	11
net_trade	Export − Import of power	Megawatt (MW)	62316
holiday	Public holiday in Germany	Binary	0
fossil_output	Energy produced by fossil fuels	Megawatt (MW)	0
biomass_output	Energy produced by biomass	Megawatt (MW)	0
geothermal_output	Energy from geothermal sources	Megawatt (MW)	0
hydro_output	Energy from hydro sources	Megawatt (MW)	0
other_output	Energy produced by other sources	Megawatt (MW)	0

Table 2.2: Variables Summary

Since simply dropping the rows with missing data points do not make sense in a time series setting, the issue of missing values will be solved using the Last Observation Carried Forward method [4] for at least most of our variables. LOCF assumes that the current value is likely to be similar to the most recent observed one which is a realistic assumption when working on hourly data where previous adjacent data points are usually correlated. Only for the net_trade specifically that a zero imputation for countries having no import or export values was made and a proxy sum across all of them was computed. This way an

[4] Abbreviation: LOCF

approximate value for the net_trade for each instance is obtained. This type of imputation in this case was preferred than simply taking the previous possible observation since it is already known how the net value was calculated. Not all columns were obtained in an hourly frequency, some of them did not have values for each specific hour, but were averaged daily. This applies to the gas and co_2 price, temperature and day ahead forecasts of renewable energies (wind and solar). Here, the average value was duplicated across all 24 hours. Biomass, fossil, geothermal, hydro, nuclear and other output were all obtained from a single non-aggregated data. For better representation of categories, their output were summer per hour for each sub-unit. For example fossil gas, fossil oil and fossil hard coal were aggregated to represent the output using fossil sources, hence the fossil_output variable. The same logic applies to the rest of the output categories. Generation from nuclear sources was exceptionally added to the other_output due to the large number of missing values it had. Instead of risking interpolation with wrong information, integrating it with the "other" category was one of many ways to solve the problem. More complex interpolation strategies could be considered at this stage that could lead to better forecasts later on, nevertheless the straightforward methods already implemented in this project can also be sufficient to proceed with and would still provide reasonable results with regards to the time constraint for this study. An extra "holiday" attribute is added to our existing dataset. The motivation behind it is to account for the hidden variations in electricity demand on holidays [5] in Germany, which can be misinterpreted by the model as typical working days.

3 Methods

In this section, all statistical methods employed in our project are explained with their respective original authors.

[5] Only national public holidays that are common for all states

3.1 Statistical Methods

3.1.1 Autocorrelation Function

The Autocorrelation Function [6] is one of the statistical method used to analyze the correlation between a time series and its lagged values. The autocorrelation function of a time series $\{z_t\}$ at lag k, denoted as ρ_k, is defined as:

$$\rho_k = \frac{\text{Cov}(z_t, z_{t-k})}{\sqrt{\text{Var}(z_t) \cdot \text{Var}(z_{t-k})}} \tag{1}$$

In equation 1 if ρ_k is close to 1, it indicates a positive correlation between the two terms. If it is close to -1, it suggests a negative correlation. Eventually, if it is near 0, it implies little to no correlation. The correlation terms are then plotted in a correlogram that will provide insights to the behavior of the time series. See (Brockwell and Davis, 2016). If the plot shows continuous spikes and regulard intervals, it may be an indication of seasonality.

3.1.2 Partial Autocorrelation Function

The Partial Autocorrelation Function[7] is a continuity of ACF and specific for identifying the order of autoregressive models (Enders, 2004). It is done by not taking into account the lags information between two terms and only including their direct influence (Brockwell and Davis, 2016).

$$\phi_{n,n} = \frac{\rho(n) - \sum_{k=1}^{n-1} \phi_{n-1,k}\rho(n-k)}{1 - \sum_{k=1}^{n-1} \phi_{n-1,k}\rho(k)} \tag{2}$$

In equation 2 $\rho_{(n)}$ is the previously calculated autocorrelation in equation 1 according to (Durbin, 1960) and is substracted from previous lags depending on the choice of k. The coefficient $\phi_{0,0}$ is usually initialized to 1. A significant spike at lag k in the PACF plot suggests that lag k is a meaningful lag in the autoregressive model. If the PACF cuts off [8] after lag p it then suggests an AR(p) model (Enders, 2004).

