ARCADA

# Forecasting of Bidding Strategies for Participation in Finnish Electricity Balancing Market

## A Case Study of Electric Vehicle Charging As Power Reserve

Bhuwan Karki

Master's Thesis

MEng in Big Data Analytics

2023

## Master's Thesis

Bhuwan Karki

Forecasting of Bidding Strategies for Participation in Finnish Electricity Balancing Market . A Case Study of Electric Vehicle Charging As Power Reserve.

Arcada University of Applied Sciences: MEng in Big Data Analytics, 2023.

## Commissioned by:

Virta Oy

## Abstract:

Electric Vehicle (EV) sales and popularity are increasing because it is a sustainable and environmentally friendly way of transportation. EVs reduce fossil fuel dependency and reduce (carbon dioxide) CO2 emissions. Due to its random and volatile nature, increasing demand for charging burdens the electric grid. EVs can be considered energy storage with available energy and capacity constraints that can provide flexibility to the power system by balancing products when aggregated. Electric vehicle charging stations can participate in frequency reserves markets by curtailing the charging current during under frequency. Frequency Containment Reserve in Disturbance (FCR-D) is a frequency-based control logic that aims to adjust the frequency of the electric system by activating reserves. The station operator's readiness to respond to under-frequency situations determines their compensation. This compensation incentivizes the operator to stay alert and take immediate action if necessary, ensuring an uninterrupted power supply. Recognizing the crucial role that the station operator plays in maintaining a reliable power grid, their compensation reflects the value of their expertise and dedication. Forecasts of charging patterns are essential to reliably providing this ancillary service. This paper aims to build a generalized model for forecasting EV charging demand and the FCR-D upward price. Forecasting demand capacity and FCR-D upward price is typically done

using traditional and deep learning approaches based on time series analysis. We use two distinct datasets to forecast demand and FCR-D upward price using both univariate and multivariate methods. The forecasting horizon is typically one month. Different forecasting algorithms include MSTL, ETS, NHiTS, TiDE, and Dlinear. MSE, MAE, RMSE, and SMAPE were used to compare model performance. According to the result, the NHITS outperforms other models in terms of long-term forecasting for univariate with the EV charging demand data and Tide for multivariate. Similarly, for forecasting the FCR-D upward Price, NHiTS for univariate and TiDE for multivariate outperform the other models.

**Keywords:** Time Series Forecasting, Demand response, electric vehicle, Balancing Services, Frequency Containment Reserve in Disturbance, Sustainability, Green transition, Bidding Strategies

# Contents

# Figures

# Tables

## Abbreviations

| | |
|---|---|
| AI | Artificial Intelligence |
| DSO | Distribution System Operator |
| DR | Demand Response |
| ENTSO-E | European Network of Transmission System Operator |
| FCP | Frequency Containment Process |
| FCR | Frequency Containment Reserve |
| FCR-D | Frequency Containment Reserve in Disturbance |
| FCR-N | Frequency containment Reserve in Normal Operation |
| HVDC | High-Voltage Direct Current |
| EVSE | Electric Vehicle Supply Equipment |
| OCCP | Open Charge Point Protocol |
| KW | Kilowatt |
| MW | Megawatt |
| GW | Gigawatt |
| TSO | Transmission System Operator |
| EV | Electric Vehicle |
| DLM | Dynamic Load Management |
| ALM | Adaptive Load Management |
| V2G | Vehicle to Grid |
| ICE | Internal Combustion Engine |
| Oyjs | Julkinen Osakeyhtiö (Public Limited Company) |
| EDA | Exploratory Data Analysis |
| TBATS | Trigonometric Seasonality, Box-Cox Transformation, ARIMA Errors, Trend and Seasonal Components |
| ARIMA | Autoregressive Integrated Moving Average |
| SARIMA | Seasonal Autoregressive Integrated Moving Average |
| LSTM | Long Short-Term Memory |
| RNN | Recurrent Neural Network |
| GRU | Gated Recurrent Unit |
| SVR | Support Vector Regression |

| | |
|---|---|
| ANN | Artificial Neural Network |
| HMM | Hidden Markov Models |
| GBTD | Gradient Boosted Decision Tree |
| STR | Seasonal-Trend Decomposition using Regression |
| MSTL | Multiple Seasonal-Trend Decomposition using Loess |
| STL | Seasonal-Trend Decomposition using Loess |
| ETS | Exponential Trend Smoothing |
| TIDE | Time-Series Dense Encoder |
| NHITS | Neural Hierarchical Interpolation for Time Series |
| MLP | Multilayer Perceptron |
| Dlinear | Decomposition Linear |
| TFT | Temporal Fusion Transformer |
| PatchTST | Patch Time Series Transformer |
| ReLU | Rectified Linear Unit |
| $CO_2$ | Carbon Dioxide |
| AIC | Akaike's Information Criterion |
| BIC | Bayesian Information Criterion |
| LOESS | Locally Estimated Scatterplot Smoothing |
| AAAI | Association for The Advancement of Artificial Intelligence |
| LTSF | Long-Term Time Series Forecasting |
| DMS | Direct Multi-Step |
| FEDFormer | Frequency Enhanced Decomposed Transformer |
| FMI | Finnish Meteorological Institute |
| SMA | Simple Moving Average |
| EMA | Exponential Moving Average |
| ADF | Augmented Dickey-Fuller |
| KPSS | Kwiatkowski-Phillips-Schmidt-Shin |
| ACF | Autocorrelation Function |
| PACF | Partial Autocorrelation Function |
| AC | Alternating Current |
| DC | Direct Current |

| | |
|---|---|
| TPE | Tree-Structured Parzen Estimator |
| Hz | Hertz |
| API | Application Programming Interface |
| MAE | Mean Absolute Error |
| MSE | Mean Squared Error |
| RMSE | Root Mean Squared Error |
| MAPE | Mean Absolute Percentage Error |
| SMAPE | Symmetric Mean Absolute Percentage Error |
| MASE | Mean Absolute Scaled Error |

## Foreword

This thesis constitutes a culmination of my academic pursuit toward a Master's degree in Big Data Analytics at Arcada University of Applied Sciences in Helsinki, Finland. The central focus of the thesis revolves around forecasting Electric Vehicle (EV) charging demand and the associated Frequency Containment Reserve (FCR-D) pricing for optimal bidding strategies within the aggregator market, leveraging historical datasets. My work manager, Dr. Juha Karppinen, Director of Energy Services, provided the specific topic.

I am deeply indebted to many individuals whose support and guidance have been instrumental in pursuing this scholarly endeavor. First, I sincerely thank my family for their unwavering encouragement. Furthermore, I sincerely appreciate my academic supervisor's invaluable insights, knowledgeable guidance, and constant support throughout the thesis development process. Lastly, I thank my managerial mentor for instigating this academic pursuit and for their ongoing support and encouragement.

Completing this thesis is a testament to the collective efforts of these individuals. I acknowledge their contributions to this scholarly endeavor with profound gratitude.

# 1. Introduction

## 1.1 Background

Efficient transportation is an indispensable element of economic development and a substantial contributor to energy consumption. It consumes an enormous amount of fossil fuel, one of the contributors to the $CO_2$ gases. Transportation emissions have almost tripled due to growing populations and economies. The transportation sector accounts for 20% of global emissions and is the second-largest carbon-polluting sector worldwide (Statista 2023).

Recently, with the advancement of battery technologies for storing a huge amount of energy, electric vehicles have become a preferred transportation choice because they are low-carbon and environmentally friendly. In Finland in 2023, there were about 84,000 EV passenger cars compared to last year, 2022, which was about 45,000. (Tiedotuskeskus 2024) . EV transportation will likely replace Internal Combustion Engine (ICE) cars shortly. The widespread adoption of electric vehicles has raised concerns about the potential impact on the electric grid due to the variability and unpredictability of EV charging loads. The power grid will be affected by peak power, frequency, and voltage variation. It is important to accurately estimate the demand for electric vehicle charging to predict how additional demand will affect the electricity grid. Forecasting EV charging helps maintain grid stability and resilience while assisting with long-term investment planning and resource allocation for charging infrastructure. Successfully forecasting EV charging demand is crucial in preserving the utility of the grid.

The peak hours are usually the busiest with the charging demand and less peak at night. With peak during the day, the EV fleet's uncontrolled charge may threaten the power outage with less infrastructure. As EV consumption increases, the prediction of the aggregated EV demand becomes more critical for the Transmission System Operator (TSO). So flexibility has become crucial to balance demand and production at every moment. A grid is healthy when the frequency does not deviate from an equilibrium value of 50 HZ in Europe.

Ancillary services are operations and services that guarantee the stability of the power

grid by maintaining the balance between demand and electricity supply at all times. Various assets such as power plants, batteries, and industrial sites can provide these services, adapting to production or consumption needs. EV charging offers a significant and rapidly growing opportunity to help power systems cope with disturbances. Hence, The rapidly expanding charging network for electric vehicles can be harnessed as a disturbance reserve to support the power system (Fingrid 2023).

Due to increased pressure on the electrical grid and fluctuations from renewable energy, demand for related services will increase, leading to higher prices. Forecasting the power consumption of EV charging stations is crucial and has many interests. Long-term forecasts allow us to anticipate the penetration of electric vehicles and estimate the need for new charging stations. In contrast, short- and mid-term forecasts will enable us to implement charging strategies or cost optimizations and provide ancillary services like Frequency Containment Reserve (FCR).

### 1.1.1 Objective

The primary goal of this thesis is to develop a forecasting model to predict:

1. The hourly demand (in MW) for a group of EV chargers over a month.

2. The hourly bid price for FCR-D upward over a month.

We use the forecasted demand to place a bid in the FCR-D upward market. We must not under-deliver the bidded capacity for the next day, as the penalty for doing so is three times the bid price of under-delivered capacity. Additionally, the thesis investigates how EVs can contribute to the energy sector's flexibility. One can bid in the aggregator market based on the demand for EV charging. The main objective of forecasting the bid price is determining the total revenue in the upcoming period. The thesis also aims to assess the effectiveness of the conventional statistical model and compare it with the complex deep learning algorithm to enhance forecasting.

### 1.1.2 Research Question

The primary research questions for the thesis are:

1. Can state-of-the-art deep learning approaches accurately forecast EV charging demand and FCR-D upward bid price for a month-long horizon?

2. What are the advantages of univariate and multivariate forecasting approaches, and how do different features contribute to forecasting accuracy?

3. Additionally, does the applied model reliably inform bidding strategies to minimize mistakes, given the penalty of three times the cost of the bid price for under-delivery during the hour?

### 1.1.3 Limitation

The energy market is very changing and volatile, and the bid price can be affected by an uncountable number of external factors, some of which may be :

- Predictable factor: spot price and weather data predict to some extent.

- Unpredictable factors: power disruption, power line failure, socio-economic and geo-political, historical, and environmental.

- Lack of access to data regarding confounding variables such as maintenance breaks and other unforeseen events.

The energy market data is erratic, and forecasting the market with such volatility is difficult when the forecast horizon is a month.

## 1.2 Commissioning Company - Virta

Virta is one the global leaders in EV charging technologies and the fastest growing EV charging platform in Europe, founded in 2013. For the fourth time in 2023, The Financial Times has recognized Virta as one of Europe's Fastest Growing Companies.(Virta 2023a). The company aims to solve climate change and move toward sustainability by bringing electric mobility to businesses, people, and the energy system with the world's most cost-efficient and reliable electric vehicle charging platform.

### 1.2.1 Virta Services

Virta provides businesses with an end-to-end charging solution that includes various services: (Virta 2023c). Of the different services, the one that interests the thesis the most is related to the energy market for FCR-D upward and demand response.

### 1.2.2 Virta Current Operation

Over 1,000 private and public entities across various industries, including retail, hotel, real estate, parking, petrol, automotive, and energy, actively use the virta charging platform. It has 34 patent families focusing on energy management capabilities, central to the future connected energy and mobility ecosystem (Virta 2023a). Virta operates in over 35 countries and has access to more than 460,000 charging points through roaming. The Virta platform delivers a new EV charging event every 2 seconds. In F land, virta has 137,000 EV drivers; roughly 33000 EV drivers are active monthly in Finland, where the total energy utilized is 3.1 GigaWatt (GW) monthly. 1950 charging events every month and almost 6000 newly registered drivers where we have 12000 operational EVSE in Finland

Among the various topics in Virta operation, demand response is a service that can automatically respond to rapid and unexpected disturbances in the power grid and help balance the grid's health. Virta is already a registered independent demand response aggregator to the Finnish transmission system operator Fingrid Oyj (Virta 2023b). On November 19, the Olkiluoto 3 nuclear power plant unexpectedly disconnected from the grid due to a turbine fault, activating this response in Finland. The charging power of charging events on Alternating Current (AC) chargers around Finland was reduced within 5 seconds for one minute due to the observed dip in grid frequency (Virta 2023b).

# 2. Finnish Energy Market

In Finland, electricity customers, including households and businesses, purchase their electricity from suppliers in an open market that has been operational for over 20 years. This market encompasses gen rating, transmitting, distributing, and selling electricity. Notably, electricity import from January to June 2023 decreased by 58%, amounting to 4.6 TeraWattHour (TWh), while exports increased by 6%, totaling 3.8 TWh compared to

the previous year (Statistics Finland 2023). In 2023, energy consumption was approximately 76.93 TW, with a total net import of 2.93 TW (LowCarbonPower 2024). Finland's electricity trading occurs on two platforms: Nord Pool Spot, the primary market for hourly electricity trading within the Nordic wholesale market (Ministry of foreign affair 2023), and the Finnish TSO Fingrid Oyj's ancillary service markets, where reserve product trading takes place through yearly and hourly contracts.

## 2.1 Day-Ahead Elspot

The Elspot in the day is the primary electricity market in the Nordic and Baltic regions, where most of the trading volume occurs. Energy traders submit bids to buy and sell energy in hourly locks for the following day. Marginal pricing determines the price and volume of energy by calculating the intersection of the cumulative demand and supply curves. Bidding for electricity sale or purchase should be placed for the next day before 13.00 Eastern European Time (EET ) (Nordpool 2023).

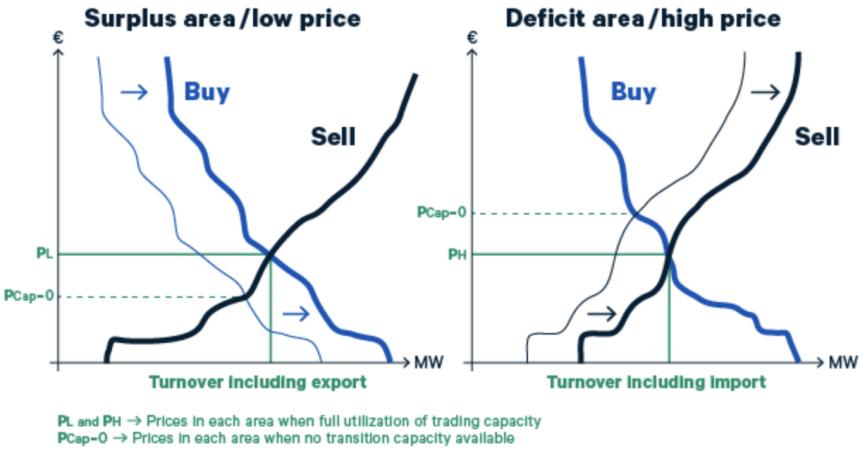



*Figure 1. Spot Price Cumulative Curve (Nordpool 2023)*

Figure (1) shows that the selling curve represents aggregate supply, and the buying curve represents aggregate demand. There is not always sufficient electricity transmission capacity, as production and consumption rarely occur in the network.

## 2.2 Intraday

The intraday market, exemplified by Nord Pool's Elbas market, facilitates trading based on consumption or production volumes calculated from the day-ahead market. Elbas operates as a balancing market for the primary Elspot (NordPool 2023), allowing traders to

adjust their transactions after the Elspot market closes. It works on a pay-as-bid principle, where the price for a specific hour's product may vary among participants.

## 2.3 Ancillary Service Market

Ancillary services are crucial functions that help grid operators maintain a reliable electrical system. They ensure the smooth flow and direction of electricity, address imbalances between supply and demand, and aid the recovery of the system after power system events (Greening The Grid 2023). In Finland and the Nordic power system, the nominal frequency of the system is 50 Hz, but fluctuations in power consumption and production are constant. Fingrid Oyj, responsible for Finland's electricity transmission grid, ensures system security and power balance. To fulfill its responsibilities, Fingrid has established marketplaces for essential ancillary services.

## 2.4 Frequency Containment Reserve Marketplaces

Finland's power system comprises power plants, the primary grid, high-voltage and other distribution networks, and consumers. Finland's power system connects with Estonia through HVDC transmission links and forms an integral part of the Nordic synchronous area, which includes Sweden, Norway, and Eastern Denmark. High-voltage direct Current (HVDC) transmission links connect the Nordic system with Central Europe. (Fingrid 2023a). Fingrid, Finland's TSO, participates in the European Network of Transmission System Operators (ENTSO-E). Most CR capacity is purchased domestically, with separate markets for FCR-N and FCR-D. TSO offers contracts to customers for automatic consumption adjustment in response to grid frequency fluctuations for frequency containment.

### 2.4.1 Frequency Containment Reserve for Normal Operation (FCR-N)

FCR-N operates within the frequency range of 49.9 to 50.1 Hz in the power system, autonomously reacting to deviations. Its activation does not rely on a control signal from the TSO, as frequency is measured on-site. FCR-N runs continuously, maintaining a frequency within the "normal operating band" of 50Hz ±0.1Hz. Activation is proportional to the frequency deviation from 50Hz, increasing linearly. Activation increases as frequency deviates from 50Hz until fully engaged at ±0.1Hz. Within 60 seconds, 63% of reserves should be activated, with full activation within 180 seconds if needed (Fingrid

2023b).

### 2.4.2 Frequency Containment Reserve for Disturbed Operation (FCR-D)

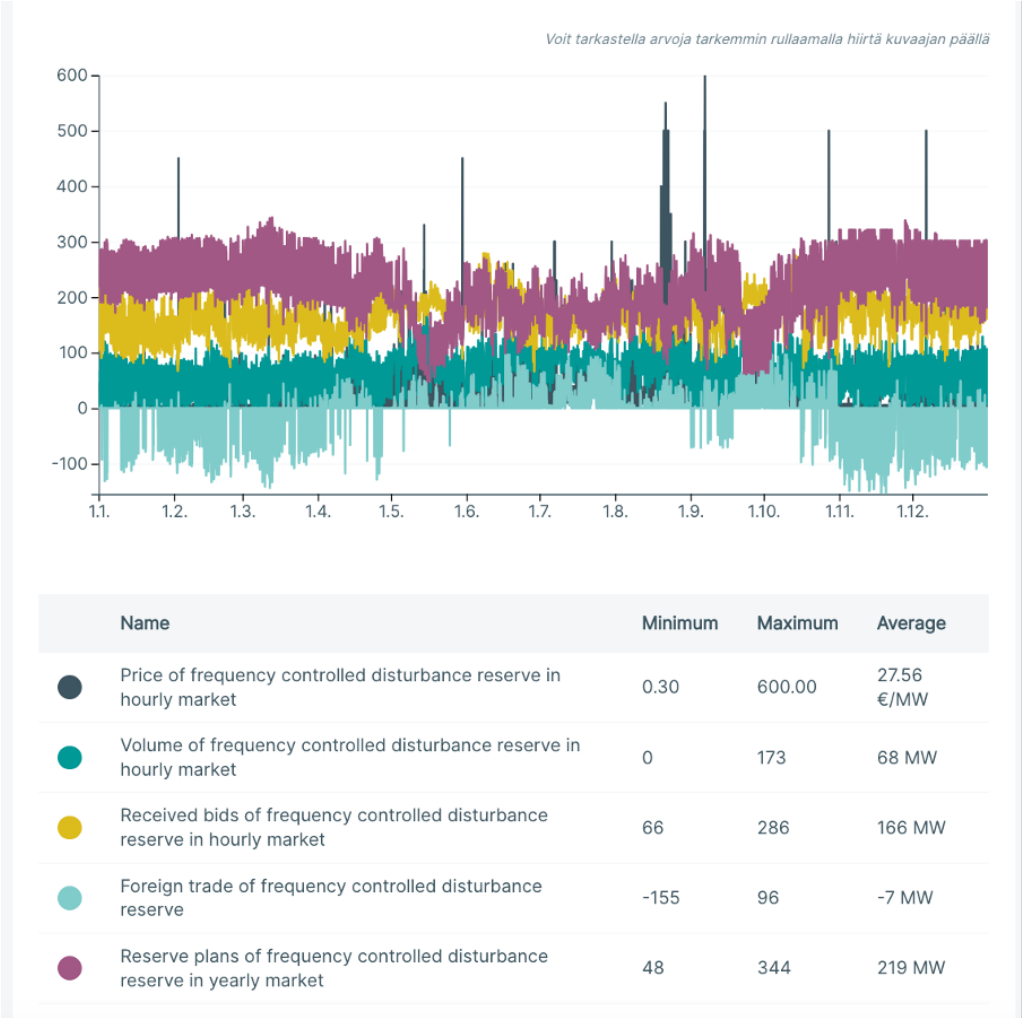
When the frequency drops below 49.9Hz, FCR-D automatically activates and must be fully activated at 49.5Hz. FCR-D is required to react faster, requiring 50% to be activated within 5 seconds and 100% to be made available within 100 seconds. In the annual market, the price remains constant throughout the calendar year. All market participants receive the same compensation for maintaining reserve capacity (Fingrid 2024). FCR-D products are reimbursed based on energy and capacity. In 2023, the annual market's capacity fee was 2.81 €/MW/h, with a volume of 345.1 MW. In 2024, the procurement volume remains almost unchanged, with a capacity of 347.8 MW. (Fingrid 2024). The reserve suppliers participating in the annual market confirm the reserve capacity they maintain daily for the following day. In addition to the yearly market, Fingrid acquires frequency stabilization reserve capacity through daily acquisition from the hourly market in Finland. The most expensive activated bid sets the upregulation price for a given hour, while the auction outcome determines the upregulation price and volume. The procurement for FCR-D upward is carried out, minimizing total costs, but in such a way that the entire procurement covers the reserve obligations of Fingrid (Fingrid 2024).

Figure (2) for the Hourly market shows the price, volume, bids, foreign trade, and reserve plan during 2023. The average price was 27.56 €/MW, and the maximum was 600€ MW. The volume is the reserve providers' free capacity; on average, it was 68 MW and a maximum of 173 MW.

There are two main types of reserves:

- Reserves aiming to increase production (e.g., auxiliary power plants)
- Reserves for cutting down the demand (e.g., big factories ) Its purpose is to contain the frequency during regular operation and disturbances.

Based on the technical specifications of Finnish TSO Fingrid, the system activates reserves at different frequency steps. (Fingrid 2023d chap. 3.1.2).




| Name | Minimum | Maximum | Average |
|---|---|---|---|
| Price of frequency controlled disturbance reserve in hourly market | 0.30 | 600.00 | 27.56 €/MW |
| Volume of frequency controlled disturbance reserve in hourly market | 0 | 173 | 68 MW |
| Received bids of frequency controlled disturbance reserve in hourly market | 66 | 286 | 166 MW |
| Foreign trade of frequency controlled disturbance reserve | -155 | 96 | -7 MW |
| Reserve plans of frequency controlled disturbance reserve in yearly market | 48 | 344 | 219 MW |

*Figure 2. Transactions on the hourly market, FCR-D upward (Fingrid 2024)*

### 2.4.3 Base Logic for Virta FCR-D Upward

The Virta network has implemented FCR-D upward on a specific group of charging points. This feature has no limit to the number of connections it can handle. However, it only operates with stations that fulfill particular criteria and have undergone appropriate testing. Once activated, the FCR-D upward backend actively monitors the frequency. Suppose the frequency drops below a certain threshold. In that case, it recalculates the power limits for each active charging point, decreasing the group's total charging power and ensuring that the charging points operate optimally even when the frequency drops below the acceptable level.

Virta's FCR-D upward adjustment logic for the charging point group is based on Fingrid's technical requirements, as shown in Figure (3). (Fingrid 2023d chap. 3.1.2).

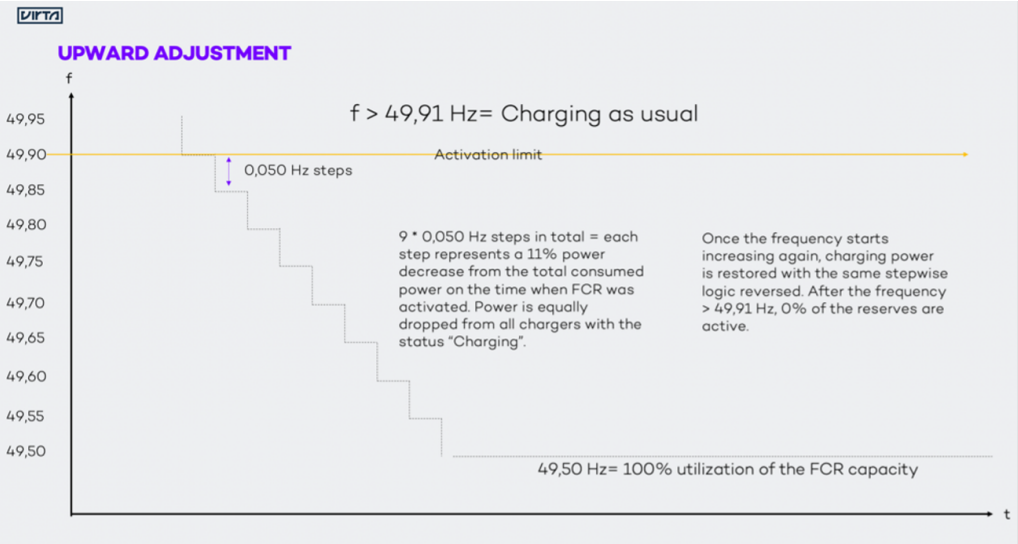



*Figure 3. FCR-D upward adjustment*

The power reduction occurs proportionally for each station based on the reported meter values. The adjustment occurs in multiple steps, depending on the frequency level. The FCR-D upward event begins and ends when the frequency crosses predefined thresholds. Throughout the event, charging points respond to frequency fluctuations, reverting to their original settings once the event concludes. The aim is to support the grid while minimizing disruption to EV drivers by avoiding unnecessary reserve activation. The steps that were previously mentioned have been modified as per the instructions provided in Table 1.

*Table 1. Step adjustment for FCR-D upward by virta*

| **Step** | **Reserve Activated** |
|---|---|
| 1st step: | f $\geq$ 49.90 = 0 % |
| 2nd step: | f $\leq$ 49.85 = 12.5 % |
| 3rd step: | f $\leq$ 49.80 = 25 % |
| 4th step: | f $\leq$ 49.75 = 37.5 % |
| 5th step: | f $\leq$ 49.70 = 50 % |
| 6th step: | f $\leq$ 49.65 = 62.5 % |
| 7th step: | f $\leq$ 49.60 = 75 % |
| 8th step: | f $\leq$ 49.55 = 87.5 % |
| 9th step: | f $\leq$ 49.50 = 100 % |
| 10th step: | f $\geq$ 49.90 = 0 % |

### 2.4.4 A Practical Example of Stepwise FCR-D Upward in Action

Figures (4) and (5) show a synthetic test scenario with a certain number of AC station graphs for FCR-D upward alongside an accurate graph. This graph illustrates meter values reported by charging points in a group, displayed over a chosen period. The yellow curve represents the group's total power output, while the black curve represents frequency. Notably, the actual power output of the group aligns with the frequency variation.

