NOVA IMS
Information Management School

MAA

Mestrado em Métodos Analíticos Avançados
Master Program in Advanced Analytics

Time Series Electricity Price Forecast on the German Day-ahead Market

Steffen Maximilian Hillmann

Dissertation presented as partial requirement for obtaining the Master's degree in Advanced Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

TIME SERIES ELECTRICITY PRICE FORECAST ON THE GERMAN DAY-AHEAD MARKET

by

Steffen Maximilian Hillmann

Dissertation presented as partial requirement for obtaining the Master's degree in Advanced Analytics

Advisor / Co Advisor: Bruno Damasio
Co Advisor*:* /

January 2022

Abstract

Due to the liberalization of the European energy market, electricity prices are now determined based on contracts on regulated markets like any other commodity. They are mainly driven by supply and demand forces. In a rather complex and competitive market where prices are characterized by high volatility and depend on various factors, an accurate forecast of the electricity spot price is a significant source of risk for many participants and, therefore, an essential aspect of effective risk management. The increasing dependence on renewable energy sources and their dependence on weather contribute to the growing importance of electricity price forecasting (EPF). Accurate forecasts provide every market participant with important information for planning bidding strategies to minimize risks and maximize profits and utilities. Extensive research has therefore been conducted in recent decades to develop methods for short-term price forecasting. This research aims to conduct a comparative study to investigate the forecasting performance by using and comparing different time series forecasting methods, including traditional statistical, machine learning, and deep learning models, for the German electricity market. Deep-learning models are gaining interest among researchers nowadays and are expected to perform better than models based on static methods. The models are used to produce a one-step forecast with hourly data of the German electricity market for the whole year 2020. Thus the thesis aims to answer the following research questions: (1) to what extent can deep learning models handle non-stationary time series better than statistical time series methods and machine learning models, and how reliable are the day-a-head electricity price forecast in practice? and (2) does the production of renewable energies for individual hours contain helpful predictive information for EPF that helps market participants to improve their bidding strategies and control risk? Final analyses included data from hourly EPEX Spot electricity prices for Germany from January 2016 until December 2020, as well as historical load and generation data from ENTSO-E and historical weather data from OpenWeatherMap. The empirical out-of-sample results show that deep learning models perform better than statistical and machine learning models. Especially the multivariate GRU outperformed other deep learning models not only in validation accuracy and prediction consistency, due to its memory about the previous time steps.

Keywords– Short-term electricity price forecasting, machine learning, neural network, deep learning, German electricity market.

Table of Content

List of Tables

List of Figures

List of Abbreviations

ACF	Autocorrelation Function
ANN	Artificial Neural Network
AR	Autoregressive Model
ARIMA	Autoregressive Integrated Moving Average Model
ARMA	Autoregressive Moving Average Model
CNN	Convolutional Neural Network
DM	Diebold-Mariano
EPEX	European Power Exchange
EPF	Electricity Price Forecasting
FFNN	Feed-Forward Neural Network
GRU	Gated Recurrent Unit
LSTM	Long-Short Term Memory
MA	Moving Average
MAE	Mean absolute error
MAPE	Mean Absolute Percentage Error
MCP	Market Clearing Price
MLP	Multi-Layer Perceptron
MWh	Megawatt hour
PACF	Partial Autocorrelation Function
ReLu	Rectified Linear Unit
RMSE	Root mean squared error
RNN	Recurrent neural network
SARIMA	Seasonal Autoregressive Integrated Moving Average
Tanh	Hyperbolic tangent

1. Introduction

Traditionally the electricity sector was dominated by government-controlled and government-regulated utilities whose electricity prices were primarily based on production costs, resulting in relative price stability as competitive pressure was low or non-existent to reduce production costs (Klitgaard and Reddy, 2000). The course of deregulation and the introduction of competitive markets in recent decades have led to a liberalization of the electricity sector in many countries, resulting in greater transparency, accountability, and efficiency (Weron and Chichester, 2006). The restructuring of the traditionally monopolistic and vertically integrated, centralized electricity sector has allowed electricity prices to be traded by market forces of supply and demand (Girish et al., 2013). The driving force behind these power sector reforms was that competition can result from inefficient use of resources, resulting in cheaper yet more reliable energy supply to the economy of most nations (Nagayama and Kashiwagi, 2007). Following the early attempts of Latin America, Britain, Australia, California, and Scandinavian, many countries have started deregulating and restructuring their electricity sector (Gountis and Bakirtzis, 2004). It is a fact that electricity is an extraordinary commodity, as it differs from other commodities because it is economically non-storable. Thus the stability of the electricity system necessitates a constant real-time balance between generation and consumption to keep the system in equilibrium (Shahidehpour et al., 2002). This leads us to the various factors that influence both the demand and, consequently, the price of electricity. Weather conditions such as temperature, wind speed, precipitation, coupled with variations in any end-user demand such as the intensity of business and daily activities, principally affect the demand and supply (Girish and Vijayalakshmi, 2013).

Consequently, these individual characteristics lead to price dynamics not witnessed in any other market, with seasonal fluctuations and sudden short-lasting and generally unforeseen price spikes (Weron, 2014). According to Bunn as well as Weron and Chichester, EPF have become an elementary business decision-making tool for companies (2004, 2006). Accurate forecasting can reduce the high costs of over-and under-contracting, leading to an oversupply and thus higher costs (Gooijer and Hyndman, 2006), inadequate supply, and thus more expensive supplementary services (Ruiz and Gross, 2008), respectively.

Before deregulation, there was little need to accurately forecast electricity prices, as electricity was not an easily tradable commodity and was supplied by regulated utilities. With deregulation, this changed, and multiple decentralized trading opportunities were established (Irena, 2020). These platforms offer market participants the opportunity to efficiently match supply and demand by optimizing against market prices and seeking arbitrage opportunities to stabilize their cash flows (Scharff and Amelin, 2016). These platforms perform an essential function in ensuring the stability of the electricity system.

Besides fixed contracts and derivates from financial markets, electricity prices are highly volatile, and market participants are not protected against high and low prices. Even though the volatility risk exists in almost any market, it is higher in electricity markets than in other markets. Next to the weather, seasonality, and non-storable electricity, several external factors significantly affect the supply and demand balance on the market and make it difficult to predict the electricity price. These factors include, among others, hydro generation production, prices of fuel or oil, or any kind of unexpected physical problems of power plants (IEA, 2021). Since economic and demographic factors are challenging to forecast, the uncertainty in EPF is more significant than in electricity load forecasting (Nti et al., 2019). According to Ku (2002), price forecasting is much more complicated as the electricity price has complex and non-linear relationships and requires both supply and demand forecasts (Figure 1).

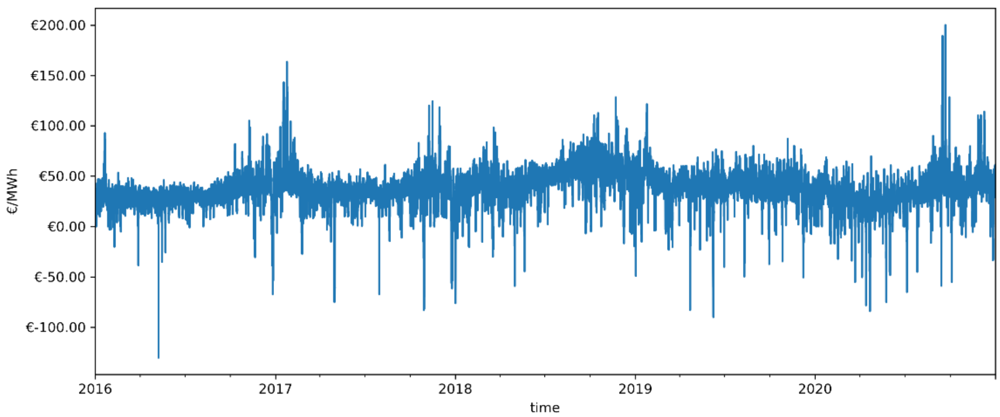


Figure 1. Profile of the Electricity Price Variation for the Whole Period from 2016 to 2020

Due to the flexible mathematical structure, artificial neural networks (ANNs) can capture complex and non-linear relationships between the input and output time series which emphasizes learning consecutive layers of progressively complex and meaningful representations (Chollet, 2018). Due to the highly volatile and uncertain nature of the electricity price, accurate price forecasts are crucial to developing effective bidding strategies. Thus the purpose of this study is to investigate the use of deep learning to forecast hourly electricity prices for the next day of the German electricity market for a better decision-making process.

Given this, the thesis aims to answer the following research questions:

1. To what extent can deep learning models handle non-stationary time series better than statistical time series methods and machine learning models, and how reliable are the day-a-head electricity price forecast in practice?
2. Does the production of renewable energies for individual hours contain helpful predictive information for EPF that helps market participants to improve their bidding strategies and control risk?

The remainder of the paper is organized as follows. Chapter 2 performs a literature review of the German electricity market structure, the current state of EPF, and used variables. Chapter 3 is devoted to a short description of the theoretical concept of time series. Chapter 4 introduces the different forecasting approaches and describes the methodology used to develop and evaluate the models. Chapter 5 explains how the data were collected and pre-processed and provides an exploratory analysis of some characteristics. Chapter 6 presents the performance of the final models, and the experimental results are provided. Finally, Chapter 7 concludes this paper presenting some policy recommendations and exposes the main limitations and suggests improvements for future research.

2. Literature Review

This chapter provides an overview of relevant literature. In each chapter, a targeted literature review is conducted that relates to the subject covered in that chapter.

2.1. German Electricity Market Structure

We use price data from the German electricity market. The Central European spot market for energy, the European Power Exchange (EPEX) in Paris, trades short-term deliverable power quantities and non-long-term power supply agreements. EPEX is the most important and, in terms of volume, the largest trading platform for electricity prices in Europe. Various regulatory computations are based on the EPEX day-ahead price, such as the feed-in tariffs for renewable energies, which is also significant from a political perspective (Viviani et al., 2021). The way electricity is traded between market participants is different from most other commodity or financial markets. The electricity spot market is geared towards trading, clearing, and settlement in the day-ahead market with its once-per-day price auction and an intraday trading session in which only position adjustments can be made. The aim is to compensate for deviations, which result from positions in day-ahead contracts and unexpected shifts in demand (Gianfreda et al., 2016). An overview of the day-ahead market, on which agents aggregate their bids and offer for the delivery of electricity before the market closes, can be seen in Figure 2.

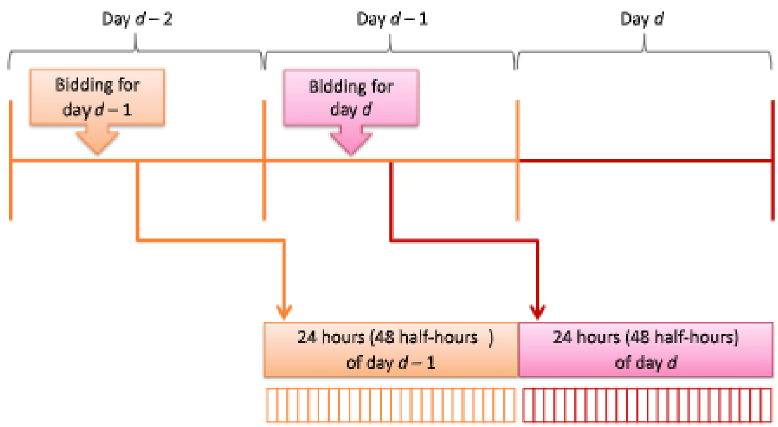
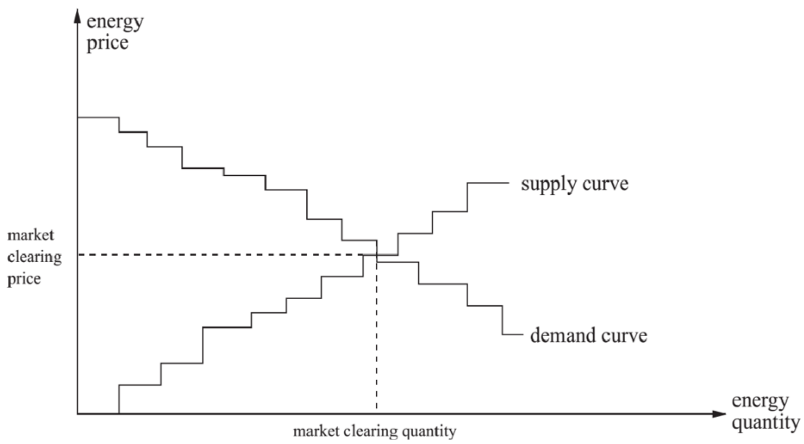


Figure 2. The electricity trading schedule for the day-ahead market in Germany (Weron, 2014).

Electricity demand is subject to fluctuations of temperature, season, and consumption patterns, leading to the periodic nature of electricity prices (Viviani et al., 2021). Price elasticity of demand is inelastic in the short term, as consumers have limited time to react and have few options to react to price changes compared to the long run (Lumen, 2021). The actual task of the electricity market is to match supply and demand to set the equilibrium price and quantity (Zhirnov, 1999). The equilibrium is the only point where the quantity demanded is equal to the quantity supplied and thus the price balances supply and demand schedules (OpenStax, 2016). The point at which the two curves intersect determines the market clearing price (MCP) and the quantity traded (Soloviova and Vargiolu, 2020). Only offers below or equal to the MCP, and only bids above or equal to the MCP were cleared and exchanged at the resulting price (Shah and Chatterjee, 2020). The remaining bids and offers were rejected or remained in the order book until settled or expired (Weron 2014).



The market clearing price and quantity is the point where the supply matches the demand.

Figure 3. The market equilibrium point

Although most price bids are favorable, even negative price bids are allowed, resulting in negative prices if demand is low or renewable generation is relatively high (Cutler et al., 2011; Keles et al., 2012).

The supply on an electricity market is reflected in the merit-order-curve, which has a significant role in price formation, as it reflects the marginal costs of electricity generation (Hagemann and Weber, 2013). The merit order is a way of ranking available energy sources, based on ascending order of price, with renewable energy sources such as hydro and wind power having almost zero marginal cost and thus at the lower end of the merit order curve (Sensfuß et al., 2008). Costs increase according to the energy source, with coal, oil, and gas being the most expensive. Baseload power plants such as nuclear and coal-fired plants generally serve as price-determining technologies when demand is low (Viviani et al., 2021). These plants are inflexible since they are not technically capable of operating in a variable mode due to their high marginal costs (Ueckerdt and Kempener, 2015). However, if demand is high, prices are determined by expensive peak power plants such as gas and oil-fired plants (Viviani et al., 2021; Sensfuß et al., 2008). These power plants are operational flexible but have high marginal costs. With a higher share of renewable energy, especially from variable sources, supply and demand will be matched in a much more coordinated and flexible way (Ueckerdt and Kempener, 2015). Difficulties arise for inflexible plants that are supposed to run continuously when more renewable energy is fed into the grid (Viviani et al., 2021). That is because start-up and ramp-up operations are rare but time-consuming for inflexible power plants (Gonzalez et al., 2018). Thus, they must accept negative margins to continuously generate electricity, which has a lower effect on electricity prices (Viviani et al., 2021).

2.2. Forecasting horizons

Based on an electricity load forecast, a prediction can be made about the amount of electricity consumed at a given time. The aim of these forecasts is efficient economic and quality planning. Proper forecasting ensures the economic viability of the service and the security of the network. Energy forecasts can be made at different levels of time interval resolution. The scope of forecasts generally depends on the reliable data available and the objective of the forecast. In general, forecasts fall broadly into three different categories: short-, medium-, and long-term EPFs, with no consensus in the literature on what the thresholds should be. Short-term EPF is typically conducted over periods ranging from minutes to hours or days in advance (Hernández et al., 2014) and is of utmost importance in facilitating electricity market operations for next-day supply planning and demand management (Khan and Jayaweera, 2018). Medium-term EPF deals with forecasting horizons from a few days to a few months or even years ahead (Suganthi and Samuel, 2012). Such forecasts are generally preferred for balance sheet calculations, revenue assessments, unit maintenance scheduling, energy trading (Al-Alawi and Islam, 1996). For long-term EPF, the forecast horizon extends to years, quarters, or even decades ahead, provides deeper insight for policymakers, and mainly focuses on investment profitability analysis and planning (Al-Alawi and Islam, 1996). As Ventosa et al. (2005) note, capacity investment decisions are the essential variables in this context. Depending on the model identified, different methods can be applied. Similar tools and techniques can be used for short- and medium-term forecasts. For long-term forecasts, a completely different approach is generally taken.

2.3. Input variables

This chapter introduces some literature reviews on the used variables. Factors that influence the price of electricity can be classified as follows: fundamental, operational, strategic, and historical factors (Figure 28, Appendix). According to Cuaresma et al. (2004), the spot price of each hour reflects the actual market value, and all determinants influencing the electricity price are included in the MCP. Thus they only use the electricity price of a spot market for modeling and forecasting. Many other researchers include exogenous variables mentioned above in modeling and forecasting the electricity price (Weron, 2006; Aggarwal et al., 2009). Thus, we address some essential factors that influence the price of electricity. For this purpose, we mention numerous literature reports on historical values, calendars, and weather characteristics.

2.4. Historical consumption and load variables

Since the electricity price is highly correlated with consumption and production, historical demand and load data are considered critical to the model's performance that several studies are based solely on past load values (Taylor and McSharry, 2007). Hourly electricity consumption reflects, among others, people's lifestyles and mechanical systems usage (McMenamin and Monforte, 2000). Since load data can be modeled with high accuracy, we assume actual demand as one of the explanatory variables. Another basic set of explanatory factors is the supply side, which includes fuel, and oil prices, hydro flows, and renewable energy sources. During periods of high demand, load factors in surrounding areas can also significantly affect local prices, reflecting the high prices of imported energy and high opportunity costs (McMenamin and Monforte, 2000).

2.4.1. Calendar variables

The literature states that calendar information has an essential part in the structure of electricity trading. There are different ways to integrate calendar variables into the time series model. Hinman and Hickey (2009) subdivided the data for regression models by hours of the day. Time of day is one of the most important factors to consider, as the price of electricity varies depending on the time of consumption. Ramentol et al. (2020) have split the data according to weekday/weekend profiles. That was further explored by Alvaret et al. (2007), who were able to identify different clusters by weekday/weekend in the Spanish market. Green et al. (2014) also worked on a similar approach using k-means clustering in the UK, splitting the data by weekend/weekday and summer/winter/shoulder season. Yu et al. (2019) labeled holidays that differ from regular days with a binary indicator. This indicator was also extended to days adjacent to holidays to indicate a typical working day affected by a holiday (Fan and Hyndman, 2011).

2.4.2. Weather variables

Several weather conditions can adversely affect the operation of power plants. Because of the high correlation between the meteorological conditions and the electricity demands (and thus price), Hong and Shahidehpour claim that these variables alone can help explain more than 70% of the variation (2015). Additionaly, Dehalwar et al. (2016) found that the price of electricity increases when the temperature decreases due to heating demand. Similarly, if the temperature increases, the electricity price also rises due to cooling demand. Energy consumption in an urban area is more susceptible to changes in weather conditions, so these variables need to be specific to the region, as climate conditions vary between countries (Fidalgo and Matos, 2007). Therefore, considering meteorological data is essential for accurately predicting the electricity price (Dehalwar et al., 2016).

2.5. Review of Related Works

To the best of our knowledge, there are few books but several reviews and survey articles that address machine learning methods in EPF. Saini et al. (2016) used linear regression to model the electricity price, while Botnen (2017) uses multiple regression. Time series models like ARIMA have also been applied for EPF of different electricity markets (Cuaresma et al., 2004; Bowden and Payne, 2008). The main disadvantage of time series models is "that they are usually based on the hypothesis of stationarity; however, the price series violate this assumption" (Amjady, 2006, pp. 116). To solve this problem, a different type of time series model known as autoregressive conditional heterocyclic models was developed, which assume a time-dependent variance (Gonzalez et al., 2005; Garcia et al., 2003). Whereas the application of these models has been successful in other commodity markets, they face difficulties when applied to electricity prices (Gonzalez et al., 2005). Shahidehpour et al. (2002) use neuronal networks to model the electricity price, while Weron and Chichester concentrates on practical applications of statistical methods for EPF (2006).

Although the results obtained using time series techniques can be good, most of these techniques are linear techniques and, therefore, unable to adequately capture the complex, non-linear behavior of the target signal (Amjady and Hemmati, 2006). Therefore, large unexpected prediction errors can occur with the time series methods. According to Monteiro et al. (2016), prices can be predicted with relatively high accuracy when only calendar variables are included, and an improvement was found when additional variables were included. Moreover, Azevedo and Vale (2006) estimate the electricity price only based on historical values, with limited coverage of hourly trends. According to existing literature and as a contribution to current research, a range of real-time information can be relevant in addition to historical market prices when modeling hourly prices. In addition to the historical market price, Harasheh (2006) found significant effects of load and feed-in from renewable energy sources, while other authors observed that sudden outages, grid congestion, and maintenance could have a significant impact on the price of electricity (Girish et al., 2013; Weron, 2006; Bunn and Martoccia, 2005). These factors, in turn, can improve the models even to detect price peaks, which would not be possible with earlier historical prices alone. As in the previous work, Keles et al. use ANN for EPF and made clear that the performance of the machine learning approach is higher than that of a competing time series model such as SARIMA since these forecasting methods are usually only effective in areas where the frequency of the data is low, such as weekly patterns (2016). Computational intelligence is required that can track the non-linear behaviors of hourly load and especially price signals (Amjady and Hemmati, 2006). To compare the performance of different models, a hybrid approach based on fuzzy neural networks was applied to the Spanish market (Amjady, 2006). Similar neuronal networks with backpropagation error mechanisms have been applied by Gao et al. (2000) and Hu et al. (2004) for California and the Chinese electricity market, respectively.