[6] Abbreviation: ACF

[7] Abbreviation: PACF

[8] All partial autocorrelations after lag p are not statistically different from zero

3.1.3 Ordinary Least Squares

The Ordinary Least Squares [9] is a fundamental concept usually used in linear regression settings to estimate the parameters of a linear relationship between a dependent variable and one or more independent variables as it is seen in equation 3:

$$y = X\beta + \epsilon \tag{3}$$

The OLS method estimates the parameters β by minimizing a loss function. The result of this optimization is analytical and can be written in matrix format defined as in equation 4 where X is the so called design matrix and y the true values (Goldberger, 1964):

$$\hat{\beta} = \left(X^T X\right)^{-1} X^T y \tag{4}$$

One example of loss functions used in this study in the Mean Squared Forecast Errors [10], see equation 5, which is a loss function that can also be used beyond the OLS method to evaluate any model performance.

$$MSFE_z = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(z_t - \hat{z}_t)^2}. \tag{5}$$

The R^2, equation 6, is also employed as a part of a feature selection process to measure the goodness of fit (Casella, 2002) .

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{6}$$

3.1.4 Autoregressive Process

Autoregressive processes [11] are models that are used to describe a time series using its own past values (Eshel). As written in equation 7, an autoregressive process of order p is written as the sum of each lagged dependant variable multiplied by a coefficient:

[9] Abbreviation: OLS

[10] Abbreviation: MSFE

[11] Abbreviation: AR

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-k} + \epsilon_t \tag{7}$$

Autoregressive models usually assume non-seasonal data. In this case, one method to estimate p order of the AR process is by using the Yule-Walker Method (Kirchgässner and Wolters, 2007) .

3.1.5 Seasonality

Seasonality decomposition is a technique used to decompose a time series into three components: trend, seasonality, and residuals. This helps in understanding the underlying patterns and making better forecasts. Mathematically, it is denoted as:

$$y_t = S_t + T_t + R_t \tag{8}$$

Where S_t is the seasonal component, capturing periodic fluctuations, T_t is the trend component(whether upward or downward trend), and R_t is the residual component capturing any irregularities (Hyndman and Athanasopoulos, 2013). This formulation is appropriate if and only if both of the three components of equation 8 are constant over time. Seasonality can hurt forecasting abilities because by doing so, it violates some key assumptions of the autoregressive model. For instance such effects can result in autocorrelated residuals which goes against the white noise assumption of non-correlated error terms (Box et al., 2015). This violation can therefore lead to biased estimations. This would also apply for data that contains other components like trend across time making it non-stationary.

3.1.6 Akaike Information Criterion

Information criteria are a type of model selection criteria usually used when the test data is not available or expensive to obtain. One widely used criterion for model selection is the Akaike Information Criterion[12]. Mainly the AIC is defined in equation 9 below:

$$\text{AIC} = 2k - 2\ln(\hat{L}) \tag{9}$$

Here, the AIC measures the goodness of fit taken by the maximized logarithmic like-

[12] Abbreviation: AIC

lihood function written as $\hat{L}$ and tries to balance it with the model's complexity using k number of parameters (Akaike, 1974). Both sides calibrate each other which reduces overfitting risk that originates from increasing the model capacity (James et al., 2013).

3.2 Relevant Tools

As part of the work, Python version 3.11 (Guido van Rossum, 1989) with its Integrated Development Environment Pycharm Community Edition 2022.2.3 (JetBrains, 2010) were used to conduct our study. Rstudio (R Core Team, 2023) was also used for some prepossessing step. External packages used in both softwares are respectively summarized in table 3.1 and table 3.2.

