Decision support in a volatile electricity market: forecasting and cost optimization

Oscar Andreas Olsson

LUND UNIVERSITY

Department of Automatic Control

MSc Thesis
TFRT-6208
ISSN 0280-5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Tryckeriet i E-huset
Lund 2023

Abstract

Given the increase in electricity prices in recent years due to two reasons; the rebound effect after the initial corona outbreak and the Russian invasion of Ukraine, the burden of paying the rising monthly expense for electricity has become an unwelcome reality for a significant part of society. The electricity trades on the open market called Nord Pool for the Nordic countries, among others, where buyers and sellers come together to find a market price for electricity each day. By forecasting future electricity prices using the machine-learning model XGBoost and a select number of features, the use of electricity can be optimized in the near future with respect to cost. Two different XGBoost models were constructed and evaluated on their ability to predict future prices. Each model was trained on a unique dataset, where the datasets are of different characteristics in terms of volatility. The first model, trained on historical electricity prices with less volatility showed a much more reliable forecasting ability than the second model, trained on historical electricity prices with much more inherent volatility. The optimizations were executed in Matlab with two different optimization solvers. The cost-optimization with the forecasted electricity prices is compared to other charging patterns, in order to determine if the forecasted prices are accurate enough to save cost. Each optimization problem had a number of defined objectives and a number of constraints assigned to it. The result of this thesis showed that the charging protocol incorporating the forecasted electricity prices while minimizing the cost produced a more cost-efficient solution in comparison to the other

charging protocols brought up.

Acknowledgements

I would like to thank my supervisor Emma Tegling who has guided and supported me through all the obstacles faced in writing this thesis.

Contents

1

Introduction

Electricity is a vital part of everyday life, from households to industries. Lifestyles have more and more became dependent on electricity. If electricity magically disappeared, the world would change overnight. Given the significance of electricity, it is a necessity for every individual requires access. Without it, the majority of daily tasks would become a burden to perform. Therefore, it is of utmost importance that every party has the ability to afford access to electricity. Unfortunately, a number of compounding factors have arisen recently, sending the electricity price skyrocketing.

1.1 An Electricity Market in Turmoil

Since the outbreak of the COVID-19 pandemic, energy systems worldwide have experienced volatile movements in pricing. Initially, energy demand decreased rapidly all over the world in mid-March-April of 2020 [Buechler et al., 2022]. Cutbacks in electricity demand coincided with lockdowns being established and enforced in European countries. A majority of the reduction in electrical demand originated from the reduction in industrial and commercial operations [Werth et al., 2021]. The unexpected decrease in demand yielded low electricity prices. By July 2020, the cumulative number of hours where the day-ahead price in Germany traded in negative territory, i.e. below zero, surpassed any other year on record. The previous record was 211 hours set in 2019 [Halbrügge et al., 2021]. Trading in negative territory indicates an imbalance between buyers and sellers in the energy market as sellers desperately compete to dispatch their energy holdings. By 2021, the demand for energy swiftly rebounded to surpass previous levels of 2019 and 2020 [Lorenczik and Copier, 2022]. The unexpected reversal in demand, partly due to unusual weather conditions and the economic recovery of 2021 as lockdowns were phased out, propped up the energy price by +470% in the Nordic countries, by +239% in France, and by +218% in Germany in comparison to 2020 day-ahead prices. The weather conditions depleted natural gas storage in Europe to lower-than-expected levels since the first months of 2021 were unusually cold. At the same time, the gas flow through Nord Stream had yet not reached pre-pandemic levels. The combination led to an explosive demand for natural gas as the gas reserves now required more gas than usual [Jääskeläinen et al., 2022]. While there were lower levels of natural gas in Europe, the invasion of Ukraine by Russia induced chaos and fear in energy and commodity markets as Russia is a large supplier of commodities, mainly crude oil and natural gas. Crude oil from Russia accounted for 45% of crude oil imports to the European Union (EU) and the United Kingdom (UK) while natural gas accounted for 35% of natural gas imports to the EU and the UK in 2020. The energy imports from Russia to the EU constituted 25% of all energy imports for the year 2021. The uncertainty of natural gas and crude oil supply triggered a five-fold increase in the price of natural gas and a doubling of the price of crude oil at the price top of 2022 [Celasun et al., 2022].

Before the pandemic, the percentage of European citizens unable to pay their energy bills or faced delays in paying them, measured at 52 million (11%), according to estimates. Furthermore, an estimated 57 million (12.7%) suffered from cold homes in the winter, and 104 million (23%) could not keep their house comfortable during the summer. [Belaïd, 2022]. The estimates are calculated before the pandemic. The numbers are undeniably elevated today, given the energy crisis and the aftermath of the pandemic. Energy poverty, also denoted as energy vulnerability, is an umbrella term used to declare a household's inability to acquire energy for essential services such as cooking, heating, cooling, and household lighting [Belaïd, 2022]. The definition of energy poverty

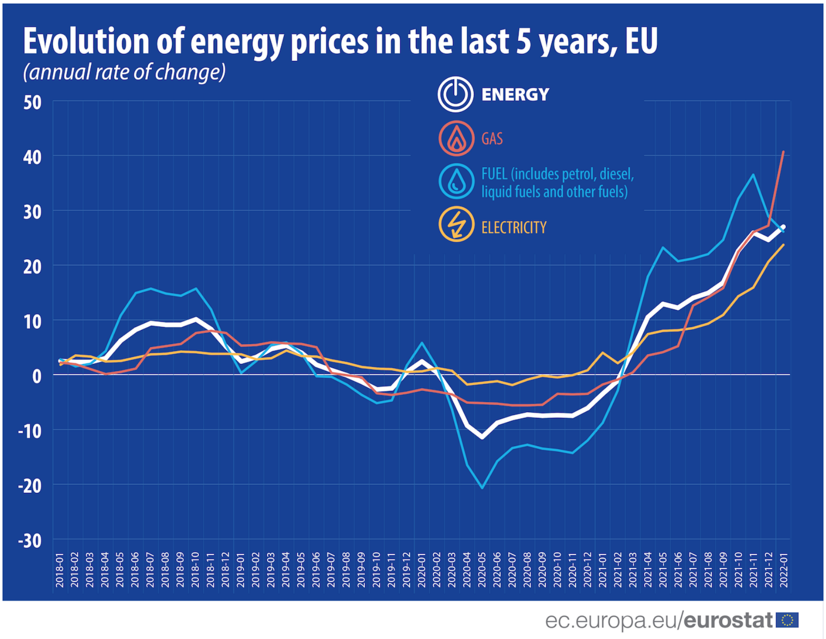


Figure 1.1 The annual rate of change of energy, gas, fuel, and electricity prices. In March of 2021, all of the aforementioned categories have risen extremely fast. The picture is taken from [*Energy inflation rate continues upward hike, hits 27%* n.d.]

varies between countries [Belaïd, 2022], however, given its broad interpretation, the brief conclusion is that 52 million live at risk of energy poverty in the EU. In response to the growing concern of strained economics of households, the European Commission has adopted multiple measures to address gas and electricity prices [EUROPEAN COMMISSION, 2021], introduced a proposal to phase out the dependency of Russian fossil fuels [EUROPEAN COMMISSION, 2022] and to assist countries in the European Union to deal with rising prices [COMMISSION, 2022]. In addition, The European Commission has in collaboration with The Internal Energy Agency (IEA) developed a manual on how to reduce energy use and save cost [*Playing my part* n.d.] for European citizens. Other governmental bodies have dealt with the crisis differently. The UK has since October implemented a package named 'Energy Bills Support Scheme'. The package provides a one-time payment of £400 allocated to pay off energy bills [*Help with your energy bills* n.d.] Spain has announced a temporary windfall tax for profits in the fiscal years 2022 and 2023 of electric utility companies. The revenue from windfall taxes is allocated to attenuate the strain on vulnerable households and firms originating from high energy prices [Baunsgaard and Vernon, 2022]. There is a long list of countries implementing similar solutions to restrain the effects of rampant energy prices. Recently, the EU adopted an additional proposal to offset the high price of natural gas. Energy ministers representing countries from the EU came to the agreement of enacting a price cap of 180 €/MWh on gas prices [Staff, 2022a]. If the natural gas price were to rise above 180 €/MWh, the price difference is compensated by the EU.

1.2 Problem

High energy prices are an issue relevant to all, independent of societal class, given how dependent all parts of society are on electricity being operational. In this thesis, the problem tackled is whether it is possible to reduce energy costs for an individual consumer by forecasting the electricity price and utilizing electricity during the expected cheapest periods. The problem is answered by producing a solution that attempts to minimize the cost of electricity consumption. The solution consists of two parts, the first part is forecasting electricity prices 5 days in the future. The second part is calculating the most cost-efficient pattern of electricity consumption, with the assumption of knowledge regarding future electricity prices. The forecast is done by analyzing contributing factors to the electricity price and training machine-learning models on collected data stretching back to 2015-01-06. In the collection, the models output a prediction for the electricity price every hour for 1 day ahead up to 5 days ahead. Calculating the most cost-efficient pattern of electricity usage is constructed with an optimization problem. The objective of the optimization problem is to find the lowest minimal value while ensuring the required electricity is provided to the object in question.

Summarized, the topics addressed are

1. Forecasting price of electricity with a 5-day window
2. Model optimization for electricity usage and present cost-reductions

1.3 Contribution

The contribution of this thesis will firstly be a tool that is applicable to individual consumers with the goal to minimize electricity costs, and secondly, the corresponding research is done to develop the tool. Forecasting as a category is a large body of research and is used in multiple sectors; weather, population, and economy. Hence, any new revelation and insight possible to improve performance are of great value. One other contribution is the experimentation if it is feasible with limited resources, data, and time to produce a reliant forecasting model that performs accurately. When forecasting the weather, the number of parameters influencing future weather is immense and the forecast is done by multiple models and multiple meteorologists. The computational requirements to forecast weather are vast. One could argue that the same is true for forecasting energy prices. Hence, it would be valuable to detail if it is feasible to produce a reliable forecast model for energy prices with one single person working with one model. Additionally, a contribution is the analytic results of what factors have the greatest influence on the electricity price.

1.4 Related work

Forecasting electricity prices is a well-researched area, the literature is rich with multiple papers analyzing the potential to forecast prices with different time horizons. The number of distinct techniques used in predictions is diverse, some names are [Javier, 2017]:

- Statistical time series models

- Artificial Neural Networks
- Wavelet transform models
- Regime-switching Markov models
- Fundamental market models
- Equilibrium models
- Ensemble and portfolio decision models

Two papers that are closely related to the subject of this thesis are outlined below.

1.4.1 Forecasting through a statistical time series model

In the paper [Contreras et al., 2003], Javier Contreras, Rosario Esínola, Francisco J. Nogales, and Antonio J. Conejo attempt to forecast the day-ahead price of electricity in markets using a time-series model by the name of ARIMA. The study is performed on power markets of mainland Spain and California, USA, where daily trading of electricity occurs. The writers of the paper studied the day-ahead electricity prices over three non-consecutive weeks selected in the calendar year 2000 when analyzing the Spanish market. The number of weeks selected for the Californian market analysis is only one. The findings from the paper are that the error between the forecasted price and the actual price is on average 10% for the Spanish market. The same number for the Californian market is around 5%. More about the ARIMA model is presented in Section 2.1.2.

1.4.2 Forecasting through an artificial neural network

In the research paper [Singhal and Swarup, 2011], Deepak Singhal and K.S. Swarup applied an artificial neural network to predict future electricity prices up to 48 hours in advance. The pair collected eights month of electricity price data and trained the neural network on six months' worth of data. It is common knowledge that there are some factors that drive the electricity price more than others. The authors bring up the following attributes as significant factors that impact the price.

1. Historical market closing prices (MCPs)
2. System loads
3. Fuel prices

The attributes of data collected for the study were time information, i.e. day of the week and the time slot of the day, load demand for electricity in the grid, and finally historical price information. By utilizing feature engineering, the number of features as input to the network was a total of 13. Some features established by applying feature engineering are the change in demand and the electricity price x number of hours in the past. For the final model, the RMSE metric for the forecast was 0.525 over 48 hours in the case of a 'normal trend' of price history. In the case of a "small spike", the RMSE was measured at 1.129. When studying the forecasting result where the electricity had a large spike in price, rising from $\sim$ 50 \$/MWh to $\sim$ 300 \$/MWh in a short amount of time, the RMSE was measured at 4.105

1.4.3 Forecasting through XGBoost

In the paper [Wu et al., 2022], Kehe Wu, Yanyu Chai, Xiaoliang Zhang, and Xun Zhao evaluate the ability to forecast electricity prices with multiple models. To name a few, ARIMA, XGBoost, and PSO-XGBoost are used. PSO, an acronym for particle swarm optimization, provides assistance in adjusting the hyperparameters for the XGBoost model. The dataset used in the paper is electricity price data from the Australian power market. The data stretches from January 2019 to December 2019. The models are evaluated based on each month in the dataset. For each month January to June, each model is trained on the first 24 days of the month and the remaining 7 days are used for testing the models. For each month July to December, each model is trained on the first 27 days, and the remaining 3 days are used for testing the models. The models are quite simplistic in the feature aspect, only having 4 features. Two of the features are classified as "time" features, being the day of the week and the hour of the day. The other two features are classified as "price data" features, being the "electricity price at adjacent time points" and the "electricity price at the same time point on adjacent days". The error metric RMSE is one of the metrics used to determine the forecasting ability of the models. Since the XGBoost models are relevant to this paper, they are mentioned below with their corresponding RMSE metrics for each month.

RMSE		
Month	**PSO-XGBoost**	**XGBoost**
January	0.6081	4.0098
February	2.5083	4.9329
March	1.3374	3.0893
April	2.9208	3.9655
May	39.2473	38.4477
June	15.3014	16.3771
July	6.9651	10.2248
August	2.3113	3.5036
September	1.2072	2.5119
October	24.2422	30.5328
November	1.5541	2.4317
December	29.2056	31.4767

Table 1.1 Table assembled with RMSE metrics from [Wu et al., 2022]

The authors establish that XGBoost with particle swarm optimization is able to produce predictions relatively close to the actual value. They mention that the particle swarm optimization of XGBoost has some details that need to be improved, which may improve the prediction accuracy of the model. [Wu et al., 2022].

1.4.4 Price forecasting & power cost optimization

In the research paper [Albahli et al., 2020], Saleh Albahli, Muhammad Shiraz, and Nasir Ayub attempt to forecast electricity prices and perform cost-optimization with the forecasted prices in the context of cloud computing. The companies that provide cloud computing are dependent on the availability of cheap electricity to power the hardware. The authors propose the use of XGBoost in order to "offload or move storage, predict electricity price, and as a result reduce energy consumption costs in data centers". The real-world data is sourced from the Independent Electricity System Operator in Ontario, Canada, with the data sampled between 2003 and 2018. The XGBoost model is fed features with laggards of the electricity price, the electricity price itself, the hour of the day, the day of the month, the month of the year, the year, and an additional feature denoted 'DateAsNum'. When examining the proposed models' ability to predict on the test set, the authors achieved an RMSE of 9.25. With the optimization model defined in the paper, Albahli, Shiraz, and Ayub are able to reduce the electricity cost by 25.32% by using the forecasted electricity prices.

2

Background

2.1 Machine-learning Preliminaries

There are subjects and concepts that have to be understood to comprehend how the modeling works and to grasp some of the reasoning done in the thesis. In this chapter, a number of subjects and terms are brought up and defined to allow the reader to understand later chapters in the thesis.

2.1.1 Error metric

The machine learning model used in this thesis has the objective to predict a value $\hat{y}$ and is trained by minimizing the error between the predicted value $\hat{y}$ and the true value y, also said to be the observed value y. The error metric to express the degree of error is Root Square Mean Error (RMSE):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{2.1}$$

where n is the number of observations, $\hat{y}_i$ is the predicted value and y_i is the observed value. RMSE is a good measurement of accuracy to compare forecasting errors of various models or model configurations for a specific variable. It is however not a good measurement between variables, as the RMSE is scale-dependent. [Christie and Neill, 2022].

The following metric is not utilized when training the model, but to concretely measure the overall performance of the models. A metric that builds upon RMSE is Mean Column-Wise Root Mean Square Error (MCRMSE). It is used when having multiple RMSE values which occur when using multivariate models since the RMSE is calculated independently of each variable, and MCRMSE averages accumulated RMSE over the number of variables. The MCRMSE is defined as [Chze and Abdullah, 2022]

$$MCRMSE = \frac{1}{m}\sum_{j=1}^{m}\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_{ij} - \hat{y}_{ij})^2} \tag{2.2}$$

which is equivalent to

$$MCRMSE = \frac{1}{m}\sum_{j=1}^{m} RMSE_j \tag{2.3}$$

where m is the number of independent RMSE values.

2.1.2 ARIMA

Forecasting of a particular variable y is often done with time series. Time series, as the name hints, consists of a series of time intervals and the corresponding measurement of the response variable y. Instances of common time series forecasting models are future stock market prices, future supply and demand curves, and future economic conditions [Taslim and Murwantara, 2022]. Time series forecasting attempts to predict future trends by analyzing previous data and patterns.

ARIMA, an acronym for Autoregressive Integrated Moving Average, is a model frequently used for forecasting time series. The ARIMA model combines Autoregressive (AR) and Moving Average (MA) with an "Integrated" part. The "I", short for "Integrated", represents an instrument to convert a non-stationary time series to be stationary [Siami-Namini et al., 2018]. The ARMA model is unable to be applied to non-stationary data.

The ARIMA model has three parameters, p, d, q, and is often written as $ARIMA(p, d, q)$. ARIMA is an extension of $ARMA(p, q)$ (Autoregressive Moving Average), which is only designed to process stationary time series, aside from ARIMA, which has the capability of processing non-stationary time series. The parameters p, q are terms for the Autoregressive(p) part and the Moving Average(q) part respectively [Zang et al., 2019].

2.1.3 Gradient descent

Gradient descent (GD) is an optimization algorithm that is often used in machine learning models to find the minimum of a loss function. The iterative algorithm will approach and localize the local minimum of a function. In order for gradient descent to be applicable, the function must be differentiable and have the property of being convex. The gradient descent is applicable for functions that are univariate or multivariate.

The "standard" gradient descent is named "Batch gradient descent", to separate it from other types of gradient descent, such as stochastic gradient descent. Batch gradient descent works by iteratively computing the gradient for a function with respect to parameters θ.

$$\theta = \theta - \eta \nabla_{\theta} J(\theta) \tag{2.4}$$

Here, $J(\theta)$ is an objective function that is supposed to be minimized. The variable η is the learning rate for the algorithm. ∇_{θ} is the gradient of the function J. [Ruder, 2016]. If a too-small value of η is chosen, it will take a longer time to arrive at the minimum. However, if a too large value is chosen for η, there is a chance of gradient descent never reaching the minimum [*Optimization in ML* n.d.] $\eta\nabla_{\theta}$ is called the step size and decreases the closer to the minimum the algorithm is due to ∇_{θ} decreasing. In the beginning, gradient descent will perform relatively large steps compared to the steps in the end. If gradient descent reaches the minimum, ∇_{θ} is equal to zero and the step size equals zero based on $\theta = \theta - [\eta\nabla_{\theta} = 0]J(\theta) = \theta$. Consequently, the algorithm stays in the same position.

A drawback of gradient descent is the inability to locate the global minima for non-convex cost functions. Functions with non-convex properties may have a local minimum, where the gradient for the cost function is zero, ultimately terminating the algorithm. The same scenario is possible with saddle points as the cost function with only locate a local maximum on one side and a local minimum on the other side of the cost function. [*What is gradient descent?* N.d.]

2.1.3.1 Gradient Boosting. Gradient Boosting applies gradient descent to obtain improved model predictions [*How gradient boosting differs from gradient descent* n.d.] In order to understand the machine-learning model in this thesis, it is necessary to understand the foundation upon which the model is built on. Gradient Boosting is the heart of the machine-learning model's foundation.

Gradient boosting builds additive regression models by fitting an initial function $F_0(x)$ (base-learner) to the current "pseudo"-residuals by the least-square metric. The "pseudo"-residuals are equal to the gradients of the loss function. The residuals are called "pseudo"-residuals because they are not conventional residuals, i.e. the difference between the predicted value and actual value, but instead the gradient of the loss function being minimized.

With Gradient Boost, the goal is to estimate a function based on the output variable y and a set of input variables $\mathbf{x} = \{x_1, x_2, ..., x_n\}$. Provided with the data $\{y_i, \mathbf{x}_i\}_1^N$ of known values $(y, \mathbf{x})$, the objective is to find a function $F^*(\mathbf{x})$ that maps $\mathbf{x}$ to every $y, F^* : \mathbf{x} \mapsto y$, such that the expected value $E_{y,x}(\Psi(y, F(\mathbf{x})))$ is minimized, where $\Psi(y, F(\mathbf{x}))$ is the loss function. All of the equations are gathered from the paper on Gradient Boosting [Friedman, 2002].

$$F^*(\mathbf{x}) = \arg\min_{F(\mathbf{x})} E_{y,x} \Psi(y, F(\mathbf{x})) \tag{2.5}$$

An appropriate loss function has to be selected by the category of the problem. The boosting generates an approximation $F^*(x)$ by adding M number of functions together where β are expansion coefficients, the parameters $\mathbf{a}$ are fit to training data, and the functions $h(x; a)$ are base learners composed of trivial functions of x.

$$F(x) = \sum_{m=0}^{M} \beta_m h(x; \mathbf{a}_m) \tag{2.6}$$

The first step is to start with a guess $F_0(\mathbf{x})$, then for $m = 1, 2, ...M$, β_m and $\mathbf{a}_m$ is given by

$$(\beta_m, \mathbf{a}_m) = \arg\min_{\beta,\mathbf{a}} \sum_{i=1}^{N} \Psi(y_i, F_{m-1}(\mathbf{x}_i) + \beta h(\mathbf{x}_i, \mathbf{a})) \tag{2.7}$$

with

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \beta_m h(\mathbf{x}; \mathbf{a}_m) \tag{2.8}$$

The function $h(x, a_m)$ is fit by using the squared error as a metric

$$\mathbf{a}_m = \arg\min_{\rho,\mathbf{a}} \sum_{i=1}^{N} [\tilde{y}_{im} - \rho h(\mathbf{x_i}; \mathbf{a})]^2 \tag{2.9}$$

where $\tilde{y}_{im}$ is the residual (”pseudo”-residual) given by the partial derivative of the loss function Ψ with respect to $F(\mathbf{x}_i)$

$$\tilde{y}_{im} = \left[\frac{\partial \Psi(y_i, F(\mathbf{x}_i))}{\partial F(\mathbf{x}_i)}\right]_{F(\mathbf{x_i})=F_{m-1}(\mathbf{x}_i)} \tag{2.10}$$

Then, β_mw is given by

$$\beta_m = \arg\min_{\beta} \sum_{i=1}^{N} \Psi(y_i, F_{m-1}(\mathbf{x}_i) + \beta h(\mathbf{x_i}; \mathbf{a}_m)) \tag{2.11}$$

When the Gradient Boost is used for constructing trees, denoted Gradient Tree Boosting, the function $h(\mathbf{x}; \mathbf{a})$ for the base learners is replaced with regression trees. [Friedman, 2002].

2.1.4 XGBoost

XGBoost is the machine-learning model that will be applied in this thesis. Therefore, it is explained to a great extent to ensure that the reader has a deeper insight into the model.

XGBoost is a machine-learning model built on gradient tree boosting, also known as Gradient Boosting Machine (GBM) or Gradient Boosted Regression Tree (GBRT). A regularized learning objective is a method to assist the optimization and reduce the likelihood of overfitting [*Deep Learning Basics Lecture 3: Regularization I* n.d.] All equations and variables in this section are taken from [Chen and Guestrin, 2016]. The regularized learning objective in XGBoost is derived from a tree ensemble model regularized learning objective. A regularized learning objective for a tree ensemble model is the sum of K additive functions. Assume there exists n data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ where $\mathbf{x_i}$ consists of m features, then the prediction $\hat{y}_i$ is defined as

$$\hat{y}_i = \sum_{k=1}^{K} f_k(\mathbf{x}_i), f_k \in \mathcal{F} \tag{2.12}$$

f_k is a unique regression tree in the space $\mathcal{F}$ where $\mathcal{F}$ is the set of all regression trees (known as CARTs). A regression tree has the structure q and consists of leaf weights w. Each leaf is assigned a score and the score of the i-th leaf is given by w_i. The relation between each regression tree and the weight w_i in the tree structure q is $f(\mathbf{x}_i) = w_q(\mathbf{x}_i)$. The prediction $\hat{y}_i$ is, therefore, the sum of K distinct regression trees. Each tree is called a base learner, also referred to as a weak learner. XGBoost combines multiple weak learners to form a strong learner. The idea of combining multiple weak learners into a strong learner is the idea is referred to as "boosting". Each individual regression tree is considered a weak learner since it only marginally improves the ability to predict the outcome variable. A strong learner is an algorithm that achieves peak accuracy of the outcome variable for supervised learning. [*Chapter 5 XGBoost* n.d.]

2.1.4.1 Regularized object in XGBoost. The regularized objective supposed to be minimized in the XGBoost is defined as $\mathcal{L}(\phi)$

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \tag{2.13}$$

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda||w||^2 = \gamma T + \lambda\frac{1}{2}\sum_{i=1}^{T} w_i^2 \tag{2.14}$$

where the variable T is the number of leaves in a regression tree f. γ is a regularization parameter that has the default value $\gamma = 0$ [*XGBoost Documentation* n.d.] λ is a parameter to control how much l_2 regularization there is [*XGBoost Documentation* n.d.]

l is a differentiable and convex loss function, two properties required for performing gradient descent. The default loss function [*XGBoost Documentation* n.d.] is given by

$$l(\hat{y}_i, y_i) = (\hat{y}_i - y_i)^2 \tag{2.15}$$

The function $\Omega(f)$ is a gauge to measure the complexity of a given tree f. By adding $\Omega(f)$ to the regularization function $\mathcal{L}$, the model is penalized for trees with high complexity. The l^2-norm is the squared sum of all the weights in the tree f. The added $\Omega(f)$ is introduced to prevent overfitting and smooth out the weights of the tree.

Let $\mathcal{L}^{(t)}$ be the regularized objective at the t-th iteration of the model. The variable $\hat{y}_i$ is the prediction of the i-th instance and $\hat{y}_i^{(t)}$ is the prediction of the i-th instance at the t-th iteration. The $\mathcal{L}^{(t)}$ objective can be expressed as

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)) + \Omega(f_t) \tag{2.16}$$

The added term f_t is chosen in a greedy manner such that the $\mathcal{L}^{(t)}$ is minimized. Equation 2.16 can be rewritten using second-order approximation of Taylor expansion [Sheng and Yu, 2022] under the assumption that $(y_i, \hat{y}_i{}^{(t-1)} + f_t(\mathbf{x}_i))$ is close to $(y_i, \hat{y}_i{}^{(t-1)})$.

$$
\begin{aligned}
l(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)) &\simeq l(y_i, \hat{y}_i^{(t-1)}) + \frac{\partial l}{\partial y_i} l(y_i, \hat{y}_i^{(t-1)})(y_i - y_i) \\
&+ \frac{\partial l}{\partial \hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})(\hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i) - \hat{y}_i^{(t-1)})) \\
&+ \frac{1}{2!} \frac{\partial^2 l}{\partial y_i^2} l(y_i, \hat{y}_i^{(t-1)})(y_i - y_i)^2 \\
&+ \frac{1}{2!} \frac{\partial^2 l}{\partial {\hat{y}_i}^{(t-1)^2}} l(y_i, \hat{y}_i^{(t-1)})(\hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i) - \hat{y}_i^{(t-1)}))^2 \\
&= l(y_i, \hat{y}_i^{(t-1)}) \\
&+ \frac{\partial l}{\partial \hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})(f_t(\mathbf{x}_i)) \\
&+ \frac{1}{2} \frac{\partial^2 l}{\partial {\hat{y}_i}^{(t-1)^2}} l(y_i, \hat{y}_i^{(t-1)})(f_t(\mathbf{x}_i))^2
\end{aligned}
\tag{2.17}
$$

As described in [Chen and Guestrin, 2016], introduce two variables g_i and h_i to simplify the expression. g_i is the first partial derivative of the loss function and h_i is the second partial derivative of the loss function.

$$g_i = \frac{\partial l}{\partial \hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)}) \tag{2.18}$$

$$h_i = \frac{\partial^2 l}{\partial {\hat{y}_i}^{(t-1)^2}} l(y_i, \hat{y}_i^{(t-1)}) \tag{2.19}$$

With g_i and h_i, the objective function becomes

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} [l(y_i, \hat{y}^{(t-1)}) + g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i)] + \Omega(f_t) \tag{2.20}$$

The expression can be simplified by removing the constant $l(y_i, \hat{y}^{(t-1)})$ and writing $\Omega(f_t)$ as the entire expression

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} [g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i)] + \gamma T + \lambda \frac{1}{2} \sum_{i=1}^{T} w_i^2 \tag{2.21}$$

Define the set I_j as the set of indices where the tree structure q with input $\mathbf{x}_i$ is the leaf with the leaf number j , i.e.

$$I_j = \{i | q(\mathbf{x}_i) = j\} \tag{2.22}$$

A concrete tree with I_j written out is shown in Figure 2.1. With I_j, $\mathcal{L}^{(t)}$ is rewritten to

$$\mathcal{L}^{(t)} = \sum_{j=1}^{T} \left[\left(\sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) w_j^2 \right] + \gamma T \tag{2.23}$$

The best weight w_j^* for a leaf is now computable by

$$w_j^* = -\frac{\sum\limits_{i \in I_j} g_i}{\sum\limits_{i \in I_j} h_i + \lambda} \tag{2.24}$$

and the best value for the regularization objective at the t-th iteration for a tree structure q is computed by

$$\mathcal{L}(q) = -\frac{1}{2} \sum_{j=1}^{T} \frac{\left(\sum\limits_{i \in I_j} g_i \right)^2}{\sum\limits_{i \in I_j} h_i + \lambda} + \gamma T \tag{2.25}$$

Example 1 Constructing a regression tree (CART)

A CART may be a foreign subject never approached before. To understand the fundamental pillars of XGBoost, the knowledge of how CARTs are constructed heavily influences the capability of understanding the algorithm. To illustrate how a CART can be constructed, follow the example displayed in this section.