Aggarwal et al. (2009) examined many papers on EPF performance. They concluded that "there is no systematic evidence of the superiority of one model over the others consistently due to the large differences in price trends (...) in different electricity markets" (Aggarwal et al. 2009, pp. 333-358). However, the key advantage of ANNs in forecasting problems is that they can infer hidden relationships in the data (Amjady and Hemmati, 2006). It has become apparent that deep learning applications outperform traditional machine learning methods in a variety of areas. The results and observations in the study by Lago et al. (2018) show that deep feed-forward, GRU (gated recurrent unit), and LSTM (long short-term memory) networks perform best on EPF in the Belgian market.

3. Theoretical Background on Time Series

This chapter introduces some basic ideas of time series analysis. The definitions of time series and their differences from timestamp data and other types of data are discussed. Most of the topics covered in this chapter are discussed in more detail in later study sections.

3.1. What is a Time Series?

A time series is a set of values observed sequentially through time, typically consisting of successive observations on quantifiable variables (Cochrane, 2005). Time series includes historical data and usually finds their application in econometrics, finance, and mainly in any domain of applied economics, engineering and social sciences in general (Hamilton, 1994). According to several papers, a time series is mathematically denoted as a set of observations, where t represents the time elapsed, and X refers to the value (Hipel and McLead, 1994; Cochrane, 2005), written as:

$$\{X_t\}\ or\ \{X_1, X_2, \cdots, X_t, \cdots, X_T\}\ or\ X_t, \qquad where\ t = 1, 2, \ldots, T \tag{1}$$

A common assumption is that a time series has inherently non-deterministic characteristics and the time series variables X_t are independent and identically normally distributed, i.e., we cannot predict with certainty what will happen in the future (Adhikari and Agrawal, 2013). However, time series are not strictly independent and identically distributed, since they follow some regular pattern (Cochrane, 2005). If the observations of the X-values are precisely determined by a mathematical formula or any other value, the series is said to be deterministic, and one can directly determine the trend from the mathematical equation. The mathematical expression that describes the probability distribution of a time series is called a statistical or stochastic process (Hipel and McLead, 1994).

For example, if the electricity price in hour h is exceptionally high, one can assume that the electricity price in the following hour will also be high. Thus the time series has a regular pattern, i.e., trend, where a value of the series should be a function of previous values. Values of variables that occur before the current observation are called lag values. (Cochrane, 2005 page). This can be denoted as:

$$X_t = f(X_{t-1}, X_{t-2}, \dots, X_{t-n}) + e_t \tag{2}$$

Where $\boldsymbol{e_t}$ is a random disturbance.

3.2. Components of a Time Series

According to Diggle (1990), a time series, can be decomposed into four main components: trend, cyclical, seasonal, and irregular components. The long-term movement is called trend, which shows the general tendency of a time series to increase, decrease or stagnate during a long period (Cochrane, 2005). Seasonal variations are fluctuations related to the calendar. The cyclical fluctuations are used to describe data fluctuations over periods (Worthington et al., 2001). The duration of a cycle extends over a more extended period and is not tied to seasonal variation. Most of the economic and financial time series exhibit behavior that is repeated over time. The irregular component is the remaining component after the seasonal and trend components have been removed. They are caused by unpredictable influences, resulting from short-term fluctuations which are not systematic, uncontrollable, and in some instances not predictable, e.g., uncharacteristic weather patterns, war, revolution, and other disasters (OECD, 2007). Combining those four components with time causes the formation of a time series and can be of two types (Verma, 2021).

$$\begin{gathered} Multiplicative\ Model: X_t = T_t * S_t * C_t * I_t \\ Additive\ Model: X_t = T_t + S_t + C_t + I_t \end{gathered} \tag{3}$$

Where X_t is a given time series value and is the trend (T), seasonal (S), cyclical (C), and irregular (I) component at time *t*. The multiplicative model is based on the assumption that the time series has exponential growth or decrement with time and the components of a time series are not necessarily independent; whereas in the additive model, it is assumed that the increasing or decreasing pattern of the time series is similar throughout the series and the components are independent of each other (Verma, 2021; Cochrane, 2005).

3.3. Time Series Analysis

Time series analysis is a statistical method suitable for analyzing data to extract meaningful statistics and other relevant data characteristics and to understand the underlying natural process, the pattern of change over time, such as autocorrelation, trend or seasonal variation, or assessing the impact of measures (Velicer and Fava, 2003). Another purpose of time series models is to forecast the future behavior of variables based on previously observed time series values (Brockwell and Davis, 2002). The main goal of a time series model is the same as for other types of predictive models, which is to create a model such that the error resulting from the difference between the actual values and the expected values is as small as possible.

According to Robert and David (2010), time series models differ from other conventional models because time series models use lag values of the target variable as predictor variables.

4. Methodology

In this section, we first present an overview of the forecasting methods used for time series forecasting. A set of metrics is needed to compare how different models perform to assess how well various implementations on the prediction problem perform. Therefore, we present the theoretical foundations of evaluating the model performance and the Diebold-Mariano (DM) test to provide statements on which models statistically outperform each other. As mentioned in the introduction, the main objective of this paper is to propose a modeling framework for EPF. Thus, we will give a broad overview of some key concepts required to understand the stochastic model and different neural networks.

4.1. Overview of Forecasting Approaches

The vast majority of papers focused on EPF started to increase dramatically in the last two decades (Weron, 2014). Increased competition in the market has led to the growing importance of probabilistic forecasting, resulting in a wide variety of methods and models over the last twenty years (Nowotarski and Weron, 2018). An overview of the existing literature on EPF was given by Weron, who categorized electricity price models into five main categories: multi-agent, fundamental, reduced-form, statistical, and computational intelligence models (2014). These groups can further be classified according to different criteria such as dynamic/static, linear/non-linear, parametric/non-parametric, and deterministic/stochastic (Aggarwal et al., 2009).

Due to their performance and our focus on predicting day-ahead prices, we will mainly focus our forecasting methods on the last two categories, the statistical and computational intelligence models. Statistical methods forecast the current price using a mathematical combination of the past prices and past or current values of exogenous variables, typically consumption, demand, or weather variables (Soytas and Sari, 2020). An important reason for working with statistical models is the physical interpretation of their components, making it possible to understand their behavior (Weron, 2014). However, statistical methods tend to perform poorly if extreme price spikes are present. Janczura et al. emphasize that once the spikes have been identified, they have to be replaced by average values (2013). Different approaches can replace these spikes, like a chosen threshold (Shahidehpour et al., 2002), the mean of neighbor prices (Weron, 2014), or a similar day value, e.g., the median of prices having the same weekday (Bierbrauer et al., 2007). Statistical methods are also criticized for their limited ability to model the non-linear behavior of electricity prices (Weron, 2014). Nevertheless, their performance is comparable to that of their non-linear alternatives. The major strength of computational intelligence methods over other methods is the capability to deal with complexity and non-linearity by creating approaches with learning elements, fuzzy logic, and evolution (Aggarwal et al., 2009; Weron, 2014). Due to this added value, they are ideally suited for short-term price models and forecasts (Harasheh 2016). At the same time, this flexibility is also their major weakness (Weron, 2014).

Various studies have used different models to determine forecasting accuracy, but it can be stated that there is no clear evidence that any particular model is superior to others (Girish et al., 2013; Hong et al., 2014; Aggarwal et al., 2009; Karakatsani and Bunn, 2008). There are advantages and disadvantages for each model. However, it is evident that some models are better suited for short-, medium- or long-term prediction of the electricity price, as seen in the following quote, "reduced-form models are generally not expected to forecast hourly prices accurately, but are expected to recover the main characteristics of electricity spot prices, typically at the daily time scale" (Weron, 2014).

4.2. Naïve Model

Inspection of the electricity price identifies clear calendar patterns, as examined in Chapter 2.4. Strong relations are observed between the exact times of day in consecutive weeks and at similar periods of the year. The importance of these correlations is that price can be predicted with high accuracy when they are included as predictor variables. The naïve method is a straightforward benchmark method for predicting time series data to determine the minimum level that more complex models should be expected to perform (Keles et al., 2012). The naive method uses previous observation directly as the forecast (Brownlee, 2018). We can use the following mathematical expression to represent the method:

$$\hat{y}_t = y_{t-n} \tag{4}$$

Where $\hat{y}$ denotes a point forecast, y denotes an observed value, and the indices t and (t-n) denote two adjacent time points.

4.3. Time Series Forecasting Using Stochastic Models

This chapter will discuss the most essential linear and non-linear stochastic time series models with their different properties and form the background for the coming chapters, where we will examine other models for time series forecasting.

4.3.1. Stationary

Stationarity is a key concept in time series analysis. There are two important forms of stationary: weak and strict stationary. A time series is said to be strictly stationary if the entire joint probability distribution does not change with a time shift (Duda, 2019) while a weak stationarity restricts the mean and variance of the time series to be finite and timeshift-invariant (Chen et al., 2021). From the definition of stationarity, it is clear that our data is non-stationary because it has a trend (changing mean) and seasonality, which means that the covariance function depends on time. This can be confirmed by Figure 4, which shows a clear stationary trend with the highest level of electricity prices around the turn of the year and a lower level in the remaining months.

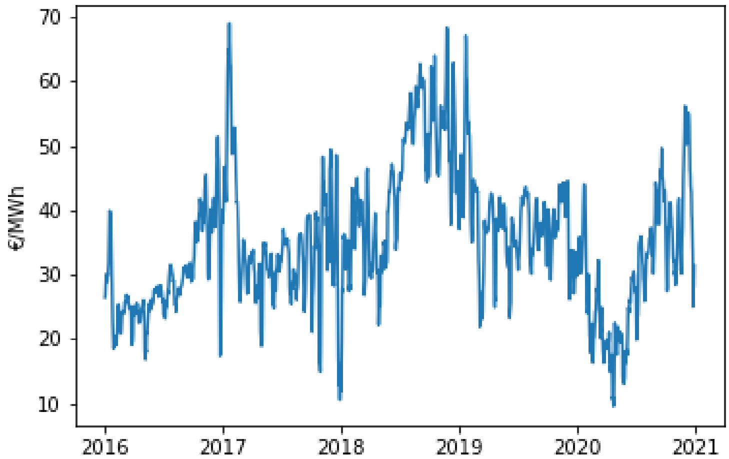


Figure 4. Stationary Detection in an overall Non-Stationary Time series

A visual inspection will almost always be insufficient to confirm stationarity. Therefore, a more accurate statement can be made with the Dickey-Fuller test, a statistical unit root test that determines how strongly a time series is defined by a trend (Dolado et al., 2002). Under the null hypothesis H_0 we have a nonstationary process (with a unit root), and under the the alternative hypothesis H_1 we have a stationary process. This means that if the null hypothesis is rejected we can conclude that the time series has no unit root, meaning it is stationary (Lyocsa et al., 2011).

4.3.2. Autocorrelation and Partial Autocorrelation

The autocorrelation function (ACF) and partial autocorrelation function (PACF) are two functions for analyzing time series data to determine the optimal model parameters. These statistical measures can be used to determine how the observations in a time series relate to each other. The ACF is a measure of the correlation between observations in a time series with its lagged values (Teusch, 2006). The autocorrelation function is defined as follows:

$$\rho_k := \rho_{Y_t, Y_{t-k}} = \frac{Cov(Y_t, Y_{t-k})}{\sqrt{Var(Y_t) Var(Y_{t-k})}} \quad (5)$$

We define $\rho_k = Cov(Y_t, Y_{t-k})$, referred to as the autocovariance at lag k. The sequence $\{\rho_k : k \geq 0\}$ is the ACF.

The PACF is used to measure the correlation between the current observation y_t and y_{t+k}, with the relationships of intermediate observations removed (Brownlee, 2018). The PACF is denoted by

$$\varphi_k = Corr(y_t, y_{t-k} \mid y_{t-1}, \ldots, y_{t-(k-1)}) \quad (6)$$

Figure 5 shows a plot of the ACF and PACF against the time lags from Figure 4.

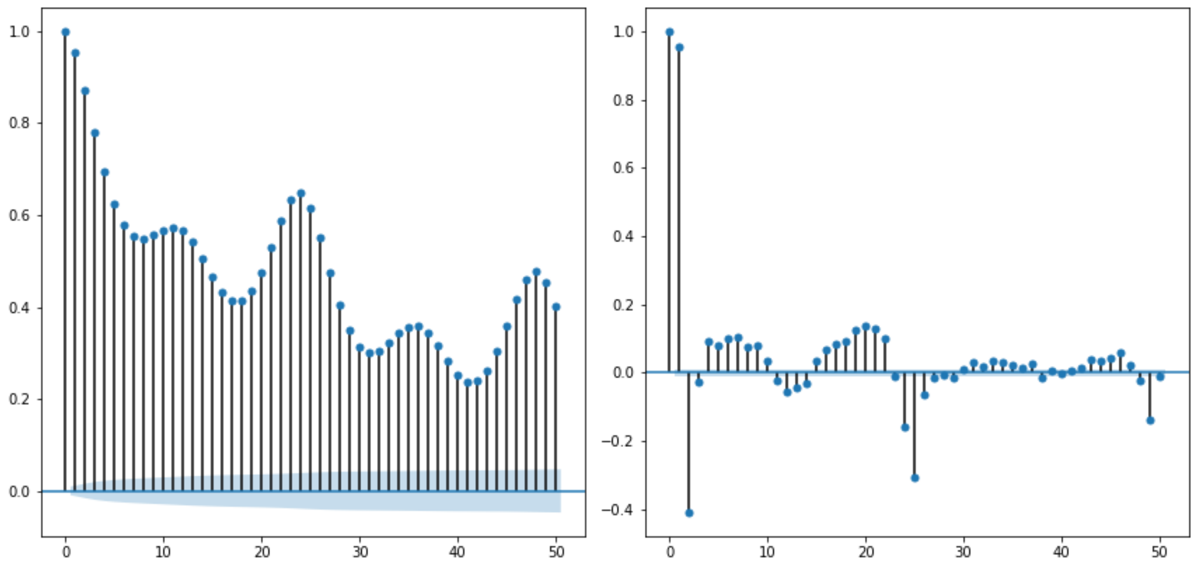


Figure 5. Correlogram and Partial Correlogram

Strong autocorrelations are evident in both plots, but we pay particular attention to the spikes that occur at lags $y_{t-12}, y_{t-24}, y_{t-48}$ and subside after that. This seems to indicate the presence of a seasonal pattern in the time series.

4.3.3. Autoregressive and Moving Average Models

Autoregressive (AR) and Moving Average (MA) models are well-known approaches in statistical time series. AR of order p is commonly denoted as AR(p) and forecasts the value of a time series based on previous values (Chan, 2011). The AR model is calculated using equation 7, where Y_t is the value of observation in period t, c is a constant, $\emptyset$ is a parameter that describes the effect of the past observations ($e.g. Y_{t-1}$ on Y_t), and ε_t is the white noise residual (Dissanayake et al., 2021).

$$Y_t = c + \sum_{i=1}^{p} \rho_i Y_{t-i} + \varepsilon_t \tag{7}$$

Likewise, a pure MA model of order q is commonly denoted as MA(q) and depends only on the lagged forecast error, which "are the errors of the AR models of the respective" lags (Dissanayake et al., 2021, p. 565). The MA model is defined by equation 8, where Y_t is the value of observation in period t, μ is the mean of the time series, q represents the order of the MA model, $\emptyset$ is a parameter that describes the effect of the past observations and ε_t are white noise error terms which has a constant mean of zero, a constant variance and no correlation (Dissanayake et al., 2021).

$$Y_t = \mu + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t \tag{8}$$

4.3.4. Autoregressive Integrated Moving Average

To solve the problems of finding the adequate Parameters for AR and MA, the Auto-Regressive Moving Average model (ARMA) was suggested, which is a simple combination of its lagged time series values (AR) and moving average of lagged forecast errors (MA) and is denoted as ARMA(p, q) (Chan, 2011). Its general formula is expressed in Equation X, where φ_j refers to the AR coefficients and θ_j refers to the MA coefficients (Song and Wang, 2003).

$$Y_t = \alpha + \sum_{i=1}^{p} \rho_i Y_{t-i}\ \varepsilon_t + \mu + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t \tag{9}$$

AR, MA, and ARMA models are used to describe stationary time series and are often used when a complex structure model is inappropriate (Liu et al., 2014). However, stationary time series with time-varying means and variances are commonly seen in socio-economic, and performance tests (Blais 2013, Tsay, 2005), while many economic and financial time series exhibit some form of trending behaviour and show non-stationary behavior due to the presence of deterministic or stochastic trends (Kotzé, 2021). This led Box and Jenkins (1970) to propose the Autoregressive Moving Integrated Moving Average (ARIMA) models (Greene, 2003), which is based on its lags and the lagged forecast errors to include the case of non-stationarity as well (Wu et al., 2020). For ARIMA models, a non-stationary time series is made stationary by using logarithmic transformation, square root transformation, or differencing (Liu et a., 2014). A stationary process is obtained by using differencing

$$\Delta y_t = y_t - y_{t-1} \tag{10}$$

where Δy_t is called the first difference. If a time series becomes stationary, we say that y_t is "integrated of order one," denoted I(1) (Hendry and Nielsen, 2007, p. 175). In general, if it is necessary to make higher differences, we need p differences to produce a stationary time series, denoted I(p), where $\rho \epsilon N$ by definition (Baumohl and Lyocsa, 2009). The resulting model is typically specified by ARIMA(p, d, q), where L defines the number of lags to be used, φ_i are the parameters of the AR part of the model, θ_i are the parameters of the MA average part and ε_t are error terms (Bakar and Rosbi, 2017).

$$\left(1 - \sum_{i=1}^{p} \rho_i L^i\right) Y_t = \left(1 + \sum_{j=1}^{q} \theta_j L^j\right) \varepsilon_t \tag{11}$$

4.3.5. Seasonal Autoregressive Integrated Moving Average

If a time series is influenced by the same effects at regular intervals, it is called a seasonal effect. A variation of ARIMA, the seasonal autoregressive integrated moving average (SARIMA) model, is used to predict seasonal time series (Suhartono, 2011). The model is specified as SARIMA$(p,d,q)(P,D,Q)S$, where p,d,q are the usual ARIMA orders, P, D, Q are the orders of the seasonal ARIMA component and S is the order of seasonality. As shown in section 4.3.2, seasonality is observed when peaks occur at regular intervals, and these can be differentiated away using a seasonal difference. In our case, a peak in the ACF was found at lag $24n$, $n \in$ N, implying daily seasonality for hourly data. Its mathematical formulation is expressed below (Strandberg, 2015).

$$(1 - \sum_{k=1}^{p} \Phi_k (B^s)^k)(1 - B^s)^D Y_t) = (1 - \sum_{l=1}^{q} \Psi_k (B^s)^l) e_t) \tag{12}$$

Although an ARIMA model can capture the linear characteristics of the time series well, it cannot analyze the dataset's complex and non-linear behavior (Tian et al., 2019). Furthermore, ARIMA requires a very exhaust feature search to find the optimal parameters, which was not performed exhaustively. Therefore, we only use ARIMA as a baseline model.

4.4. Time Series Forecasting Using Artificial Neural Networks

ANNs are a popular framework for machine learning that has proved very successful in solving various problems, such as image and speech recognition and fraud detection. They have also been used extensively for forecasting financial data series and at EPF. ANN are mathematical models inspired by the biological structure of the human brain that uses learning algorithms to model complex patterns (Kumar and Sharma, 2014). Unlike conventional techniques for EPF, ANN can approximate any linear and non-linear function arbitrarily accurately using a finite but significant number of neurons and can be applied to a broad scope of application (Lin and Jegelka, 2018).

4.4.1. The Artificial Neural Network Architecture

According to the literature, the most known and widely used type of ANNs in forecasting problems are multi-layer perceptrons (MLPs), which is a feed-forward neuronal network (FFNN) (Zhang, 2003). Depending on the number of layers, they can be further classified as either single-layer or multi-layer. A single-layer perceptron has just two layers of input and output and can only learn linear functions, whereas a multi-layer perceptron contains hidden layers apart from the input and output layer and can also approximate any non-linear functions (Singh and Banerjee, 2019). An MLP „connects multiple layers in a directed graph, which means that the signal path through the nodes only goes in one direction" (Rudrappa, 2021, p. 183). An MLP network with one hidden layer is depicted below:

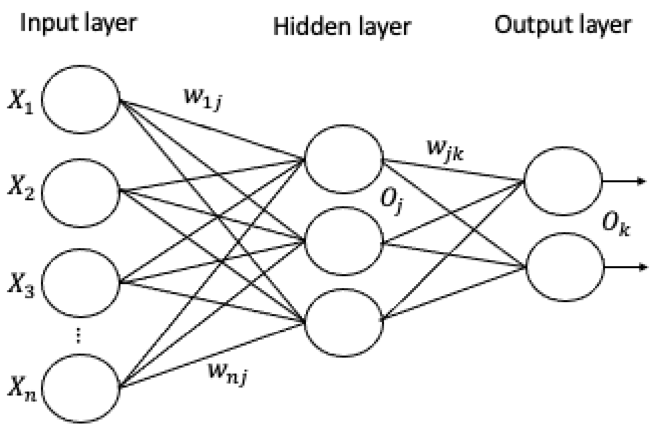


Figure 6. Schematic diagram of a Multilayer Feed-Forward Neural Network (Own representation based on Han and Kamber, 2012)

The input layer is responsible for receiving the input signal to be processed, whereas the output layer performs the required task such as prediction and returns the output data. Any other layer placed between the input and output layer processes the inputs and sends the result to the output layer are referred to as hidden layers (Joshi and Gupta, 2020). Unlike a single-layer network, there is at least one hidden layer between the input and output layers (Sazli, 2006). The goal is to find the optimal size of the layers to make the learning process as efficient as possible (Meller et al., 2020), where the presence of one or more hidden layers allows the network to extract higher-order statistics (Sazli, 2006). The following is the mathematical representation of an FFNN with a single hidden layer (Balkin and Ord, 2000):

$$x_t = \phi_0\left(w_{co} + \sum_h w_{ho}\phi_h\left(w_{ch} + \sum_i w_{ih}x_{t-j_i}\right)\right) \tag{13}$$

Where $\mathrm{w_{ch}}$ denotes the weight for the connections between the constant input and the hidden neurons, w_{co} denotes the weight of the direct connection between the constant input and the output (Balkin and Ord, 2000). Moreover, w_{ho} and w_{ih} denote the weights for the connections between the hidden nodes to the output nodes and input nodes to hidden nodes, respectively. The two functions $\phi_o$and ϕ_h denote the activation functions for the hidden and output layer, respectively (Noor-Ul-Amin, 2010).