Package	**Version**
tidyverse	2.0.0
kableExtra	1.4.0
lubridate	1.9.3
readxl	1.4.3
readODS	2.2.0
data.table	1.15.4
dplyr	1.1.4

Table 3.1: List of R Packages

Package	**Version**
pip	23.0.1
Pandas	1.5.1
matplotlib	3.7.1
seaborn	0.12.2
statsmodels	0.13.5
numpy	1.23.4
scikit-learn	1.3.2
mlxtend	0.23.1
holidays	0.48
datetime	$*$ [11]
warnings	$*$ [11]

Table 3.2: List of Python Packages

[11] The datetime and warnigns package are a part of python core distribution and hence do not have a specific version.

4 Empirical Results

The results obtained from the practical implementation of this study are detailed in this section and are divided into three parts. The first part starts with a simple autoregressive model, then the second part includes other predictors and finally the last part implement deseasoning on the data with the final chosen predictors.

4.1 Simple Autoregressive Model

In fig. 4.1, the smoothed load data across time is plotted with a monthly aggregation. It clearly shows a monthly seasonal patterns, with peaks and troughs recurring at a 12 month intervals. The lowest happens mid-year and the highest occur and the end of one year and the beginning of the next. Two significant drops takes place around the year 2020 and 2023, which could be attributed respectively to the COVID-19 pandemic and to the Ukraine war.

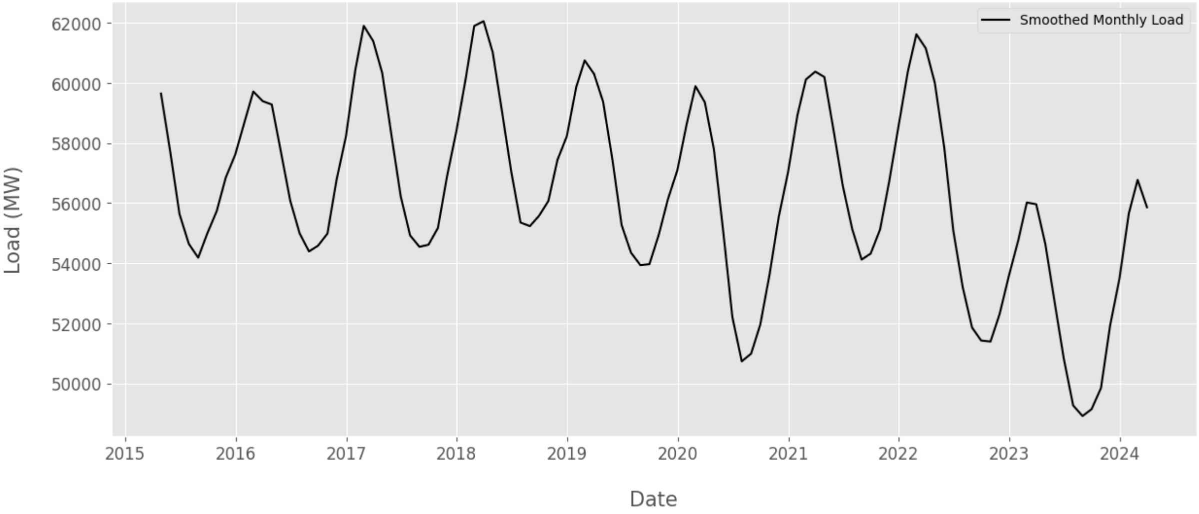
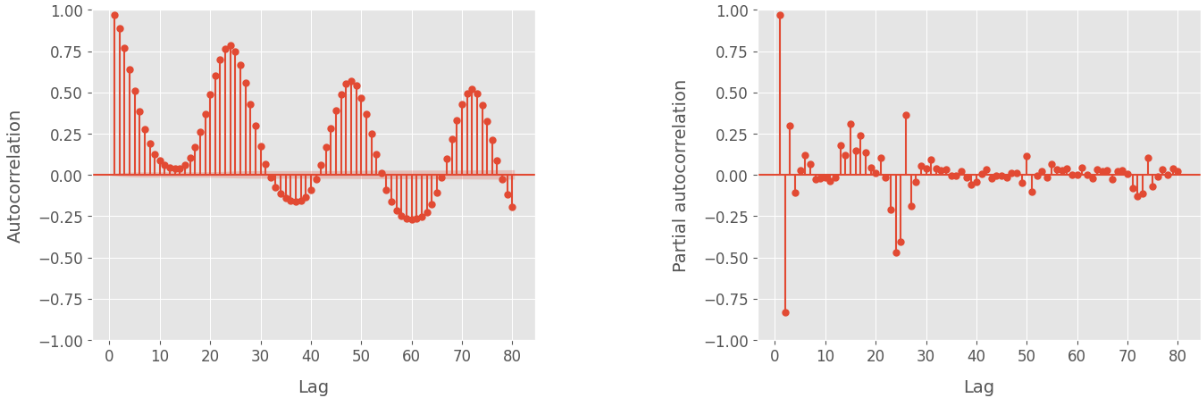


