

Palestine Technical University – Kadoorie

College of Engineering and Technology

Department of Computer Systems Engineering

Course name:

Graduation Project

Project title:

**Forecasting Electrical Current – Tubas Electrical Company Study Case**

By:

Heba Abd-Alrazeq - 202010955

Supervisor:

Dr. Yosef Daraghmeh

Tulkarm, Palestine

21 March 2023

**ACKNOWLEDGEMENT**

First and foremost, I would like to thank to my supervisor, Dr. Yosef Daraghmeh, for his patience, guidance, encouragement, and support. I have benefited greatly from his wealth of knowledge and meticulous editing and detailed feedback have been very important to us.

Thanks to the interviewees, who so generously took time out of their schedules to participate in my project and make this possible.

A special thanks goes to my colleagues and friends for their help and for many helpful discussion and good ideas along the way, and for those who helped me even in little things.

Last but not least, I would like to thank my family, especially my parents, for their encouragement, patience, and assistance in the past years. We are forever indebted to our parents, who have always kept me in their prayers.

**ABSTRACT**

In the scope of this project, two predicting models will be trained and tested with real data from Tubas Electrical Company, then the results of the models will be compared, The data is the 3 phases current values for 2019 year from Tubas Electrical Company, the data will be preprocessed and analyzed, then used to train the models, some different periods of training data will be chosen to train the models. Then I will augment the data into 4 years and the same models will be trained with the augmented data. A percentage accuracy for the models results will be calculated, then the results will be analyzed and compared between the models considering periods of training and data augmentation to decide the best model.

Contents

[List of Figures 5](#_Toc170070302)

[List of Tables 7](#_Toc170070303)

[CHAPTER 1: INTRODUCTION 8](#_Toc170070304)

[1.1 Background 8](#_Toc170070305)

[1.2 Problem statement: 9](#_Toc170070306)

[1.3 Objectives: 9](#_Toc170070307)

[CHAPTER 2: RELATED WORKS 10](#_Toc170070308)

[2.1 Studies and Researches 10](#_Toc170070309)

[CHAPTER 3: METHODOLOGY & TECHNOLOGIES 13](#_Toc170070310)

[3.1 Technologies 13](#_Toc170070311)

[3.1.1 Integrated Development Environments 13](#_Toc170070312)

[3.1.2 Python is the main programing language 14](#_Toc170070313)

[3.1.3 Excel Sheets 14](#_Toc170070314)

[3.2 Methodology 15](#_Toc170070315)

[3.2.1 Data Collection and Preprocessing 15](#_Toc170070316)

[3.2.2 Exploratory Data Analysis (EDA) 18](#_Toc170070317)

[3.3 Prediction 21](#_Toc170070318)

[3.3.1 Splitting training and testing 21](#_Toc170070319)

[3.3.2 ARIMA Modeling 24](#_Toc170070320)

[3.3.3 Prophet Modeling 30](#_Toc170070321)

[CHAPTER 4: RESULTS AND ANALYSIS 33](#_Toc170070322)

[4.1 Results of predictions 33](#_Toc170070323)

[4.2 Comparative Analysis 35](#_Toc170070324)

[4.3 conclusion 43](#_Toc170070325)

[CHAPTER 5: FUTURE WORK AND POTENTIAL DEVELOPMENT 44](#_Toc170070326)

[REFRENCES 45](#_Toc170070327)

# **List of Figures**

Figure

3.1 Original Data

3.2 Columns to process

3.3 Data before interpolating

3.4 Data after interpolation

3.5 Data after manipulation

3.6 Hourly data plot.

3.7 Hour of the day plot.

3.8 Monthly plot.

3.9 Day of week plot.

3.10 9/3(months) train and test.

3.11 10/2(months) train and test.

3.12 11/1 (months) train and test.

3.13 36/12 (months) train and test.

3.14 42/6 (months) train and test.

3.15 45/3 (months) train and test.

3.22 9/3 (months) train/test and forecast using prophet.

3.23 10/2 (months) train/test and forecast using prophet.

3.24 11/1 (months) train/test and forecast using prophet.

3.25 36/12 (months) train/test and forecast using prophet.

3.26 42/6 (months) train/test and forecast using prophet.

3.26 45/3 (months) train/test and forecast using prophet.

4.1 ARIMA vs Prophet with 9/3 train and test.

4.2 ARIMA vs Prophet with 10/2 train and test.

4.3 ARIMA vs Prophet with 11/1 train and test.

4.4 ARIMA vs Prophet with 36/12 train and test.

4.5 ARIMA vs Prophet with 42/6 train and test.

4.6 ARIMA vs Prophet with 45/3 train and test.

4.7 ARIMA results with all periods before augmentation.

4.8 ARIMA results with all periods after augmentation.

4.9 Prophet results with all periods before augmentation.

4.10 Prophet results with all periods after augmentation.

4.11 ARIMA results with 9/3 and 36/12 periods.

4.12 ARIMA results with 10/2 and 42/6 periods.

4.13 ARIMA results with 11/1 and 45/3 periods.

4.14 Prophet results with 9/3 and 36/12 periods.

4.15 Prophet results with 10/2 and 42/6 periods.

4.16 Prophet results with 11/1 and 45/3 periods.

# **List of Tables**

Table

3.1 MAPE of modeling with ARIMA before augmenting

3.2 MAPE of modeling with ARIMA after augmenting

3.3 MAPE of modeling with Prophet before augmenting

3.4 MAPE of modeling with Prophet after augmenting

# **CHAPTER 1: INTRODUCTION**

## **1.1 Background**

The electric power sector is one of the fastest growing energy sectors due to the increase in electricity consumption. In fact, there are challenges associated with the management of current electric power systems and their sustainability in the medium and long term, which were formed at the beginning of the current century. The changing patterns of consumption of electric energy have contributed to the increase in consumption at peak times, which raises a challenge in how power plants respond to this increase. Another challenge is related to the aging of electrical networks and the continuous need to upgrade them. Healthy electrical networks have specifications that can be summarized in four points:

1- Flexibility or the ability of the network to respond to the increasing demand for electrical energy when there are changes and challenges related to the future.

2- The network's ability to electrically link all consumers and producers.

3- Reliability, that is, the electric power must be continuous and uninterrupted, of good quality and the ability to adapt to unexpected events.

4- The operation of this network should be economical.

A problem occurs if the electrical loads exceed the limit permitted by the electricity companies. Consequently, the capacity of the transformers becomes unable to meet the needs of all the loads, so the companies disconnect the electrical current from the less important loads and provide it for the necessary loads such as factories or hospitals.

Forecasting electrical loads is very important as it helps in developing future plans for the expansion of generating stations and the increase in renewable energy sources, and also developing plans to build new transmission lines, in addition to helping in replacing and rehabilitating existing networks. Therefore, the forecasting process is considered very important in developing networks and making them smart grid.

## **1.2 Problem statement:**

Predicting the future values for the 3 phases is the main purpose of this project, and choosing the best model to develop is the most important part, training and testing more than one model and calculating there accuracies will help us deciding which model is the best to develop, in this project two models will be trained with real data from Tubas Electrical Company, and the results will be compared to choose the best model.

## **1.3 Objectives:**

The main objective of this project is to predict the electrical load at low voltage for enabling the electrical providers to manage the electric distribution during peak hours.

This objective can be achieved by:

1- Developing a short-term forecasting computational model that enables providers to foresee future load values and control the network accordingly

2- Utilizing real world data collected from the grid at Tubas Company.

3- Evaluating the model so that the result can be accurate.

4- Identifying the time horizon in which the model can be accurate.

# **CHAPTER 2: RELATED WORKS**

## **2.1 Studies and Researches**

There are several short term Forecasting models proposed in the literature. For example, the models are based on the principle of machine learning such as **autoregressive moving average** (ARMA [27]), so that the basic condition for using this model is the stability of the time period and the existence of a sequence that contains certain characteristics such as trend and seasonality. There is also another model **Auto-regressive integrated moving average** (ARIMA [11]) that relies on the instability of the time period to reflect the change in different data patterns.

There is also a model **seasonal** **Auto-regressive integrated moving average** (SARIMA [29]), which is an adaptive model that deals with unstable data and requires only previous values ​​for a time series, but it does not work well for long-term periods. The **generalized additive models** (GAM [30-32]) model was also proposed in order to model the relationship between variables and apply it to the very short-term Forecasting. As for the long-term prediction, it used **temporal fusion transformers-based** (TFT [28]) algorithms, fuzzy logic algorithms, random forest, and genetic algorithms, so that it combines more than one model [1].

The work was divided into two sets of data to conduct the study on six different models with different objectives and techniques:

The first group is related to the national total power load and the second group is related to the measurement of active energy in all the 100,000 secondary substations in Portugal and refers to consumers and energy storage devices in addition to the lost part naturally or as solar energy or wind energy.

The second group refers to substations, which is the inter-voltage between medium voltage and low voltage, in addition to smart meters that work on separating voltage from high to low level [1].

The set of models used for forecasting in [1] are:

**GLMLF-B: It is a generalized linear criterion for predicting loads**, and it is a classic regression model for forecasting at the system level as a whole, so that it was used on a set of data in the period (2016-2019) and tested on it. Providing Forecasting as an input to the model reduces prediction errors following these values ​​and obtaining accurate values [1].

**(GAMLF-SL):** **The generalized additive model-based load forecasting**, This model includes the functions of the series of covariates, thus obtaining non-linear effects in order to maintain an interpretable structure. This model requires defining the best available set of input variables [1].

**GAMLF-SLE**: **The prediction of load based on this type of model is a regression model** for the system level resulting from a group of learners from the previous model so that the methods of a group of learners are obtained and combined to make a comprehensive decision. The WM algorithm is based on weighted voting to build a composite algorithm from the output. A set of well-known algorithms so that at least one achieves good performance, and the algorithm is able to support individual and non-specific results and is able to make a time delay, and is able to learn in independent and specific periods thus designing and calibrating predictors exclusively for specific systems and able to deal with results for different groups. Henceforth, a specified methodology was systematically adhered to, comprising the following steps: initial formulation and meticulous selection of input variables, precise delineation of the model's structural framework, establishment of a robust calibration and control model, rigorous model evaluation, and subsequent derivation of an augmented set of input variables. Iterations within this methodical approach persist until an optimal equilibrium is attained, harmonizing applicability, interpretability, and performance precision. In conclusion, the load Forecasting predicated upon the generalized model is rigorously evaluated utilizing dataset [1].

The discernible merits of this approach encompass notably heightened accuracy when juxtaposed against the standard model (GLMLF-B), while accounting for identical variables. Evidently, this methodology effectuates an enhancement in average accuracy and a notable reduction in predictive errors, accentuating its efficacy and potential [1].

Thus, this algorithm **GAMLF-SLE** to be an effective way to achieve better accuracy than the general-purpose predictor (STLF) when (GAM [30-32]) was used. Therefore, this algorithm is considered more accurate on specific days [1].

Compared to general-purpose days, the combination of more than two technologies leads to an improvement in overall accuracy [1]. For example, GAMLF-SSL aims to predict energy loads on thousands of kilometers at a more detailed level compared to the previous model, so that the more detailed the Forecasting, the more complex and difficult it is. When applying this model, the researchers needed at least a year to test the prediction model [1].

This [1] proposes methodologies for load forecasting by advocating the integration of smart meters within substations. The imperative behind ensuring comprehensive data availability is twofold: firstly, to facilitate effective maintenance strategies and secondly, to empower network load prediction. Notably, this study transcends mere theoretical exploration, extending its purview towards practical applicability, interpretability, and result reproducibility within a meticulously structured framework. This framework is designed to imbibe artificial intelligence principles, as underscored by relevant scholarly references [1].

To expedite data acquisition within narrow time frames, the integration of household load data was advocated. In this endeavor, the Markov model was leveraged to probabilistically distribute temporal load occurrences. A rigorous comparative analysis ensued, assessing the predictive accuracy and efficacy of this approach in contrast to conventional methodologies. Conventional methods for load Forecasting suffer from inaccuracies attributable to the stochastic nature of household load behaviors.

In response, load Forecasting methodologies were redesigned, aligning with analogous load patterns observed on comparable days, notably considering load frequencies vis-à-vis weather conditions and temperature. To operationalize this approach, the self-recursive autoregressive integrated moving average (ARIMA [11]) algorithm was employed, providing a robust model for load forecasting [2].

# **CHAPTER 3: METHODOLOGY & TECHNOLOGIES**

The main purpose of this project is to evaluate and analyze the use of data mining techniques for power consumption forecasting, to produce models which are comprehensive and reliable.

## **3.1 Technologies**

In this section, I will discuss the ide I worked on, programing languages, and database.

### **3.1.1 Integrated Development Environments**

* **Jupyter Notebook**

Jupyter Notebook is an open-source web application that allows you to create and share documents containing live code, equations, visualizations, and narrative text. It is an interactive computational environment, in which you can combine code execution, rich text, mathematics, plots, and rich media into a single document.

Reasons for Using Jupyter Notebook:

* **Interactive Computing:** Jupyter Notebook's dynamic nature allows for immediate execution and visualization of code, speeding up development and enhancing data understanding.
* **Integrated Data Visualization:** It supports various libraries for creating visualizations, making it easy to generate and visualize insights within the same environment.
* **Reproducibility and Sharing:** Notebooks can be easily shared, facilitating collaboration and ensuring reproducibility of experiments by combining narrative text with code and outputs.

### **3.1.2 Python is the main programing language**

Python is most popular language in ML & AI fields. Because Python comes with a huge amount of libraries. Many of the libraries are for Artificial Intelligence and Machine Learning. Some of the libraries is scikit-learn (for data mining, data analysis and machine learning), pylearn2 (more flexible than scikit-learn), etc. The list keeps going and never ends. For other languages, students and researchers need to get to know the language before getting into ML or AI with that language. This is not the case with python. Even a programmer with very basic knowledge can easily handle python. Apart from that, the time someone spends on writing and debugging code in python is way less when compared to C, C++ or Java. This is exactly the students of AI and ML wants. They don’t want to spend time on debugging the code for syntax errors, they want to spend more time on their algorithms and heuristics related to AI and ML. Not just the libraries but their tutorials, handling of interfaces are easily available online. People build their own libraries and upload them on GitHub or elsewhere to be used by others .Python has a solid claim to being the fastest growing major programming language. Recommended to check ground breaking statistics on incredible growth of python and why is python growing so quickly from stack overflow [18].