From a dataset randomly constructed on the fly, assume that we have the following properties as portrayed in Table 2.1 and the goal is to predict the electricity price based on the properties.

Index	Windspeed (m/s)	Demand (MW)	Natural gas (EUR)	Electricity price (EUR/MWh)
1	2.5	2500	120	65
2	7	2700	250	120
3	6.1	7200	50	55
4	12.1	1960	79	24
5	5.2	9652	92	155
6	1.1	1010	105	74

Table 2.1 Table with the features.

The average electricity price from the data entries are:

$$\mu' = \frac{1}{6}\sum_{i=1}^{6} E_i = 81.33 \tag{2.26}$$

Based on Table 2.1, the best guess we could come up with is μ', as it would be the guess that generates the minimum error with respect to all values in the 'Electricity price' column. Is this the best guess we could come up with? No!

The residual r for each data entry is the electricity price of each data point minus the average electricity price. The residual column is added in Table 2.2

It is now possible to construct a regression tree. For this example, the number of leaves allowed is restricted to 4 to reduce the complexity of the tree. If the number of residuals in a leaf is greater than 1, the average is calculated for that leaf.

Gradients	Windspeed (m/s)	Demand (MW)	Natural gas (EUR)	Electricity price (EUR/MWh)	Residual
g_1, h_1	2.5	2500	120	65	-16.33
g_2, h_2	7	2700	250	120	38.66
g_3, h_3	6.1	7200	50	55	-26.33
g_4, h_4	12.1	1960	79	24	-57.33
g_5, h_5	5.2	9652	92	155	73.66
g_6, h_6	1.1	1010	105	74	-7.33

Table 2.2 Table with the residuals added.

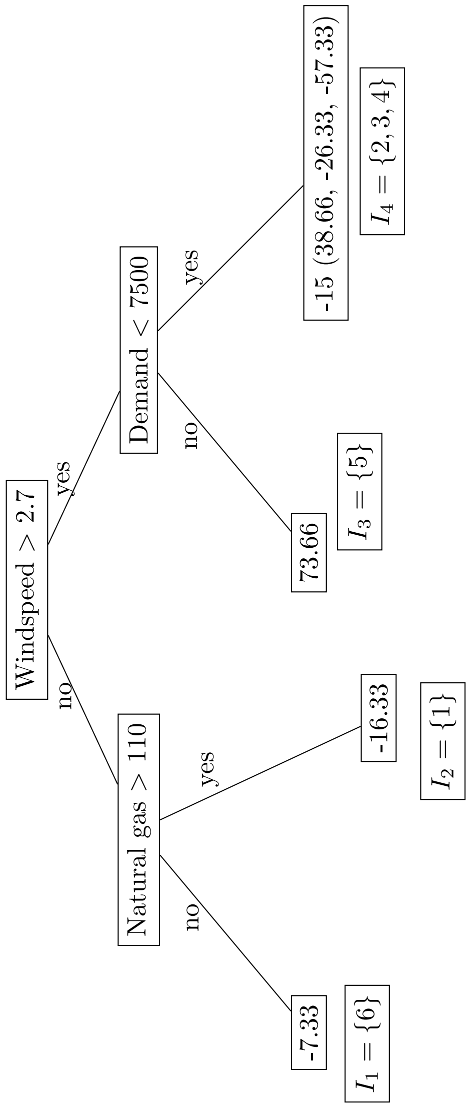


Figure 2.1 A constructed regression tree from the dataset. The cutoff values have been chosen arbitrarily. On how to construct the tree, inspiration from [Statquest, 2019] has been used.

By using the simple tree in Figure 2.1, it is possible to predict the electricity price when the windspeed is 3.2 m/s, and the demand is 1200 MW. Let r_i be the i-th residual. The predicted electricity price is

$$\hat{y}_i^1 = \mu' + r_i = \mu + (-15) = 66.33 \tag{2.27}$$

It is also possible to predict the electricity price when the windspeed is 1.5 m/s and the price of natural gas is 20 EUR. The predicted electricity price is

$$\hat{y}_i^2 = \mu' + (-7.33) = 74.00 \tag{2.28}$$

A learning rate α is often used in the prediction, such that $\hat{y}_i = \mu + \alpha r_i$. Assume $\alpha = 0.1$, then the corresponding predictions above become

$$\hat{y}_i^{1'} = \mu' + 0.1 \cdot (-15) = 79.83 \tag{2.29}$$

and

$$\hat{y}_i^{2'} = \mu' + 0.1 \cdot (-7.33) = 80.60 \tag{2.30}$$

With the tree, the residuals are updated. When calculating the new residuals, the average income is no longer used, instead, the predicted electricity prices are used when calculating the residuals.

The new table with updated residuals is shown in 2.3

Gradients	Windspeed (m/s)	Demand (MW)	Natural gas (EUR)	Electricity price (EUR/MWh)	Residual$_1$	Residual$_2$
g_1, h_1	2.5	2500	120	65	-16.33	-14.69
g_2, h_2	7	2700	250	120	38.66	40.16
g_3, h_3	6.1	7200	50	55	-26.33	-24.83
g_4, h_4	12.1	1960	79	24	-57.33	-55.83
g_5, h_5	5.2	9652	92	155	73.66	66.30
g_6, h_6	1.1	1010	105	74	-7.33	-6.60

Table 2.3 Table with updated residuals.

Generally, the residual decreased overall except for row 2 where the residual increased from 38.66 to 40.16. The RMSE for the residuals in column residual_1 is

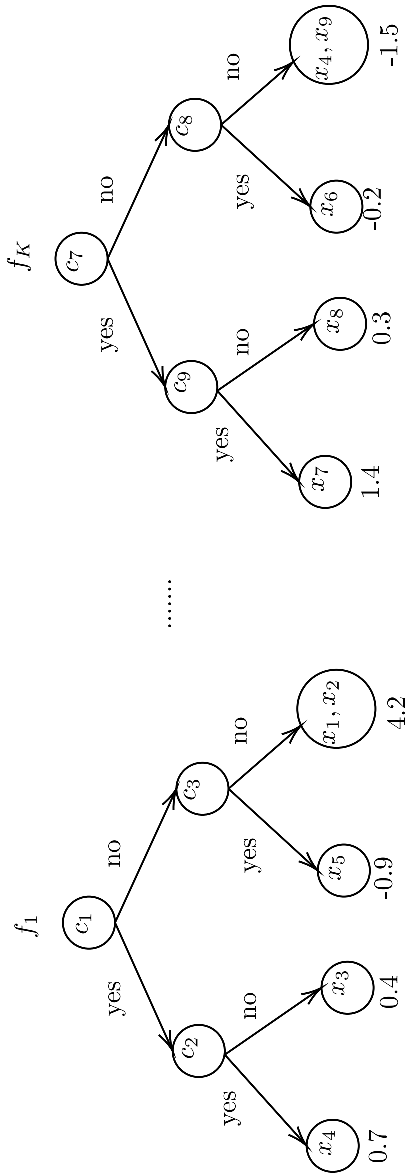
$$\sqrt{\frac{r_{1,1}^2 + r_{1,2}^2 + r_{1,3}^2 + r_{1,4}^2 + r_{1,5}^2 + r_{1,6}^2}{6}} = 43.25 \qquad (2.31)$$

The RMSE for the residuals in column residual_2 is

$$\sqrt{\frac{r_{2,1}^2 + r_{2,2}^2 + r_{2,3}^2 + r_{2,4}^2 + r_{2,5}^2 + r_{2,6}^2}{6}} = 40.83 \qquad (2.32)$$

An improvement in only one iteration. The RMSE decreased marginally, which is why each respective tree is referred to as a weak learner. The constructed CART is only one of many possible trees that can be built. This could be a possible tree f_k constructed by XGBoost where the number of leaves is equal to 4, and the depth of the tree is equal to 2.

Assume that we have the following K different regression trees where each c_i is a conditional statement based on features of the model.


f_1
c_1
yes
no
c_2
c_3
yes
no
yes
no
x_4
0.7
x_3
0.4
x_5
-0.9
x_1, x_2
4.2
......
f_K
c_7
yes
no
c_9
c_8
yes
no
yes
no
x_7
1.4
x_8
0.3
x_6
-0.2
x_4, x_9
-1.5

To calculate x_4, the value of x_4 in each respective tree is summed up, as $f(x_4) = f_1(x_4) + ... + f_K(x_4) = 0.7 + ... + (-1.5) = \delta_4$. In the paper about XGBoost [Chen and Guestrin, 2016], there is a similar visualization of how the value of an input x_i is determined. Inspiration from that paper has been used here.

2.2 The electricity market

A power market, synonymous with an electricity market, is a much more intricate marketplace than a market trading stocks or commodities. The electricity sold by suppliers is dispatched via the electrical grid that consumers connect to in order to obtain their purchased electricity. The electrical grid is a delicate system that requires constant surveillance and requires much more precision in comparison to when transporting and delivering commodities.

The Swedish electrical grid's balance between production and consumption is maintained by Svenska Kraftnät, which ensures a frequency of 50Hz. However, the responsibility to meet demand lies with the electricity suppliers, who are held liable under the Electricity Act for not delivering enough power or supplying too much. [*Balance responsibility* n.d.] Imbalances can arise from factors such as weather, and overloading transmission lines poses a blackout risk. To mitigate such imbalances, Sweden and Finland maintain reserves that can act as both supply and demand [Khodadadi et al., 2020].

Deregulation of several sectors has been a common economic policy since the 1980s, prevalent in sectors such as energy, telecommunications, aviation, financial services, and railways. The proponents of deregulation reason that deregulation improves the efficiency of infrastructures, making the sectors behave more like 'normal' industries. [Högselius and Kaijser, 2010]. The electricity market in Sweden underwent deregulation in 1996, establishing a wholesale market with Norway under the name Nord Pool where customers are able to select the supplier of electricity. Before, consumers would purchase electricity from local companies that either produced their own electricity or purchased it from other producers. There was no competition in the electricity market at the retail level, only at an industrial level [Niclas Damsgaard, 2005].

2.2.1 Electricity sources

Today, the electricity generated in Sweden is composed of hydro, nuclear, wind, thermal, and lastly solar. The distribution for 2021 was 43% hydro, 31% nuclear, 17% wind, 9% thermal and 1% solar. Solar represents a minor generation source. In total, approximately 60% of electricity generation came from renewable sources. [*Elproduktion och förbrukning i Sverige* n.d.] The electricity in Sweden today originates from both domestic and international sources [*The electricity market* n.d.]

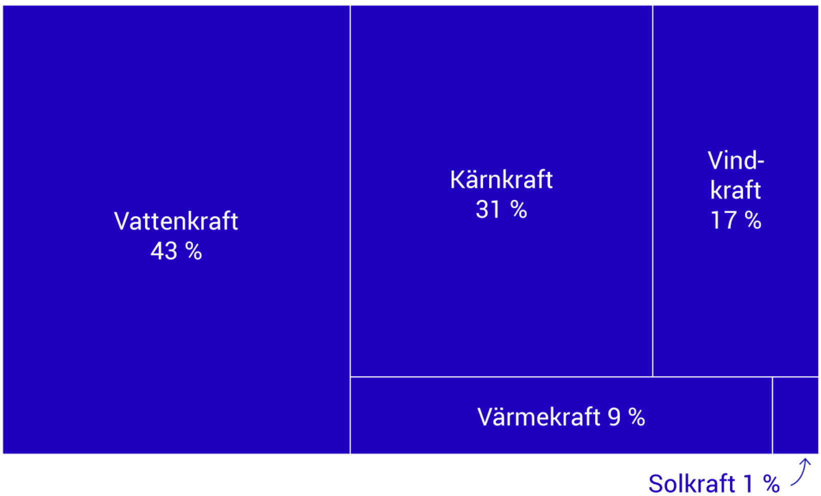


Figure 2.2 The different sources of electricity in Sweden for 2021. The large majority comes from three sources: Water, nuclear, and wind. Picture taken from [*Elproduktion och förbrukning i Sverige* n.d.]

2.2.2 Nord Pool

For the Nordic (Norway, Sweden, Denmark, Finland) and Baltic regions, the UK, and Central Western Europe (Belgium, Germany, the Netherlands, Luxemburg, France, and more [Uddin et al., 2022]) electricity can be traded on Nord Pool [*About us* n.d.] Nord Pool is one of the dominating markets in the Nordics and is one of the oldest exchanges in the world for electricity. Svenska Kraftnät jointly owns Nord Pool in partnership with Svenska Kraftnäts' Nordic and Baltic counterparts [*Operations and Electricity Markets* n.d.] The large majority of power available for sale on Nord Pool is supplied by the Nordic countries, e.g. Denmark, Sweden, Finland, and Norway. [Uddin et al., 2022].

On Nord Pool, buyers and sellers of electricity place hourly bids for the coming day. Market participants submit orders stating what quantity they are willing to sell or buy, and at what prices. The trading occurs on the spot market Elspot where suppliers and consumers are paired together on a sale supply curve and demand curve to find a spot price. The term spot price signals that the price reflects the current Short Run Marginal Cost (SRMC) at a given spot in time and space. The SRMC represents the cost to increase production by one marginal unit, without performing any new investments [Wangensteen, 2007]. The spot price is calculated frequently, in this case hourly, to reflect the operational state of the network. It also has to reflect the marginal cost including generation to where the customer is located. At the intersection of the supply curve and demand curve, the market-clearing price is calculated [Hjalmarsson, 2000]. The electricity traded on Nord Pool is physically delivered.

There exists a short-term market on Nord Pool called Elbas. The market allows for trading all hours of the year and serves as a secondary market to the spot market Elspot. Power market participants on Elbas are able to adjust their balances closer to the electricity being delivered. [Wangensteen, 2007]. The spot price on Nord Pool is determined as depicted in 2.3. The figure is taken from [Bang et al., 2023]. The system price is the unconstrained price in the spot market where it is assumed that there are no transmission restrictions in the grid. The system price serves as a reference for other financial trades in the Nordic market.[Wangensteen, 2007]. It means that if a producer has sold a number of MWh units in a forward contract for 10 EUR/MWh and the system price is realized at 9 EUR/MWh, the producer collects the difference in cash, i.e. 1 EUR/MWh [Lundin, 2021]. The price also acts as a reference for bilateral contracts that are established outside of power markets [Jablstrok;onacute;ska et al., 2012].

Outside of Nord Pool, there exists a futures market on NASDAQ OMX, the Stockholm exchange. In the futures market, consumers can trade contracts of electricity that stretch multiple years and offer long-term security rather than short-term security of supply [*Electricity trade* n.d.]

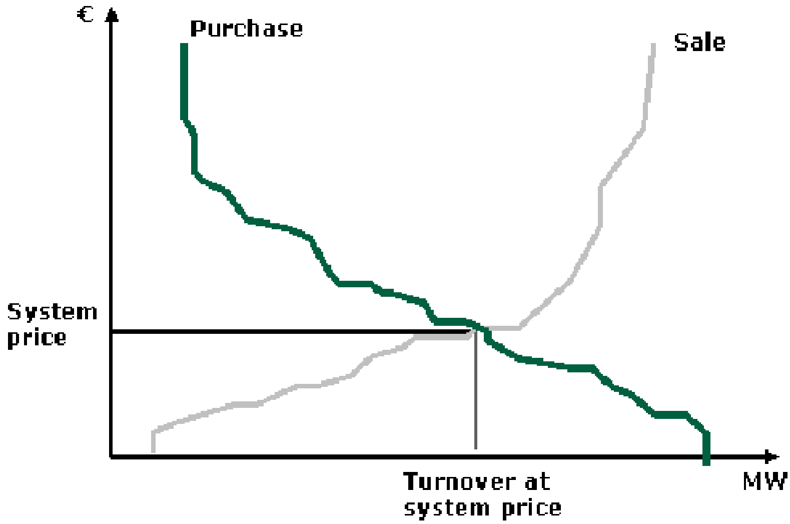


Figure 2.3 The demand and supply curves dictate where the system price ends up. At the intersection, the system price is found. Picture taken from [Bang et al., 2023]

2.2.3 Bidding areas

The Nord Pool markets are divided into distinct bidding areas since available transmission capacity may vary which has an impact on the amount of power that can be transmitted. Therefore different areas predominantly have different prices for a given day. The number of bidding areas per country is not fixed, Finland and Latvia both constitute one bidding area. Sweden constitutes four bidding areas (SE1, SE2, SE3, SE4). Germany constitutes four bidding areas, however, they always have the same price no matter what. [*About us* n.d.] Bidding zones of predominantly Nordic countries are displayed in 2.4.

Figure 2.4 The bidding zone, primarily showing the zones in the Nordic countries. Picture gathered from [Kuosmanen and Johnson, 2021].

2.2.4 Import & export

Sweden conducts electricity imports and exports with Norway, Finland, Denmark, Poland, Germany, and Lithuania. Sweden is a net exporter, regularly exporting more electricity than importing. [*Exchange* n.d.] Since the trading is conducted between borders, the overall demand for the Swedish electricity system is not merely the demand from the Swedish customers but the sum of demand from all countries trading with Sweden. One influence on the electricity price in Sweden is Germany.

Germany has a power market of greater size than the power markets of the Nordics combined. As a result, Germany influences a significant demand on the Nordic power markets by trading with Denmark, Sweden, and Norway, and therefore impacts the price formation for the entire region. [Jääskeläinen et al., 2022]. The country has to a high degree maintained its dependency on commodities such as oil and natural gas to obtain a secured electricity supply. Moreover, the country does not have a high share of natural resources and relies on imports from other countries for the electricity supply to meet the demand. The circumstances of the German power market sway the electricity price in the direction of natural gas and coal commodity prices, as there are little to no other options for producing electricity in the magnitude required. The dependency is beneficial when commodity prices are low, and conversely, when the commodity prices are high, the dependency is detrimental to consumers as electricity prices rise due to increased production costs. In addition to the dependency on commodities, Germany has also made it increasingly dependent on specific countries to deliver the commodities, instead of diversifying the source of the commodities. [*Germany - Country Commercial Guide* n.d.]

When Russia ceased operations of the Nord Stream pipelines, 60% of the total gas supply to Germany was cut off ["Ukraine war: How Germany ended reliance on Russian gas" 2022], driving the demand for electricity from other sources upwards. As Germany deals with the offset in supply, the country explores other options, and since Germany is desperate for any kind of solution, they are likely more willing to bid higher for electricity than other actors to ensure that Germany receives the required supply. As a result, electricity prices in Nordic countries rose sharply in line with Germany's supply issues. On the 19th of August, Gazprom, a Russian state energy giant stated that it will halt natural gas flow to Europe for 3 days by the end of the month for maintenance [Staff, 2022b]. Russia shut down the Nord Stream pipelines at the end of August [“Nord Stream 1: How Russia is cutting gas supplies to Europe” 2022] for good.

2.2.5 European Emission Allowances

By 2005, Nord Pool introduced the trading of European Emission Allowances (EU Allowances, EUA). Emission allowances are a market tool to restrict pollution in a manner such that an agent cannot release more pollution than the EAUs held by the agent. The allowances are ”an entitlement to emit one tonne of carbon dioxide equivalent gas.” [*EUA Futures* n.d.] The allowances provide an economic stimulus to reduce emissions as the number of EAUs needed decreases. Previously, it was known that the trading of emission allowances influenced the price of the power market [Jablstrok;onacute;ska et al., 2012]. [Jablstrok;onacute;ska et al., 2012] showed that the introduction of EAU trading has injected more irregularity in the price. The extent to which EAUs have an impact on the power market stretches beyond adding a new cost factor to the electricity price. There is greater speculation about the electricity price that has been present since the trading of EUAs was established.

2.2.6 Renewable sources

Over the last two decades, the share of energy from renewable sources has increased from 9.6% to 21.8% in the EU [*Renewable energy statistics* n.d.] A challenge that grid operators, buyers, and sellers on the market face are the irregular supply of renewable energy. Wind turbines only generate energy while the wind is blowing, and solar panels only generate energy while the sun is shining. For grid operators, this means that they cannot simply adjust the feed-in from the wind turbines and the solar panels as there is not a steady supply. Therefore, there has to be sufficient stored capacity to deal with the shortcomings of renewable energy.

An important term is merit-order. The marginal cost for wind turbines is low, hence why a higher wind generation may push the electricity price lower, which is called the merit-order effect of wind generation [Mulder and Scholtens, 2013]. Study Figure 2.5 to see a visual illustration of how the merit-order effect works [*Setting the power price: the* *merit order effect* n.d.] In [Ketterer, 2014], Janina C. Ketterer studied the link between wind generation and electricity prices in Germany. C. Ketterer establishes that the feed-in to the grid from renewable sources affects the electricity price by adjusting the merit-order curve to the right which excludes the most expensive generators of electricity. The outcome of the study showed that "intermittent wind generation" decreased the electricity price in Germany and boosted price volatility. [Ketterer, 2014].

In [Maciejowska, 2020], Katarzyna Maciejowska performed a similar study to [Ketterer, 2014] on the electricity price in Germany. Katarzyna Maciejowska also included solar energy in conjunction with wind energy. Maciejowska came to the conclusion that both wind and solar exert a dampening effect on the electricity price, a conclusion in line with the study by C. Ketterer. Evidently, as renewable sources become more prevalent in the market, the electricity market has to adapt to the changing environment. Energy from renewable sources evolves into an increasingly important component of the market and the pricing of electricity. [Maciejowska, 2020].

2.2.7 Market manipulation

As with any market, the ability to execute market manipulation exists. Nord Pool has a team allocated to oversee the trading taking place on the platform. The team is responsible for detecting unusual activity due to insider trading or market manipulation [*Market surveillance* n.d.] One exemplary case of market manipulation is the case of Barclays Bank PLC. Barclays Bank, by employing complex financial instruments, placed a multi-million dollar bet on electricity prices rising above a set price by a set time in the future. The bank then made trades on the electricity market to boost the price, even though most of the trades were unprofitable. As the purchasing of electricity inflated the price, Barclays Bank would win on its bet, collecting a sum far outsizing the losses from the unprofitable trades. Barclays was ultimately forced to pay $435 million in civil damages and $34.9 million in unjust profits. [Schöne, 2009].

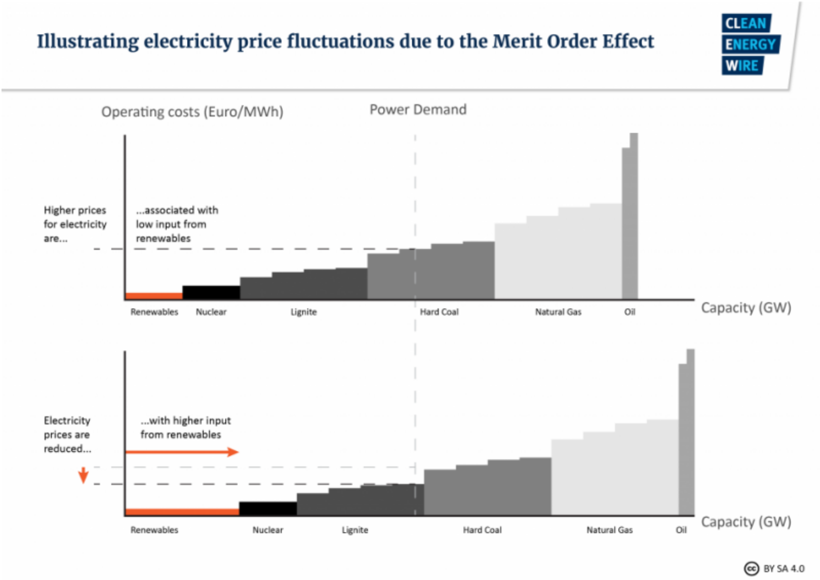


Figure 2.5 An illustration of the merit-order effect. When more sources of electricity with cheaper operating costs are introduced, the most expensive sources of electricity are pushed out. The effect reduces the overall cost of electricity, as the prices for producing electricity are lowered. Picture gathered from [*Setting the power price: the merit order effect* n.d.]

2.2.8 "Inefficient" pricing

A meaningful point to bring up is that the market price does not necessarily have to rely on variables that normally influence the electricity price. A study conducted in 2014 by Katarzyna Maciejowska, the same author as in [Maciejowska, 2020], found that fundamental drivers such as gas price, wind generation, and demand played a minor part in the pricing action for the UK market, instead, speculation or spot price shocks were responsible for up to 95% of the price volatility [Weron, 2014].

2.3 Optimization

There are two different solvers in Matlab used for optimization. The first one, 'intlinprog', is a Mixed-Integer Linear Programming (MILP) solver and builds on Branch-and-bound. The solver 'intlinprog' is used for the optimization of a single objective. The second solver, 'gamultiobj', finds the optimal values on the Pareto Front. The solver 'gamultiobj' is used for the optimization of multiple objectives.

2.3.1 Branch-and-bound

Branch-and-bound is a divide-and-conquer algorithm that divides a problem into subproblems over a tree structure, called the branch-and-bound tree. To illustrate how branch-and-bound algorithms operate, it is a good idea to define an example function that is supposed to be maximized. Given z, the goal is to find the input x in the domain S such that $f(x)$ is maximized.

$$z = \max\{f(x) : x \in S\} \tag{2.33}$$

If the subproblems are defined as $z^k = \max\{f(x) : x \in S^k\}$ for the indices $k = 1, 2$, such that $S = S^1 \cup S^2$, then the original problem can be expressed as

$$z = \max_k \ z^k \tag{2.34}$$

In Figure 2.6, the branching of the feasible region S, also called problem S, is located at the root node, and then branching from the root node is performed. Each z^k is said to be a node of the tree. From the root node S, when the problem is divided into subproblems S^1 and S^2, the process is called branching. S^1 and S^2 are said to be branches of node S. If the branching were to continue indefinitely, the problem S would only be divided into further explicit enumerations of the feasible region S. To avoid explicit enumeration, the branch-and-bound algorithm prunes a branch whenever possible, meaning that a subproblem is not branched into further nodes. A branch is pruned by utilizing bounds on the objective value of the subproblem. The use of branches and bounds details why the algorithm is called branch-and-bound. A node can be pruned if it fulfills any of the following scenarios

- Pruning by infeasibility. If a feasible region of a node subproblem is empty, the node will be pruned.
- Pruning by bound. If the upper bound calculated for a node subproblem is not greater than the lower bound on z, then the node is pruned because there is no point in searching for a feasible region of that node when we know that the best objective value we can obtain is not better than a solution we already know.
- Pruning by optimality. When it is possible to find the optional solution to a node subproblem S^i, then the node is pruned and its solution is stored as the incumbent if its objective value is better than the best we know so far.

If none of these conditions are fulfilled, branching of the node is done, to decompose the problem into smaller problems. [Kianfar, 2011] have been used as a source of information for this entire subsection about branch-and-bound.

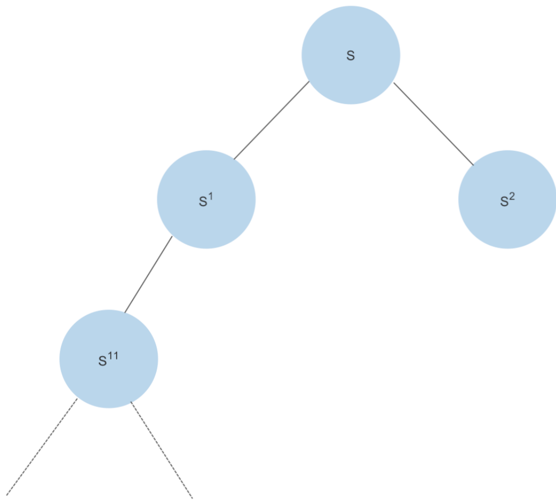


Figure 2.6 Tree fragmented down into subproblems. Each subproblem is a node itself. Inspiration taken from [Kianfar, 2011]

2.3.2 Pareto Front

In multi-objective optimization, there exist two or more objectives, which may or may not be in conflict with each other, that are tasked to be optimized. Since the optimization is done with respect to multiple objectives, there is no solution that specifically only optimizes a single objective. Instead, there have to be trade-offs between the objectives themselves, to obtain the solution where the optimization of the objectives themselves is as optimal as possible. There can be a finite or infinite number of optimal solutions, depending on the problem. [Tian et al., 2023]. Assume there exists an objective $f(x)$, where the objective itself consists of multiple different objectives $f_1, f_2, ...f_m$, which are supposed to be optimized. Then the multi-objective optimization can be expressed as

$$f(x_1, ..., x_n) \mapsto (f_1(x_1, ..., x_n), ..., f_n(x_1, ..., x_n)) \tag{2.35}$$

where $f : \mathcal{X} \to F$, where $\mathcal{X}$ is a n dimensional decision space and F is a m dimensional objective space.

To find the optimal optimization, multi-objective evolutionary algorithms are used to find the approximation between trade-offs such that the solution with respect to every objective is optimal. The approximation is said to be the Pareto Front. [Tušar and Filipič, 2015].