Figure 4.1: Running monthly average variation of electricity load from 2015 to 2024 in Germany

To gain more insight both ACF and PACF are plotted in fig. 4.2a. There, the seasonality is more obvious with the presence of regular strong positive and negative spikes. These spikes decays across lags with their pics happening at regular intervals being 1, 24, 48, etc. In fig. 4.2b of the PACF, a remarkable spike at lags 1,2 and 3 is seen which suggests that the 3 past values can have an influence on the current value. Significant correlations between lags 20 and 30 is also spotted suggesting that data points of one specific hour may also be influenced by a day before instances.



(a) ACF Plot

(b) PACF Plot

Figure 4.2: ACF and PACF plots of total loads between 2015 and 2024 in Germany

Based on these informations, different AR processes are fitted, each time including an additional lag to the model while measuring its AIC. The fig. 4.3 is obtained.

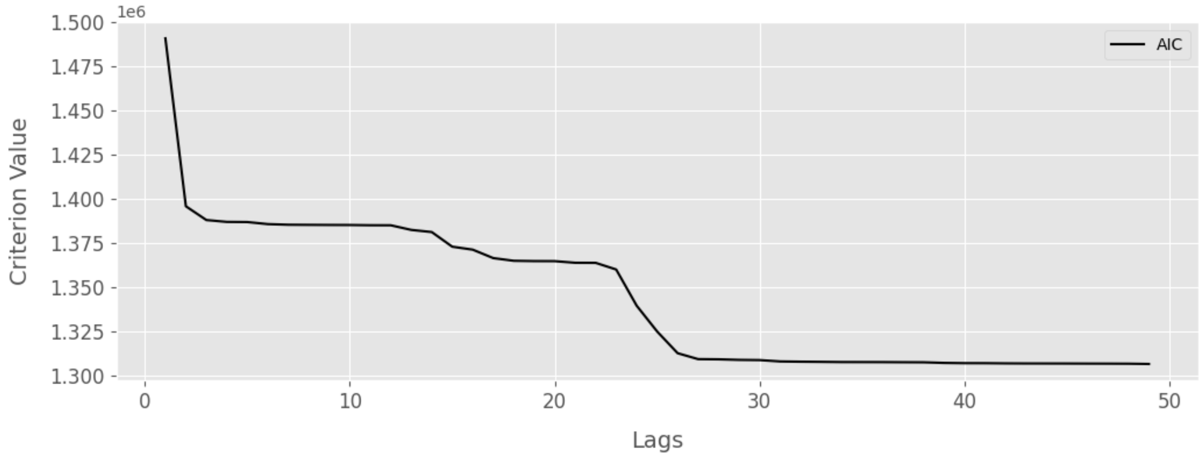


Figure 4.3: AIC values across different AR processes fitted with the total loads data according to different lags