### **3.1.3 Excel Sheets**

The data where sent by the company as excel sheets containing values of current and all attributes and we worked on it manually and then created a csv file for the clean data, since csv files is easy to read with python.

Using CSV files to upload data into Jupyter Notebook provides a simple, compatible, and efficient method for data management. The seamless integration with Pandas and other data analysis libraries enhances the overall workflow, enabling effective data exploration and analysis in a versatile and user-friendly environment.

In my project, I imported data from CSV files into Jupyter Notebook using Pandas. The `read\_csv` function in Pandas enables easy loading of data into DataFrames, which are ideal for data manipulation and analysis.

## **3.2 Methodology**

### **3.2.1 Data Collection and Preprocessing**

The data where sent by the company in a csv file that contains many attributes as appearing in figure (3.1).

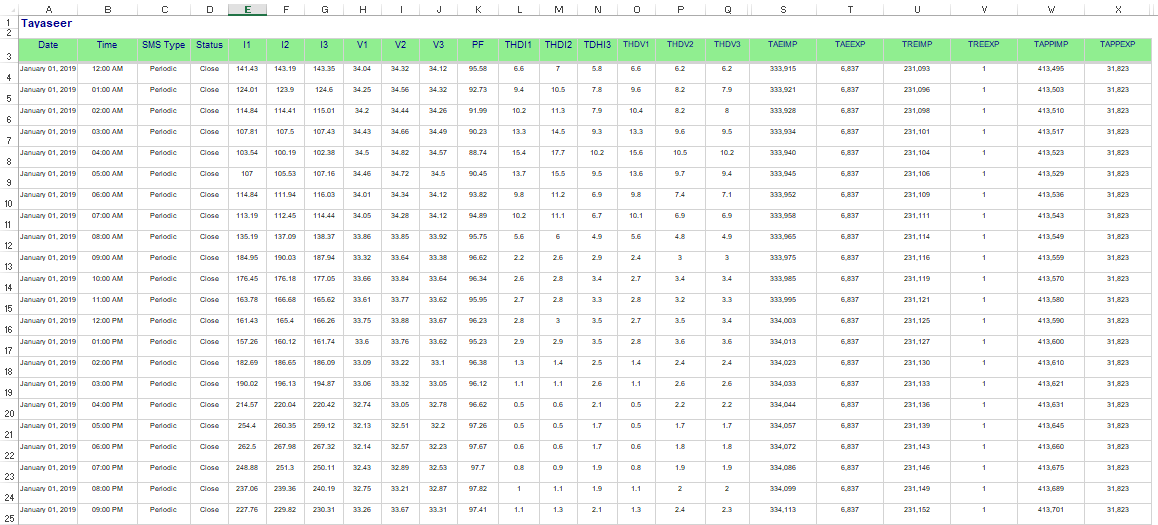


Figure 3.1: Original Data

The needed columns where taken to a new xlsx file to be cleaned since there is many null values and these need to be filled.

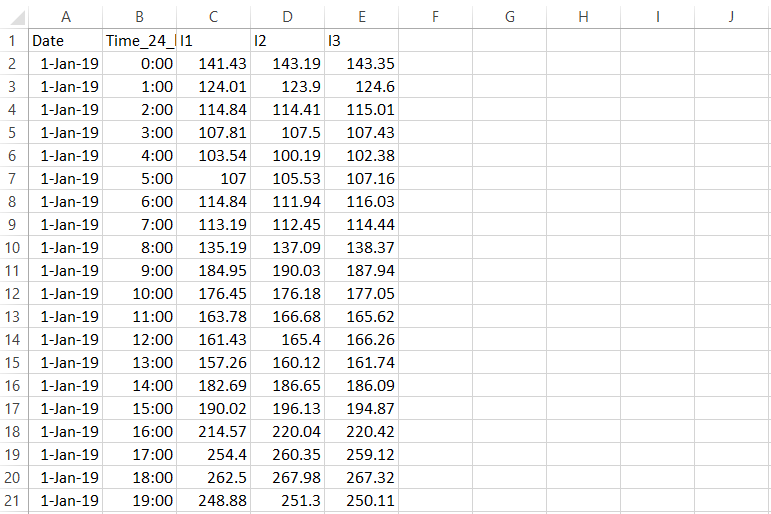


Figure 3.2: columns to process

A plot for the data after interpolating it.

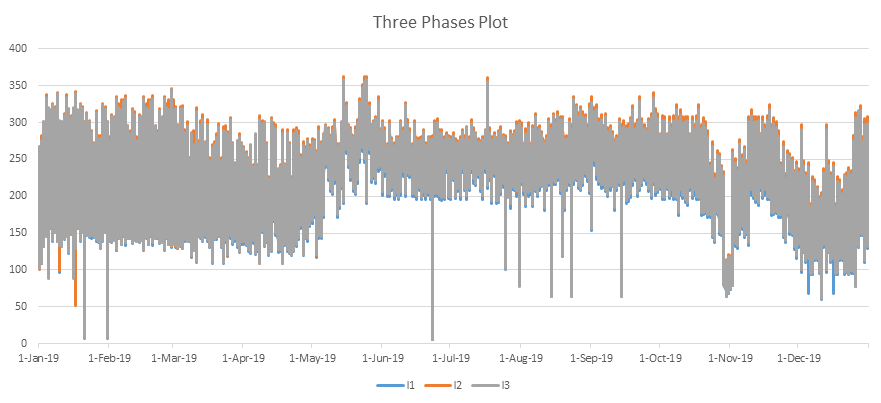
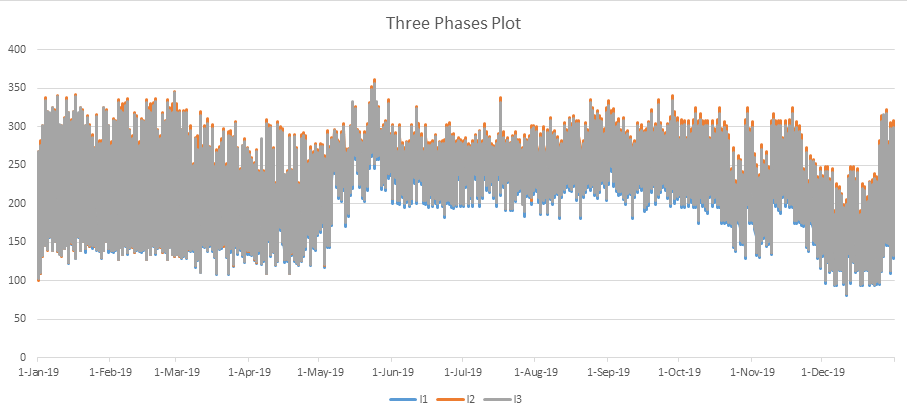


Figure 3.3 Data before interpolating

But as seen in the plot there is many outliers and these need to be corrected, after correcting them manually the plot appears like this.

Figure 3.4: Data after interpolation

The plot shows that there are some outlying in the last three months. But to be sure we reached out the company to ensure if that is normal or not, they have told us that is not normal and some interrupts was happening at that time. These interrupts may be caused of some problems or issues in the company's power transformers. To get a more accurate results in our prediction we must fix these outliers, the outlying is happening in the last days of October and all of November and December, and the best values to get for these months is, get data From April, add noise to it, and replace the outlying in October with it, the same for March with November, and February with December. This is the plot after manipulating.

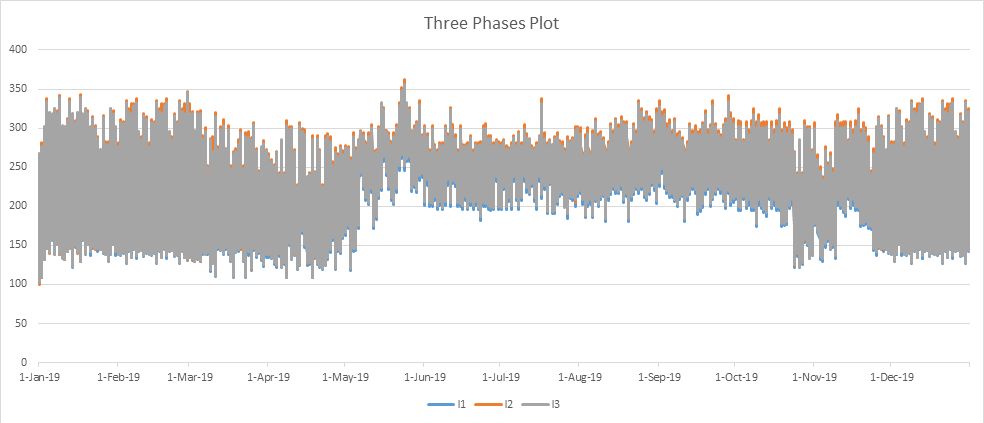


Figure 3.5: data after manipulation

### **3.2.2 Exploratory Data Analysis (EDA)**

Exploratory Data Analysis is a foundational step in data analysis, providing essential insights that guide the entire analytical process. By understanding, cleaning, and visualizing the data, EDA ensures that analyses are based on accurate, well-prepared data, leading to more reliable and meaningful conclusions.

Our dataset contains of almost 8,000 rows and 5 attributes:

• Date: This column contains the Date.

• Time: This column contains the time in 24 hour format.

• I1: This column contains phase 1 current read at that date and time.

• I2: This column contains phase 2 current read at that date and time.

• I3: This column contains phase 3 current read at that date and time.

Graphically data analysis, first let’s look at the data in general.

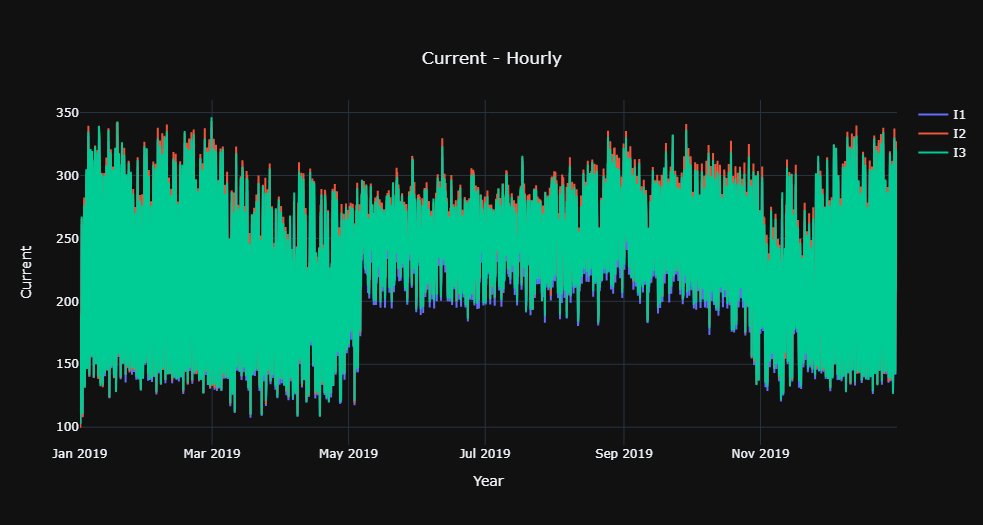


Figure 3.6: Hourly data plot.

The plot shows the Hourly time series data from Tubas Electrical Company, for the year 2019, the data contains 3 phases reads, a read for each hour, the plot shows that the lower values increases in the months May to October, but the upper values do not differ that much between the months.

For a better Data understanding, let’s look at the data by hour of the day

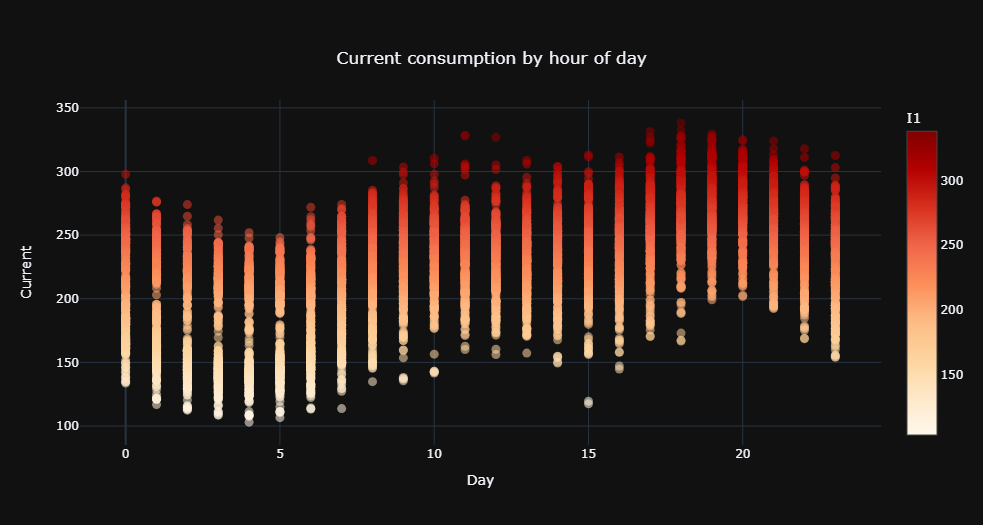


Figure 3.7: Hour of the day plot.

This graph shows the peak times, the times when energy is most consumed, and the times when it is least consumed.

Now let’s plot the data monthly.