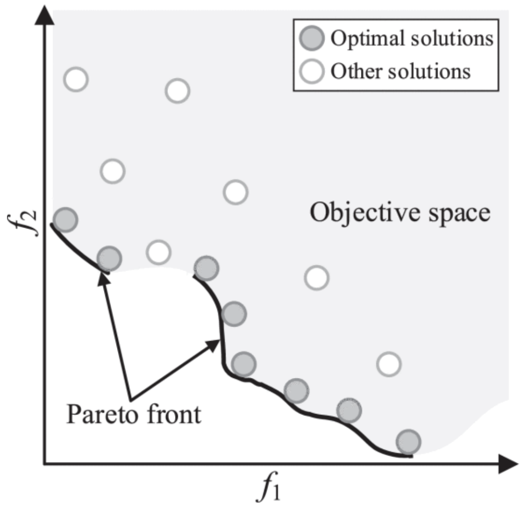


Figure 2.7 An illustration of the optimal solutions on the Pareto Front, from [Tian et al., 2023]. In the plot, The solutions between the objective f_1 and f_2 are displayed.

3

Prediction of electricity prices

To be able to forecast future electricity prices, which are numerical values, a machine-learning model is applied that trains on relevant data and outputs predicted electricity prices for the future. In this chapter, the focus is on forecasting future electricity prices. The two parts that will have the most focus are the configurations of the model that is used to forecast, and the dataset the model trains on. The model is to enable prediction of the electricity price 24 hours, 25 hours, 26 hours, all the way up to 120 hours, in advance.

The configurations of the model can have a large impact on the result, hence, it is necessary to understand what each parameter does and what combination of values results in the optimal solution. Having any parameter too large or too small drastically influence the ability of the model to train.

The dataset consists of multiple variables, some of which have a more apparent effect on electricity prices than others. Some variables are derived from others, not being an independent measurement of their own but instead various measurements between one or more variables. The variables have been gathered from different sources, later discussed in this chapter.

3.1 Model

A machine-learning model works by taking in unlabeled or labeled data. The data the model learns from is denoted as training data. If the training data is labeled, it is classified as *supervised* learning, and if the training data is not labeled, it is classified as *unsupervised* learning. During supervised learning, the model adjusts its weights until it has been fitted to the most optimal solution found, which occurs in the cross-validation during training. During unsupervised learning, the model attempts to discover patterns to solve association or clustering problems. A prime example of an area where supervised learning is used is predictive analytics. In predictive analytics, analytic systems can provide insight into business data points. It describes what enterprises can expect from enacting a specific business decision. [*Supervised Learning* n.d.] In Figure 3.1, the architecture of a supervised model is displayed. The model in this thesis takes supervised data as input, meaning it uses supervised learning.

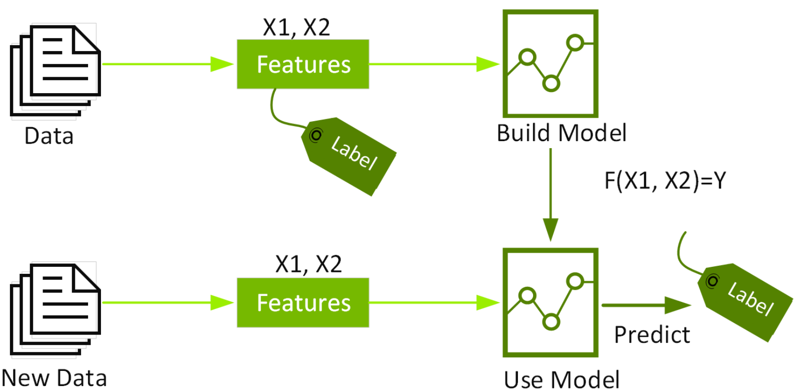


Figure 3.1 How a supervised machine-learning model is assembled, picture taken from [*XGBoost* n.d.] An unsupervised model would take data without labels, skipping the step for adding features to $X1, X2$ as shown in the picture.

The data is divided into two sets, training, and test data. The terms are defined as [*Training and Test Sets: Splitting Data* n.d.]:

- Training data - a subset to train a model
- Test data - a subset to test the trained model

A common split is 80-20 for the training and test set, respectively [*Training and Test Sets: Splitting Data* n.d.] The same split between the training set and test set is used in this thesis.

The machine-learning model used in predicting the price of electricity is XGBoost, described in Section 2.1.4. The XGBoost model was primarily chosen based on its status as a leading machine-learning library for multiple sorts of problems [*XGBoost* n.d.] and its flexibility with regard to hyperparameter tuning [*XGBoost versus Random Forest* n.d.] A hyperparameter is a parameter to the model that is adjustable by the user [*Parameters and Hyperparameters in Machine Learning and Deep Learning* n.d.] and regulates the learning process during training. XGBoost supports classification, regression, and ranking problems [*XGBoost* n.d.] The problem categorization in this thesis is a regression problem.

Besides its status as a leading library for machine learning for its superior prediction performance [*XGBoost* n.d.], XGBoost comes with features and performance boosts other libraries lack, some examples of superior aspects are [Dhaliwal et al., 2018]:

1. XGBoost is able to deal with missing data. XGBoost detects missing data and takes care of it.

2. XGBoost is optimized and is approximately ten times more efficient than other similar existing models, allowing for more experimental work such as adjusting parameters to be done in the same time frame as in comparison with other models.

3. Parallel processing is enabled, taking advantage of all available resources on the machine on which it's running. It is highly effective in classifying and preprocessing data.

4. Regularization enables efficient prevention of data overfitting.

To construct the model in this thesis, python is used as the programming language. Below is the XGBoost package imported into the python environment. The XGBoost library contains 6 different modules, and the module best fitting for the task at hand has to be chosen.

```
import xgboost as xgb

model = xgb.
            XGBRFRegressor()
            XGBRegressor()
            XGBModel()
            XGBClassifier()
            XGBRanker()
            XGBRFClassifier()
```

The task of forecasting numerical values, based on predictor variables, is said to be of regression character. There exists another very common characterization of some problems, said to be a problem of classification. With classification, the goal is to predict discrete class labels. In this thesis, the goal is to predict discrete continuous values, hence the problem is a regression problem. [*What is regression?* N.d.]

Since the task at hand is a regression problem, the XGBRegressor is used.

```
    model = xgb.XGBRegressor()
```

In this chapter, there will be two different types of model categorizations discussed. The first type of models will be referred to as the 'baseline models' and works on data from 2015 to 2018. The second type of models will be referred to as the 'primary models' which work on data from 2020 to 2023. Both models attempt to predict the electricity price from 24 hours up to 120 hours in advance.

There are mainly two approaches to dividing the problem into concrete applications. The first way is to train one model for each respective hour, $t+24, t+25, ..., t+120$ hours out. The result is 97 different models.

The second approach is to use one single model to predict 97 values in total, each for the respective hour in the future. When applied to XGBoost, XGBoost builds 97 different models internally, each model for each respective target variable [*Multiple Outputs* n.d.] However, to the user, it only appears to be one model.

Algorithm 1 Train 97 different models with one target variable

function FORECAST($N = 97$)
  for $i \leftarrow 1$ to N **do**
    $X, y_i \leftarrow$ Features and target variable for model_i
    $\text{Xtrain}, \text{ytrain}_i, \text{Xtest}, \text{ytest}_i \leftarrow X, y_i$
    $\text{model}_i \leftarrow$ initialize with hyperparameters
    $\text{model}_i.\text{fit}(\text{Xtrain}, \text{ytrain}_i)$
  end for
end function

Algorithm 2 Train 1 model

function FORECAST
  $X, y \leftarrow$ Features and target variables for model
  $\text{Xtrain}, \text{ytrain}, \text{Xtest}, \text{ytest} \leftarrow$ Features and target variables for model, divided into train and test sets
  model $\leftarrow$ initialize with hyperparameters
  model.fit(X, y)
end function

The first approach is applied in this thesis. There is no technical aspect to why the first approach is used, as the two alternatives are equivalent. It is rather a personal reason, as it appears to be a much more flexible approach to processing the results of the models.

A target variable is a variable that the model attempts to predict [*Target Variable* n.d.] To show what the input to the model is, and what the model outputs, assume that there exist feature values for 6 individual hours, and for each respective hour, the goal is to forecast the electricity price 24 hours up to 120 hours in the future. For each hour as input to the models with the corresponding feature values, the output is one single value. The number of hours in the future the prediction is done for depends on which of the 97 different models are utilized. Hence, if the input to the model is six different instances in time with the corresponding variable values, and each of those six inputs is evaluated by all models, predicting 24 hours in advance up to 120 hours in advance, the output is 97 different lists, where each list contains 6 values. Each model produces a list of six different predictions, one prediction for each input. Each index in the output from the model corresponds to the index of the input. In the output from the model predicting the price x hours out, index 1 in the output list represents the predicted price x hours from hour 1. In the same output, index 2 would represent the predicted price x hours from hour 2.

The scenario can be illustrated as shown underneath in Figure 3.2.

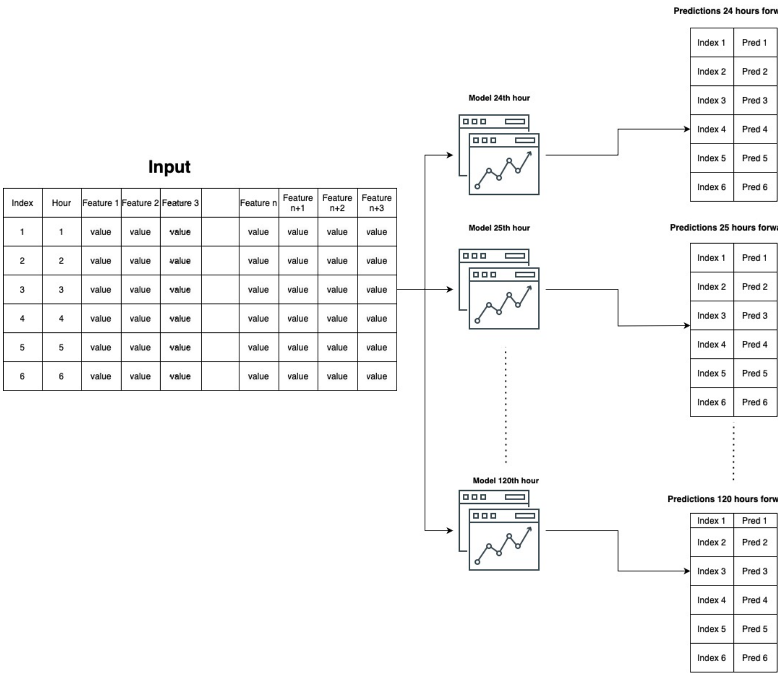


Figure 3.2 A visualization of how the input is processed to the individual models and what the output of each model is. The number of outputs from each model is dependent on how many inputs are fed to it.

3.1.1 Hyperparameters

The XGBoost model comes with a wide range of hyperparameters. The hyperparameters are valuable since it allows the operator to adjust the model according to individual needs. Data is different in every case, and the model is not a one-fits-all by default. With the number of hyperparameters being so great, only a collection of the most relevant hyperparameters are brought up. The hyperparameters discussed in this section are listed in the entry below.

1. n_estimators
2. seed
3. eval_metric
4. max_depth
5. learning_rate
6. objective
7. gamma
8. alpha
9. lambda

n_estimators is the number of gradient-boosted trees constructed when calculating the predicted value $\hat{y}_i$ [*Python API Reference* n.d.] Rewinding back to Equation 2.12, n_estimators sets the value of the constant K.

seed is a random number used to enable reproducibility. It means that with the same input, the output will always be identical. Gradient boosting methods are non-deterministic and the result is different each time. It is especially important when performing benchmarking. [*How to Use Random Seeds Effectively* n.d.] Without a set seed, it would be impossible to tell if removing or adding a feature decreased or increased the accuracy of the model. With a fixed number set as seed, the reproducibility of XGBoost is possible.

eval_metric is as the name hints, the evaluation metric for the model. The metric provided to the XGBoost model is used for observing the training result and early stopping [*Python API Reference* n.d.]

max_depth is the maximum depth of the base learners [*Python API Reference* n.d.], otherwise known as the decision trees [*XGBoost Documentation* n.d.] The default value of max_depth is 6. Increasing the maximum depth will make the model more complex, and hence more likely to overfit [*XGBoost Documentation* n.d.]

learning_rate is the step size for the regularized object being minimized. The learning_rate also goes under the name "eta" in the documentation [*XGBoost Documentation* n.d.] [*XGBoost Documentation* n.d.] describes it as the "step size shrinkage in update to prevents overfitting. After each boosting step, we can directly get the weights of the new features, and eta shrinks the feature weights to make the boosting process more conservative". This is where gradient, described in 2.1.3.1, emerges in XGBoost. The learning_rate has to be chosen under certain conditions in order to ensure that the algorithm finds the optimal solution, i.e. the minimum. The default learning_rate, or eta, is 0.3 [*Python API Reference* n.d.]

objective is the parameter deciding which loss function to be used in the model. The loss function is described under 2.1.4.1 in Equation 2.15.

gamma is the parameter γ in Equation 2.13. It regulates how the number of leaves in the tree is penalized as it is the constant in front of T, the number of leaves in the tree. $\gamma = 0$ tells the model not to penalize the complexity based on the number of leaves. The higher the value of γ, the more trees with fewer leaves are promoted. [*XGBoost Documentation* n.d.] states that gamma is the minimum loss reduction to justify creating further partitions of a leaf node in the tree. The higher the value of gamma, the higher the loss reduction has to be in order to expand a leaf node. As a result, the model produces more conservative estimates. [*XGBoost Documentation* n.d.]

alpha is the regularization of the first order, l_1, of the tree weights w_i [*Python API Reference* n.d.] [*XGBoost Documentation* n.d.] Regularization of the first order is referred to as "Lasso regression" [Guestrin, 2021]. The regularization is calculated as $l_1 = ||w_i||_1 = \sum_{i=1}^{T} |w_i|$ and with α, it becomes $l_1 = \alpha \sum_{i=1}^{T} |w_i|$. The higher the value of alpha, the more conservative estimations are produced from the model [*XGBoost Documentation* n.d.] $\alpha = 0$ is the default value [*XGBoost Documentation* n.d.]

lambda regulates the regularization of the second order, l_2, of the tree weights w_i [*Python API Reference* n.d.] [*XGBoost Documentation* n.d.] The second regularization of the second order is called "Ridge regression" [Guestrin, 2021]. The regularization is calculated as $l_2 = ||w_i||_2 = \sum_{i=1}^{T} w_i^2$ and with lambda, it becomes $l_2 = \lambda \sum_{i=1}^{T} w_i^2$. The expression appears in Equation 2.13 with a constant $\frac{1}{2}$ in front. The higher the value of lambda, the higher the requirements are for reducing the loss. [*XGBoost Documentation* n.d.]

3.1.2 Hyperparameter tuning

The tuning of hyperparameters has been performed on the primary models. The baseline models have the exact same parameters as the primary models. However, since the primary models are used for later application in this thesis, the hyperparameters have been tuned based on the result of RMSE gain or loss of the primary models, not the baseline models.

The first parameter that was tuned is learning_rate, i.e. the learning rate.

The learning rate, as previously discussed, is a delicate parameter that has to be chosen with care in order to achieve the best possible result. In Figure 3.3, the RMSE of the 24th hour is depicted, showing how the RMSE is variate based on the learning rate. The minimum RMSE is found at a learning rate of 0.22, different from the value used in this thesis. The reason why is that the learning rate of 0.22 fits well when looking at the 24th hour forecasted and not hours 25, 26, up to 120. It only looks at one single hour and ignores the rest. A learning rate of 0.08 is used due to the following two reasons

1. 0.08 has shown to be a valuable learning rate when testing models while continuously adding new features to its input

2. 0.08 reduces the overall error of forecasting

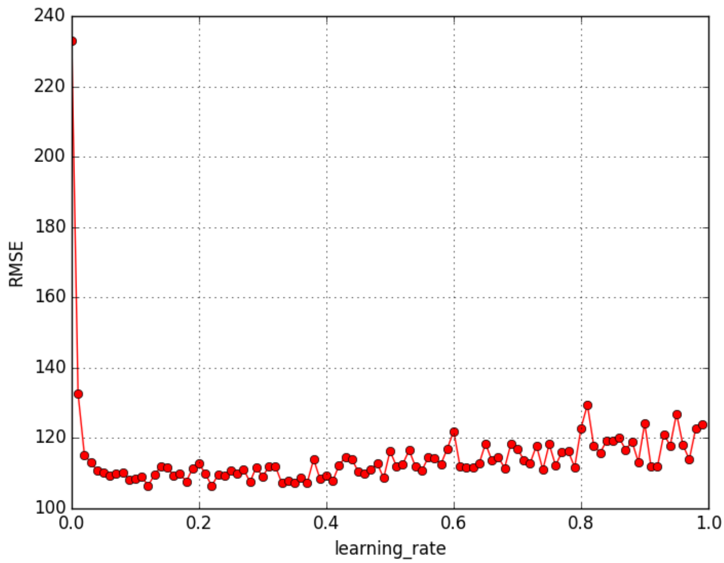


Figure 3.3 RMSE for the 24th-hour primary model dependent on the learning rate. The RMSE quickly descends and stabilizes for a while, starting to trend higher after 0,4.

There isn't any set of values for the seed that enables the model to produce more accurate results. As long as any number is chosen, and remains constant, the seed fulfills its duty. The seed was set to 14.

The eval_metric is set to 'rmse', being the most appropriate evaluation metric for the model. 'rmse' equals the equation under 2.1.1.

The number of n_estimators was initially set to 40 when experimenting with different features. It takes vast time to train models, therefore n_estimators being set to 40 gives a faster depiction of how the model performs, without having to wait for all 150 estimators to finish. Whenever the best version yet of the model was found based on 40 estimators, it was then trained on 150 estimators.

early_stopping_rounds was set to 30. With the early_stopping_rounds parameter, the n_estimator can essentially be set to anything above the number K_c where K_c is the number of iterations required to reach the most optimal solution. This is because the model will stop training after adding 30 trees without seeing any improvement and not necessarily continue until it reaches n_estimators iterations.

The max_depth was set to the default value of 6 [*XGBoost Documentation* n.d.] Having a max_depth less than 6 does not produce sufficiently complex trees to improve the accuracy of the model. On the other hand, having a max_depth of more than 6 resulted in the model producing trees with too high of complexity, adding noise to the predictions and depreciating the accuracy.

The objective is the most straightforward hyperparameter, being set to the default loss function stated in 2.1.4 under Equation 2.15. The value of the objective hyperparameter is 'reg:squarederror' where 'reg' stands for regression [*XGBoost Documentation* n.d.]

Gamma is untouched and set to the default value of $\gamma = 0$ [*XGBoost Documentation* n.d.]

Alpha is left unchanged at $\alpha = 0$.

Lambda is untouched and set to the default value of $\lambda = 1$ [*XGBoost Documentation* n.d.]

3.2 Dataset

The data used for training the models are gathered from multiple sources. The data collected consists of different attributes that are thought to contribute to the electricity price.

The different attributes of data are listed below.

- Energy data
  Energy data is the historical hourly price of electricity. It has the unit EUR/MWh. The data have been gathered from [*Dashboard* n.d.] The reason for collecting the data and feeding it as a feature of the model is that future prices may have a correlation with historical pricing movements. The writers in [Uribe et al., 2022] used the same source for collecting data about electricity markets.

  There are good arguments for analyzing historical data. For example, the entire foundation behind ARIMA is built on analyzing past data by constructing moving averages models (not the same thing as a simple moving average) and autoregression.

There are numerous factors that can help with forecasting. Quantifiable factors such as historical prices affect the electricity price [Kuo and Huang, 2018].

- Historical weather data
  The historical weather data contains information about hourly temperature, windspeed, and precipitation. The data have been gathered from open-meteo.com [*Free Weather API* n.d.] The idea is that the wind, precipitation, wind gusts, temperature, and weather code may have an impact on the electricity price. The wind is especially considered important since it accounts for quite a large portion of all electricity generated in Sweden. The wind is measured in m/s, the precipitation is given in mm/h, wind gusts are in m/s, the temperature is in Celsius, and the weather code is a numeric value corresponding to some weather state, given by the values in Table 3.2.

  Authors in [Weron, 2014], [Chang et al., 2019], [Lehr and Valdes, 2021], and [Jääskeläinen et al., 2022] all agree that weather-related factors play an important role in the electricity price.

- System demand for Sweden
  The system demand for Sweden is the hourly demand on the electrical grid in the entire of Sweden. It is provided in units of MW. The data have been gathered from [*Dashboard* n.d.] The idea is that the demand for electricity is a factor in determining the electricity price.

  Authors in [Weron, 2014] agree that demand is a factor in pricing. Deepak Singhal and K.S. Swarup showed that the load demand is a strong factor determining the electricity price [Singhal and Swarup, 2011], corresponding to what the authors in [Weron, 2014] claimed.

  Authors in [Stéphane et al., 2019] evaluated the long-term elasticity of demand in France and found the price elasticity in relation to demand is -0,8. The finding is in line with previous research.

The negative elasticity implies that there is an inverse relationship between the electricity price and the demand, i.e. when the electricity price is low, the consumption rises, and vice versa.

- System demand for Germany
  The system demand for Germany is the hourly demand for electricity in the entirety of Germany. It is provided in units of MW. The data has been gathered from [*Dashboard* n.d.]

  The reason for using system demand in Germany follows the same conclusion in the bullet-point above, i.e. the system load in Germany impacts the electricity price in Germany. As previously stated, Germany has an impact on the entire price formation in the region. One additional point is that if the price is high in Germany, the incentive to import from neighboring countries is higher, driving up the demand in related countries. Therefore, the current demand in Germany could play a contributing factor to the price of electricity in Sweden.

- System generation in Sweden
  System generation in Sweden is the hourly generation of electricity provided to the electrical grid. It is provided in units of MW. The data has been gathered from [*Dashboard* n.d.]

  The electricity price is susceptible to changes in short-term demand contra supply and can rapidly sway the market in either direction depending on the variable setting off the imbalance [Afanasyev et al., 2021]. Therefore, it is important to capture both sides of the spectrum.

- System generation in Germany
  System generation in Germany is the hourly generation of electricity provided to the electrical grid. It is provided in units of MW. The data has been gathered from [*Dashboard* n.d.]

  It follows the same reasoning between Sweden and Germany as previously discussed.

- Oil data
  Oil data contains information about daily historical prices of

Crude, WTI, and Brent oil. It is provided in units of US dollars/barrel.

Mauro Castelli, Aleš Groznik, and Aleš Popovič conducted forecasting of electricity prices on the German energy exchange by using novel genetic programming. In their paper, they claim that the spot price of crude oil, in dollars, impacts the spot electricity price in the model they built. [Castelli et al., 2020].

The authors in [Weron, 2014] also agree that oil price is a fundamental driver of the electricity price.

- Natural gas data
  Natural gas data contains information about daily historical prices of natural gas prices. The natural gas price is collected from the ICE Dutch TTF Natural Gas Futures - Apr 23 [*ICE Dutch TTF Natural Gas Futures - Apr 23 (TFMBMc1)* n.d.] It is provided in EUR/MMBTU.

  The authors in [Weron, 2014] agree that the natural gas price is a fundamental driver of the electricity price.

  [Uribe et al., 2022] demonstrated that natural gas has a greater impact on the electricity price than the weather in the countries analyzed.

- Natural gas flow from Russia to Germany
  Natural gas flow from Russia to Germany contains information about how much natural gas is being transferred from Russia to Germany per month. It is provided in the unit of Mm3/h.

  Considering 60% of the natural gas to Germany was imported from Russia, the idea is that the pipeline shutting down would have a considerable influence on the price of electricity.

- Unavailable electricity to the grid
  Unavailable electricity to the grid contains information about outages to the electrical grid, both planned and unplanned. It is provided in MW. The data has been gathered from [*Dashboard* n.d.] It is provided in units of MW.

The authors in [Weron, 2014] agree that forced outages are a fundamental driver of the electricity price.

- Scheduled energy exchange with countries
  Scheduled energy exchange with countries contains information about the scheduled exchange of electricity with other countries from and to Sweden. It is provided in MW. The data has been gathered from [*Dashboard* n.d.]

  If a country has a large quantity of electricity being exported, the supply is inherently lower pushing prices higher. The scheduled exchange is available for the following day on [*Dashboard* n.d.]

- Physical energy exchange with countries
  Physical energy exchange with countries contains information about the physical exchange of electricity with other countries from and to Sweden. It is provided in MW. The data has been gathered from [*Dashboard* n.d.]

  The physical exchange is the actual flow of electricity between borders, and the number is supposed to be the same quantity as the scheduled exchange.

- Reservoirs of Sweden
  The levels in the reservoirs are being tracked and the information contains details about the water levels in Sweden. It is provided in units of TWh.

  The authors in [Huisman et al., 2014] found that hydro supply has a decreasing effect on the day-ahead electricity prices.

  The data was provided by Nord Pool through a joint account for students on their file-sharing platform.

- Pricing of carbon emission rights
  The carbon emissions limit how much carbon a party can emit. Most companies receive an annual allocation of free allowances [*What is emissions trading?* N.d.] It is provided in units of EUR.

  Pricing of carbon emission rights influences the price according to [Jablstrok;onacute;ska et al., 2012].

- EUR/SEK currency pair
  While the Euro trends higher versus the Swedish crown, more purchasing power is afforded to Germany. In reverse, Germany loses purchasing power. The feature of the EUR/SEK pair evaluates if the purchasing power of Germany has an influence on the electricity price.

- Coal prices
  The unit is provided in EUR/MWh. Germany is reliant on coal, which begs the question if high coal prices promote Germany to reduce electricity production from coal and pivot to alternative sources, where one source consists of imports.

  The authors in [Weron, 2014] agree that the coal price is a fundamental driver to a lesser extent of the electricity price.

3.2.1 Extrapolating to patch gaps in data

The time series is ideally built without gaps between two points in time. If there were to be gaps in the time series, it could make it harder to capture the relationship between variables. The reason is that there might be a change in the price that could be explained by a set of variables, however, if the model is never given the opportunity to analyze it, it will unfortunately not find the association. Another reason why there are not supposed to be any gaps in the time series is that the model should be able to perform electricity prices for any day of the week and for all hours of the week. If the pricing behaves differently during weekends and there is no data for it, it cannot be captured.

The datasets with patches in data have missing data that come in different forms. Some data sets are missing values for weekends, while other datasets only have one datapoint for an entire month. There are 5 datasets that do not contain continuous data and would inject gaps in the final dataset used for training the model. The solution to extrapolate data is tersely illustrated below.

3.2.1.1 Series extrapolated.

1. Oil prices
   The first such dataset is the time series for oil prices. The oil

markets only trade during weekdays, so the oil price remains the same as the last closing price on Friday. The data was extrapolated such that Saturdays and Sundays had the same oil price as the last closing price on the preceding Friday.

2. Natural gas prices
   The second dataset is the price of natural gas. Natural gas trades during the same hours as oil does, so the same issue occurs here with missing prices for weekends. The data was extrapolated such that Saturdays and Sundays had the same natural gas price as the last closing price on the preceding Friday.

3. Carbon emission prices
   The third dataset is the price of carbon emission units. Carbon emission units are strictly only traded Monday through Friday. Here, the last closing price on Friday was extrapolated for Saturday and Sunday with the preceding Friday.

4. Magazine content
   The fourth dataset is the dataset of magazine content. The magazine contents gathered from Nord Pool are only recorded once a week, so the data have been extrapolated to all hours from Monday through Sunday for each respective week.

5. Natural gas imports
   The fifth dataset is the dataset for natural gas imports from Russia to Germany. The values are only prompted monthly which means that the data first has to be extrapolated to a daily time frame, and then lastly to an hourly time frame.

The extrapolation of data may be a deviation from the truth, which may add some instability to the model. However, considering the option of more data to train on, in contrast to potential small errors, it is worth it.

3.3 Manipulating data to assemble features

3.3.1 Moving averages

The moving average used in this thesis is called a simple moving average, which calculates the average in a sliding window of size k starting at index j, such that the moving average is

$$MA(k, j, \mathbf{y}) = \frac{1}{k} \sum_{i=j-k}^{j} \mathbf{y}_i \tag{3.1}$$

sampled from the vector $\mathbf{y}$ consisting of numbers.

3.3.2 Shift operation

The shift operation in Python within the package "pandas" moves the values k steps forward if k is negative and backward if k is positive. The table below shows how different values of shift rearrange the values. A positive shift tells index i the value of k steps back in history, i.e. backward-looking. A negative shift tells index i the value of k steps forward in history, i.e. forward-looking. If there is no value accessible, the column takes on the value NaN.

3.4 Baseline models

The data which the primary models train on are extremely volatile. It may produce some instabilities and skewed results. Hence, the idea of a baseline model was prompted, where the idea was to construct similar models of XGBoost to the primary models but train on data with less volatility. If the implementation of the baseline models was proven to be efficient with reasonable results, the possible instabilities and skewed results in the primary models could be explained by the inherently volatile data. Data from 2015 to 2018 were fetched and used to represent the scenario of less volatility in the electricity market.

3.4.1 Mismatched distributions

To showcase the difference in the price behavior of what is now referred to as the baseline models, being trained on data from 2015

Index	Value	$k=-1$	$k=-4$	$k=2$
1	x_1	x_2	x_5	NaN
2	x_2	x_3	x_6	NaN
3	x_3	x_4	x_7	x_1
4	x_4	x_5	x_8	x_2
5	x_5	x_6	x_9	x_3
6	x_6	x_7	x_{10}	x_4
7	x_7	x_8	x_{11}	x_5
8	x_8	x_9	x_{12}	x_6
9	x_9	x_{10}	x_{13}	x_7
10	x_{10}	x_{11}	x_{14}	x_8
11	x_{11}	x_{12}	x_{15}	x_9
12	x_{12}	x_{13}	x_{16}	x_{10}
13	x_{13}	x_{14}	x_{17}	x_{11}
14	x_{14}	x_{15}	x_{18}	x_{12}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Table 3.1 How shifting works in Python.

to 2018, and the primary models, being trained on data from 2020 to 2023, see Figure 3.4. The distribution curves of the electricity price for both 2015-2018 and 2020-2023 are shown in Figure 3.4, where the spread of values in the 2020-2023 case is much larger than in the 2015-2018 scenario. The table below illustrates data points showcasing the differences between the two datasets.