Due to limited computing power, fig. 4.3 stops at lag 50. Extending the lags further will reveal that the AIC starts to increase again at around lag 2000. Therefore, this parametric criterion did not prove to help much in determining an optimal order for the time series. This could be due to the seasonality effect or the quality of the data itself. Either ways, the exact causes of this will not be the main focus. As an alternative $p = 3$ is chosen based on the fig. 4.2b PACF plot due to the highly correlated values it exhibits. To assess the AR(3) model's performance, predictions on the same dataset it was trained on are generated and an MSFE of ≈ 1317 is obtained. By definition, this idea isn't recommended in real world applications. Such approach can lead to biased evaluations because the model would always provide good results for the training data already trained on. This would lead to a lack of conclusion whether our model would have generalized well to new unseen data or not and it would not be know if it actually captured the real process or was only overfitted. After the full in-sample prediction, the forecasts results of the demand were subsetted between 8 p.m and 8 a.m of each day starting from 1st of January 2023 until 15th of March 2024. The fig. 4.4 below is the result:

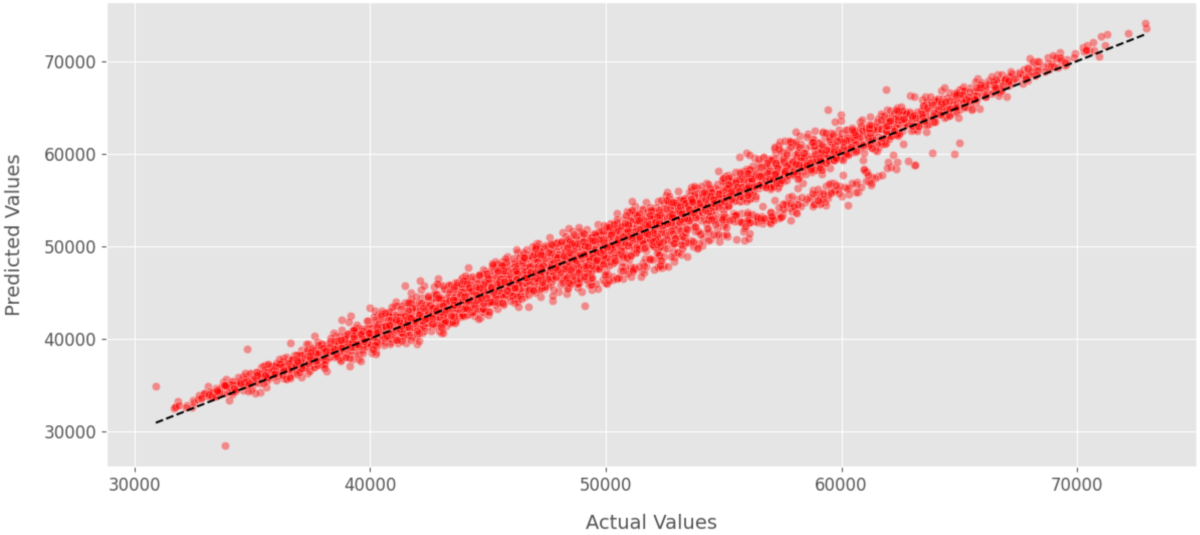


Figure 4.4: Scatter plot of actual load values against predicted ones between January 2023 and March 2024 for the time frame between 8 p.m and 8 a.m

The motivation behind the specific time interval (daily from 8 p.m to 8 a.m) was chosen because that period is known for posing challenges to electricity providers in accurately forecasting the actual load, often leading to overestimation of demand, incurring high costs and causing negative pricing. That is why the behavior of our model was of interest in this frame. Only the part starting from 2023 and not for all N instances were selected since the number of data points are large and cannot be efficiently visualized. Since the MSFE of the subset is very close to the MSFE of the total sample forecast $MSFE_{subset} = 1338$, $MSFE_{subset} \approx MSFE_{total}$, it is assumed to be representative of the training set. More formal tests could be implemented to "statistically" reach that conclusion, but due to time constraints it was not done in this study scope. To briefly explain the plot, if points happens to be exactly at the dashed black line, it means that a perfect forecast was made, if not, then it would be observed how far was that instance from the actual result. Multiple conclusions can be derived based on the scatter plot. The first one is that the forecast instances closely mirror the patterns observed in the actual values, closely forecasting higher loads as the actual loads increase. Hence, data points are aligned along the black dashed line. The absence of data points in the top-left and bottom-right quadrants of the graph indicates that the model does not produce extreme outliers. It does not forecast high loads for low actual loads and vice versa. It is also seen that the model's errors were symmetrically distributed at first, with instances of overestimation occurring as frequently as underestimation. This was the case until reaching 50000. Starting from there, points somehow diverge creating a gap between the overestimated and underestimated loads. Exceeding the value of 62,000, the model demonstrates a systematic bias towards overestimation, consistently predicting values equal or higher than the actual demand. This is also shown by a significant low number of data points in that region under the dashed line separator. Many causes could be behind this resulting divergence and seasonality might be one of them. For the moment, it is not the main focus to understand it but it would certainly constitute an interesting research for a future study. Incorporating other variables to our AR model is the next step as it is indicated by the next section.