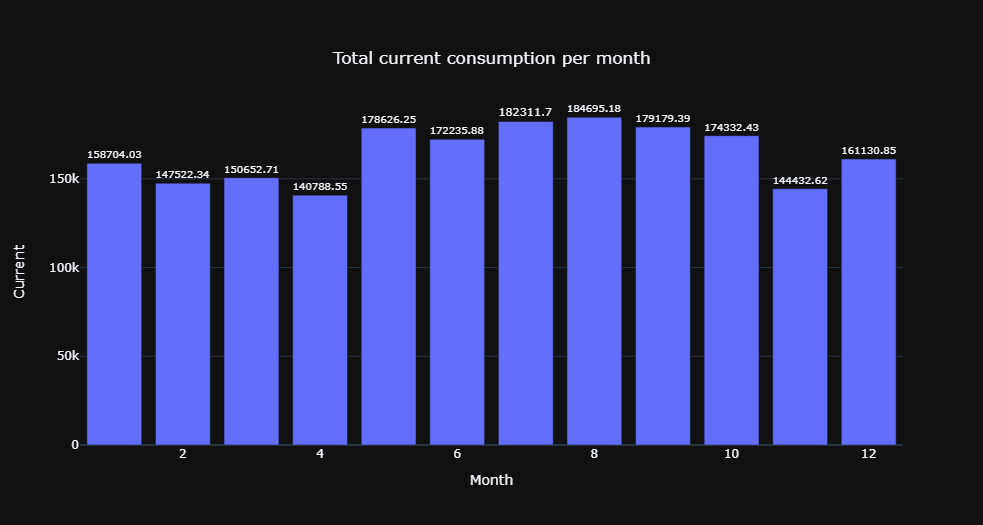


Figure 3.8: Monthly plot.

The plot shows the total current values for each month, the month from May to October have the highest values.

Let’s have a look at the data considering day of week.

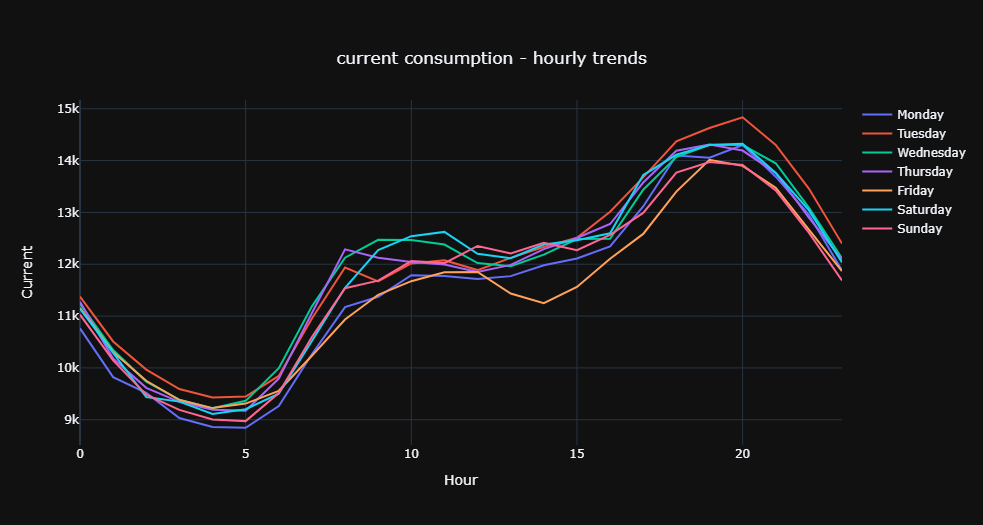


Figure 3.9: Day of week plot

This plot shows the daily trends, Friday have the lowest values from 12 Pm to 7 Pm, but Tuesday have the highest value from 7 Pm to 5 Am.

## **3.3 Prediction**

Two models where chosen to be used for predicting the future value, the data will be split into train and test. In this project we are going to split the data into three different periods for training and testing, then the data will be augmented into 4 years and the prediction models applied again into three different periods, the accuracy will be calculated for each model for all periods, and then compared between the models and the data before and after augmentation.

### **3.3.1 Splitting Training and Testing**

The periods of training and testing splitting before augmenting the data: 9/3 (months) train and test, 10/2 train and test, and 11/1 train and test

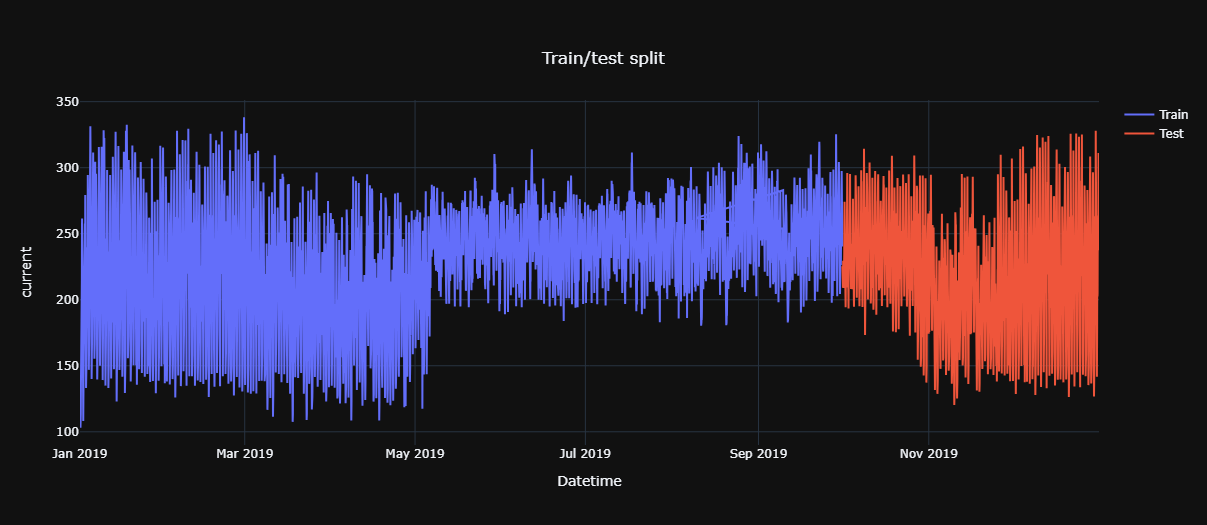


Figure 3.10: 9/3(months) train and test.

This plot shows the train and test splits, the first 9 months will be used to train the model and the other 3 month will be used in testing the model results.

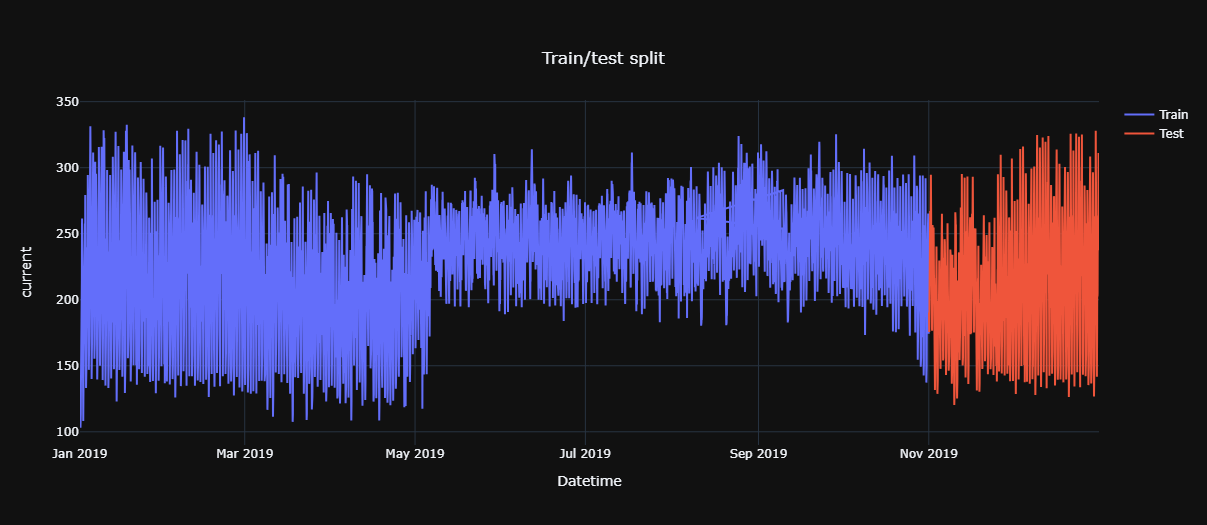


Figure 3.11: 10/2(months) train and test.

This plot shows the train and test splits, the first 10 months will be used to train the model and the other 2 month will be used in testing the model results.

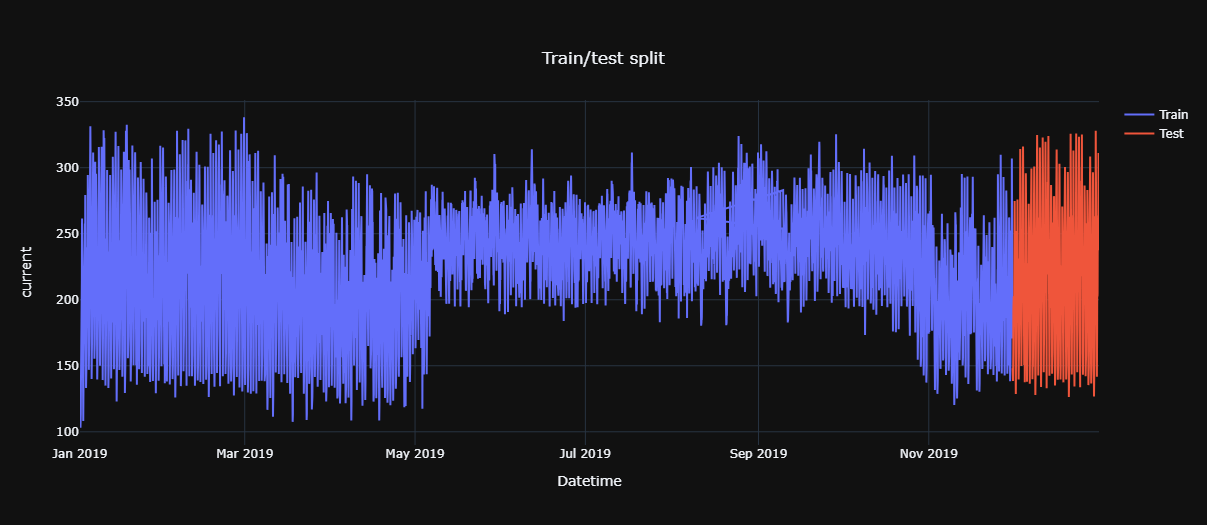


Figure 3.12: 11/1 (months) train and test.

This plot shows the train and test splits, the first 11 months will be used to train the model and the other month will be used in testing the model results.

And a new periods of training and testing where split after augmenting the data into 4 years, the periods of training and testing splitting where 36/12 (months) train and test, 42/6 train and test, and 45/3 train and test



Figure 3.13: 36/12 (months) train and test.

This plot shows the train and test splits, the first 36 months will be used to train the model and the other 12 month will be used in testing the model results.



Figure 3.14: 42/6 (months) train and test.

This plot shows the train and test splits, the first 42 months will be used to train the model and the other 6 month will be used in testing the model results.



Figure 3.15: 45/3 (months) train and test.

This plot shows the train and test splits, the first 45 months will be used to train the model and the other 3 month will be used in testing the model results.

### **3.3.2 ARIMA Modeling**

The ARIMA (AutoRegressive Integrated Moving Average) model is a popular and widely used statistical method for time series forecasting. It is a powerful tool for understanding and predicting future points in a series based on its own past values and past errors. ARIMA models are particularly useful in fields such as economics, finance, and any domain where understanding temporal patterns is crucial.

**Components of ARIMA**

The ARIMA model is composed of three main components:

**1. AutoRegressive (AR) Component:**

* This component involves regressing the variable on its own lagged (past) values. The order of the AR component is denoted by p, which represents the number of lagged observations included in the model.

**2. Integrated (I) Component:**

* The integration part of the model involves differencing the data to make it stationary, which means removing trends and seasonal structures that can affect the forecasting performance. The order of differencing is denoted by d.

**3. Moving Average (MA) Component:**

* This component involves modeling the error term as a linear combination of past error terms. The order of the MA component is denoted by q, which represents the number of lagged forecast errors included in the model.

#### ARIMA Model Notation

An ARIMA model is typically denoted as ARIMA(p, d, q), where:

* p is the number of lag observations included in the model (the lag order).
* d is the number of times that the raw observations are differenced.
* q is the size of the moving average window.

The ARIMA components can be challenging to estimate accurately. Traditionally, these components are determined through a combination of statistical techniques and domain knowledge. However, an alternative approach involves using brute force methods to identify the optimal parameters as we done in our project.

#### Brute Force Estimation Method

Brute force estimation involves systematically testing a wide range of possible values for p, d, and q to find the combination that best fits the data. This approach is computationally intensive but can be effective in identifying the optimal ARIMA model components. Here is a summary of how brute force estimation works:

**1.** **Define the Parameter Grid:**

* Establish a range of possible values for p, d, and q. For instance, p and q might range from 0 to 5, while d might range from 0 to 2.

**2. Fit Models for Each Combination:**

* Iterate through all possible combinations of p, d, and q. For each combination, fit an ARIMA model to the data.

**3. Evaluate Model Performance:**

* For each fitted model, calculate a performance metric such as the Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC), or Mean Squared Error (MSE). These metrics help in comparing the models' goodness of fit and penalize model complexity.

**4.** **Select the Optimal Model:**

* Identify the combination of p, d, and q that results in the best performance metric. This combination is considered the optimal set of parameters for the ARIMA model.

Now a function to estimate the components where defined, and

The three periods of training and testing where trained and tested using ARIMA implementation in python.