The individual neurons are connected by synaptic weights, where the connections have randomly initialized weights that are changed during the learning process (Meller et al., 2020). The training process is done by continuously adjusting the weights, called backpropagation, which is a supervised learning algorithm where the network learns the desired output from various input data (Oliveira et al., 2014) to find weights that minimize the difference between the actual and desired output (Whittington and Bogacz, 2017). Each neuron receives a weight via the connected synapses and produces an output bypassing the weighted sum of those input signals using an activation function (Sazli, 2006; Han and Kamber, 2012). The activation function maps the weighted inputs to the next layer as an input to repeat the whole process until the minimum error is reached (Karazi et al., 2019).

FFNNs are very powerful, but they have the crucial limitation that the previous input can not be reused, as there is no memory to store the information (Gruslys et al., 2019). Since these networks only consider the current time, it has no explicit notion of order in time. Thus they can not remember anything about the past except its training (Donges, 2019). Researchers usually refer to FFNN as a static network (Bucci, 2019), where the information flow takes place in only one direction, from the input layer to the output layer, where the response is independent of the previous node (Rawat and Wang, 2017).

To overcome the shortcomings of the FFNN, the so-called recurrent neural networks (RNN) can be used where „connections between units form a directed cycle“ which allows internal feedbacks (Poznyak et al., 2018, p. 59). This type of network creates an internal state that allows dynamic temporal behavior, i.e., passing data from input to output, but also from later layers to earlier layers (Bucci, 2019).

4.4.2. Activation Functions

Activation functions are used explicitly in ANNs to calculate the sum of input signals and their corresponding weights, while the present data is manipulated by gradient processing, usually gradient descent (Nwankpa et al., 2018). Finally, an activation function is applied to produce an output signal of the respective layer, which is then used as an input in the next layer of the ANN (Sharma et al., 2020). They are used to increase the expressiveness of neural networks, which allows them to acquire the meaning of artificial intelligence (Wang et al., 2020). The activation function is the core of the structure of a deep neural network and can be either linear or non-linear depending on the function it represents. Common activation functions include sigmoid, hyperbolic tangent (tanh), and Rectified Linear Unit (ReLu) (Figure 7), which are all non-linear activation functions capable of learning the complex relationship between the input and output data.

The sigmoid or logistic activation function is given by

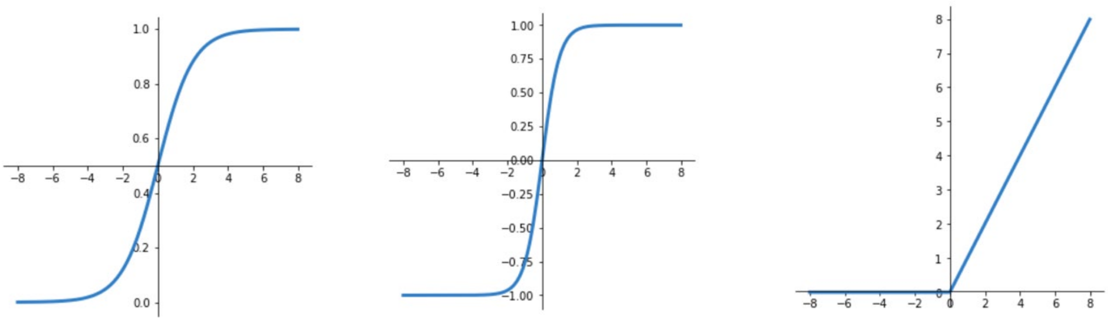
$$\sigma(x) = \frac{1}{1 + e^{-x}} \epsilon(0,1) \tag{14}$$

The tanh activation function is given by

$$\sigma(\mathrm{x}) = \frac{\mathrm{e}^{\mathrm{x}} - \mathrm{e}^{-\mathrm{x}}}{\mathrm{e}^{\mathrm{x}} + \mathrm{e}^{-\mathrm{x}}} \epsilon(-1,1) \tag{15}$$

The ReLU activation function is given by

$$\sigma(\mathrm{x}) = \max\{0, \mathrm{x}\} \epsilon \mathbb{R}^{+} \tag{16}$$



a) Sigmoid function b) tanh function c) ReLu function

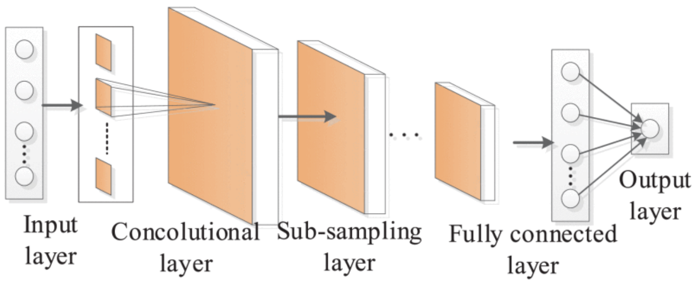
Figure 7. The Shape of Three Common Activations Functions

4.4.3. Convolution Neural Networks

Convolution neural networks (CNN) were proposed by Lecun et al. (1998) and were initially developed for computer vision and image processing (Lara-Benitez et al., 2020). CNNs have a deep, multi-layer feed-forward architecture and an impressive ability to generalize compared to networks with fully connected layers (Shen et al., 2019; Nebauer, 1998).

The network is typically used for tasks where the data has high local correlation (Tian et al., 2018) and are widely being used in many different domains due to its remarkable performance (Wang et al., 2016), such as image and text recognition problems (Szegedy et al., 2015), medical tasks (LeCun et al. 2015), object and vehicle detection (Luo et al., 2017) as well as different natural language processing tasks (Bahdanau et al., 2015). CNNs can automatically capture semantically significant information from the obtained high-dimensional raw data and obtain the main features using the convolution operation, „which is a sliding filter that generates feature maps and aims to capture recovered patterns in different regions of the data" (Gupta et al., 2014, p. 353) to reduce the data dimensions (Li et al., 2021). By applying this feature extracting process, data in the same feature map can share the same set of weights, due to which the number of trainable parameters is substantially reduced (Shen et al., 2019), resulting in improved generalization (Arel et al. 2010) and ensure some local shift and distortion invariance (Shen et al., 2019). These properties make CNNs suitable for dealing with one-dimensional data such as time series. Motivated by the performance in these different areas, Gamboa (2017) started to apply CNN for time series analysis (Fawaz et al., 2019).

A CNN architecture usually comprises an input layer, convolutional layers, pooling layers, fully connected layers, and output layer, as shown in Figure 8 (Hoseinzade and Haratizadeh, 2018).



Note: The CNN architecture comprises 3 layers: two consecutive convolutional-pooling layers and a fully-connected classification layer

Figure 8. The Architecture of a CNN (Zhang et al., 2020)

The input layer serves as a dimensional converter to transform the day-ahead electricity price time series into a 2D image since the input features are one-dimensional data, while the convolution layer is two-dimensional data (Zhang et al. 2020).

The convolutional layer is supposed to adopt the convolutional operations for each output feature map by convolving the inputs with weight-sharing (Eren et al., 2019) and forms an output feature map in terms of a user-defined activation function as follows (Ragab et al. 2020):

$$x_j^l = f\left(\sum_{i=1}^{m} x_i^{l-1} \cdot k_{ij}^l + b_j^l\right) \tag{17}$$

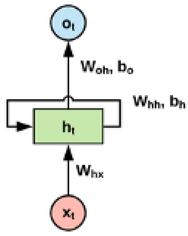
Where $\mathrm{x_j^l}$ represents the output value after convolution, k is the number of convolution kernels, j indicates the size of kernels, m refers to the feature input map from the previous layer $\mathrm{x_i^{l-1}}$ (Ragab et al. 2020). Moreover, $\mathrm{k_{ij}^l}$ and $\mathrm{b_j^l}$ are the weight and offset of the ith input map and the jth output map corresponding to the lth convolution layer respectively, f is a user-defined activation function, and $(\cdot)$ represents the convolution operator (Zhang et al., 2020; Shen et al., 2019).

The sub-sampling layer aims to achieve an information filtering task by replacing the pooled feature map with a sub-sampling function to speed up the computing and prevent overfitting while reducing dimensionality (Khan et al., 2020 Torres et al., 2021).

The fully connected layer is equivalent to the fully connected network in the traditional FFNN and is responsible for converting the extracted features to the final output (Hoseinzade and Haratizadeh, 2018). Compared to the convolutional layer, the fully connected layer can handle one-dimensional data. Thus, the two-dimensional data from the subsampling layer are converted into one-dimensional data (Zhang et al., 2020). Finally, the previous layer's output is fed into the fully connected layer. The dot product of the weight vector and the input vector is calculated to get the final output (Zhou et al., 2016).

4.4.4. Recurrent Neural Network

A RNN is a class of neural networks tailored to process a sequence of arbitrary and is mathematically similar to that of an FFNN. However, to capture dynamic relations, they use the input pattern of a sequence and its previous output of its internal state of a sequence (Figure 9). The additional connection, called feedback, changes the inputs of the individual neurons, allowing the model to recombine previous values and put the network into a new state (Petneházi, 2018). The weights and biases do not differ in all layers throughout the process, which allows them to memorize the information from the previous layer (Caterini and Chang, 2018). Thus, the complexity is minimized by using each output as input for the next hidden layer, which is depicted in Figure 9, "where O_t is the output state, h_t is the current timestamp, h_{t-1} is the previous timestamp, and x_t is passed as the input neuron" (Balas et al., 2020). The general form of an RNN can be considered as a weighted, directed, and cyclic graph containing three different types of nodes, namely the input, the hidden, and the output node (Zhang et al., 2016). The loop represents the cell state and allows information to be better recognized through the hidden layer activations passed from one step of the network to the next.



Note: Inner working of an RNN cell

Figure 9. The Architecture of a basic RNN (Olah, 2015)

The activation of the current input is computed as a function f of the previous hidden and current neuron, which can be expressed as

$$h_t = f\,(h_{t-1}, x_t) \tag{18}$$

For explicitly calculating h_t, one need to choose a proper activation function and the weights for the previous timestamps and the current input, which leads to the expression

$$h_t = tanh\,(W_{hh}h_{t-1}, W_{hx}x_t) \tag{19}$$

Where *tanh* is the activation function, *w* is the weight, *h* is the single hidden vector, $\mathrm{W_{hh}}$ is the weight at recurrent hidden neuron, and $\mathrm{W_{hx}}$ is the weight at the current input neuron (Balas et al., 2020). The formula for the output state is expressed in Equation 20, where O_t is the output neuron and W_{oh} is the weight at the output layer.

$$O_t = W_{oh,}h_t \tag{20}$$

While the FFNN processes all the values in the sequence simultaneously, the RNN can be thought of as multiple copies of the same network that pass the sequence in order, updating the cell state as they pass each value x_t in the sequence and then producing an output when they reach the final value. The chain-like nature of the unrolled RNN can be seen in Figure 10.

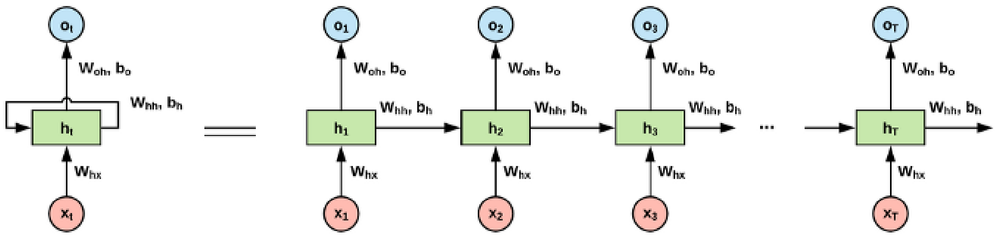


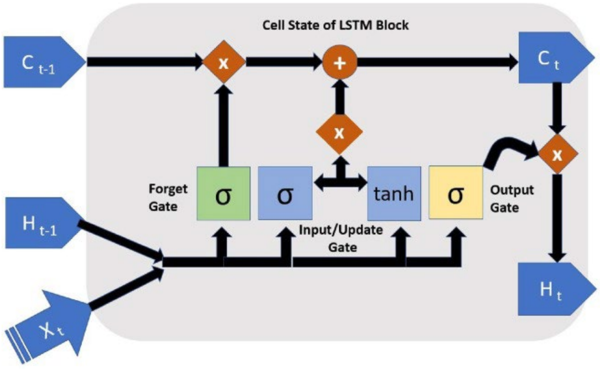
Figure 10. The Architecture of an unrolled RNN

Due to the feedback loops in the recurrent layer, it is possible to retain information in memory over time. However, Bengio et al. (1994) observed that RNNs suffer from some limitations in solving problems that require learning long-term temporal dependencies because the gradients of the loss function tend to vanish or explode as they are backpropagated through time (Chung et al., 2014). Both the exploding and vanishing gradients problem results in a precarious learning situation since the variance of the gradients increases (Sutskever et al., 2011). As the gradient shrinks, the parameter updates become insignificant, which means no objective learning process occurs (Arbel, 2018). Due to the gradient problems, RNNs struggle, not just because of the variations in gradient magnitudes but also because this hampers learning the long-term dependencies due to short-term dependencies (Chung et al., 2014). Another problem of RNN is that these networks can not give accurate predictions from recent information. Unfortunately, as the size of the gap grows, RNN does not produce efficient results because it cannot learn to combine the information (Olah, 2015).

Variants of RNN architectures have been proposed to solve the issues and eliminate the limitations: long short-term memory (LSTM) and gated recurrent unit (GRU).

4.4.5. Long-Short Term Memory

LSTM was initially proposed by Hochreiter and Schmidhuber to specifically address the exploding or vanishing gradients problem during the training phase, resulting in the failure to learn long-term sequential dependencies in data. LSTMs are tremendously good at a wide variety of problems especially in its superior performance in modeling short- and long-term dependencies in data (Bianchi et al., 2017). To solve the problem of long-term dependencies, LSTM applies three different gates to the data in which the information is processed step by step for further analysis. The vanishing gradient problem is solved by the fact that there is no bias towards the most recent observations, but it „keeps a constant error flows back through time“ (Bianchi et al., 2017, p. 25). While a RNN takes in its previous hidden state and the current input and outputs a new hidden state, an LSTM does equal, except it also takes in its old cell state and outputs its new cell state (Ugurlu et al., 2018). The main difference between RNN and LSTM lies in the repetitive modules of these neural networks. While an RNN has a relatively simple structure, such as a single tanh layer (Figure 8), an LSTM cell is composed of several layers and different types of memory blocks within a module, which are controlled by gates, resulting in more complex processing of information Figure 11). These gates control the flow of information to the hidden neurons and preserve the extracted features from previous time steps (Le et al., 2015).



Note: Complete internal architecture of an LSTM cell with all its layers and memory blocks.

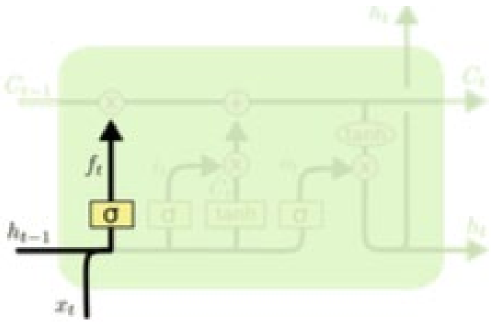
Figure 11. Internal Architecture of an LSTM (Nabi et al., 2021)

An LSTM network has three gates that manage the contents of the cell states: forget gate, the input gate, and the output gate. The horizontal line represents the state cell at the top of the cell in Figure 11. These gates are „simple logistic functions of weighted sums, where the weights can be learned by backpropagation" (Petneházi, 2018, p. 2). Each gate in the cell has a specific and unique functionality. The transition equations that modulate the information flow across the cells and calculate the output are the following (Wu et al., 2020):

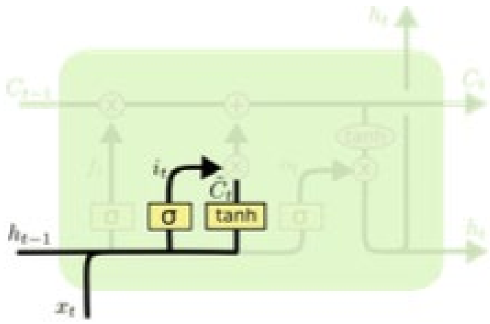
$$\begin{aligned} forget\ gate:\ f_t &= \sigma(W_f x_t + U_f h_{t-1} b_f) \\ candidate\ state:\ \tilde{C}_t &= tanh(W_c x_t + U_c h_{t-1} + b_c) \\ input\ state:\ i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i) \\ cell\ state:\ C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \\ output\ gate:\ o_t &= \sigma(W_o x_t + U_0 h_{t-1} + b_o \\ output:\ h_t &= o_t \odot tanh(C_t) \end{aligned} \tag{21}$$

x_t is the input vector at time t. $W_f, W_c, W_i, and\ W_o$ are the weight matrices applied to the cell's input. $U_f, U_c, U_i, and\ U_o$ are square matrices that define the weights of the recurrent connections, $b_f, b_c, b_i, and\ b_o$ are the bias term of the related gates, while σ is a sigmoid or logistic function and ⊙ is the Hadamard product function (Bianchi et al., 2017).

The "forget gate," represented in Figure 12, controls which information from the cell state in that particular timestamp h_{t-1} should be discarded. Mathematically, it calculates the proportion f_t, given by the sigmoid function, taking the hidden state from the previous state h_{t-1} and the current input x_t. The output of the sigmoid function is between 0 and 1 for each number in the cell state Ct-1, so $f_t \in (0, 1)$, where a value of 1 expresses to keep the previous information, and a value of 0 means reject all previous information.

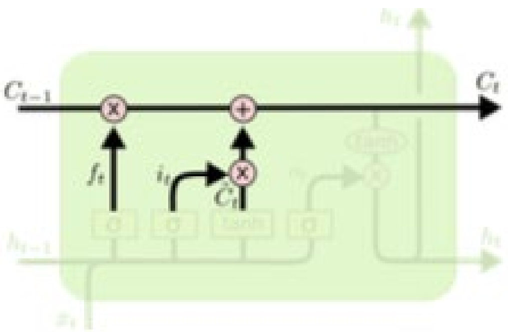


Note: Flow of information
Figure 12. Forget Gate

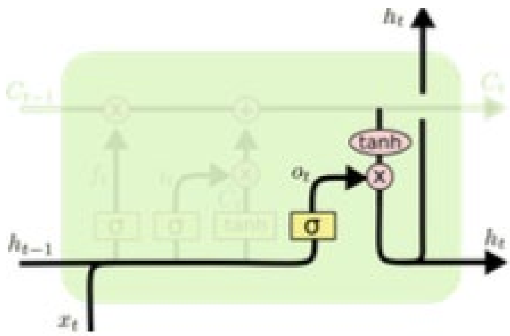


Note: Computations for new cell state
Figure 13. Input Gate

Next, the input gate operates on the previous state h_{t-1}, after being modified by the forget gate, and uses the hyperbolic tangent function $tanh$ to calculate the candidate $\tilde{C}_t$ for the new current cell state that could be used to create the new current cell state (Figure 13). The input gate then forms a vector i_t of values between 0 and 1, which influences the flow of information from candidates into the cell state. Values approaching 0 prevent the inflow of information, whereas values approaching 1 enable the inflow of new information (Gudikandula, 2021). The selection vector i_t is calculated using the sigmoid function σ from the weighted input x_t and the weighted past activation h_{t-1}, as with the forget gate. The cell state input vector is calculated with the same inputs and its own weights. To generate the new cell state C_t, we multiply the old state C_{t-1} by f_t and add the new candidate's values $\tilde{C}_t$, scaled by how much we decided to update each state value (Figure 14).