Dataset	Maximum price (€/MWh)	Mean (€/MWh)	Standard deviation (€/MWh)
2015-2018	255.02	32.6	14.4
2020-2023	799.97	87.1	103.9

The ratio of the standard deviations, $\frac{103.9}{14.4}$, is close to 7.2, showing the difference in volatility between the two datasets.

3.4.2 Preprocessing & feature engineering

The features supplied to the baseline models are provided in Tables 3.3, 3.4, and 3.5. The scoring of the models is evaluated based on the

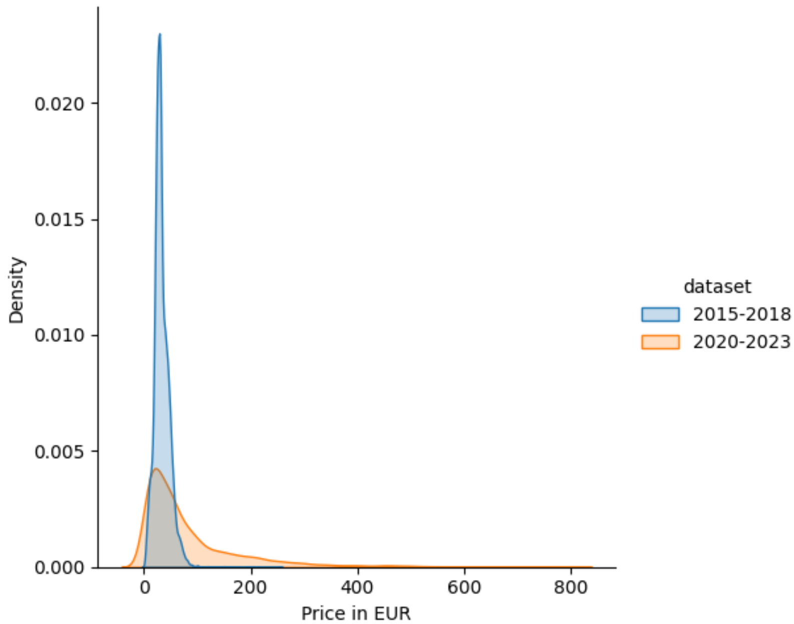


Figure 3.4 Distribution curves of electricity prices in the two different time periods examined. Evidently, 2015-2018 has a lower degree of volatility than 2020-2023, having a higher level of concentrated density around its mean. The data of electricity prices are taken from [*Dashboard* n.d.]

MCRMSE metric of the predicted test data.

3.4.2.1 Weather. The first weather feature is the temperature. The temperature dictates some degree of electricity usage, as when it is abnormally cold, there is more electricity used for heating. In the opposite direction, if it is abnormally warm, there will be more electricity used for cooling (ACs, etc). [Uribe et al., 2022] states that temperature has a positive effect on the electricity price as more electricity is used for cooling in warmer countries. In colder countries, temperature also has a positive effect on the electricity price as more electricity is used for heating. At 13.00 each day when the electricity price is determined, there are weather forecasts available to gather information from. The temperature features are the current temperature, the temperature in 1 hour, 2 hours, ..., 120 hours, with the names 'Temp, Temp t+1, Temp t+2, ..., Temp t+120' located in Table 3.3. Generally, there is a limit to how far out the weather forecasts are reliable. Therefore, the limitation is set at 5 days in the future, as 5-day forecasts are right about 90% of the time [*How Reliable Are Weather Forecasts?* N.d.]

With wind, the windspeed for 1 hour, 2 hours, ..., and 120 hours in the future proved valuable. As wind turbines produce a substantial part of the total electricity generated in Sweden, it is expected that future wind forecasts should play a part in the electricity price auction. The wind features 'Windspeed', 'Windspeed t+1', ..., and 'Windspeed t+120' are in Table 3.3. Wind gusts, partially linked to the windspeed feature, record the wind gusts for 1 hour, 2 hours, ... 120 hours in the future in Table 3.3 under the names 'Wind gusts, Wind gusts t+1, Wind gusts t+2, ..., Wind gusts t+120'.

The precipitation for 1 hour, 2 hours, ... 120 hours in the future showed that it was a valuable addition to the set of features. It is displayed in Table 3.3 under the names 'Precip, Precip t+1, Precip t+2, ..., Precip t+120'.

The last weather feature is 'weathercode' in Table 3.3, which is a feature for "Weather condition as a numeric code" [*Weather Forecast API* n.d.] as stated by the documentation for Open-Meteo.com [*Free Weather API* n.d.] It is denoted by the names 'Weathercode, Weathercode t+1, Weathercode t+2, ..., Weathercode t+120'.

The weathercode feature can take on multiple whole-number values [*Weather Forecast API* n.d.], described in Table 3.2.

The original idea was that the weather forecasts would provide the values for the weather features. However, finding a service that provides historical weather forecasts were either inapt because it did not provide weather forecasts for the dates needed, or wanted monetary compensation for the data. To circumvent the hurdle, the weather data points are shifted backward to simulate a forecast. The problem with this approach is that the 'forecasts' in the model are always guaranteed to be true, as opposed to how weather forecasts operate in the real world, without a guarantee. This is different from how the model originally was

Code	Description
0	Clear sky
1,2,3	Mainly clear, partly cloudy, and overcast
45, 48	Fog and depositing rime fog
51, 53, 55	Drizzle: Light, moderate and dense intensity
56, 57	Freezing Drizzle: Light and dense intensity
61, 63, 65	Rain: Slight, moderate and heavy intensity
66, 67	Freezing Rain: Light and heavy intensity
71, 73, 75	Snow fall: Slight, moderate, and heavy intensity
77	Snow grains
80, 81, 82	Rain showers: Slight, moderate, and violent
85, 86	Snow showers slight and heavy
95 *	Thunderstorm: Slight or moderate
96, 99 *	Thunderstorm with slight and heavy hail

Table 3.2 The weather codes and their descriptions from [*Weather Forecast API* n.d.]

supposed to operate, by taking data from a point in time with historical data and the current weather forecast to forecast the electricity prices. However, it will have to suffice given the inability to access historical weather forecasts.

The 'forecasted' (the true value in reality) value for x hours in advance is shifted backward by x steps, $k = -x$, and so on for all values 1, 2,..., 120 for windspeed, precipitation, weather code, wind gusts, and temperature.

3.4.2.2 Commodities. Performing feature engineering quickly showed that there is no positive correlation between the electricity price and the oil price, both the Brent and WTI variants. The difference between WTI and Brent is that Brent is extracted from the North Sea and WTI is extracted primarily in the US, Texas [*Brent vs WTI: Which crude to trade in 2022?* N.d.] Feeding the daily change in price for both variants to the model proved helpful, denoted 'Brent change %' and 'WTI change %' in Table 3.4. The idea here is that the trends of oil prices tend to follow a change in energy demand in terms of oil, and a part of that energy is used to convert into electricity. Hence, if the price of oil is expensive, the result is that electricity prices rise. The idea is the same for when the oil prices tend downward, i.e. the energy demand is relatively low. Sweden has very little if any, generation of electricity by utilizing oil. However, as previously discussed in 2.2, Sweden's electricity price depends on the neighboring countries Sweden conducts imports and exports of electricity with.

The price of natural gas, instead of only derivatives of the price (i.e. daily change in percentage), plays a role in predicting the electricity price and improves the prediction accuracy. The price itself, with the daily price change in percentage and the mean and standard deviation of the daily change expressed in percentage of natural gas prices, are supplied to the model, making the prediction accuracy better.

3.4.2.3 Miscellaneous. The total aggregated electricity supplied to the grid, feature 'Supply SE4' in Table 3.5, gives off an indication of how much supply there is on the supply side. The buyers on the spot market can then determine how aggressive the pricing should be from their side, based on the current trend of how much amount of supply is available.

The hour of the day plays an important role when examining the electricity price. It is injected by the feature 'Hour' in Table 3.5. The demand for electricity in MW per hour during the day is plotted in Figure 3.5 and the average price per hour during the day is plotted in Figure 3.6. The figures show that both the demand and price vary depending on the hour of the day.

Providing the feature with the hour of the day allows the model to pick up the obvious pattern that the price is generally higher for some hours during the day and generally lower for some hours during the day.

The feature 'Net MW scheduled' in Table 3.5 says how much electricity is being imported minus how much electricity is being exported, on a scheduled basis. When more electricity is being exported than imported, the supply of electricity diminishes, pushing prices higher if the demand remains constant. If there is more being imported than exported, the supply expands, pushing prices lower if the demand remains constant.

The feature 'Cheapest hour percentage' is the probability of each hour of the day statistically being the most expensive hour of the day, over the period 2014-12-31 to 2018-12-31. The probabilities are gathered by analyzing at what hour the lowest electricity price occurs for all days in the dataset and then dividing each hourly count by the total days

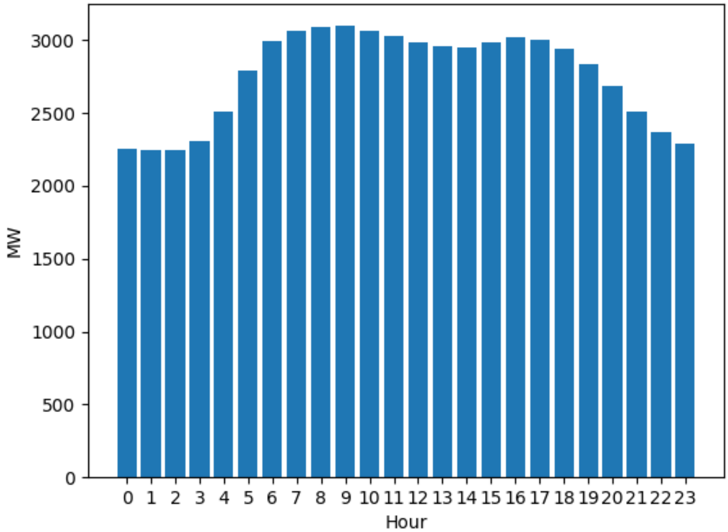


Figure 3.5 Average demand in Sweden per hour during the day during 2014-12-31 - 2018-12-31. The demand pikes around 09.00 and is at its lowest point during the evening/night.

analyzed. The result is a probability in the range $[0, 1]$ for each hour. The distribution is shown in Figure 3.7. The probabilities have been calculated by analyzing the prices from [*Dashboard* n.d.]

The feature 'Exp hour percentage' follows the same logic but for the most expensive hour of the day instead of the cheapest, over the period 2014-12-31 to 2018-12-31. The probabilities are gathered in the same way as with the cheapest hour, but now analyzing what hour during the day the most expensive electricity price occurs. The probability of hour x being the most expensive hour is shown in Figure 3.8. The probabilities have been calculated by analyzing the prices from [*Dashboard* n.d.]

While the moving averages provide a historical mean, the slope of the curve says how fast a given curve changes over time, hourly in this

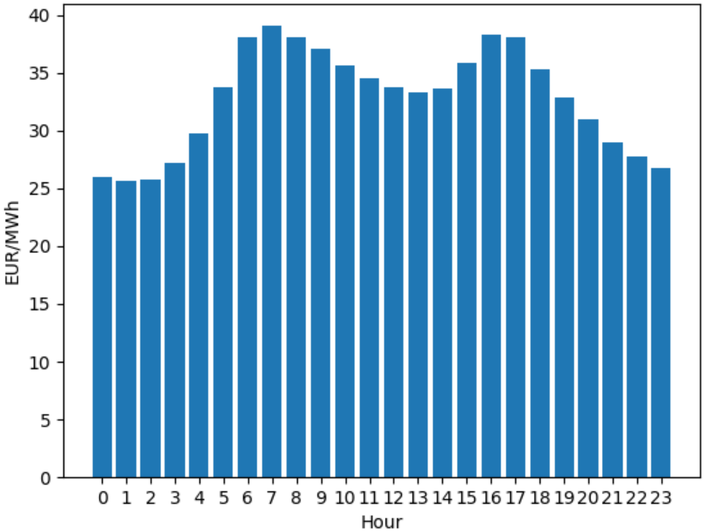


Figure 3.6 Average price per hour during the day during 2015-01-01 - 2019-01-01. The graph has two distinct peaks, one at 07.00 and the other at 16.00.

case. The information is useful as it tells the model how the electricity price behaves in relation from hour to hour. For example, if the slope is significant in any direction, it lets the model derive that with the current values of features to the model, the price is more volatile and there is a greater level of uncertainty. Then, hopefully, after studying a number of scenarios with changing slopes and fluctuations in electricity price, the model finds a valuable relationship between the two variables. There are two different features that involve the slope of the electricity price curve. The 'Slope 3 days mean' in Table 3.5 looks at the mean of the slope of the last 72 data points. 'Slope day before' is simply the slope of the day before.

The feature 'Price 2 days ago' in Table 3.5 is the value of the electricity price 48 hours ago, hence the feature is shifted by 48 hours. It

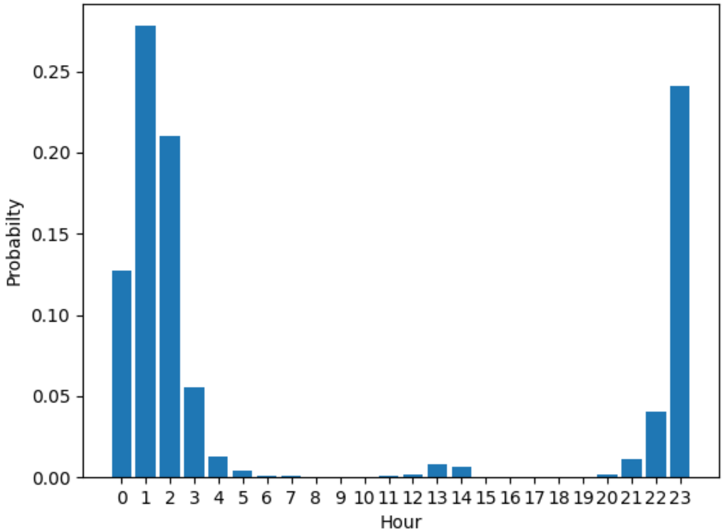


Figure 3.7 Probability of an hour being the cheapest during the day, between 2014-12-31 - 2018-12-31. 01.00 is statistically most often the cheapest hour of the day.

can be viewed as a laggard that takes advantage of the properties, in the same way, the ARIMA model does with the parameter p detailed in 2.1.2.

'Load Germany' in Table 3.5 is the electricity in demand for Germany. It covers the entire country which is divided into multiple bidding zones. The demand for electricity in Germany proving to increase the accuracy of the model supports the argument made in [Jääskeläinen et al., 2022], of Germany having an impact on the pricing formation in the nearby region.

'Supply Germany' in Table 3.5 increasing the accuracy of forecasting also naturally support the argument in [Jääskeläinen et al., 2022]. It shows that the supply of electricity in Germany impacts the price formation.

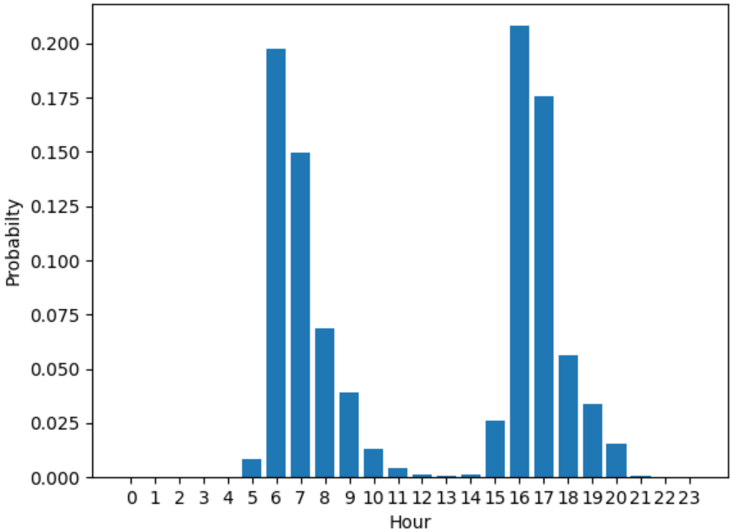


Figure 3.8 Probability of each hour being the most expensive during the day, between 2015-01-01 - 2019-01-01. 16.00 is statistically the most expensive hour of the day, with 06.00 coming in at a close second.

'Load SE4' in Table 3.5 is the demand for electricity in the SE4 region. The demand naturally impacts the concluded price.

'Price' in Table 3.5 is the current electricity price. Inserting the feature of the current electricity price in order to forecast future electricity prices improve the model by a considerable degree, supporting the statements made in [Kuo and Huang, 2018] of quantifiable factors such as historical prices affecting the electricity price.

As discussed before, the electricity price for the following day is determined at 13.00 on the preceding day. The features introduced that represent the next day's prices are called 'Price next+1, Price next+2, Price next+3, ..., Price next+24'. The next 24 hours all have the same value for 'Price next+1', 'Price next+2', ..., 'Price next+24'. 'Price next+1' is not the price in the next hour, but simply the price for the next day at 00.00. 'Price next+2' is simply the price for the next day at 01.00 and not the price in two hours. The model can now take advantage of the upcoming prices to make better predictions about the upcoming future.

At the same time, the model is fed the hourly electricity prices of the last 24 hours, in features by the name 'Price last-1, Price last-2, Price last-3, ..., Price last-24'. The model can now utilize the last 24 hours' worth of prices, and the upcoming 24 hours of prices, to make much better predictions.

The main reason why multiple rolling averages have been experimented with as features and possibly why it has improved the model is that moving averages smoothen out the curve by removing noise that makes it difficult to find a meaningful relationship. In doing so, it maintains the main movement of the curve, without all of the minor fluctuations. [*Hyndman, R.J., Athanasopoulos, G. (2021) Forecasting: principles and practice, 3rd edition, OTexts: Melbourne, Australia.* N.d.] The reason why values related close in time are sampled is that they are thought to share trend-cycle components similar in value. The trend component can then be calculated by taking the average of observations surrounding the data point. [*MATH6011: Forecasting* n.d.]

Feature	Description
Windspeed	The current hourly windspeed in m/s
Windspeed t+1	Windspeed in 1 hour
Windspeed t+2	Windspeed in 2 hours
Windspeed t+3	Windspeed in 3 hours
⋮	⋮
Windspeed t+120	Windspeed in 120 hours
Precip	The current hourly precipitation in mm
Precip t+1	Precipitation in 24 hours
Precip t+2	Precipitation in 25 hours
Precip t+3	Precipitation in 26 hours
⋮	⋮
Precip t+120	Precipitation in 120 hours
Wind gusts	The current hourly wind gusts in m/s
Wind gusts t+1	Wind gusts in 1 hours
Wind gusts t+2	Wind gusts in 2 hours
Wind gusts t+3	Wind gusts in 3 hours
⋮	⋮
Wind gusts t+120	Wind gusts in 120 hours
Weathercode	The current hourly weather code
Weathercode t+1	Weather code in 1 hour
Weathercode t+2	Weather code in 2 hours
Weathercode t+3	Weather code in 3 hours
⋮	⋮
Weathercode t+120	Weather code in 120 hours
Temp	The current temperature in Celsius
Temp t+1	The temperature in 1 hour
Temp t+2	The temperature in 2 hours
Temp t+3	The temperature in 3 hours
⋮	⋮
Temp t+120	The temperature in 120 hours

Table 3.3 Weather features for the baseline models.

Feature	Description
Natural gas price	Price in EUR/MWh
Change natural gas %	Change in natural gas price over the last 24 hours in percent
Change natural gas mean	The mean of the moving average of natural gas price changes over the last 7 days
Change natural gas std	The standard deviation of the moving average of natural gas price changes over the last 7 days
Brent change %	Change in Brent oil price over the last 24 hours
WTI change %	Change in WTI oil price over the last 24 hours

Table 3.4 Commodity features for the baseline models.

Feature	Description
Hour	The hour of the day (0,1,....23)
Dayofweek	The day of the week (Monday, Tuesday, ...)
Cheapest hour percentage	The chance of the current hour being the cheapest for the entire day
Exp hour percentage	The chance of the current hour being the most expensive hour for the day
Price mean 6 days	The mean of the electricity price the last 6 days
Price std 6 days	The standard deviation of the electricity price the last 6 days
Slope 3 days mean	The mean of the moving average of the gradient of the electricity price over the last 3 days
Slope day before	The gradient of the electricity price of the day before the current day
Load Germany	The demand for electricity in Germany
Supply Germany	The supply of electricity in Germany
Supply SE4	Total MW electricity produced to the grid in SE4
Load SE4	The demand of electricity in SE4
Net MW scheduled	Net MW of electricity exported and imported to SE4
Price last-24	Price 24 hours ago
⋮	⋮
Price last-3	Price 3 hours ago
Price last-2	Price 2 hours ago
Price last-1	Price 1 hour ago
Price	The current electricity price
Price next+1	Price at 00.00 the next day
Price next+2	Price at 01.00 the next day
Price next+3	Price at 02.00 the next day
⋮	⋮
Price next+23	Price at 23.00 the next day

Table 3.5 Miscellaneous features for the baseline models.

3.4.3 Forecasting prices with baseline models

The baseline models are trained on data from 2015-01-07 22:00:00 to 2018-03-12 12:00:00 which is 80% of the entire dataset, conforming to the common 80-20 ratio described in [*Training and Test Sets: Splitting*

Data n.d.] The MCRMSE for the test dataset is 12.942 EUR/MWh, which is the average value for RMSE over the entire dataset.

The figure in 3.9 shows the average RMSE error for each forecasted hour. It is evident that the 24th hour is easily the best predicted hour. The RMSE then climbs with different rates of speed upwards, having some brief declines before resuming the uptrend. There is a spike at the 110th hour, making it the hardest hour to forecast. Therefore, it will be used to graph how the predictions compare to the true values for electricity prices. Why the worst forecasted hour is chosen to graph the forecasting ability is due to every other hour forecasted is guaranteed to the better, showing the worst result provides a worst case and every other case is better.

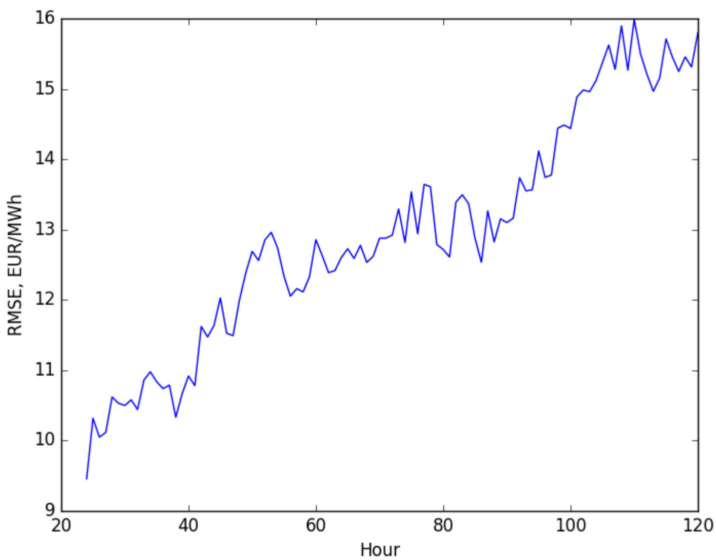


Figure 3.9 The average RMSE error per hour forecasted. The highest average RMSE is found at the 110th hour.

The RMSE for the test dataset with respect to the 110th hour in advance is 15.987 EUR/MWh, and the values for the 110th hour are depicted in 3.10 for both the test set and the predicted set.

Figure 3.10 How the baseline models perform on the test data. In the graph, only the 110-hour projections are shown. The red line is the continuously projected electricity price 110 hours out, and the yellow line is what the electricity prices turned out to be.

Graphically, it is hard to tell how well the model performs by looking at Figure 3.10 since it spans a longer time horizon. Therefore, a sample from the test dataset has been taken in order to show how the model performs in a shorter time frame. The values between 2018-04-05 00:00:00+00:00 and 2018-04-09 00:00:00+00:00 have been picked as a sample. In Figure 3.11, the 110th hour in advance for the predicted dataset and the test dataset have been plotted. The RMSE in this case is 4.371 EUR/MWh.

Now, in order to see how the model performs by forecasting 24 hours to 120 hours in advance, which is the *entire idea* with the model, one hour from the test dataset is picked. The hour is 2018-06-011 00:00:00+00:00. The result is a forecast starting from 2018-06-12 00:00:00+00:00 until 2018-06-16 00:00:00+00:00. The forecast is shown in Figure 3.12. The RMSE value, in this case, is 9.246 EUR/MWh.

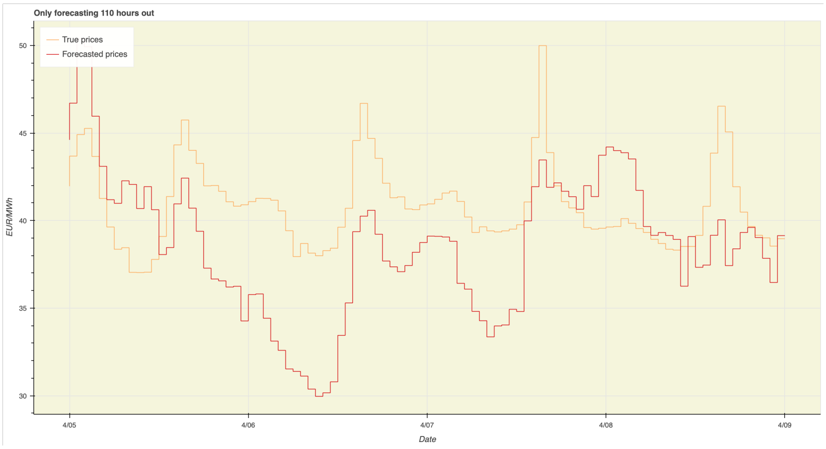


Figure 3.11 Repeatedly projecting the price in 110 hours, from 2018-04-05 to 2018-04-09. The yellow line is the prices that later turned out to be true and the red line is the projected prices.

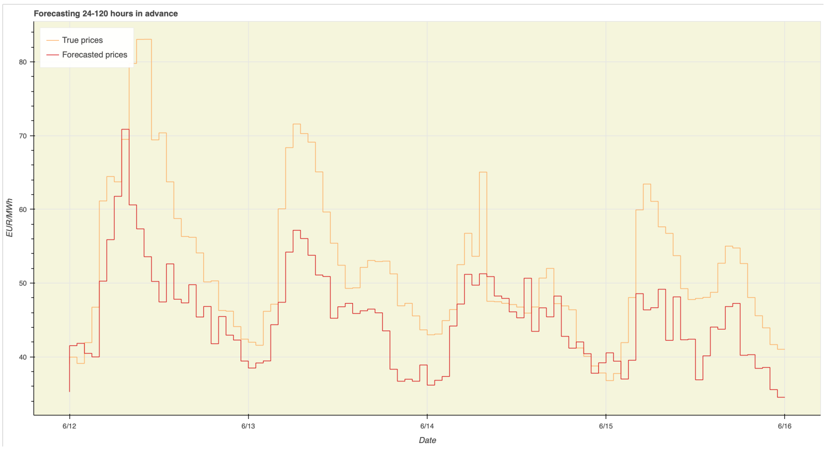


Figure 3.12 The projected price for the coming 5 days when the forecast is executed on 2018-06-11 00:00:00+00:00. The yellow line is what the price turned out to be. The red line is the projected coming prices.

3.5 Primary models

The baseline models, trained on data from 2015-2018, showed to be not too bad at forecasting electricity prices. With the observation of the result from the baseline, it is possible to deduce that applying the same methodology to the primary models, being trained on data from 2020-2023, should hopefully yield a reasonable result, given the circumstances. Since the baseline models showed somewhat of a good ability to forecast, the hope is that the RMSE for the primary models is to a majority attributed to the unstable price action, and is not attributed to the 'imperfections' in the model. Naturally, the RMSE will be several times higher for the primary models compared to the baseline models.

3.5.1 Preprocessing & feature engineering

The baseline and the primary models share a number of identical features. The identical features already explained in 3.4.2 are therefore not brought up here as they operate with the same intentions as described previously.

3.5.1.1 Weather. The weather features are almost the same as in the baseline model. For the weather features, instead of starting 1 hour out, 2 hours out, and up to 120 hours out, it instead starts at 24 hours out, all the way up to 120 hours. The table of features is found in Table 3.6.

3.5.1.2 Commodities. For the table of commodities, 3.7, the table is extended with two additional features to the already existing features from 3.4. The 'WTI USD' feature is the price in U.S. dollars for a barrel of West Texas Intermediate (WTI) oil. The 'Brent USD' feature is the price in U.S. dollars for a barrel of Brent oil. Oil prices have a greater impact on the electricity price in 2020-2023 than in 2015-2018 based on how the primary models respond to the data in comparison to how the baseline models respond to the data.

3.5.1.3 Miscellaneous. The model has of course been fed the feature 'hour' and 'dayofweek', which enables the model to find relationships between the price and the hour of the day, and the day of the week. As presented in Section 3.4.2, the demand for electricity depends on the hour of the day, and so does the electricity price. Below are the same statistics presented but for 2020-2023.

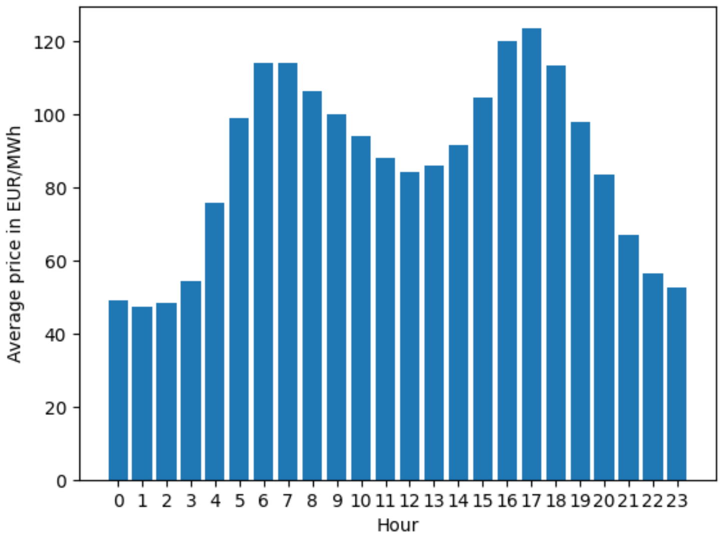


Figure 3.13 Average electricity price in EUR/MWh between 2020-01-01 00:00:00 and 2023-01-24 23:00:00. The price usually peaks in the morning at 06.00/07.00 and in the afternoon at 17.00.