The result of prediction using ARIMA for the three periods before augmentation:

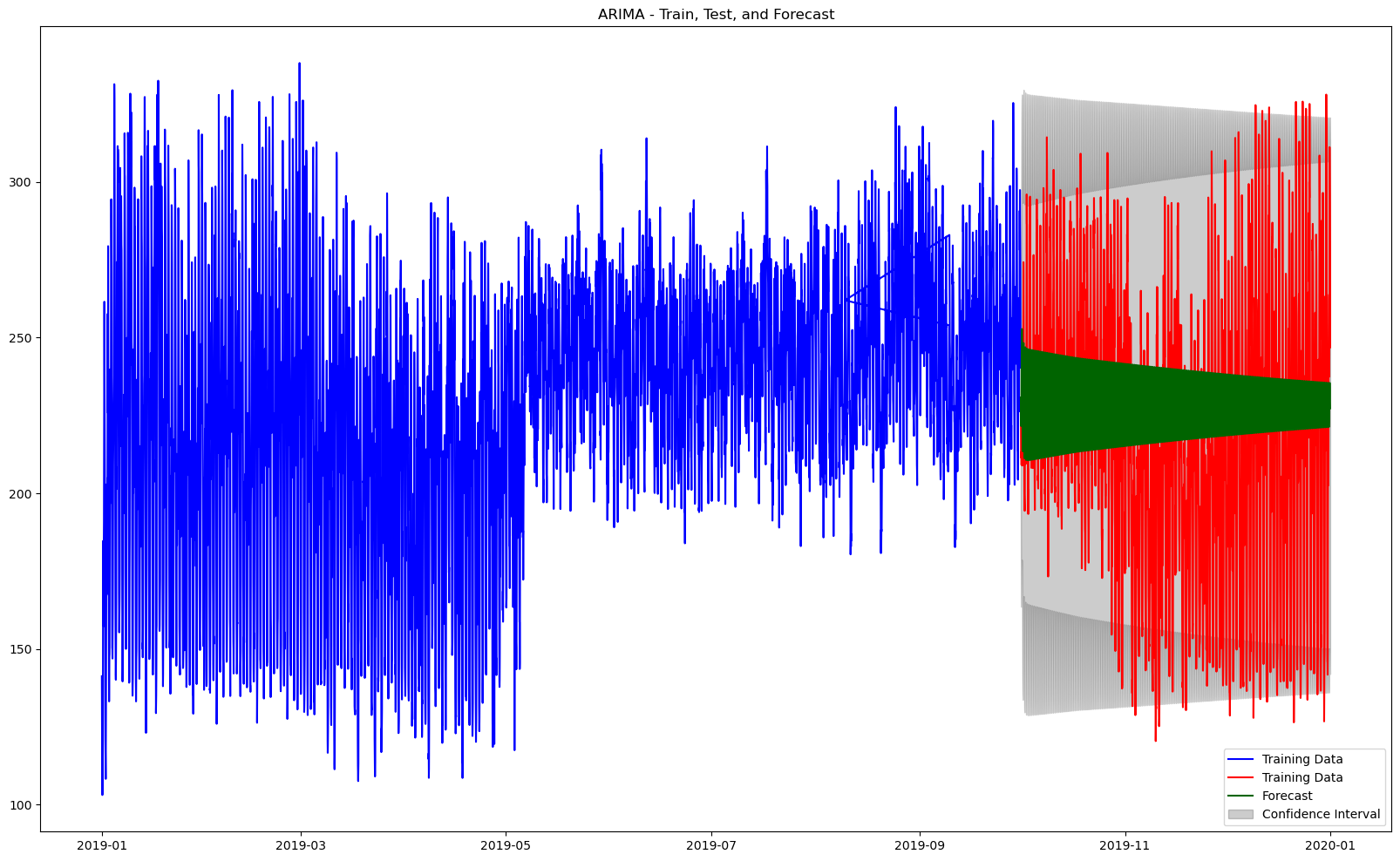


Figure 3.16: 9/3 (months) train/test and forecast using ARIMA.

This plot shows the results of ARIMA predicting model for the period 9/3 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations from October 2019 to December 2019.
* The green part is the mean values, a line starting from October 2019 onward, showing the predicted electrical consumption.
* The confidence interval is the shaded area around the mean values, indicating the uncertainty of the predictions.

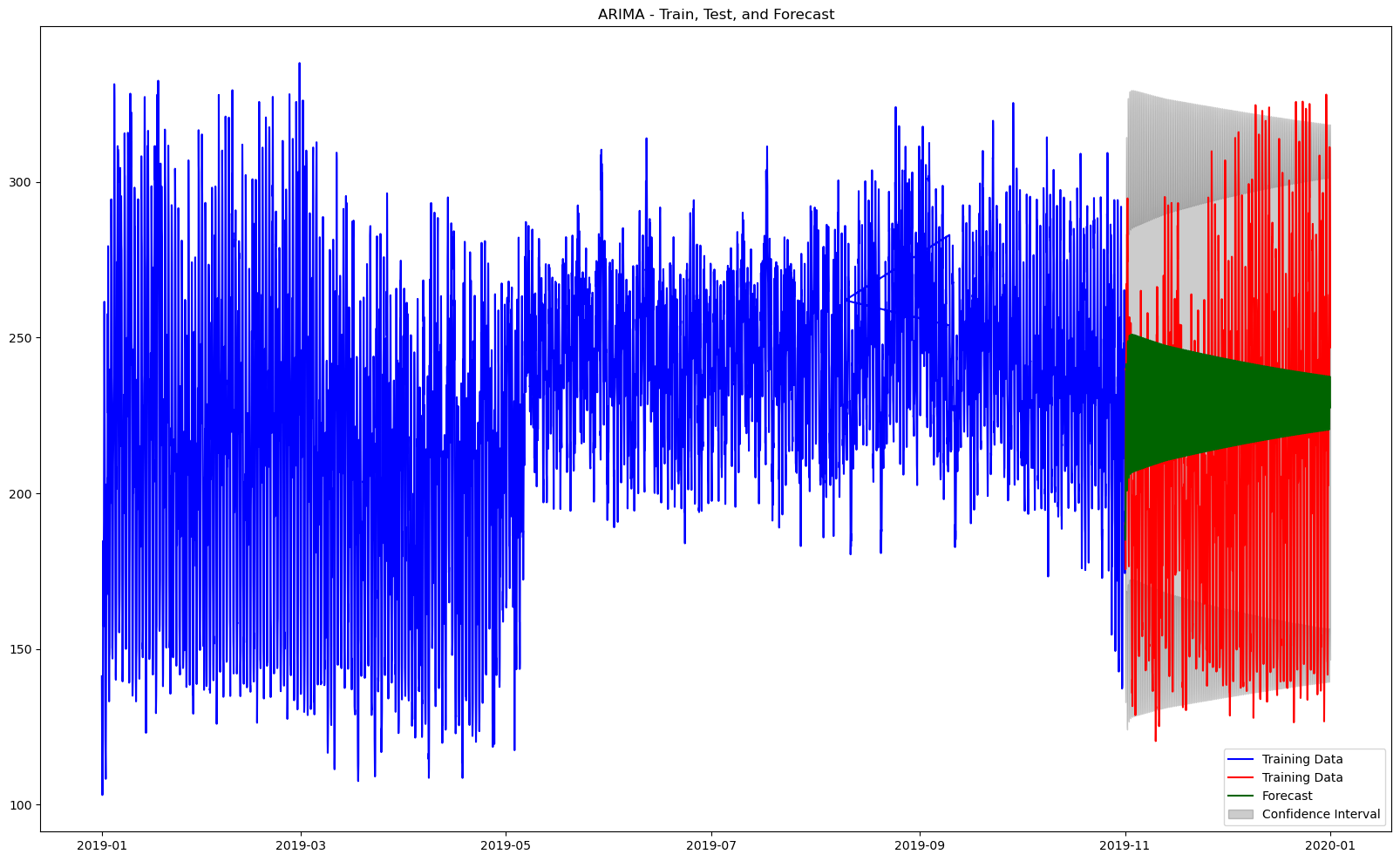


Figure 3.17: 10/2 (months) train/test and forecast using ARIMA.

This plot shows the results of ARIMA predicting model for the period 10/2 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations from November 2019 to December 2019.
* The green part is the mean values, a line starting from November 2019 onward, showing the predicted electrical consumption.
* The confidence interval is the shaded area around the mean values, indicating the uncertainty of the predictions.

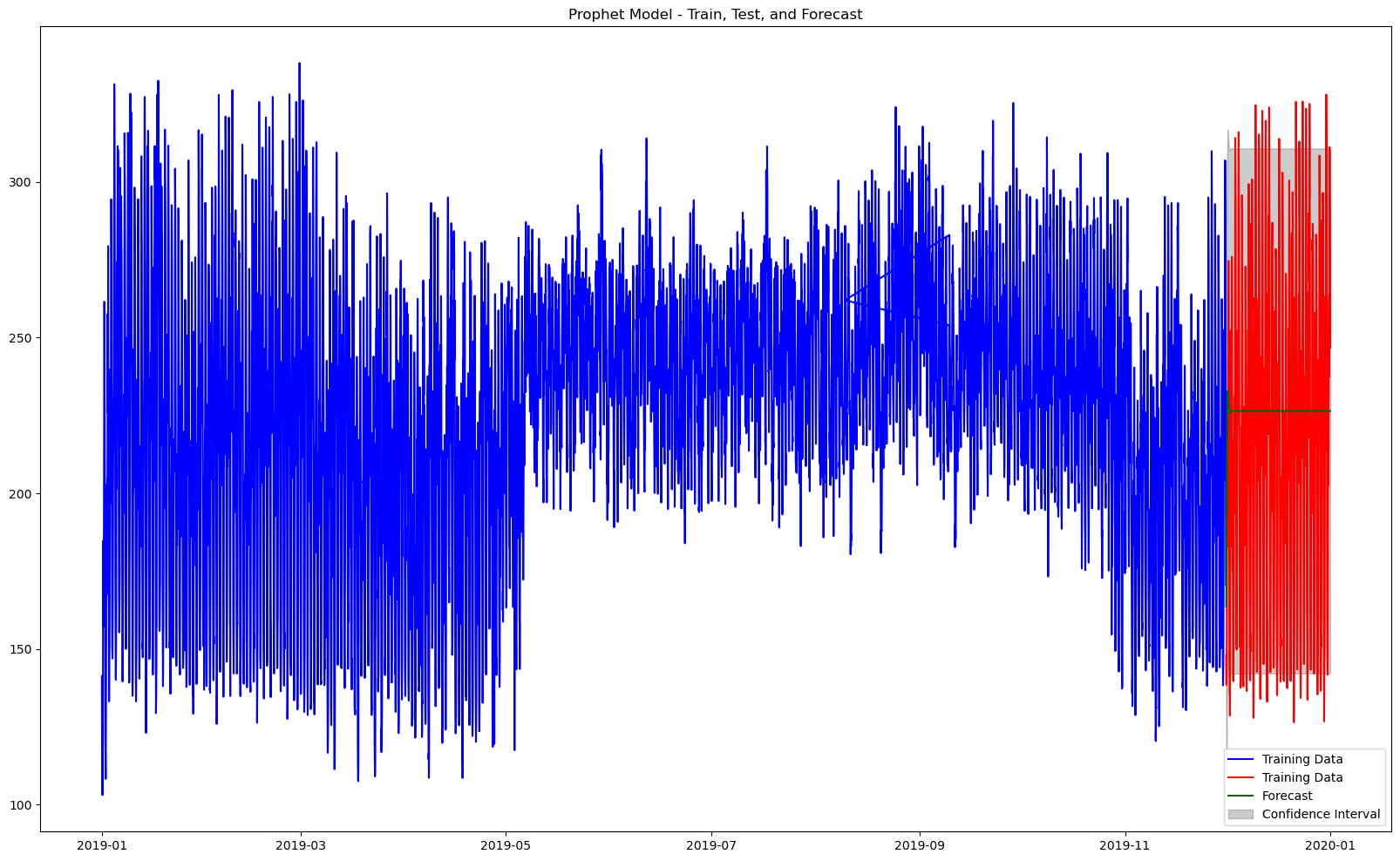


Figure 3.18: 11/1 (months) train/test and forecast using ARIMA.

This plot shows the results of ARIMA predicting model for the period 9/3 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations for December 2019.
* The green part is the mean values, a line starting from December 2019 onward, showing the predicted electrical consumption.
* The confidence interval is the shaded area around the mean values, indicating the uncertainty of the predictions.

The result of prediction using ARIMA for the three after before augmentation:

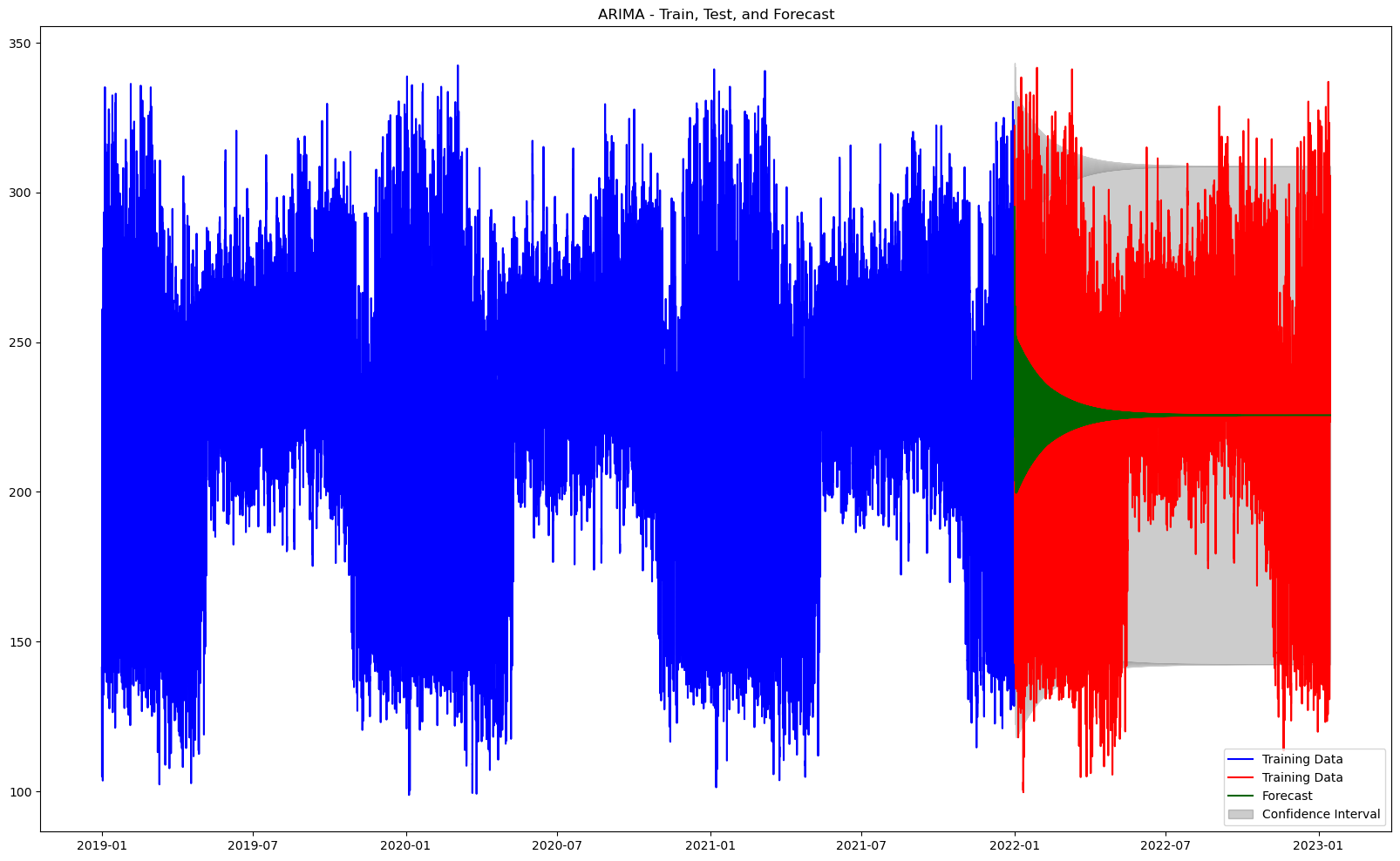


Figure 3.19: 36/12 (months) train/test and forecast using ARIMA.