Note: Computations for the state value.
Figure 14. Update Gate



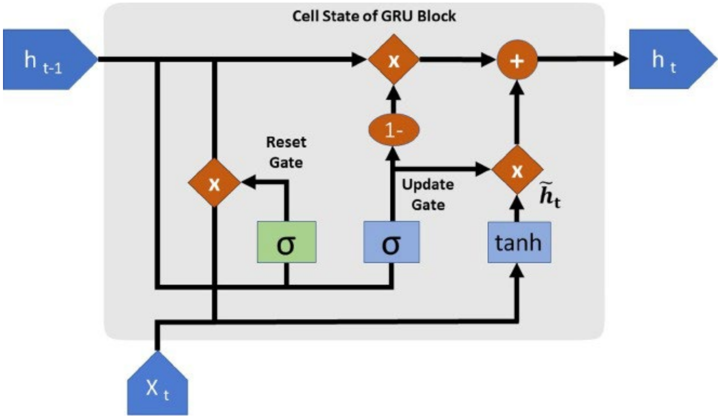
Note: Computations for the output.
Figure 15. Output Gate

Once we have calculated the new current cell state C_t, we can finally calculate the cell output h_t (Figure 15). The output will be a filtered version of the cell state (Olah, 2015). We use the sigmoid layer to decide which values of the cell state will be used for the output. The cell state is then transformed through the tanh function to give weightage to the essential values ranging from -1 to 1 (Gudikandula, 2021). By multiplying the cell state with the result of the output gate o_t, we decide exactly which information we want to use for the following part of the network and which part of the information is unnecessary for further analysis (Olah, 2015). Therefore, we have managed to reduce the amount of information passed on without losing essential information about the context of the sequence.

The architecture of an LSTM reduces the issue of vanishing gradients due to the absence of non-linear transfer functions applied to the cell state (Bianchi et al., 2017). However, exploding gradients are still a problem with LSTM. A solution approach used for this is gradient clipping (Pascanu et al., 2013). Here, the gradient is scaled to a smaller value as soon as it exceeds a threshold value. This has the effect of stabilizing the training since large upward fluctuations are compensated for. Due to their sequential architecture, LSTMs suffer from high complexity in the hidden layer, as an LSTM network has about four times as many parameters as a simple RNN with the same size of hidden layers (Mikolov et al., 2014). A slight variation of the SLTM is the Gated Recurrent Unit (GRU), which is more straightforward than the standard LSTM model because it combines the forget and input gates into a single update gate, incorporates cell state and hidden state, and makes some other changes (Misra, 2021).

4.4.6. Gated Recurrent Unit

GRU has many similarities with LSTM and can be represented as a variant of LSTM that adaptively captures dependencies at different time scales (Cho et al., 2014). Like LSTMs, GRUs have also proven to be a viable option to solve the vanishing or exploding gradient problem that occurs in typical RNNs, when learning long-term dependencies (Nabi et al., 2021, p. 8). While LSTMs have higher memory requirements due to the many memory cells in their architecture, GRUs have gating units that modulate the flow of information within the unit without requiring separate memory cells (Chung et al., 2014; Salehinejad et al., 2018). A schematic depiction of GRU is reported in Figure 16. Similar to LSTMs, GRU also uses tanh and sigmoid functions to compute necessary values. The main difference is in the number of gates and weights since GRU does not have a separate output gate nor a separate forget and input/update gate (Petneházi, 2018; Nabi et al., 2021).



Note: Complete internal architecture of an GRU cell with all its layers and memory blocks.

Figure 16. The Internal Architecture of a GRU (Nabi et al., 2021)

GRUs make use of two gates. They combine the forget and input gates into a single update gate that controls how much of the previous memory will be kept (Bianchi et al., 2017). Unlike the update gate, GRUs have a reset gate, which can reset the memory of the cell and "determines how to combine the new input with the previous memory" (Ugurlu et al., 2018, p. 10). The state equations of the GRU are the following:

$$
\begin{aligned}
reset\ gate:\ r_t &= \sigma(W_r h_{t-1} + R_r x_t + b_r) \\
current\ state:\ \tilde{h}_t &= \sigma(h_{t-1} \odot r_t) \\
candidate\ state:\ z_t &= tanh\left(W_z \tilde{h}_{t-1} + R_z x_t + b_z\right) \\
update\ gate:\ u_t &= \sigma(W_u h_{t-1} + R_u x_t + b_u) \\
new\ state:\ h_t &= (1 - u_t) \odot h_{t-1} + u_t \odot z_t
\end{aligned}
\tag{22}
$$

The weights and biases follow the same notation as mentioned before. A GRU cell has much more parameters than an RNN unit, but due to the structural difference, it has fewer parameters than the LSTM cell resulting in a less complex design, making it more computationally efficient and easier to train (Nabi et al., 2021). In an empirical comparison of GRU and LSTM, Chung et al. and Józefowicz et al. concluded that GRU could outperform LSTM on a suit of tasks „both in terms of generation capabilities and in terms of the time required to achieve convergence and update parameters." (2014, p. 1; 2015).

4.5. Model Evaluation

In our study, the ADF statistic (-15.833) is less than the critical value at 1% (-3.431), so we reject the null hypothesis H_0 for any significance, which means that evidence was found stationarity. Similar results were obtained by a study by Akay (2015), who confirms the absence of a unit root in Australian spot electricity prices. Forecast evaluation methods focus on error calculation. Hence, the main goal is to predict the day-ahead price each hour of the following day. Our primary measures of performance will be based on accuracy. To assess the prediction performance of the models, different statistical measures can be utilized. Perhaps the most commonly used metrics to measure the accuracy of forecasts are those based on absolute error (Weron and Chichester, 2006): MAE (Mean Absolute Error) and RMSE (Root Mean Square Error), where the error is defined as the difference between the observed value and its forecast for the corresponding period (Dissanayake et al., 2021). As Equation 23 indicates, MAE is calculated as the average of the forecasted value $\hat{y}$ and the actual value y without considering the direction. MAE is a linear score that penalizes all errors proportionately, which means that it does not weigh the different types of errors more or less, but at the cost of greater magnitudes errors (Permatasari et al., 2018). Additionally to MAE, we use RMSE as a performance metric to evaluate the model choice. As Equation 24 shows, RMSE is calculated as the square root of the MSE. Since the difference between predicted and target values is squared, RMSE gives relatively high weight to large errors, thus minimizing significant errors (Chai and Draxler, 2014). MAE and RMSE are denoted by:

$$MAE = \sqrt{\frac{1}{T}\sum_{h=1}^{T}\left(P_h - \hat{P}_h\right)} \tag{23}$$

$$RMSE = \sqrt{\frac{1}{T}\sum_{h=1}^{T}\left(P_h - \hat{P}_h\right)^2} \tag{24}$$

Where P_h is the actual price at hour h, $\hat{P}_h$ is the corresponding forecasted price for P_h, and T is the number of observations.

Since it is hard to compare absolute errors between different data sets, one might use measures based on absolute percentage errors like the MAPE (Mean Absolute Percentage Error). MAPE works well in load forecasting, but its values become very large for electricity price trajectories with an MCP close to zero or even negative values, regardless of the actual absolute errors (Singh and Mohanty, 2015). Thus MAPE is „dominated by the periods of low prices and is not very informative“ (Lago et al., 2021, p. 11). In practice, the best forecast model is expected to yield a low RMSE and MAE concurrently. Therefore we use both KPIs named in Eq. 23 and Eq. 24, which allows for a deeper understanding and assessment of the merits and limitations of the models.

4.6. Diebold-Mariano Test

The traditional evaluation criteria presented above make it possible to measure the performance of the models in relation to each other by comparing their accuracy. However, it is impossible to efficiently assess whether the difference in predictive performance is significant from a statistical point of view (Lago et al., 2021). To address this problem, the significance of differences is assessed using the Diebold-Mariano method, which provides a quantitative method for assessing predictive accuracy (Chen et al., 2014). The DM test builds a covariance stationary loss function $L\left(\varepsilon_{d,h}^{A}\right)$ to calculate the loss differential series using Eq. 25, where A and B are the prediction models, and $\varepsilon_1^A..\varepsilon_N^A$ and $\varepsilon_1^B..\varepsilon_N^B$ are the associated forecasting errors.

$$\Delta_{d,h}^{A,B} = L\left(\varepsilon_{d,h}^{A}\right) - L\left(\varepsilon_{d,h}^{B}\right) \tag{25}$$

To determine whether one model predicts more accurately than another, we need to perform the DM test, where the null hypothesis H0 of A has an accuracy equal to or worse than B, against the alternative hypothesis H1, having a better accuracy (Chen, 2014):

We conduct the one-sided DM test, with the null hypothesis given as:

$$\mathrm{H}_0 = \mathrm{E}\left(\Delta_{\mathrm{d,h}}^{\mathrm{A,B}}\right) \leq 0 \tag{26}$$

The alternative hypothesis that one is better than the other is given as:

$$\mathrm{H}_1 = \mathrm{E}\left(\Delta_{\mathrm{d,h}}^{\mathrm{A,B}}\right) > 0 \tag{27}$$

The null hypothesis is usually rejected if the p-value is less than the commonly selected significance level of 5% (Lago et al., 2021). Thus we can conclude that the predictive accuracy of model A is significantly better than that of model B (Maciejowska and Weron, 2016).

5. Data

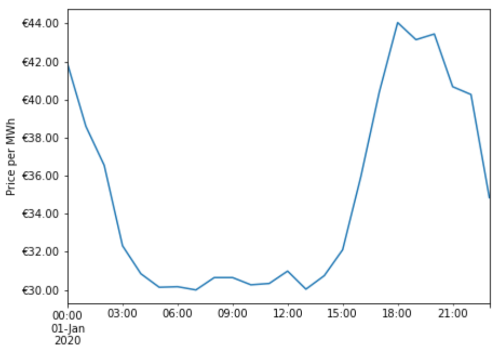
This chapter presents and describes the data used for this study. Extensive data preparation is required to ensure that the applied machine learning models accept the data and form a proper and coherent timeline. The datasets consist of files of different formats that need to be combined into a single dataset with one observation per hour. The necessary pre-processing steps are discussed in Section 5.2. Finally, an exploratory analysis of some characteristics and the electricity price is done in Section 5.3.

5.1. Data Collection

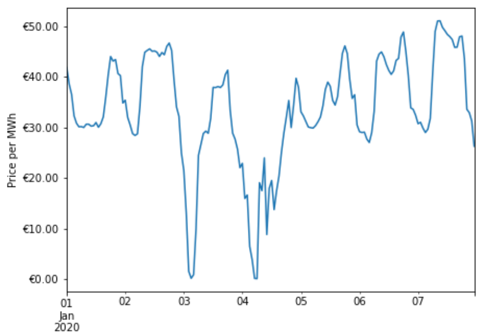
For this study, the EPEX Spot day-ahead market covering hourly electricity prices for Germany from January 2016 until December 2020 was used. Guided by the analysis in Section 1 of what fundamental factors drive electricity prices in Germany, we have obtained a set of explanatory variables. Historical load features, the actual generation of production from variable renewables, weather, and calendar effects are traditionally used as solid determinants of electricity price. The hourly load and generation profiles for Germany were retrieved from ENTSO-E. According to ENTSO-E, load is the power consumed by the grid, including network losses but excluding consumption of pumped storage and generating auxiliaries (ENTSO-E, 2021). The load data includes all energy that German power plants sell to consumers. According to the correlation, the load is the best indicator of electricity price in the German market area. The historical weather data was retrieved from the OpenWeatherMap website. Five features were added to the forecast models, including the temperature, pressure, wind speed, humidity, and precipitation rate, which are the main driver for the seasonal variation of electricity pricing. We computed a Germany-wide weighted average based on the population of the five major cities. This is justified because a heatwave that hits a city with twice the population will likely to require more total energy than in a smaller city.

5.2. Data Pre-processing

Several pre-processing steps are required to extract the relevant data and convert it into data structures accepted by our machine learning libraries. The purpose of these steps is to obtain a complete and orderly time series for the relevant period. The datasets must have the same granularity to combine the different data sets into a consistent time series. Due to application purposes, we aggregated the weather data, which was provided with a higher grain. Since the empirical analysis focuses on estimating the variation of price elasticity throughout the day on an hourly basis, we obtain 24 electricity prices per day. Hence we trained the model with one data point for each hour, respectively. Figure 17 displays the electricity price variation of 24 and 168 hours. Various types of seasonality influence electricity prices. In the early morning hours, prices are relatively low, sometimes even zeros. In addition, there are often two price peaks during the day, one before and one after lunchtime.



a) Based on 24h



b) Based on 168h

Figure 17. Price distribution of hourly prices based on 24h (day) and 168h (week)

Some data are cyclical by nature, and time is a good example: minutes, hours, days of the week, and month follow cycles. All these values have periodic properties (Moon et al., 2018). For example, since December and January are temporally adjacent, they have similar properties in terms of weather. However, if we represent them numerically with the absolute values, the difference in categorical format is 11. This is not desirable because the machine learning model should recognize that the two months are only one month apart. To reflect this periodicity, we convert the monthly data into continuous data using (Jung et al., 2020):

$$Month_x = sin\left(\frac{360}{12} * Month\right) \tag{28}$$

$$Month_y = cos\left(\frac{360}{12} * Month\right) \tag{29}$$

To achieve numerical stability and a smooth training process, we need to standardize the data. According to Hastie et al., neuronal networks perform much better when continuous variables have similar scales (2009). By standardizing the weights in convolutional layers, small values for weight initiations make a small incremental change to the weights during the backpropagation (Jun, 2021). According to Brownlee (2019), poor data scaling leads to a slow training process, while no scaling can also lead to exploding gradients in various regression problems, causing the training process to fail. Since the datasets provide high numerical values and the model is sensitive to magnitude, we perform the min-max scaling to transform the data range into a more reasonable range. In this approach, the data is scaled to a fixed range between [0,1]. Min-max scaling is usually done using the following equation:

$$\hat{x} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{30}$$

Where x represents the original data feature, $\hat{x}$ the scaled data, x_{min} and x_{max} the minimum and the maximum of the selected feature for the whole dataset, respectively. After the relevant data and forecasting models have been selected, the prediction performance in the test set will be used to compare the different models by forecasting the day-ahead electricity price.

The purpose of predicting is to capture patterns in the training data, which enables the model to generalize its performance to unseen data (Hastie et al., 2009). Hence, we divided the data into three contiguous periods. The test set contains data for 2020, and the validation set consists of the preceding year, 2019. These sets, therefore, contain 8,760 and 8,784 observations, respectively. The training set contains the preceding 26,278 observations from the 1st January

2016 to the 31st December 2018, which is about 59.97% of the total data. For dividing the dataset into different periods, one can use the train test split from the sklearn package, which might yield more robust and reliable estimates on the performance for unseen data (Hastie et al., 2009). However, this approach does not work for time series data where the temporal order is randomly shuffled and then split, as there is no dependency from one observation to another (Mayo, 2020; Chollet, 2018). Basic descriptive summary statistics for the periods without any data pre-processing are presented in Table 1.

	Train	Validation	Test	Full
Observations	26278	8760	8784	43822
Mean	35.89	37.80	30.47	35.15
Standard deviation	17.38	15.36	17.49	17.21
Skewness	-0.004	-1.42	-0.28	-0.29
Kurtosis	5.16	8.96	6.55	5.71
Min	-130.09	-90.01	-83.94	-130.09
Max	163.52	121.46	200.04	200.04

Table 1. Summary Statistics in the Training, Test, and Validation Sets

Table 1 shows kurtosis and skewness for the electricity price, which can arise from peak values or autocorrelation (Bierbrauer et al., 2007). While the series has a large excess kurtosis in the validation and test period, it is more limited in the training period. This indicates that extreme observations have become more frequent. Hence, outliers are detected before conducting the empirical analysis. According to Mugele et al., we filtered values that exceed three times the standard deviation of the original price (2005; Gianfreda, 2010). These observations are then replaced by the mean of the respective series using linear interpolation. Thus, we try to balance the disturbing influence of outliers on the analysis and the disturbing influence induced by replacing extreme values.

After replacing the outliers with the corresponding values, the adjusted time series plot can be observed in Appendix Figure 26b. It may seem that the price series has become even spikier after the outlier treatment. However, this is not the case as the outliers were not visible in Figure 26a only because of the area of the Y-axis. The summary statistics for the price after outlier treatment can be found in Table 2.

	Train	Validation	Test	Full
Observations	26278	8760	8784	43822
Mean	35.97	38.01	30.59	35.27
Standard deviation	16.43	14.27	16.32	16.22
Skewness	0.37	-0.59	-0.049	-0.29
Kurtosis	1.37	3.04	1.64	1.61
Min	-32.58	-33.57	-33.67	-33.67
Max	103.87	103.69	103.22	103.87

Table 2. Summary Statistics for the Price after outlier treatment

5.3. Exploring the Data

Before proceeding to the methodology used in this study, we take a closer look at electricity trading in Germany. The characteristics of electricity prices are not only crucial for understanding the market but also for assessing the practical reliability of the models. Section 5.3.1 and 5.3.2 explore the calendar and weather effects of trading in the electricity market. Subsequently, Section 5.3.3 examines the electricity price and load behavior and its possible dependence on the delivery hour.

5.3.1. Calendar Effects

The electricity price and the available capacity change over time. Some of these changes, such as unforeseen outages, are unpredictable, while others follow predictable cycles related to various calendar effects to capture daily, weekly, seasonally, or annually variations of price patterns. Using the high-frequency data, we assess the calendar effect in different time frequencies. Figure 18 illustrates the electricity price observations for a month over the length of the study. Two seasonal periods with a similar pattern were distinguished: a winter period with high electricity prices (Sep–Feb) and a summer period with a lower electricity price than winter (Mar–Aug). In European markets, the annual cycle can arise because consumption during the summer months is lower compared to the winter months. Capturing weekly and intraday variations could further improve the forecast model since these cycles may also result from a reluctance to work outside business hours, resulting in less capacity being made available to the market. A distinction was made between weekdays and weekends as well as business hours and holidays by adding a dummy variable for each of those periods (Figure 19).

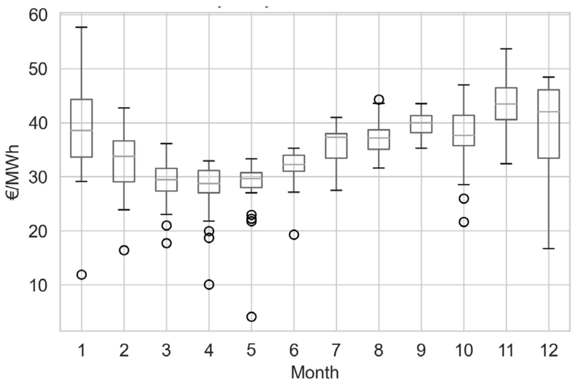


Figure 18. Hourly electricity price distribution over each month

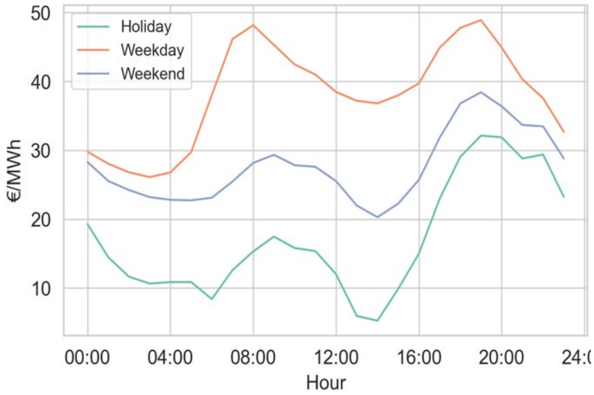


Figure 19. Average Daily Electricity Price by Hour

The dummy variables are added using binary values. For example, the weekday/weekend dummy variable X can be defined as follows:

$$\begin{cases} X = 1, & if\ the\ observation\ is\ on\ a\ weekend\ day \\ X = 0, & if\ the\ observation\ is\ not\ on\ a\ weekend\ day \end{cases} \tag{31}$$

To analyze the role of calendar effects in the forecasting model, different forecasting scenarios were initially developed. The results confirm that calendar effects exist. Hence, we decided to include calendar variables to account for the possible effects of cycles.

5.3.2. Weather

In this study, measured weather data were added as additional parameters for the investigation of the electricity price since the weather effect on electricity can be significant (Beccali et al., 2008). Figure 20 shows that as wind speed goes up, electricity price goes too low accordingly. In February 2019, the installed capacity of wind power in Irland was accounted for 49% of electricity demand, with February 2020 providing 56% of demand met by wind energy (EirGrid 2020). This clearly shows the importance of wind energy. From a seasonal perspective, both heating and cooling demand cause high prices in winter and summer, respectively. However, due to the high share of hydropower plants in electricity generation, prices tend to decrease in summer. The winter period accounts for much higher electricity prices than the summer period, as shown in Figure 21. The average hourly electricity price for the winter period is 38 Euro per MWh, whereas the electricity price for the summer period is 32 Euro per MWh.