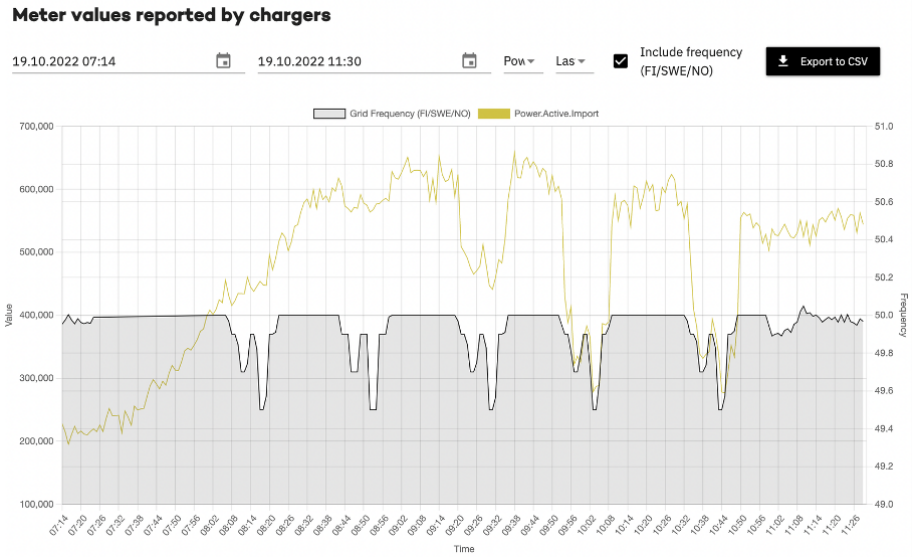



*Figure 4. The maximum power output in the graph below is somewhere around 700kW*

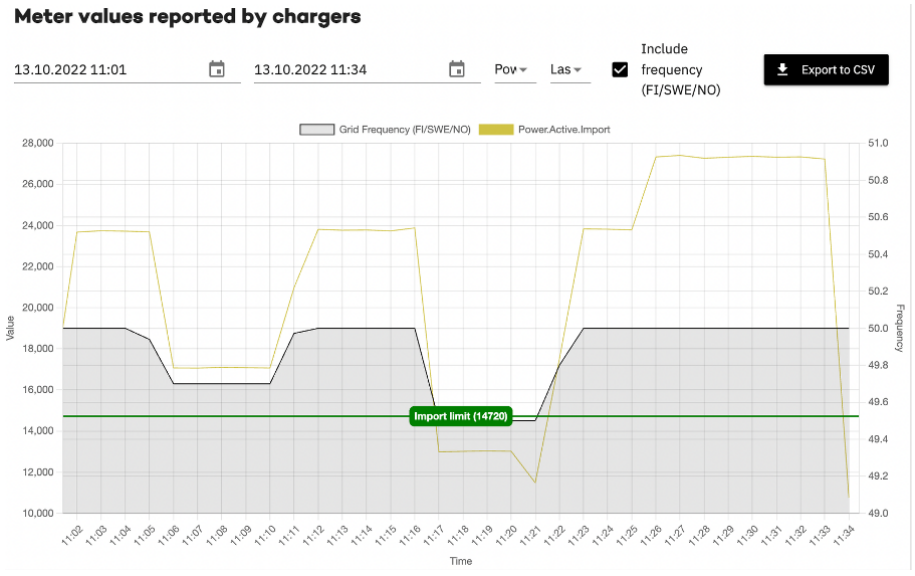



*Figure 5. Frequency restored to 50 Hz for 5 minutes*

### 2.4.5 Technical Requirement

Companies must fulfill specific technical requirements to participate in the reserve markets, providing FCR-D upward. They must undergo pre-qualification tests to demonstrate

compliance. The Reserve Unit must include a controller that adjusts power based on frequency measurement when participating in reserve maintenance. The activation of FCR-D upward is based on automatic local control, as Fingrid does not send control signals. (Fingrid 2024). Table 2 shows the upward summary of the technical requirements of FCR-D upward.

*Table 2. FCR-D upward Technical Requirement*

| **Description** | **FCR-D upward** |
|---|---|
| Minimum Bid size | 1 MW |
| Activation Performance | - Power: 86 % / 7.5 s<br>- Energy: 3.2 x capacity* / 7.5 s<br>- With fast frequency change from 49.9 Hz to 49.5 Hz |
| Stability requirement | - Dynamic FCR-D up: valid<br>- tested with sine wave test in frequency domain<br>- static FCR-D up : not valid |

FCR-D upward is divided into dynamic and static products. Dynamic FCR-D upward can be regulated continuously and deactivate the reserve quickly when frequency deviation decreases. Static FCR-D upward can not be constantly held and requires more recovery time after activation.

**Procurement For Hourly market**

Fingrid purchases FCR-D upward from the domestic yearly and hourly markets. Virta currently operates in an Hourly bidding market. Table 3 shows how the hourly market procurement is carried out.

*Table 3. Procurement for Hourly Market*

| **Hourly market** |
|---|
| A reserve provider can participate in the hourly market<br>by making a separate agreement with Fingrid. |
| It is possible to enter the hourly market<br>even in the middle of the year. |
| TSO buys only the required amount of reserve. |
| Bids for the hours in the following 24-hour period<br>must be submitted by 6.30 pm (EET). |
| Reserve providers may submit daily offers for their reserve capacity.<br>A provider that has a yearly agreement may participate in the hourly market<br>only if it has supplied the reserve amount specified in the yearly agreement in full. |
| Payment is set based on the most expensive bid used separately for each hour. |

Balancing Service Provider (BSP) submits hourly bids to the FCR-D upward hourly markets for the next day (CET time zone) until 18.30 (EET). For each hour, Fingrid places the bids in price order, giving priority to the cheapest bid. Fingrid confirms the transactions for the next calendar day by 22.00 (EET).

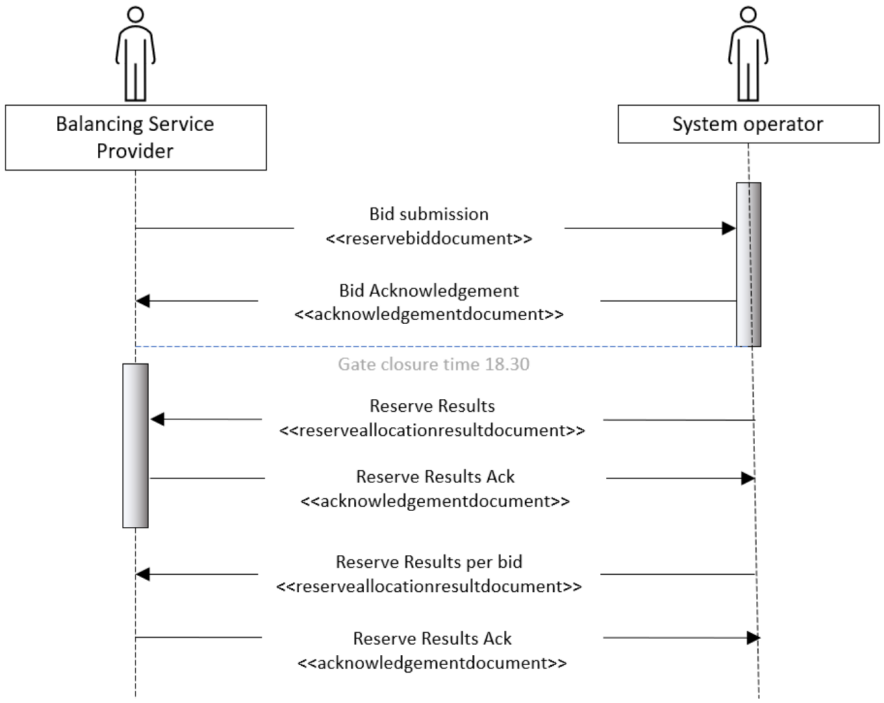



*Figure 6. The communication regarding the bidding process to the FCR-markets (Entsoe 2023)*

# 3. Related Work

In recent years, forecasting techniques for EV charging demand have seen significant advancements, driven by the growing importance of electric mobility. This section presents a comprehensive literature review encompassing various methodologies for forecasting short-term and long-term EV charging loads. Electricity markets require efficient management to ensure reliable supply and stability, particularly with the increasing integration of EVs into the grid. This section reviews pertinent literature on forecasting methodologies for ancillary electric power services and EV charging networks.

Amini et al. (2016) applied Autoregressive Integrated Moving Average (ARIMA) to forecast the daily charging demand of EV parking lots By adjusting the integrated and autoregressive order parameters to minimize the mean square error (MSE). Zhu et al. (2019) introduces, deep learning approaches in super-short-term minute-level short-term EV charging load forecasting. The author uses Artificial Neural Network (ANN), Recurrent neural network (RNN), Long Short-term Memory (LSTM), Gated Recurrent Unit (GRU), stacked autoencoders (SAEs), and Bidirectional Long Short-Term Memory (Bi-LSTM) models. The LSTM model is superior to the other methods and is competent in forecasting extra-short-term EV charging load. Shahriar et al. (2021) used the historical charging data in conjunction with weather, traffic, and events data to predict EV session duration and energy consumption using algorithms like random forest, Support Vector Machine (SVM), eXtreme Gradient Boosting (XGBoost), and deep neural networks. The ensemble learning method achieved the best predictive performance. Ye et al. (2021) studied the short-term demand for public charging stations. Their study included Random Forest, ANN, and LSTM with features utilization rates, traffic flows, demographic properties, EV registration, and points of interest for modeling. Study results show that the LSTM model achieves the best prediction performance. Mohammad et al. (2023) The paper proposes two encoder-decoder models based on convolution-extended short-term memory networks (ConvLSTM) and bidirectional ConvLSTM (BiConvLSTM) in combination with the LSTM network. In terms of (Root Mean Squared Error) RMSE, the average outcomes of two proposed models, the ConvLstm-BiLstm and BiConvLstm-Lstm, outperform other state-of-art encoder-decoder-based models LSTM, CNN, CNN-

LSTM, LSTM, and BiLSTM .Koohfar et al. (2023) use the Transformer-based model for predicting EV charging demand for both short-term and long-term forecasting. The author benchmarked the transformer-based model against ARIMA, Seasonal Autoregressive Integrated Moving Average( SARIMA), LSTM, and RNN. The transformer outperformed the other models in long-term forecasting while slightly lagging behind LSTM in short-term prediction. In the context of multivariate long-horizon forecasting, Transformer Vaswani et al. (2023) approaches have dominated the landscape in recent years, including Autoformer (Wu et al. 2021), an encoder-decoder model with decomposition capabilities and an approximation to attention based on Fourier transform, (Zhou et al. 2020). A simple linear global univariate model can outperform the transformer-based multivariate approaches for long-term forecasting (Zeng et al. 2023), which learns a linear mapping from context to the horizon. Challu et al. (2022) proposed NHiTS, which provides an average accuracy improvement of almost 20% over the latest Transformer architectures while reducing the computation time by an order of magnitude (50 times). Das et al. (2023) proposed a Time-series Dense Encoder (TiDE) for long-term time-series forecasting, being able to handle covariates and non-linear dependencies. The method can match or outperform prior approaches on popular long-term time-series forecasting benchmarks while being 5-10x faster than the best transformer-based model.

Kirsch & Singh (1995) conducted pioneering work on reserve pricing and costs from a market perspective, delineating various cost components. Their study provided insights into the complexities of reserve power pricing, addressing foregone sales, uneconomic operation costs, start-up/shut-down costs, and efficiency losses. The SARIMAX and ANN forecasts, which combine autoregressive and external predictors, have a significantly higher forecast quality than the purely autoregressive SARIMA model, and the forecast is purely based on exogenous predictors. Olsson & Soder (2008) modeled real-time balancing power market prices in the Nordic region using combined SARIMA and discrete Markov process models. Their findings underscored the suitability of the developed model combination for generating real-time balancing power price scenarios. Dimoulkas et al. (2016) applied Hidden Markov Models (HMMs) to forecast balancing market prices in the Nordic region, highlighting the challenges in predicting reserve activation-based prices because of random occurrences and the price is hardly predictable.

Their analysis revealed better predictability for up-regulation volumes and premiums than down-regulation counterparts. Giovanelli et al. (2017) examined forecasting FCR prices for the day ahead, comparing ARIMA with machine learning methods. Machine learning methods outperformed ARIMA predictions for certain periods, including Linear Regression, SVR, and gradient-boosted decision Trees (GDBT). Giovanelli et al. (2018) focused on predicting day-ahead prices for ancillary markets, with a case study on the FCR-N market. Their study revealed the superior performance of an Artificial Neural Network (ANN) model over Support Vector Regression (SVR) and Autoregressive Integrated Moving Average (ARIMA) models. Kraft et al. (2020) utilized SARIMA, SARIMAX, and neural network models for FCR price forecasting, emphasizing the benefits of combining autoregressive and exogenous factors. While neural networks outperformed in forecast quality, econometric approaches offered advantages in interpretability. Kempitiya et al. (2020) They proposed a generalized model for global frequency reserve markets, employing a Bayesian neural network approach for price prediction. Their model demonstrated high accuracy, achieving an uncertainty accuracy (UA) value of 78.88% using MC-Dropout for uncertainty estimation. Einolander & Lahdelma (2022) Developed an event-based simulation model to assess the demand response potential of EV charging networks. Using multivariate copulas, the model generates artificial charge events to simulate real-world data, achieving a mean absolute percentage error (MAPE) of 3.27% for daily dispatchable FCR-D potentials and 4.65% for weekly predictions.

## 3.1 Research Gap

Many authors have worked with time series forecasting tasks related to various sectors, of which EV demand forecasting and FCR-D upward bid price are among them. There has not been enough research related to EV charge demand prediction. Even with the research out there, most of the research is either carried out on the simulated data or with a few data points from a specific location of the charging point, which does not entirely help in understanding the problem. Most EV charge forecasting research utilizes the classical statistical approach like ARIMA and its variants, classical machine learning approaches like SVM, RF, and XGBoost, and neural-based approaches like ANN, RNN, LSTM, and its variants. The topics related to EV charge demand response forecasting have not yet explored many of the state-of-the-art models for time series forecasting. This

thesis explores the gaps in other research, mainly for long-term time series forecasting for a horizon of 720 hours. The thesis examines a classical and state-of-the-art deep learning-based approach that others have not explored yet, which has shown great potential in forecasting electricity, Electricity Transformer Temperature (ETT), and weather datasets. The thesis employs real-world data from a charging point across different locations in Finland.

Similarly, to forecast the bid price of FCR-D upward, most of the research carried out in the Nordic market is related to FCR-N. Even with FCR topics, there are limited studies, and they do not carry out any state-of-the-art deep learning approaches for forecasting. None of the thesis tries to forecast a long horizon period. The other author's research discusses the potential of EVs to be a part of a power reserve to ensure power balance. The provision of FCR with large-scale Battery Energy Storage Systems (BESS) has been investigated and shown to be possibly profitable, depending on. Some research showcases the EV as a power adjustment based on the simulated data, but none showcases its real-world implications. Overall, this thesis tries to overcome the research gaps and utilize the newer and promising deep learning approaches that any other author has not explored. This thesis attempts to show that, despite challenges, providing ancillary services can benefit the owner, the aggregator, and the grid.

# 4. Research Method

## 4.1 Time-series Analysis and Forecasting

Time series analysis and forecasting are pivotal in understanding and predicting patterns within sequential data recorded over specific time intervals. These intervals can range from yearly to monthly or even finer resolutions. Unlike conventional regression problems, time series analysis introduces unique challenges due to its time-dependent nature, inherent trends, and seasonality. In contrast to linear regression models, where observations are assumed to be independent, time series data exhibits temporal dependencies. This dependence on time introduces complexities such as seasonality trends, where patterns repeat over specific periods. The value of a data point at a given time depends on the values of previous data points in the sequence. Understanding these characteristics is crucial for accurate forecasting.

Two primary methods are commonly employed for time series forecasting:

1. Univariate Time Series Forecasting: This method analyzes only two variables - time and the field to be forecasted. It simplifies the analysis by focusing solely on the target variable's temporal evolution.

2. Multivariate Time Series Forecasting: In contrast, multivariate forecasting incorporates multiple variables, with time as one of the dimensions. This approach allows for more comprehensive modeling by considering various parameters alongside temporal trends.

### 4.1.1 Time Series Analysis

Before forecasting, a thorough analysis of the time series data is essential. This analysis aims to develop models that capture underlying patterns, facilitating accurate predictions. A fundamental aspect of this analysis involves decomposing the time series into four constituent parts:

1. Level: Represents the baseline value of the series if it follows a linear trend.

2. Trend: Describes the optional linear increasing or decreasing behavior observed

over time.

3. Seasonality: Captures recurring patterns or cycles within the data.

4. Noise: Accounts for variability in observations that the model cannot explain.

Understanding these components enables practitioners to discern meaningful patterns and relationships within the data, laying the groundwork for effective forecasting.

### 4.1.2 Time Series Forecasting

Forecasting entails predicting future observations based on historical data and established models. It involves utilizing fitted or trained models to extrapolate future trends accurately. Various statistical, machine learning, and neural-based models are commonly employed. As part of this thesis, we have selected specific models for evaluation. These models are chosen based on the benchmark from the year 2023, which has shown tremendously promising results on data such as electricity, ETT, and weather data. There are many models based on MLP, transformers, and large language models, but for the thesis, we are sticking with the MLP-based but promising model that has outperformed even the transformers. The top models that are chosen are:

**Statistical-Based Approach**

- Multiple Seasonal-Trend decomposition using Loess (MSTL)
- Exponential Trend Smoothing (ETS)

**State-of-the-Art Time Series Forecasting Approach**

- Neural Hierarchical Interpolation for Time Series (NHiTS)
- Time-series Dense Encode (TiDE)
- Decomposition Linear (DLinear)

## 4.2 Multiple Seasonal-Trend decomposition using Loess (MSTL)

There are many methods to decompose a time series with a single seasonal component, such as Seasonal-Trend Decomposition using Loess (STL ) (Cleveland et al. 1990) and X-13-ARIMA-SEATS (Bell & Hillmer 1984). Many time series can have multiple seasonality; Bandara, Hyndman, and Bergmeir proposed a new algorithm for multi-seasonal decomposition called Multiple Seasonal-Trend decomposition using Loess (MSTL) (Hyndman et al. 2008). MSTL builds on STL, a decomposition method that can extract a single seasonal component. Methods that extract multiple seasonal components include Prophet (Taylor & Letham 2018), TBATS (De Livera et al. 2011), and STR (Dokumentov & Hyndman 2022). Prophet and TBATS infer the seasonal components as part of training a forecasting model. In an additive decomposition, MSTL assumes that the following equation can express the time series decomposition:

$$y_t = \widehat{T}_t + \hat{S}_t^{(1)} + \hat{S}_t^{(2)} + \cdots + \hat{S}_t^{(N)} + \hat{R}_t \tag{1}$$

The MSTL model handles multiple seasonal patterns(e.g., daily, weekly, yearly, etc), captures complex patterns, reduces the impact of noise and outliers, and provides accurate forecasts. These features make the MSTL model valuable for exploratory time series analysis, data forecasting, and pattern detection, especially when dealing with multiple seasonal components and intricate trends.

### 4.2.1 Loess Overview

Loess is a nonparametric smoothing method that fits a smooth curve to scatter plots (Cleveland & Devlin 1988). Loess computes the smooth curve at any given point by fitting a polynomial to a data window around that point. Loess performs a weighted regression using nearest neighbors for each point in the time series.

**The Loess calculation involves the following steps:**

For each point in the time series, we select a neighboring window and assign weights based on proximity using a tri-cubic function.

$$w(x) = \begin{cases} \left(1 - |x|^3\right)^3 & |x| < 1 \\ 0 & |x| \geq 1 \end{cases} \tag{2}$$

This function has positive values only between **-1** and **1**. Outside of this interval, the function is zero. We must normalize the distance by dividing it by the maximum value observed in $D$. More concretely,

$$w(x) = \begin{cases} \left(1 - \left|\frac{d(x,x')}{\max_i d(x_i,x')}\right|^3\right)^3 & |x| < 1, x_i \in D \\ 0 & |x| \geq 1 \end{cases} \tag{3}$$

We use ***d(x, x')*** to denote the distance between ***x*** and ***x'***, one of the k's nearest neighbors. By normalizing the values, more considerable distances are assigned lower weights. The point with the maximum distance will weigh zero, and the one at zero distance will have the highest possible weight of one. This technique assigns higher importance to the training data closest to the prediction point, achieving the "locality" effect.

Perform weighted regression by using the assigned weights and neighbors. Obtain the fitted value for point t through local regression.

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

$$\hat{y} = \beta^T X \tag{5}$$

The process is repeated for all points in the time series, thus obtaining a smoothed estimate of the trend.

### 4.2.2 MSTL Algorithm

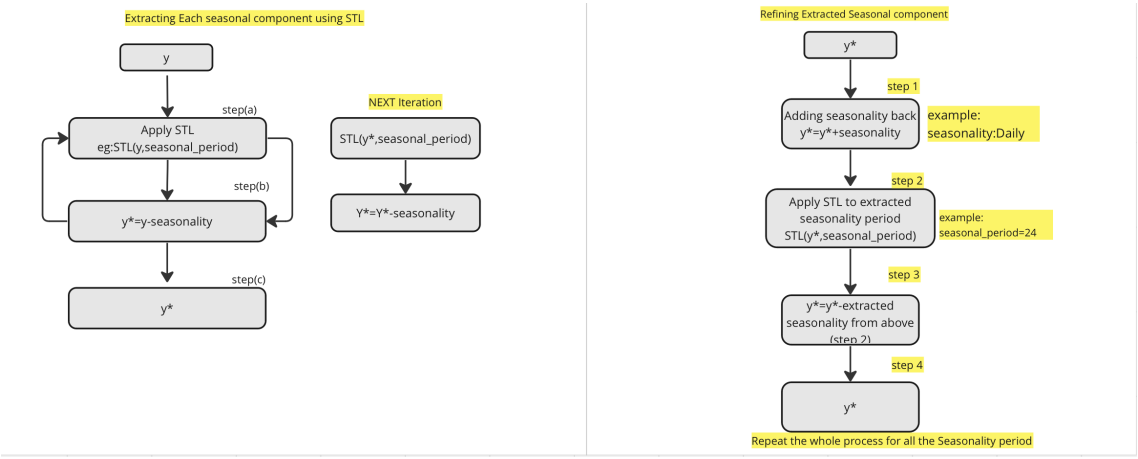



*Figure 7. MSTL algorithm*

The steps involved in the MSTL algorithm are explained below, as shown in Figure 7. **Seasonal Component Extraction:** The algorithm follows an iterative process that begins with the shortest period and moves towards the most extended period (e.g., hourly, daily, weekly). Starting with longer seasonality would erroneously include shorter seasonality as part of the more extended seasonal component. The algorithm extracts each seasonal component using the STL method and subtracts it from the time series. We continue the process of iteration until we accurately extract all seasonal elements.

**Refinement of Extracted Components:**

The algorithm refines each seasonal component by adding it back to each iteration's fully deseasonalized time series. Then, the algorithm re-extracts the same seasonal component using STL and subtracts the new estimate from the time series. This process is repeated for each seasonal component to enhance accuracy.

**Trend Extraction:** During the refinement step, we extract the trend component from the most extended seasonal period of the last STL fit.

**Residual Calculation:** Subtracting the trend component from the fully deseasonalized time series gives us the residual component.

After these steps, the algorithm provides a trend and refined seasonal and residual components. If no seasonal component is specified, the algorithm smooths the data to derive the trend component.

## 4.3 Exponential smoothing (ETS):

ETS stands for "Error-Trend-Seasonality" and defines how specifically the components interact with each other. Based on the type of error, trend, and seasonality, Pegels (1969) proposed a taxonomy, which was then developed further by Hyndman et al. (2002) and refined by Hyndman et al. (2008). According to this taxonomy, error, trend, and seasonality can be:

- Error: **Additive** (A) or **Multiplicative** (M);
- Trend: **None** (N), or **Additive** (A), or **Additive damped** (Ad), or **Multiplicative** (M), or **Multiplicative damped** (Md);
- Seasonality: **None** (N), **Additive** (A), or **Multiplicative** (M).

For each exponential smoothing method in the ETS model, (Hyndman et al. 2008) describes two possible innovations in state space models, one corresponding to a model with additive and multiplicative errors. The classification from Figure 8 presents 30 possible models. The two models produce equivalent point forecasts using the same parameter values but different prediction intervals.

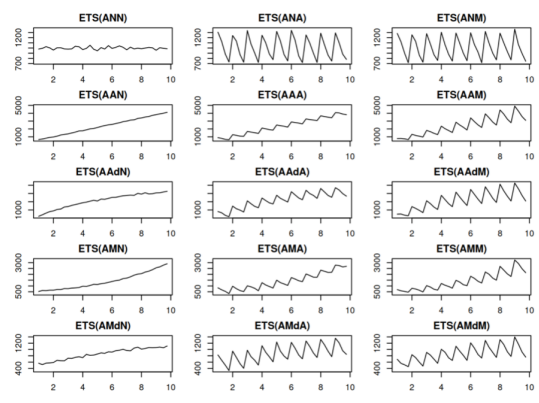



Figure 4.1: Time series corresponding to the additive error ETS models.

(a)

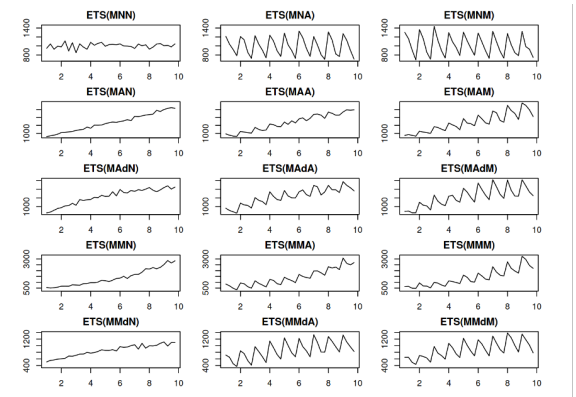



Figure 4.2: Time series corresponding to the multiplicative error ETS models.

(b)

*Figure 8. (a) Additive error (b) Multiplicative error ETS model (openforecast 2023)*

### 4.3.1 Model Selection

Information criteria such as AIC, $AIC_c$, and BIC can be used to select the appropriate ETS model for the time series data.