This plot shows the results of ARIMA predicting model for the data after augmentation for the period 36/12 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations from January 2022 to January 2023.
* The green part is the mean values, a line starting from January 2022 onward, showing the predicted electrical consumption.
* The confidence interval is the shaded area around the mean values, indicating the uncertainty of the predictions.

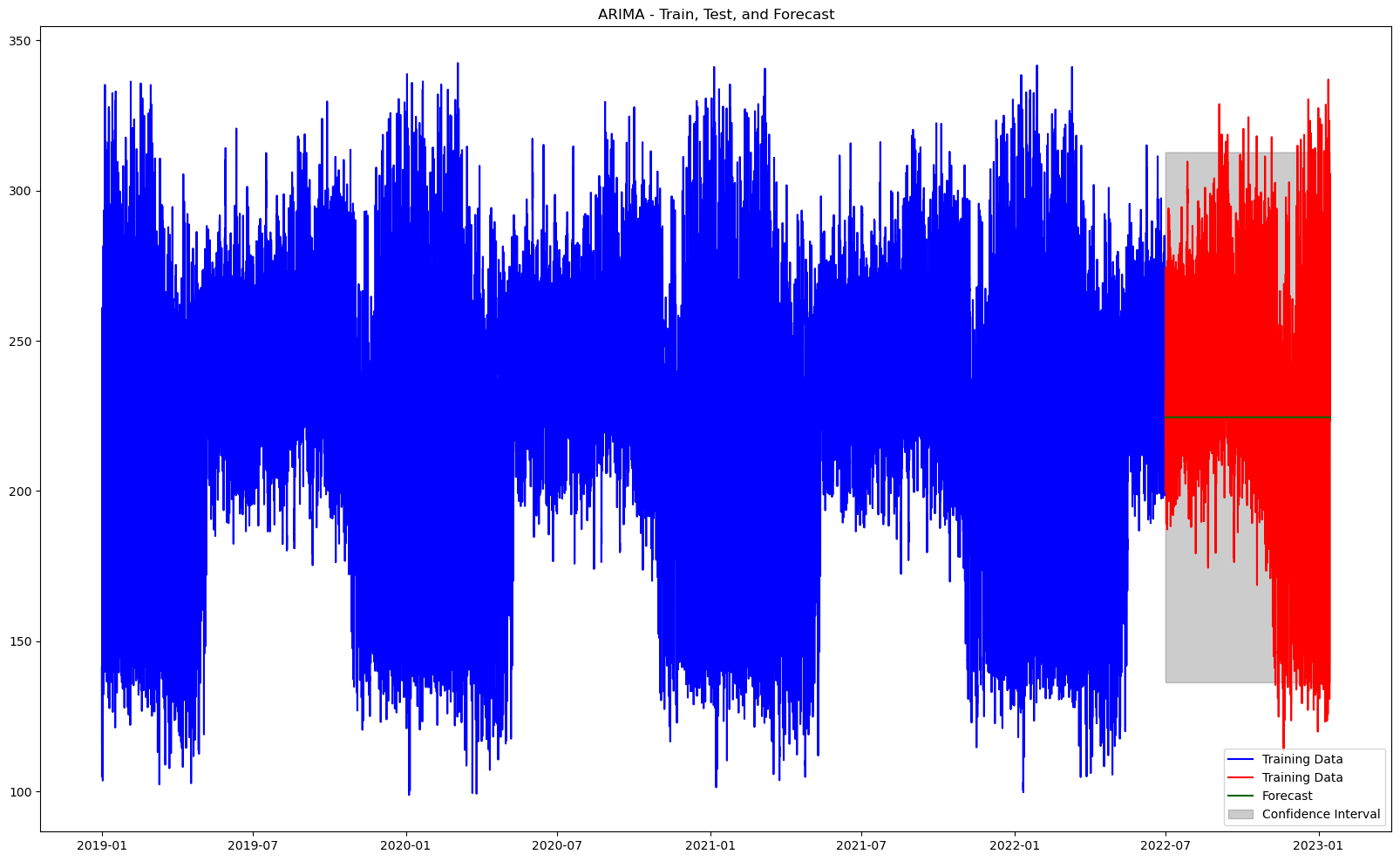


Figure 3.20: 42/6 (months) train/test and forecast using ARIMA.

This plot shows the results of ARIMA predicting model for the data after augmentation for the period 42/6 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations from June 2022 to January 2023.
* The green part is the mean values, a line starting from June 2022 onward, showing the predicted electrical consumption.
* The confidence interval is the shaded area around the mean values, indicating the uncertainty of the predictions.

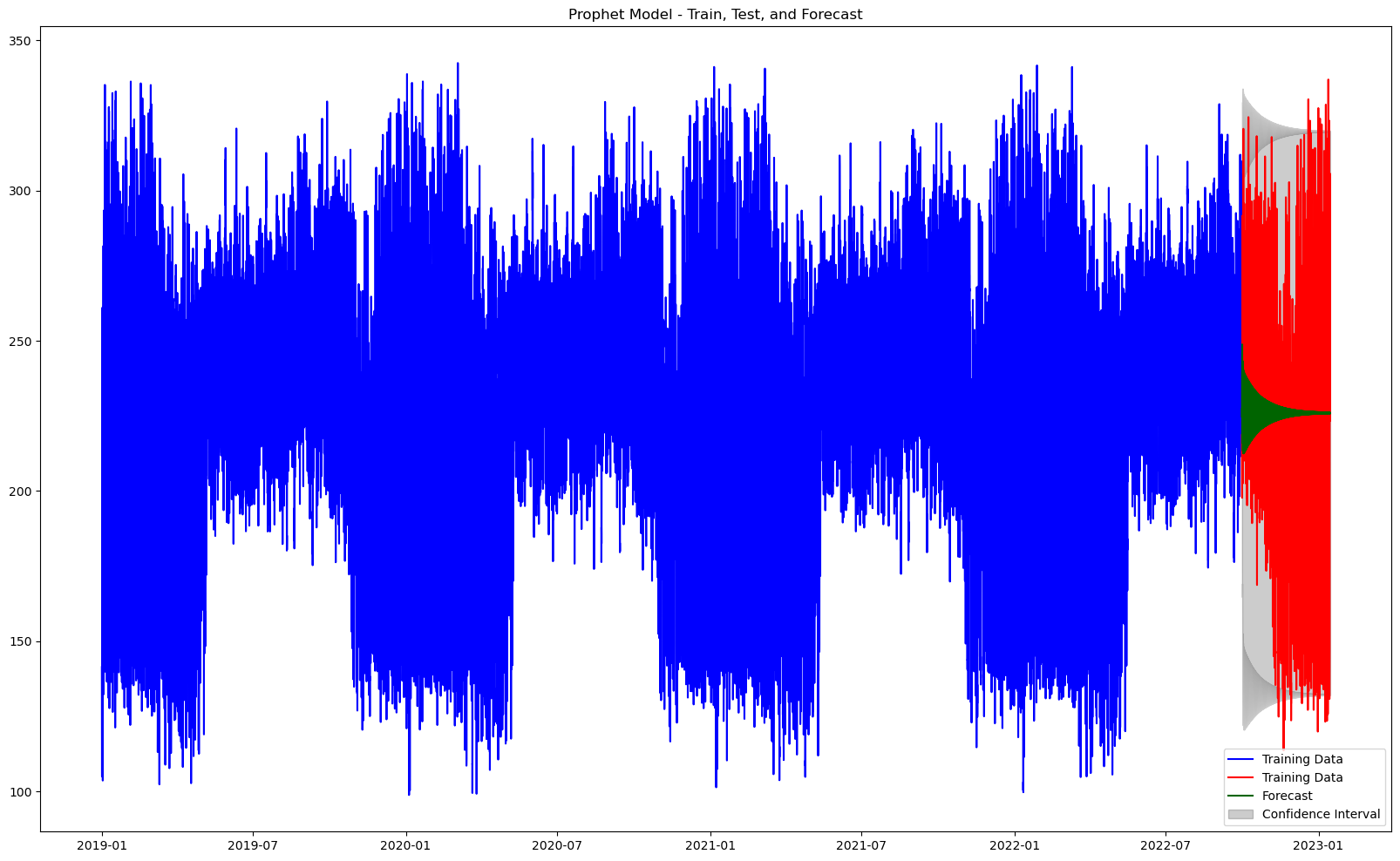


Figure 3.21: 45/3 (months) train/test and forecast using ARIMA.

This plot shows the results of ARIMA predicting model for the data after augmentation for the period 45/3 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations from October 2022 to January 2023.
* The green part is the mean values, a line starting from October 2022 onward, showing the predicted electrical consumption.
* The confidence interval is the shaded area around the mean values, indicating the uncertainty of the predictions.

### **3.3.3 Prophet Modeling**

Prophet is an open-source forecasting tool developed by Facebook's Core Data Science team. It is designed to facilitate the analysis and forecasting of time series data. The model is particularly useful for datasets that exhibit strong seasonal effects and several seasons of historical data.

**Key Features**

1. **Ease of Use:** Prophet is designed to be user-friendly and straightforward. It requires minimal pre-processing and provides a simple interface for creating accurate forecasts.
2. **Handling Missing Data:** The model can accommodate missing data points and outliers without requiring extensive data cleaning.
3. **Seasonality and Holidays:** Prophet automatically accounts for daily, weekly, and yearly seasonality. Users can also add custom seasonalities and holidays to the model.
4. **Trend Changepoints:** The model can detect changes in trends and adjust forecasts accordingly, making it robust for data with multiple trend shifts.

#### Implementation in Python

Prophet is available as a library in Python, making it accessible for data scientists and analysts. To use Prophet, the fbprophet library is imported and a Prophet object is instantiated. The model is then fit to the data

**Advantages**

* **Automation:** Automatically detects and models seasonality and trend changepoints.
* **Flexibility:** Users can specify additional seasonalities and include holidays to improve forecasts.
* **Scalability:** Suitable for large datasets and can be used for both short-term and long-term forecasting.

The three periods of training and testing where trained and tested using Prophet model implementation in python.

The result of prediction using Prophet for the three periods before augmentation:

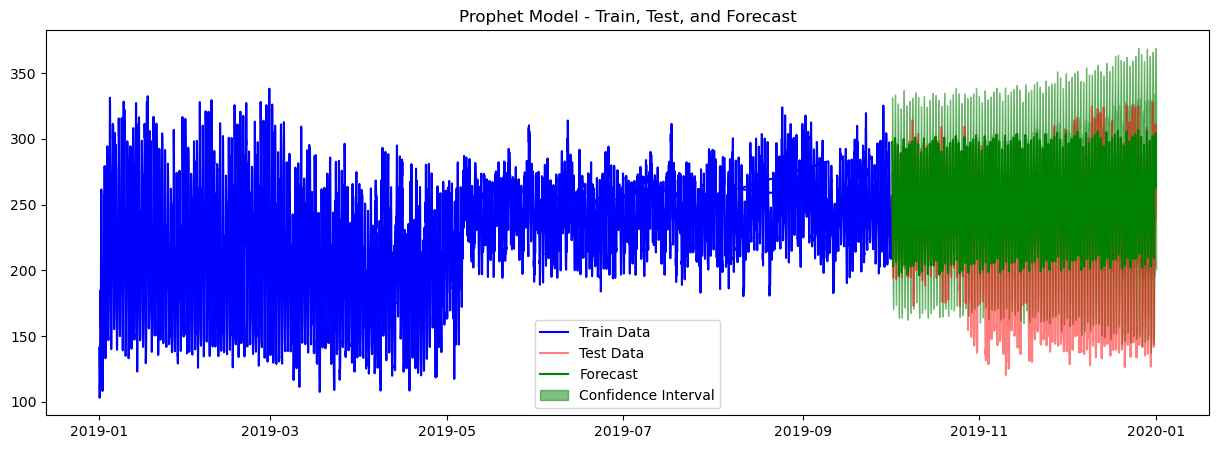


Figure 3.22: 9/3 (months) train/test and forecast using prophet.

This plot shows the results of Prophet predicting model for the period 9/3 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations from October 2019 to December 2019.
* The green part is the mean values, a line starting from October 2019 onward, showing the predicted electrical consumption.
* The confidence interval is the green shaded area around the mean values, indicating the uncertainty of the predictions.