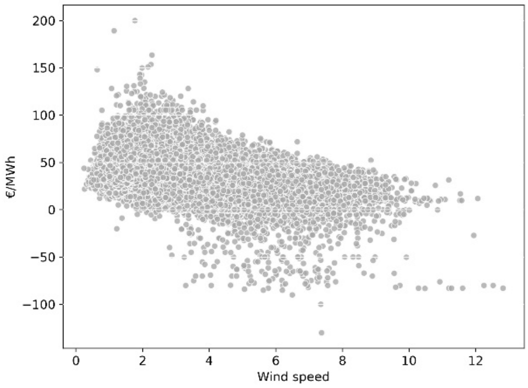


Figure 20. Correlation between Electricity Price and Wind Speed

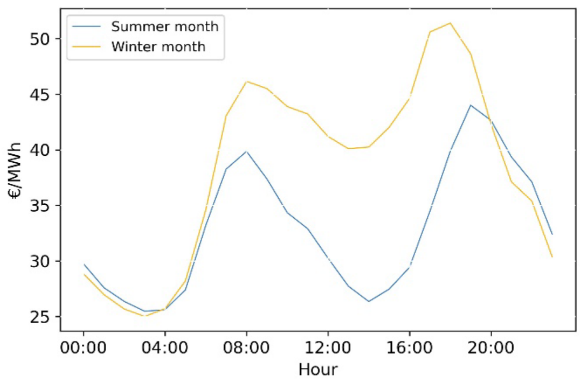


Figure 21. Electricity Price based on Seasonality

5.3.3. Price and Total Load

A significant daily variation in the time series can be seen at the mean hourly electricity load (Figure 22). The plot shows differences in the level for night hours (00:00-6:00, 20:00-00:00) compared to daytime. Peak values in the morning hours (08:00-12:00) and evening hours (17:00-19:00) are also significant. Especially in the morning, variation in load levels is higher than at other times. Furthermore, there are clear drops in electricity load during public holidays, indicating variation in demand on these days. Overall, the load patterns on most weekdays are similar, while different but distinct patterns are observed at weekends.

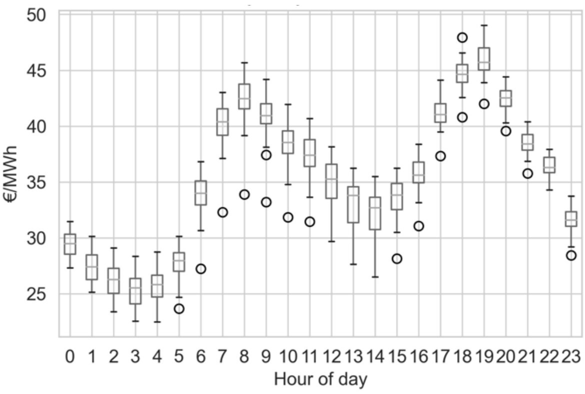


Figure 22. Price Distribution

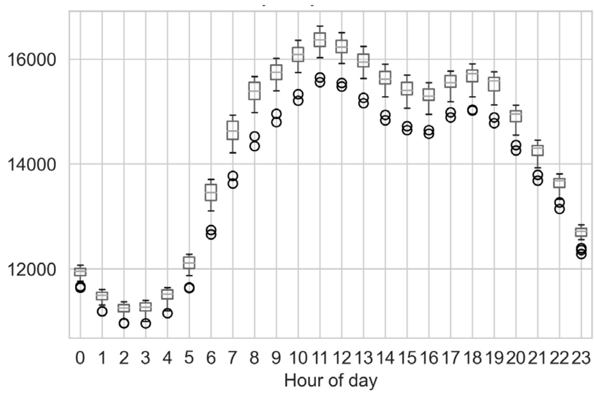


Figure 23. Load Distribution

The level and volatility of hourly EPEX electricity prices may vary depending on each hour of the day (Hagfors et al., 2016), which is also the case in our dataset, as shown in Figure 23. The electricity price follows the load pattern and varies across the day because of changes in the electricity demand. This underlines that an increase in demand generally tends to higher prices, and decreases in demand tend to lower prices (Edgmand et al., 1996). Similarly, the price follows a weekly/yearly pattern as the different seasons influence the energy demand. Especially at peak times, higher standard deviations can be observed. A high standard deviation implies high price volatility and represents a possibility for higher returns on flexible investments such as the day-ahead action (Kern, 2020).

A slight downward trend in the electricity price from 2018 to 2020 can be observed in Table 3. While we believe that the covid-19 pandemic led to a reduction in demand for electricity, another critical factor is the continued increase in the number and capacity of renewables. The table also shows the number of negative prices for the corresponding years.

Year	**Mean**	**Min.**	**Max.**	**Number of negative Prices**
2016	28,98 €	-130.09€	104.96 €	97
2017	34,20 €	-83.06 €	163.52 €	145
2018	44.47 €	-76.01 €	128.26 €	134
2019	37,67 €	-90.01 €	121.46 €	210
2020	30,47 €	-83.94 €	200.04 €	298

Table 3. Negative Prices for each Year

Negative electricity prices mean that generators do not receive payment for generated electricity but have to pay other parties to take their electricity (Barbour et al., 2014). The reason for negative electricity prices lies primarily in the inflexibility of the power plants. These plants have produced electricity despite the negative price signal from the stock exchange. Together with the electricity from renewable energies, this has led to an electricity surplus (Podewils, 2014). After studying negative electricity prices in the German day-ahead spot market, Genoese et al. found that wind has the highest correlation with negative electricity prices (2010). Wind generation and other renewables that receive subsidies have zero to negative marginal costs compared to traditional power generators (Kern, 2020). So far, negative electricity prices had mainly occurred at night, when demand was low, but wind power generation remained high. Due to the combination of an expansion of renewable energies, a delayed coal phase-out, and lower electricity demand due to more energy efficiency, negative electricity prices are expected to be recorded for around 1.000 hours in 2022 (Huneke et al., 2021).

Table 4 shows an overview of electricity prices and their correlation with the renewable energy sources used. For simplicity, we have grouped the daytime hours into four time blocks: Night (23h-5h), Morning (5h-11h), Noon (11h-17h), and Evening (17h-23h). The results confirm the literature, as the most negative electricity prices occur when the electricity price is most negatively correlated with the wind. Another fascinating result is that the second most negative prices occur in the evening. As we can see in the table, the correlation between electricity price with hydropower, photovoltaic and other renewables was highest at this time, but for wind, it was the lowest correlation of all time zones. This suggests that the more renewable energy is generated, the more negative prices will be in the future. Table 7 in the Appendix illustrates a more detailed list of correlations between the electricity price with each renewable energies source for every hour.

Hour	# Neg. Prices	Biomass	Hydropower	Wind	Photovoltaics	Other_renewable
23-5	55	-0,14	-0,09	**-0,58**	-0,01	0,002
5-11	33	-0,06	-0,16	**-0,38**	-0,16	0,1
11-17	54	0,05	**-0,22**	**-0,36**	**-0,35**	**0,18**
17-23	8	-0,09	-0,18	**-0,41**	-0,13	-0,01

Table 4. Correlation between Electricity Price and Renewable Energies Sources

Figure 29 and Figure 30 in the Appendix illustrate the mean hourly electricity price on the German market in more detail, with a daily variation becoming apparent. There are more distinct morning and evening peaks during summer and winter. The increased color intensity in the afternoon hours in summer corresponds to the use of air conditioning.

6. Empirical Results

In this section, various ANNs in comparison with some other approaches which do not apply deep learning techniques have been performed to investigate the performance concerning electricity price. The performance of these models serves as benchmarks against which the performance of all the models selected in the previous sections can be assessed. Finally, we compare the performance differences using the Diebold-Mariano test.

6.1. Quantitative Analysis

In our first experimental setup, we used linear and non-linear learning models to perform the hourly EPF to obtain and evaluate model forecasts in the test set. Table 3 summarises the results (RMSE and MAE) of each of these five approaches, on the whole, test set. The naïve approach, which traditionally serves as the benchmark, performs the worst with a total RMSE of 13.74 EUR/MWh and an MAE of 9.51 EUR/MWh, which is a good indicator that the other models were able to use the predictors to improve performance.

Model	Naive	Linear Regression	Ridge Regression	KNN	XGBoost
MAE	9.51	4.18	4.33	6.14	**3.27**
RMSE	13.74	4.61	5.84	8.41	**4.53**

Table 5. Comparative Table of Machine Learning Models

Not surprisingly, the more advanced benchmarks perform better than the naïve approach that predicts the electricity spot price. The non-linear gradient boosting model performs best on both metrics, which achieves an RMSE of 4.53 EUR/MWh and an MAE of 3.27 EUR/MWh on the test set.

So far, we have seen that it is possible to predict the electricity spot price with relatively good accuracy. In the second experimental setup, deep learning was used to improve the performance. Table 4 summarizes the results for the univariate and multivariate neural networks and the results of the statistical ARIMA model.

		Univariate					**Multivariate**				
Model	ARIMA	GRU	CNN	LSTM	MLP	CNN-LSTM	GRU	CNN	LSTM	MLP	CNN-LSTM
MAE	4.37	2.51	**2.44**	2.49	2.73	2.45	2.50	2.78	3.28	3.58	3.58
RMSE	5.24	3.72	3.73	3.78	3.98	3.76	**3.51**	3.79	4.37	4.63	4.63

Table 6. Comparative Table of Deep Learning Models

We can see some significant differences between the different metrics of the models. As expected, the ARIMA model has the worst performance in both metrics and is not suitable for short-term EPF. A more detailed hyperparameter optimization could further improve the performance of the ARIMA model. In any case, due to its linear nature, the algorithm cannot capture the non-linear behavior and therefore cannot represent the complex behavior. The best results in terms of RMSE were obtained by the multivariate GRU with a value of 3.51 EUR/MWh and an MAE of 2.50 EUR/MWh, while the univariate CNN obtained the best results in terms of MAE with a value of 2.44 EUR/MWh and an MAE of 3.73 EUR/MWh on the test set.

For simplicity, we focus only on the three best performing univariate and multivariate models, namely GRU, LSTM, and CNN. The following diagram shows the RMSE error profile of the predictions and the median electricity price for the test set.

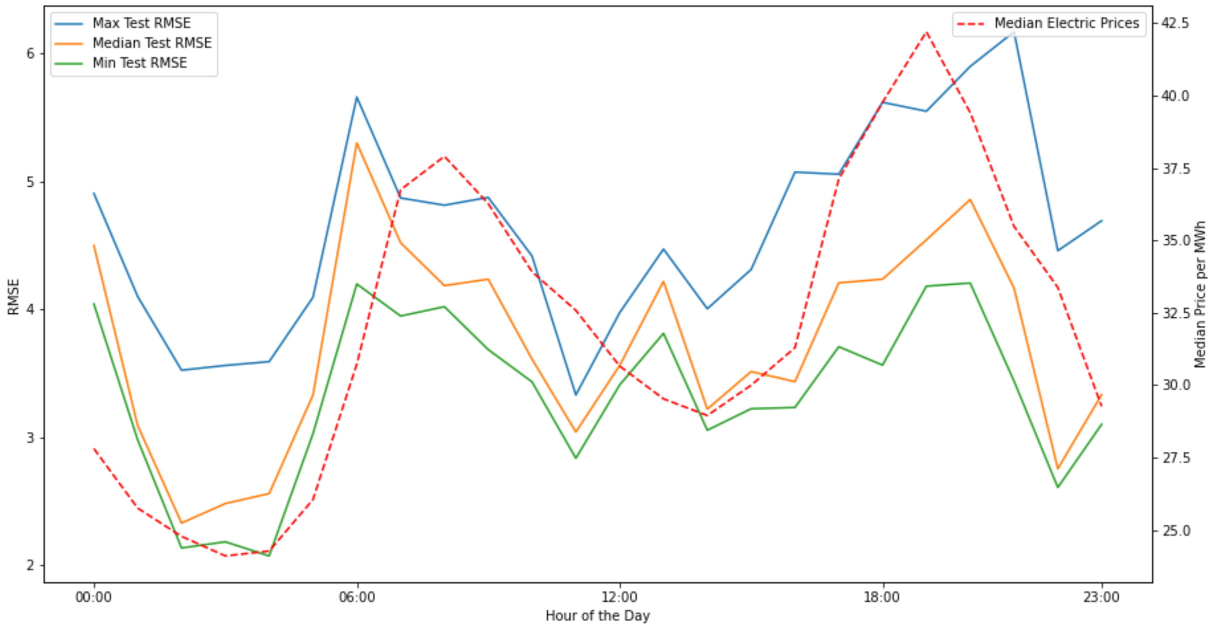


Figure 24. Summarize Statistics of RMSE

Different market conditions may characterize price forecasts for different hours. These include fluctuations in the number of active participants and the amount of available capacity. Figure 31 and Table 8 in the Appendix show that the different models' performance is not uniform throughout the day but varies depending on the hour. Due to the low RMSE, it looks like the models are generally best at predicting electricity prices during the night and midday hours, as the figure show error peaks at 06:00 and a new peak at 20:00. It also seems that the models are more accurate in times of high electricity prices.

Interestingly, compared to the other deep learning models, the multivariate GRU has an exceptionally high RMSE in the night hours 2-5 and sporadically between 11-16. Therefore, the higher RMSE can probably be attributed to its poor performance in the first hours of the day. In summary, the univariate CNN and the multivariate GRU show the best overall performance throughout the day.

The weekly (Monday-Sunday) performance results are consistent with the overall average performance without exceptional cases. The multivariate GRU performs best every weekday, illustrated in Figure 32 and Table 9 in the Appendix. Regarding the breakdown of forecast error for days of the week, the figure shows that Sundays and Mondays have relatively higher forecast errors. We also evaluate the monthly performance of each model in Figure 18 and Table 6 Appendix. The results for each month are generally consistent with the overall average performance with some exceptional cases. The figure shows the forecast error by month, where April and September exhibit the highest forecast error. The results demonstrate the relatively good performance of the model. However, there are three months (April – June) where the univariate GRU and univariate LSTM performs better than the multivariate GRU. For a closer look at the months in which the model performed worse than the other models, these were divided into weeks of the year, shown in Figure 33 and Table 10 in the Appendix.

6.2. Hourly estimation

The following figure shows the forecast and actual prices for the 7^{th} and 8^{th} of January 2020 for the German electricity market. Various forecast results are shown and compared with the actual prices. The black line represents the actual price, while the colored lines show the forecast price of the respective model. It can be said that the model behavior is acceptable where there are no high spikes in the price.

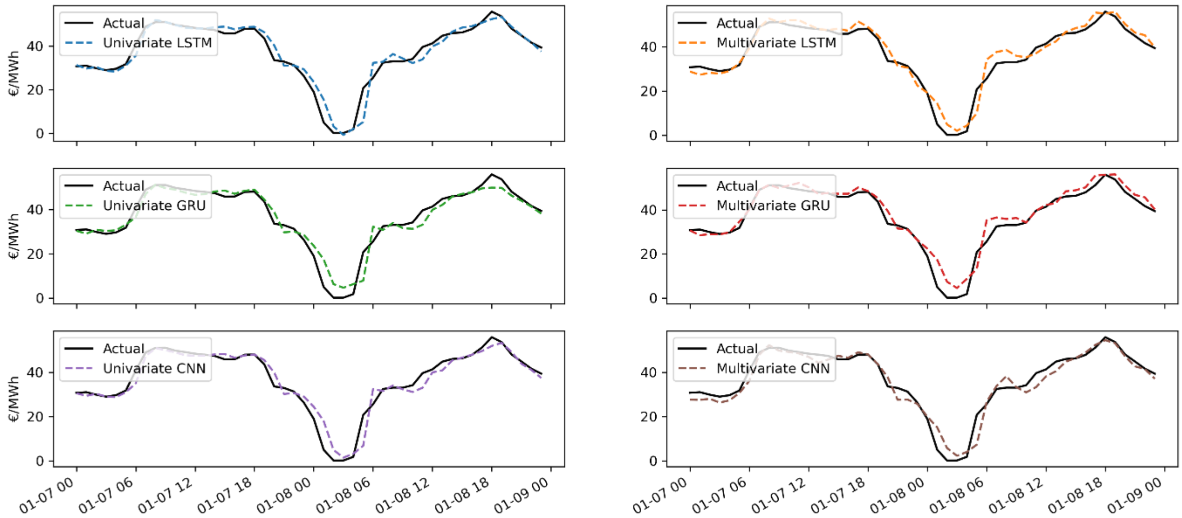


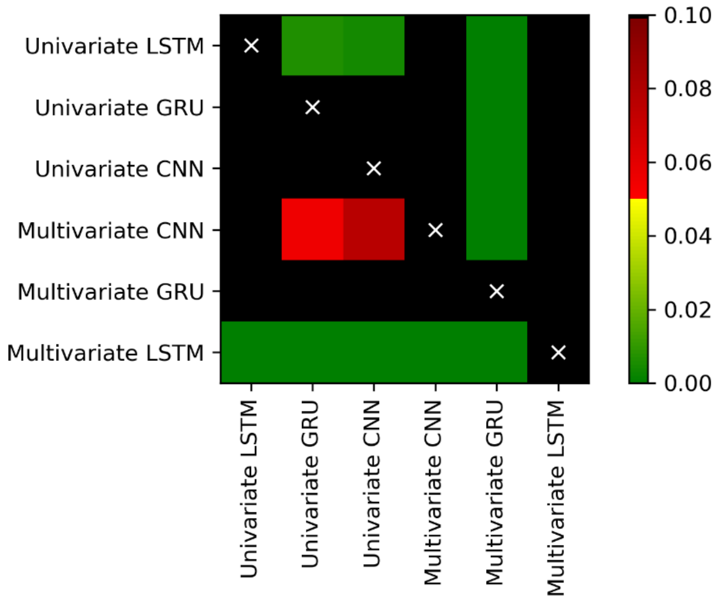
Figure 25. Hourly Estimation of each Deep Learning Model

As can be seen in all figures, the behavior of the EPF was primarily captured by the models. The predicted results are close to the actual market values, especially at the hours with high prices. However, some parts could not be predicted accurately due to spikes and high volatility of the price during a day. Some misclassification exists but is being predicted at the wrong time, which can be seen in almost every figure at around 06:00 on the 8th of August. From the comparative table of deep learning models and the figure above, we can see that the GRU model can learn the dynamics of the price movement better than any other model. The structure of the GRU cell can explain this.

For a more detailed analysis of the prediction performance, we plot the distribution of the error of the best-performing model in Figure 35 in the Appendix, which shows that the prediction error is normally distributed around zero. Thus, the model is appropriate for forecasting purposes. The classification of the forecast error has essential implications for the physical relationships in the electricity market and bidding strategies.

6.3. Evaluation Based on DM

Table 4 and Table 5 can be used to rank the different models, but no statistically significant conclusions can be drawn about the performance of the forecasts. We use an open-source python library named *epftoolbox* that provides easy access to a DM test to highlight the statistical significance of the performance difference between all model variations that take the correlation structure into account (Diebold and Mariano, 1995).



Note: Results of the DM tests are defined by the loss differential series in between different models. The closer the p-values are to zero (dark green), the more statistically significant is the performance between the forecasts of a model on the x-axis compared to the forecast of a model on the y-axis.

Figure 26. Results of the DM tests.

In Figure 26, we show the p-values for the DM tests between neural network based methods and compare the performance of the models with each other. We use heat maps arranged in a chessboard shape to indicate the range of the obtained p-values. It tests the forecasts of each pair of transformations against each other and uses a color map to show p-values. The closer the p-values are to zero (dark green), the more statistically significant is the performance between the model's forecasts in the x-axis compared to the forecast of a model on the y-axis. For instance, the last row is green, indicating that the forecasts of multivariate LSTM significantly outperformed those of all other models. Furthermore, the performance of the univariate GRU and univariate CNN is statistically proven by the DM test. According to the profit loss values, the multivariate GRU significantly outperforms all the other models, including univariate CNN, which achieved the best results on MAE.

7. Conclusion

The liberalization of the electricity market led to the reduction of the electricity price and improved efficiency by determining prices based on supply and demand forces on the market. Since electricity is characterized by high volatility and peaks, it significantly differs from other commodities traded on the financial exchange market. Thus an accurate price forecast is fundamental to market participants. Next to the consumption, which affects the prices significantly, various factors such as weather conditions, weekly seasonality due to differences in consumption during weekdays, weekends, and holidays are strongly correlated with the electricity price. Even though many studies have been developed to analyze and forecast the electricity price efficiently, it is difficult to achieve high accuracy of forecasts. The majority of the EPF models are adaptations of popular models for financial and econometrics applications, with varying degrees of success. Thus, this study investigates the predictive capabilities using different approaches, mainly deep learning models, in the problem of EPF in the German day-ahead market.

We evaluated the forecast accuracy and provided the accuracy metrics and statistical testing results. By using our forecasting results in Table 5 and Table 6, we show that neural network based methods outperform statistical and machine learning methods in the application of short decision time-frames. We used both univariate and multivariate configurations of the MLP, CNN, LSTM, and GRU to estimate the electricity prices. According to our results, GRU is the best performing in statistically significant terms and terms of RMSE compared to CNN and MLP. Due to its memory about the previous time steps, which is crucial for estimating electricity prices of the day-ahead market, makes them the method of choice for time series problems. Another significant observation is that GRU perform better and train faster than LSTM due to the fewer parameters needed to be learned. One additional comparison we made was comparing different renewable energies productions for individual hours. The results show that most negative prices occur when wind and photovoltaic production is the highest.

In this study, ANNs have been investigated for accurate electricity price estimation. We have conducted a detailed analysis of time series forecasting methods and highlighted that GRUs can solve these problems with high accuracy compared to various neural network-based methods and statistical techniques.

Based on the thesis, further research can be conducted to find a means to solve the problem of volatility. For this paper, we have adjusted for widely divergent prices using the third standard deviation. This eliminated highly deviant prices, which, according to some authors, may occur more frequently in the future. Therefore, one can build a model which focuses on spikes to predict a volatile environment efficiently. In addition, natural language processing methods can be used to understand the impact of news on the EPEX Spot intraday and day-ahead price. To the best of our knowledge, there have not been studies using natural language processing on the German electricity market.