Akaike's Information Criterion (AIC) for ETS models is as follows:

$$\text{AIC} = -2\log(L) + 2k \tag{6}$$

Where L is the likelihood of the model and k is the total number of parameters and initial states that have been estimated (including the residual variance).

The AIC corrected for small sample bias $AIC_c$ is defined as

$$AIC_c = AIC + \frac{2k(k+1)}{T-k-1} \tag{7}$$

and the Bayesian Information Criterion (BIC) is

$$\text{BIC} = \text{AIC} + k[\log(T) - 2] \tag{8}$$

Three combinations (Error, Trend, Seasonal) can lead to numerical difficulties. Specifically, the models that can cause such instabilities are ETS(A,N,M), ETS(A,A,M), and ETS(A,Ad,M) due to division by values potentially close to zero in the state equations. We usually do not consider these particular combinations when selecting a model.

When we deal with time series data, multiplicative error models prove helpful only when the data is strictly positive. However, these models lose numerical stability in the presence of zeros or negative values. Therefore, we will not consider multiplicative error models if the time series contains non-positive values. Instead, we will apply only the six fully additive models.

## 4.4 Neural Hierarchical Interpolation for Time Series (NHiTS)

January 2022 Neural Hierarchical interpolation for Time Series forecasting (NHiTS) was proposed by Challu et al. (2022) which enhances N-BEATS (Oreshkin et al. 2019) that improves the treatment of the input and the output which result in the better accuracy and lower computation cost. On the long-horizon forecasting task, NHiTS improved accuracy by 25% on the Association for The Advancement of Artificial Intelligence (AAAI's) best paper award, the Informer (Zhou et al. 2020), while being 50x faster. (Challu et al. 2022). NHiTS samples the time series at different rates, which can help learn the short-term and long-term effects of the series. Generating the prediction combines the forecast obtained at a different time scale, which is called hierarchical interpolation. The NHiTS model

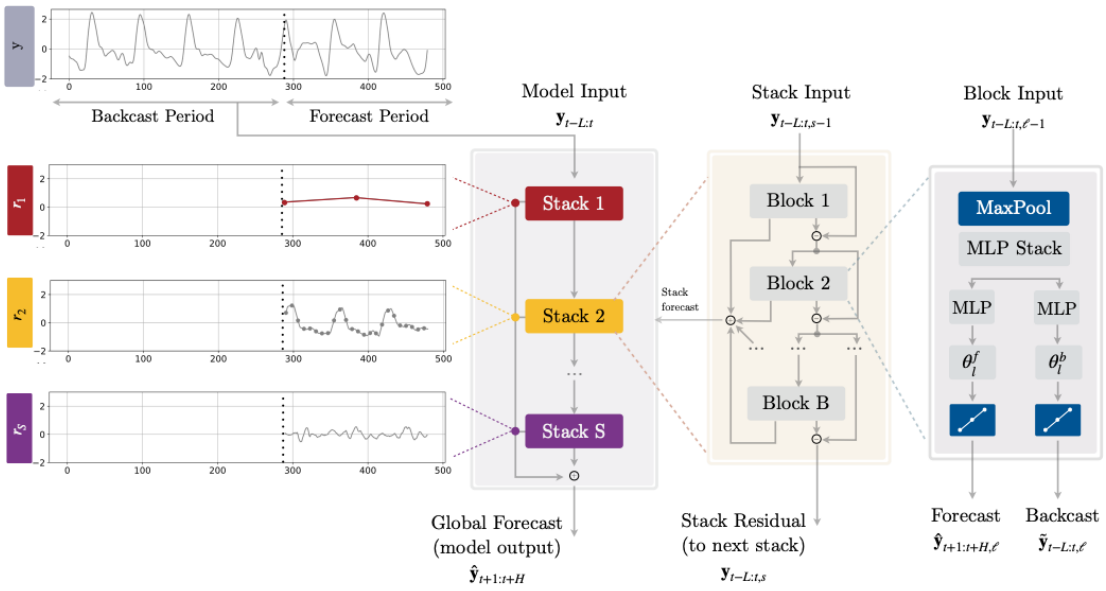



*Figure 9. Architecture of NHiTS Image from Challu et al. 2022*

comprises stacks and blocks, with residual connections between each block within a stack. The architecture of NHiTS in Figure 9 shows that the final prediction is the sum of partial predictions from each stack.

### 4.4.1 Multi-Rate Signal Sampling

The MaxPool layer within a block facilitates multi-rate sampling in NHiTS. Each stack in NHiTS has its kernel size, which determines the sampling rate. Larger kernel sizes smooth the series and focus on long-term effects, while smaller ones maintain short-term effects, as shown in Figure 10. This multi-rate processing reduces the MLP input width for most

blocks. Utilizing a multi-rate sampling technique with sub-sampling layers accomplishes several benefits: It reduces memory usage and computation, mitigates overfitting by decreasing the number of learnable parameters, and preserves long-term forecasting.

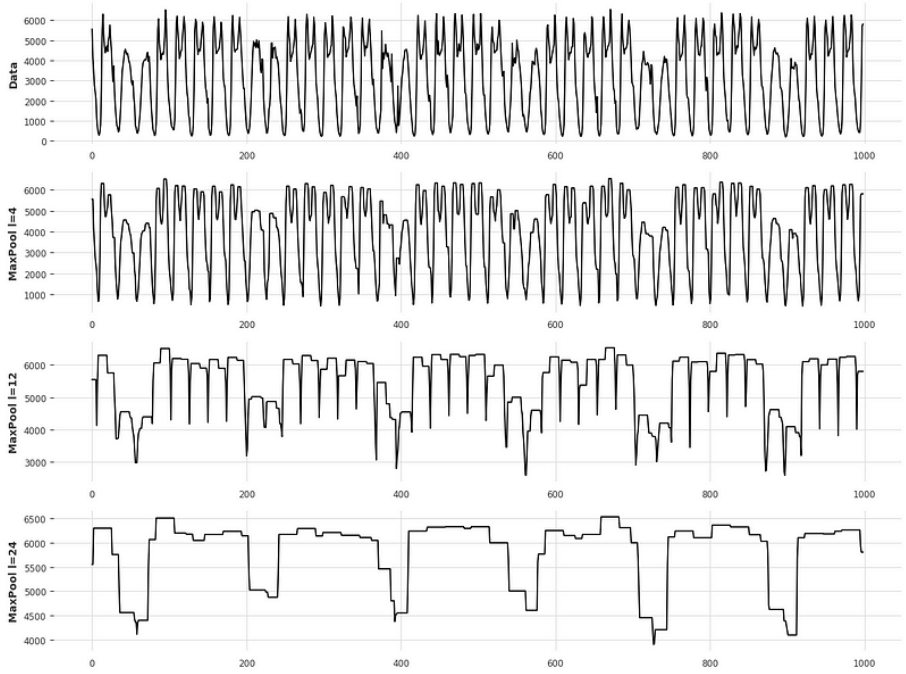



*Figure 10. The effect of kernel size at the MaxPool layer*

Equation (9) provides the NHiTS multi-rate sampling. Given block l input $y_t - L;t$ (the input to the first block l = 1 is the network-wide input, $y_t - L;t,1 = y_t - L : t$ ), this operation can be formalized as follows:

$$y^{(p)}_{t-L:t,l} = \text{MaxPool}\left(y_{t-L:t,l}, k_l\right) \tag{9}$$

where $l$ (the number of blocks) is getting $y_{t-L:t,l}$ ( the input) (Challu et al. 2022)

### 4.4.2 Non-Linear Regression

Once the input signal passes the MaxPool layer, the block uses fully connected networks to perform a regression and output a forecast and a backcast. Backcast is the information captured by the block. So, we remove the backcast from the input signal and pass the resulting signal to the next block through residual connections. That way, the next can model any information not captured by a particular block.

### 4.4.3 Hierarchical Interpolation

The Hierarchical interpolation aims to reduce the prediction cardinality in forecasting models, which is the number of elements in a set. As the forecast horizon lengthens, this cardinality can cause computational challenges and unnecessary model complexity (Challu et al. 2022). Each stack in NHiTS has an expressiveness ratio ( number of predictions per unit of time) that specializes in treating the series at a different rate because the MaxPool layer sub-samples the series. For instance, one stack focuses on short-term effects hourly, another sub-samples every 12 hours, and the next every 24 hours. Consequently, each stack operates at its scale, resulting in different cardinalities. Combining predictions from different time scales achieves hierarchical interpolation, resulting in fewer outputs.

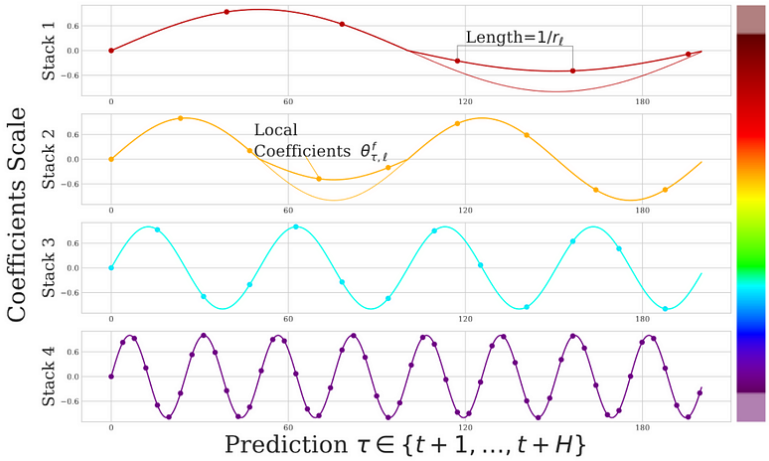



*Figure 11. Hierarchical interpolation in NHiTS Challu et al. 2022*

Figure (11) shows that each stack outputs a different number of predictions because it specializes in its scale (or frequencies). Stack 1 looks at long-term effects, making its predictions more spaced in time. On the other hand, Stack 4 looks at short-term effects and outputs more granular predictions that are closer in time.

## 4.5 Time-series Dense Encode (TiDE)

Several transformer-based models have emerged for time series forecasting, such as Informer Zhou et al. (2020), Autoformer Wu et al. (2021), PatchTST Nie et al. (2022), and TFT Lim et al. (2021), targeted at long-term horizons. However, recent studies indicate that simple linear models can outperform these transformer-based approaches on various benchmark datasets (Zeng et al. 2023). In April 2023, Google researchers in-

troduced TiDE (Time-series Dense Encoder), a novel encoder-decoder model explicitly designed for time series forecasting (Das et al. 2023). TiDE deliberately avoids the attention mechanism of transformer models and instead utilizes MLPs for efficient training and faster inference. During training, TiDE encodes past data with covariates. It decodes the learned representation with known future covariates to make predictions, demonstrating superior performance over state-of-the-art transformer models for long-time horizon forecasts.

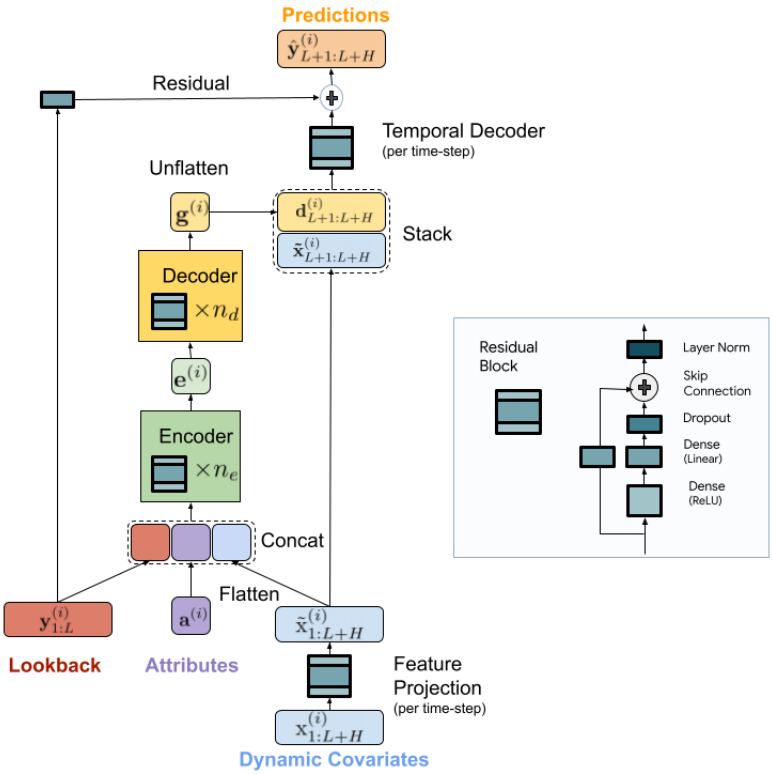



*Figure 12. The architecture of TiDE (Das et al. 2023)*

The TiDE architecture processes each series independently, simultaneously passing one series and its covariates. There are three main components to the model shown in Figure (12):

- An Encoder
- A Decoder
- A Temporal Decoder

They all rely on the residual block architecture.

### 4.5.1 Residual Block

The decoder, encoder, and temporal decoder rely on the residual block to serve as the architecture's base layer. A residual block is a type of neural network component that consists of a multilayer perceptron (MLP) with one hidden layer and a Rectified Linear Unit (ReLU) activation function. After the MLP, there is a dropout layer, an entirely linear skip connection, and a final layer normalization step. The network uses this component repeatedly for encoding, decoding, and making predictions.

### 4.5.2 The Encoder

The encoding step maps a time series's past and covariates to a dense representation of the feature (Das et al. 2023). It has two-step :

- **Feature projection**: The residual block maps $x_t^{(\iota)}$ dynamic covariates (exogenous variables varying in time) into a lower dimensional projection.

$$\tilde{x}_t^{(i)} = \text{ResidualBlock}\left(x_t^{(i)}\right) \tag{10}$$

  the future value of the feature is necessary for multivariate forecasting Therefore, the model must treat a look-back window and a horizon sequence. These sequences can get very long, so by projecting to a lower dimensional space, we keep the lengths manageable and allow the model to treat longer sequences, both in terms of historical window and horizon of forecast.

- **Dense encoder**: The encoder concatenates the past of a series with its attributes and projections of past and future covariates. The encoder, a stack of residual blocks, learns a representation of the inputs, effectively creating a learned embedding. Once the encoder completes the encoding, it sends the embedding to the decoder for further processing. The equation gives the Dense encoder:

$$e^{(i)} = \text{Encoder}\left(\mathbf{y}_{1:L}^{(i)}; \tilde{x}_{1:L+H}^{(i)}; a^{(i)}\right) \tag{11}$$

### 4.5.3 Decoder

It takes the learned representation of the encoder and generates predictions.

**Dense Decoder:** The dense decoder, comprised of a stack of residual blocks, takes the encoded information and produces a matrix for the temporal decoder as the first step.

$$\begin{aligned} g^{(i)} &= \text{Decoder}\left(e^{(i)}\right) \quad \in \mathbb{R}^{p.H} \\ D^{(i)} &= \text{Reshape}\left(g^{(i)}\right) \quad \in \mathbb{R}^{p \times H} \end{aligned} \tag{12}$$

It takes an input from the encoding $e^{(i)}$ and maps it to a vector $g^{(i)}$ of size $H \times p$, where p is the decoder output dimension. The vector $g^{(i)}$ goes through reshape to get a matrix $D^{(i)} \in \mathbb{R}^{d \times H}$.

The decoded output is stacked with projected features to capture the direct effects of future covariates. For example, holidays are punctual events that can impact specific time series. With this residual connection, the model can capture and leverage that information.

**Temporal Decoder:** The temporal decoder produces the final predictions for a given time series using a residual block with an output size of 1. It maps the decoded vector $d_t^{(\iota)}$ at $t-th$ horizon time-step concatenated with the projected covariates $\tilde{x}_{L+t}^{(i)}$, i.e.

$$\hat{\mathbf{y}}_{L+t}^{(i)} = \text{TemporalDecoder}\left(d_t^{(i)}; \tilde{x}_{L+t}^{(i)}\right) \quad \forall t \in [H] \tag{13}$$

## 4.6 Decomposition Linear (DLinear)

Researchers have recently introduced several Transformer-based solutions for long-term time series forecasting (LTSF). However, despite their effectiveness, Transformers exhibit significant shortcomings in long-term forecasting. They struggle to extract temporal relations efficiently from long sequences due to their permutation-invariant self-attention mechanism (Zeng et al. 2023). Additionally, increasing the look-back window size does not reduce forecasting errors, indicating a failure to utilize additional historical data effec-

tively. Furthermore, Transformers fail to adapt their temporal attention, assigning similar importance to different points in time regardless of their specific characteristics. This limitation can hinder their predictive capability, mainly when dealing with non-stationary real-world data.

Zeng et al. (2023) introduce a set of embarrassingly simple one-layer linear models named long-term time series forecasting (LTSF). To validate their performance, the author challenges transformer-based LTSF solutions with direct multi-step (DMS) forecasting strategies via a temporal linear layer called LTSF-Linear. The LTSF-Linear directly regresses historical time series through a weighted sum operation for future prediction. The mathematical expression is :

$$\hat{X}_i = WX_i \tag{14}$$

Where $W \in \mathbb{R}^{T \times L}$ is a linear layer along the temporal axis. $\hat{X}_i$ and $X_i$ are the prediction and input for each $i_t h$ variate. LTSF-Linear shares weights across different variate and does not model any spatial correlations. *LTSF-Linear* is a set of linear models. *Vanilla Linear* is a one-layer linear model. Two variants with two prepossessing methods, DLinear and NLinear, are introduced to handle time series across different domains. For our thesis, we are focusing on DLinear.

DLinear combines a Decomposition scheme used in Autoformer (Wu et al. 2021) and FEDformer (Zhou et al. 2022) with linear layers. Figure (13) shows the DLinear architecture where input raw data is decomposed into trend and seasonal components using a moving average kernel. After decomposition, it follows one linear layer to the Remainder and Trend. We obtain the final prediction by adding the output of both linear layers. DLinear has proven more effective than vanilla linear models in managing trends and seasonal patterns, especially in data with clear trends.

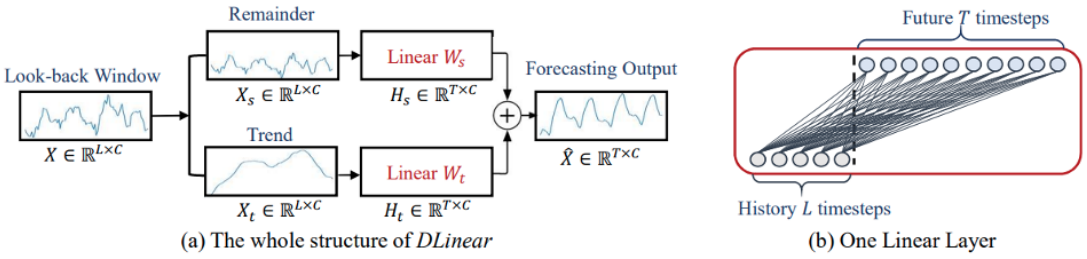



*Figure 13. The architecture of DLinear (Zeng et al. 2023)*

### 4.6.1 Benefits of Using Simple Linear Systems

Using a more straightforward model like DLinear offers several advantages. DLinear captures short- and long-range temporal relations and has a lower memory footprint, fewer parameters, and faster inference speed than existing transformers. Additionally, obtaining DLinear is easy and does not require tuning model hyper-parameters.

# 5. Data Preparation and Exploratory Analysis

We used a quantitative approach to collect and analyze data for the thesis research. We evaluated the applied algorithm for forecasting using quantitative metrics. We aimed to forecast the Virta charge demand response and the Fingrid FCR-D upward price for two tasks. The method involves **univariate** and **multivariate** approaches to estimate the feature relationship contributing to forecasting. Statistical processes analyze time series data to develop models that capture underlying patterns.

## 5.1 Dataset

There are two datasets for the forecasting task. Most quantitative prediction problems use time series data collected at regular intervals or cross-sectional data collected at a single point in time.

### 5.1.1 Electric Vehicle Charge Data

We utilize historical charging data from the Virta database and weather data from the Finnish Meteorological Institute (FMI) to predict the power required to charge electric vehicles in Finland at any particular hour. This forecasting task utilizes data collected between 2018 and 2023 and includes information on AC and DC station types that we tested for demand response purposes.

*Table 4. Raw data from the virta data source*

| STATION ID | CUSTOMER ID | CHARGE_START TIME_AT | CHARGE ID | CHARGE_STOP TIME_AT | CHARGED MINS | CHARGE Wh |
|---|---|---|---|---|---|---|
| 1 | 12 | 2023-04-06T 17:56:17Z | 21 | 2023-04-06T 19:03:40Z | 68 | 12142 |
| 2 | 123 | 2022-04-08T 20:00:00Z | 31 | 2022-04-09T 09:39:12Z | 820 | 35270 |
| 3 | 124 | 2018-08-23T 10:01:40Z | 33 | 2018-08-23T 11:27:57Z | 87 | 5000 |

Table 4 shows that the data gives the information related to charge watt-hour, based on the Charge_id , start time, and stop_time. The data are stored for each charge_id how long the charge happened and the total watt consumption during the charging period. We need to observe data at regular hourly intervals for our forecasting task. However, since some charges last for an hour while others may last for half a day, we cannot resample the data based on the charge start time for all events. To transform the data into hourly intervals and remove the irregularly spaced time series, we calculate the total duration of the charge

for each charge ID, finding the average amount of charge utilized for each hour.

We extract weather-related data from the Finnish Meteorological Institute (FMI) by downloading data from different regions of Finland, such as Helsinki, Oulu, Rovaniemi, and Jyväskylä, and then find the mean temperature data in Finland. Table 5 presents the transformed data.

*Table 5. Virta charge data Transformed Hourly interval*

| ds | 2018-01-01 00:00:00 | 2018-01-01 01:00:00 | 2018-01-01 02:00:00 | 2018-01-01 03:00:00 |
|---|---|---|---|---|
| unique_id | DC | DC | DC | DC |
| TOTAL_STATION_ACTIVE | 41 | 41 | 41 | 41 |
| CHARGED_SESSION | 2 | 2 | 1 | 1 |
| STATION_USED | 2 | 2 | 1 | 1 |
| CUSTOMER_CHARGING | 2 | 2 | 1 | 1 |
| TOTAL_STATION_VENDOR_USED | 2 | 2 | 1 | 1 |
| TOTAL_STATION_MODEL_USED | 2 | 2 | 1 | 1 |
| TOTAL_SECONDS | 2423 | 1297 | 900 | 1500 |
| Pressure (msl) (hPa) | 1001.525 | 1000.8 | 1000.475 | 999.975 |
| Precipitation amount (mm) | 0.0 | 0.0 | 0.0 | 0.5 |
| Relative humidity (%) | 88.0 | 89.75 | 90.25 | 92.0 |
| Precipitation intensity (mm/h) | 0.0 | 0.0 | 0.0 | 0.15 |
| Snow depth (cm) | 20.0 | 20.0 | 20.0 | 21.0 |
| Air temperature (degC) | -1.575 | -1.575 | -1.75 | -1.9 |
| Dew-point temperature (degC) | -3.325 | -3.075 | -3.1 | -3.025 |
| Wind direction (deg) | 141.5 | 144.25 | 148.25 | 144.0 |
| Gust speed (m/s) | 6.0 | 6.975 | 6.675 | 6.675 |
| Wind speed (m/s) | 4.175 | 4.625 | 4.3 | 4.4 |
| y | 0.028892 | 0.015145 | 0.01026 | 0.01662 |

### 5.1.2 Frequency Containment Reserve For Disturbance in Upward Dataset

For the forecasting task of FCR-D upward price, data was extracted from the Fingrid open data API (Fingrid 2023c) for the period 2018-2023. Key features considered include energy production and consumption, FCR-D bid information, weather, spot price, and intraday trading. Some variables, such as inertia, were not included.

The extracted API data is transformed into an hourly interval data frame, as shown in Table (6).

*Table 6. FCR-D upward data extracted from the Fingrid API*

| **index** | 2021-01-01 00:00:00+00:00 | 2021-01-01 01:00:00+00:00 | 2021-01-01 02:00:00+00:00 |
|---|---|---|---|
| **production(MW)** | 6890.0 | 6829.0 | 6855.0 |
| **consumption(MW)** | 9206.0 | 9042.0 | 9010.0 |
| **mean_import/export** | -2315.05 | -2211.225 | -2153.365 |
| **up_price(€/MW)** | 3.0 | 3.0 | 3.0 |
| **up_volume(MW)** | 95.4 | 95.4 | 95.4 |
| **up_volume_yearly(MW)** | 366.7 | 366.6 | 367.5 |
| **bid(MW)** | 284.0 | 289.0 | 289.0 |
| **nordic_traade(MW)** | -164.8 | -164.7 | -165.6 |
| **min_hz** | 49.894 | 49.901 | 49.909 |
| **mean_hz** | 49.92385 | 49.93505 | 49.9474 |
| **max_hz** | 49.949 | 49.96 | 49.979 |
| **temp** | -0.124639 | -0.283821 | -0.358821 |
| **wind(MW)** | 201.0 | 187.0 | 183.0 |
| **hydro(MW)** | 1371.345 | 1330.56 | 1375.105 |
| **nuclear(MW)** | 2788.51 | 2778.3 | 2768.95 |
| **surplus_defict(MW)** | 1906.749 | 2742.42 | 2451.58 |
| **up-regulation_power_trans** | 0.0 | 0.0 | 0.0 |
| **up-regulation_activated_power** | 0.0 | 0.0 | 0.0 |
| **mwh_price** | 24.35 | 23.98 | 23.72 |
| **industrial(MWh/h)** | 1216.19 | 1223.4 | 1213.02 |
| **other_production(MWh/h)** | 74.0 | 74.0 | 74.0 |
| **cogeneration_production(MWh/h)** | 1239.13 | 1237.64 | 1238.54 |
| **intraday_EE_FI** | 1612.2 | 1586.4 | 1564.0 |
| **intraday_FI_EE** | 419.8 | 445.6 | 468.0 |
| **intraday_FI_RUS** | 280.0 | 280.0 | 280.0 |
| **intraday_FI_SE1** | 2600.0 | 2555.3 | 2519.9 |
| **intraday_FI_SE3** | 1252.2 | 1200.0 | 1200.0 |
| **intraday_RUS_FI** | 0.0 | 0.0 | 0.0 |
| **intraday_SE1_FI** | 0.0 | -44.7 | -80.1 |
| **intraday_SE3_FI** | -1147.8 | -1200.0 | -1200.0 |
| **com_intraday_FI_SE3** | -46.7 | 12.1 | 63.3 |
| **com_FI_EE** | 609.3 | 584.0 | 568.7 |
| **com_FI_SE1** | -1559.0 | -1523.6 | -1478.8 |
| **imbalance_FI_SE** | 110.443 | 158.852 | 138.785 |
| **FI_RUS_FI_stock** | -140.0 | -140.0 | -140.0 |

## 5.2 Exploratory Data Analysis

In time series Exploratory data analysis, these methods are mainly taken into consideration:

- Collecting the data and cleaning it.
- Preparation of visualization for time series.
- Distribution test
- stationary test
- Time series Decomposition.
- Autocorrelation test

### 5.2.1 Moving Averages

In time series analysis, moving averages offer a valuable means of discerning the average movement over a specific period. Using both the simple moving average (SMA) and the exponential moving average (EMA), we can effectively mitigate short-term fluctuations, identify enduring trends, and accurately capture prevailing market movements. EMA, in particular, is a robust tool for trend identification and noise reduction achieved by gradually diminishing the weight of past data points and prioritizing recent observations. Consequently, EMA exhibits more excellent responsiveness than SMA, rapidly adapting to changing market conditions and offering greater sensitivity to evolving trends.