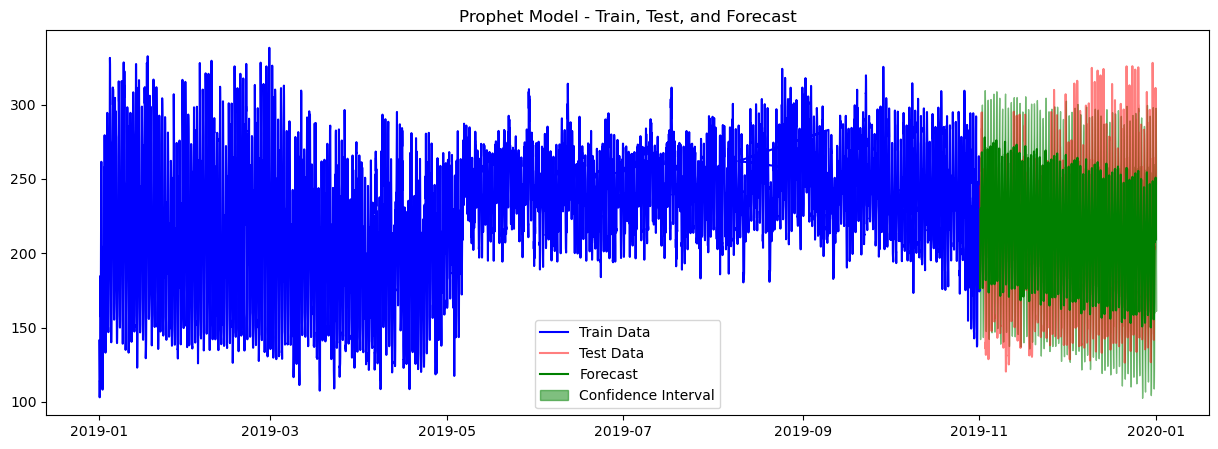


Figure 3.23: 10/2 (months) train/test and forecast using prophet.

This plot shows the results of Prophet predicting model for the period 10/2 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations from November 2019 to December 2019.
* The green part is the mean values, a line starting from November 2019 onward, showing the predicted electrical consumption.
* The confidence interval is the green shaded area around the mean values, indicating the uncertainty of the predictions.

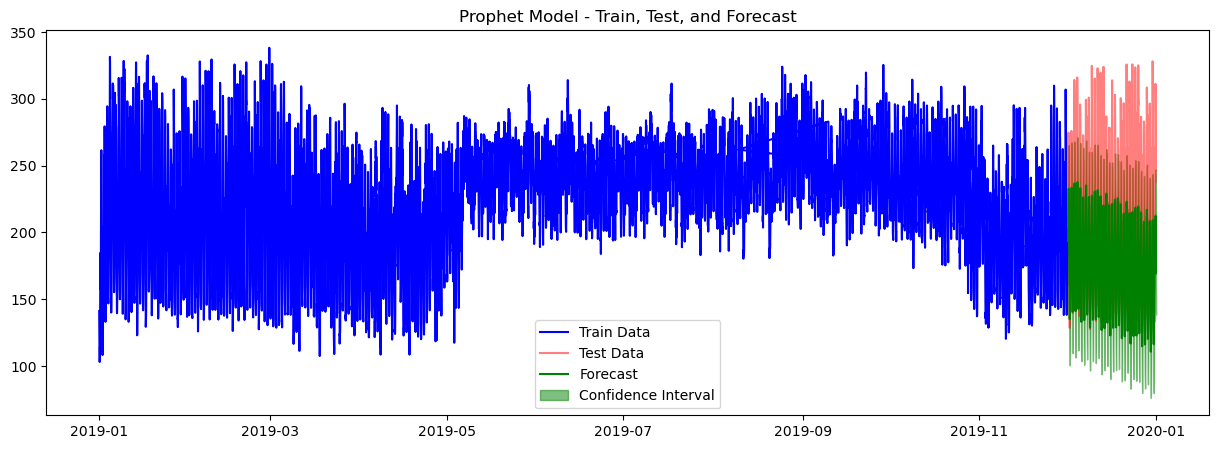


Figure 3.24: 11/1 (months) train/test and forecast using prophet.

This plot shows the results of Prophet predicting model for the period 11/1 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations for December 2019.
* The green part is the mean values, a line starting from December 2019 onward, showing the predicted electrical consumption.
* The confidence interval is the shaded area around the mean values, indicating the uncertainty of the predictions.

The result of prediction using Prophet for the three periods after augmentation:

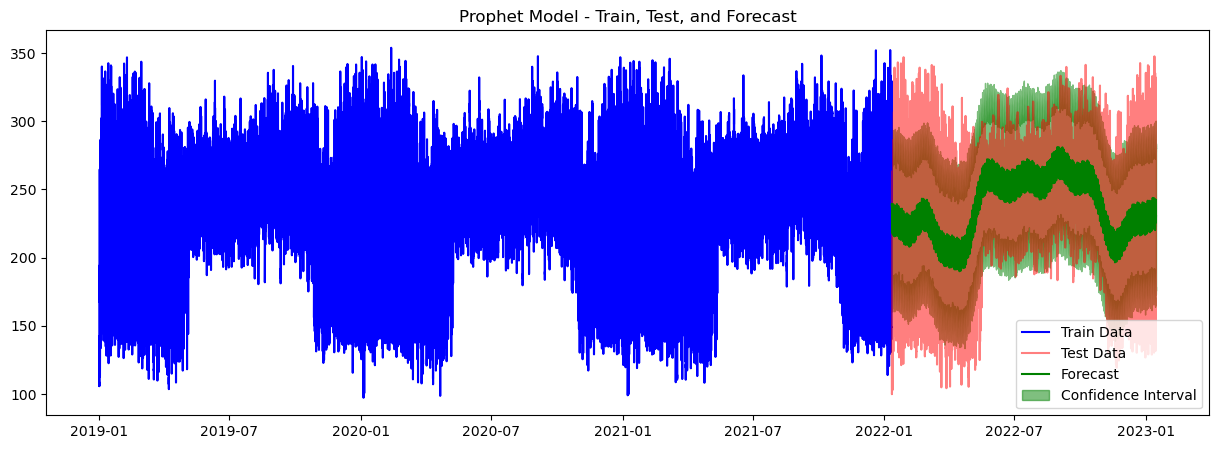


Figure 3.25: 36/12 (months) train/test and forecast using prophet.

This plot shows the results of Prophet predicting model for the data after augmentation for the period 36/12 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations from January 2022 to January 2023.
* The green part is the mean values, a line starting from January 2022 onward, showing the predicted electrical consumption.
* The confidence interval is the shaded area around the mean values, indicating the uncertainty of the predictions.

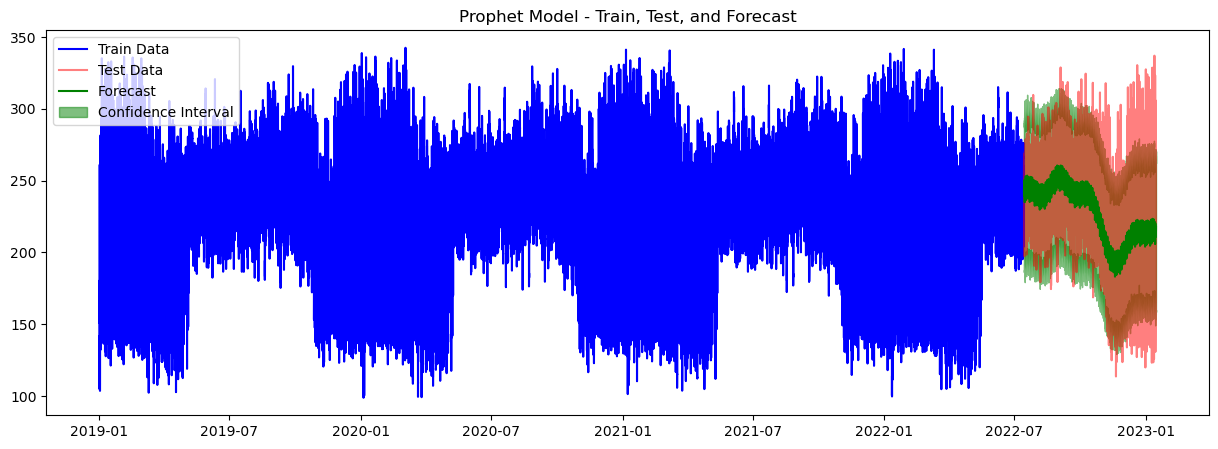


Figure 3.26: 42/6 (months) train/test and forecast using prophet.

This plot shows the results of Prophet predicting model for the data after augmentation for the period 42/6 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations from June 2022 to January 2023.
* The green part is the mean values, a line starting from June 2022 onward, showing the predicted electrical consumption.
* The confidence interval is the shaded area around the mean values, indicating the uncertainty of the predictions.

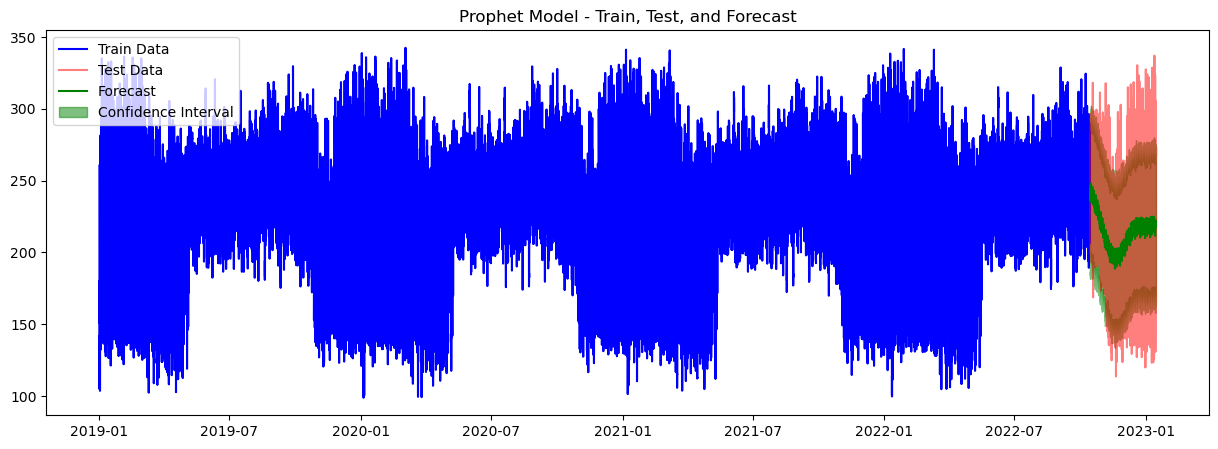


Figure 3.26: 45/3 (months) train/test and forecast using prophet

This plot shows the results of Prophet predicting model for the data after augmentation for the period 45/3 months of training and testing.

* The blue part is the training data appearing on the left, reflecting the historical data.
* The red part is the testing data appearing on the right, showing actual observations from October 2022 to January 2023.
* The green part is the mean values, a line starting from October 2022 onward, showing the predicted electrical consumption.
* The confidence interval is the shaded area around the mean values, indicating the uncertainty of the predictions.

# 

# **CHAPTER 4: RESULTS AND ANALYSIS**

## **4.1 Results of predictions**

The MAPE (mean absolute percentage error) for all prediction where calculated.

**Modeling with ARIMA before augmenting**

|  |  |  |  |
| --- | --- | --- | --- |
| Train/test(months) | Phase 1 | Phase 2 | Phase 3 |
| 9 / 3 | 81.35% | 80.32% | 80.74% |
| 10 / 2 | 77.00% | 76.01% | 76.37% |
| 11 / 1 | 76.46% | 76.45% | 77.01% |

Table 3.1: MAPE of modeling with ARIMA before augmenting

**Modeling with ARIMA after augmenting**

|  |  |  |  |
| --- | --- | --- | --- |
| Train/test(months) | Phase 1 | Phase 2 | Phase 3 |
| 36 / 12 | 82.11% | 81.41% | 82.01% |
| 42 / 6 | 84.99% | 84.60% | 84.88% |
| 45 / 3 | 81.47% | 80.75 % | 81.12% |

Table 3.2: MAPE of modeling with ARIMA after augmenting

**Modeling with Prophet before augmenting**

|  |  |  |  |
| --- | --- | --- | --- |
| Train/test(months) | Phase 1 | Phase 2 | Phase 3 |
| 9 / 3 | 79.40% | 78.45 % | 79.25% |
| 10 / 2 | 86.03% | 85.55% | 85.98% |
| 11 / 1 | 82.71% | 83.06% | 83.07% |

Table 3.3: MAPE of modeling with Prophet before augmenting

**Modeling with Prophet after augmenting**

|  |  |  |  |
| --- | --- | --- | --- |
| Train/test(months) | Phase 1 | Phase 2 | Phase 3 |
| 36 / 12 | 82.51% | 82.00% | 82.28% |
| 42 / 6 | 85.38% | 85.00% | 85.26% |
| 45 / 3 | 81.07% | 80.60% | 80.93% |

Table 3.4: MAPE of modeling with Prophet after augmenting

## **4.2 Comparative Analysis**

Comparing the results of both ARIMA and Prophet Models result before augmentation for all periods:

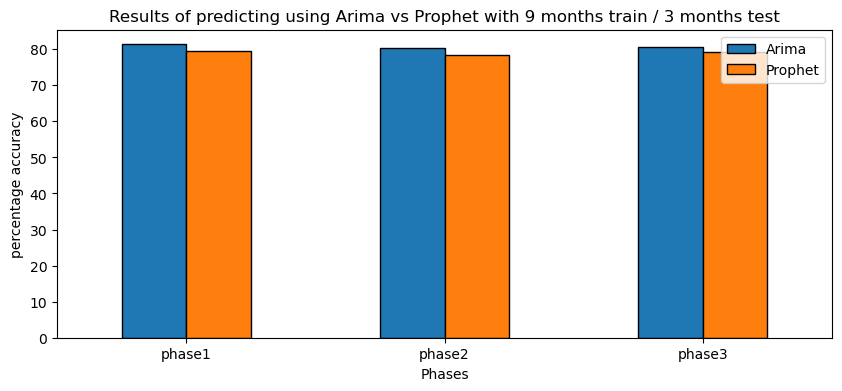


Figure 4.1: ARIMA vs Prophet with 9/3 train and test.