The primary models also take a look at the distribution of the probability of each hour being the cheapest and most expensive during the day, respectively. Figure 3.15 shows the probability of each hour being the cheapest over the day, while Figure 3.16 shows the probability of each hour being the most expensive over the day.

The feature 'Supply Germany slope 7-day mean" calculate the mean of the last seven days' worth of values in terms of the available supply of electricity in Germany.

There are numerous features dependent on the supply and demand

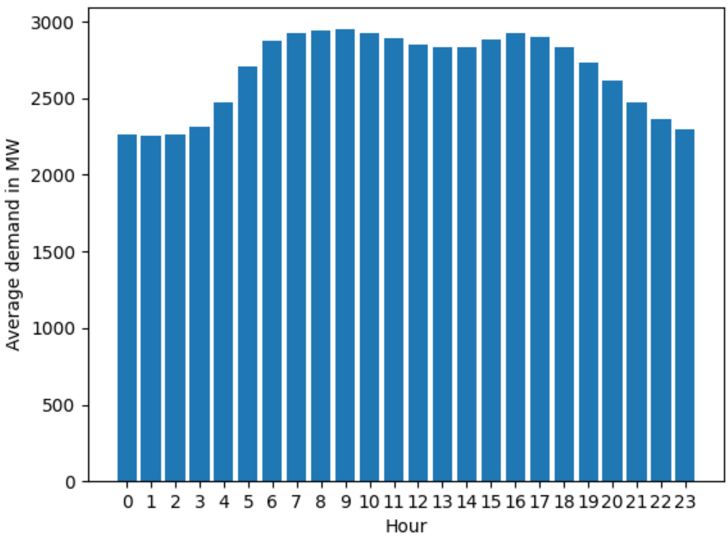


Figure 3.14 Average demand for electricity in MW between 2019-12-31 23:00:00+00:00 and 2023-01-19 23:00:00+00:00. During the day, the demand rises, and during the night, the demand diminishes.

of electricity in Germany. In addition to 'Supply Germany slope 7-day mean', there exists the features 'Supply Germany 5-day mean', 'Supply Germany 1-day mean', 'Supply Germany', 'Load Germany 10-day median', 'Load Germany 10-day mean', and 'Load Germany s-7 1 day mean'. The feature 'Supply Germany 5-day mean' takes the mean of the supply over the last 5 days. 'Supply Germany 1-day mean' does the same thing over a day. 'Supply Germany' is the last recorded electricity supply in Germany. Feature 'Load Germany s-7 1-day mean' shifts all values backward by 168 steps and then takes the mean over a 24-hour period. Feature 'Load Germany 10-day median' calculates the median of all the demand for electricity in Germany over the last 10 days. Feature 'Load Germany 10-day mean' calculates the mean of the demand for electricity in Germany over the last 10 days.

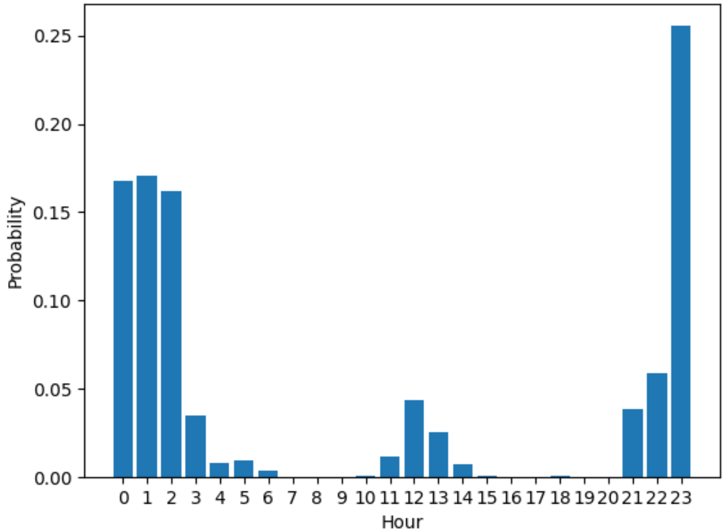


Figure 3.15 Probability of an hour being the cheapest during the day, between 2019-12-31 and 2023-01-19. 23.00 is statistically most often the cheapest hour of the day.

The features 'Supply SE4', 'Load SE4', 'Net MW scheduled', 'Price', 'Price last-24, Price last-23, ..., Price last-1', 'Price next+1, Price next+2, ..., Price next+23' is exactly the same as in Table 3.5.

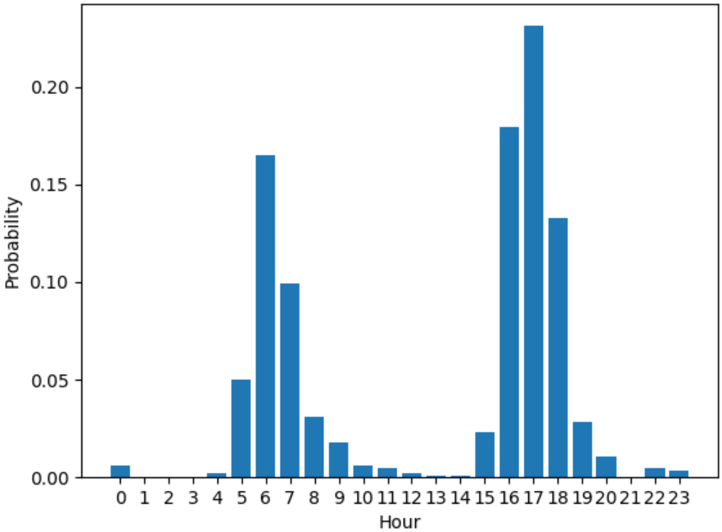


Figure 3.16 Probability of each hour being the most expensive during the day, between 2019-12-31 and 2023-01-19. 17.00 is statistically the most expensive hour of the day.

Feature	Description
Windspeed	The current hourly windspeed in m/s
Windspeed t+24	Windspeed in 24 hours
Windspeed t+25	Windspeed in 25 hours
Windspeed t+26	Windspeed in 26 hours
⋮	⋮
Windspeed t+120	Windspeed in 120 hours
Precip	The current hourly precipitation in mm
Precip t+24	Precipitation in 24 hours
Precip t+25	Precipitation in 25 hours
Precip t+26	Precipitation in 26 hours
⋮	⋮
Precip t+120	Precipitation in 120 hours
Wind gusts	The current hourly wind gusts in m/s
Wind gusts t+24	Wind gusts in 24 hours
Wind gusts t+25	Wind gusts in 25 hours
Wind gusts t+26	Wind gusts in 26 hours
⋮	⋮
Wind gusts t+120	Wind gusts in 120 hours
Weathercode	The current hourly weather code
Weathercode t+24	Weather code in 24 hours
Weathercode t+25	Weather code in 25 hours
Weathercode t+26	Weather code in 26 hours
⋮	⋮
Weathercode t+120	Weather code in 120 hours
Temp	The current temperature in celsius
Temp t+24	The temperature in 24 hours
Temp t+25	The temperature in 25 hours
Temp t+26	The temperature in 26 hours
⋮	⋮
Temp t+120	The temperature in 120 hours

Table 3.6 The weather features for the primary models.

Feature	Description
Natural gas price	Price in EUR/MWh
Change natural gas %	Change in natural gas price over the last 24 hours in percent
Change natural gas mean	The mean of the moving average of natural gas price changes over the last 7 days
Change natural gas std	The standard deviation of the moving average of natural gas price changes over the last 7 days
WTI USD	The price of WTI in USD
Brent USD	The price of Brent in USD
Brent change %	Change in Brent oil price over the last 24 hours
WTI change %	Change in WTI oil price over the last 24 hours

Table 3.7 Commodity features

Feature	Description
Hour	The hour of the day (0,1,....23)
Dayofweek	The day of the week (Monday, Tuesday, ...)
Cheapest hour percentage	The chance of the current hour being the cheapest for the entire day
Exp hour percentage	The chance of the current hour being the most expensive hour for the day
Supply Germany slope 7-day mean	The 7-day mean of the slope of the supply of electricity in Germany
Supply Germany 5-day mean	The 5-day mean of the electricity supply in Germany
Supply Germany 1-day mean	The 1-day mean of the electricity supply in Germany
Supply Germany	The total electricity supply in Germany
Load Germany s-7 1-day mean	The demand for electricity in Germany, shifted by 7 days, and the rolling mean over 1 day
Load Germany 10-day median	The demand for electricity in Germany, rolling median over 10 days
Load Germany 10-day mean	The demand for electricity in Germany, rolling mean over 10 days
Supply SE4	Total MW electricity produced to the grid in SE4
Load SE4	The demand of electricity in SE4
Net MW scheduled	Net MW of electricity exported and imported to SE4
Price last-24	Price 24 hours ago
⋮	⋮
Price last-3	Price 3 hours ago
Price last-2	Price 2 hours ago
Price last-1	Price 1 hour ago
Price	The current electricity price
Price next+1	Price at 00.00 the next day
Price next+2	Price at 01.00 the next day
Price next+3	Price at 02.00 the next day
⋮	⋮
Price next+23	Price at 23.00 the next day

Table 3.8 The miscellaneous features for the primary models.

3.5.2 Forecasting prices with primary models

The primary models are trained on data from 2020-01-06 00:00:00 to 2022-06-08 01:00:00. The training data represents 80% of the dataset. The average RMSE metric for each forecasted hour is displayed in 3.17. The average RMSE starts with a short-lived spike upwards that then transpires into a spike downwards, finding its bottom at the 28th hour. The lowest average RMSE is found at hour 28, meaning that it is the easiest hour to project the price for. After the 28th hour, the average RMSE rises steadily, with minor downturns before it swings upwards again, reaching a top at the 120th hour. The MCRMSE for the test set here is 130.393 EUR/MWh, quite the difference from the baseline models.

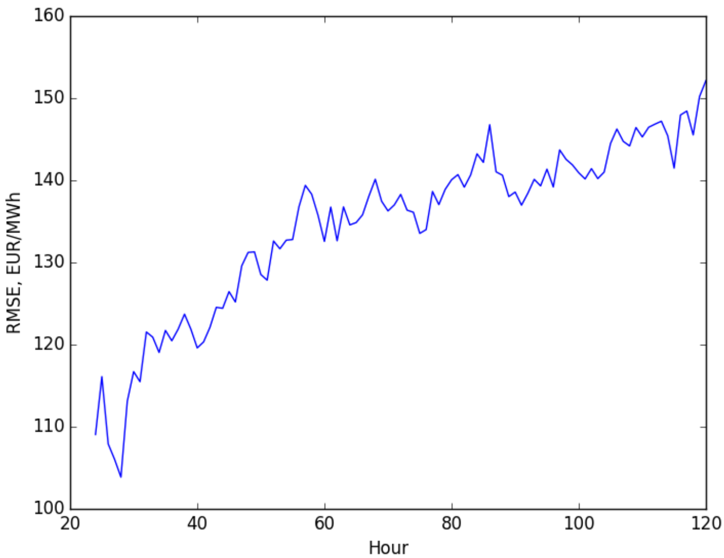
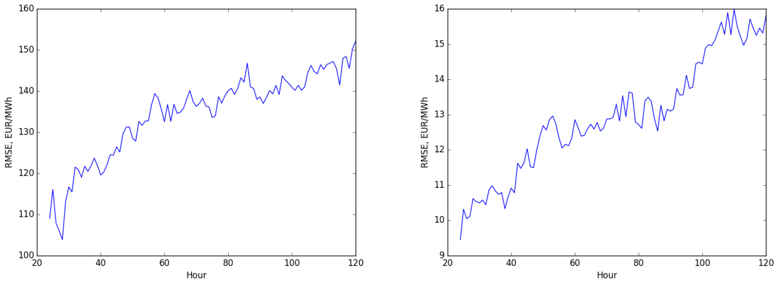


Figure 3.17 Average RMSE for each hour forecasted

Illustrating the average RMSE for the primary models and the baseline models side by side, the order of magnitude in difference quickly becomes apparent. The graphs also do have diverging patterns of how the RMSE increases.



(a) Average RMSE for the primary models. The highest average RMSE is found at the 120th hour.

(b) Average RMSE for the baseline models. The highest average RMSE is found at the 110th hour.

Figure 3.18 Difference in the graph over the average RMSE error. The primary models portray a slower incline in the RMSE while having a much higher starting point. For the primary models, the average RMSE starts out at 110, while it starts out at 9.5 for the baseline models.

Since the 120th hour is the forecasted hour with the worse score according to the evaluation metric, it is used to show how the test set and the predicted set differ. See Figure 3.19. The RMSE metric for the 120th hour is 152.136 EUR/MWh.

As the image portrays, the red (forecasted prices) line does not always follow along with the spikes in the yellow (true prices) line, and they diverge at many locations. Exactly as in the baseline models, a shorter time period has been picked out to showcase how the forecasted and true electricity prices differ in a more detailed view. The RMSE on the test set for the 120th-hour forecasting is 34.981 EUR/MWh.

Looking at how it performs in forecasting the next 24 hours to 120 hours in the future, it can be seen in Figure 3.21. The forecast is done on 2023-01-05 00:00:00 and therefore forecasts the electricity price for 2023-01-06 00:00:00 - 2023-01-10 00:00:00

The RMSE for the 5-day forecast in Figure 3.21 is 41.365. The fore-

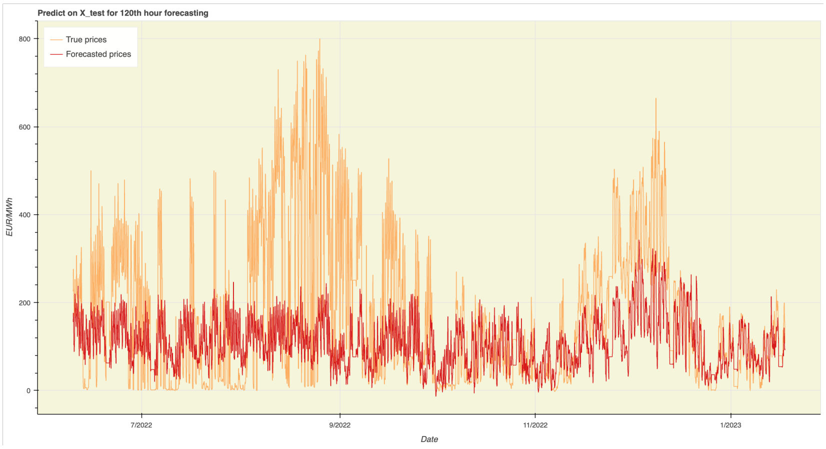


Figure 3.19 How the primary models perform on the test data. In the graph, only the 120-hour projections are shown. The red line is the continuously projected electricity price 120 hours out, and the yellow line is what the electricity prices turned out to be.

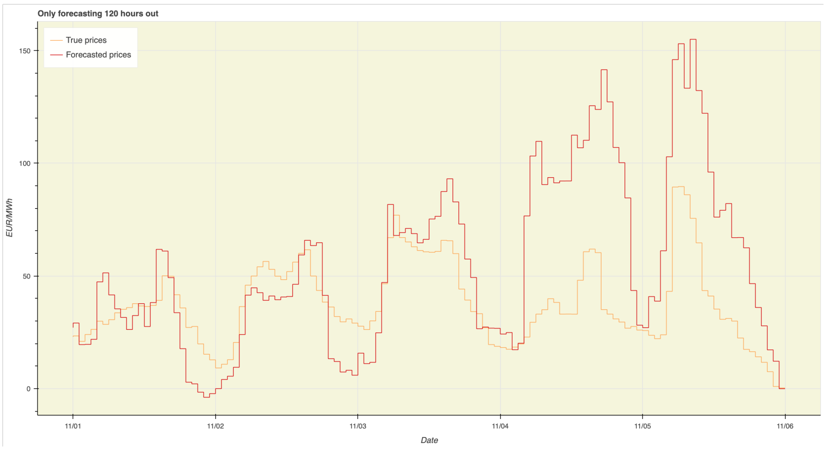


Figure 3.20 Repeatedly projecting the price in 120 hours, from 2022-11-01 to 2022-11-06. The red line is the projected prices and the yellow line is the prices that later were determined.

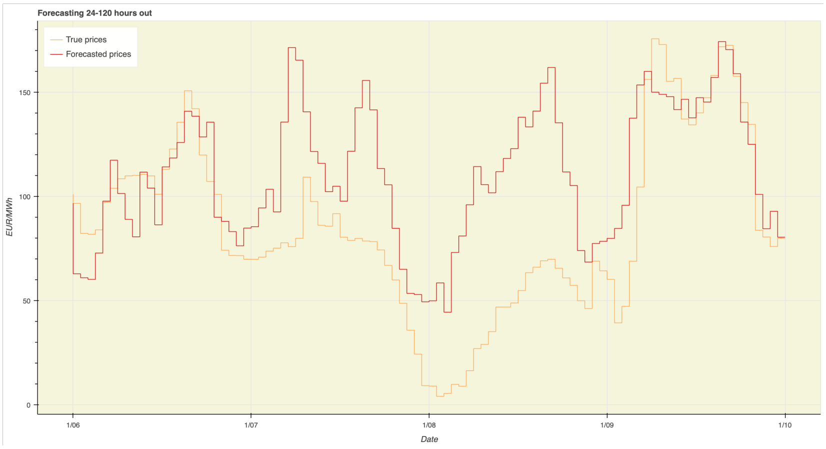


Figure 3.21 The projected price for the coming 5 days when the forecast is executed on 2023-01-05 00:00:00+00:00. The red line is the projected coming prices. The yellow line is what the price turned out to be.

cast starts out somewhat reasonable, for the most part tracking the true price up until 7/01/23. Then, the true prices diverge and the forecast does not follow up on the pattern. The forecast expects the price to stay bound in a range of 40-180 EUR/MWh over the entire forecasting period while the actual price dips down to about 10 EUR/MWh at one point. One noteworthy fact is that the forecasted electricity price sets a low point for the interval, remarkably close to the point in time where the actual electricity price also sets the lowest price for the interval. After the forecast diverges from the real electricity price between 7/01/23 to 9/01/23, the forecast becomes much more reliable again, staying very close to the true price up until the forecast ends on 10/01/23. Naturally, depending on which time frame is selected, the result differs. Some time frames generate good-looking results while some generate worse results.

4

Optimization of charging protocols

In this chapter, the optimization of different charging protocols is presented. There is a total of seven unique charging protocols that are tested for their cost-efficiency. Cost efficiency is synonymous with the ability of a charging protocol to produce a charging pattern and measure its cost against other charging patterns generated by other charging protocols. If charging protocol a produces a charging pattern that is cheaper than a charging pattern produced by charging protocol b with the same information available, a is said to be more cost-efficient than b. The main question that is to be answered is if the first charging protocol, which utilizes the forecasted prices for the future, is able to be more or less cost-efficient than the other charging protocols.

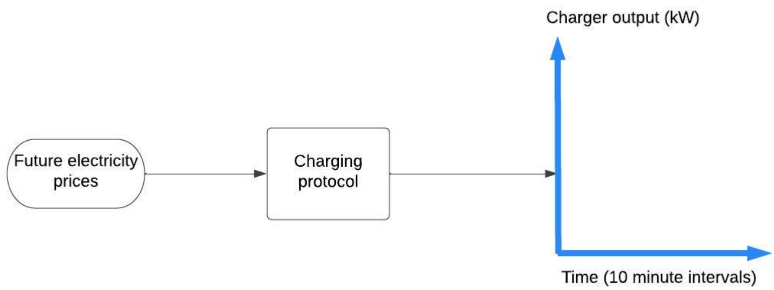


Figure 4.1 The process of producing charging patterns. The charging protocol takes future electricity prices and the output from each protocol is a charging pattern that the charger will adhere to.

All of the charging protocols have their origin in the same problem definition. The problem definition is based on the "Optimal Dispatch of Power Generators: Problem-Based" [*Optimal Dispatch of Power Generators: Problem-Based* n.d.] In the optimal dispatch problem, the optimization is of two gas-fired electric generators to generate the maximum profit, the revenue minus the operational cost. Since the electricity price varies over the day, there are times during the day when the profit is higher to operate the gas-fired generators, and this is something that can be taken advantage of. How the optimal dispatch problem relates to the optimization in this thesis is its optimization of the sum of a variable, dependent on another variable that determines how much energy is consumed. In this thesis, the variable being summed is either the cost or the SoC over the entire period. The electricity cost or SoC is dependent on how much the charger decides to consume in terms of electric energy, instead of the energy from fuel sources. The fuel cost of the generators is swapped out for the electricity cost at the same time as the gas-powered generators are swapped out by an electrical charging device that is used to charge an electric vehicle.

Two of the seven charging protocols are strictly focused on the cost, where the goal is to minimize the cost.

In the five other charging protocols, the focus is heavily on the summation of the SoC over the entire time period, being either minimized or maximized. For these five charging protocols, no definition of any variable changes from the other two protocols focused on cost, but the optimization objective is now dependent on the SoC rather than the cost, except for one protocol which incorporates both the cost and the SoC.

The motivation behind the case of maximizing the summation of SoC is to always have the car with as full of a battery as possible, being charged as fast as possible, due to the primary two circumstances:

1. The car has to leave before it is planned to leave.
2. A drive travels much longer than expected.

In the first case, if the car has to leave before it is planned to leave, the car will be at the maximum SoC it had time to reach before the car leaves. Normally, a person follows a pre-defined schedule and knows when to use the car. However, there are unplanned events happening every now and then that drastically change the daily plan. If that happens and the car needs to be used earlier than planned, it is best if the car was charging at maximum power, as it means the car will be charged to the highest level it had time to reach. If the charger only operated at half the maximum power, the SoC when leaving the garage with the car may not be sufficient. The second case also involves unplanned events, where the car needs to travel further than planned. In that case, having as much charge as possible when leaving the garage is preferred, as the extra distance to be traveled may be hard to determine beforehand.

The motivation behind the case of minimizing the summation of SoC is due to:

1. Battery longevity. The state of charge status and the temperature of the battery both have an impact on the life of the battery [Tamura and Kikuchi, 2018]. Longer floating times at high levels of SoC have been shown to have a negative impact on the life of the battery [Mussa et al., 2017].

2. Less electricity required. Having the SoC at the bare minimum reduces both cost and the impact on the electrical grid.

3. Charging efficiency. The charge acceptance of a battery is higher at low levels of SoC. Charge acceptance reduces as the SoC of the battery increases [Roy et al., 2022].

Having an efficient battery with a long lifetime is an optimal scenario, which is why the minimization of the SoC is experimented with.

4.1 Charging scheduler

To experiment with different charging protocols, an optimization instrument named "Charging scheduler" is introduced. It defines a number of constraints that apply to the charging regimes and provides a method of calculating the cost of each charging protocol. With the instrument, a schedule can be built that has a focus to either reduce overall cost or optimize the SoC of the car, while serving the needs of the car. The needs of the car may be that it will travel x kilometers between 14:00 and 16:00.

4.1.1 Optimization problem

The total cost of charging in the time period T is defined as

$$C_T = \sum_{t=0}^{T} c(t) \left(\sum_{i=0}^{10} \frac{i}{10} s_i(t) \right) \tag{4.1}$$

where $c(t)$ is a piecewise function representing the forecasted electricity price in EUR/kWh at time t. The function $s_i(t)$ is defined as

$$s_i(t) = \begin{cases} 1, \text{if regime } s_i \text{ is enabled at time } t \\ 0, \text{if regime } s_i \text{ is disabled at time } t \end{cases} \tag{4.2}$$

The charger has 11 regimes, the first regime s_0 represents the charger being non-operational. Each succeeding regime represents an increase of 1 kW in output, reaching a maximum output of 10 kW for regime s_{10}.

The regimes can be represented as a vector

$$\mathbf{s} = [s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}] \tag{4.3}$$

where $s_i(t)$ outputs a 0 if regime i is disabled at time t, and a 1 if regime i is enabled at time t. The charger cannot be in two regimes simultaneously, which can be written as

$$s_\sigma(t) = \sum_{i=0}^{10} s_i(t) \leq 1 \tag{4.4}$$

at any time t. The subscript σ stands for the sum of all charging regimes s_i at time t. $s_\sigma(t)$ would naturally be equal to 0 when no regime is enabled.

4.1.2 Constraints

To define the constraints of the optimization problem, the theory of "State of Charge" is introduced. State of Charge is defined as

$$SoC(t) = \frac{Q(t)}{Q_{nominal}} \tag{4.5}$$

where $Q(t)$ is the current charge of the battery and $Q_{nominal}$ is the maximum charge of the battery [Tribioli and Bella, 2022]. In this paper, Q_{nominal} is set to $150kWh$. The unit of the state of charge is percent. To preserve the lifetime of the battery, it is critical to ensure that the battery does not overcharge or over-discharge. The term $\text{SoC}_{S^{\cdot}min}$ is introduced to represent the minimum state of charge allowed for the battery. A second term $\text{SoC}_{S^{\cdot}max}$ is introduced to represent the maximum state of charge allowed for the battery. The constraint on the state of charge is therefore [Cheikh-Mohamad et al., 2021]

$$\text{SoC}_{S^{\cdot}min} \leq \text{SoC}(t) \leq \text{SoC}_{S^{\cdot}max} \tag{4.6}$$

for all $t \in T$. $\text{SoC}_{S^{\cdot}min}$ is set to $\frac{10}{150}\%$ and $\text{SoC}_{S^{\cdot}max}$ is set to $\frac{115}{150}\%$.

When looking at the j-th interval, interval j has a state of charge SoC_j where the next state of charge SoC_{j+1} is dependent on the previous SoC, SoC_j, such that

$$\text{SoC}_{(j+1)} = \text{SoC}_{(j)} - \frac{P_{d(j)}}{Q_{nominal}} \tag{4.7}$$

where $P_{d(j)}$ is the change in battery charge depending on if the car is moving, standing still, or charging during segment j.

The expression for $\text{SoC}_{(j)}$ at the j-th interval can be expressed as the initial state minus all the draws of the battery, plus all the charges to the battery. $P_{d(i)} < 0$ if the car is being charged and $P_{d(i)} > 0$ if energy is being consumed from the battery. For $P_{d(i)} = 0$, the battery is neither being charged nor being expended of energy.

$$\text{SoC}_{(j)} = \text{SoC}_{(0)} - \sum_{i=1}^{j} \frac{P_{d(i)}}{Q_{nominal}} \tag{4.8}$$

$\text{SoC}_{(0)}$ is the initial state of charge, set to 20%.

In this optimization problem, each interval is 10 minutes, 6 intervals per hour. The expressions for SoC are derived from [Kusakana, 2015], with minor changes in definitions and symbols.

With an expression for $\mathrm{SoC}_{(j)}$, it can now be expressed with $\mathrm{SoC}_{\mathrm{min}}$ and $\mathrm{SoC}_{\mathrm{max}}$.

At any interval j, the following condition must hold, or else no optimization is feasible.

$$\mathrm{SoC}_{S\cdot min} \leq \mathrm{SoC}_{(j)} \leq \mathrm{SoC}_{S\cdot max} \Longleftrightarrow \mathrm{SoC}_{S\cdot min} \leq \mathrm{SoC}_{(0)} - \sum_{i=0}^{j} \frac{P_{d(i)}}{Q_{nominal}} \leq \mathrm{SoC}_{S\cdot max} \tag{4.9}$$

While the car is not at rest, it cannot be charged. Let $r(t)$ be a function defined as

$$r(t) = \begin{cases} 1, \text{if the car is at rest, ready to be charged} \\ 0, \text{if the car is not at rest, not ready to be charged} \end{cases} \tag{4.10}$$

$r(t)$ places a constraint on $s_\sigma(t)$ such that $s_\sigma(t)$ can only be equal to 1 while $r(t)$ is equal to 1. $s_\sigma(t)$ can be equal to 0 independent of $r(t)$.

The energy consumption of the battery is set at $0.2kW/km$. The number is taken from an average energy consumption from multiple models of electric vehicles [*Energy consumption of full electric vehicles* n.d.] The user of the electric vehicle will have to input approximately which hours the car will be driven, and the estimated distance. With the average energy consumption per kilometer, $P_{d_{(j)}}$ can be substituted at every interval j.

4.2 Implementation in Matlab

The implementation in Matlab reflects the information in Section 4.1. As previously mentioned, two different solvers, 'intlinprog' and 'gamultiobj' is utilized to solve the optimization problems.