### 5.2.2 Distribution Test

Distribution describes the relationship between data points in their sample space and the frequency of variable values. In machine learning, data distribution profoundly affects model outcomes and accuracy. Distribution impacts machine learning models, including training-test alignment, feature characteristics, class imbalance, generalization, and the balance of bias and variance. It's essential to consider the distribution of the training and test datasets and individual features within the data, as these can affect model performance. The ability of a model to generalize well to new, unseen data largely depends on the diversity of the training dataset. Overfitting and underfitting can occur when the training data is not representative of the broader dataset, affecting the accuracy and generalizability of the model. The data distribution influences the balance between bias and variance in a model, which impacts the model's performance. Understanding and addressing distribution-related issues are crucial for building accurate and robust machine-learning models as the field evolves.

### 5.2.3 Stationary Check

Stationarity characterizes a time series with a constant mean, variance, and covariance independent of time. If a time series exhibits stationarity over a specific time interval, it will maintain this property at subsequent times. The properties of the stationary time series remain consistent regardless of the observation time. Consequently, time series with trends or seasonality are not stationary. The stationarity of a time series can be verified using either the Augmented Dickey-Fuller (ADF) or Kwiatkowski-Phillips-Schmidt-Shin

(KPSS) methods.

**Augmented Dickey-Fuller (ADF) Test**

ADF is a statistical or unit root test to test whether a given time series is stationary. Unit roots are a cause for non-stationary. When conducting the ADF test, one usually assumes a specific condition: If the p-value is less than or equal to the significance level. The default significance level is 0.05.

- **p-value > significance level**: Fail to reject the null hypothesis (H0); Series is **non-stationary.**
- **P-value <= significance level**: Reject the null hypothesis (H0); Series is **stationary**.

**The Kwiatkowski-Phillips-Schmidt-Shin (KPSS) Test**

KPSS is a type of Unit root test that tests for the stationary of a given series around a deterministic trend. In other words, the test is somewhat similar to the ADF test.

- **P-value>0.05**: Series is trend **stationary** .
- **P-value<0.05**: Series is **non-stationary**.

The KPSS test tests the hypothesis opposite to the one in the ADF test.

### 5.2.4 Decomposition

To decompose a time series, one needs to break it down into its various components:

- Trend and Level: Trend is the overall motion of the series, which may consistently increase over time, decrease over time, or a combination of both.
- Seasonality: regular seasonal pattern in the time series
- Random: After accounting for trend and seasonality, the residual component, also known as statistical noise, represents the remaining variation.

All series have a level and noise, but trend and seasonality are optional. The combination of the components in a time series can be **Additive** and **Multiplicative**

*Additive decomposition,*

$$y_t = S_t + T_t + R_t \tag{15}$$

Where $y_t$ is the data, $S_t$ is the seasonal component, $T_t$ is the trend cycle component, and $R_t$ is the remainder component, all in period $t$.

*multiplicative decomposition*

$$y_t = S_t \times T_t \times R_t \tag{16}$$

- The additive model is useful when the seasonal variation is relatively constant.
- The multiplicative model is useful when the seasonal variation increases over time.

### 5.2.5 Autocorrelation plots

Autocorrelation helps detect patterns and check for randomness. It measures the correlation between the series and its lagged values( previous point in time). There are two types of autocorrelation plots that we can use.

**Autocorrelation Function (ACF)**

ACF shows the value of the correlation coefficient between the series and its lagged values at different time intervals. That is why it's known as the complete auto-correlation plot.

- The ACF allows us to determine how much past observations influence future ones. If the ACF shows significant autocorrelation in several lags, it indicates that the series has long-term memory and that past observations are relevant for predicting future ones.
- The shape of the ACF can reveal the presence of moving average components in the time series. Lags, where the ACF shows a significant correlation, may indicate the order of an MA model.

**Partial Autocorrelation (PACF)**

PACF eliminates the indirect effects of intermediate lags and measures the direct correlation between an observation and its lagged values. It helps identify the direct dependence between an observation and its lag values without the influence of intermediate lags.

- The shape of the PACF can reveal the presence of auto-regressive components in the time series. Lags in which the PACF shows a significant correlation may indicate the order of an AR model.

- The PACF is used in conjunction with the ACF to determine the order of an AR or MA model. By analyzing both the ACF and the PACF to identify significant lags, one can build a suitable model for time series analysis and forecasting.

### 5.2.6 Correlation Table

Correlation, a statistical measure, quantifies the degree of linear association between two variables, indicating how they change together consistently. It serves as a means of elucidating superficial relationships without implying causation. The correlation coefficient, denoted by $r$, is a dimensionless metric that ranges from -1 to +1. An $r$ value closer to zero signifies a weaker linear relationship. Positive values $r$ represent a positive correlation, indicating that both variables tend to increase simultaneously. Conversely, negative $r$ values indicate a negative correlation, where an increase in one variable corresponds to a decrease in the other.

## 5.3 Electric Vehicle Charge Data EDA

Figure (14) shows the charge data relevant to each hour interval from 2018-2023. The charge is divided into AC and DC categories since we want to forecast demand based on station type. Here, the data are taken only for some specific station models, like Alfens for AC type, based on internal testing.

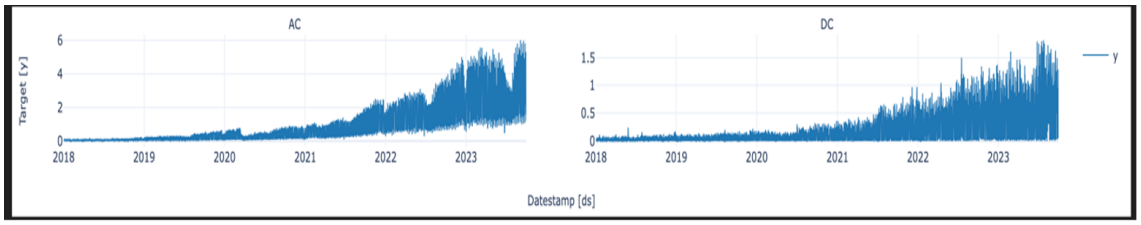



*Figure 14. Charge data from (2018-2023) for AC and DC*

Figure 15 illustrates the distribution graphs for both AC and DC types, revealing a log-normal distribution pattern. This distribution characterizes random variables whose logarithms adhere to a normal distribution, commonly observed in various natural phenomena. Examples include tissue and blood pressure sizes, income distribution, and even lengths. The log-normal distribution extends finance due to its representation of growth processes independent of size. This distribution finds application in diverse fields such as epidemiology, stock market analysis, and technological assessments such as file sizes.

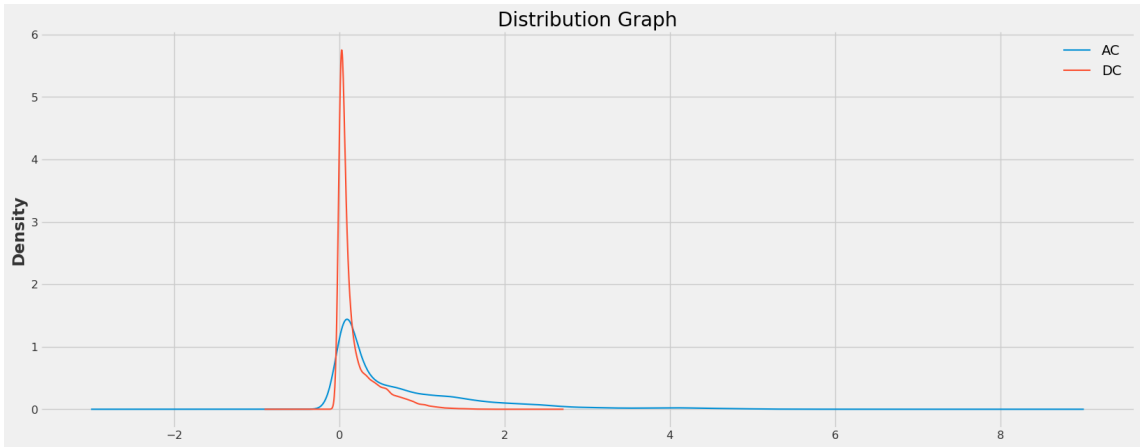
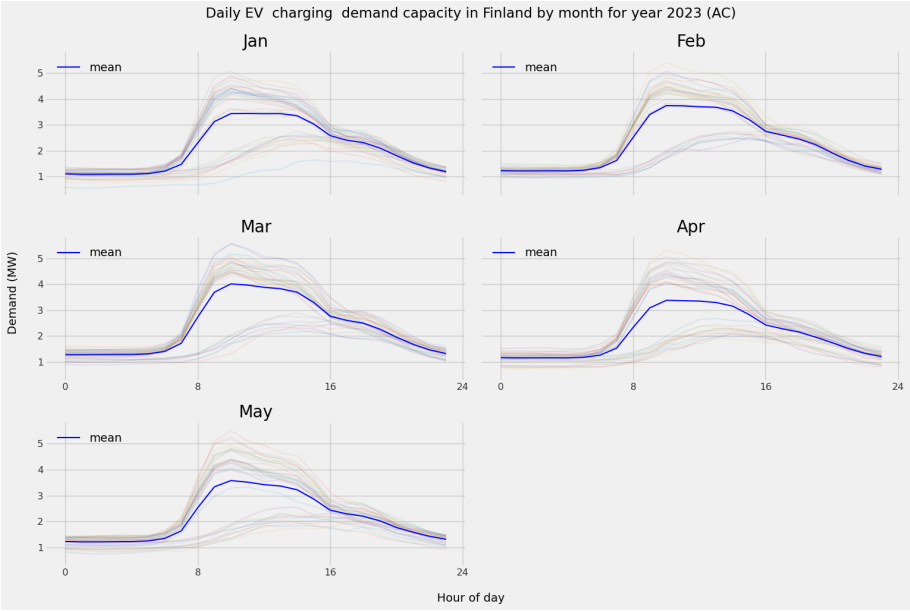


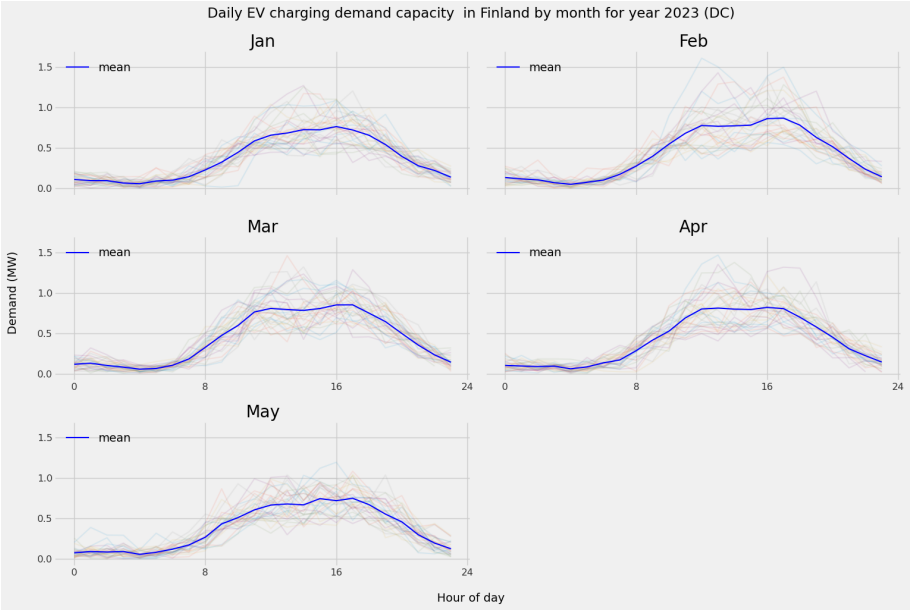

*Figure 15. Distribution Graph*

In 2023, Finland experienced the highest demand for electric vehicle (EV) charging capacity from 8 a.m. to 4 p.m. This trend remains consistent throughout the year, with a disproportionately high number of charging sessions during daytime hours. As a result,

peak power demand is highest during those hours, while overnight, there are fewer charging sessions and a lower peak power demand. Figure (16) illustrates this trend in the daily EV charging capacity market by month.




(a)

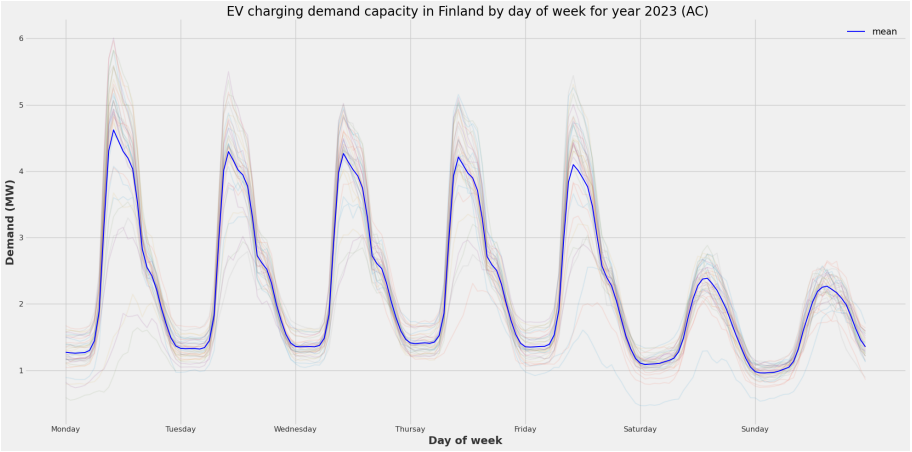



(b)

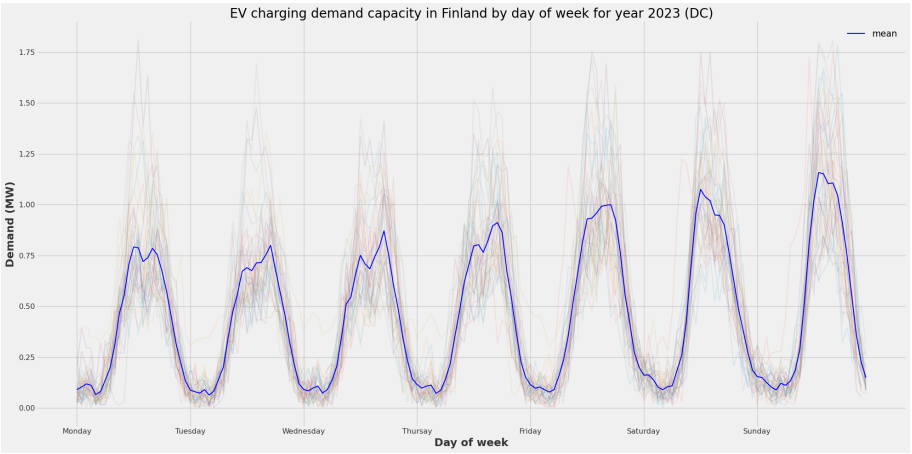
*Figure 16. Daily Ev charging Capacity Demand in Finland by month for the year 2023 (a) AC (b) DC*

Figure (17) shows the graph for AC and DC charging demand capacity by day of the week in Finland. We can see that weekdays have more AC capacity than weekends. People use charging stations on weekdays more frequently than on weekends when usage

is relatively lower. The reason can be that people usually leave their car at the office, and we can see that the charge is carried out more during office hours and working days than on holidays.



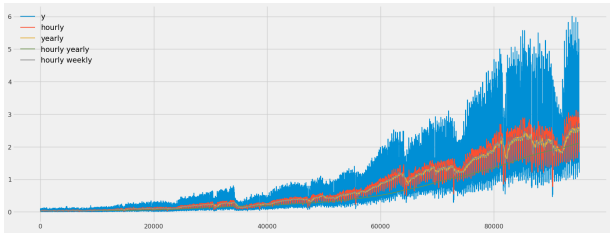

(a)



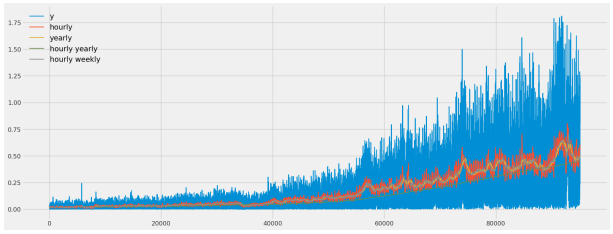

(b)

*Figure 17. (a) Weekly EV charging capacity in Finland For AC type (b) Weekly EV charging capacity Finland in for DC*

Figures (18 and 19) show the plot for SMA and EMA, which helps us look at the trend in the data. We can see an upward trend with the EMA, which helps us to identify the pattern in the data. The figures for AC and DC show that they are not stationary since the series indicate linear trends .

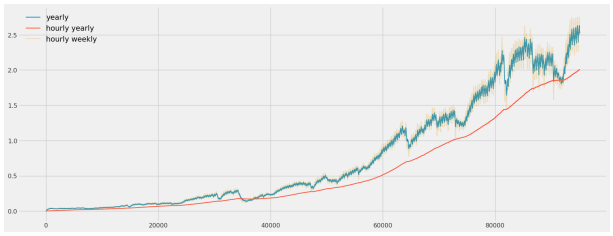



(a)

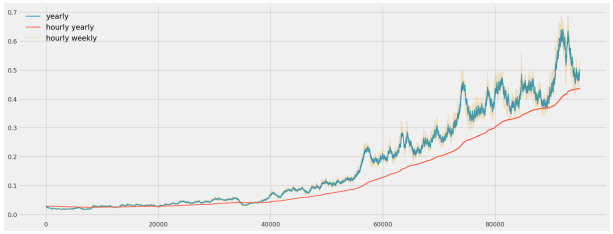



(b)

*Figure 18. (a) SMA for AC charging capacity (b) SMA for DC*




(a)




(b)

*Figure 19. (a) EMA for AC charging capacity (b) EMA for DC*

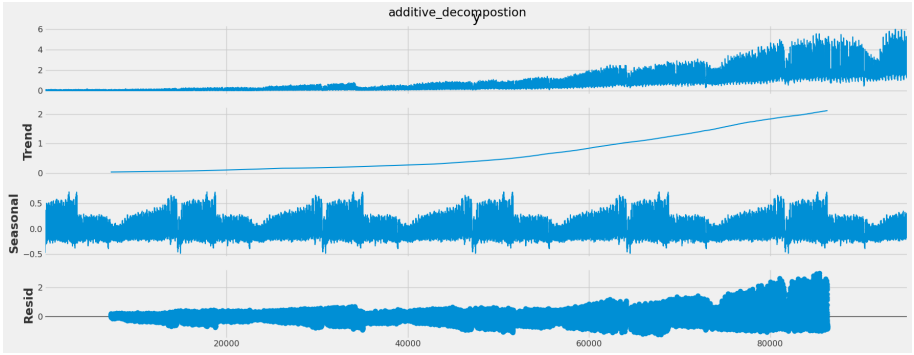
A double stationary test using Kwiatkowski-Phillips-Schmidt-Shin and augmented Dickey-Fuller statistical tests is carried out, showing that it is not stationary, as shown in Table (7). KPSS indicates nonstationary, and ADF indicates stationarity. Then, the series is difference stationary. Differencing is to be used to make series stationary.

*Table 7. Stationary test for Virta Dataset*

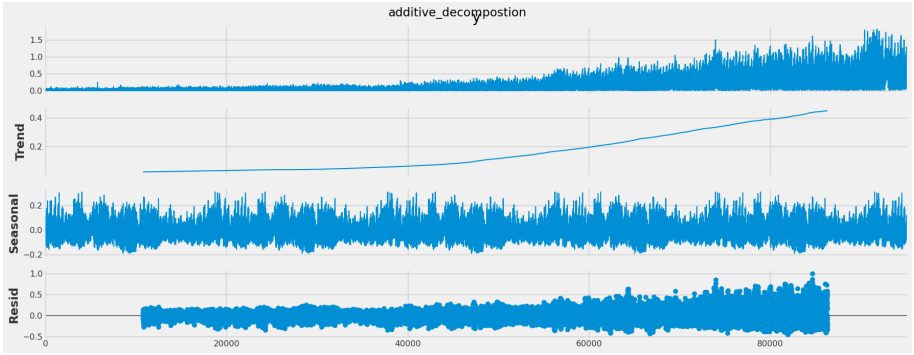
| | **Test statistics** | **p-value** | **Result** |
|---|---|---|---|
| **ADF** | -6.286563e+00 | 3.685787e-08 | Stationary |
| **KPSS** | 35.27668453184683 | 0.01 | Not stationary |

From Figure (20), additive decomposition decomposes the charge data into trend, seasonal, and residual components. These components can be combined to reconstruct the data shown in the top panel. Notice that the seasonal component changes slowly over time, so any two consecutive years have similar patterns, but years far apart may have different seasonal patterns. After subtracting the seasonal and trend-cycle components from the data, the residual components shown in the bottom panel of Figure (20) remain. We can see that the trend and seasonality information extracted from the series seems reasonable. The residuals are also interesting, showing periods of high variability in later years of the series. On doing the seasonality test with the Darts seasonality check, we can find multiple seasonality of :

( **24 48 72 96 120 144 168 192**)




(a)




(b)

*Figure 20. (a) AC decomposition (b) DC Decomposition with the period=8766*

Figure (21) of the Virta dataset shows the characteristics with high positive and negative correlations with the charge amount. Features whose range is above +0.02 or -0.02 that features will have some importance with the charge amount. Relevant features with a high positive correlation include the total_seconds, total_station_model_used, total_station_vendor_used, customer_charging_station_used, charged_session and station_active . The date-time and temperature related feature has a low correlation with the charge amount.

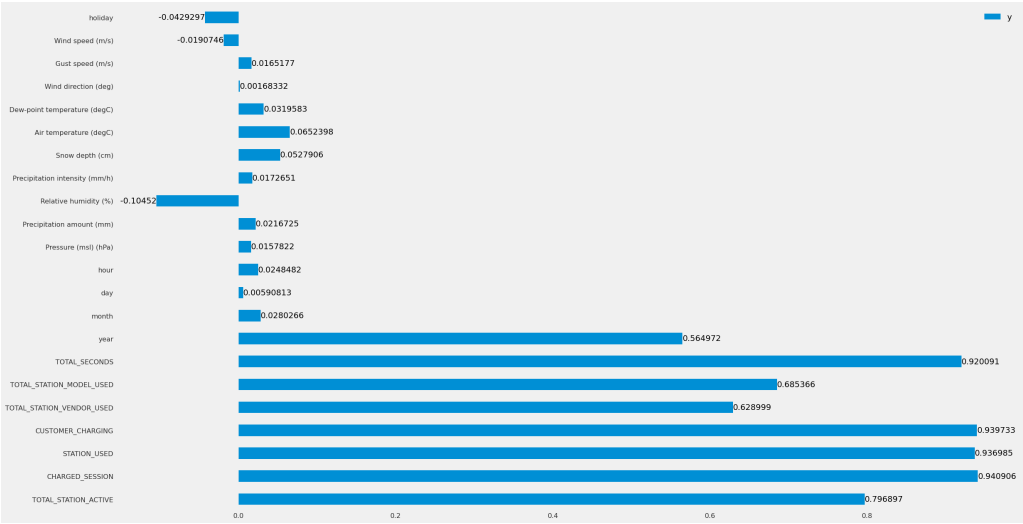


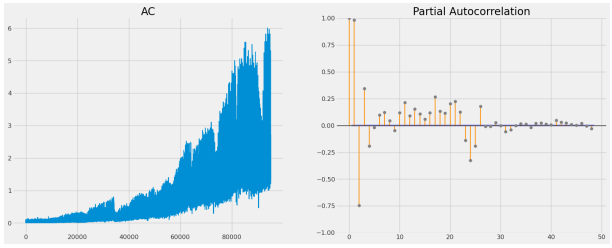

*Figure 21. Correlation plot for the Virta charge data*

### Autocorrelation Plot for Charge Data

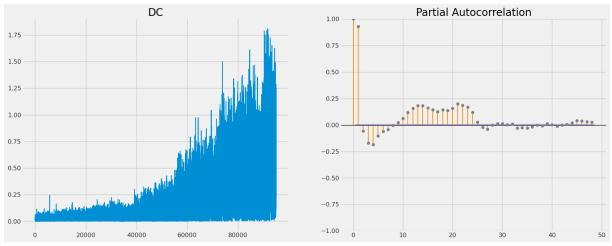
The autocorrelation plots (22 and 23) generated for the Virta dataset reveal noteworthy observations discernible through the analysis of the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF). In particular, numerous autocorrelations surpass the confidence interval of 95%, denoting statistical significance at the conventional threshold of 5%. Key observations from this analysis include:

- Several autocorrelations are significantly non-zero. Therefore, the time series is non-random.
- There is evident significance across various lags in ACF and PACF plots, as indicated by their notable elevation above the designated significance threshold.
- A discernible geometric decay pattern within the PACF plot further clarifies the

temporal autocorrelation dynamics inherent within the dataset.