4.2 AR Model with Additional Predictors

The basic autoregressive structure is now enhanced to include other predictors already described in table 2.2. In this manner, the objective is to search for the most promising ones. To achieve this, one variable at a time is incorporated into the AR(3) and its resulting MSFE is assessed relative to the baseline model. All features, except for holidays, are lagged by one hour because they represent information used to predict the loads of one hour ahead, rather than the current hour load. Based on these parameters, a list of forecast errors for each variable is obtained and summarized in table 4.1 below.

The range of the new losses vary between 1300 and 1317 where the latter is already the baseline loss. The feature that have led to the most error drop in comparison to the basic AR was the fossil_output. Solar_forecast, hydro_output and temperature show potential to make the model better since they also led to a lower estimation error. The rest either did not drop the loss much or did not even change it like the case with the net_trade. If the model is rebuilt using all features, the error measure will stand at 1207 which is lower than the loss from any individual variable. A good possibility here can be that the model have overfitted to the data, learning even the noise rather than the underlying real

Predictor	MSFE
fossil_output_lagged	1300
solar_forecast_lagged	1310
temp_lagged	1311
hydro_output_lagged	1313
holiday	1315
biomass_output_lagged	1315
other_output_lagged	1315
co2_prices_lagged	1315
wind_forecast_lagged	1316
gas_prices_lagged	1316
geothermal_output_lagged	1317
net_trade_lagged	1317

Table 4.1: Sorted Forecast Errors by Lagged Features Fitted individually with an AR(3)

patterns. It is true that combining variables can sometimes enhance performance due to the interaction effects (Kuhn and Johnson, 2019) however, even if it was the case now, still including all variables without a proper method for feature selection might also have allowed the model to learn the random effects. To determine which features to specifically include, two methods will be considered. The first one is the AIC-based feature selection and second one is the forward stepwise model selection. The one that leads to the subset with better performance will be selected for the rest of the study.

4.2.1 AIC-based feature selection

The AIC values were not included in table 4.1 because they were large numbers reaching 6 figures, making them hard to read. However, if calculated using the same fitted variables and sorted in decreasing order, the criterion scores would also decrease monotonically, mirroring the pattern of the MSFE results. The first 7 variables that result in the lowest AIC, which also correspond to the lowest MSFE, will be chosen. These variables range from fossil_output to other_output according to table 4.1. The number of features 7 was arbitrarily chosen, and other numbers might yield better or worse results. However, this is beyond the scope of the current study and can be worth pursuing. The selected subset yielded a forecast error of 1229 which lies between the scores of individual fitted variables ($\approx$ 1300) and the score for fitting all variables ($\approx$ 1207). The lower error compared to individual predictors can be attributed to the interaction effect (Kuhn and Johnson, 2019). This means that a subset of variables can collectively have more predictive power than when considered individually. Nevertheless, the higher error compared to the full model might indicate a trade-off between model complexity and overfitting. In this case by including the most relevant variables, the model might generalize better to unseen data at the expense of higher bias.