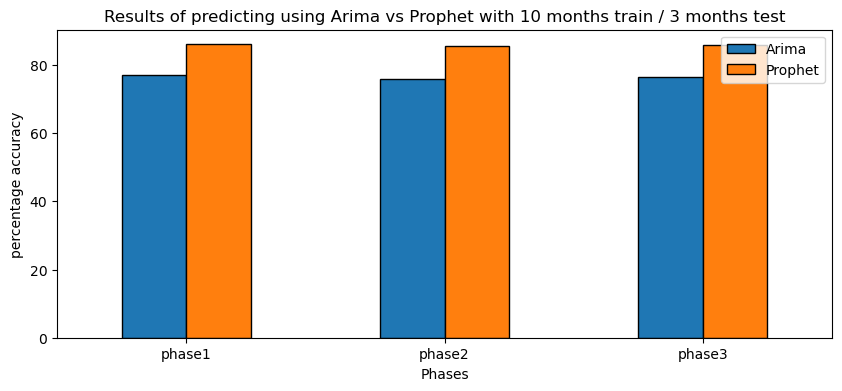


Figure 4.2: ARIMA vs Prophet with 10/2 train and test.

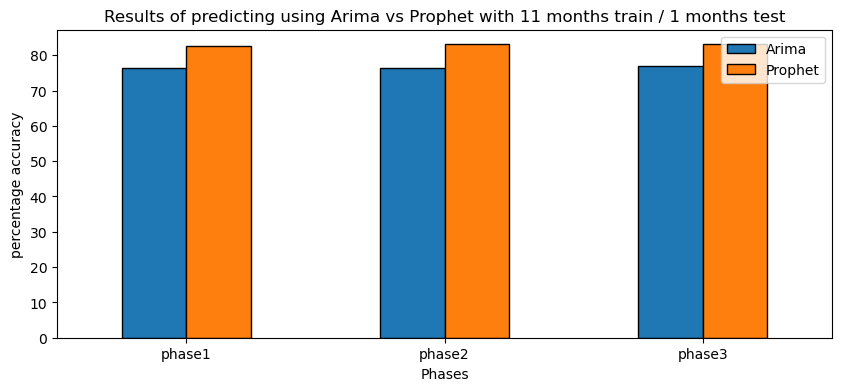


Figure 4.3: ARIMA vs Prophet with 11/1 train and test.

Comparing the results of both ARIMA and Prophet Models result after augmentation for all periods:

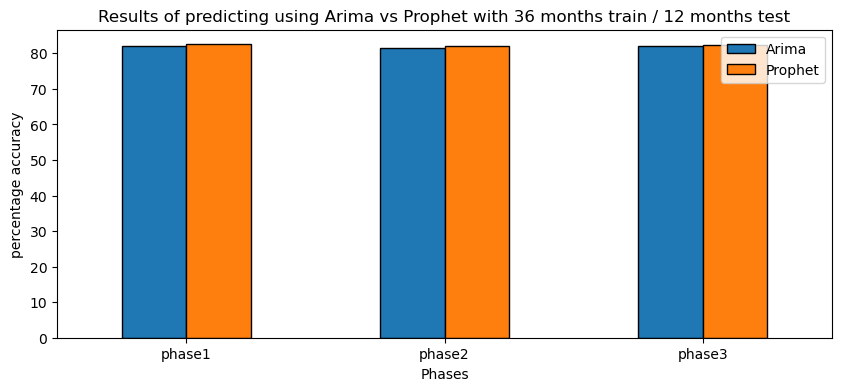


Figure 4.4: ARIMA vs Prophet with 36/12 train and test.

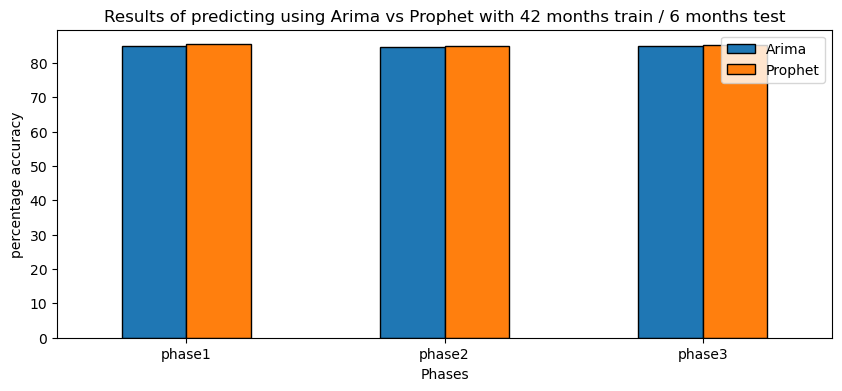


Figure 4.5: ARIMA vs Prophet with 42/6 train and test.

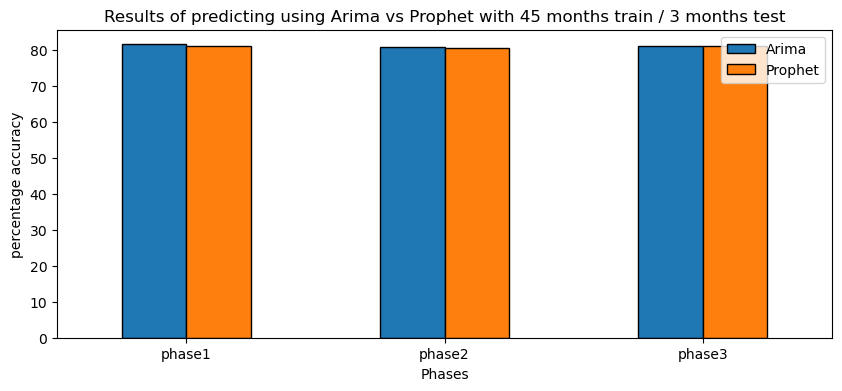


Figure 4.6: ARIMA vs Prophet with 45/3 train and test.

The plots shows that in the three periods before augmentation there where a difference with MAPE for the two models, the ARIMA model had a better result in the period 9/3 train and test, but clearly the Prophet had a better results with the other periods, when the period increased the prophet had better results.

The difference between the two models seem to be very little with the augmented data

Now let’s compare the results of ARIMA before and after augmentation.

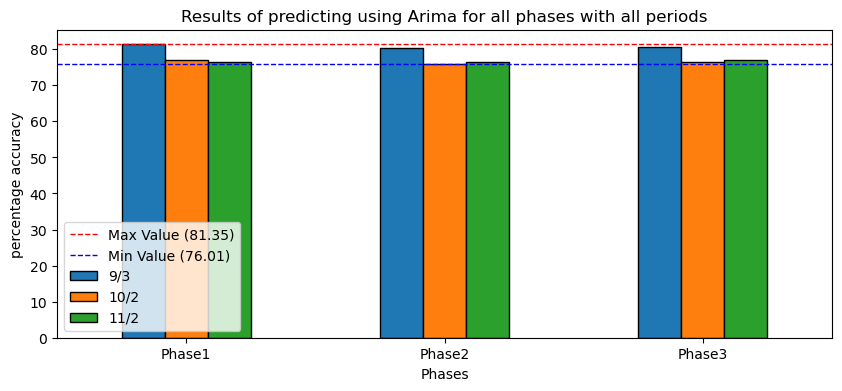


Figure 4.7: ARIMA results with all periods before augmentation.

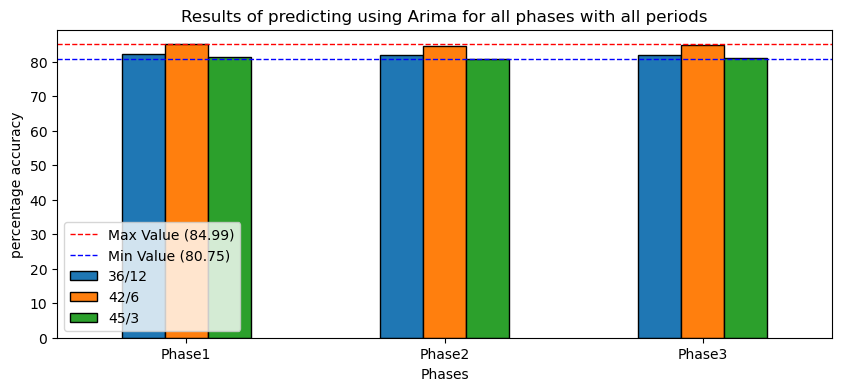


Figure 4.8: ARIMA results with all periods after augmentation.

With one-year data (data before augmentation) the highest MAPE where at 9/3 train and test, and there was a clear difference with the min and max value with almost 0.05 difference. With 4-years data (data after augmentation) the highest MAPE where with 42/6 train and test period, with almost 0.0 4difference between min and max.

Now let’s compare the results of Prophet before and after augmentation.

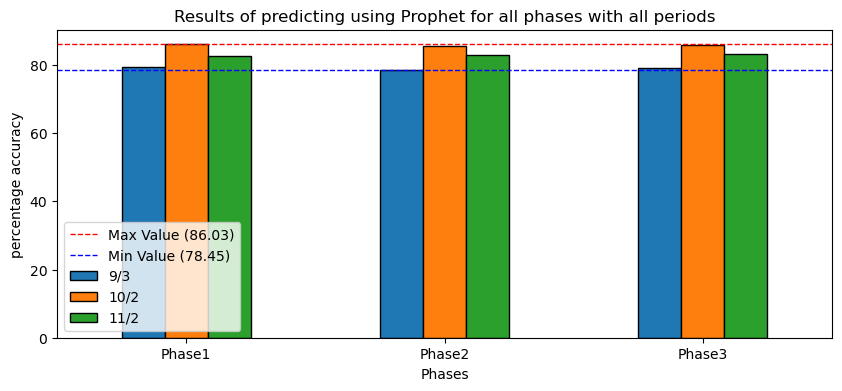


Figure 4.9: Prophet results with all periods before augmentation.

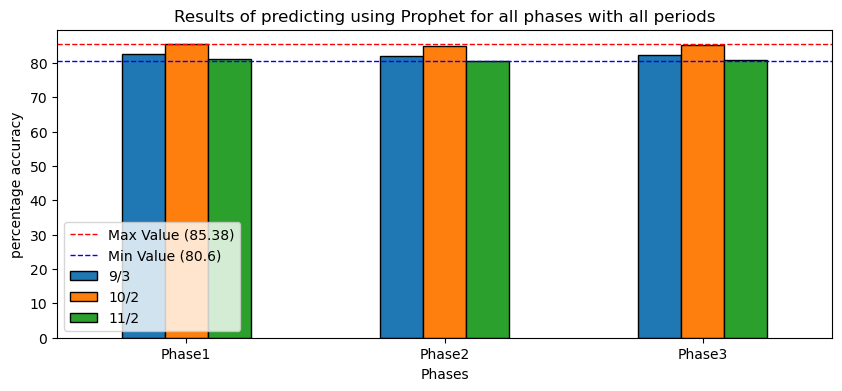


Figure 4.10: Prophet results with all periods after augmentation.

With one-year data (data before augmentation) the highest MAPE where at 10/2 train and test, and there was a clear difference with the min and max value with almost 0.06 difference. With 4-years data (data after augmentation) the highest MAPE where with 42/6 train and test period, with almost 0.0 4difference between min and max.

Now let’s compare periods before and after augmentation.

For ARIMA periods

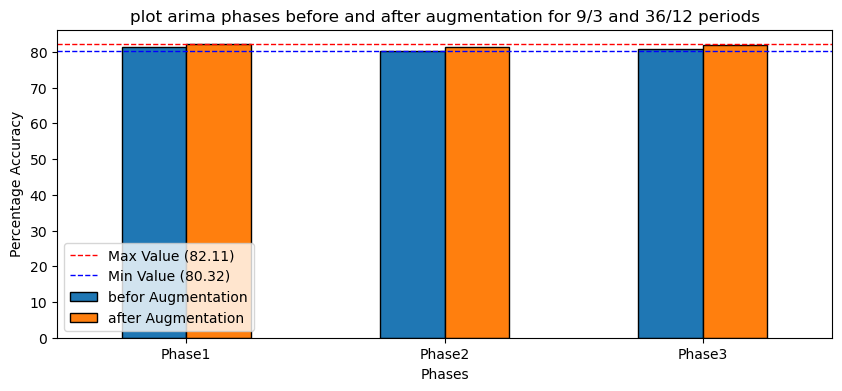


Figure 4.11: ARIMA results with 9/3 and 36/12 periods.

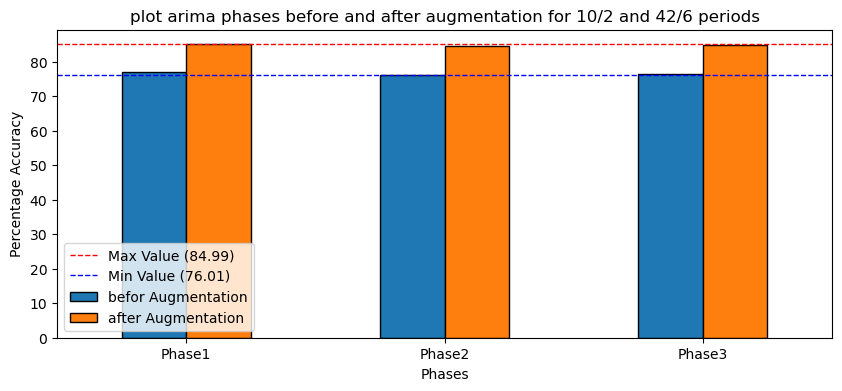


Figure 4.12: ARIMA results with 10/2 and 42/6 periods.