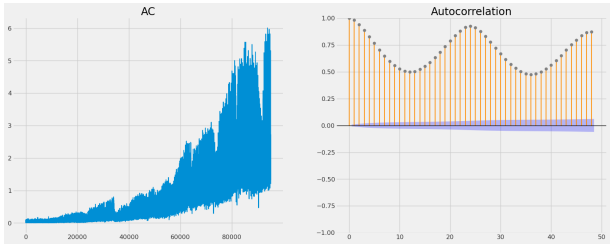



(a)

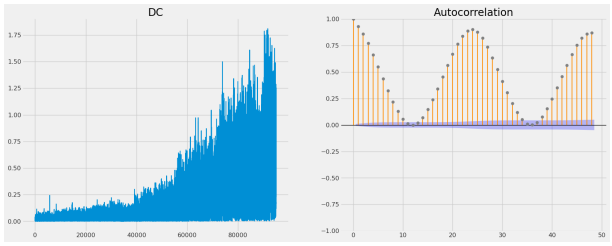



(b)

*Figure 22. (a) AC PACF (b) DC PACF with lags=48*




(a)




(b)

*Figure 23. (a) AC ACF (b) DC ACF with lags=48*

## 5.4 FCR-D Upward EDA

This section analyzed trend of FCR-D upward price and presented our findings in the following sections:

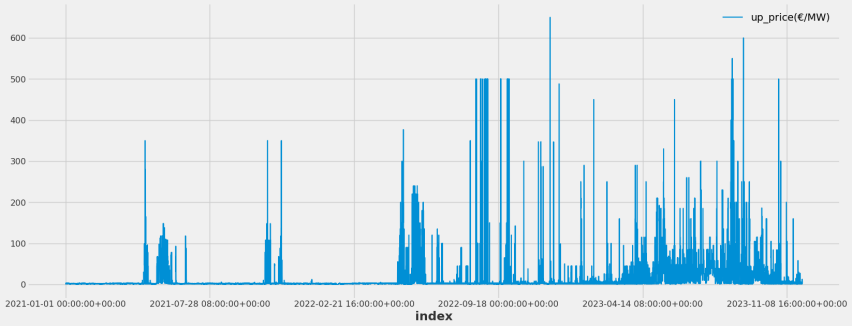



*Figure 24. FCR-D upward Price*

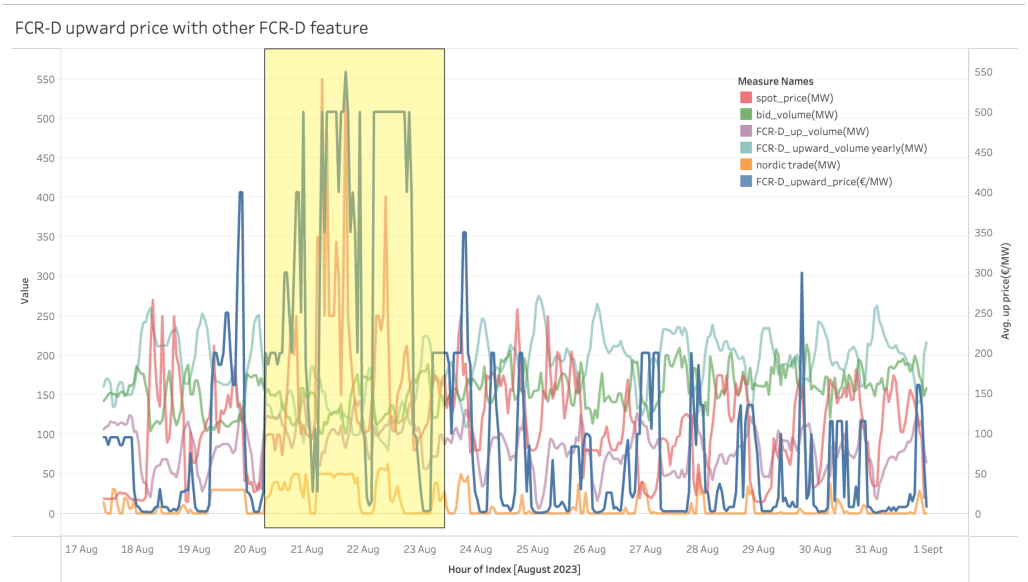
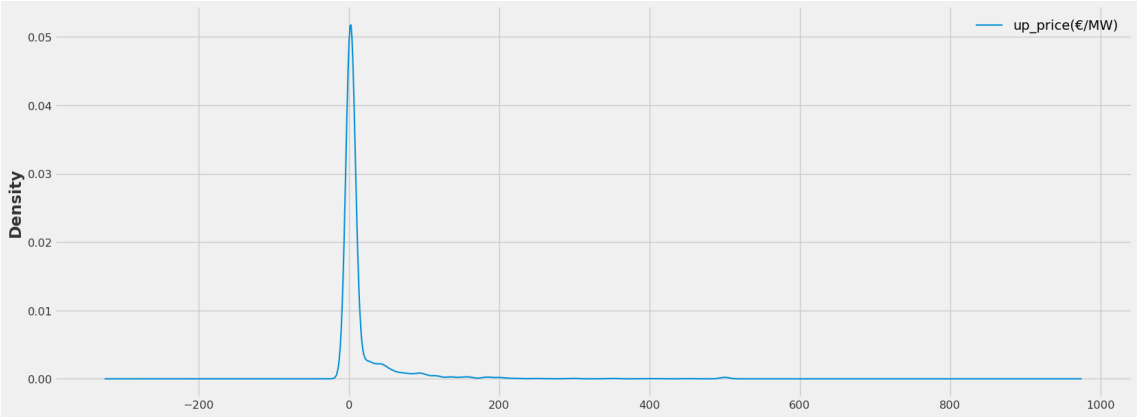
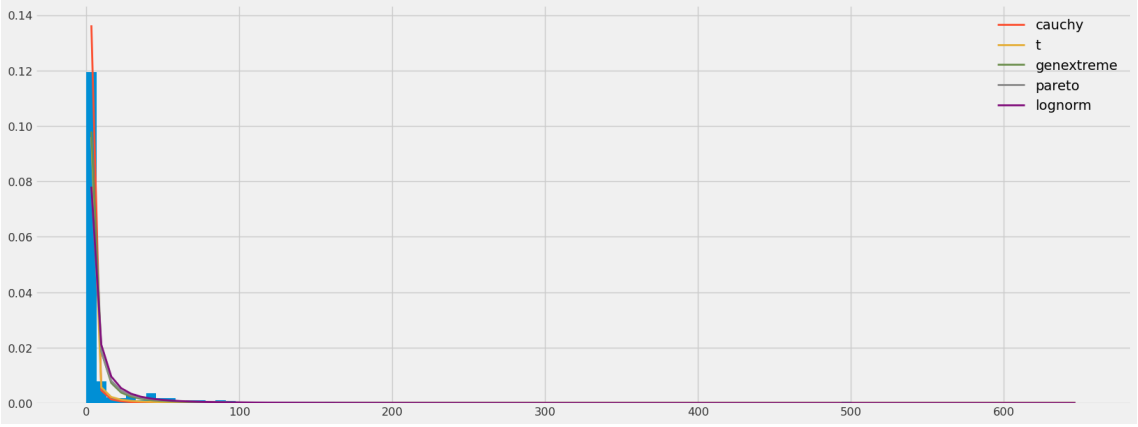



*Figure 25. FCR-D upward price with other features related to FCR-D*

Figures 24 and 25 show the time series graph for FCR-D upward price from 2021 to 2023. From Figure 24, we can see how the FCR-D upward price fluctuates. We can see the high prices during the year 2023 compared to 2021. the graph is irregular and has erratic data with high and low peaks. Figure 25 shows the pattern for the period taken from August 17 to the end of August. We can see the price from 2 to almost 600 euros. The graph also includes other features related to the FCR-D event. During the trend of the FCR-D upward price, spot prices increased, and Nordic trade was also high. However, the yearly bid volume for FCR-D was low, as was the hourly bid volume. This resulted in a low FCR-D upward hourly volume.




(a)




(b)

*Figure 26. (a) Distribution plot (b) distribution type test plot*

Figure 26 shows the distribution of the FCR-D upward. Figure 26 (b) shows a test to verify which distribution the graph follows. The distribution follows the Cauchy distribution. Cauchy distribution, or Lorentz distribution, is a continuous probability distribution that arises in various statistical contexts. It does not have a mean, variance, or higher moments defined. Despite its lack of a mean and variance, it does have a median and mode. In finance, investors use the Cauchy distribution to model the distribution of investment returns. It can better capture the heavy tails often observed in financial data than the Gaussian distribution.

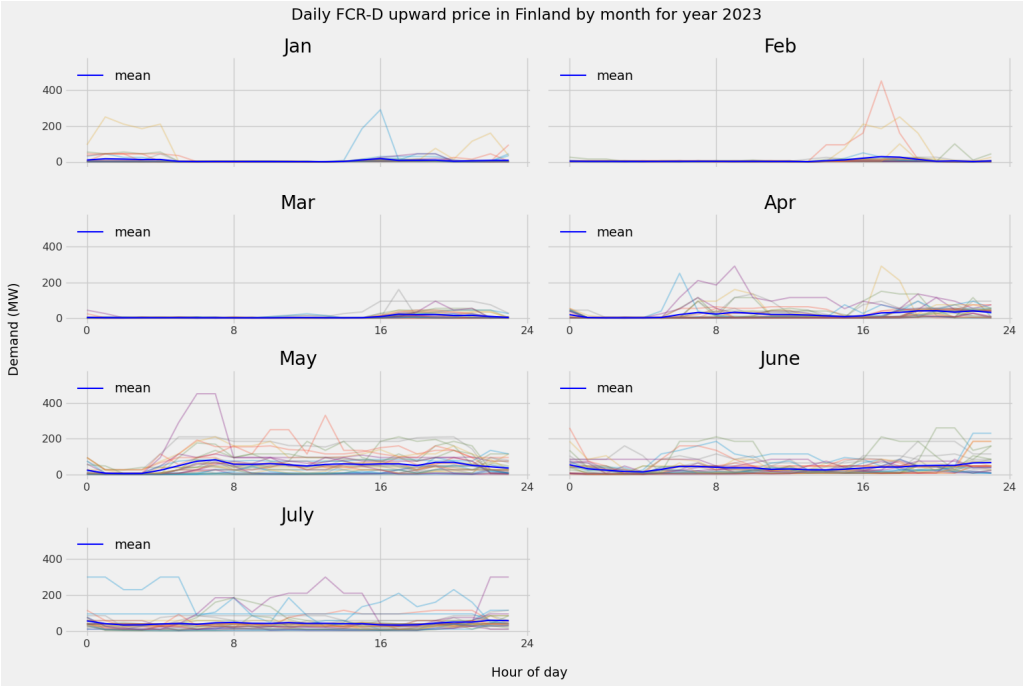



*Figure 27. Daily FCR-D upward price in Finland by month for the year 2023*

In Figure 27, we can see the prices of Daily FCR-D upward in Finland for different months in 2023. The blue line represents the average price for the given month. As we can observe, the prices are not constant and fluctuate abruptly. The FCR-D upward prices have varied considerably around their means. The price distributions for both services are positively skewed, with median prices below the mean and, occasionally, far exceeding the means. The graph shows that the prices fluctuate significantly during spring (April, May) and are higher in summer than in winter.

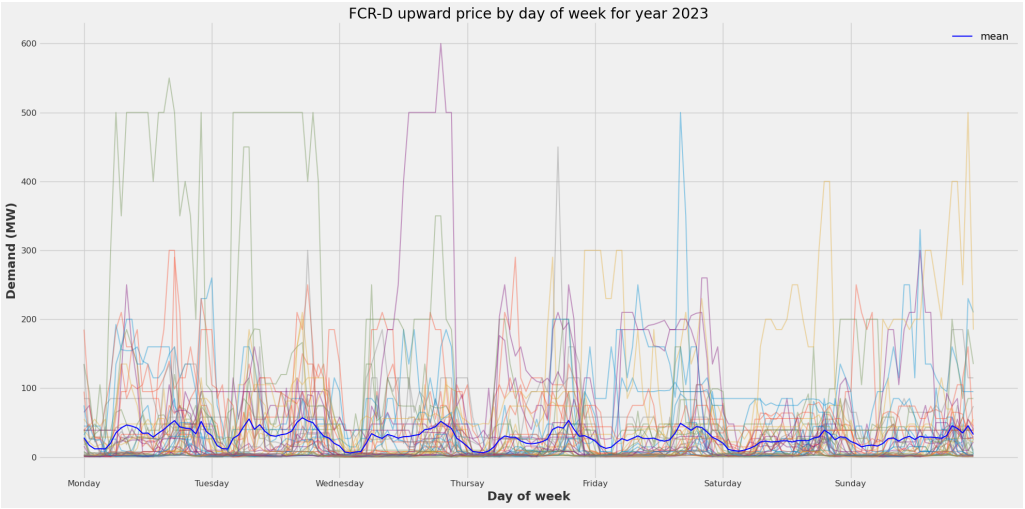
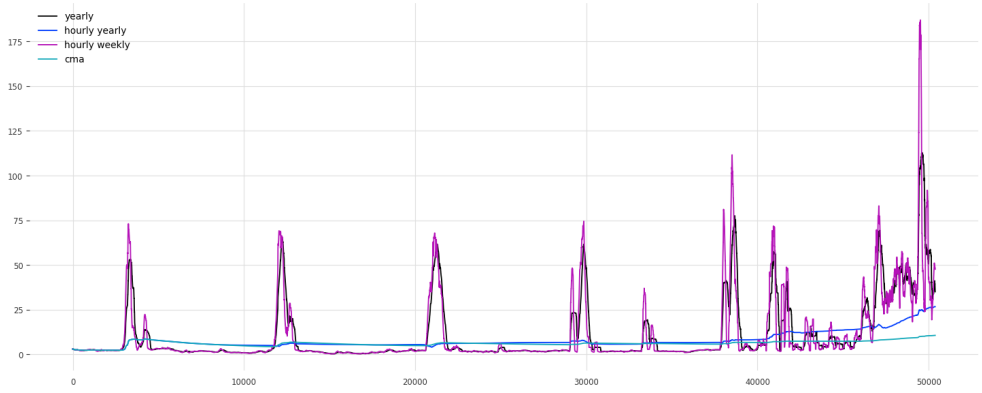
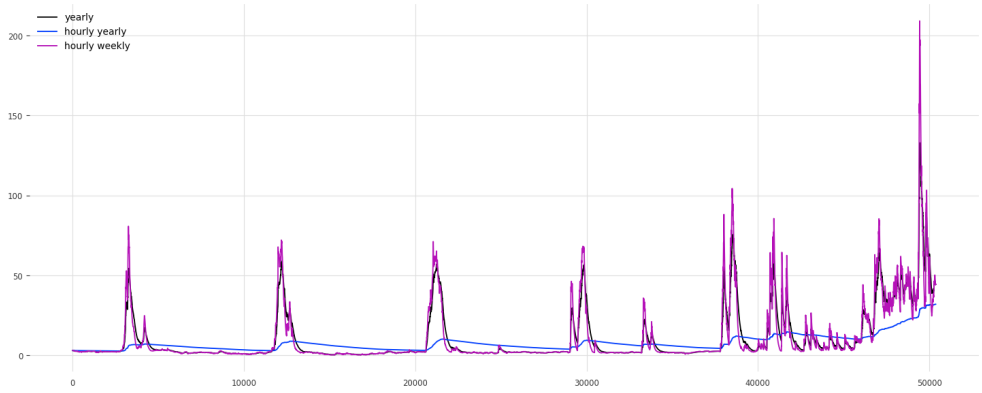



*Figure 28. FCR-D upward price by day of the week in Finland for the year 2023*

Figure 28 shows the graph for the price of the FCR-D upward by the day of the week. The blue line is the mean price for the weekday, and the others are the actual price for the given week of the day. From the figure, we can see higher peaks during the weekdays and higher fluctuations in the price above the mean.




(a)




(b)

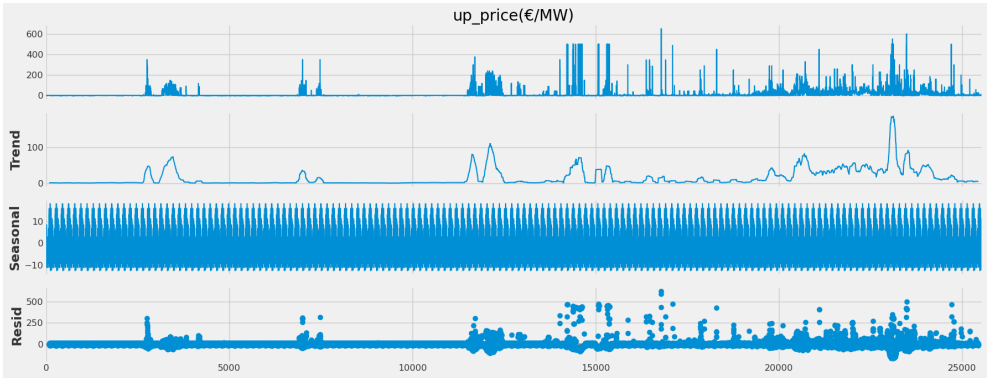
*Figure 29. (a) SMA plot (b) EMA plot*

Figure 29 shows the graph for the price of the FCR-D upward over the last three months. SMA and EMA help us understand the data's trends and patterns.

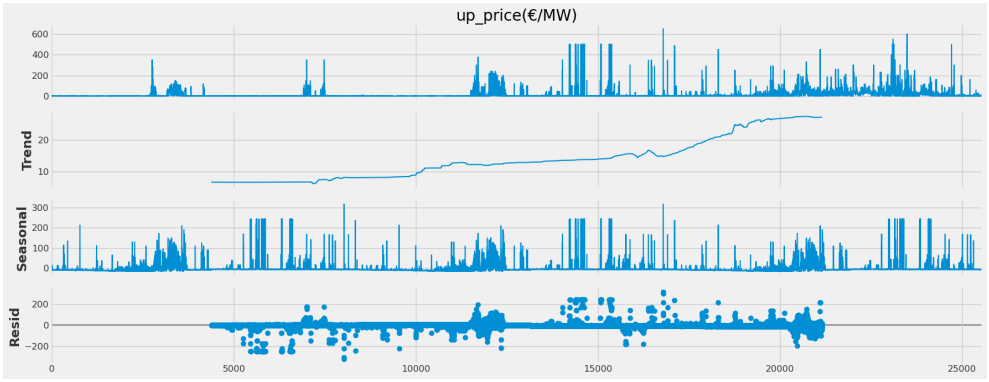
*Table 8. Stationary test for FCR-D upward*

| | **Test Statistics** | **p-value** | **Result** |
|---|---|---|---|
| **ADF** | -11.153482 | 0.000000 | Stationary |
| **KPSS** | 4.203506008415505 | 0.01 | Not stationary |

Table 8 shows the stationary test result for the price of the FCR-D upward. On conducting ADF, we get the result as stationary, and KPSS gives us Not Stationary. So we can conclude that the series is difference stationary. Differencing is to be used to make series stationary.



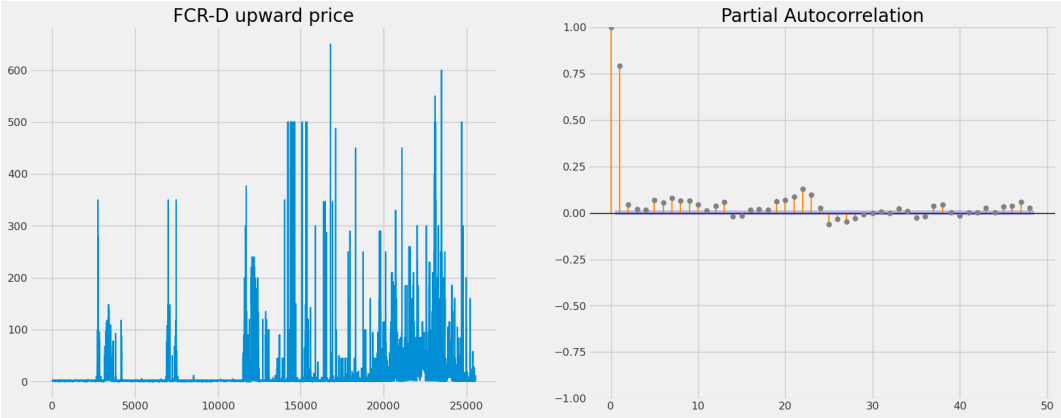

(a)



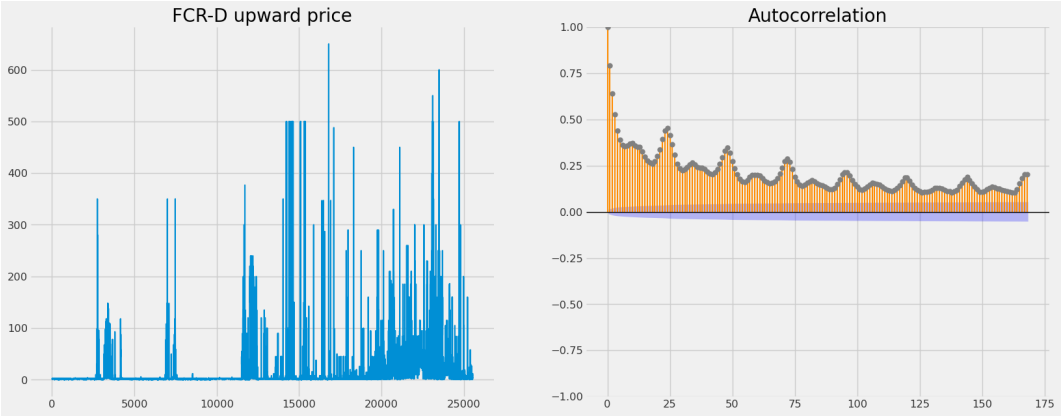

(b)

*Figure 30. (a) Decomposition with hourly weekly period= 168 (b) Decomposition with period=8766 (hour yearly)*

Figure 30 illustrates the decomposition of the price of the FCR-D upward into its constituent components trend, seasonal, and residual elements, with an additive decomposition approach over two distinct periods, 168 and 8766. With the more extended period of 8766, the seasonality insight becomes challenging. We can see it has a consistent trend for a duration, followed by a subsequent upsurge. When analyzed within the seasonal period 168, a cyclic pattern emerges characterized by a flat trend interspersed with periodic peaks, reverting to a flat trajectory. The seasonal component changes slowly, so any two consecutive years have similar patterns, but years far apart have different seasonal patterns. The seasonal trend showing a unique occurrence around 15000 may indicate overfitting. The remaining component in the bottom panel is the leftover seasonal and trend-cycle components subtracted from the data. Additionally, the residuals exhibit intriguing temporal dynamics, featuring intervals of heightened variability in the later years of the dataset. Conducting a seasonality test yields multiple seasonalities, denoted as : **(10, 24, 34, 38, 48, 59, 72, 82, 95, 106, 120, 132, 144, 154, 167, 178, 192)**.




(a)




(b)

*Figure 31. (a) PACF (b) ACF with lag =168*

In the ACF and PACF graph from Figure 31, we can see significant autocorrelation. We can make the following observations:

- Several autocorrelations are significantly non-zero. Therefore, the time series is non-random.
- From ACF and PACF, almost all lags are significant since they are well above the significance line.
- When we set the ACF parameter to 168, it shows high autocorrelation between adjacent and near-adjacent lags (lag=1 to lag=168).

Figure 32 shows the FCR-D upward price correlation plot with the other features. We can see how each feature correlates with the upward price of the FCR-D. Any features whose range is above +0.05 or below -0.05 that feature are taken with the bid price.

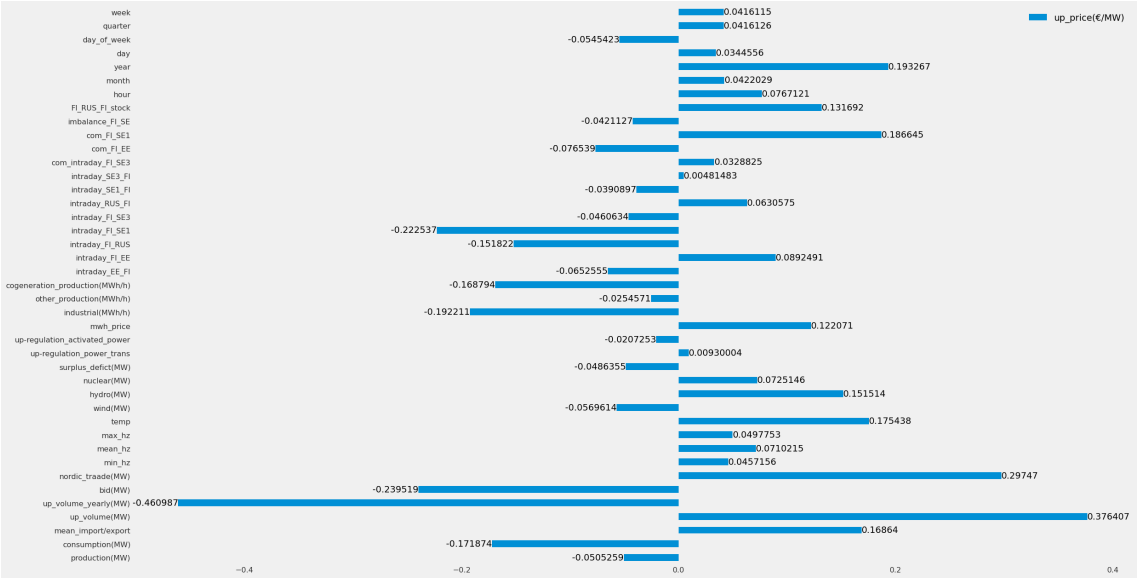



*Figure 32. Correlation plot of FCR-D upward price*

# 6. Experiments

This section details the steps to answer the research question concerning the previously described models. It details the tools of analysis and model evaluation.

## 6.1 Feature Selection and Preprocessing

Preprocessing data is crucial for forecasting tasks. Including too many parameters can create unhelpful forecasting correlations and cause the gradient descent to become stuck in local minima. We can divide the forecasting task into univariate and Multivariate.

For the multivariate task, we pass the other features as Covariates in darts (Herzen et al. 2022) and the nixtla library (Garza et al. 2022). Covariate series are time series that the models can take as inputs. Still, they will not forecast but provide additional information/context that can be useful to improve the prediction of the target series (variable we wish to predict in the future). The covariate types are shown in Figure 33.

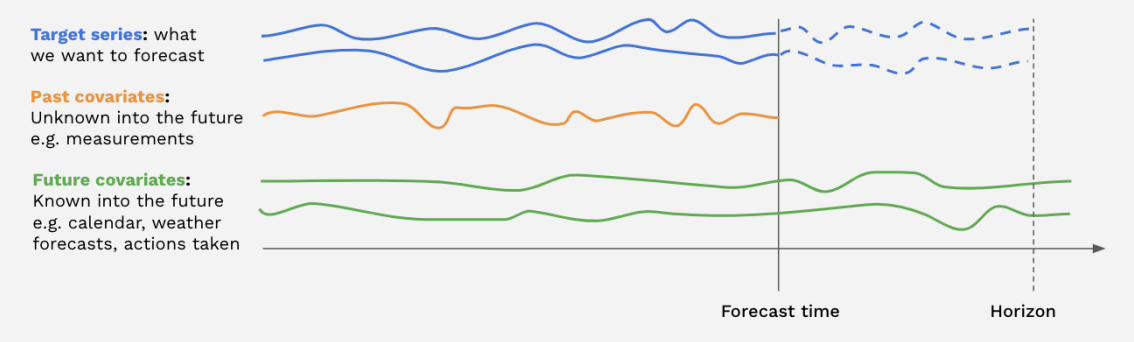



*Figure 33. Covariates types in time series*

We can pass the feature as :

- Past covariates: In the future, signals that require measurement may arise and be unknown.

- Future covariates: time series whose values are known into the future at prediction time (till forecast horizon). e.g., calendar data, holidays, weather forecasts, etc.

## 6.2 Training, Validation, and Test Data

We have divided the dataset into three main subsets, as shown in Figure 34:

- The training set (black graph) trains the model to learn the hidden features in the data. The training set should have different inputs so that the model is trained in all conditions and can predict any data sample that may appear in the future.

- During training, the model uses the validation set ( blue graph) to validate its performance. It is essential during hyperparameter tracking to help optimize the model's configuration. The validation set acts as a checkpoint, allowing for the detection of overfitting.

- The model uses the test set (purple graph) to assess its accuracy and precision on data that it has not previously encountered.