4.2.2 Forward Stepwise Selection

When selecting a set of also 7 variables like before but using forward stepwise selection, similarities and differences compared to the AIC method are observed. Features such as fossil_output, temperature, solar_forecast, hydro_output, and other_output remain consistent between both methods. However, holiday and biomass_output are now replaced by co2_prices and geothermal_output in the new set of features. This switch have lead to obtaining an MSFE of 1209 which is actually lower than the AIC method ($\approx$ 1229) but almost the same as including all variables ($\approx$ 1207). This is a proof that selecting a certain combination of features can lead to competitive performance compared to using all variables and that relying solely on individual performances for selection may not result in an optimal subset, as it overlooks potential interactions among variables. This being clarified, the variables obtained from the forward stepwise selection method will be used for the next section.

4.3 Predictive Model using Deseasonalized Data

Based on the previous analysis shown in fig. 4.1, it was descriptively evident that the electricity load had a strong seasonality on a yearly scale (frequency equaling to 12 months). It is possible to use other statistical tests to confirm such claim, but for the scope of this study, descriptive analysis was the basis. The deseasonalization process was only performed on a scale of 8760 hours, corresponding to the number of hours in a year, to effectively capture and remove annual repeating patterns. Substracting this seasonal component from our original data, the variation without the periodic effects is obtained. Not only the autoregressive structure was deseasoned, but also the added new predictors. All of them are numerical and are assumed to be seasonal. For some cases like the temperature data, the periodic effect is logically incorporated since thermal readings changes with each "weather" season. For the rest, even if they don't have any periodicity, the seasonal component will be just zero. All variables are then fitted again and tested.

Now judging by the error score, this step led to an improvement in model performance. The MSFE dropped from 1209 to almost the half reaching 682. Further detrending can also offer less error score, but for our electricity load study the main focus is only on the seasonality component. The seasonal part is then added back to the forecast and they are plotted again according to the same time interval choice for fig. 4.4

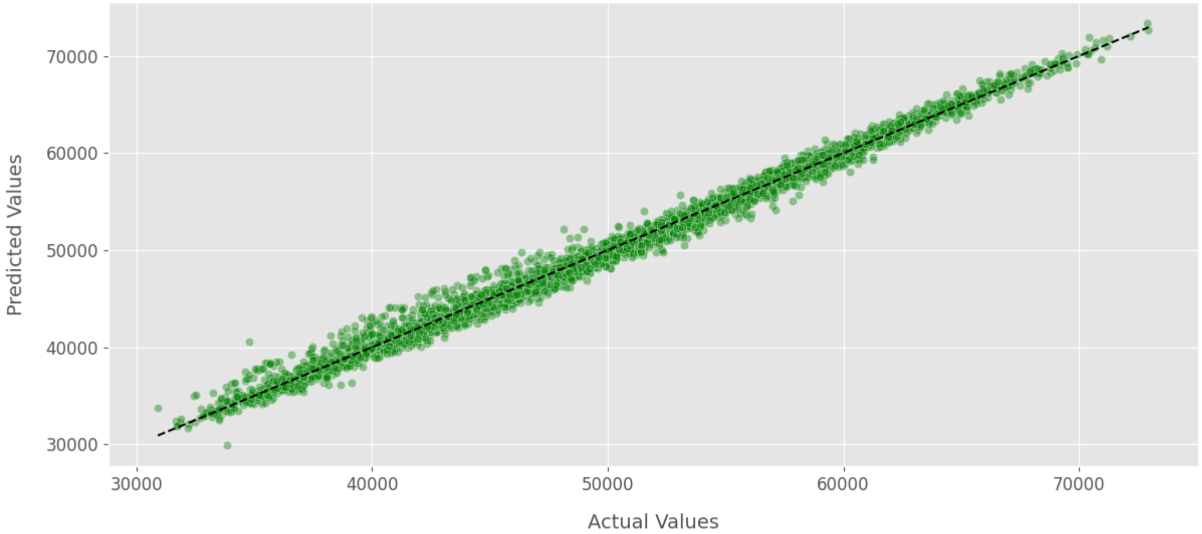


Figure 4.5: Scatter plot of actual loads values against predicted deseasoned ones between January 2023 and March 2024 for the time frame between 8 p.m and 8 a.m

The systematic bias introduced in fig. 4.4 is now eliminated and forecasted instances are more symmetrical in fig. 4.5 even after the value of 50000, which earlier showed a clear divergence. Additionally, the frequent over and underestimation always exists but they are more centered and varies less from the real value after deseasonalization. This can also be an effect from the newly external predictors added to our model, but in either case, the elimination of the periodic fluctuation showed even more added value than all previous steps and proved importance in electricity forecast.