4.2.1 Variables

There is a variable, denoted y, that has 11 different regimes. Each regime corresponds to each charging regime, previously discussed in Section 4.1.2. Each respective regime has a lower bound of 0 and an upper bound of 1, being either off or on. Each hour is divided into 6 segments, 10 minutes each. There is a total of 121 hours in the time frame, meaning that there is a total of 726 segments of 10 minutes each. The variable 'y' is a matrix of 726 rows and 11 columns. Row j represents the j-th time segment and column i represents s_i, returning a 0 if s_i is not enabled, and a 1 if s_i is enabled. Indexing the matrix by j-th row and the i-th column would return the status of s_i in the j-th segment.

$$\mathrm{y} = \begin{bmatrix} s_0^1 & s_1^1 & s_2^1 & s_3^1 & s_4^1 & s_5^1 & s_6^1 & s_7^1 & s_8^1 & s_9^1 & s_{10}^1 \\ s_0^2 & s_1^2 & s_2^2 & s_3^2 & s_4^2 & s_5^2 & s_6^2 & s_7^2 & s_8^2 & s_9^2 & s_{10}^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ s_0^{726} & s_1^{726} & s_2^{726} & s_3^{726} & s_4^{726} & s_5^{726} & s_6^{726} & s_7^{726} & s_8^{726} & s_9^{726} & s_{10}^{726} \end{bmatrix} \tag{4.11}$$

A matrix yElectricity is introduced, keeping track of the possible values each regime assumes during all 726 segments. The yElectricity matrix naturally also has 726 rows and 11 columns, as it is the charging outputs for each state during each segment in time. Row j represents the j-th time segment and column i represents s_i, the wattage output. Indexing the matrix by j-th row and the i-th column would return the wattage output for s_i^j. Each wattage output is divided by 6 since s_i has an output of i kW and each hour is divided into 6 segments.

$$\mathrm{yElectricity} = \left.\begin{bmatrix} 0 & \frac{1}{6} & \frac{2}{6} & \frac{3}{6} & \frac{4}{6} & \frac{5}{6} & \frac{6}{6} & \frac{7}{6} & \frac{8}{6} & \frac{9}{6} & \frac{10}{6} \\ 0 & \frac{1}{6} & \frac{2}{6} & \frac{3}{6} & \frac{4}{6} & \frac{5}{6} & \frac{6}{6} & \frac{7}{6} & \frac{8}{6} & \frac{9}{6} & \frac{10}{6} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \frac{1}{6} & \frac{2}{6} & \frac{3}{6} & \frac{4}{6} & \frac{5}{6} & \frac{6}{6} & \frac{7}{6} & \frac{8}{6} & \frac{9}{6} & \frac{10}{6} \end{bmatrix}\right\} \text{726 rows} \tag{4.12}$$

The cost c_i is taken from a vector 'yCost' that contains the hourly prices. In one case, 'yCost' contains the forecasted prices plus the prices for the initial 24 hours that are not forecasted. In that case, c_1 to c_{24} are the initial hourly prices, c_{25} to c_{121} are the forecasted hourly prices. In the other six cases, c_1 to c_{121} are the true electricity prices. To divide each hour into six segments, each element is repeated six times at every position. If the initial vector was described as

$$\text{yCost} = [c_1, c_2, c_3, ..., c_{121}] \tag{4.13}$$

After repeating each element six times, the vector consists of 726 values and is now described as

$$\text{yCost} = [\underbrace{c_1, ..., c_1}_{\text{6 elements}}, \underbrace{c_2, ..., c_2}_{\text{6 elements}}, ..., \underbrace{c_{121}, ..., c_{121}}_{\text{6 elements}}] = [c_1, c_2, c_3, ..., c_{726}] \tag{4.14}$$

z is the Hadamard product, otherwise called the element-wise multiplication [Horn, 1990], of *yElectricity* and y.

$$z = yElectricity \circ y = \begin{bmatrix} \frac{0}{6}s_0^1 & \frac{1}{6}s_1^1 & \frac{2}{6}s_2^1 & \cdots & \frac{10}{6}s_{10}^1 \\ \frac{0}{6}s_0^2 & \frac{1}{6}s_1^2 & \frac{2}{6}s_2^2 & \cdots & \frac{10}{6}s_{10}^2 \\ \frac{0}{6}s_0^3 & \frac{1}{6}s_1^3 & \frac{2}{6}s_2^3 & \cdots & \frac{10}{6}s_{10}^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{0}{6}s_0^{726} & \frac{1}{6}s_1^{726} & \frac{2}{6}s_2^{726} & \cdots & \frac{10}{6}s_{10}^{726} \end{bmatrix} \tag{4.15}$$

where $z(j,i) = \frac{i}{6}s_i(j)$

A final variable is introduced to have a final expression for the solver when solving the optimization problem with regard to cost. The variable to be optimized is named 'electricityCost' which is declared as an 'optimexpr' in Matlab with the same size as the previously declared variable 'y'. As the name suggests, it is an expression to be optimized. The size is therefore 726 rows with 11 columns. Index j of returns the j-th segment, and the i-th index of the j-th segment is the electricity price times the value of s_i at the j-th segment. The value of s_i at the j-th segment is of course gathered from 'y' and the electricity consumed during the j-th is gathered from 'yElectricity'. Let c_j be the cost at the j-th time segment.

$$\text{electricityCost} = \begin{bmatrix} c_1 z(1,0) & c_1 z(1,1) & c_1 z(1,2) & \cdots & c_1 z(1,10) \\ c_2 z(2,0) & c_2 z(2,1) & c_2 z(2,2) & \cdots & c_2 z(2,10) \\ c_3 z(3,0) & c_3 z(3,1) & c_3 z(3,2) & \cdots & c_3 z(3,10) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{726} z(726,0) & c_{726} z(726,1) & c_{726} z(726,2) & \cdots & c_{726} z(726,10) \end{bmatrix} \tag{4.16}$$

The expression in Matlab for the cost of electricity during the 726 segments of time is as follows:

$$\text{totalCostElectricity} = \text{sum}(\text{sum}(\text{electricityCost})) \tag{4.17}$$

The first sum of electricityCost returns the matrix

$$sum(\text{electricityCost}) = \begin{bmatrix} c_1 z(1,0) + c_1 z(1,1) + \cdots + c_1 z(1,10) \\ c_2 z(2,0) + c_2 z(2,1) + \cdots + c_2 z(2,10) \\ c_3 z(3,0) + c_3 z(3,1) + \cdots + c_3 z(3,10) \\ c_{726} z(726,0) + c_{726} z(726,1) + \cdots + c_{726} z(726,10) \end{bmatrix} \tag{4.18}$$

and the second sum returns the sum of all the rows. The sum of all rows be rewritten by moving the common factors of terms such that it now looks like

$$\begin{aligned} \text{sum}(\text{sum}(\text{electricityCost})) = c_1(z(1,0) + \cdots + z(1,10)) + \cdots \\ + c_{726}(z(726,0) + \cdots + z(726,10)) \end{aligned} \tag{4.19}$$

and the equation can, in turn, be rewritten with summation and $s_i(t)$ instead of z since $z(t,i) = \frac{i}{6} s_i(t)$

$$\text{sum}(\text{sum}(\text{electricityCost})) = c_1(\sum_{i=0}^{10} \frac{i}{6} s_i(1)) + \cdots + c_{726}(\sum_{i=0}^{10} \frac{i}{6} s_i(726)) \tag{4.20}$$

and since c_t is the cost at the i-th segment, each c_i can be replaced by the cost function $c(t)$. The result is

$$\text{sum}(\text{sum}(\text{electricityCost})) = \sum_{t=0}^{726} c(t) \left(\sum_{i=0}^{10} \frac{i}{6} s_i(t) \right) \tag{4.21}$$

and the original equation from 4.1 with $T = 726$ becomes apparent. The SoC is defined as

$$\text{SoC}_{(j)} = SoC_{(0)} + \frac{\overbrace{\text{cumsum}(charge) - \text{cumsum}(carCon)}^{\text{current charge expressed in SoC excluding } SoC_{(0)}}}{Q_{\text{nominal}}} \tag{4.22}$$

where 'cumsum' is the cumulative summation from the start, up to the current point in time. The definition in 4.28 is another way to express 4.9. 'charge' contains the charge for each segment in time and 'carCon' contains the energy consumption for each segment in time. If during the j-th segment, the car drives 20km, the expenditure of energy is $0.2 \cdot 20 = 4$ kW, hence $\text{carCon}_{(j)} = 4$. $\text{charge}_{(j)}$ is dependent on which regime the charger is in during the j-th segment. This is the SoC term that five of the charging protocols attempt to minimize or maximize the summation of. The sum of the SoC term in Matlab is given by $sum(SoC)$.

4.2.2 Global constraints in Matlab

The constraints in this section are applied to all optimization problems. In some optimization problems, there are additional constraints to the constraints listed in this section.

With the objective functions 'totalCostElectricity' and the summation of SoC, $\text{sum}(SoC)$, the constraints to the objective functions can be set.

The first constraint restricts the charger from being in multiple regimes at a time, which is infeasible. The Equation 4.4 describes the property of this constraint at a specific point in time. To generalize the constraint over the entire variable 'y', it can be expressed as

$$\sum_{i=0}^{10} s_i^j \leq 1, \text{ for } j = 1, 2, ..., 726. \tag{4.23}$$

The constraint is called 'powercons' in the Matlab implementation.

The second constraint on the objective function restricts the charger from being on whenever the car is driving. Being on of course means that the charger is on and charging the vehicle. A vector 'carDriving' is introduced with 0 whenever the car is not driving and 1 when the car is driving. The vector consists of 726 elements.

$$\text{carDriving} = [d^1, d^2, d^3, ..., d^{726}] \tag{4.24}$$

The constraint is defined as

$$d^j + \sum_{i=1}^{10} s_i^j \leq 1, \text{ for } j = 1, 2, ..., 726. \tag{4.25}$$

The constraint is called 'noChargeWhenNotPossible' in the Matlab implementation.

A constraint that builds upon the previous constraint is to introduce the vector 'r' discussed in 4.1.2. r is a vector with the same size as 'carDriving' and is 0 if the car is not at home ready to be charged and 1 if the car is at home ready to be charged.

$$r = [r^1, r^2, r^3, ..., r^{726}] \tag{4.26}$$

The resulting constraint is

$$\sum_{i=1}^{10} s_i^j - r^j \leq 0, \text{ for } j = 1, 2, ..., 726. \tag{4.27}$$

meaning that the charger can only be enabled while the car is at home, ready to be charged. The constraint is called 'chargeOnlyWhen-Home' in the Matlab implementation.

The SoC constraint is defined as

$$\text{SoC}_{S \cdot min} \leq SoC_{(0)} + \frac{\text{cumsum}(charge) - \text{cumsum}(carCon)}{Q_{\text{nominal}}} \leq \text{SoC}_{S \cdot max} \tag{4.28}$$

The SoC constraint is split into two different constraints, with the names "SoCLowerBound" and "SoCUpperBound" in the Matlab implementation.

4.2.3 Defining an optimization problem

The problem in Matlab starts with defining an optimproblem, which instantiates an optimization problem [*What is regression?* N.d.] In the line below, an optimization problem of 'ObjectiveSense', meaning that the problem can either be a minimization or maximization, and 'minimize' lastly means that the problem is a minimization problem.

```
dispatch = optimproblem('ObjectiveSense','
    minimize');
```

Then, the problem given the name 'dispatch' has two accessible fields, Constraints and Objective. The objective field sets the relevant function that is to be optimized.

```
dispatch.Objective = f;
```

Lastly, the required constraints can be added by accessing the Constraints field of dispatch. The constraints must be given an individual name and the definition of the constraint. Some examples of two constraints are shown below.

```
dispatch.Constraints.SoCLowerBound = SoC >=
    SoC_min;
dispatch.Constraints.SoCUpperBound = SoC <=
    SoC_max;
```

If an optimproblem has two or more objective functions, one can add multiple objects by assigning each objective a name. This will enable the optimproblem to separate the objectives from each other.

```
dispatch = optimproblem('ObjectiveSense','
    minimize');
dispatch.Objective.soc = sum(SoC);
dispatch.Objective.cost = totalElectricityCost;
```

4.2.4 Input of driving pattern

To know when the car can be charged and when it is driving, the operator has to input the driving schedule for the upcoming days. The required details are during which hours the car will be operational, and the estimated distance the car will travel. It must also be known if the car arrives at home at the end of the drive, to verify if it is possible to charge or not at the end of the trip. This assumes that there is no available charging spot when parked outside of the home. To record a value of whether the car is at home or not, a column 'Home at end' is introduced. The value will be 0 if the car is not at home at the end of the drive, and 1 if the car is at home at the end of the drive.

The structure of the information required, and example values are provided in Table 4.1.

Start	End	Estimated distance	Home at end
2022-04-01 13:00	2022-04-01 15:45	120km	1
2022-04-02 08:05	2022-04-02 08:35	45km	0
2022-04-02 16:10	2022-04-02 16:45	45km	1

Table 4.1 The structure of the driving pattern

With the start and finish times, and the estimated distance to drive, it is possible to restrict whenever the car may or may not be charged, and how much energy is consumed during the trip, going off the average energy expenditure per kilometer stated in 4.1.

4.3 Different charging protocols

To test the cost-efficiency of the charging protocol with the forecasted prices from the primary models, other charging protocols are introduced. The charging protocols will be compared against each other with cost in mind. If the charging protocol which attempts to minimize the cost with the forecasted prices generally outperforms any other charging protocol, the forecasted prices can be deemed accurate enough to produce a favorable result when it comes to scheduling the charging.

The charging protocols that will be compared against each other are the following:

1. **Cost-optimized with forecast:** cost-optimization with forecasted electricity prices.

2. **Maximum SoC:** charge such that the sum of the SoC over the entire period is as high as possible. The charger immediately powers on with full power when the charger is plugged in. This can be seen as the naive protocol that would be the most straightforward way of charging.

3. **Minimum SoC:** charge such that the sum of the SoC over the entire period is as low as possible.

4. **Minimum SoC during 22-06:** only charge during the night, 22-06. It will try to charge during these hours such that the sum of the SoC over the entire period is as low as possible.

5. **Minimum SoC during 00-12:** only charge after midnight and 12 hours forward, 00-12. It will try to charge during these hours such that the sum of the SoC over the entire period is low as possible.

6. **Cost-optimized each day:** cost-optimization for the next 24 hours, with the charger updated with the new prices for the coming day at midnight of that day.

7. **Cost-optimized & minimum SoC:** charge such that the sum of the SoC over the entire period is as low as possible while trying to minimize the cost for the next 24 hours. At midnight, the charger is updated with the electricity prices for the next 24 hours.

It is important to note that not every charging protocol is developed with minimizing cost in mind. charging protocols 3, 4, 5, and 7 are developed based on the core idea of prolonging the lifetime of the battery, while protocol 2 attempts to reflect what is the most likely charging pattern used by most people with electric vehicles.

Of course, in some protocols, it isn't as simple as dividing each protocol into a single specific category, i.e. cost optimization or prolonging the lifetime of the battery. For charging protocols 4 and 5, the goal is to prolong the battery lifetime, while attempting to reduce cost at the same time by charging during what is normally cheap hours. Charging protocol 7 tries to find the optimal solution between the two objectives, not reducing the cost too much at the expense of the battery lifetime. Charging protocol 2 attempts to reflect what is thought to be the most common charging pattern and does not consider either cost optimization or prolonging the battery's lifetime.

To reiterate, if the first charging protocol is able to be more cost-efficient than the other six charging protocols, the first charging protocol is of use, even though the RMSE between the forecasted values and the true values is exceptionally high.

4.3.1 Cost of different charging protocols

Ten points in the dataset have been picked as samples, which hopefully serve as a guide for the cost-efficiency of each charging protocol. With the seven listed charging protocols, only graphs for $t_j = 2023 - 01 - 05$ are displayed, in order to keep the section as concise as possible. The cost for each charging protocol for each point in time will be displayed in a table, and the mean of the values will be calculated as an indication of its average cost.

The same driving routine will be used for every evaluation, only at different points in time. The starting point in time is denoted t_j which is the j-th point of the selected samples, where t_{j+i} is the i-th day from the starting point t_j. The starting time for each t_j will always be at midnight. The first drive will be at 07:00 the succeeding day of the day when the driving routine starts. The driving pattern can be expressed in a table with the following properties:

Start	End	Estimated distance	Home at end
t_{j+1} 07:00	t_{j+1} 17:00	295km	1
t_{j+2} 14:30	t_{j+2} 17:00	167km	0
t_{j+2} 22:30	t_{j+2} 23:00	50km	1
t_{j+3} 17:00	t_{j+3} 17:30	52km	0
t_{j+3} 18:30	t_{j+3} 23:30	150km	1
t_{j+4} 19:10	t_{j+4} 19:50	47km	1

Table 4.2 The driving routine for the experiments. The same driving routine is always applied, with different charging protocols.

Ten different values for t_j are picked, displayed in the seven Tables

4.3, 4.4, 4.5, 4.6 4.7, 4.8 and 4.9.

The driving pattern is called 'Moving', equivalent to the vector d aforementioned, which is shown below.

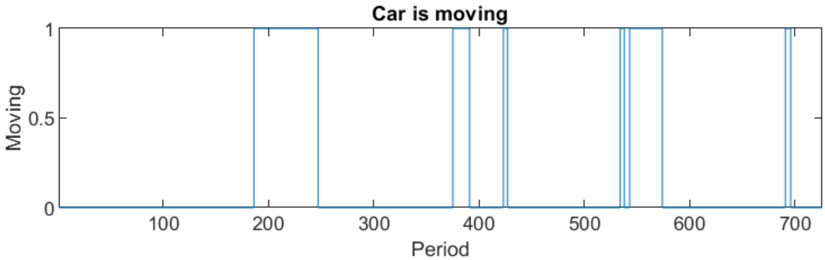


Figure 4.2 While Moving equals 1, the car is moving. Else, the car is at rest. From the graph, there are six different driving routes scheduled.

and the vector r aforementioned is displayed below.

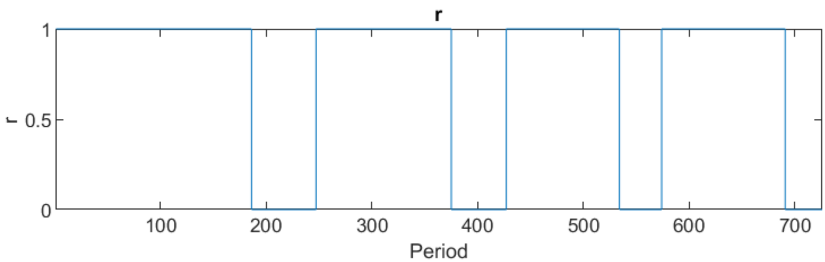


Figure 4.3 While r equals 1, the car may be charged. Else, it may not be charged. r is guaranteed to be 0 when the car is driving. From the graph, there exists 4 time intervals where the car may be charged.

4.3.1.1 First charging protocol. The first case is cost optimization with the forecasted electricity prices. The starting point in time is 2023-01-05 00:00. The forecast is displayed in Figure 3.21. The forecasted electricity price is for 2023-01-06 00:00 to 2023-01-10 00:00. The prices for the entire day of 2023-01-05 are already known at the start.

For the first case, the objective for the optimproblem is set to

```
dispatch = optimproblem('ObjectiveSense','
    minimize');
dispatch.Objective = totalCostElectricity;
```

The constraints are defined as

```
dispatch.Constraints.powercons = powercons;
dispatch.Constraints.noChargeWhenNotPossible =
    noChargeWhenNotPossible;
dispatch.Constraints.SoCLowerBound = SoC >=
    SoC_min;
dispatch.Constraints.SoCUpperBound = SoC <=
    SoC_max;
dispatch.Constraints.chargeAtHome =
    chargeOnlyWhenHome;
```

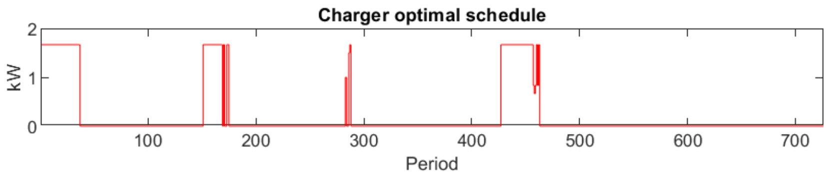


Figure 4.4 The charging pattern for the first protocol, t_j = 2023-01-05.

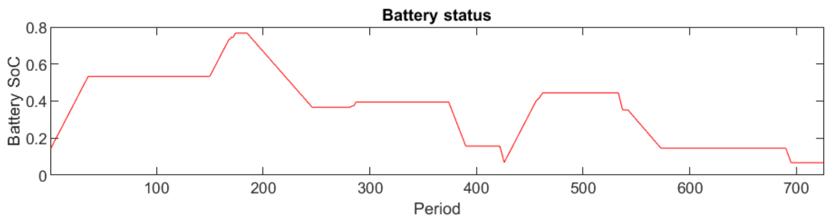


Figure 4.5 The battery status in SoC for the first protocol, t_j = 2023-01-05.

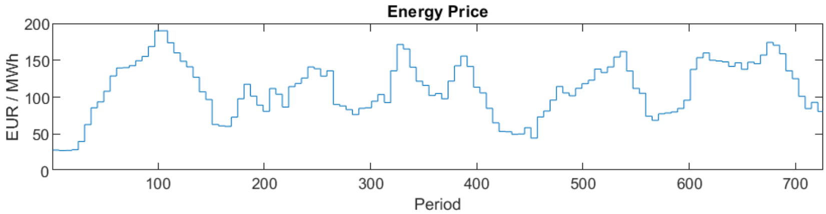


Figure 4.6 The price in EUR/MWh. The hourly electricity price for the first 24 hours is known, for hours 25-120, the prices are projected. t_j = 2023-01-05.

Starting date of scheduled driving (t_j)	Projected cost (EUR)	Cost (EUR)
2022-06-25	8.3652	19.6296
2022-07-05	5.5062	9.2148
2022-08-01	5.1888	1.083
2022-08-17	26.8108	37.4353
2022-09-07	12.9378	24.6793
2022-09-17	6.0011	5.6351
2022-11-17	5.4963	11.5044
2022-12-01	22.7686	37.5447
2022-12-20	12.43	16.1877
2023-01-05	7.7047	6.0663

Table 4.3 The cost for the first protocol

The "projected cost" is the optimal value from the optimization with the forecasted values. When applying the charging pattern from the optimization that gave the projected cost to the true prices, the result is stated in the column "Cost". The cumulative sum for all different starting points for the driving pattern is 168.980, with the average cost being 16.898.

4.3.1.2 Second charging protocol. The second protocol is to charge the battery such that it maximizes the summation of the SoC over the entire time period. This will simulate the case of always plugging in the charger when the car gets home, and the car charges at full power until it either reaches $\text{SoC}_{S \cdot max}$ or until the charger has to be disconnected as the car is about to leave. If the distance from the current SoC to $\mathrm{SoC}_{S^{\cdot}max}$ is less than $\frac{10}{6}$, then the charger can't operate with full power and chooses the maximum charging regime out of the possible charging regimes that are valid.

For the second case, the objective for the optimproblem is set to

```
dispatch = optimproblem('ObjectiveSense','
    maximize');
dispatch.Objective = sum(SoC);
```

The constraints are identical to the constraints in the first protocol.

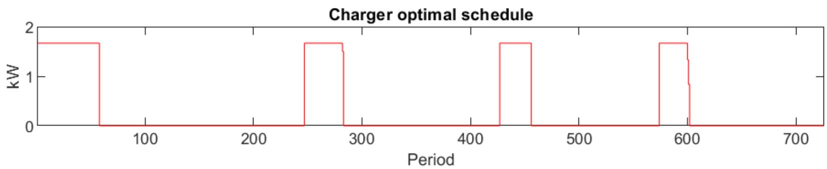


Figure 4.7 The charging protocol for the second protocol. t_j = 2023-01-05.

The corresponding SoC graph over the battery status with the charging pattern is depicted below.

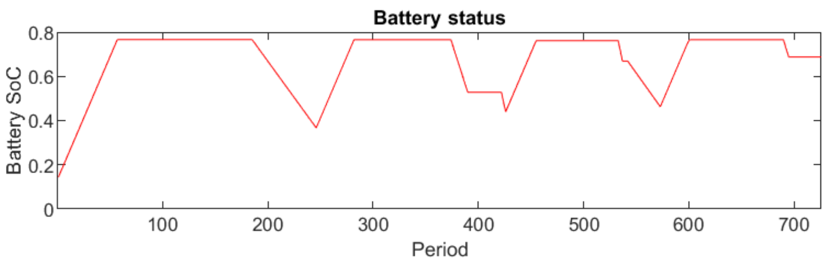


Figure 4.8 The corresponding battery SoC graph. t_j = 2023-01-05.

The price is not utilized in the optimization, as the objective is only to maximize the total summation of all the states of charges.

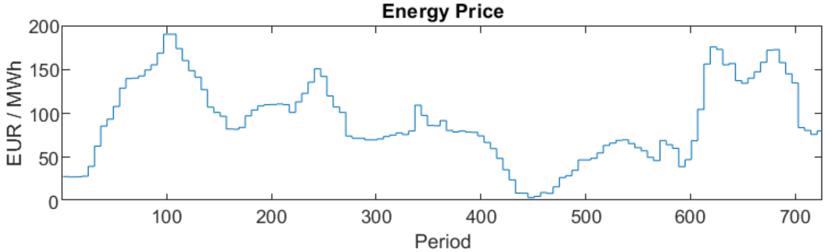


Figure 4.9 The price in EUR/MWh. The price does not play a factor in the optimization. t_j = 2023-01-05.

Starting date of scheduled driving (t_j)	Cost (EUR)
2022-06-25	51.7100
2022-07-05	32.3729
2022-08-01	13.6601
2022-08-17	79.1752
2022-09-07	67.8239
2022-09-17	25.5869
2022-11-17	25.9701
2022-12-01	69.1177
2022-12-20	31.0029
2023-01-05	14.8290

Table 4.4 The cost for the second protocol

The cumulative sum for all different starting points for the driving pattern is 411.245, with the average cost being 41.125.

4.3.1.3 ***Third charging protocol.*** The third scenario is to charge the car only as much as is needed for the next ride(s). It is equivalent to keeping the summation of the state of charges to a minimum. The charger would charge the car for the next single ride if the car is able to charge again after the ride. If there are multiple rides with no charge in between, it has to charge for all the rides beforehand.

```
dispatch = optimproblem('ObjectiveSense','
    minimize');
```

```
dispatch.Objective = sum(SoC);
```

The constraints are identical to the constraints in the first protocol.

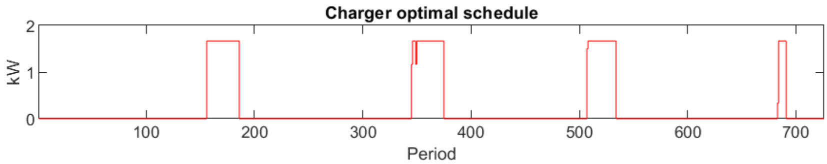


Figure 4.10 The charging pattern for the third protocol, t_j = 2023-01-05.

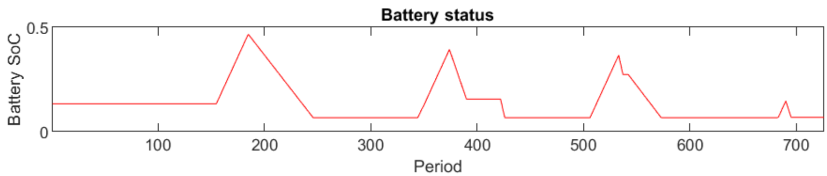


Figure 4.11 The battery SoC graph for the third protocol, t_j = 2023-01-05.

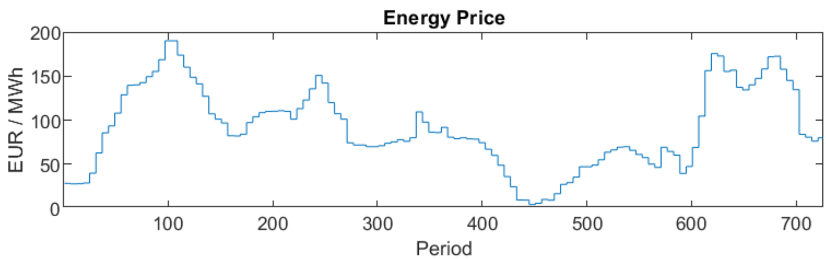


Figure 4.12 The price in EUR/MWh. The price does not play a factor in the optimization. t_j = 2023-01-05.

Starting date of scheduled driving (t_j)	Cost (EUR)
2022-06-25	39.1834
2022-07-05	23.9657
2022-08-01	2.5175
2022-08-17	59.7281
2022-09-07	49.0956
2022-09-17	31.7300
2022-11-17	27.7800
2022-12-01	49.1463
2022-12-20	32.0049
2023-01-05	13.4097

Table 4.5 The cost for the third protocol

The cumulative sum for all different starting points for the driving pattern is 328.561, with the average cost being 32.856.

4.3.1.4 Fourth charging protocol. The fourth protocol is to charge during the night, during the hours 22 - 06. The total summation of SoC is minimized.

```
dispatch = optimproblem('ObjectiveSense','
    minimize');
dispatch.Objective = sum(SoC);
```

The constraints are defined as

```
dispatch.Constraints.powercons = powercons;
dispatch.Constraints.noChargeWhenNotPossible =
    noChargeWhenNotPossible;
dispatch.Constraints.SoCLowerBound = SoC >=
    SoC_min;
dispatch.Constraints.SoCUpperBound = SoC <=
    SoC_max;
dispatch.Constraints.chargeForSelectHours =
    chargeForSelectHours;
dispatch.Constraints.chargeAtHome =
    chargeOnlyWhenHome;
```

The constraint 'canChargeAtNight' makes sure that the car is only able to be charged during 22 - 06. The total summation of SoC is minimized.

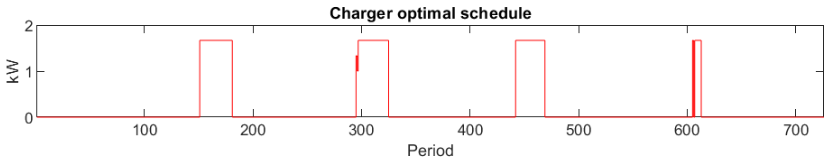


Figure 4.13 The charging pattern for the fourth protocol, t_j = 2023-01-05.