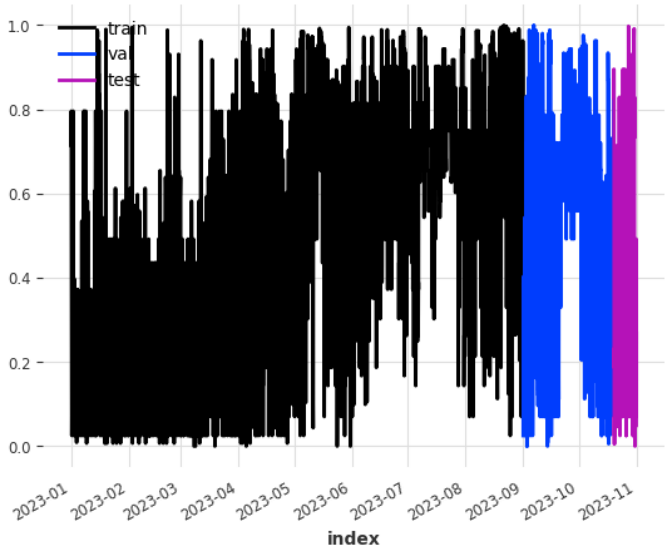



*Figure 34. Data splitting into Train(black), Evaluation(blue), and Test(purple)*

## 6.3 Hyperparameter Tuning

It is the process of selecting the optimal set of the optimal configuration for a machine learning model to achieve the best performance. Optuna will be employed for the hyperparameter tuning task in the thesis, as the selection of hyperparameters heavily influences the performance of deep learning models for time series forecasting, outperforming classical statistical and machine learning-based approaches. (Akiba et al. 2019). Op-

tuna is a hyperparameter optimization software framework that can quickly implement different state-of-the-art optimization methods to perform hyperparameter optimization rapidly with excellent performance. By default, Optuna implements a Bayesian optimization algorithm (TPE) (Watanabe 2023). Optuna enables users to adopt state-of-the-art algorithms for sampling hyperparameters and pruning unpromising trials. Compared to traditional methods such as grid search, Optuna dramatically speeds up optimization time and performance. It also allows users to plot optimization histories to better understand the model.

## 6.4 Training and Evaluation Metrics

As shown in Figure (35), time series have different error and evaluation metrics, and it's not easy to select one metric that is best for all tasks. It is essential not to look at one error metric when evaluating the model's performance since each metric has advantages and weaknesses. So, it always depends on the use case underlying data.

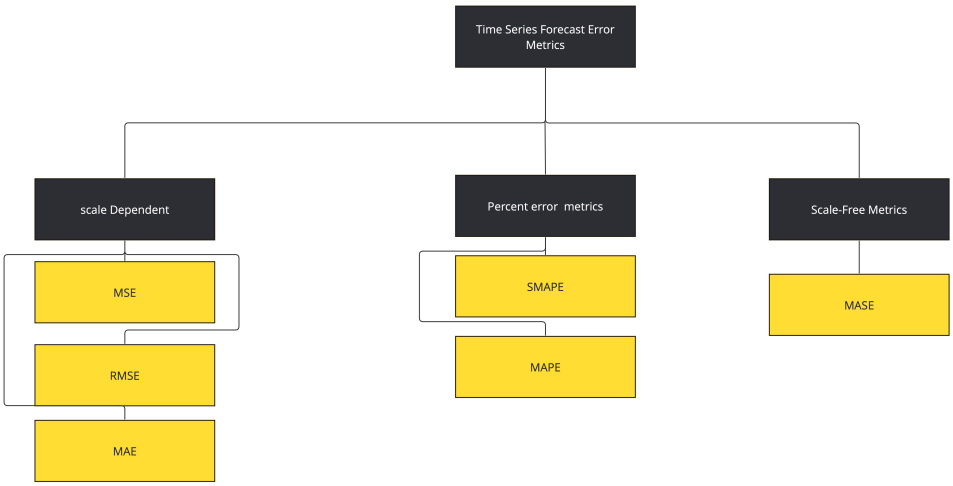



*Figure 35. Different Error Metrics*

### 6.4.1 Scale Dependent

When we express error metrics in the same units as the underlying data, they depend on the scale.

**Mean Absolute Error (MAE)**

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{17}$$

MAE is the average absolute difference between the actual ($y_i$) and predicted ( $\hat{y}_i$) values. If you want to penalize an outlier, it is not advisable to use MAE. Using MAE can help assess accuracy on a single series effectively. However, it is important to note that comparing MAE across different series is not feasible due to its scale-dependent nature, which makes it unsuitable for comparing accuracy on different scales.(Hyndman & Koehler 2006)

**Mean Squared Error (MSE)**

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{18}$$

MSE is the mean of the squared error. The squaring term in MSE heavily weights significant errors, which is suitable for focusing on large errors and putting more attention on outliers (huge errors)

**Root Mean Squared Error (RMSE)**

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{19}$$

Root Mean Squared Error is the square root of MSE, which avoids unit loss in MSE. It has the same advantages as MAE and MSE but is sensitive to outliers. According to Willmott & Matsuura (2005), using RMSE as a measure of error is inappropriate, and they recommend using MAE instead. Chai & Draxler (2014) recommended that the RMSE metric instead of MAE be used when evaluating and optimizing different models in the presence of a Gaussian error distribution.

### 6.4.2 Scale Independent

**Mean Absolute Percentage Error (MAPE)**

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i} * 100\right| \tag{20}$$

The mean absolute percentage error (MAPE) is a widely used metric in time series forecasting. It calculates the average absolute difference between the actual and predicted values divided by the true value. MAPE is scale-independent and used to compare forecast performance between different time series. MAPE also comes with some disadvantages. First, it generates infinite or undefined values for zero or close-to-zero actual values (Kim & Kim 2016). When using percentages, MAPE does not make sense. One should avoid using MAPE.

**Symmetric Mean Absolute Percentage Error (SMAPE)**

To avoid the asymmetry of the MAPE, Armstrong & Forecasting (1985) proposed the "adjusted MAPE".

$$\overline{\text{MAPE}} = 100\,\text{mean}\left(2\left|y_t - \hat{y}_t\right| / \left(y_t + \hat{y}_t\right)\right) \tag{21}$$

The absence of absolute denominator values in the current method can produce negative or infinite error metrics.
Makridakis (1993) proposed a similar metric and called it SMAPE.

$$SMAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{\left(|y_i| + |\hat{y}_i|\right)/2} \tag{22}$$

The SMAPE is the average across all forecasts made for a given horizon. Its advantages are that it avoids MAPE's problem of significant errors when y-values are close to zero and the large difference between the absolute percentage errors when y is more critical than y-hat and vice versa. Unlike MAPE, which has no limits, it fluctuates between 0% and 200%. The modified version of SMAPE ensures that the metric's results will always

be between 0% and 100%:

$$sMAPE = \frac{100\%}{n} \sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{(|y_i| + |\hat{y}_i|)} \tag{23}$$

### 6.4.3 Scale-Free Error Metrics

Hyndman and Koehler (Hyndman & Koehler 2006) proposed a new metric — the scale-free error metric. Their idea was to scale the error based on the in-sample MAE from a naive (*random walk*) forecast method.

#### Mean Absolute Scaled Error (MASE)

$$\begin{aligned} MASE &= \frac{MAE}{MAE_{\text{in-sample , naive}}} \\ MAE &= \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \end{aligned} \tag{24}$$

To calculate MASE, you need to divide the MAE by the MAE of a naive benchmark that is based on our training data and is in-sample. Values of MASE greater than 1 indicate that the forecasts are worse, on average, than in-sample one-step forecasts from the *naive model*. MASE will not produce infinite or undefined values, making it a suitable metric for time series data with zeros except in the irrelevant case where all historical data are equal. However, MASE is still vulnerable to outliers (Chen et al. 2017). To obtain a consistent estimator of the series' scale, we need to assume that the time series difference between periods is stationary and use the scaling factor.

Table 9 compares the different evaluation metrics on various properties.

*Table 9. Evaluation metrics properties*

| | **MAE** | **MSE** | **RMSE** | **MAPE** | **SMAPE** | **MASE** |
|---|---|---|---|---|---|---|
| **Simplicity** | High | Medium | Medium | High | Medium | Medium |
| **Interpretability** | High | Low | High | High | High | Low |
| **Scale-Invariant** | No | No | No | No | No | Yes |
| **Penalizes large errors** | No | Yes | Yes | No | No | No |
| **Outlier sensitivity** | Medium | High | High | Medium | Medium | Medium |
| **Symmetric** | Yes | Yes | Yes | No | Yes | Yes |
| **Original scale** | Yes | No | Yes | No | No | No |
| **Sensitivity to division by 0** | No | No | No | Yes | Yes | Yes |

## 6.5 Development Tools

Python is the primary programming language for working with time series data and forecasting model development, and it has become the de facto programming language in AI. The additional packages used for the project are

- SciKit Learn for feature scaling,
- Numpy and Pandas, for data transformation
- Matplotlib and Plotly for visualization
- Optuna for hyperparameter tuning
- Darts and nixtla and modeling and forecasting.

# 7. Results

Forecasting time series data is critical in various fields, often categorized into univariate and multivariate approaches. This study investigates the efficacy of different forecasting models applied to both methodologies across two datasets. We prioritize identifying models suitable for long-term forecasting to achieve accurate long-horizon predictions. Specifically, we evaluate models with a horizon of h=720, signifying a substantial forecasting duration. We use Bayesian optimization with the Optuna framework to optimize performance and find the best algorithms. The dataset is partitioned into training, evaluation, and test sets to facilitate rigorous model assessment and validation. This systematic approach enables a comprehensive comparison of forecasting models, examining their effectiveness over prolonged periods.

We used a machine with the following specifications to perform the training and forecasting of the models:

- GPU, specifically the NVIDIA RTX 2080Ti with 11GB of memory capacity.
- The CPU for processing tasks is an Intel® Core™ i9-9900KS CPU running at 4.00GHz with 16 cores.
- The operating system is a Linux-based system, specifically Ubuntu version 22.04.3 LTS.

## 7.1 Electric Vehicle Charge Demand Forecasting

We utilize Mean Squared Error (MSE) as the loss function in our training process. We consider all possible pairs of look-back and horizon that we can derive from the training period in each epoch. It is important to note that two mini-batches may have overlapping time points. Prior research on long-term forecasting commonly used this approach (Wu et al. 2021). For each (look-back, horizon) pair from the test set, we evaluate the model through rolling validation/evaluation.

### 7.1.1 Univariate Forecasting For Charge Demand

We are exploring different modeling algorithms and a unified model approach to determine the optimal predictive framework for forecasting the power curve of both AC and DC charging stations in Finland. To evaluate the performance of machine learning models, we commonly use various metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Symmetric Mean Absolute Percentage Error (SMAPE).

Table (10) shows the model's performance evaluation with the default settings of the darts and nixtla library model. NHiTS, TiDE, and MSTL achieve notably higher accuracy rates among the various methods. The table shows that NHiTS performs well with the AC-type chargers and Tide with the DC-type power graph forecasting. Therefore, in hyperparameter optimization, we focus specifically on MSTL, which has a statistical foundation, and NHiTS, which employs a deep learning framework. We aim to refine and tune these methods further to enhance their performance.

*Table 10. Univariate model performance on charge power profile dataset*

| Index | MAE | MSE | RMSE | SMAPE |
|---|---|---|---|---|
| **MSTL_AC** | 0.491203 | 0.4691 | 0.684909 | 15.921484 |
| **ETS_AC** | 1.006908 | 2.270910 | 1.506954 | 39.419789 |
| **nhits_AC** | 0.347398410659474 | 0.34731327849630333 | 0.5893329097346451 | 13.096508788823868 |
| **Tide_AC** | 0.3740527214143861 | 0.4223811126071762 | 0.6499085417250463 | 14.194860664246942 |
| **Dlinear_AC** | 0.567929019933126 | 0.7388673493074928 | 0.8595739347534294 | 22.377008217379014 |
| **nhits_DC** | 0.1261951236417534 | 0.02846478114790042 | 0.16871508867881502 | 34.082422833018065 |
| **Tide_DC** | 0.12047099043102591 | 0.0255768517539771 | 0.15992764537120246 | 32.8190611593662 |
| **Dlinear_DC** | 0.13520005187575135 | 0.03245776927450568 | 0.18016039874097103 | 36.83964803445115 |
| **MSTL_DC** | 0.201352 | 0.075343 | 0.274487 | 32.416013 |
| **ETS_DC** | 0.355107 | 0.235080 | 0.484850 | 89.434685 |

**MSTL Approach**:

In the process of parameter optimization utilizing the MSTL method, careful adjustment of the parameter known as **seasonal_length** is essential, which governs the length of the seasonal patterns within the time series data.

According to Table (11), we can identify the optimal configuration by setting the seasonal length parameter within the range of **[24, 168]**. This setting leads to significantly reduced SMAPE values, indicating enhanced forecasting accuracy and improved robustness in capturing the underlying seasonal dynamics of the data. We employed the MSTL method using the Statforecast library from nixtla, with the trend component forecasted through an AutoETS model.

*Table 11. Parameter to fine-tune for MSTL*

| **Seasonal_length** | [24,48,96] | [24,168] | [24,48] | [24,48,168] | [24,192] | [24,168,192] | [12,24] | [12,24,168] |
|---|---|---|---|---|---|---|---|---|

Based on the data presented in Table (12), we can observe the performance of the model trained with the best hyperparameters. The model trained with these hyperparameters decreased the AC SMAPE value from 15.921 to 7.1592, accounting for a 55.02% reduction. The DC SMAPE value also reduced from 32.41 to 30.72, accounting for 5.2% of the RMSE reduction.

*Table 12. The MSTL model evaluation metrics table with the best parameter.*

| **Model** | **MAE** | **MSE** | **RMSE** | **SMAPE** |
|---|---|---|---|---|
| **MSTL_AC** | 0.169554 | 0.059708 | 0.244352 | 7.159238 |
| **MSTL_DC** | 0.112292 | 0.022241 | 0.244352 | 30.723868 |

Figures 36 and 37 show the forecasting capabilities of the MSTL model when confronted with data it has not previously encountered. In the case of the AC dataset, it is evident that the MSTL model demonstrates remarkable accuracy in its forecasting capabilities. Similarly, when analyzing the DC dataset, the MSTL model exhibits a comparable level of precision in forecasting outcomes on unfamiliar data points.

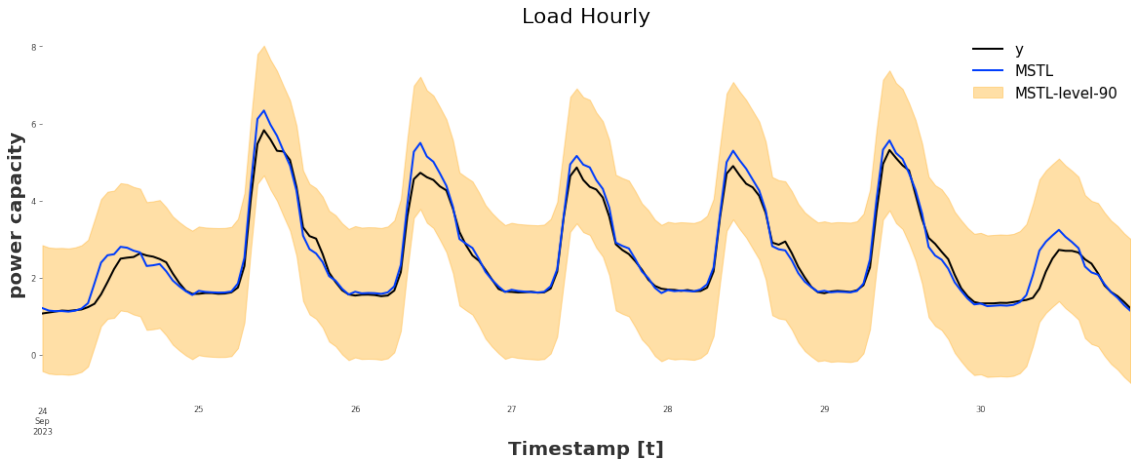



*Figure 36. MSTL AC*

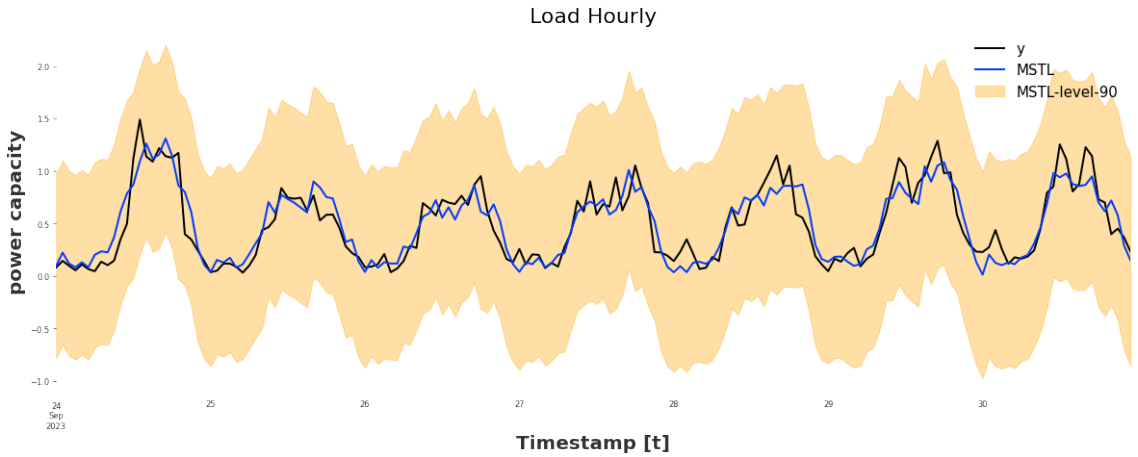



*Figure 37. MSTL DC*

## NHiTS Approach

NHiTS employs a diverse array of parameters for tuning purposes, as detailed in Table (13), to enhance the fine-tuning process. This fine-tuning process determines and highlights the optimal parameter configuration in the designated column labeled **best_param**. The cumulative duration expended to get the best parameter amounted to approximately 125 minutes across 20 epochs of training.

*Table 13. Parameter used to fine-tune the NHiTS model*

| | **Description** | **Parameter** | Best_param |
|---|---|---|---|
| **input_size** | input_size: int,<br>autoregressive inputs size, y=[1,2,3,4]<br>input_size=2 ->y_[t-2:t]=[1,2]. | [48,72,96,128,336,720,1440] | 336 |
| **n_pool_kernel_size** | list with the size of the windows<br>to take a max/avg over. | [[2,2,2],[4,4,4],[8,8,8],[8,4,1],<br>[16,8,1],[2,2,1],[4,4,2],[16,8,4]] | [[2,2,2]] |
| **mlp_unit** | Structure of hidden layers for each stack type | [[[512,512],[512,512],[512,512]],<br>[[256,256],[256,256],[256,256]],<br>[[128,128],[128,128],[128,128]],<br>[32,32],[32,32],[32,32]] | [[[512,512],<br>[512,512],<br>[512,512]]] |
| **activation** | activation function | [‘ReLU’, ‘Softplus’, ‘Tanh’, ‘SELU’,<br>‘LeakyReLU’, ‘PReLU’, ‘Sigmoid’] | tanh |
| **n_blocks** | Number of blocks for each stack | [[1,1,1],[2,2,2],[4,4,4],<br>[8,8,8],[6,6,6][16,16,16] | [8,8,8] |
| **dropout_prob_theta** | Dropout for NHITS basis. | [0,1] | 0.48782 |
| **scaler_type** | type of scaler for temporal inputs normalization | ["standard","minmax","robust"] | standard |
| **pooling_mode** | input pooling module type | [‘MaxPool1d’, ‘AvgPool1d’]. | MaxPool1d |
| **interpolation_mode** | interpolation basis | [‘linear’, ‘nearest’] | "nearest" |
| **n_freq_downsample** | list with the stack’s coefficients<br>(inverse expressivity ratios) | [[4,2,1],[8,4,2],[16,8,4],[2,2,1]] | [4,2,1] |

Figure 38 shows the hyperparameter importance of the model and which model parameters are highly important for training the model.

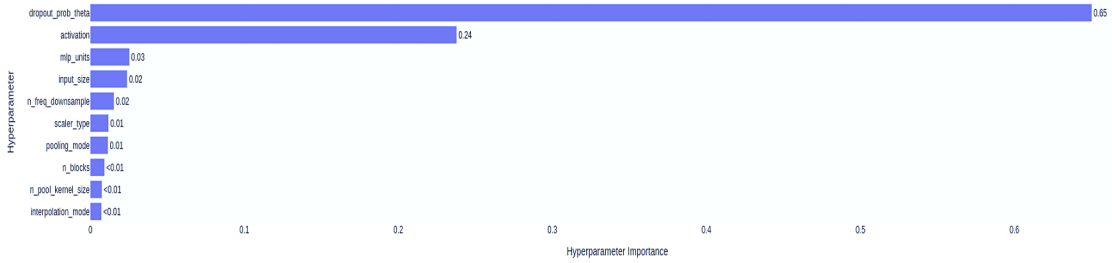



*Figure 38. NHiTS hyperparameter importance*

The training process utilized a batch size of 128 and an output horizon of 720, with Mean Absolute Error (MAE) as the loss function. The model was trained under specific parameter configurations, as presented in Table 13.

*Table 14. The model evaluation metrics table trained with the best hyperparameter for NHiTS*

| **Model** | **MAE** | **MSE** | **RMSE** | **SMAPE** |
|---|---|---|---|---|
| **NHITS_AC** | 0.165617 | 0.063177 | 0.251351 | 6.486849 |
| **NHITS_DC** | 0.116378 | 0.024488 | 0.6485 | 30.780519 |

After implementing the NHiTS methodology, we observed a modest improvement in the performance of our forecasting model compared to the MSTL approach shown in Table 14. However, when we evaluated the performance for DC forecasts, we found that the MSTL method outperformed NHiTS by a slight margin of 0.1838%, as indicated by the SMAPE metric.

Figures 39 and 40 show the forecasting capabilities of the NHiTS model for both AC and DC demand for the 720 horizon on unseen data.

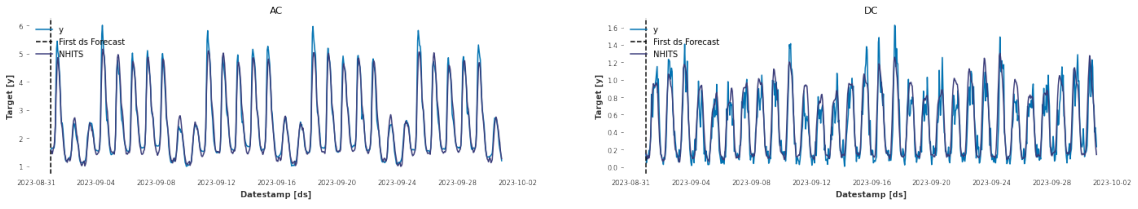
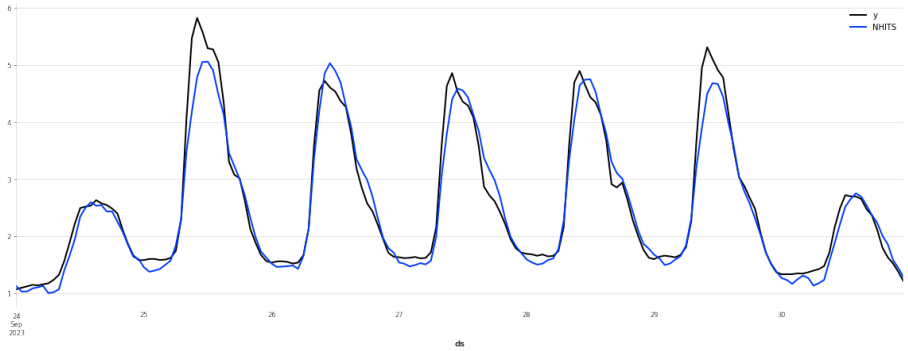
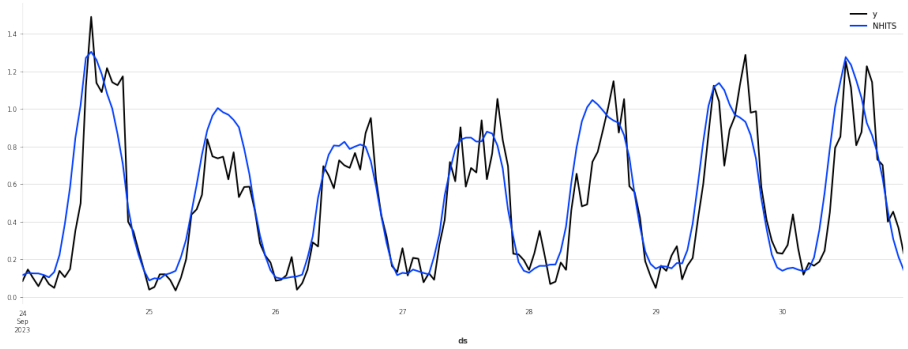



*Figure 39. NHiTS forecast*




(a)




(b)

*Figure 40. Forecast on unseen data for NHiTS (a) AC power graph (b) DC power graph*

### 7.1.2 Multivariate Forecasting for Charge Demand

In multivariate forecasting, a comprehensive approach involves training three distinct MLP-based architectures on previously unseen data and predicting the same set of unseen data. The outcomes of these efforts are detailed in Table (14), providing a comparative analysis of the results obtained from all three models using various evaluation metrics on scaled data. Upon inspection of the evaluation table, it becomes evident that the TiDE architecture performs better than the other models, particularly in forecasting AC and DC power graphs. Given these findings, it is reasonable to conclude that **TiDE** is optimal for hyperparameter tuning, promising enhanced forecasting accuracy and reliability.

*Table 15. Multivariate forecasting with the different model evaluation metrics with scaled data*

| | NHiTS AC | TiDE AC | Dlinear AC | NHiTS DC | TiDE DC | Dlinear DC |
|---|---|---|---|---|---|---|
| **MAE** | 0.074476 | 0.050259 | 0.082944 | 0.075127 | 0.052155 | 0.076228 |
| **MSE** | 0.013366 | 0.004776 | 0.012060 | 0.010527 | 0.005195 | 0.010531 |
| **RMSE** | 0.115611 | 0.069111 | 0.109817 | 0.102600 | 0.072079 | 0.102620 |
| **SMAPE** | 16.382371 | 11.780025 | 20.973077 | 36.049018 | 28.454319 | 49.09355 |

**TiDE Approach**:

For the Task of Multivariate time series Forecasting, compared with other models, TiDE shows a Promising result. So, TiDE is selected for the hyperparameter tuning process to find the best parameter that can yield an optimal model that minimizes the loss function.

Table (16) shows the hyperparameter optimization task with Optuna. Through fine-tuning, we get the optimal parameters for our model training. The total duration for hyperparameter tuning was 395 minutes and 28 seconds to traverse the parameter landscape and converge upon the most promising configuration.