Furthermore, hybrid methods can combine deep learning with statistical models to simulate various characteristics of electricity prices. According to Chollet (2018), the testing and validating results of the ANNs will only become better with more data. Since the structure of hidden layers enables the RNNs to store, remember, and process past complex information for a very long period, one might increase the number of inputs related to the price dynamic of the German electricity market (Salehinejad et al., 2018). They are adding information of external markets directly connected to Germany, such as neighboring countries, to improve the predictive accuracy of the proposed models. If many potential features are considered to train the models, it would be advisable to add a feature selection step to only keep the most informative ones. Future researchers could therefore also consider these approaches.

EPF have become increasingly important in everyday business of power utilities due to the rise of more competitive electricity markets. Thus, an accurate foreasting model is essential to increase the effectiveness of all market players and maximize their profits and hedge against risk. Not only producers and buyers will profit with such a forecasting model, but regulators will also gain from the results since regulatory changes and policy implications in such a volatile and constantly changing market are highly needed. On basis of the results, all market participants and market regulators can discuss not only the short-term prices, but also the long-term prices, and much more importantly the effects of demand optimisation, as well as new business models, such as flexibility markets. Regulators may also take into account developments following the global economic and financial crisis and the recent global health crisis (COVID-19) to prevent the likelihood of potential extreme events that could have huge consequences for the whole business.

References

Adhikari, R. & Agrawal, R. K. (2013). An Introductory Study on Time Series Modeling and Forecasting. arXiv:1302.6613 [cs, stat].

Aggarwal, S. K., Saini, L. M., & Kumar, A. (2009). Electricity price forecasting in deregulated markets: A review and evaluation. International Journal of Electrical Power and Energy Systems, 31, 13–22.

Akay, T. (2015), Forecasting Stylised Features of Electricity Prices in the Australian National Electricity Market (Unpublished Doctoral Dissertation). RMIT University, Melbourne, Australia.

Álvarez, F., Troncoso, A., Riquelme, J. C. & Riquelme, J. M. (2007). Discovering patterns in electricity price using clustering techniques. Renewable Energy and Power Quality, 1 (05), 174–181. doi:10.24084/repqj05.245.

Al-Alawi, S., & Islam, S. (1996). Principles of electricity demand forecasting. I. Methodologies. Power Engineering Journal, 10, 139-143.

Amjady, N. (2006). Day-Ahead Price Forecasting of Electricity Markets by a New Fuzzy Neural Network. Power Systems, IEEE Transactions on, 21, 887–896. doi:10.1109/TPWRS.2006.873409.

Amjady, N. & Hemmati, M. (2006). Energy price forecasting - problems and proposals for such predictions, in IEEE Power and Energy Magazine, vol. 4, no. 2, pp. 20-29, March-April 2006, doi: 10.1109/MPAE.1597990.

Arbel, N. (2018). How LSTM networks solve the problem of vanishing gradients. Medium. Zugriff am 18.10.2021. Verfügbar unter: https://medium.datadriveninvestor.com/how-do-lstm-networks-solve-the-problem-of-vanishing-gradients-a6784971a577.

Arel, I., Rose, D. C., & Karnowski, T. P. 2010). Deep machine learning-a new frontier in artificial intelligence research [research frontier]. IEEE computational intelligence magazine 5 (4): 13-18.

Azevedo, F. & Vale, Z. A. (2006). Forecasting Electricity Prices with Historical Statistical Information using Neural Networks and Clustering Techniques. IEEE PES Power Systems Conference and Exposition, pp. 44-50, doi: 10.1109/PSCE.2006.296247.

Bahdanau D, Cho K, Bengio Y (2015) Neural Machine Translation by Jointly Learning to Align and Translate. In: International Conference on Learning Representations.

Bakar, N. A. & Rosbi, S. (2017). Autoregressive Integrated Moving Average (ARIMA) Model for Forecasting Cryptocurrency Exchange Rate in High Volatility Environment: A New Insight of Bitcoin Transaction. International Journal of Advanced Engineering Research and Science, 4 (11), 130–137. doi:10.22161/ijaers.4.11.20.

Balas, V. E., Mishra, B. K. & Kumar, R. (2020). Handbook of Deep Learning in Biomedical Engineering: Techniques and Applications. Academic Press.

Balkin, Y. D. & Ord, J. K. (2000). Automatic neural network modeling for univariate time series. International Journal of Forecasting, 16, 509–515.

Barbour, E., Wilson, IA. G, Hall, P. & Radcliffe, J. (2014). Can negative electricity prices encourage inefficient electrical energy storage devices?. International Journal of Environmental Studies. 71. 10.1080/00207233.2014.966968.

Baumohl, E. & Lyocsa, S. (2009). Stationarity of Time Series and the Problem of Spurious Regression. SSRN Electronic Journal. doi:10.2139/ssrn.1480682.

Beccali M, Cellura M, Lo Brano V & Marvuglia A. (2008). Short-term prediction of household electricity consumption: assessing weather sensitivity in a Mediterranean area. Renew Sustain Energy Rev 2008;12(8):2040–65. http://dx.doi.org/10.1016/j.rser. 2007.04.010.

Bengio, Y., Simard, P. and Frasconi, P. (1994). “Learning long-term dependencies with gradient descent is difficult,” IEEE transactions on neural networks, vol. 5, no. 2, pp. 157–166.

Bianchi, F. M., Maiorino, E., Kampffmeyer, M. C., Rizzi, A. & Jenssen, R. (2017). Recurrent Neural Network Architectures (SpringerBriefs in Computer Science). Recurrent Neural Networks for Short-Term Load Forecasting (S. 23–29). Cham: Springer International Publishing. doi:10.1007/978-3-319-70338-1_3.

Bierbrauer, M., Menn, C., Rachev, S. & Trück, S. (2007). Spot and derivative pricing in the EEX power market. Journal of Banking & Finance 31 (11), 3462–3485.

Blais, J. A. R. (2013). “Optimal modeling and filtering of stochastic time series for geoscience applications,” Mathematical Problems in Engineering, vol. 2013, Article ID 895061, 8 pages.

Botnen Holm, T. (2017). The future importance of short term markets: An analyse of intraday prices in the Nordic intraday market; Elbas (Master’s Thesis). Norwegian University of Life Sciences (NMBU).

Box, G.E.P. & Jenkins, G. (1970). “Time Series Analysis, Forecasting and Control”, Holden-Day, San Francisco, CA.

Brownlee, J. (2018). How to Grid Search Naive Methods for Univariate Time Series Forecasting. Machine Learning Mastery. Accessed 4th October 2021. Available at: https://machinelearningmastery.com/how-to-grid-search-naive-methods-for-univariate-time-series-forecasting/.

Brownlee, J. (2019). How to use Data Scaling Improve Deep Learning Model Stability and Performance. Machine Learning Mastery. Accessed 28th September 2021. Available at: https://machinelearningmastery.com/how-to-improve-neural-network-stability-and-modeling-performance-with-data-scaling/.

Brownlee, J. (2018). Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python. Machine Learning Mastery.

Bunn, D.W. (2004). Modelling prices in competitive electricity markets. United Kingdom: John Wiley.

Bunn, D. W., & Martoccia, M. (2005). Unilateral and collusive market power in the electricity pool of England and Wales. Energy Economics, 7, 305-315.

Caterini A.L., Chang D.E. (2018) Recurrent Neural Networks. In: Deep Neural Networks in a Mathematical Framework. SpringerBriefs in Computer Science. Springer, Cham. https://doi.org/10.1007/978-3-319-75304-1_5.

Chai, T. & Draxler, R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? Geosci. Model Dev., 7. doi:10.5194/gmdd-7-1525-2014.

Chan, N. H. (2011). Autoregressive Moving Average Models (Wiley Series in Probability and Statistics). Time Series (S. 23–37). Hoboken, NJ, USA: John Wiley & Sons, Inc. doi:10.1002/9781118032466.ch3.

Chen, J. C., Rubin, E. A. & Cornwall, G. J. (2021). Data Science for Public Policy (Springer Series in the Data Sciences). Cham: Springer International Publishing. doi:10.1007/978-3-030-71352-2.

Chen, H., Wan, Q. & Wang, Y. (2014). Refined Diebold-Mariano Test Methods for the Evaluation of Wind Power Forecasting Models. Energies, 7 (7), 4185–4198. doi:10.3390/en7074185.

Cho, K, Van Merriënboer, B, Gulcehre C, Bahdanau, D, Bougares, F, Schwenk, H, Bengio Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation, arXiv preprint arXiv:1406.1078.

Chollet, F. (2018). Deep Learning with Python. Shelter Island, New York: Manning Publications Co.

Chung, J., Gulcehre, C., Cho, K. & Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. arXiv:1412.3555 [cs].

Cochrane, J.H. (2005) Time Series for Macroeconomics and Finance. Manuscript. University of Chicago.

Cuaresma, J., Hlouskova, J., Kossmeier, S. & Obersteiner, M. (2004). Forecasting electricity spot-prices using linear univariate time-series models. Applied Energy, 77 (1), 87–106. Elsevier.

Cutler, N. J., Boerema, N. D., MacGill, I. F. & Outhred, H. R. (2011). High penetration wind generation impacts on spot prices in the Australian national electricity market (Sustainability of biofuels). Energy Policy, 39 (10), 5939–5949. doi:10.1016/j.enpol.2011.06.053.

De Gooijer, J. G. & Hyndman, R. J. (2006). 25 years of time series forecasting (Twenty five years of forecasting). International Journal of Forecasting, 22 (3), 443–473. doi:10.1016/j.ijforecast.2006.01.001.

Dehalwar, V., Kalam, A., Kolhe, M. & Zayegh, A. (2016). Electricity load forecasting for Urban area using weather forecast information (S. 359). doi:10.1109/ICPRE.2016.7871231.

Dolado, J. J., Gonzalo, J., & Mayoral, L. (2002). A Fractional Dickey-Fuller Test for Unit Roots. Econometrica, 70(5), 1963–2006. http://www.jstor.org/stable/3082026.

Donges, N. (2019). Recurrent Neural Networks (RNN): What It Is & How It Works | Built In. Accessed 18th October 2021Available at: https://builtin.com/data-science/recurrent-neural-networks-and-lstm.

Diebold, F. X. & Mariano, R. S. (1995). Comparing Predictive Accuracy, 41.

Dissanayake, Bhanuka, Hemachandra, Osanda, Lakshitha, Nuwan, Haputhanthri, Dilantha, & Wijayasiri, Adeesha. (2021). A Comparison of ARIMAX, VAR and LSTM on Multivariate Short-Term Traffic Volume Forecasting. Proceedings of the 28th FRUCT Conference, 28, 564–570. https://doi.org/10.5281/zenodo.4514955.

Duda, J. (2019). Exploiting statistical dependencies of time series with hierarchical correlation reconstruction. arXiv:1807.04119 [cs, stat].

Edgmand, M. R., Moomaw, R. L. & Olson, K. W. (1996). Economics and contemporary issues. Fort Worth: Dryden Press.

ENTSO-E (2021). Accessed 19th November 2021. Available at: https://eepublicdownloads.entsoe.eu/clean-documents/pre2015/publications/ce/Load_and_Consumption_Data.pdf.

Eren, L, Ince, T, Kiranyaz, S. (2019). A generic intelligent bearing fault diagnosis system using compact adaptive 1D CNN classifier. Journal of Signal Processing Systems 91(2): 179–189.

EirGrid (2020). EirGrid Reveals Record Breaking WInd Enery Levels. Accessed 20th September 2021. Available at: https://www.eirgridgroup.com/newsroom/record-wind-levels-feb-20/.

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

Fawaz, I. H., Forestier, G., Weber, J., Idoumghar, L. & Muller, P.-A. (2019). Deep learning for time series classification: a review. Data Mining and Knowledge Discovery, 33 (4), 917–963. doi:10.1007/s10618-019-00619-1.

Fidalgo, J. N. & Matos, M. A. (2007). Forecasting portugal global load with artificial neural networks. In International Conference on Artificial Neural Networks, pages 728– 737. Springer.

Gamboa, J. C. B. (2017). Deep Learning for Time-Series Analysis. arXiv:1701.01887 [cs].

Gao, F., X. Guan, X.R. Cao, A. Papalexopoulos (2000). "Forecasting Power Market Clearing Price and Quantity Using a Neural Network Method" IEEE PES Winter Meeting, pp: 2183-2188.

Garcia, R. C., Contreras, J., Akkeren, M. V. and Garcia, J. B. C. (2003). "A GARCH forecasting model to predict day, ahead electricity prices," in Proc. Work. Applied Infrastructure, Berlin, Germany.

Genoese, F., Genoese, M. & Wietschel, M. (2010). Occurrence of negative prices on the German spot market for electricity and their influence on balancing power markets. Presented at the 7th International Conference on the European Energy Market, Madrid, 23-25 June.

Gianfreda, A. (2010). Volatility and volume effects in European electricity spot markets. Economic Notes 39 (1), 47–63.

Gianfrea, A., Parisio, L. & Pelagatti, M. (2016). The impact of RES in the Italien day.ahead and balancing markets. Energy Journal, 37:161-268.

Girish, G. P., Panda, A. K., & Rath, B. N. (2013). Indian Electricity Market. Global Business and Economics Anthology, 1, 180–191.

Girish, G. P., Vijayalakshmi, S. (2013). Determinants of Electricity Price in Competitive Power Market. International Journal of Business and Management, 8(21), 70-75.

Girish, G. P., Vijayalakshmi, S., Ajaya, K., & Badri, R. (2013). Forecasting Electricity Prices in Deregulated Wholesale Spot Electricity Market: A Review, 4(1), 32-42.

Gountis, V. P. & Bakirtzis, A. G. (2004). Bidding Strategies for Electricity Producers in a Competitive Electricity Marketplace. IEEE Transactions on Power Systems, 19 (1), 356–365. doi:10.1109/TPWRS.2003.821474.

Green, R., Staffell, I. & Vasilakos, N. (2014). Divide and Conquer? k-Means Clustering of Demand Data Allows Rapid and Accurate Simulations of the British Electricity System. IEEE Transactions on Engineering Management, 61, 251–260. doi:10.1109/TEM.2013.2284386.

Greene, W. H. (2003). Econometric Analysis. Engelwood Cliffs. Prentice Hall.

Gruslys, A., Munos, R., Danihelka, I., Lanctot, M. & Graves, A. (2016). Memory-Efficient Backpropagation Through Time. Advances in Neural Information Processing Systems (Band 29). Curran Associates, Inc.

Gonzalez, A. M., San Roque, A. M. and Gonzalez, J. G. (2005). "Modeling and forecasting electricity prices with input/output hidden Markov models," IEEE Trans. Power Syst., vol. 20, no. 1, pp. 13–24.

Gonzalez-Salazar, M. A., Kirsten, T. & Prchlik, L. (2018). Review of the operational flexibility and emissions of gas- and coal-fired power plants in a future with growing renewables. Renewable and Sustainable Energy Reviews, 82, 1497–1513. doi:10.1016/j.rser.2017.05.278.

Gupta S, Girshick R, Arbeláez P, Malik J (2014). Learning rich features from RGB-D images for object detection and segmentation. In: Proceedings of the ECCV. Springer, pp 345–360.

Hagemann, S., & Weber, C. (2013). An Empirical Analysis of Liquidity and its Determinants in the German Intraday Market for Electricity (EWL Working Paper). University of Duisburg-Essen.

Han, J. & Kamber, M. (2012). Data mining: concepts and techniques (3rd ed.). Burlington, MA: Elsevier.

Harasheh, M. (2016) Forecasting Wholesale Electricity Prices With Artificial Intelligence Models: The Italian Case. Preprints 2016, 2016070001, doi: 10.20944/preprints201607.0001.v1.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Elements of Statistical Learning – Data Mining, Inference, and Prediction (2nd ed.). Springer.

Hendry, D. & Nielsen, B. (2007). Econometric Modeling: A Likelihood Approach. Econometric Modeling: A Likelihood Approach.

Hernández-Callejo, L., Baladrón Zorita, C., Aguiar, J., Carro, B., Sanchez-Esguevillas, A., Lloret, J. et al. (2014). A Survey on Electric Power Demand Forecasting: Future Trends in Smart Grids, Microgrids and Smart Buildings. Communications Surveys & Tutorials, IEEE, 16, 1460–1495. doi:10.1109/SURV.2014.032014.00094.

Hinman, J. & Hickey, E (2009). Modeling and forecasting short-term electricity load using regression analysis. Journal of IInstitute for Regulatory Policy Studies.

Hipel, K.W. & McLeod, A.I. (1994). Time Series Modelling of Water Resources and Environmental Systems. Amsterdam, Elsevier.

Hong, T., Pinson, P., & Fan, S. (2014). Global Energy Forecasting Competition 2012. International Journal of Forecasting, 30(2), 357–363.

Hong, T. & Shahidehpour, M. (2015). Load forecasting case study. EISPC, US Department of Energy.

Hoseinzade, E. & Haratizadeh, S. (2018). CNNPred: CNN-based stock market prediction using several data sources. arXiv:1810.08923 [cs, q-fin, stat].

Hu, Z., Y. Yu, Z. Wang, W. Sun, D. Gan, Z. Han "Price Forecasting Using an Integrated Approach" IEEE Int. Conf. on Electric Utility, April 2004, Hong Kong.

Huneke, A. F., Claußner, M., Fernahl, A. & Linkenheil, C. P. (2021). Im Auftrag des Bundesministeriums für Wirtschaft und Energie, 32.

IEA (2021). What is behind soaring energy prices and what happens next?, IEA, Paris Accessed 30th October 2021. Available at: https://www.iea.org/commentaries/what-is-behind-soaring-energy-prices-and-what-happens-next.

Irena (2020), Innovation landscape brief: Peer-to-peer electricity trading, International Renewable Energy Agency, Abu Dhabi.

Joshi, R. C. & Gupta, B. B. (Hrsg.). (2020). Security, Privacy, and Forensics Issues in Big Data: (Advances in Information Security, Privacy, and Ethics). IGI Global. doi:10.4018/978-1-5225-9742-1.

Józefowicz, R., Zaremba, W., & Sutskever, I. (2015). An Empirical Exploration of Recurrent Network Architectures. ICML.

Jun. (2021). Weight Initialization and Batch Normalization. jun-devpBlog. Accessed 28th September 2021. Available at: https://medium.com/jun-devpblog/dl-5-weight-initialization-and-batch-normalization-7442e1246e0f.

Karakatsani, N. V., & Bunn, D. W. (2008). Forecasting electricity prices: the impact of fundamentals and time-varying coefficients. International Journal of Forecasting, 24(4), 764–785.

Karazi, S. M., Moradi, M. & Benyounis, K. Y. (2019). Statistical and Numerical Approaches for Modeling and Optimizing Laser Micromachining Process-Review. Reference Module in Materials Science and Materials Engineering (S. B9780128035818117000). Elsevier. doi:10.1016/B978-0-12-803581-8.11650-9.

Keles, D., Genoese, M., Möst, D. & Fichtner, W. (2012). Comparison of extended mean-reversion and time series models for electricity spot price simulation considering negative prices. Energy Economics, 34 (4), 1012–1032. doi:10.1016/j.eneco.2011.08.012.

Kern, T. (2020). European day-ahead electricity prices in 2020.

Khan, Z. A. & Jayaweera, D. (2018). Approach for forecasting smart customer demand with significant energy demand variability. 2018 1st International Conference on Power, Energy and Smart Grid (ICPESG) (S. 1–5). Mirpur Azad Kashmir: IEEE. doi:10.1109/ICPESG.2018.8384528.

Kotzé, K. (2021). Nonstationarity. Accessed 30th December 2021. Available at: https://kevinkotze.github.io/ts-6-unit-roots/.

Ku, Anne (2002). Forecasting to Understand Uncertainty in Electricity Prices. Platts Energy Business and Technology.

Kumar, Er. Parveen & Sharma, Er. Pooja (2014). Artificial Neural Networks-A Study International Journal Of Emerging Engineering Research And Technology Volume 2, Issue 2, May 2014, Pp 143-148.

Klitgaard, T. & Reddy, R. (2000). Lowering Electricity Prices through Deregulation, 6.

Lago, J., De Ridder, F. and De Schutter, B. (2018), Erratum to forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms [appl. energy 221 (2018) 386405], Applied Energy 229.

Lago, J., Marcjasz, G., De Schutter, B. & Weron, R. (2021). Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark. Applied Energy, 293, 116983. doi:10.1016/j.apenergy.2021.116983.

Lara-Benítez, Pedro & Carranza-García, Manuel & Riquelme, José. (2020). An Experimental Review on Deep Learning Architectures for Time Series Forecasting.International Journal of Neural Systems. 31. 10.1142/S0129065721300011.

Le, Q. V., Jaitly, N. and Hinton, G. E (2015). "A simple way to initialize recurrent networks of rectified linear units," arXiv preprint arXiv:1504.00941.

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521:436–444

Lecun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998). Gradient-based learning applied to document recognition, in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, doi: 10.1109/5.726791.

Li, J., Liu, H., Wang, D. & Bi, T. (2021). Classification of Power Quality Disturbance Based on S-Transform and Convolution Neural Network. Frontiers in Energy Research, 9, 708131. doi:10.3389/fenrg.2021.708131.

Lin, H. & Jegelka, S. (2018). ResNet with one-neuron hidden layers is a Universal Approximator. Advances in Neural Information Processing Systems (Band 31). Curran Associates, Inc. Accessed 19th November 2021. Available at: https://proceedings.neurips.cc/paper/2018/hash/03bfc1d4783966c69cc6aef8247e0103-Abstract.html.

Liu, C., Wang, Z., Fu, H. & Zhang, Y. (2014). A Novel Approach for Nonstationary Time Series Analysis with Time-Invariant Correlation Coefficient. Mathematical Problems in Engineering, 2014, e148432. Hindawi. doi:10.1155/2014/148432.