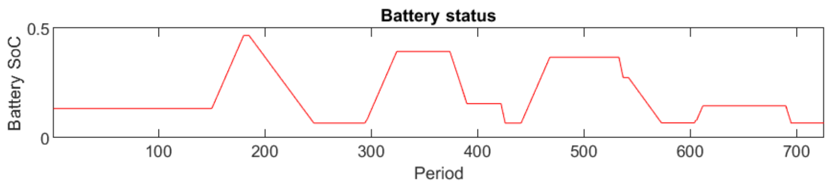


Figure 4.14 The battery SoC graph for the fourth protocol, t_j = 2023-01-05.

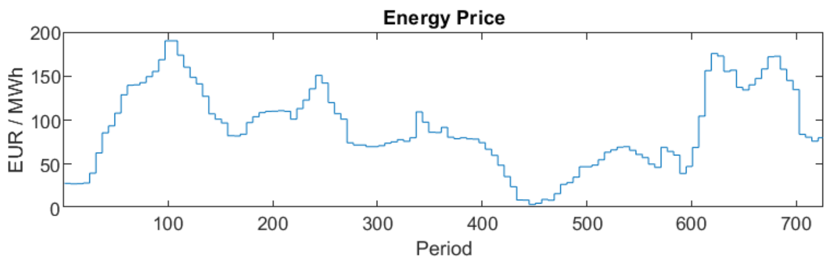


Figure 4.15 The price in EUR/MWh. The price does not play a factor in the optimization. t_j = 2023-01-05.

Starting date of scheduled driving (t_j)	Cost (EUR)
2022-06-25	37.6816
2022-07-05	14.8561
2022-08-01	0.9501
2022-08-17	43.2250
2022-09-07	28.0739
2022-09-17	17.5794
2022-11-17	20.9167
2022-12-01	39.3229
2022-12-20	21.3977
2023-01-05	9.5237

Table 4.6 The cost for the fourth protocol

The cumulative sum for all different starting points for the driving pattern is 233.527, with the average cost being 23.353.

4.3.1.5 Fifth charging protocol. The fifth protocol is to charge during the hours 00-12 while minimizing the summation of SoC.

```
dispatch = optimproblem('ObjectiveSense','
    minimize');
dispatch.Objective = sum(SoC);
```

The constraints are defined as

```
dispatch.Constraints.powercons = powercons;
dispatch.Constraints.noChargeWhenNotPossible =
    noChargeWhenNotPossible;
dispatch.Constraints.SoCLowerBound = SoC >=
    SoC_min;
dispatch.Constraints.SoCUpperBound = SoC <=
    SoC_max;
dispatch.Constraints.chargeForSelectHours =
    chargeForSelectHours;
dispatch.Constraints.chargeAtHome =
    chargeOnlyWhenHome;
```

The constraint 'canChargeAtNight' makes sure that the car is only able to be charged during 00 - 12.

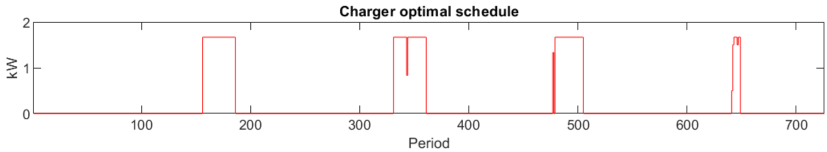


Figure 4.16 The charging pattern for the fifth protocol, t_j = 2023-01-05.

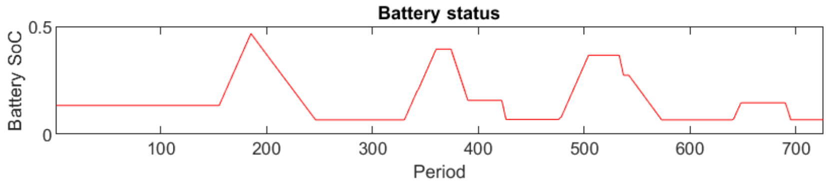


Figure 4.17 The battery SoC graph for the fifth protocol, t_j = 2023-01-05.

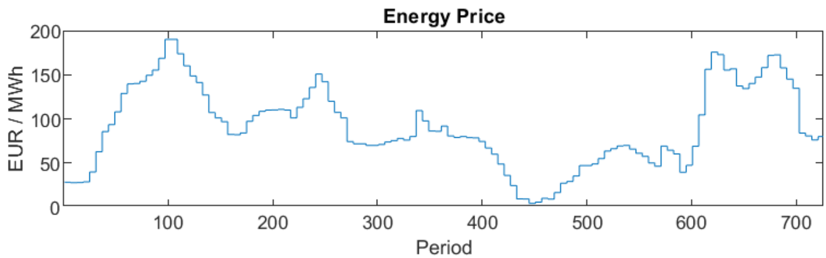


Figure 4.18 The price in EUR/MWh. The price does not play a factor in the optimization. t_j = 2023-01-05.

Starting date of scheduled driving (t_j)	Cost (EUR)
2022-06-25	37.2157
2022-07-05	26.4858
2022-08-01	2.8345
2022-08-17	58.1893
2022-09-07	46.8930
2022-09-17	32.6274
2022-11-17	26.3367
2022-12-01	46.6908
2022-12-20	32.2013
2023-01-05	12.3623

Table 4.7 The cost for the fifth protocol

The cumulative sum for all different starting points for the driving pattern is 321.837, with the average cost being 32.184.

4.3.1.6 Sixth charging protocol. In the sixth protocol, the objective is once again to minimize cost. In this scenario, the charger is updated at midnight with the electricity prices for the next 24 hours. The charger only has knowledge about the next 24 hours at midnight. The time horizon spans five days, which means that the charger attempts to minimize the cost for the current day, and does so repeatedly five times, one time for each day in the time span.

For the sixth case, the objective for the optimproblem is set to

```
dispatch = optimproblem('ObjectiveSense','
   minimize');
dispatch.Objective.cost = totalElectricityCost;
```

The constraints are identical to the constraints in the first protocol.

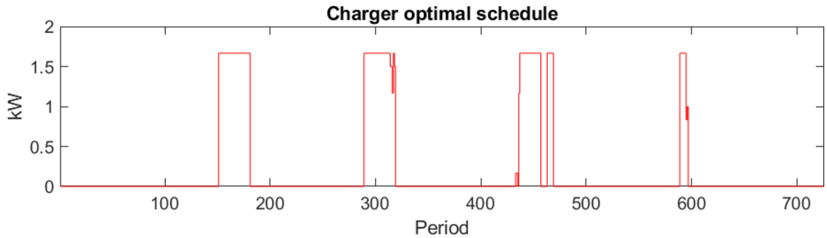


Figure 4.19 The charging pattern for the sixth protocol, t_j = 2023-01-05.

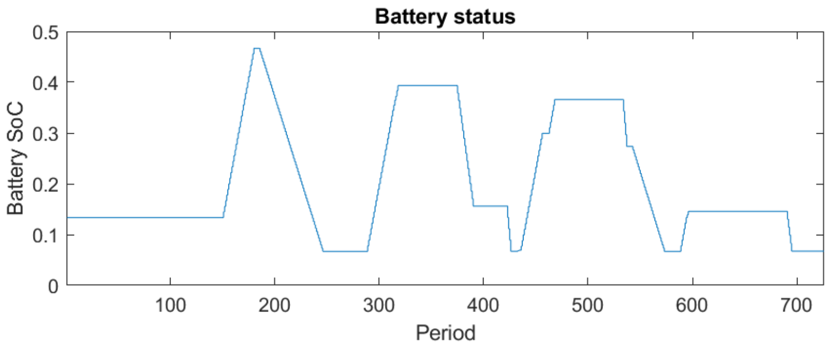


Figure 4.20 The battery SoC graph for the sixth protocol, t_j = 2023-01-05.

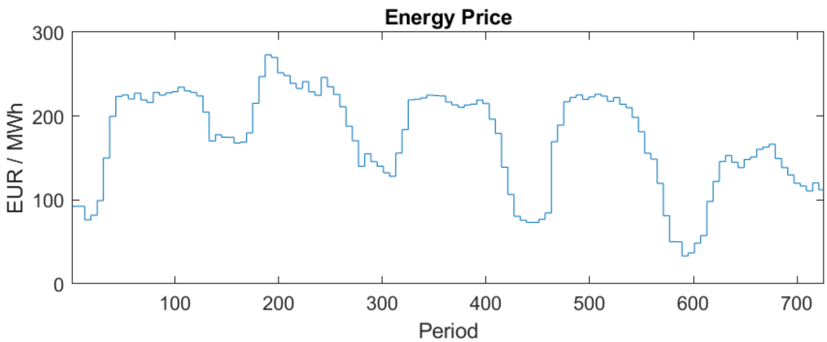


Figure 4.21 The price in EUR/MWh. t_j = 2023-01-05.

Starting date of scheduled driving (t_j)	Cost (EUR)
2022-06-25	30.8060
2022-07-05	5.8239
2022-08-01	0.5129
2022-08-17	31.8186
2022-09-07	19.6584
2022-09-17	7.4529
2022-11-17	19.1756
2022-12-01	37.4064
2022-12-20	19.3342
2023-01-05	8.7511

Table 4.8 The cost for the sixth protocol

The cumulative sum for all different starting points for the driving pattern is 180.74, with the average cost being 18.074.

4.3.1.7 Seventh charging protocol. For this protocol, the intlinprog solver is replaced by 'gamultiobj' which is a solver capable of handling multi-objective optimization.

The objectives are set to the summation of SoC, and the total electricity cost. The solver attempts to minimize both objectives.

```
dispatch = optimproblem('ObjectiveSense','
    minimize');
dispatch.Objective.soc = sum(SoC);
dispatch.Objective.cost = totalElectricityCost;
```

The constraints are identical to the constraints in the first protocol.

The Pareto fronts are

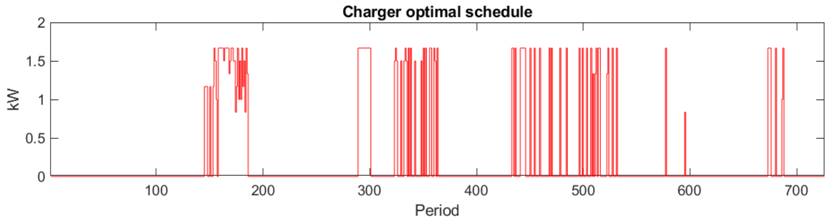


Figure 4.22 The charging pattern for the seventh protocol, t_j = 2023-01-05.

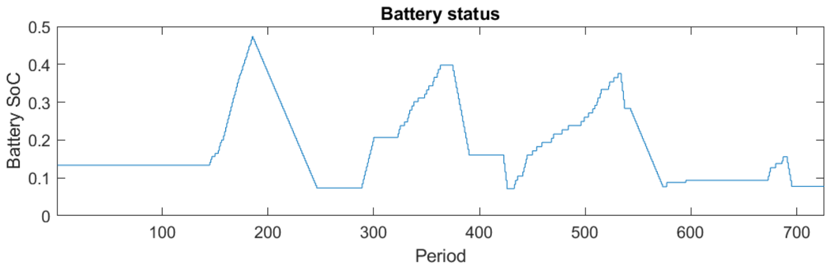


Figure 4.23 The battery SoC graph for the seventh protocol, t_j = 2023-01-05.

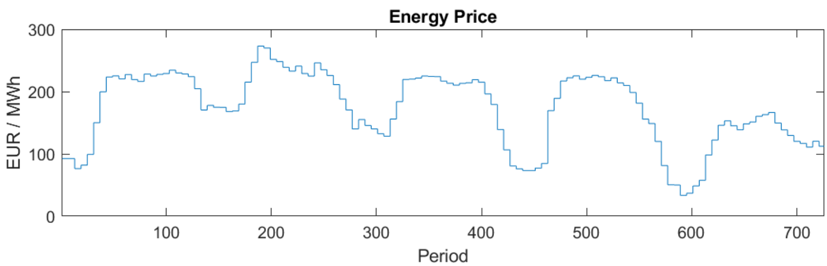


Figure 4.24 The price in EUR/MWh. t_j = 2023-01-05.

Starting date of scheduled driving (t_j)	Cost (EUR)
2022-06-25	37.4130
2022-07-05	17.8977
2022-08-01	2.1862
2022-08-17	50.8032
2022-09-07	39.1703
2022-09-17	25.0074
2022-11-17	24.0288
2022-12-01	44.9339
2022-12-20	27.5942
2023-01-05	11.66019

Table 4.9 The cost for the seventh protocol

The cumulative sum for all different starting points for the driving pattern is 280.695, with the average cost being 28.070.

4.3.1.8 Cost-efficiency. Displaying the average costs for each charging protocol

1. 16.898 EUR
2. 41.125 EUR
3. 32.856 EUR
4. 23.353 EUR
5. 32.184 EUR
6. 18.074 EUR
7. 28.070 EUR

Measuring the cost-efficiency based on the averages, the first protocol is on average

- 58.91% cheaper than protocol 2
- 48.57% cheaper than protocol 3

- 27.64% cheaper than protocol 4
- 47.50% cheaper than protocol 5
- 6.51% cheaper than protocol 6
- 39.80% cheaper than protocol 7

When measuring the cost-efficiency based on the averages, the second protocol, the naive baseline, is on average

- 143.37% more expensive than protocol 1
- 25.17% more expensive than protocol 3
- 76.10% more expensive than protocol 4
- 27.78% more expensive than protocol 5
- 127.54% more expensive than protocol 6
- 46.50% more expensive than protocol 7

The Table 4.10 has a cross in the cell for the charging protocol which is the most cost-efficient, for every t_j.

From Table 4.10, the first protocol and the sixth protocol have a charging protocol that generates the most favorable outcome in terms of cost-efficiency. In five cases, the best charging protocol is the first protocol. In the remaining five cases, the best charging protocol is the sixth protocol.

Starting date of scheduled driving (t_j)	protocol 1	protocol 2	protocol 3	protocol 4	protocol 5	protocol 6	protocol 7
2022-06-25	X						
2022-07-05						X	
2022-08-01						X	
2022-08-17						X	
2022-09-07						X	
2022-09-17	X						
2022-11-17	X						
2022-12-01						X	
2022-12-20	X						
2023-01-05	X						

Table 4.10 Most cost-efficient protocol for each data point t_j.

5

Discussion

Reiterating the results from the two different forecast models, the baseline models showed a much more reliable forecast ability than the primary models. The primary models showed a less reliable forecast ability in comparison. It was naturally expected since the data being trained on and being tasked to forecast is much more volatile for the primary models. It can also be motivated that the factors behind the electricity price for the dataset the primary models are trained on are much more difficult to capture. There are a few worthwhile points to discuss regarding the models and the points are brought up in 5.1.

The optimization of cost with forecasted prices showed an encouraging result, being the most cost-efficient protocol 50% of the time. The method of comparison is obviously not as detailed as one would prefer since it only studies six other charging protocols, and only does so for ten points in time, all operating with the same driving regime. There are a few points to bring up about the cost-optimization which will be discussed in 5.2.

5.1 Forecasting electricity prices

In the process of establishing features for the XGBoost models, the process has primarily been to incorporate factors where other papers have demonstrated a correlation between the factor and the electricity price. Then, the data for these factors have been gathered and manipulated in multiple ways such that it assists the XGBoost models to capture the behavior of the factor. In reality, factors that are thought to have an impact on the electricity price without any specific paper establishing the correlation can be constructed as long as the motivation for doing so is sufficient. If the feature decreases the RMSE metric for the models, it could potentially be an underlying factor of the price. In reverse, if the feature increases the RMSE metric, it is most likely not an underlying factor of the price. However, the primary approach in this thesis was to go by already established factors of the electricity price that other papers have stated. The word 'primarily' is used, as not every factor has a source to a paper stating its effect on the electricity price. One example is the 'EUR/SEK currency pair' that was introduced in the models with a motivation to why it could potentially make the model better, but it was later removed as it showed a deteriorating effect on the forecasting ability of prices.

The task of being able to forecast future values of any variable is complex and cannot be understated, especially for variables where the number of underlying factors is immense. Some of the most probable factors linked to the variable itself have to be designated, which isn't always an easy task. Indeed, a few probable components of the electricity price brought up in this thesis and integrated with the XGBoost model showed an improvement in the ability of the XGBoost models to forecast future electricity prices.

There is arguably an uncountable number of components of the electricity price that have not been covered in this thesis. Not only are there current components that have not been covered, the components themselves can alter over time, becoming either less or more relevant. This means that a component of the XGBoost models at the moment could become irrelevant in the future. A prime example of an alternating component is the invasion of Ukraine. The component of political instability and war evidently had an instantaneous impact on the electricity price, changing almost overnight. In addition to the political instability, the following consequences of the instability are hard to predict, and even more difficult to determine the impact of. An example of a consequence of the war in Ukraine is the abrupt end of the Nordstream pipeline. An example of a component becoming less and less relevant is for example the price of commodities such as gas, coal, and oil. As more and more countries adopt environmental measures, phasing out the use of commodities with high carbon prints, the electricity price would logically become less dependent on the price of these commodities. In general, most components of the electricity price could be categorized as societal, political, governmental, economical, or environmental in nature. From time to time, components of these categories rapidly develop and with it changes the outlook of future electricity prices.

Not only are there apparent components of the pricing, e.g. what one would normally assume to have an impact on the electricity price, but there are also obscured components, e.g. the market manipulation of the electricity price. As late as May of 2023, a major producer of electricity in Sweden has been accused of market manipulation [Johanna and TT, 2023]. There are countless cases of the electricity market being manipulated, with one specific case that has been brought up in this thesis, and there is no reason to assume that the manipulation will cease. It continues to operate as a 'hidden' component, i.e. being both hard to detect and hard to measure unless extensive research is performed. Market manipulation is not considered to have a large impact on the result achieved in this thesis, however, it is something to keep in mind as it is prevalent.

5.1.1 Assessing performance with previous research

To evaluate the forecasting ability and its performance in this thesis, the forecasting results can be compared with three other papers brought up under Section 1.4. Without the three papers, it is hard to determine the outcomes without personal speculation concerning the result.

The first comparison is the result obtained in [Wu et al., 2022]. Observing the RMSE values in Table 1.1, it is evident how large the spread is between the minimum value and the maximum value, spanning all the way from 0.6081 for January to 39.2473 for May. The time horizon for the test data in [Wu et al., 2022] is between 3 and 4 days depending on the month, which is close to the time horizon in this thesis, being 4 days, since the forecasting is done for a total of 97 hours. Two core properties that differ in the execution are the amount of data the models are trained on and the number of features in each model. The models in this thesis are trained on much more extensive data and the models are constructed with many more features. Regarding the RMSE metrics, the baseline models are in the ballpark of the RMSE range established in [Wu et al., 2022], with the MCRMSE for the baseline models measuring in at 12.942 EUR/MWh, and the example illustrated in Figure 3.12 measuring in at 9.246 EUR/MWh. Unfortunately, the paper does not include any attempts to forecast data with the same inherent volatility that exists in the dataset the primary models are trained on, making it hard to draw a conclusion regarding the primary models. Lastly, the volatility in the datasets used in [Wu et al., 2022] is for the most part higher in comparison to the 2015-2018 dataset, with exceptions in March and April where the difference between the lowest electricity price and the highest electricity price for the month is lower. Conclusively, the results of forecasting ability with the baseline models show a result in line with the results from [Wu et al., 2022], which is encouraging.

The second comparison is with the paper conducted by Deepak Singhal and K.S. Swarup, [Singhal and Swarup, 2011]. They obtained an RMSE measurement of 0.525 when attempting to project the future electricity prices during a "normal trend" in the electricity price, 1.129 in the case of a "small spike" and 4.105 in the case of a "large spike". The volatility in 3.12 is around the same level of volatility as what is considered a "normal trend" for Singhal And Swarup. The RMSE in figure 3.12 is more than 17 times higher than the RMSE for a "normal trend", measuring in at 9.246. It is however suitable to loosely compare the two numbers, as the numbers from Singhal and Swarup provide an additional gauge to measure the performance of the machine-learning models. Even though the magnitude of the RMSE is 17 times greater in this thesis, it could be considered somewhat close in absolute terms, since the difference is $9.246 - 0.525 = 8.721$. If the RMSE difference in absolute terms was 50, one could start to question the validity of the baseline models.

The third comparison is with the paper by Albahli, Shiraz, and Ayub where they achieved a RMSE of 9.25, a number remarkably close to the RMSE of 9.246 in 3.12. Having used the same architecture of XGBoost for a similar task, it is an additional assurance to see that the baseline models achieved a similar result, strengthening the belief that the baseline models are of exemplary performance in tandem with other research conducted.

As stated, it is hard to draw any conclusion between the primary models and previous research, as there has been no paper found yet that attempts to perform a similar task to what the primary models are doing.

5.1.2 Comparing the baseline models and primary models

The primary models are not in the vicinity of the result obtained from either the baseline models or any previous research analyzed. It could be argued that the RMSE metrics for the primary models distantly protocol some of the RMSE values from [Wu et al., 2022]. However, there is still an enormous gap. The MCRMSE metric of the primary models was expected to be much higher than the baseline models, but not *ten* times higher. After all, the quote between the standard deviations for the two datasets was approximately 7.2, which implies a substantial increase in the dispersion of values from the mean in the more volatile dataset between 2020 and 2023. Considering all factors, it sounds probable that a ten times increase is reasonable given the circumstances.

Notably, the side-by-side comparison of the average RMSE for the baseline models and the primary models illustrated two different behaviors. The trend for both models is that the average RMSE primarily gauges higher, with occasional stagnation. The baseline models find their highest average RMSE for the 110th hour, different from the 120th hour the primary models find most difficult to forecast. The reason why this occurs could be due to multiple reasons, having to do both with the models and the dataset. To start off with, the models do not share the same exact features. This will inevitably produce two distinct models. One probable explanation is that the models capture behavior differently since they deviate from each other, not being identical. Having different features will establish a different ability to capture patterns and trends. Falling back to the dataset, which has been thoroughly established to be of different characteristics, it is bound to produce models of different capabilities and strengths. It may also be a question of under- or overfitting, even though it has been tried to be mitigated as much as possible by using appropriate values for parameters to the model.

5.2 Optimization of charging patterns

When comparing the charging pattern generated by the first charging protocol to other charging protocols, it is evident that the cost-optimized charging protocol with forecasted prices is one of the best. However, there are two apparent points that must be addressed. The first point is that there are a limited number of charging protocols that are evaluated and compared. The second point is the *very* limited number of samples that have been picked and used to evaluate the cost of each charging protocol.

5.2.1 Charging protocols & limited sampling

Only having six additional charging schemes is bound not to test the robustness of the first charging protocol. With the rise of electric vehicle charging stations, more and more sophisticated protocols are inevitably developed. If more time was allocated to the optimization part of this thesis, more in-depth research could have been done to find the currently most optimal charging schedules that currently exist on the market.

The charging scheme number two is supposed to be the naive protocol, assuming that most people normally would plug in the car immediately when they get home and let it charge at full power, until it reaches the maximum level. The other four (excluding the first, second, and sixth protocols) charging protocols were established based on what was assumed to be potential cost-competitive protocols, while also keeping the health of the battery in mind. Based on cost-efficiency, the protocols in reality turned out to not be cost-efficient at all. For example, charging during the hours 22-06 when the electricity price usually trades at around its lowest point for the day was thought to be a cost-effective implementation while not totally disregarding the health of the battery. The fifth protocol builds upon the same idea, but the available hours are adjusted to 00-12. Even though the hours 00-12 predominantly include some of the priciest hours of the day for electricity, the idea was that it may produce favorable results in some scenarios. Admittedly, there could be the possibility of protocols four and five being more cost-efficient in certain situations if the number of points evaluated is vastly expanded.

One relevant question is then, if more points were sampled and evaluated, would it at any time beat the first or sixth protocol which currently dominates in cost-efficiency? Unfortunately, due to a shortage of time, it was not possible to do it for more than the ones selected in the thesis. Even though the shortcomings and obstacles in the thesis, it was possible to produce a consistent result, at least for the random points picked in time. The consistent result is that the first protocol and the sixth protocol with the forecast of electricity prices always outperformed the other charging protocols.

In hindsight, the sixth and seventh protocols could have been modeled differently. At this time, the scenario is that the charger receives the new prices for the coming day at 00.00, having the prices for the next 24 hours forward. In reality, the electricity prices for the next day are released at 13.00 of the current day. If the charger was updated with the prices every day at 13.00, the charger would have access to the next 35 hourly prices, making it capable of optimizing the cost with a longer time horizon. In the case where the charger is updated at 13.00 with the coming days' prices, it could determine if it should charge during the current day or wait until the next day, depending on the prices.

One other consideration can be to allow the car to charge at other locations that are not only at home in the garage. With the rapid expansions of fast chargers at multiple locations, it wouldn't be a rare scenario for a driver of an electric vehicle to use a fast charging station if the car is traveling longer than the range of the battery.

The first protocol with the forecasted electricity prices showed on average to be the cheapest charging protocol. It is on average 6.5% cheaper than the second most cost-efficient charging protocol and on average 58.9% cheaper than the most cost-inefficient charging protocol. The most cost-inefficient charging protocol is the charging protocol believed to be the naive charging approach in this thesis. As the introduction stated, the idea was to investigate whether it is possible to reduce energy costs for an individual consumer. If the second charging protocol is what most people adhere to, then the answer to the specific question can be answered. In this thesis, the answer is convincingly yes.

To compare the result of cost optimization in this thesis to a paper that performed the two fundamental steps identical to this thesis, i.e. forecasting electricity prices and performing cost optimization with the forecasted prices, one can take a closer look at the paper [Albahli et al., 2020] that is initially brought up in the Section 1.4. The reason why the authors perform forecasting and optimization is similar to the reason in this thesis, as the authors state, they do it due to "recent spikes in electricity prices". The paper can provide a metric to compare the result of the cost optimization. In the paper, the authors were able to reduce costs by 25.32%, which is impressive. If charging protocol two is the protocol to compare against, as that is what is seen as the naive protocol, charging protocol one is on average 58.91% cheaper.

5.2.2 The driving routine

To make the research more consistent and bulletproof, in addition to more charging protocols and samples in time with forecasts, the driving pattern itself could be modified for different scenarios, not only going by the driving routine in Table 4.2. The planned driving routine has always been during the same hours of the day, for a fixed number of kilometers, and with static values if the car is able to charge at the end of the planned drive or not. Having the car operate during other hours of the day than what is currently used to evaluate the cost-efficiency will very likely have an impact on the final result of the cost-efficiency.

5.2.3 Time intervals

The time intervals have been divided into segments of 10 minutes each. The number of segments an hour is divided into dictate the level of granularity. If an hour was divided into more intervals, the charger could have performed more granular switches between different types of wattage outputs, most likely reducing the overall cost, at least when minimizing the cost. To have a more granular approach, the number of intervals each hour is divided into could be increased to for example 30, meaning each interval corresponds to 2 minutes.

6

Conclusion & future work

In this thesis, the question of whether cost optimization with forecasted electricity prices could reduce the financial expense of charging an electric vehicle was evaluated. In order to answer the question, the question was divided into two parts, one part dealing with the forecasting ability and the other part dealing with the optimization.

6.1 Models

The forecasting ability was put to the test using the machine-learning model XGBoost which is a state-of-the-art model. The models were divided into two categories, the baseline models and the primary models. The idea behind the baseline models was to build a model and train it on less volatile data, in order to measure its performance. It is then a baseline to measure against when constructing the primary models, which are trained on much more volatile data. Both categories of models consist of 97 models themselves, where each individual model has the task to forecast the electricity price for a particular hour in the future. With 97 unique models, the ability to forecast future electricity prices for a total of 97 hours was established. The first model of each category forecasted the electricity price 24 hours in advance and the last model of each category forecasted the electricity price 120 hours in advance.

Both of the models were trained on data with features that were thought to have a link in the determination of the electricity price.

There is a small difference in the features of the baseline models and the primary models. The data for training and testing have been gathered from various sources that provide information regarding weather, cross-border trading, electricity prices, etcetera.

The baseline models performed much better at forecasting future electricity prices in comparison to the primary models, based on the RMSE and MCRMSE metrics. The baseline models showed a rather positive result, being able to forecast future electricity prices quite well. Studying related research to the machine-learning models in this thesis, the RMSE metric of the baseline models was generally in line with the results of the studied papers. The case of the primary models is different, as there has not been any study found yet that attempts to forecast electricity prices on data with the same inherent volatility. Given the volatility and a large number of factors of the price, the assumption is that the primary models perform well given the circumstances.

6.2 Optimization

Seven unique charging protocols, i.e. optimization of charging with defined objectives and constraints, have been established where each protocol outputs a particular charging pattern based on the pricing information provided to it and its objectives to optimize. Two of the constructed protocols stood out in terms of cost-efficiency. The first charging protocol that excelled is charging protocol number one, utilizing the forecasted electricity prices from the primary models. Even though the forecasts from the primary models are quite diverging from the true prices established on the electricity market in the future, the predictions are close enough in order for the charger to determine if it should charge during the current day or wait til the future to charge. The second charging protocol that excelled in cost-efficiency is protocol number six. The sixth protocol attempts to optimize the cost during the next 24 hours, not having knowledge about the future electricity prices. In total, it optimizes the cost for the next 24 hours for 5 consecutive days.

Each optimization had a number of constraints placed on it. Most of the constraints placed on the optimization problems are fundamental, such as the car cannot be charged when it is driving, the state of charge of the battery must remain in a defined range in order to promote battery longevity, and the charger cannot be in multiple regimes, i.e. charging outputs in kW, at the same time. Each optimization problem had one or more objectives that it aimed to either minimize or maximize.

Not all charging protocols that are compared to the first charging protocol are designed with merely cost-efficiency in mind. Some protocols, especially numbers three, four, and five are designed with battery longevity in mind. Number seven attempts to minimize both the cost and the sum of the state of charge. Protocol number one and six merely attempts to minimize the cost of charging, i.e. the cost-efficiency, with no regard for how the charging is performed and the impact it has on the battery. How the charging of the battery is executed has implications on the state and the longevity of the battery. Finding a balance between the cost and longevity of the battery would be the most optimal solution.