*Table 16. Hyperparameter tuning for the TiDE model*

| Parameter | Description | Values | Best_value |
|---|---|---|---|
| **input_chunk_length** | Number of time steps in the past to take as a model input (per chunk) | [72,96,12,8,196] | 196 |
| **num_encoder_layers** | The number of residual blocks in the encoder. | [1,2,4,6,8,16,32] | 1 |
| **num_decoder_layers** | The number of residual blocks in the decoder. | [1,2,4,6,8,16,32] | 16 |
| **decoder_output_dim** | The dimensionality of the output of the decoder | [8,16,32,64,48,72,128,256] | 72 |
| **hidden_size** | The width of the layers in the residual blocks of the encoder and decoder. | [16,64,48,72,128,256,512] | 48 |
| **temporal_width_past** | The width of the layers in the past covariate projection residual block | [4,8,16,32,48,64,72,168,128,256] | 72 |
| **temporal_width_future** | The width of the layers in the future covariate projection residual block | [2,4,6,8,16,32,48,64,72,128,168,256] | 64 |
| **temporal_decoder_hidden** | The width of the layers in the temporal decoder. | [8,16,32,48,64,72,128,256] | 72 |
| **dropout** | Dropout probability is to be used in fully connected layers. | [ 0.0, 1] | 0.2696966 |
| **use_layer_norm** | use layer normalization in the residual blocks. | [True,False] | True |
| **user_reversible_instance_norm** | use reversible instance normalization RINorm against distribution shift . It is only applied to the features of the target series and not the covariates. | [True,False] | True |

During model training, the **add_encoders** section utilizes parameters related to past and future covariates to train the model. Table (17) below displays the model optimizers used for gradient descent optimization.

*Table 17. An additional parameter for model Training with TiDE*

| | |
|---|---|
| **use_static_covariates** | True |
| **optimizer_kwargs** | {'gradient_clip_val': 1,'max_epochs': 20,<br>'accelerator': 'gpu', 'callbacks': []} |
| **lr_scheduler_cls** | torch.optim.lr_scheduler.ExponentialLR |
| **lr_scheduler_kwargs** | {'gamma': 0.999} |
| **batch_size** | 128 |
| **random state** | 42 |
| **add_encoders** | {'cyclic': {'future': ['month', 'day', 'hour', 'weekday', 'quarter', 'day_of_week']},<br>'datetime_attribute': {'future': ['year']},<br>'position': {'past': ['relative'], 'future': ['relative']},<br>'transformer': Scaler} |

Figure (41) shows the importance of the hyperparameter in minimizing the error when trained with different sets of parameters to find the best parameter.

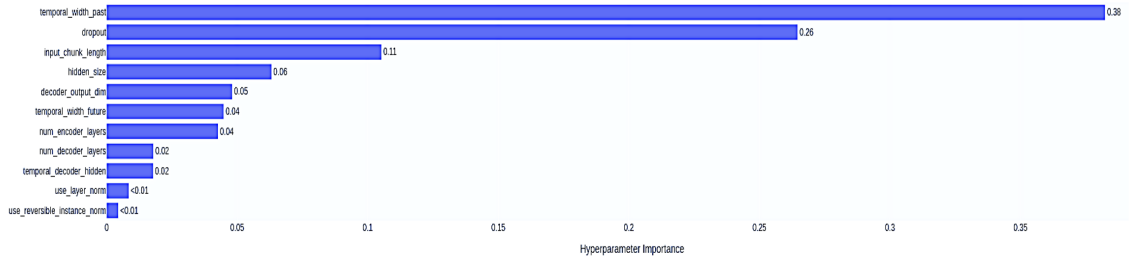



*Figure 41. Hyperparameter importance for the TiDE*

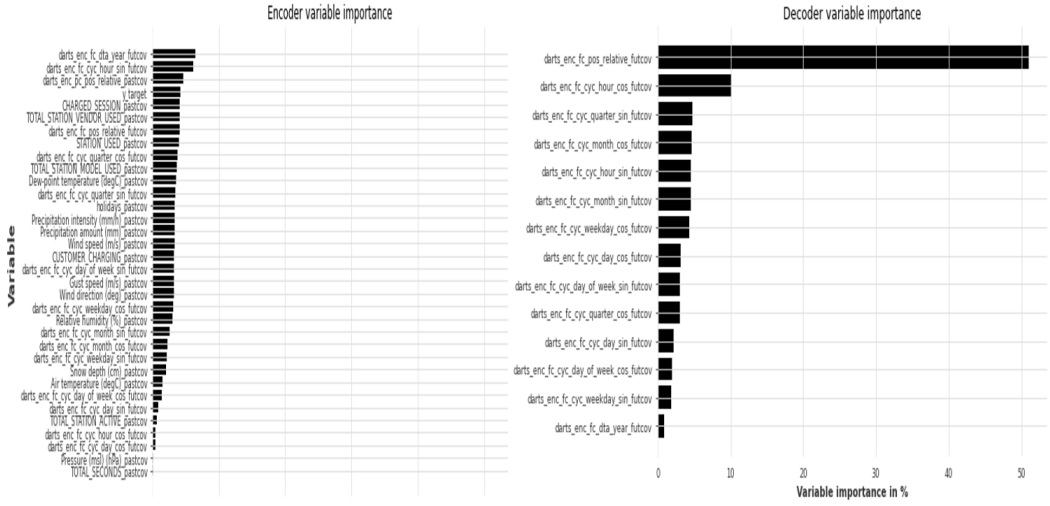



*Figure 42. Feature Importance for the charge data*

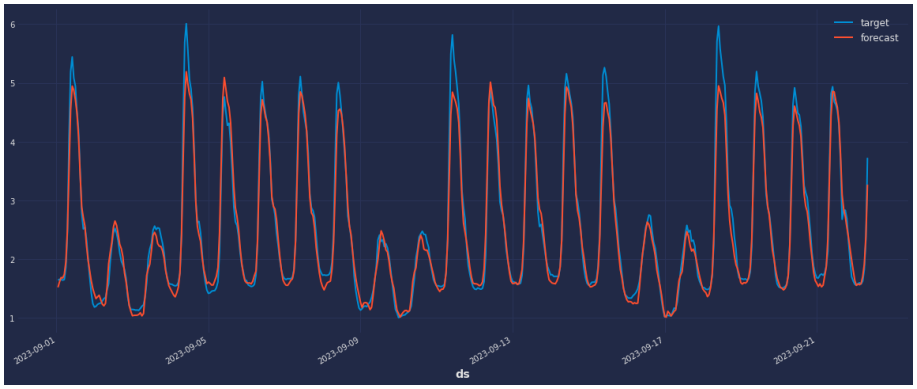
Figure (42) displays the crucial features required for training a model specifically designed to predict data points not encountered before. Each feature in this visualization holds its unique importance in the training process, working together to enhance the model's ability to forecast outcomes effectively.

*Table 18. TiDE model evaluation result*

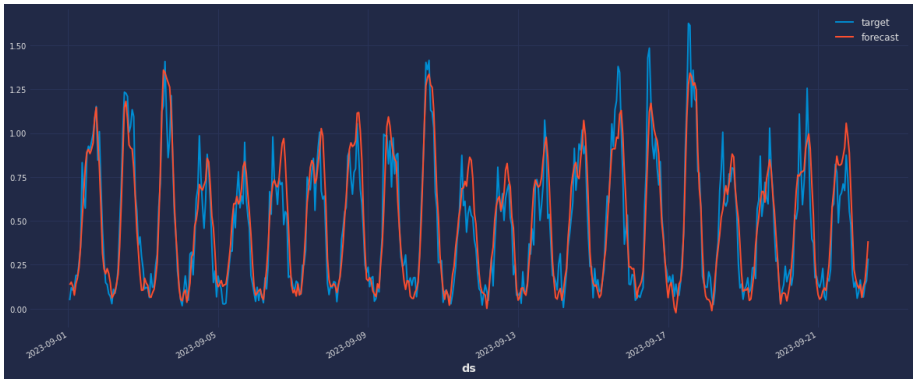
| | MAE | MSE | RMSE | SMAPE |
|---|---|---|---|---|
| **Tide_AC** | 0.150418 | 0.048977 | 0.221306 | 5.836538 |
| **Tide_DC** | 0.109839 | 0.011626 | 0.133618 | 28.454319 |

Table (18) presents the TiDE model's performance when forecasting with different evaluation metrics. The TiDE model's multivariate methodology outperforms the univariate NHiTS and MSTL approaches. In the univariate scenario, the NHiTS and MSTL models achieved the best forecasted SMAPE values of 6.48 and 30.72 for AC and DC, respectively. However, the multivariate TiDE framework improved these SMAPE values significantly, earning 5.83 for AC and 28.45 for DC, indicating a significant increase in forecast accuracy, with a 10.03% improvement for AC and a 7.39% improvement for DC.

Figure 43 shows the forecasted result of the TiDE model on the unseen data.




(a)




(b)

*Figure 43. Forecast on unseen data for TiDE (a) AC power graph (b) DC power graph*

## 7.2 FCR-D Upward Price Forecasting

We use univariate and multivariate approaches to extract insights and patterns from the data to forecast the FCR-D Upward prices. Then, we assess the performance of these approaches through detailed comparative analysis, allowing us to understand their effectiveness in predicting the FCR-D Upward prices.

### 7.2.1 Univariate Forecasting for FCR-D Upward Price

Figure (44) visualizes various forecasting models. The graph incorporates the "yeo-johnson" transformation, a statistical technique for enhancing the interpretability and accuracy of the data.

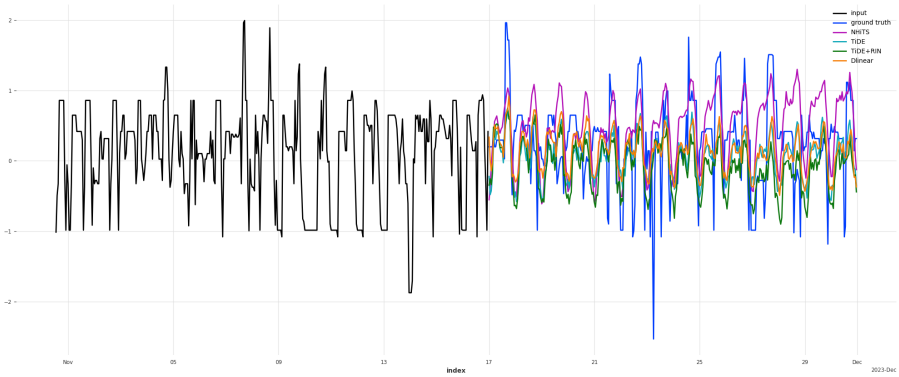



*Figure 44. Scaled model forecast for FCR-D upward price*

Tables (19) and (20) analyze model evaluations, offering insights into their performance across different temporal horizons. These evaluations concern two reference horizons: one spanning 720 hours and the other comprising 72 hours. Examining these evaluations, one can understand how each model performs over varying periods.

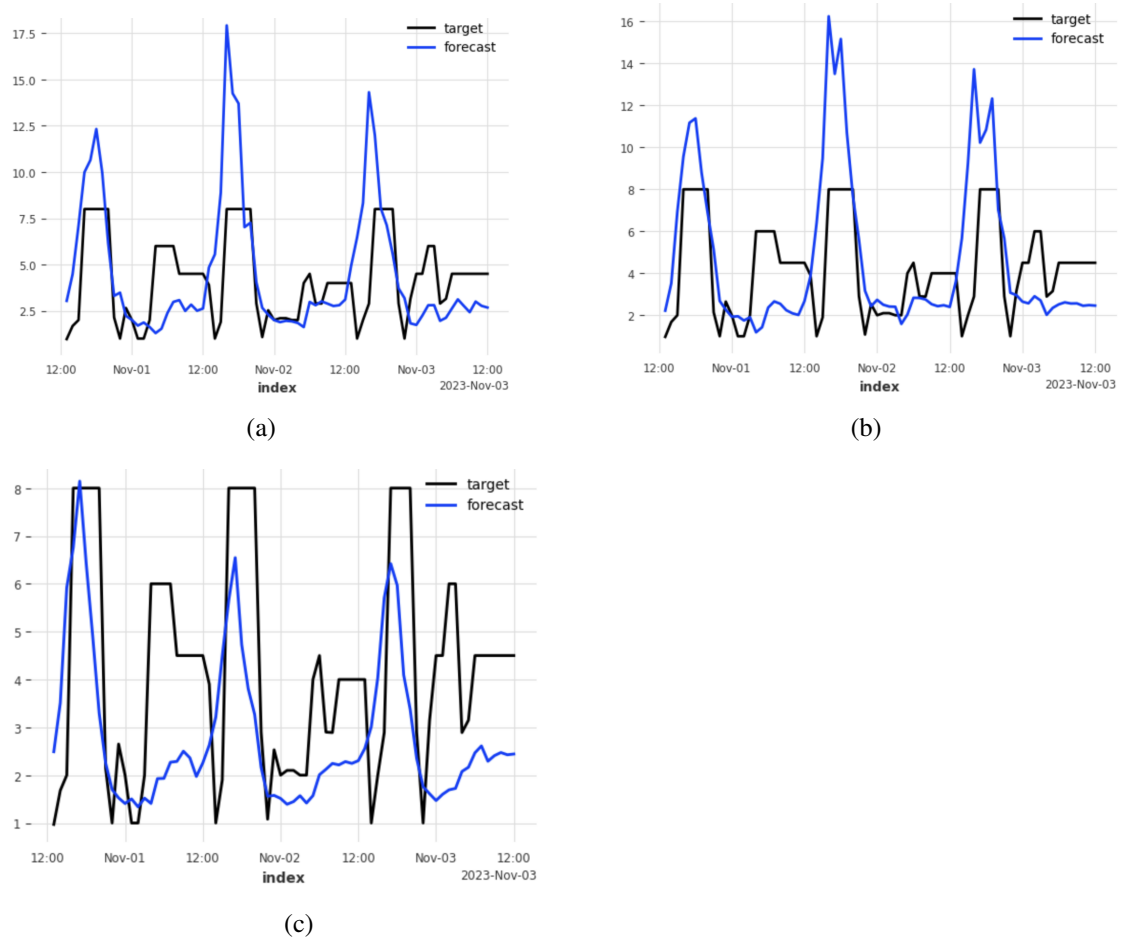
*Table 19. Univariate forecasting for the FCR-D upward with horizon=720 hour*

| | MAE | MSE | RMSE | SMAPE |
|---|---|---|---|---|
| **MSTL** | 23.283919 | 1450.159433 | 38.080959 | 119.737634 |
| **NHiTS** | 4.23081363 | 194.468555 | 13.9451983 | 65.0386820 |
| **TiDE** | 3.85474245 | 186.570071 | 13.6590655 | 57.2209815 |
| **Dlinear** | 3.77824850 | 191.435872 | 13.8360352 | 56.5151639 |

*Table 20. Univariate forecasting for the FCR-D upward with horizon=72 hour*

| | MAE | MSE | RMSE | SMAPE |
|---|---|---|---|---|
| **NHiTS** | 1.99788964 | 5.59623690 | 2.36563668 | 57.5107570 |
| **TiDE** | 2.26479584 | 9.88127985 | 3.14345031 | 51.7645672 |
| **Dlinear** | 2.36985965 | 10.0905978 | 3.17657013 | 54.2650236 |

Figure (45) shows the forecasting capabilities of three different models on the FCR-D upward price. The figure shows a horizon of 72 for three models. From the graph, TiDE and Dlinear forecast high values for the peaks, whereas NHiTS forecasts them as close to the peak value. Similarly, TiDE performs better than the other two approaches regarding low price value.




(a) (b)

(c)

*Figure 45. Forecast on unseen data for NHiTS (a) TiDE (b) Dlinear (c) NHiTS*

**NHiTS for Univariate Approach**

Due to its exceptional efficiency, NHiTS is the top choice for hyperparameter tuning in univariate methods. It ensures faster model convergence and optimizes computational resources, making it ideal for large-scale datasets or resource-constrained environments. In addition, using the Nixtla library along with Optuna for hyperparameter tuning enhances the efficiency and effectiveness of the tuning process. This combination simplifies the search for the best hyperparameters while providing a flexible framework for thoroughly exploring the hyperparameter space, resulting in optimized model performance. The ideal parameters for model forecasting based on hyperparameter tuning are similar to those in Table (13). We fine-tuned the hyperparameters of the NHITS model to achieve the best parameter for FCR-D upward to improve upward price forecasting for two horizon periods, 72 and 720 hours. Through this fine-tuning process, we minimized the model's error, and Table (21) lists the best parameters for 2 different horizons.

*Table 21. The best parameter for NHiTS with horizon 72 and 720*

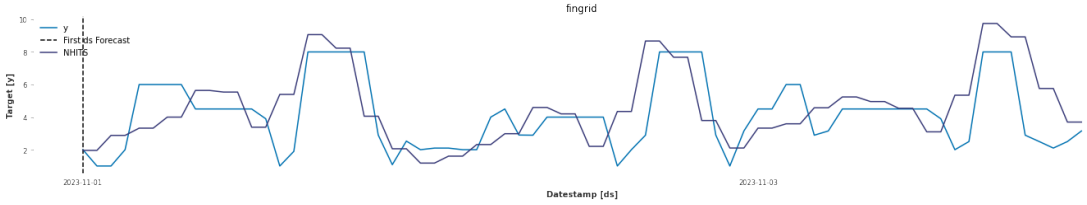
| | **horizon=72** | **horizon=720** |
|---|---|---|
| **input_size** | 38 | 95 |
| **n_pool_kernel_size** | [4, 4, 4] | [8, 4, 1] |
| **mlp_units** | [[512, 512],<br>[512, 512],<br>[512, 512]] | [[256,256],<br>[256,256],<br>[256,256]] |
| **activation** | LeakyReLU, | PReLU |
| **n_blocks** | [4, 4, 4] | [2, 2, 2] |
| **dropout_prob_theta** | 0.5706586722339906, | 0.7140907874265439 |
| **scaler_type** | standard | invariant |
| **pooling_mode** | MaxPool1d | MaxPool1d |
| **interpolation_mode** | nearest, | linear |
| **n_freq_downsample** | [8, 4, 2] | [8, 4, 2] |

The findings indicate that, in the context of FCR-D upward price forecasting, short-term forecasting yields superior results compared to its long-term counterpart, as shown in Table (22). This discrepancy arises from the inability of long-term forecasting to effectively anticipate high peaks, which occur less frequently than low peaks due to the pronounced variability and erratic characteristics inherent in the dataset. Evaluation of forecasting performance utilizes Mean Absolute Error (MAE) as a metric, aiming to minimize MAE across various parameter configurations. Consequently, short-term forecasting exhibits greater efficacy than long-term forecasting, chiefly attributable to the latter's deficiency in

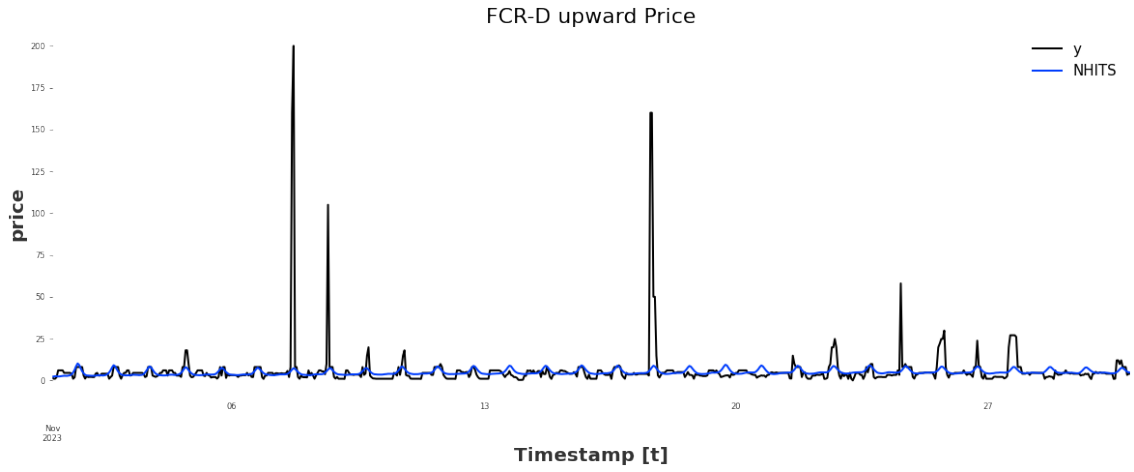
accurately capturing infrequent high peaks.

*Table 22. NHiTS model performance metrics with h=72 and 720*

| h=72 | MAE | MSE | RMSE | SMAPE |
|---|---|---|---|---|
| **NHITS** | 1.512989 | 3.97233 | 1.993071 | 39.478564 |
| **h=720** | | | | |
| **NHITS** | 3.461108 | 241.799 | 15.54991 | 48.988366 |




(a)




(b)

*Figure 46. Univariate forecast on unseen data for NHiTS FCR-D upward price (a) horizon=72 (b) horizon=720*

Figure 46 shows the forecasting capabilities of the NHiTS model on the unseen data.

## 7.3 Multivariate Forecasting for FCR-D Upward Price

We used a multivariate forecasting methodology that resembles the one used in capacity forecasting. Tables (23) and Figure (47) depict the evaluation of the model on new data, along with the forecast plot. The models were trained with default parameters and a 192 input horizon using Quantile transformation. We evaluated the model for two different time horizons, 72 and 720, and compared the results using a provided table. The TiDE model performed better than its counterparts for short-term and long-term forecasting. The model exhibited good performance across various metrics, demonstrating its effec-

tiveness in predicting outcomes accurately over different timeframes.

*Table 23. Multivariate forecasting for the FCR-D upward with horizon=72 and 720*

| h=72 | MAE | MSE | RMSE | SMAPE |
|---|---|---|---|---|
| **NHiTS** | 3.623256949 | 25.8843196 | 5.08766348 | 57.8549751 |
| **TiDE** | 1.89620672 | 5.1346097 | 2.26596771 | 50.47363687 |
| **Dlinear** | 2.73339983 | 25.4660342 | 5.04638824 | 52.2062613 |
| **h=720** | | | | |
| **NHiTS** | 4.6144929 | 236.8211421 | 15.38899419 | 62.053940 |
| **TiDE** | 4.02479571 | 239.22995 | 15.5640694 | 57.336956 |
| **Dlinear** | 11.0328461 | 665.586171 | 25.7989567 | 189.561033 |

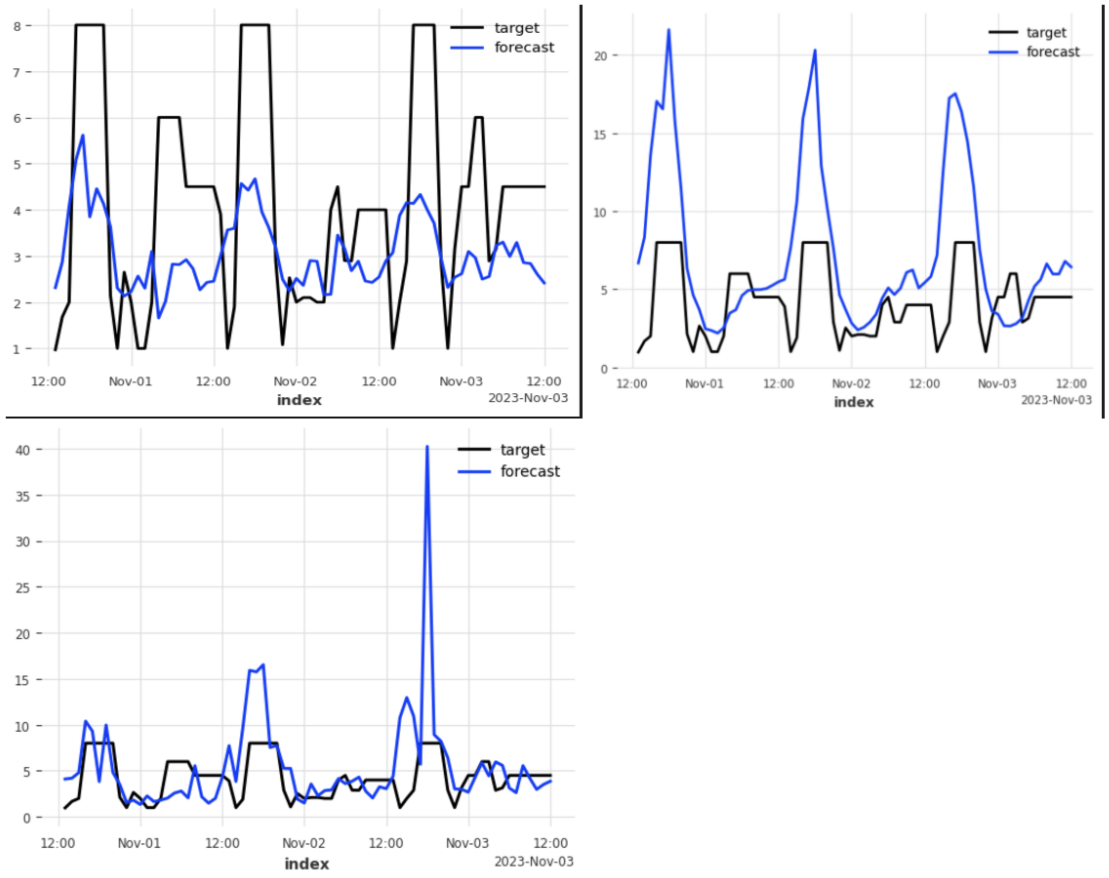



*Figure 47. Multivariate forecast Left: TiDE Right: NHiTS Bottom_left: Dlinear*

### TiDE for Multivariate FCR-D Upward Price Forecasting

After assessing the outcomes from Table (23), we selected TiDE to optimize hyperparameters. We adjusted according to the specifications in Tables (16 and 17) and fine-tuned the hyperparameters for horizons (72 and 720) to identify the optimal values. Table 24 provides the details resulting from the subsequent hyperparameter tuning process. We trained the models to use Mean Absolute Error (MAE) as the designated loss function

and scaled them with "yeo-johnson."

*Table 24. The best hyperparameter value for the TiDE model multivariate approach*

| Parameter | Best_value |
|---|---|
| **input_chunk_length** | 120 |
| **num_encoder_layers** | 24 |
| **num_decoder_layers** | 96 |
| **decoder_output_dim** | 8 |
| **hidden_size** | 64 |
| **temporal_width_past** | 256 |
| **temporal_width_future** | 96 |
| **temporal_decoder_hidden** | 8 |
| **dropout** | 0.41126 |
| **use_layer_norm** | False |
| **user_revesible_instance_norm** | False |

Figure 48 displays the importance of the parameters during the model training and evaluation.