5 Conclusion

Optimising electricity production by forecasting its demand was the main objective that motivated this research. This study takes place in the German market and seeks to reduce operational costs for German transmission system operators. Hence, having a model that closely approximates real consumption would al low to make informed decisions about electricity generation. The principal question that was started with was whether such loads could be forecasted over time or not, whether they could be supplemented by other variables and finally is it better to keep them seasonal or not. Eventually, the discovery was that electricity consumption can actually be predicted over time and by using time series models it was possible to forecast one hour ahead loads. Extending them to include multiple other predictors was a reinforcement and had led to more accurate predictions even after deseasonalization. From an initial loss of 1317 to a final loss of 682, the models performance had almost a 50% improvement. This reduction in error was attributed to several key steps implemented during this project. Most important one was selecting a robust subset of all available features that have led to a more reliable forecasting results. In this case it was the fossil, hydro, geothermal, other energy output, temperature data, solar forecasts and co_2 prices data that showed promising potential. Although the data was not perfectly clean due to missing instances and values in a non-hourly frequency, with basic preprocessing the MSFE reached 682 after deseasoning them. Therefore it is believed that the model could have provided lower losses if the data was more thoroughly processed. Throughout the study, fixed settings and assumptions were maintained. By holding these constant, it was clear how the errors fluctuated and the rationality behind its variation as explained in each section. With different parameters, the model might have yielded different characteristics but the core principle remains the same. That is selecting a robust subset of features minimizes noisy data and that eliminating seasonality can be fundamental to enhancing the time series performance. Future work could explore the integration of more complex interpolations, more comprehensive autoregressive processes and even additional assumptions to further enhance predictive capabilities. This study followed one of many possible approaches, and while it had its limitations, it still proved to be effective.

Bibliography

H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 1974.

George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, and Greta M. Ljung. *Time Series Analysis: Forecasting and Control*. Wiley, 5th edition, 2015.

Peter J Brockwell and Richard A Davis. *Introduction to Time Series and Forecasting*. Springer, 3rd edition, 2016.

Georges Casella. *Statistical Inference*. Duxbury/Thomson Learning, second edition, 2002.

J Durbin. The fitting of time-series models. *Review of the International Statistical Institute*, 28, 1960.

EEX. European energy exchange. `https://www.eex.com/en/`, 2002. Accessed: May 16, 2024.

Walter Enders. *Applied Econometric Time Series*. Wiley, 2 edition, 2004.

ENTSO-e. Entso-e. `https://www.entsoe.eu/`, 2008. Accessed: May 14, 2024.

Gidon Eshel. The yule walker equations for the ar coefficients.

Arthur S. Goldberger. *Classical Linear Regression*. John Wiley & Sons, New York, 1964.

Guido van Rossum. Python software foundation. `https://www.python.org/`, 1989. Accessed: May 12, 2024.

Rob J Hyndman and George Athanasopoulos. *Forecasting Principles and practice*. Otexts, 2 edition, 2013.

Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning*. Springer, 2013.

JetBrains. Pycharm community edition. `https://www.jetbrains.com/pycharm/`, 2010. Accessed: May 12, 2024.

Gebhard Kirchgässner and Jürgen Wolters. *Introduction to Modern Time Series Analysis*. Springer, 2007.

Max Kuhn and Kjell Johnson. *Feature Engineering and Selection: A Practical Approach for Predictive Models*. CRC Press, 2019.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2023. URL `https://www.R-project.org/`.