The first charging protocol using the forecasted electricity prices was measured to be the best-performing charging protocol based on its average implementation cost. Protocols three, four, five, and seven showed to not be cost-efficient charging protocols at all.

6.3 Future research

The XGBoost model can be used as the foundation which other features can be integrated into. With an already diverse set of features to work upon, more and more components of the electricity price can be added. If features are added with a correlation to the electricity price, the performance will be improved.

With regard to the XGBoost models, more research can be performed on the optimization of hyperparameters for the model. The authors in [Wu et al., 2022], cited in Section 1.4 utilized particle swarm optimization to find the optimal parameters for their XGBoost model. A similar method can be executed in order to find the optimal hyperparameters for the models in this thesis.

With further research, the answer to the question, of whether the first protocol and the sixth protocol are always more optimal charging schedules, could be answered more profoundly. The idea of further research would then be to establish several more charging protocols and sample as many points in time as possible. The more evaluations being performed, the more reliable the result is. Ideally, more cost-reduction optimization should be performed that also considers the longevity of the battery.

References

About us (n.d.). https://www.nordpoolgroup.com/en/About-us/. Accessed: 2023-01-31.

About us (n.d.). https://www.nordpoolgroup.com/en/the-power-market/Bidding-areas/. Accessed: 2023-01-31.

Afanasyev, D. O., E. A. Fedorova, and E. V. Gilenko (2021). "The fundamental drivers of electricity price: a multi-scale adaptive regression analysis." *Empirical Economics* **60**:4, pp. 1913–1938. ISSN: 03777332. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=ecn&AN=1891764&site=eds-live&scope=site.

Albahli, S., M. Shiraz, and N. Ayub (2020). "Electricity price forecasting for cloud computing using an enhanced machine learning model". *IEEE Access* **8**, pp. 200971–200981. DOI: 10.1109/ACCESS.2020.3035328.

Balance responsibility (n.d.). https://www.svk.se/en/national-grid/operations-and-electricity-markets/balance-responsibility/. Accessed: 2023-02-07.

Bang, N., F. Fock, and M. Togeby (2023). "9-06-2011 the existing nordic regulating power market".

Baunsgaard, M. T. and N. Vernon (2022). *Taxing Windfall Profits in the Energy Sector*. International Monetary Fund.

Belaïd, F. (2022). "Implications of poorly designed climate policy on energy poverty: global reflections on the current surge in energy prices". *Energy Research Social Science* **92**, p. 102790. ISSN: 2214-6296. DOI: https://doi.org/10.1016/j.erss.2022.102790. URL: https://www.sciencedirect.com/science/article/pii/S2214629622002936.

Brent vs WTI: Which crude to trade in 2022? (N.d.). https://capital.com/brent-vs-wti-which-crude-oil-to-trade. Accessed: 2023-04-16.

Buechler, E., S. Powell, T. Sun, N. Astier, C. Zanocco, J. Bolorinos, J. Flora, H. Boudet, and R. Rajagopal (2022). "Global changes in electricity consumption during covid-19". *iScience* **25**:1, p. 103568. ISSN: 2589-0042. DOI: https://doi.org/10.1016/j.isci.2021.103568. URL: https://www.sciencedirect.com/science/article/pii/S2589004221015388.

Castelli, M., A. Groznik, and A. Popovič (2020). "Forecasting electricity prices: a machine learning approach". *Algorithms* **13**, p. 119. DOI: 10.3390/a13050119.

Celasun, O., A. Mineshima, N. Arregui, V. Mylonas, A. Ari, I. Teodoru, S. Black, K. Zhunussova, D. Iakova, and I. Parry (2022). "Surging energy prices in europe in the aftermath of the war: how to support the vulnerable and speed up the transition away from fossil fuels". *IMF Working Papers* **2022**:152, p. 1. DOI: 10.5089/9798400214592.001. URL: https://doi.org/10.5089/9798400214592.001.

Chang, Z., Y. Zhang, and W. Chen (2019). "Electricity price prediction based on hybrid model of adam optimized lstm neural network and wavelet transform." *Energy* **187**. ISSN: 0360-5442. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edselp&AN=S0360544219314768&site=eds-live&scope=site.

Chapter 5 XGBoost (n.d.). https://bookdown.org/v_anandkumar88/docs2/xgboost.html. Accessed: 2023-03-15.

Cheikh-Mohamad, S., M. Sechilariu, and F. Locment (2021). "Pv-powered charging station: energy management and cost optimization." *2021 IEEE 30th International Symposium on Industrial Electronics (ISIE), Industrial Electronics (ISIE), 2021 IEEE 30th International Symposium on*, pp. 1–6. ISSN: 978-1-7281-9023-5. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edseee&AN=edseee.9576324&site=eds-live&scope=site.

Chen, T. and C. Guestrin (2016). "Xgboost: A scalable tree boosting system". *CoRR* **abs/1603.02754**. arXiv: 1603.02754. URL: http://arxiv.org/abs/1603.02754.

Christie, D. and S. P. Neill (2022). "8.09 - measuring and observing the ocean renewable energy resource". In: Letcher, T. M. (Ed.). *Comprehensive Renewable Energy (Second Edition)*. Second Edition. Elsevier, Oxford, pp. 149–175. ISBN: 978-0-12-819734-9. DOI: https://doi.org/10.1016/B978-0-12-819727-1.00083-2. URL: https://www.sciencedirect.com/science/article/pii/B9780128197271000832.

Chze, O. N. and A. A. Abdullah (2022). "Covid-19 mrna vaccine degradation prediction by using deep learning algorithms". In: *2022 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, pp. 444–450. DOI: 10.1109/3ICT56508.2022.9990617.

COMMISSION, E. (2022). "Council regulation (eu) 2022/1854 of 6 october 2022 on an emergency intervention to address high energy prices". *Official Journal of the European Union* **LI 261/1**.

Contreras, J., R. Espinola, F. Nogales, and A. Conejo (2003). "Arima models to predict next-day electricity prices". *IEEE Transactions on Power Systems* **18**:3, pp. 1014–1020. DOI: 10.1109/TPWRS.2002.804943.

Dashboard (n.d.). https://transparency.entsoe.eu/. Accessed: 2023-03-14.

Deep Learning Basics Lecture 3: Regularization I (n.d.). https://www.cs.princeton.edu/courses/archive/spring16/cos495/slides/DL_lecture3_regularization_I.pdf. Accessed: 2023-02-03.

Dhaliwal, S. S., A.-A. Nahid, and R. Abbas (2018). "Effective intrusion detection system using xgboost". *Information* **9**:7. ISSN: 2078-2489. DOI: 10.3390/info9070149. URL: https://www.mdpi.com/2078-2489/9/7/149.

Electricity trade (n.d.). https://www.svk.se/en/national-grid/operations-and-electricity-markets/electricity-trade/. Accessed: 2023-02-08.

Elproduktion och förbrukning i Sverige (n.d.). https://www.scb.se/hitta-statistik/sverige-i-siffror/miljo/elektricitet-i-sverige/. Accessed: 2023-02-06.

Energy consumption of full electric vehicles (n.d.). https://ev-database.org/cheatsheet/energy-consumption-electric-car. Accessed: 2023-04-25.

Energy inflation rate continues upward hike, hits 27% (n.d.). https://ec.europa.eu/eurostat/web/products-eurostat-news/-/ddn-20220225-2. Accessed: 2023-03-29.

EUA Futures (n.d.). https://www.theice.com/products/197/EUA-Futures. Accessed: 2023-02-08.

EUROPEAN COMMISSION (13, 2021). *Tackling rising energy prices: a toolbox for action and support.* COMMUNICATION FROM THE COMMISSION TO THE EUROPEAN PARLIAMENT, THE EUROPEAN COUNCIL, THE COUNCIL, THE EUROPEAN ECONOMIC AND SOCIAL COMMITTEE AND THE COMMITTEE OF THE REGIONS COM(2021) 660 final. Version 2021-10-13. EUROPEAN COMMISSION, Brussels, Belgium.

EUROPEAN COMMISSION (8, 2022). *REPowerEU: Joint European Action for more affordable, secure and sustainable energy.* COMMUNICATION FROM THE COMMISSION TO THE EUROPEAN PARLIAMENT, THE EUROPEAN COUNCIL, THE COUNCIL, THE EUROPEAN ECONOMIC AND SOCIAL COMMITTEE AND THE COMMITTEE OF THE REGIONS COM(2022) 108 final. Version 2022-3-8. EUROPEAN COMMISSION, Strasbourg, France.

Exchange (n.d.). https://www.nordpoolgroup.com/en/Market-data1/Power-system-data/Exchange1/SE/Hourly1/?view=chart. Accessed: 2023-02-07.

Free Weather API (n.d.). https://open-meteo.com/. Accessed: 2023-03-15.

Friedman, J. H. (2002). "Stochastic gradient boosting". *Computational Statistics Data Analysis* **38**:4. Nonlinear Methods and Data Mining, pp. 367–378. ISSN: 0167-9473. DOI: https://doi.org/10.1016/S0167-9473(01)00065-2. URL: https://www.sciencedirect.com/science/article/pii/S0167947301000652.

Germany - Country Commercial Guide (n.d.). https://www.trade.gov/country-commercial-guides/germany-energy. Accessed: 2023-02-07.

Guestrin, C. (2021). *Lasso regression: regularization for feature selection.* http://cs229.stanford.edu/notes2021fall/lecture10-lasso-regression.pdf. Accessed: 2023-03-16.

Halbrügge, S., P. Schott, M. Weibelzahl, H. U. Buhl, G. Fridgen, and M. Schöpf (2021). "How did the german and other european electricity systems react to the covid-19 pandemic?" *Applied Energy* **285**, p. 116370. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2020.116370. URL: https://www.sciencedirect.com/science/article/pii/S0306261920317475.

Help with your energy bills (n.d.). https://www.gov.uk/get-help-energy-bills/getting-discount-energy-bill. Accessed: 2023-01-31.

Hjalmarsson, E. (2000). *Nord Pool : a power market without market power.* Working papers in economics / Department of Economics, School of Economics and Commercial Law, Göteborg University (Print): 28. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=cat07147a&AN=lub.1255544&site=eds-live&scope=site.

Högselius, P. and A. Kaijser (2010). "The politics of electricity deregulation in sweden: the art of acting on multiple arenas." *Energy Policy* **38**:5, pp. 2245–2254. ISSN: 0301-4215. URL: https://ludwig.

lub . lu . se / login ? url = https : / / search . ebscohost . com / login . aspx ? direct = true & AuthType = ip , uid & db = edselp & AN = S0301421509009501&site=eds-live&scope=site.

Horn, R. A. (1990). "The hadamard product". In: *Proc. Symp. Appl. Math.* Vol. 40, pp. 87–169.

How gradient boosting differs from gradient descent (n.d.). https :// explained . ai / gradient - boosting / descent . html # sec : 3 . 3. Accessed: 2023-03-29.

How Reliable Are Weather Forecasts? (N.d.). https://scijinks.gov/ forecast-reliability/#:~:text=The\%20Short\%20Answer\ %3A,right\%20about\%20half\%20the\%20time.. Accessed: 2023-03-20.

How to Use Random Seeds Effectively (n.d.). https : / / towardsdatascience . com / how - to - use - random - seeds - effectively-54a4cd855a79. Accessed: 2023-03-15.

Huisman, R., D. Michels, and S. Westgaard (2014). "Hydro reservoir levels and power price dynamics: empirical insight on the nonlinear influence of fuel and emission cost on nord pool day-ahead electricity prices". *The Journal of Energy and Development* **40**:1/2, pp. 149–187. ISSN: 03614476. URL: http : / / www . jstor . org / stable / 24813098 (visited on 2023-05-29).

Hyndman, R.J., Athanasopoulos, G. (2021) Forecasting: principles and practice, 3rd edition, OTexts: Melbourne, Australia. (N.d.). OTexts. com/fpp3. Accessed: 2023-03-08.

ICE Dutch TTF Natural Gas Futures - Apr 23 (TFMBMc1) (n.d.). https : / / www . investing . com / commodities / ice - dutch - ttf - gas-c1-futures-historical-data. Accessed: 2023-03-05.

Jääskeläinen, J., K. Huhta, and S. Syri (2022). "The anatomy of unaffordable electricity in northern europe in 2021". *Energies* **15**:20, p. 7504. ISSN: 1996-1073. DOI: 10 . 3390 / en15207504. URL: http : //dx.doi.org/10.3390/en15207504.

Jablstrok;onacute;ska, M., S. Viljainen, J. Partanen, and T. Kauranne (2012). "The impact of emissions trading on electricity spot market price behavior." *International Journal of Energy Sector Management* **6**:3, pp. 343–364. ISSN: 1750-6220. URL: https : / / ludwig .

lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edsemr&AN=edsemr.10.1108.17506221211259664&site=eds-live&scope=site.

Javier, C. (2017). *Forecasting Models of Electricity Prices. [Elektronisk resurs].* MDPI (Multidisciplinary Digital Publishing Institute. ISBN: 9783038424154. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=cat07147a&AN=lub.6176639&site=eds-live&scope=site.

Johanna, C. and TT (19, 2023). "Vattenfall misstänks för prismanipulation". *Svenska Dagbladet.* URL: %5Curl%7Bhttps://www.svd.se/a/zEXL5b/vattenfall-misstanks-for-prismanipulation%7D (visited on 2023-05-19).

Ketterer, J. C. (2014). "The impact of wind power generation on the electricity price in germany." *Energy Economics* **44**, pp. 270–280. ISSN: 0140-9883. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edselp&AN=S0140988314000875&site=eds-live&scope=site.

Khodadadi, A., L. Herre, P. Shinde, R. Eriksson, L. Soder, and M. Amelin (2020). "Nordic balancing markets: overview of market rules." *2020 17th International Conference on the European Energy Market (EEM), Energy Market (EEM), 2020 17th International Conference on the European*, pp. 1–6. ISSN: 978-1-7281-6919-4. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edseee&AN=edseee.9221992&site=eds-live&scope=site.

Kianfar, K. (2011). "Branch-and-bound algorithms". In: *Wiley Encyclopedia of Operations Research and Management Science.* John Wiley Sons, Ltd. ISBN: 9780470400531. DOI: https://doi.org/10.1002/9780470400531.eorms0116. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/9780470400531.eorms0116. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470400531.eorms0116.

Kuo, P.-H. and C.-J. Huang (2018). "An electricity price forecasting model by hybrid structured deep neural networks". *Sustainability* **10**:4, p. 1280. ISSN: 2071-1050. DOI: 10.3390/su10041280. URL: http://dx.doi.org/10.3390/su10041280.

Kuosmanen, T. (1.) and 3.) Johnson A.L. (2 (2021). "Conditional yardstick competition in energy regulation." *Energy Journal* **42**. ISSN: 01956574. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edselc&AN=edselc.2-52.0-85090479349&site=eds-live&scope=site.

Kusakana, K. (2015). "Operation cost minimization of photovoltaic–diesel–battery hybrid systems". *Energy* **85**, pp. 645–653. ISSN: 0360-5442. DOI: https://doi.org/10.1016/j.energy.2015.04.002. URL: https://www.sciencedirect.com/science/article/pii/S0360544215004181.

Lehr, L. and F. Valdes (2021). "Forecasting electricity prices from european single day-ahead coupling using artificial neural networks." *2021 IEEE Electrical Power and Energy Conference (EPEC), Electrical Power and Energy Conference (EPEC), 2021 IEEE*, pp. 202–207. ISSN: 978-1-6654-2928-3. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edseee&AN=edseee.9621718&site=eds-live&scope=site.

Lorenczik, S. and J. J. Copier (2022). "Electricity market report - july 2022", p. 62. URL: https://iea.blob.core.windows.net/assets/660c2410-218c-4145-9348-c782e185dcdf/ElectricityMarketReport-July2022.pdf.

Lundin, E. (2021). "Market power and joint ownership: evidence from nuclear plants in sweden*." *Journal of Industrial Economics* **69**:3, pp. 485-536 –536. ISSN: 14676451. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edselc&AN=edselc.2-52.0-85120794977&site=eds-live&scope=site.

Maciejowska, K. (2020). "Assessing the impact of renewable energy sources on the electricity price level and variability – a quantile regression approach." *Energy Economics* **85**. ISSN: 0140-9883. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edselp&AN=S0140988319303275&site=eds-live&scope=site.

Market surveillance (n.d.). https://www.nordpoolgroup.com/en/trading/Market-surveillance/. Accessed: 2023-04-22.

MATH6011: Forecasting (n.d.). https://www.southampton.ac.uk/~abz1e14/papers/Forecasting.pdf. Accessed: 2023-03-10.

Mulder, M. and B. Scholtens (2013). "The impact of renewable energy on electricity prices in the netherlands". *Renewable Energy* **57**, pp. 94–100. ISSN: 0960-1481. DOI: https://doi.org/10.1016/j.renene.2013.01.025. URL: https://www.sciencedirect.com/science/article/pii/S0960148113000505.

Multiple Outputs (n.d.). https://xgboost.readthedocs.io/en/stable/tutorials/multioutput.html. Accessed: 2023-03-23.

Mussa, A. S., M. Klett, M. Behm, G. Lindbergh, and R. W. Lindström (2017). "Fast-charging to a partial state of charge in lithium-ion batteries: a comparative ageing study". *Journal of Energy Storage* **13**, pp. 325–333. ISSN: 2352-152X. DOI: https://doi.org/10.1016/j.est.2017.07.004. URL: https://www.sciencedirect.com/science/article/pii/S2352152X17301913.

Niclas Damsgaard, R. G. (2005). "Regulatory reform in the swedish electricity industry – good or bad?" URL: https://wwwsnsse.cdn.triggerfish.cloud/uploads/2005/09/sos_elmarknad_2005_eng.pdf.

"Nord Stream 1: How Russia is cutting gas supplies to Europe" (29, 2022). *BBC*. URL: https://www.bbc.com/news/world-europe-60131520 (visited on 2022-09-29).

Operations and Electricity Markets (n.d.). https://www.svk.se/en/national-grid/operations-and-electricity-markets/. Accessed: 2023-02-07.

Optimal Dispatch of Power Generators: Problem-Based (n.d.). https://se.mathworks.com/help/optim/ug/optimal-dispatch-problem-based.html. Accessed: 2023-04-25.

Optimization in ML (n.d.). https://www.cs.cmu.edu/~10606/gradient.pdf. Accessed: 2023-02-01.

Parameters and Hyperparameters in Machine Learning and Deep Learning (n.d.). https://towardsdatascience.com/parameters-and-hyperparameters-aa609601a9ac#:~:text=Hyperparameters\%20are\%20parameters\%20whose\%20values,parameters\%20that\%20result\%20from\%20it.. Accessed: 2023-02-22.

Playing my part (n.d.). https://www.iea.org/reports/playing-my-part?utm_content=buffer9b1c6&utm_medium=social&utm_source=twitter-ieabirol&utm_campaign=buffer. Accessed: 2023-01-31.

Python API Reference (n.d.). https://xgboost.readthedocs.io/en/stable/python/python_api.html. Accessed: 2023-02-21.

Renewable energy statistics (n.d.). https://ec.europa.eu/eurostat/statistics-explained/SEPDF/cache/7177.pdf. Accessed: 2023-03-07.

Roy, A., R. B. Patil, and R. Sen (2022). "The effect of fast charging and equalization on the reliability and cycle life of the lead acid batteries". *Journal of Energy Storage* **55**, p. 105841. ISSN: 2352-152X. DOI: https://doi.org/10.1016/j.est.2022.105841. URL: https://www.sciencedirect.com/science/article/pii/S2352152X22018291.

Ruder, S. (2016). "An overview of gradient descent optimization algorithms". *CoRR* **abs/1609.04747**. arXiv: 1609.04747. URL: http://arxiv.org/abs/1609.04747.

Schöne, S. (2009). *Auctions in the Electricity Market. Bidding when Production Capacity Is Constrained.* Lecture Notes in Economics and Mathematical Systems: 617. Springer Berlin Heidelberg. ISBN: 9783540853640. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=cat02271a&AN=atoz.ebs470233e&site=eds-live&scope=site.

Setting the power price: the merit order effect (n.d.). https://www.cleanenergywire.org/factsheets/setting-power-price-merit-order-effect. Accessed: 2023-04-16.

Sheng, C. and H. Yu (2022). "An optimized prediction algorithm based on xgboost." *2022 International Conference on Networking and Network Applications (NaNA), Networking and Network Applications (NaNA), 2022 International Conference on, NANA*, pp. 1–6. ISSN: 978-1-6654-6131-3. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edseee&AN=edseee.9985004&site=eds-live&scope=site.

Siami-Namini, S., N. Tavakoli, and A. Siami Namin (2018). "A comparison of arima and lstm in forecasting time series". In: *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 1394–1401. DOI: 10.1109/ICMLA.2018.00227.

Singhal, D. and K. Swarup (2011). "Electricity price forecasting using artificial neural networks". *International Journal of Electrical Power Energy Systems* **33**:3, pp. 550–555. ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2010.12.009. URL: https://www.sciencedirect.com/science/article/pii/S0142061510002231.

Staff, R. (19, 2022a). "Eu agrees gas price cap with 180 eur/mwh trigger - document". *Reuters.* URL: https://www.reuters.com/article/ukraine-crisis-eu-gas-price-idUSP6N31E00X (visited on 2022-12-19).

Staff, R. (19, 2022b). "Nord stream 1 pipeline to shut for three days in latest fuel blow to europe". *Reuters.* URL: https://www.reuters.com/business/energy/gazprom-says-nord-stream-1-pipeline-shut-three-days-end-aug-2022-08-19/ (visited on 2022-08-19).

Statquest (2019). *Gradient boost part 1 (of 4): regression main ideas.* https://www.youtube.com/watch?v=3CC4N4z3GJc.

Stéphane, A., C. Vincenzo, and R. Benoît (2019). "Price elasticity of electricity demand in france." *Economie et Statistique / Economics and Statistics* **513**:1, pp. 91–103. ISSN: 0336-1454. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edsper&AN=edsper.estat.0336.1454.2019.num.513.1.10921&site=eds-live&scope=site.

Supervised Learning (n.d.). https://www.ibm.com/se-en/topics/supervised-learning. Accessed: 2023-03-15.

Tamura, S. and T. Kikuchi (2018). "V2g strategy for frequency regulation based on economic evaluation considering ev battery longevity". In: *2018 IEEE International Telecommunications Energy Conference (INTELEC)*, pp. 1–6. DOI: 10.1109/INTLEC.2018.8612431.

Target Variable (n.d.). https://h2o.ai/wiki/target-variable/#:~:text=The\%20target\%20variable\%20is\%20the\%20variable\%20whose\%20values\%20are\%20modeled,value\%20of\%20the\%20target\%20variable.. Accessed: 2023-03-28.

Taslim, D. G. and I. M. Murwantara (2022). "A comparative study of arima and lstm in forecasting time series data." *2022 9th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), Information Technology, Computer, and Electrical Engineering (ICITACEE), 2022 9th International Conference on*, pp. 231–235. ISSN: 978-1-6654-7150-3. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edseee&AN=edseee.9924148&site=eds-live&scope=site.

The electricity market (n.d.). https://ei.se/ei-in-english/electricity/the-electricity-market. Accessed: 2023-02-05.

Tian, Y., L. Si, X. Zhang, K. C. Tan, and Y. Jin (2023). "Local model-based pareto front estimation for multiobjective optimization". *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **53**:1, pp. 623–634. DOI: 10.1109/TSMC.2022.3186546.

Training and Test Sets: Splitting Data (n.d.). https://developers.google.com/machine-learning/crash-course/training-and-test-sets/splitting-data. Accessed: 2023-03-15.

Tribioli, L. and G. Bella (2022). "10 - automotive hybrid electric systems: design, modeling, and energy management". In: Lo Faro, M. et al. (Eds.). *Hybrid Technologies for Power Generation*. Hybrid Energy Systems. Academic Press, pp. 279–312. ISBN: 978-0-12-823793-9. DOI: https://doi.org/10.1016/B978-0-12-823793-9.00010-3. URL: https://www.sciencedirect.com/science/article/pii/B9780128237939000103.

Tušar, T. and B. Filipič (2015). "Visualization of pareto front approximations in evolutionary multiobjective optimization: a critical review and the prosection method". *IEEE Transactions on Evolutionary Computation* **19**:2, pp. 225–245. DOI: 10.1109/TEVC.2014.2313407.

Uddin, G. S., O. Tang, M. Sahamkhadam, F. Taghizadeh-Hesary, M. Yahya, P. Cerin, and J. Rehmea (2022). *Analysis of Forecasting Models in Electricity Market Under Volatility: What We Learn from Sweden.* Springer Nature Singapore. ISBN: 978-981-19426-5-5. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edssjb&AN=edssjb.978.981.19.4266.2.5&site=eds-live&scope=site.

"Ukraine war: How Germany ended reliance on Russian gas" (23, 2022). *BBC*. URL: https://www.bbc.com/news/world-europe-63709352 (visited on 2022-11-23).

Uribe, J. M., S. Mosquera-López, and O. J. Arenas (2022). "Assessing the relationship between electricity and natural gas prices in european markets in times of distress." *Energy Policy* **166**. ISSN: 0301-4215. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edselp&AN=S0301421522002439&site=eds-live&scope=site.

Wangensteen, I. (2007). *Power System Economics: The Nordic Electricity Market*. Tapir Academic Press. ISBN: 9788251922005. URL: https://books.google.se/books?id=pSdkAgAACAAJ.

Weather Forecast API (n.d.). https://open-meteo.com/en/docs. Accessed: 2023-04-15.

Weron, R. (2014). "Electricity price forecasting: a review of the state-of-the-art with a look into the future". *International Journal of Forecasting* **30**:4, pp. 1030–1081. ISSN: 0169-2070. DOI: https://doi.org/10.1016/j.ijforecast.2014.08.008. URL: https://www.sciencedirect.com/science/article/pii/S0169207014001083.

Werth, A., P. Gravino, and G. Prevedello (2021). "Impact analysis of covid-19 responses on energy grid dynamics in europe." *Ap-*

plied Energy **281**. ISSN: 0306-2619. URL: https://ludwig.lub.lu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,uid&db=edselp&AN=S030626192031480X&site=eds-live&scope=site.

What is emissions trading? (N.d.). https://www.emissionsauthority.nl/topics/what-is-emissions-trading. Accessed: 2023-04-27.

What is gradient descent? (N.d.). https://www.ibm.com/topics/gradient-descent. Accessed: 2023-02-01.

What is regression? (N.d.). https://h2o.ai/wiki/regression/. Accessed: 2023-05-16.

What is regression? (N.d.). https://se.mathworks.com/help/optim/ug/optimproblem.html. Accessed: 2023-05-20.

Wu, K., Y. Chai, X. Zhang, and X. Zhao (2022). "Research on power price forecasting based on pso-xgboost". *Electronics* **11**:22, p. 3763. ISSN: 2079-9292. DOI: 10.3390/electronics11223763. URL: http://dx.doi.org/10.3390/electronics11223763.

XGBoost (n.d.). https://www.nvidia.com/en-us/glossary/data-science/xgboost/. Accessed: 2023-03-14.

XGBoost Documentation (n.d.). https://xgboost.readthedocs.io/en/stable/. Accessed: 2023-02-03.

XGBoost versus Random Forest (n.d.). https://www.qwak.com/post/xgboost-versus-random-forest#:~:text=A\%20model\%20whose\%20parameters\%20adjust,entire\%20ensemble\%20(Random\%20Forest).&text=Working\%20With\%20Unbalanced\%20Data\%20\%E2\%80\%93\%20The,we\%20have\%20a\%20class\%20imbalance.. Accessed: 2023-03-14.

Zang, Z., Z. Li, Q. Zhu, F. Chang, W. Xu, and B. Lv (2019). "Application of arima and markov combination model in medium and long term electricity forecasting". In: *2019 IEEE 3rd International Electrical and Energy Conference (CIEEC)*, pp. 287–292. DOI: 10.1109/CIEEC47146.2019.CIEEC-2019144.

Lund University
Department of Automatic Control
Box 118
SE-221 00 Lund Sweden

Document name
MASTER'S THESIS

Date of issue
June 2023

Document Number
TFRT-6208

Author(s)
Oscar Andreas Olsson

Supervisor
Emma Tegling, Dept. of Automatic Control, Lund University, Sweden
Richard Pates, Dept. of Automatic Control, Lund University, Sweden (examiner)

Title and subtitle
Decision support in a volatile electricity market: forecasting and cost optimization

Abstract

Given the increase in electricity prices in recent years due to two reasons; the rebound effect after the initial corona outbreak and the Russian invasion of Ukraine, the burden of paying the rising monthly expense for electricity has become an unwelcome reality for a significant part of society. The electricity trades on the open market called Nord Pool for the Nordic countries, among others, where buyers and sellers come together to find a market price for electricity each day. By forecasting future electricity prices using the machine-learning model XGBoost and a select number of features, the use of electricity can be optimized in the near future with respect to cost. Two different XGBoost models were constructed and evaluated on their ability to predict future prices. Each model was trained on a unique dataset, where the datasets are of different characteristics in terms of volatility. The first model, trained on historical electricity prices with less volatility showed a much more reliable forecasting ability than the second model, trained on historical electricity prices with much more inherent volatility. The optimizations were executed in Matlab with two different optimization solvers. The cost-optimization with the forecasted electricity prices is compared to other charging patterns, in order to determine if the forecasted prices are accurate enough to save cost. Each optimization problem had a number of defined objectives and a number of constraints assigned to it. The result of this thesis showed that the charging protocol incorporating the forecasted electricity prices while minimizing the cost produced a more cost-efficient solution in comparison to the other charging protocols brought up.

Keywords

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title
0280-5316

ISBN

Language
English

Number of pages
1-155

Security classification

Recipient's notes

http://www.control.lth.se/publications/