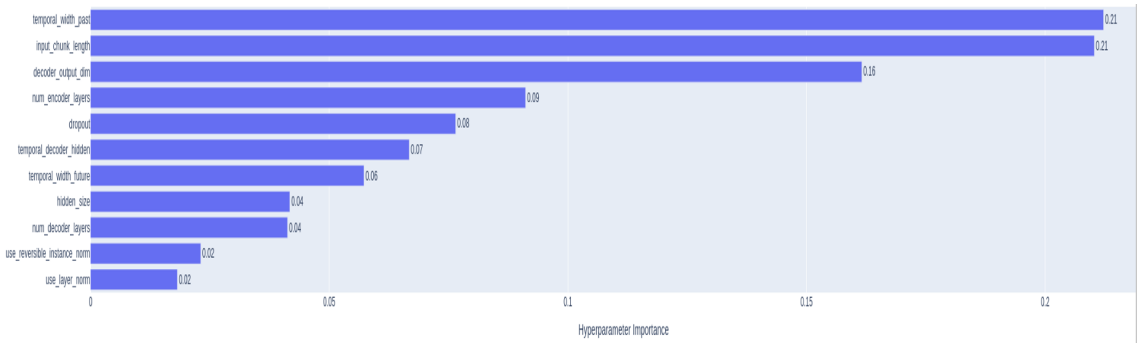



*Figure 48. Hyperparameter Importance for TiDE FCR-D upward*

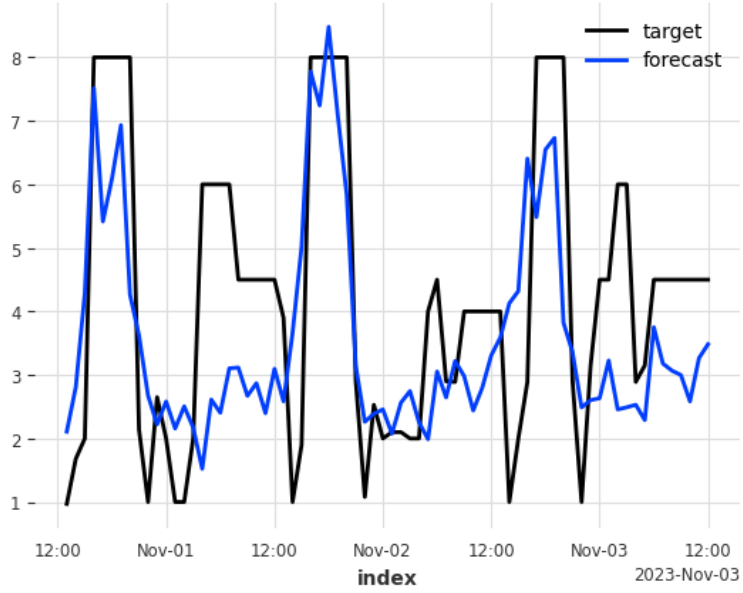
The TiDE approach reduces the error from 50.47 to 42.93 for the 72-hour horizon and 57.33 to 49.36 for the 720-hour horizon given in Table 25. However, The TiDE approach only marginally reduced the error rate despite having more feature parameters.

*Table 25. Model evaluation after finding the best hyperparameter*

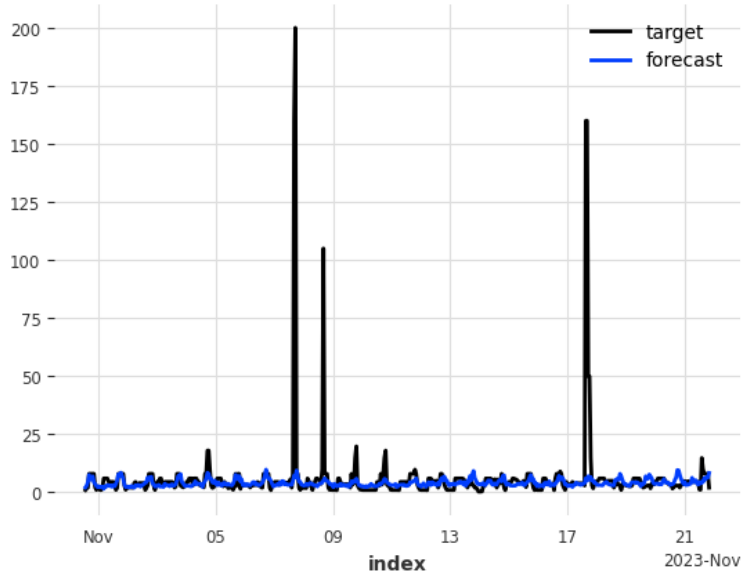
| TiDE | MAE | MSE | RMSE | SMAPE |
|---|---|---|---|---|
| **72** | 1.5216958177280 | 3.479279363 | 1.8652826496 | 42.9382356358210 |
| **720** | 3.474955342 | 244.820066463 | 15.64672702 | 49.366241592706 |

After examining both multivariate and univariate methodologies, the univariate approach is notably practical in forecasting upward price movements within a 72-hour horizon for

the FCR-D upward. Despite including features, their incorporation fails to improve the model's performance. This outcome is potentially due to the dataset's inherently erratic characteristics.




(a)




(b)

*Figure 49. Multivariate forecast on unseen data for TiDE (a) time segment (b) horizon=720*

# 8. Conclusions

In conclusion, the thesis explored various forecasting techniques for two distinct tasks: charge power profile forecasting for demand response and FCR-D upward price forecasting. The research demonstrated that MLP-based architectures showed promising performance in charge demand forecasting, exhibiting satisfactory results even with basic models. However, when facing more complex techniques such as NHiTS and TiDE for univariate and multivariate Forecasting, the trade-off between training time and inference capabilities became apparent, with slower training but faster inference on GPU platforms.

Regarding FCR-D upward price forecasting, the thesis identified significant challenges due to high data variability and erratic properties, leading to lower accuracy in long-horizon forecasting. Many external factors influence the FCR-D upward price, which is challenging to consider when developing the forecasting model. While transformer-based architectures may present a potential solution, their application was limited in this study. Nonetheless, the primary objective of forecasting power profiles for long horizons remains crucial, aiming to determine bidding capabilities and revenue generation thresholds.

Further exploration into transformer-based architectures and other advanced algorithms could enhance the accuracy of FCR-D upward price forecasting. Another approach could be to cap out the high FCR-D upward price to a certain threshold so the model can forecast the low peak price with high accuracy and the high peak to consider as an anomaly case, which seems to happen less frequently. Additionally, investigating the scalability and performance trade-offs of different models, particularly long-horizon forecasting scenarios, is essential for practical implementation in energy markets. Addressing these challenges will ultimately contribute to more informed decision-making processes and improved revenue optimization strategies in the energy sector.

Iterative processes such as model training and hyperparameter tuning can enhance model performance. These can be augmented by expanding the parameter space with diverse values and employing sophisticated feature selection methodologies.

## 8.1 Discussion of Results

The section discusses the Result of the key question for the thesis.

**RQ 1: Can state-of-the-art deep learning approaches accurately forecast EV charging demand and FCR-D upward bid price for a month-long horizon?**

Long-term forecasting is crucial in the energy sector. However, it presents challenges such as dealing with complex data, selecting models, estimating parameters, and managing uncertainty. Despite these challenges, long-term forecasting is vital for optimizing resource allocation, identifying risks, and making informed decisions in Energy market markets. EV charging follows a seasonal pattern with high and low utilization during the day and night. Weekdays see more usage than weekends and holidays, making it easy to forecast long-term. However, FCR-D upward bid prices exhibit randomness, challenging forecasting for a month-long horizon.

In exploring different modeling approaches for forecasting, using classical and state-of-the-art deep learning methods, we found that even simple models like MSTL can perform better in data with a consistent pattern for a long horizon. The MSTL model is faster to train and inference. By employing MLP-based state-of-the-art methods, the forecasting performance improves compared to MSTL. However, these models require more training time and computational power, and inference is slower on the CPU than on the GPU. When comparing different approaches based on the current best model for time series forecasting for our task, NHiTS and TiDE performed better. NHiTS is better for univariate, and TiDE is better for multivariate approach. In conclusion, advanced models are more accurate when trained on vast amounts of data and capable of learning data patterns.

**RQ 2: What are the advantages of univariate and multivariate forecasting approaches, and how do different features contribute to forecasting accuracy?**

With the univariate approach for both cases, MSTL, a statistical approach, is suitable for long-term forecasting with an EV charge profile. With the state-of-the-art approach, the NHiTS approach was slightly better at forecasting the AC power profile than the DC

power profile. Similarly, the MSTL approach performed worse than other state-of-the-art approaches for bid price forecasting. Also, with hyperparameter tuning, the NHiTS performance was significantly better. The challenge with FCR-D upward forecasting is due to data variability and the erratic nature of the data. My model could not accurately forecast the higher bids.

In exploring whether a multivariate approach can enhance our forecasting capabilities, we included multiple features and trained a model. Our findings indicate that the MLP approach significantly reduces the margin of error for forecasting the charge profile. Through hyperparameter tuning using the TiDE model, we observed improved performance compared to the univariate approach. We included features like holidays, hours, and Date Time as future covariates, while variables like station, charge-related information, and temperature as past covariates. Incorporating these variables reduced model error, demonstrating that adding features minimizes error. Our research indicates that multivariate approaches are superior to forecasting the charge profile.

In the case of FCR-D upward bid price forecasting, selecting various features related to bid, spot price, intraday features, and electricity features did not help us improve accuracy significantly. The univariate approach performed better with hyperparameter tuning.

**RQ 3: Does the applied model reliably inform bidding strategies to minimize mistakes, given the penalty of three times the cost of the bid price for under-delivery during the hour?**

Our study aims to assess the reliability of the model used for bidding strategy to minimize errors. If there is under-delivery, a penalty of three times the bid price will be applied for the specific hour. We aim to reduce over-bidding and ensure our model does not overestimate the charge profile. The bid value should closely align with the charge profile value for a given hour. We strive to predict the charge profile as accurately as possible.

When forecasting the charge power profile for the month horizon, our forecasted value matches the unseen data with high confidence. So we can be sure that our forecasted

value can be used to place the next day's bid. The predicted values matched the actual charge profile with high confidence on weekdays and weekends. However, the predicted values were slightly higher on public holidays than the exact power profile. Lower capacity is likely due to decreased usage of the charging stations on holidays. Our model struggled to account for this relationship, as it's difficult to understand human behaviour. In conclusion, the forecasted value to place a bid reduces the error compared to the manual way of guessing tomorrow's charge profile. This method tends to lessen the overbidding and under-delivery.

The primary objective of forecasting an upward price for FCR-D is to estimate the price for the upcoming days and the monthly revenue if we bid with the forecasted capacity. Overall, we have found that forecasting for the long term is more complex than predicting for the short term, such as two days ahead. The forecasting model could predict the low hourly bid, but our model could predict the pattern but not the exact bid price for the hour where the bid is high. For example, in a given hour, the bid price can range from 2-10 euros, and suddenly, the next hour, it can increase by 10 to 20 times. The model can capture the pattern but cannot forecast the exact value.

## 8.2 Impact of Research

### 8.2.1 Sustainability

Improved understanding and utilisation of charge demand forecasting can encourage a sustainable transition. By accurately predicting EV charging demand, operators can optimize resource allocation, enhance charging infrastructure, ensure reliable energy supply, and promote renewable energy sources, thereby supporting broader sustainability goals and advancing the transition to cleaner energy systems.

### 8.2.2 Green transition

The transition to green energy management is crucial for predicting EV charging demand and FCR bid prices. companies can anticipate the rise in electric vehicle adoption enabling efficient allocation of resources and planning of infrastructure. Integrating renewable energy into electric vehicle charging networks improves grid stability, reliability, and eco-friendliness. This is crucial for promoting environmental sustainability, driving

innovation in the energy sector, and moving towards a greener future.

### 8.2.3 Bidding Strategies

Efficient operation and stability of the grid are crucial for charging EVs. Bidding strategies use market dynamics, energy supply and demand, and grid requirements to optimize bid prices. Bidding strategies based on reliable forecasting models can enhance the efficiency and competitiveness of energy markets. Availability of charging stations, grid load forecasts, energy price fluctuations, and regulatory constraints are some factors that affect these strategies. A data-driven bidding strategy helps anticipate energy price fluctuations and adjust bids accordingly. Anticipating future demand and pricing dynamics, market participants can optimize their bidding strategies to minimize costs, maximize revenue, and ensure the reliable provision of energy services.

# References


Akiba, Takuya; Sano, Shotaro; Yanase, Toshihiko; Ohta, Takeru & Koyama, Masanori. 2019, Optuna: A next-generation hyperparameter optimization framework, In: *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2623–2631.

Amini, M Hadi; Kargarian, Amin & Karabasoglu, Orkun. 2016, ARIMA-based decoupled time series forecasting of electric vehicle charging demand for stochastic power system operation, *Electric Power Systems Research*, vol. 140, , pp. 378–390.

Armstrong, J Scott & Forecasting, Long-Range. 1985, From crystal ball to computer, *New York ua*, vol. 348, .

Bell, William R & Hillmer, Steven C. 1984, Issues involved with the seasonal adjustment of economic time series, *Journal of Business & Economic Statistics*, vol. 2, no. 4, pp. 291–320.

Chai, Tianfeng & Draxler, Roland R. 2014, Root mean square error (RMSE) or mean absolute error (MAE)?–Arguments against avoiding RMSE in the literature, *Geoscientific model development*, vol. 7, no. 3, pp. 1247–1250.

Challu, Cristian; Olivares, Kin G.; Oreshkin, Boris N.; Garza, Federico; Mergenthaler, Max & Dubrawski, Artur. 2022, N-HiTS: Neural Hierarchical Interpolation for Time Series Forecasting, *CoRR*, vol. abs/2201.12886, . Available: https://arxiv.org/abs/2201.12886.

Chen, Chao; Twycross, Jamie & Garibaldi, Jonathan M. 2017, A new accuracy measure based on bounded relative error for time series forecasting, *PloS one*, vol. 12, no. 3, p. e0174202.

Cleveland, Robert B; Cleveland, William S; McRae, Jean E & Terpenning, Irma. 1990, STL: A seasonal-trend decomposition, *J. Off. Stat*, vol. 6, no. 1, pp. 3–73.

Cleveland, William S & Devlin, Susan J. 1988, Locally weighted regression: an approach to regression analysis by local fitting, *Journal of the American Statistical Association*, vol. 83, no. 403, pp. 596–610.

Das, Abhimanyu; Kong, Weihao; Leach, Andrew; Sen, Rajat & Yu, Rose. 2023, Long-term Forecasting with TiDE: Time-series Dense Encoder, *arXiv preprint arXiv:2304.08424*.

De Livera, Alysha M; Hyndman, Rob J & Snyder, Ralph D. 2011, Forecasting time series with complex seasonal patterns using exponential smoothing, *Journal of the American Statistical Association*, vol. 106, no. 496, pp. 1513–1527.

Dimoulkas, Ilias; Amelin, Mikael & Hesamzadeh, Mohammad Reza. 2016, Forecasting balancing market prices using Hidden Markov Models, In: *2016 13th International Conference on the European Energy Market (EEM)*, IEEE, pp. 1–5.

Dokumentov, Alexander & Hyndman, Rob J. 2022, STR: Seasonal-trend decomposition using regression, *INFORMS Journal on Data Science*, vol. 1, no. 1, pp. 50–62.

Einolander, Johannes & Lahdelma, Risto. 2022, Explicit demand response potential in electric vehicle charging networks: Event-based simulation based on the multivariate copula procedure, *Energy*, vol. 256, , p. 124656.

Entsoe. 2023, *Bidding FCRImplementation Guide*, [https://www.fingrid.fi/globalasset]. Available: https://www.fingrid.fi/globalassets/dokumentit/fi/sahkomarkkinat/reservit/implementation-guide-fcr.pdf.

Fingrid. 2023, *The electric vehicle charging network as a disturbance reserve*, [https://www.fingrid.fi/]. Published Nov 11 2023. Available: https://www.fingridlehti.fi/en/the-electric-vehicle-charging-network-as-a-disturbance-reserve/.

Fingrid. 2023a, *Electricity system of Finland*, [https://www.fingrid.fi/]. Accessed Jan 20 2024. Available: https://www.fingrid.fi/en/grid/development/electricity-system-of-finland/.

Fingrid. 2023b, *Frequency containment reserves (FCR products)*, [https://www.fingrid.fi/]. Accessed Jan 20 2024. Available: https://www.fingrid.fi/en/electricity-market/reserves_and_balancing/frequency-containment-reserves/.

Fingrid. 2023c, *Open data on the electricity market and the power system*, [https://data.fingrid.fi]. Accessed Jan 20 2024. Available: https://data.fingrid.fi/en/datasets.

Fingrid. 2023d, *Technical Requirements for Frequency Containment Reserve Provision in the Nordic Synchronous Area*, [https://www.fingrid.fi/]. Cited Jan 20 2024. Available: https://www.fingrid.fi/globalassets/dokumentit/fi/sahkomarkkinat/reservit/liite-2-taajuuden-vakautusreservien-fcr-teknisten-vaatimusten-todentaminen-ja-hyvaksyttamisprosessi.pdf.

Fingrid. 2024, *Frequency containment reserves (FCR-N, FCR-D up and FCR-D down), transactions in the hourly and yearly markets*, [https://www.fingrid.fi/]. Accessed Jan 20 2024. Available: https://www.fingrid.fi/en/electricity-market-information/reserve-market-information/frequency-controlled-disturbance-reserve/.

Fingrid. 2024, *Frequency containment reserves FCR products*, [https://www.fingrid.fi/]. Accessed Jan 20 2024. Available: https://www.fingrid.fi/en/electricity-market/reserves_and_balancing/frequency-containment-reserves/technical-requirements.

Fingrid. 2024, *Procurement of frequency containment reserves from annual market 2024*, [https://www.fingrid.fi/]. Accessed Jan 20 2024. Available: https://www.fingrid.fi/en/news/news/2023/procurement-of-frequency-containment-reserves-from-annual-market-2024/.

Garza, Federico; Canseco, Max Mergenthaler; Challú, Cristian & Olivares, Kin G. 2022, StatsForecast: Lightning fast forecasting with statistical and econometric models, *PyCon: Salt Lake City, UT, USA*.

Giovanelli, Christian; Liu, Xin; Sierla, Seppo; Vyatkin, Valeriy & Ichise, Ryutaro. 2017, Towards an aggregator that exploits big data to bid on frequency containment reserve market, In: *IECON 2017-43rd Annual Conference of the IEEE Industrial Electronics Society*, IEEE, pp. 7514–7519.

Giovanelli, Christian; Sierla, Seppo; Ichise, Ryutaro & Vyatkin, Valeriy. 2018, Exploiting artificial neural networks for the prediction of ancillary energy market prices, *Energies*, vol. 11, no. 7, p. 1906.

Greening The Grid. 2023, *Ancillary Services*, [https://greeningthegrid.org/]. Accessed Jan 20 2024. Available: https://greeningthegrid.org/integration-in-depth/ancillary-services.

Herzen, Julien; Lässig, Francesco; Piazzetta, Samuele Giuliano; Neuer, Thomas; Tafti, Léo; Raille, Guillaume; Van Pottelbergh, Tomas; Pasieka, Marek; Skrodzki, Andrzej; Huguenin, Nicolas et al.. 2022, Darts: User-friendly modern machine learning for time series, *The Journal of Machine Learning Research*, vol. 23, no. 1, pp. 5442–5447.

Hyndman, Rob; Koehler, Anne B; Ord, J Keith & Snyder, Ralph D. 2008, *Forecasting with exponential smoothing: the state space approach*, Springer Science & Business Media.

Hyndman, Rob J & Koehler, Anne B. 2006, Another look at measures of forecast accuracy, *International journal of forecasting*, vol. 22, no. 4, pp. 679–688.

Hyndman, Rob J; Koehler, Anne B; Snyder, Ralph D & Grose, Simone. 2002, A state space framework for automatic forecasting using exponential smoothing methods, *International Journal of Forecasting*, vol. 18, no. 3, pp. 439–454.

Kempitiya, Thimal; Sierla, Seppo; De Silva, Daswin; Yli-Ojanperä, Matti; Alahakoon, Damminda & Vyatkin, Valeriy. 2020, An Artificial Intelligence framework for bidding optimization with uncertainty in multiple frequency reserve markets, *Applied Energy*, vol. 280, , p. 115918.

Kim, Sungil & Kim, Heeyoung. 2016, A new metric of absolute percentage error for intermittent demand forecasts, *International Journal of Forecasting*, vol. 32, no. 3, pp. 669–679.

Kirsch, Laurence D & Singh, Harry. 1995, Pricing ancillary electric power services, *The Electricity Journal*, vol. 8, no. 8, pp. 28–36.

Koohfar, Sahar; Woldemariam, Wubeshet & Kumar, Amit. 2023, Prediction of Electric Vehicles Charging Demand: A Transformer-Based Deep Learning Approach, *Sustainability*, vol. 15, no. 3, p. 2105.

Kraft, Emil; Keles, Dogan & Fichtner, Wolf. 2020, Modeling of frequency containment reserve prices with econometrics and artificial intelligence, *Journal of Forecasting*, vol. 39, no. 8, pp. 1179–1197.

Lim, Bryan; Arık, Sercan Ö; Loeff, Nicolas & Pfister, Tomas. 2021, Temporal fusion transformers for interpretable multi-horizon time series forecasting, *International Journal of Forecasting*, vol. 37, no. 4, pp. 1748–1764.

LowCarbonPower. 2024, *Electricity in Finland in 2023*, [https://lowcarbonpower.org]. Available: https://lowcarbonpower.org/region/Finland.

Makridakis, Spyros. 1993, Accuracy measures: theoretical and practical concerns, *International journal of forecasting*, vol. 9, no. 4, pp. 527–529.

Ministry of foreign affair. 2023, *Electricity market*, [https://tem.fi/]. Available: https://tem.fi/en/electricity-market.

Mohammad, Faisal; Kang, Dong-Ki; Ahmed, Mohamed A & Kim, Young-Chon. 2023, Energy Demand Load Forecasting for Electric Vehicle Charging Stations Network based on ConvLSTM and BiConvLSTM Architectures, *IEEE Access*.

Nie, Yuqi; Nguyen, Nam H; Sinthong, Phanwadee & Kalagnanam, Jayant. 2022, A time series is worth 64 words: Long-term forecasting with transformers, *arXiv preprint arXiv:2211.14730*.

NordPool. 2023, *Intraday market*, [https://www.nordpoolgroup.com]. Accessed Jan 20 2024. Available: https://www.nordpoolgroup.com/en/the-power-market/Intraday-market/.

Nordpool. 2023, *Price calculation*, [https://www.nordpoolgroup.com]. Available: https://www.nordpoolgroup.com/en/trading/Day-ahead-trading/Price-calculation/.

Olsson, Magnus & Soder, Lennart. 2008, Modeling real-time balancing power market prices using combined SARIMA and Markov processes, *IEEE Transactions on Power Systems*, vol. 23, no. 2, pp. 443–450.

openforecast. 2023, *ETS taxonomy*, [https://openforecast.org/]. Cited Jan 20 2024. Available: https://openforecast.org/adam/ETSTaxonomy.html.

Oreshkin, Boris N; Carpov, Dmitri; Chapados, Nicolas & Bengio, Yoshua. 2019, N-BEATS: Neural basis expansion analysis for interpretable time series forecasting, *arXiv preprint arXiv:1905.10437*.

Pegels, C Carl. 1969, Exponential forecasting: Some new variations, *Management Science*, pp. 311–315.

Shahriar, Sakib; Al-Ali, Abdul-Rahman; Osman, Ahmed H; Dhou, Salam & Nijim, Mais. 2021, Prediction of EV charging behavior using machine learning, *Ieee Access*, vol. 9, , pp. 111576–111586.

Statista. 2023, *Transportation emissions worldwide: statistics and facts*, [www.statista.com]. Accessed December 18, 2023. Available: https://www.statista.com/topics/7476/transportation-emissions-worldwide/topicOverview.

Statistics Finland. 2023, *Value of energy imports fell by 37 per cent in January to June 2023*, [https://www.stat.fi/]. Cited March 10 2023. Available: https://www.stat.fi/en/publication/cl8mv6u6r0utg0cvzr25mp2uk.

Taylor, Sean J & Letham, Benjamin. 2018, Forecasting at scale, *The American Statistician*, vol. 72, no. 1, pp. 37–45.

Tiedotuskeskus, Autoalan. 2024, *Electric passenger cars in traffic use*, [https://www.aut.fi/]. Cited Jan 02, 2024. Available: https://www.aut.fi/en/statistics/vehicle_fleet/number_of_electric_vehicles.

Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N.; Kaiser, Lukasz & Polosukhin, Illia. 2023, *Attention Is All You Need*.

Virta. 2023a, *Charge the world with us*, [https://www.virta.global/]. Cited 2023. Available: https://www.virta.global/company.

Virta. 2023b, *Electric vehicles were used as a part of the response as Europe's largest nuclear power plant in Finland suffered an unexpected outage on Sunday, 19 November.*, [https://www.virta.global/news/]. Cited Nov 30 2023. Available: https://www.virta.global/news/electric-vehicles-used-as-part-of-the-response-in-a-nuclear-power-plant-outage-in-finland.

Virta. 2023c, *End to end charging solution*, [https://www.virta.global]. Cited 2023. Available: https://www.virta.global/charging-solution.

Watanabe, Shuhei. 2023, Tree-structured Parzen estimator: Understanding its algorithm components and their roles for better empirical performance, *arXiv preprint arXiv:2304.11127*.

Willmott, Cort J & Matsuura, Kenji. 2005, Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance, *Climate research*, vol. 30, no. 1, pp. 79–82.

Wu, Haixu; Xu, Jiehui; Wang, Jianmin & Long, Mingsheng. 2021, Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting, *Advances in Neural Information Processing Systems*, vol. 34, , pp. 22419–22430.

Ye, Zuzhao; Wei, Ran & Yu, Nanpeng. 2021, Short-term Forecasting for Utilization Rates of Electric Vehicle Charging Stations, In: *2021 IEEE International Smart Cities Conference (ISC2)*, IEEE, pp. 1–7.

Zeng, Ailing; Chen, Muxi; Zhang, Lei & Xu, Qiang. 2023, Are transformers effective for time series forecasting?, In: *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, pp. 11121–11128.

Zhou, Haoyi; Zhang, Shanghang; Peng, Jieqi; Zhang, Shuai; Li, Jianxin; Xiong, Hui & Zhang, Wancai. 2020, Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting, *CoRR*, vol. abs/2012.07436, . Available: https://arxiv.org/abs/2012.07436.

Zhou, Tian; Ma, Ziqing; Wen, Qingsong; Wang, Xue; Sun, Liang & Jin, Rong. 2022, FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting, In: Kamalika Chaudhuri; Stefanie Jegelka; Le Song; Csaba Szepesvari; Gang Niu & Sivan Sabato, eds., *Proceedings of the 39th International Conference on Machine Learning*, *Proceedings of Machine Learning Research*, vol. 162, PMLR, pp. 27268–27286. Available: https://proceedings.mlr.press/v162/zhou22g.html.

Zhu, Juncheng; Yang, Zhile; Guo, Yuanjun; Zhang, Jiankang & Yang, Huikun. 2019, Short-term load forecasting for electric vehicle charging stations based on deep learning approaches, *Applied sciences*, vol. 9, no. 9, p. 1723.