Luo, X., Shen, R., Hu, J., Deng, J., Hu, L., and Guan, Q. (2017). A Deep Convolution Neural Network Model for Vehicle Recognition and Face Recognition. Procedia Computer Science 107: 715-720.

Lumen (2021). Examples of Elastic and Inelastic Demand. Provided by: Lumen Learning. Accessed 30th September 2021. Available at: https://courses.lumenlearning.com/wmopen-microeconomics/chapter/examples-of-elastic-and-inelastic-demand/.

Lyocsa, S., Vyrost, T. & Baumohl, E. (2011). Unit-Root and Stationarity Testing with Empirical Application on Industrial Production of CEE-4 Countries. SSRN Electronic Journal. doi:10.2139/ssrn.1785223.

Maciejowska, K. & Weron, R. (2016). "Short- and Mid-Term Forecasting of Baseload Electricity Prices in the U.K.: The Impact of Intra-Day Price Relationships and Market Fundamentals," in IEEE Transactions on Power Systems, vol. 31, no. 2, pp. 994-1005, doi: 10.1109/TPWRS.2015.2416433.

Mayo, M. (2020). Dataset Splitting Best Practices in Python. KDnuggets. Accessed 29th September 2021. Available at: https://www.kdnuggets.com/dataset-splitting-best-practices-in-python.html/.

McMenamin, J. S. & Monforte, F. A. (2000). Statistical Approaches to Electricity Price Forecasting. In A. Faruqui & K. Eakin (Hrsg.), Pricing in Competitive Electricity Markets (S. 249–263). Boston, MA: Springer US. doi:10.1007/978-1-4615-4529-3_15.

Mugele, C., Rachev, S. & Trück, S. (2005). Stable modeling of different European power markets. Investment Management and Financial Innovations 2 (3), 65–85.

Meller, B., Matula, K., & Chłąd, P. (2020). Image classification with Feed-Forward Neural Networks.

Mikolov, T., Joulin, A., Chopra, M, S (2014). Mathieu, and M. Ranzato, “Learning longer memory in recurrent neural networks,” arXiv preprint arXiv:1412.7753.

Nabi, K. N., Tahmid, M. T., Rafi, A., Kader, M. E., & Haider, M. A. (2021). Forecasting COVID-19 cases: A comparative analysis between recurrent and convolutional neural networks. Results in physics, 24, 104137. https://doi.org/10.1016/j.rinp.2021.104137.

Nagayama, H. & Kashiwagi, T. (2007). Evaluating electricity sector reforms in Argentina: lessons for developing countries? Journal of Cleaner Production, 15 (2), 115–130. doi:10.1016/j.jclepro.2005.11.056.

Nebauer, C. (1998). Evaluation of convolutional neural networks for visual recognition. IEEE Transactions on Neural Networks 9 (4): 685-696.

Noor-Ul-Amin, M. (2010). Forecasting with Neural Networks: A comparative study using the data of emergency service. arXiv:1010.3501 [stat].

Nti, I. K., Samuel, A.-A. & Michael, A. (2019). Predicting Monthly Electricity Demand Using Soft-Computing Technique, 06 (06), 7.

Nwankpa, C., Ijomah, W., Gachagan, A. & Marshall, S. (2018). Activation Functions: Comparison of trends in Practice and Research for Deep Learning. arXiv:1811.03378 [cs].

OECD (2007). Data and Metadata Reporting and Presentation Handbook. OECD. doi:10.1787/9789264030336-en.

Olah, C. (2015). Understanding LSTM Networks -- colah’s blog. Accessed 15th October 2021. Available at: https://colah.github.io/posts/2015-08-Understanding-LSTMs/.

Oliveira, T., Barbar, J. & Soares, A (2014). Multilayer Perceptron and Stacked Autoencoder for Internet Traffic Prediction. 11th IFIP International Conference on Network and Parallel Computing (NPC), Ilan, Taiwan. pp.61-71, ff10.1007/978-3-662-44917-2_6ff. ffhal-01403065f.

OpenStax Economics, Principles of Economics. OpenStax CNX. (2016) http://cnx.org/contents/69619d2b-68f0-44b0-b074-a9b2bf90b2c6@11.330.

Pascanu, R., Mikolov, T., Bengio, Y. (2013). On the difficulty of training Recurrent Neural Networks, arXiv:1211.5063.

Permatasari, C. I., Sutopo, W. & Hisjam, Muh. (2018). Sales forecasting newspaper with ARIMA: A case study (S. 030017). Gehalten auf der THE 1ST INTERNATIONAL CONFERENCE AND EXHIBITION ON POWDER TECHNOLOGY INDONESIA (ICePTi) 2017, Jatinangor, Indonesia. doi:10.1063/1.5024076.

Petneházi, G. (2018). Recurrent Neural Networks for Time Series Forecasting. arXiv:1901.00069 [cs, stat].

Podewils, C. (2014). Negative Strompreise werden häufiger. Stiftung Mercator. Accessed 21th September 2021. Available at: https://www.stiftung-mercator.de/de/pressemitteilungen/negative-strompreise-werden-haeufiger/.

Poznyak, T., Chairez, I. & Poznyak, A. (2018). Ozonation and Biodegradation in Environmental Engineering: Dynamic Neural Network Approach. Elsevier.

Ragab, M., Jadid Abdulkadir, S., Aziz, N., Al-Tashi, Q., Alyousifi, Y., Alhussian, H. et al. (2020). A Novel One-Dimensional CNN with Exponential Adaptive Gradients for Air Pollution Index Prediction. Sustainability, 12, 10090. doi:10.3390/su122310090.

Ramentol, E., Schirra, F. & Wagner, A. (2020). Short- and long-term forecasting of electricity prices using embedding of calendar information in neural networks. arXiv:2007.13530 [stat].

Rawat, W. & Wang, Z. (2017). Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. Neural Computation, 29, 1–98. doi:10.1162/NECO_a_00990.

Rudrappa, Gujanatti. (2021). Machine Learning Models Applied for Rainfall Prediction. Revista Gestão Inovação e Tecnologias. 11. 179-187. 10.47059/revistageintec.v11i3.1926.

Saini, D.K., Saxena, A., & Bansal, R.C. (2016). Electricity price forecasting by linear regression and SVM. 2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE), 1-7.

Ruiz, P. A. & Gross, G. (2008). Short-Term Resource Adequacy in Electricity Market Design. IEEE Transactions on Power Systems, 23 (3), 916–926. doi:10.1109/TPWRS.2008.926094.

Salehinejad, H., Sankar, S., Barfett, J., Colak, E. & Valaee, S. (2018). Recent Advances in Recurrent Neural Networks. arXiv:1801.01078 [cs].

Sazli, M. H. (2006). A brief review of feed-forward neural networks. Communications, Faculty Of Science, University of Ankara, 11–17. doi:10.1501/0003168.

Scharff, R., & Amelin, M. (2016). Trading behaviour on the continuous intraday market ELBAS. Energy Policy, 88, 544–557.

Sensfuß, F., Ragwitz, M. & Genoese, M. (2008). The merit-order effect: A detailed analysis of the price effect of renewable electricity generation on spot market prices in Germany. Energy Policy, 36 (8), 3086–3094. doi:10.1016/j.enpol.2008.03.035.

Shah, D. & Chatterjee, S. (2020). A comprehensive review on day-ahead electricity market and important features of world's major electric power exchanges. International Transactions on Electrical Energy Systems, 30 (7), e12360. doi:10.1002/2050-7038.12360.

Shahidehpour, M., Yamin, H., & Li, Z. (2002). Market operations in electric power systems: forecasting, scheduling, and risk management. Wiley.

Sharma, S., Sharma, S., Scholar, U. & Athaiya, A. (2020). ACTIVATION FUNCTIONS IN NEURAL NETWORKS, 4 (12), 7.

Shen, X., Tian, X., He, A., Sun, S. & Tao, D. (2019). Transform-Invariant Convolutional Neural Networks for Image Classification and Search. arXiv:1912.01447 [cs, eess, stat].

Shumway, R H., & Stoffer, David S. (2010). Time Series Analysis and its Applications (3rd Edition). Springer.

Singh, J. & Banerjee, R. (2019). A Study on Single and Multi-layer Perceptron Neural Network," 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), pp. 35-40, doi: 10.1109/ICCMC.2019.8819775.

Singh, N. & Mohanty, S. (2015). A review of price forecasting problem and techniques in deregulated electricity markets. Journal of Power and Energy Engineering, vol. 3(9), p. 1.

Strandberg, R. (2015). Using Time Series Analysis to Forecast Daily Municipal Water Demand.

Soloviova, M. & Vargiolu, T. (2020). Efficient representation of supply and demand curves on day-ahead electricity markets.

Soytas, U. and Sari, R. Routledge Handbook of Energy Economics. (2020). Routledge & CRC Press. Accessed 21st September 2021. Available at: https://www.routledge.com/Routledge-Handbook-of-Energy-Economics/Soytas-Sari/p/book/9781032089195.

Suganthi, L. & Samuel, A. A. (2012). Energy models for demand forecasting—A review. Renewable and Sustainable Energy Reviews, 16 (2), 1223–1240. doi:10.1016/j.rser.2011.08.014.

Suhartono. (2011). Time Series Forecasting by using Seasonal Autoregressive Integrated Moving Average: Subset, Multiplicative or Additive Model. Journal of Mathematics and Statistics, 7 (1), 20–27. doi:10.3844/jmssp.2011.20.27.

Sutskever, J. Martens, & G. E. Hinton (2011). Generating text with recurrent neural networks, in Proceedings of the 28th International Conference on Machine Learning (ICML-11), pp. 1017–1024.

Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V & Rabinovich, A (2015). Going deeper with convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 1–9.

Taylor, J. &McSharry, P (2007). Short-term load forecasting methods: An evaluation based on european data. IEEE Transactions on Power Systems, 22(4):2213– 2219.

Teusch, A. (2006). Introduction to the Spectral and Time Series Analysis with Examples from Geodesy (München: Verlag der Bayerischen Akademie der Wissenschaften).

Tian, C., Ma, J., Zhang, C. & Zhan, P. (2018). A Deep Neural Network Model for Short-Term Load Forecast Based on Long Short-Term Memory Network and Convolutional Neural Network. Energies, 11, 3493. doi:10.3390/en11123493.

Tian, C., Wang, H., & Luo, X. (2019). Time-series modelling and forecasting of hand, foot and mouth disease cases in China from 2008 to 2018. Epidemiology and Infection, 147, E82. doi:10.1017/S095026881800362X.

Torres, J. F., Hadjout, D., Sebaa, A., Martínez-Álvarez, F. & Troncoso, A. (2021). Deep Learning for Time Series Forecasting: A Survey. Big Data, 9 (1), 3–21. doi:10.1089/big.2020.0159.

Tsay, R. S. (2005). Analysis of Financial Time Series, Wiley Series in Probability and Statistics, John Wiley & Sons, New York, NY, USA, 2nd edition.

Ueckerdt, F. & Kempener, R. (2015). From Baseload to Peak: renewables provide a reliable solution, 16.

Ugurlu, U., Oksuz, I. & Tas, O. (2018). Electricity Price Forecasting Using Recurrent Neural Networks. Energies, 11 (5), 1255. doi:10.3390/en11051255.

Velicer, W. F., & Fava, J. L. (2003). Time Series Analysis. In J. Schinka & W. F. Velicer (Eds.), Research Methods in Psychology(581-606). Volume 2, Handbook of Psychology (I. B. Weiner, Editor-in-Chief.). New York: John Wiley & Sons.

Ventosa, M., Baíllo, Á., Ramos, A. & Rivier, M. (2005). Electricity market modeling trends. Energy Policy, 33 (7), 897–913. doi:10.1016/j.enpol.2003.10.013.

Verma, Y. (2021). Why Decompose a Time Series, and How? Analytics India Magazine. Accessed 19th November 2021. Available at: https://analyticsindiamag.com/why-decompose-a-time-series-and-how/.

Viviani, E., Di Persio, L. & Ehrhardt M. (2021). Energy Markets Forecasting. From Inferential Statistics to Machine Learning: The German Case. Energies; 14(2):364. https://doi.org/10.3390/en14020364.

Yu, Z., Niu, Z., Tang, W. & Wu, Q. (2019). "Deep Learning for Daily Peak Load Forecasting–A Novel Gated Recurrent Neural Network Combining Dynamic Time Warping," in IEEE Access, vol. 7, pp. 17184-17194, doi: 10.1109/ACCESS.2019.2895604.

Wang, J., Lin, J., and Wang, Z. (2016) “Efficient convolution architectures for convolutional neural network.” In Wireless Communications and Signal Processing (WCSP), 2016 8th International Conference on (pp. 1-5).

Weron, R. (2014). Electricity price forecasting: A review of the state-of-the-art with a look into the future | Elsevier Enhanced Reader, doi:10.1016/j.ijforecast.2014.08.008.

Weron, R. & Chichester, W. (2006). Modeling and forecasting electricity loads and prices: a statistical approach.

Whittington, J. & Bogacz, R. (2017). An Approximation of the Error Backpropagation Algorithm in a Predictive Coding Network with Local Hebbian Synaptic Plasticity. Neural Computation, 29, 1–34. doi:10.1162/NECO_a_00949.

Worthington, I., Britton, C. & Rees, A. (2001). Economics for business : blending theory and practice. Harlow [u.a.] : Prentice Hall [u.a.].

Wu, N., Green, B., Ben, X. & O’Banion, S. (2020). Deep Transformer Models for Time Series Forecasting: The Influenza Prevalence Case. arXiv:2001.08317 [cs, stat].

Zhang, G.P. (2003). “Time series forecasting using a hybrid ARIMA and neural network model”, Neurocomputing 50, pages: 159–175.

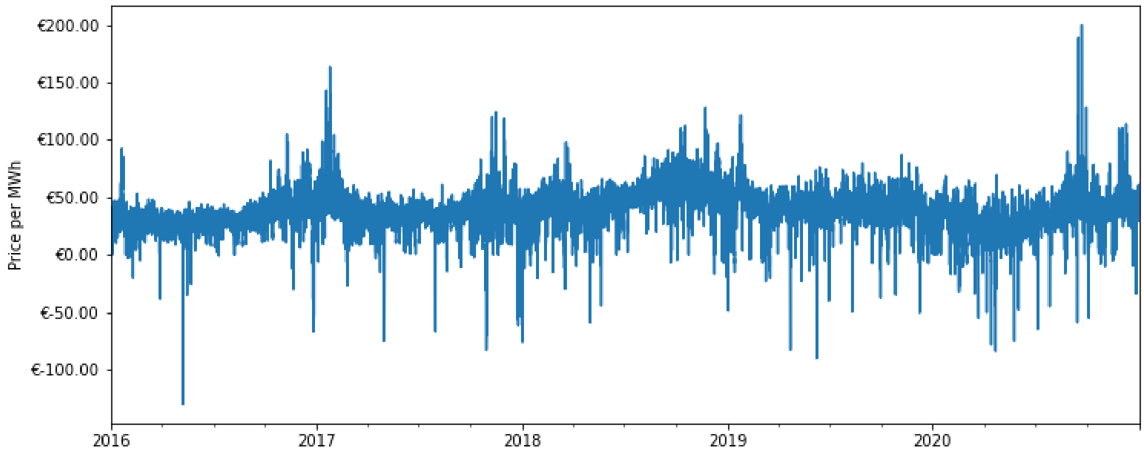
Zhang, R., Li, G. and Ma, Z. (2020). "A Deep Learning Based Hybrid Framework for Day-Ahead Electricity Price Forecasting," in IEEE Access, vol. 8, pp. 143423-143436, doi: 10.1109/ACCESS.2020.3014241.

Zhang, S., Wu, Y., Che, T., Lin, R. (2016). Memisevic, R. R. Salakhutdinov, Y. Bengio, Architectural Complexity Measuresof Recurrent Neural Networks, in: D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, R. Garnett (Eds.), Advances inNeural Information Processing Systems 29, Curran Associates, Inc., 1822–1830.

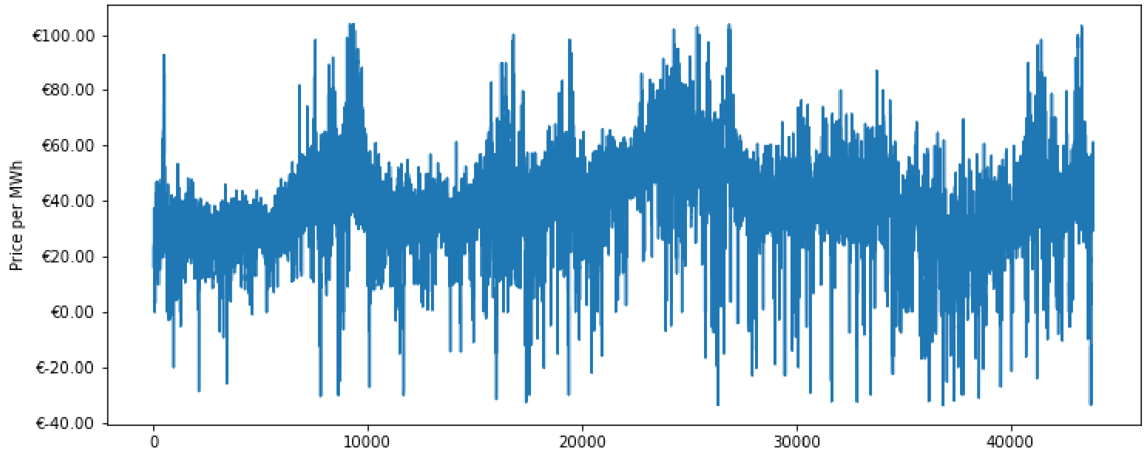
Zhirnov, V.L. (1999). The basics of price determination in electric power industry: lectures summary (NNSTU publishing, Novosibirsk).

Zhou, Y., Wang, H., Xu, F., and Jin, Y. Q. (2016) “Polarimetric SAR image classification using deep convolutional neural networks.” IEEE Geoscience and Remote Sensing Letters 13 (12): 1935-19.

Appendix



a) Time series before Outlier Treatment



b) Time series after Outlier Treatment

Figure 27. Time series before (a) and after (b) Outlier Treatment

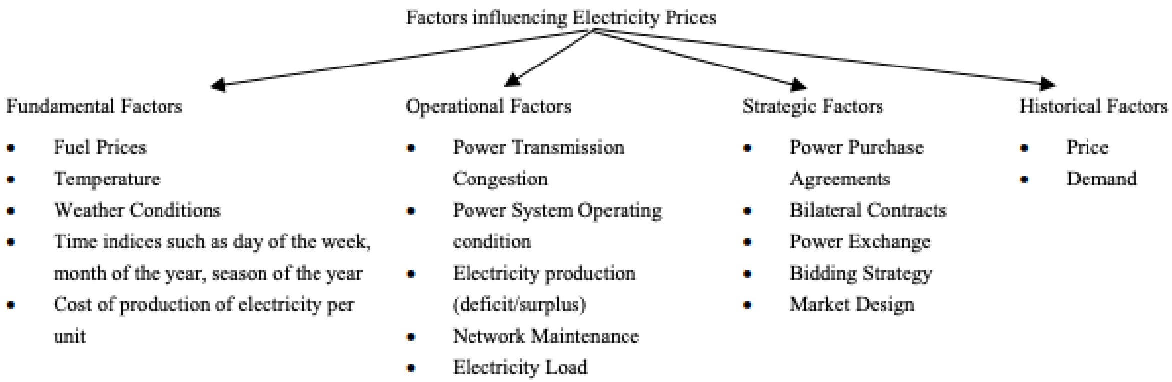


Figure 28. Factors influencing Electricity Price (Girish and Vijayalakshmi, 2013)