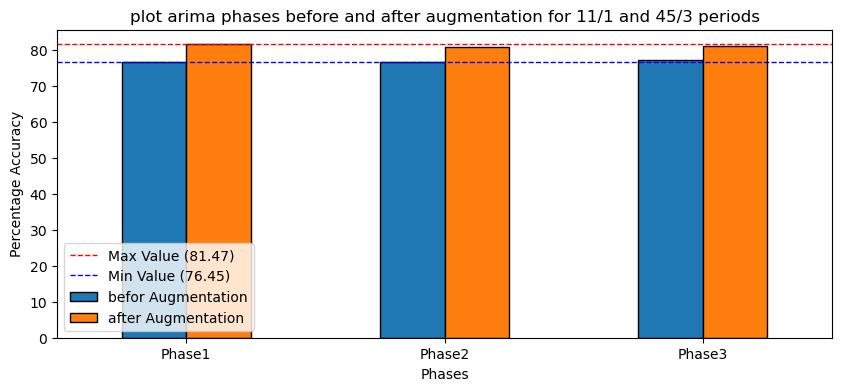


Figure 4.13: ARIMA results with 11/1 and 45/3 periods.

Clearly the range of the results of ARIMA model where increased after augmentation for all periods.

For Prophet periods

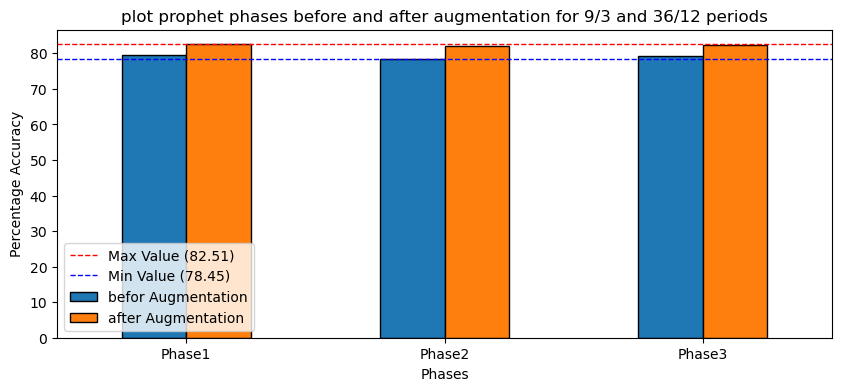


Figure 4.14: Prophet results with 9/3 and 36/12 periods.

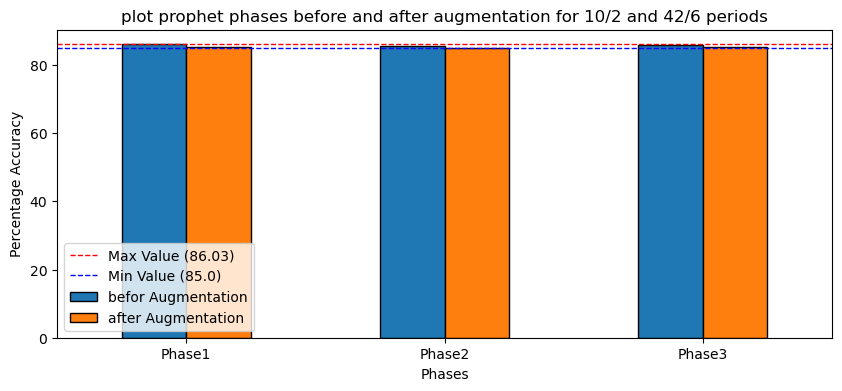


Figure 4.15: Prophet results with 10/2 and 42/6 periods.

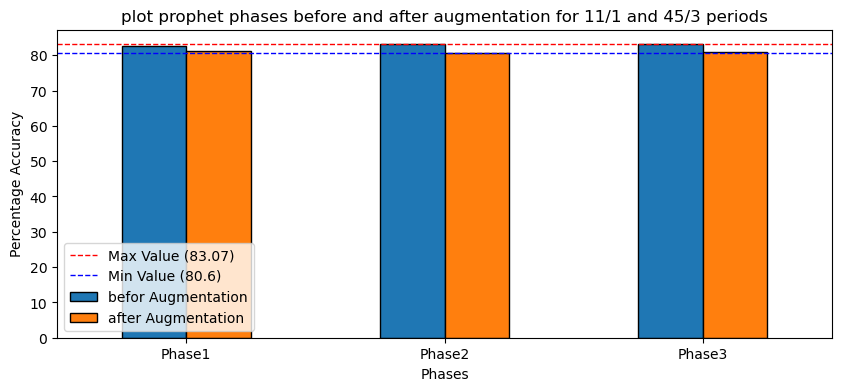


Figure 4.16: Prophet results with 11/1 and 42/3 periods.

The results for Prophet did not clearly affected with the augmentation.

## **4.3 conclusion**

* Range of accuracy of the two models had increased after augmentation
* The 42/6 train and test period had the highest MAPE for both ARIMA and Prophet with augmented data.
* Results of ARIMA and prophet where close with augmented data.
* In general the prophet model is better than ARIMA for this time series data, since it had a greater MAPE range in most periods of training and testing.

# **CHAPTER 5: FUTURE WORK AND POTENTIAL DEVELOPMENT**

This project is a part of a scientific research that uses 6 models to predict the future power consumption, Tubas electrical company case study, and chooses the best model to develop for the company.

The future of my project will be:

* Train and test another 4 models with the same data and same periods of training and testing.
* Choose the model with the best accuracy to be developed.
* Writing this work as a scientific research paper and publish it.

# **REFRENCES**

1- Pinheiro, M. G., Madeira, S. C., & Francisco, A. P. (2023). Short-term electricity load forecasting—A systematic approach from system level to secondary substations. *Applied Energy*, *332*, 120493.

‏

2- L. Huang, G. Zhou, J. Li and S. Yang, "Short-term Load Prediction Based on User Behaviors Analysis," 2020 4th International Conference on Smart Grid and Smart Cities (ICSGSC), Osaka, Japan, 2020, pp. 18-23, doi: 10.1109/ICSGSC50906.2020.9248540.

3- A. Garulli, S. Paoletti and A. Vicino, "Models and Techniques for Electric Load Forecasting in the Presence of Demand Response," in IEEE Transactions on Control Systems Technology, vol. 23, no. 3, pp. 1087-1097, May 2015, doi: 10.1109/TCST.2014.2361807.

4-M. A. Ahmed, K. S. Reddy, J. Prakash, A. Rai and B. P. Singh, "Statistical analysis of load and its frequency response for load forecasting in a medium voltage distribution system," 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, 2017, pp. 1146-1149, doi: 10.1109/ICECDS.2017.8389621.

5- Z. Cao, C. Wan, Z. Zhang, F. Li and Y. Song, "Hybrid Ensemble Deep Learning for Deterministic and Probabilistic Low-Voltage Load Forecasting," in IEEE Transactions on Power Systems, vol. 35, no. 3, pp. 1881-1897, May 2020, doi: 10.1109/TPWRS.2019.2946701.

6-X. Qifeng, W. Qiang, Y. Zhilin and L. Shufen, "The Monthly Electricity Load Forecast Based on Composite Model," 2015 7th International Conference on Information Technology in Medicine and Education (ITME), Huangshan, China, 2015, pp. 664-667, doi: 10.1109/ITME.2015.57.

7- J. Yang and Q. Wang, "A Deep Learning Load Forcasting Method Based on Load Type Rcognition," 2018 International Conference on Machine Learning and Cybernetics (ICMLC), Chengdu, China, 2018, pp. 173-177, doi: 10.1109/ICMLC.2018.8527022.

8- J. Yang and Q. Wang, "A Deep Learning Load Forcasting Method Based on Load Type Rcognition," 2018 International Conference on Machine Learning and Cybernetics (ICMLC), Chengdu, China, 2018, pp. 173-177, doi: 10.1109/ICMLC.2018.8527022.

9- A. Bracale, G. Carpinelli, P. De Falco and T. Hong, "Short-term industrial load forecasting: A case study in an Italian factory," 2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), Turin, Italy, 2017, pp. 1-6, doi: 10.1109/ISGTEurope.2017.8260176.

10-H. Dong, Y. Gao, X. Meng and Y. Fang, "A Multifactorial Short-Term Load Forecasting Model Combined With Periodic and Non-Periodic Features - A Case Study of Qingdao, China," in IEEE Access, vol. 8, pp. 67416-67425, 2020, doi: 10.1109/ACCESS.2020.2986031.

11-Lee Y-S, Tong L-I. Forecasting time series using a methodology based on autore gressive integrated moving average and genetic programming. Knowl-Based Syst 2011;24(1):66–72.

12- K.-B. Song, Y.-S. Baek, D. H. Hong, and G. Jang, “Short-term load forecasting for the holidays using fuzzy linear regression method,” IEEE Trans. Power Syst., vol. 20, no. 1, pp. 96–101, Feb. 2005.

13- Choi H, Ryu S, Kim H. Short-term load forecasting based on ResNet and LSTM. In: 2018 IEEE international conference on communications, control, and computing technologies for smart grids. IEEE; 2018, p. 1–6.

14- Bouktif S, Fiaz A, Ouni A, Serhani MA. Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches. Energies 2018;11(7):1636.

15-J. Vermaak and E. C. Botha, Recurrent neural networks for short-term load forecasting, IEEE Trans. Power Syst., vol. 13, no. 1, pp. 126132, Feb. 1998.

16-D.C.Park,M.A.El-Sharkawi,R.J.Marks,L.E.Atlas,andM.J.Damborg, Electric load forecasting using an arti cial neural network, IEEE Trans. Power Syst., vol. 6, no. 2, pp. 442449, May 1991.

17- T. Hong. Short term electric load forecasting. Ph.D. dissertation, North Carolina State University, 2010.

18-M. Espinoza, J. A. K. Suykens, R. Belmans, and B. De Moor, “Electric load forecasting,” IEEE Control Syst., vol. 27, no. 5, pp. 43–57, Oct. 2007.

19-E. E. Elattar, J. Goulermas, and Q. H. Wu, “Electric load forecasting based on locally weighted support vector regression,” IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 40, no. 4, pp. 438–447, Jul. 2010

20-Nair V Hinton G E ”Rectified linear units improve restricted Boltzmann machines”[C] International Conference on Machine Learning DBLP 2010 807-814

21-Yao Lixiao;Liu Xueqin.A Wavelet Analysis Based Combined Monthly Load Forecasting[H].Power Syst em Technology Vol.31(19),pp.5-68.Article in a conference proceedings.

22-S. Fan and R. J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model," IEEE Trans. Power Syst., vol. 27, no. 1, pp. 134-141, Feb. 2012.

23- P. Kankanala, S. Das, and A. Pahwa, "AdaBoost+: An ensemble learning approach for estimating weather-related outages in distribution systems," IEEE Trans. Power Syst., vol. 29, no. 1, pp. 359-367, Jan. 2013.

24- D. L. Shrestha and D. P. Solomatine, "Experiments with AdaBoost.RT, an improved boosting scheme for regression," Neural Computation, vol. 18, no. 7, pp. 1678-1710, July 2006.

25- H. Drucker, "Improving regressors using boosting techniques," in International Conference on Machine Learning, 1997, pp. 107-115.

26-M. Assaad and H. Cardot, "A new boosting algorithm for improved time-series forecasting with recurrent neural networks," Inf. Fusion, vol. 9, no. 1, pp. 41-55, Jan. 2008.

27-S.-J. Huang and K.-R. Shih, “Short-term load forecasting via ARMA model identification including non-Gaussian process considerations,” IEEE Trans. Power Syst., vol. 18, no. 2, pp. 673–679, May 2003.

28-Lim B, Arık SÖ, Loeff N, Pfister T. Temporal fusion transformers for interpretable multi-horizon time series forecasting. Int J Forecast 2021.

29-Chakhchoukh Y, Panciatici P, Mili L. Electric load forecasting based on statistical robust methods. IEEE Trans Power Syst 2010;26(3):982–91.

30-Pierrot A, Goude Y. Short-term electricity load forecasting with generalized additive models. Proc ISAP Power 2011;2011.

31-Cho H, Goude Y, Brossat X, Yao Q. Modelling and forecasting daily electricity load via curve linear regression. In: Modeling and stochastic learning for forecasting in high dimensions. Springer; 2015, p. 35–54.

32-Amato U, Antoniadis A, De Feis I, Goude Y, Lagache A. Forecasting high resolution electricity demand data with additive models including smooth and jagged components. Int J Forecast 2021;37(1):171–85.

33-N. Amjady, Short-term bus load forecasting of power systems by a new hybrid method, IEEE Trans. Power Syst., vol. 22, no. 1, pp. 333341, Feb. 2007.

34-S. Fan and L. Chen, Short-term load forecasting based on an adaptive hybrid method, IEEE Trans. Power Syst., vol. 21, no. 1, pp. 392401, Feb. 2006.

35-Sela, J.G., "Weather forecasting on parallel architectures", Journal of Parallel Computing, Vol. 21,1995, pp. 1639-1654.

36- Baillie, c., Michalakes, J. and Skalin, R., "Regional weather modeling on parallel computers", Journal of Parallel Computing, Vol. 23, 1997, pp. 2135-2142.

37-Jingchao Zhang, Anhe Yan, Zhuoya Chen and Kun Gao, "Dynamic synthesis load modeling approach based on load survey and load curves analysis," 2008 Third International Conference on Electric Utility Deregulation and Restructuring and Power Technologies, Nanjing, 2008, pp. 1067-1071, doi: 10.1109/DRPT.2008.4523565.

38- X. Pei, "Prophet algorithm-based power load forecasting model," 2023 IEEE 3rd International Conference on Electronic Technology, Communication and Information (ICETCI), Changchun, China, 2023, pp. 1498-1503, doi: 10.1109/ICETCI57876.2023.10176778. keywords: {Industries;Analytical models;Load forecasting;Production;Predictive models;Power grids;Data models;Prophet algorithm;Power load forecasting;COVID-19;Regional power grid},