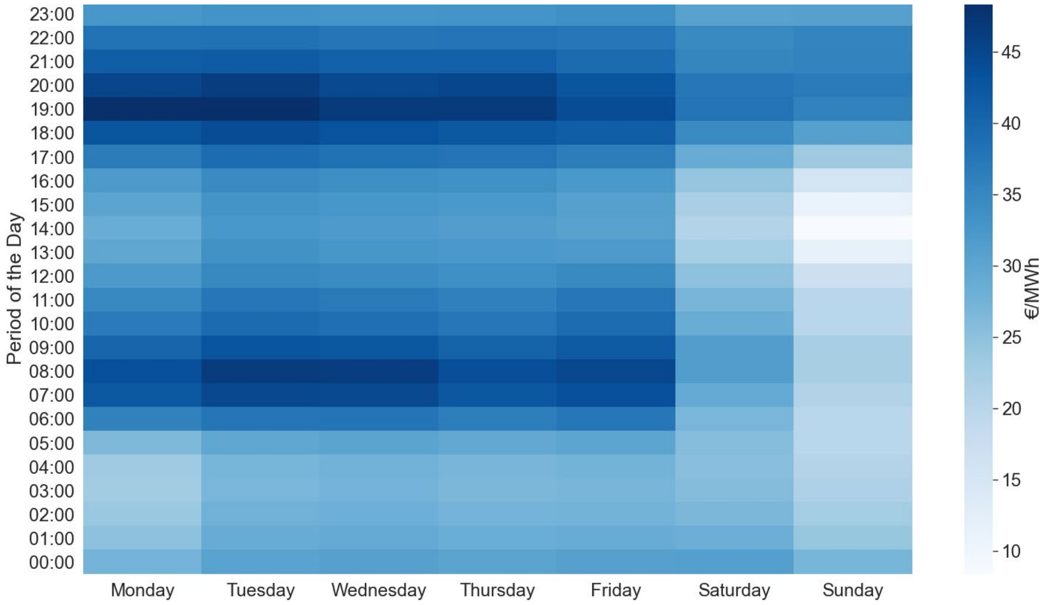


Figure 29. Mean Price for each hour of the Day – Summer

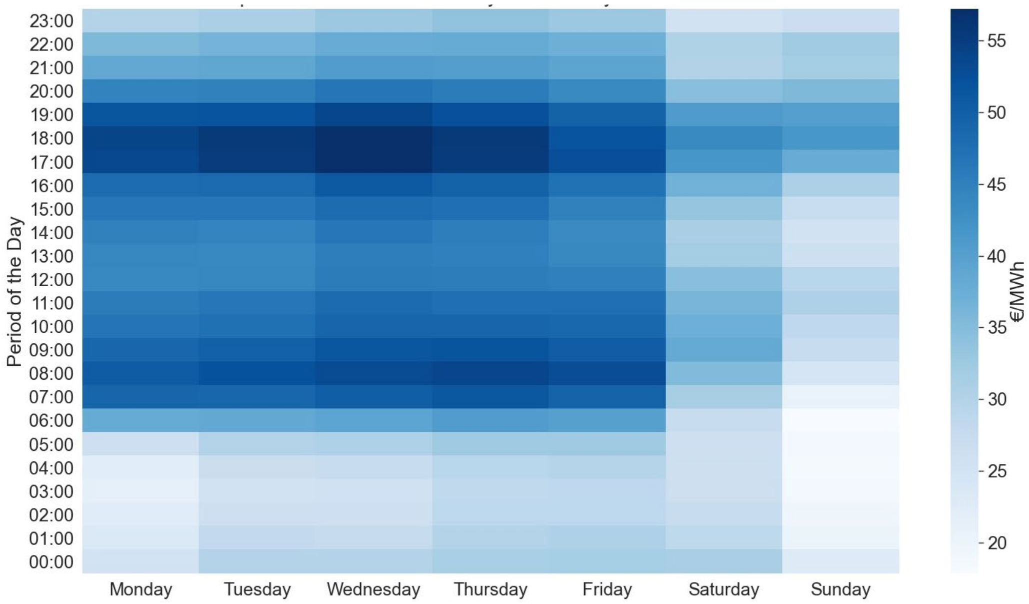


Figure 30. Mean Price for each hour of the Day – Winter

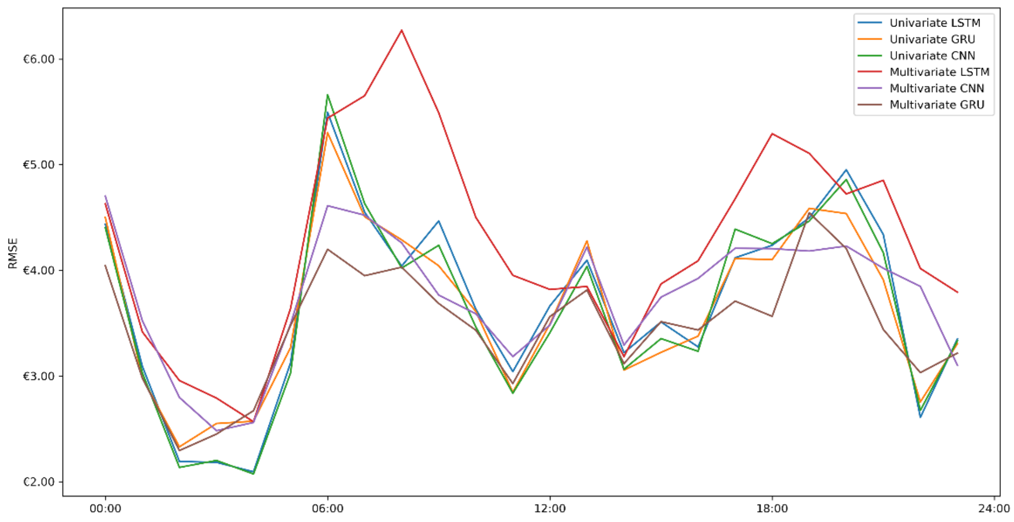


Figure 31. RMSE by hour of Day.

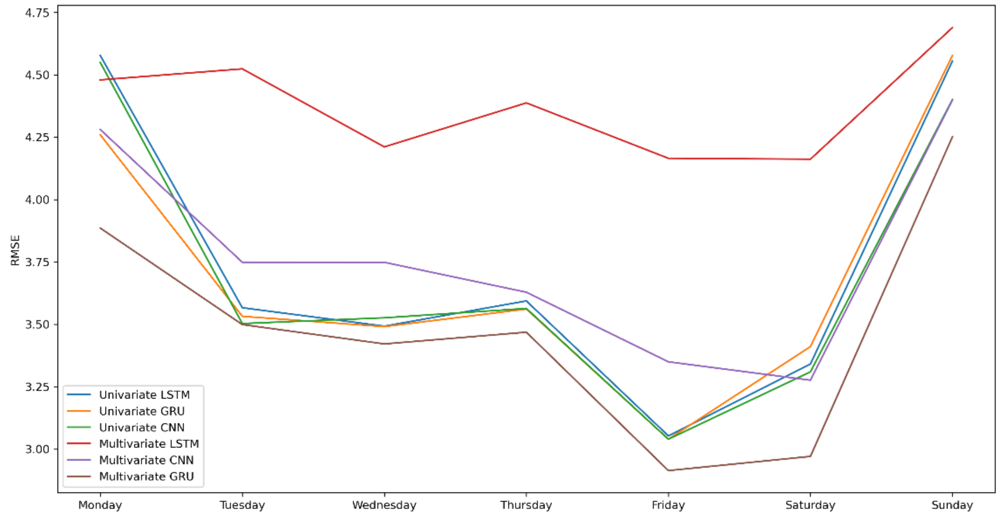


Figure 32. RMSE by Weekday.

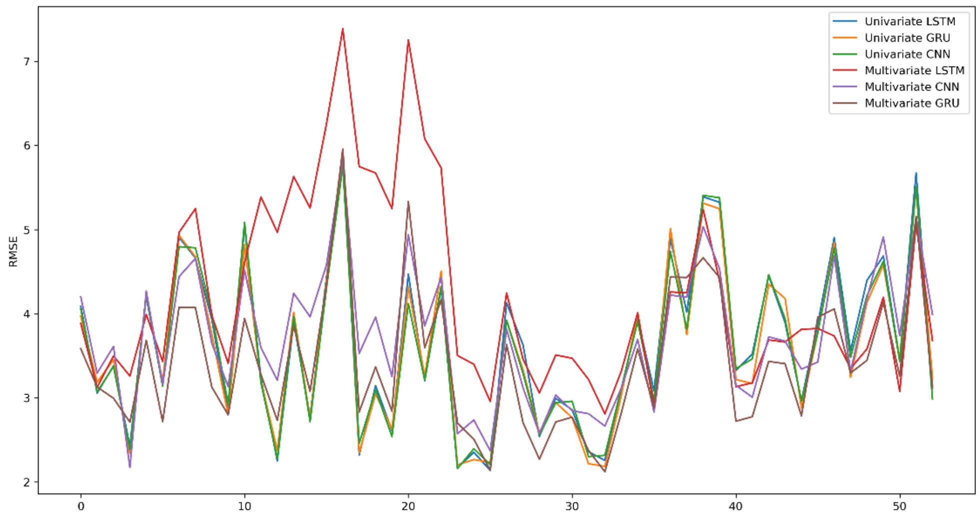


Figure 33. RMSE by Week.

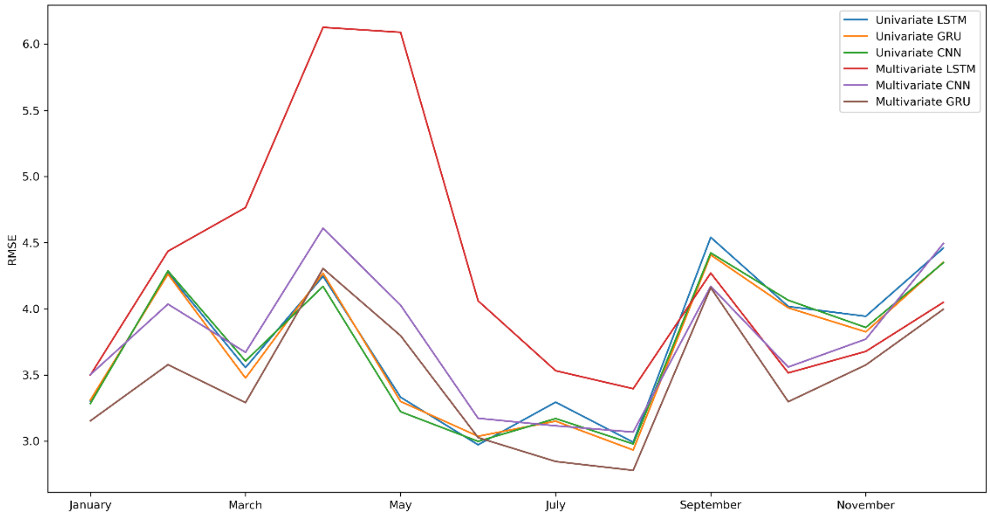


Figure 34. RMSE by Month

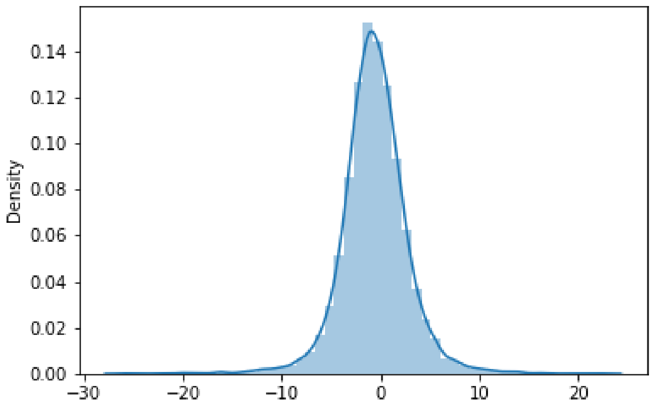


Figure 35. Error Distribution

Hour	# Neg. Prices	Biomass	Hydropower	Wind	Photovoltaics	Other_renewable
0	45	-0.15	-0.08	-0.59	0.02	-0.03
1	55	-0.15	-0.09	-0.59	0.01	0.01
2	67	-0.13	-0.08	-0.60	0.02	0.03
3	73	-0.12	-0.09	-0.60	0.02	0.04
4	64	-0.11	-0.11	-0.57	-0.04	0.04
5	43	-0.10	-0.14	-0.50	-0.07	0.05
6	46	-0.09	-0.11	-0.40	-0.09	0.06
7	31	-0.06	-0.14	-0.33	-0.14	0.09
8	21	-0.05	-0.14	-0.33	-0.17	0.09
9	25	-0.04	-0.19	-0.35	-0.23	0.09
10	34	-0.01	-0.21	-0.37	-0.28	0.11
11	32	0.01	-0.21	-0.38	-0.31	0.13
12	43	0.04	-0.21	-0.38	-0.32	0.16
13	68	0.04	-0.19	-0.36	-0.34	0.18
14	77	0.05	-0.19	-0.34	-0.36	0.20
15	63	0.07	-0.23	-0.34	-0.38	0.21
16	40	0.08	-0.27	-0.33	-0.39	0.22
17	14	0.09	-0.31	-0.30	-0.38	0.22
18	6	0.04	-0.30	-0.33	-0.30	0.16
19	2	-0.09	-0.21	-0.38	-0.16	0.03
20	7	-0.17	-0.12	-0.46	-0.02	-0.09
21	12	-0.20	-0.06	-0.51	0.04	-0.13
22	6	-0.19	-0.08	-0.52	0.02	-0.12
23	23	-0.17	-0.07	-0.56	0.02	-0.08

Table 7. Full list of Correlation between Electricity Price and Renewable Energies

Hour	Univariate LSTM	Univariate GRU	Univariate CNN	Multivariate LSTM	Multivariate CNN	Multivariate GRU
0	4.435.864	4.499.245	4.403.563	4.628.751	4.702.056	4.042.640
1	3.091.224	2.982.151	3.028.925	3.417.006	3.522.164	2.978.858
2	2.191.081	2.329.441	2.134.507	2.956.739	2.796.507	2.294.576
3	2.182.642	2.550.081	2.200.409	2.790.888	2.482.235	2.450.006
4	2.093.087	2.571.904	2.072.285	2.567.137	2.559.600	2.671.570
5	3.124.128	3.268.090	3.027.716	3.639.448	3.491.334	3.478.182
6	5.495.216	5.301.780	5.660.732	5.439.315	4.610.753	4.198.684
7	4.549.785	4.506.574	4.631.643	5.651.510	4.521.542	3.948.635
8	4.039.761	4.287.902	4.021.311	6.271.030	4.259.295	4.029.524
9	4.466.427	4.042.516	4.236.588	5.490.138	3.765.325	3.685.625
10	3.632.361	3.611.038	3.464.040	4.501.458	3.588.513	3.433.045
11	3.041.195	2.845.574	2.836.303	3.952.295	3.183.207	2.928.267
12	3.662.791	3.492.235	3.409.575	3.818.591	3.482.119	3.562.260
13	4.095.594	4.278.464	4.036.660	3.846.690	4.219.437	3.813.081
14	3.220.372	3.055.590	3.063.799	3.182.770	3.292.107	3.117.088
15	3.511.217	3.223.153	3.353.582	3.870.961	3.745.797	3.513.897
16	3.274.936	3.376.983	3.233.252	4.089.503	3.922.728	3.434.980
17	4.118.521	4.111.092	4.389.212	4.675.716	4.207.777	3.708.278
18	4.236.532	4.101.240	4.251.697	5.292.364	4.205.108	3.564.287
19	4.496.278	4.585.933	4.465.746	5.106.766	4.181.834	4.544.159
20	4.951.523	4.536.574	4.859.124	4.722.500	4.229.590	4.206.228
21	4.338.619	3.909.965	4.166.542	4.851.468	4.019.238	3.439.498
22	2.607.728	2.754.275	2.674.323	4.017.107	3.847.511	3.031.577
23	3.350.158	3.305.948	3.333.440	3.792.957	3.101.468	3.216.204

Table 8. RMSE by hour of day.

	Univariate LSTM	Univariate GRU	Univariate CNN	Multivariate LSTM	Multivariate CNN	Multivariate GRU
Monday	4.577.092	4.259.075	4.548.801	4.479.332	4.280.385	3.885.445
Tuesday	3.566.319	3.531.921	3.502.883	4.523.528	3.748.529	3.498.846
Wednesday	3.492.471	3.490.613	3.525.992	4.210.767	3.748.341	3.421.284
Thursday	3.593.578	3.561.085	3.563.370	4.387.151	3.628.478	3.468.277
Friday	3.052.738	3.039.549	3.039.554	4.165.078	3.349.650	2.913.714
Saturday	3.340.745	3.410.468	3.309.338	4.161.545	3.276.002	2.970.586
Sunday	4.554.971	4.577.181	4.400.628	4.688.867	4.398.552	4.251.404

Table 9. RMSE by Weekday.

	Univariate LSTM	Univariate GRU	Univariate CNN	Multivariate LSTM	Multivariate CNN	Multivariate GRU
0	4.089.535	3.970.581	4.057.803	3.884.902	4.201.148	3.584.523
1	3.056.962	3.187.061	3.071.386	3.126.195	3.294.615	3.129.284
2	3.381.196	3.455.033	3.373.685	3.498.980	3.611.051	2.996.041
3	2.427.785	2.338.842	2.387.659	3.259.613	2.172.241	2.711.525
4	4.206.710	4.234.472	4.245.901	3.991.276	4.269.528	3.684.861
5	3.142.522	3.187.673	3.139.748	3.432.344	3.159.002	2.717.732
6	4.908.121	4.932.685	4.797.615	4.967.057	4.440.907	4.077.453
7	4.662.792	4.677.397	4.784.595	5.249.663	4.657.473	4.073.197
8	3.939.353	3.753.432	4.005.767	3.981.159	3.648.983	3.129.275
9	2.813.271	2.797.363	2.921.738	3.410.611	3.134.364	2.798.278
10	5.047.353	4.823.737	5.088.274	4.610.310	4.519.670	3.944.016
11	3.234.754	3.201.945	3.181.496	5.387.886	3.597.652	3.279.225
12	2.251.040	2.376.903	2.280.337	4.969.621	3.210.381	2.732.824
13	3.909.431	4.017.636	3.966.985	5.630.899	4.244.419	3.854.662
14	2.774.412	2.759.266	2.717.174	5.258.318	3.964.027	3.076.984
15	4.426.995	4.391.040	4.300.578	6.256.248	4.578.705	4.341.019
16	5.911.473	5.960.286	5.756.007	7.387.280	5.836.505	5.958.704
17	2.321.724	2.347.250	2.460.256	5.749.869	3.528.631	2.832.903
18	3.146.417	3.048.539	3.093.731	5.673.725	3.959.766	3.367.138
19	2.610.365	2.629.965	2.540.967	5.250.418	3.252.245	2.843.031
20	4.470.352	4.316.964	4.121.920	7.256.182	4.940.335	5.333.761
21	3.204.443	3.257.238	3.212.052	6.079.838	3.856.416	3.594.756
22	4.302.264	4.506.581	4.326.112	5.735.706	4.433.043	4.166.780
23	2.166.699	2.204.665	2.160.345	3.503.069	2.575.125	2.701.358
24	2.354.366	2.266.176	2.395.541	3.400.124	2.736.546	2.508.569
25	2.136.816	2.233.432	2.205.946	2.957.502	2.371.742	2.138.178
26	4.128.428	3.920.241	3.921.026	4.248.089	3.816.364	3.635.770
27	3.631.594	3.331.938	3.292.303	3.460.634	3.121.407	2.707.140
28	2.542.205	2.587.989	2.556.964	3.059.449	2.576.399	2.271.717
29	2.994.321	2.944.111	2.943.648	3.509.829	3.033.563	2.714.333
30	2.858.121	2.764.768	2.960.186	3.471.229	2.847.922	2.771.191
31	2.360.839	2.217.310	2.302.209	3.220.180	2.811.844	2.375.509
32	2.256.861	2.186.207	2.317.532	2.810.478	2.664.784	2.120.883
33	3.100.456	3.024.763	3.112.396	3.316.694	3.116.838	2.819.892

34	3.942.284	4.012.055	3.908.680	4.010.549	3.695.788	3.582.919
35	3.084.604	2.908.734	2.973.043	2.893.467	2.832.536	2.884.213
36	4.894.375	5.012.576	4.741.344	4.259.373	4.221.733	4.436.808
37	4.019.445	3.750.456	3.817.268	4.249.106	4.197.477	4.431.924
38	5.389.866	5.314.443	5.407.667	5.241.325	5.035.839	4.666.195
39	5.325.450	5.248.308	5.379.124	4.391.884	4.534.469	4.430.828
40	3.326.593	3.217.635	3.347.414	3.132.979	3.149.670	2.723.226
41	3.521.056	3.171.891	3.464.166	3.179.114	3.007.727	2.776.214
42	4.452.844	4.350.698	4.464.456	3.689.470	3.725.270	3.434.066
43	3.895.361	4.182.303	3.941.390	3.663.734	3.675.125	3.406.575
44	2.964.571	2.882.669	2.961.726	3.815.606	3.342.282	2.786.766
45	3.883.397	3.772.871	3.776.946	3.824.803	3.425.905	3.960.050
46	4.905.055	4.843.884	4.781.565	3.736.295	4.688.594	4.057.777
47	3.560.312	3.244.696	3.481.527	3.349.406	3.313.027	3.296.491
48	4.396.363	4.135.785	4.215.830	3.581.572	4.183.283	3.446.517
49	4.686.378	4.579.506	4.622.376	4.196.663	4.913.098	4.135.465
50	3.458.226	3.465.018	3.433.476	3.077.790	3.740.904	3.258.225
51	5.673.426	5.480.707	5.511.641	5.063.729	5.154.069	5.157.049
52	3.143.433	3.232.865	2.988.360	3.682.933	3.991.917	3.115.658

Table 10. RMSE by Week.

	Univariate LSTM	Univariate GRU	Univariate CNN	Multivariate LSTM	Multivariate CNN	Multivariate GRU
January	3.303.283	3.312.096	3.285.045	3.500.524	3.503.421	3.154.395
February	4.270.591	4.261.938	4.286.215	4.435.311	4.035.619	3.578.364
March	3.556.468	3.477.380	3.606.089	4.764.809	3.671.490	3.291.741
April	4.246.137	4.269.304	4.169.532	6.126.177	4.608.855	4.304.638
May	3.332.335	3.300.404	3.223.057	6.088.761	4.027.748	3.797.368
June	2.973.159	3.037.590	2.999.041	4.060.578	3.173.397	3.027.566
July	3.294.982	3.151.727	3.171.379	3.532.575	3.116.727	2.847.070
August	2.992.854	2.933.634	2.979.628	3.396.848	3.069.410	2.780.193
September	4.540.161	4.407.112	4.422.633	4.270.444	4.170.256	4.159.442
October	4.017.728	4.007.418	4.065.044	3.516.255	3.560.586	3.298.738
November	3.943.890	3.826.207	3.859.389	3.678.516	3.771.268	3.576.167
December	4.458.535	4.351.783	4.348.083	4.049.093	4.493.311	3.996.776

Table 11. RMSE by Month.