



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 Daraghme

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 model will be trained and tested with real data from Tubas Electrical Company, then the results will be compared.

The data is the 3 phases current values for 2019 year, the data will be preprocessed and analysed, then used to train the models, some different periods of training data will be chose to train the models. Then I will augment the data into 4 years and the same models will be trained with the augmented data.

The results will be analyzed and compared between the models considering periods of training and data augmentation.

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 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 programming 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 were sent by the company in a csv file that contains many attributes as appearing in figure (3.1).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	S	T	U	V	W	X
1	Tayaseer																						
2																							
3	Date	Time	SMS Type	Status	I1	I2	I3	V1	V2	V3	PF	THDI1	THDI2	TDH13	THDV1	THDV2	THDV3	TAE1MP	TAE2XP	TRE1MP	TRE2XP	TAPP1MP	TAPP2XP
4	January 01, 2019	12:00 AM	Periodic	Close	141.43	143.19	143.35	34.04	34.32	34.12	95.58	6.6	7	5.8	6.6	6.2	6.2	333.915	6.837	231.093	1	413.495	31.823
5	January 01, 2019	01:00 AM	Periodic	Close	124.01	123.9	124.6	34.25	34.56	34.32	92.73	9.4	10.5	7.8	9.6	8.2	7.9	333.921	6.837	231.096	1	413.503	31.823
6	January 01, 2019	02:00 AM	Periodic	Close	114.84	114.41	115.01	34.2	34.44	34.26	91.99	10.2	11.3	7.9	10.4	8.2	8	333.928	6.837	231.098	1	413.510	31.823
7	January 01, 2019	03:00 AM	Periodic	Close	107.81	107.5	107.43	34.43	34.66	34.49	90.23	13.3	14.5	9.3	13.3	9.6	9.5	333.934	6.837	231.101	1	413.517	31.823
8	January 01, 2019	04:00 AM	Periodic	Close	103.54	100.19	102.38	34.5	34.82	34.57	88.74	15.4	17.7	10.2	15.6	10.5	10.2	333.940	6.837	231.104	1	413.523	31.823
9	January 01, 2019	05:00 AM	Periodic	Close	107	105.53	107.16	34.46	34.72	34.5	90.45	13.7	15.5	9.5	13.6	9.7	9.4	333.945	6.837	231.106	1	413.529	31.823
10	January 01, 2019	06:00 AM	Periodic	Close	114.84	111.54	116.03	34.01	34.34	34.12	93.82	9.8	11.2	6.9	9.8	7.4	7.1	333.952	6.837	231.109	1	413.536	31.823
11	January 01, 2019	07:00 AM	Periodic	Close	113.19	112.45	114.44	34.05	34.28	34.12	94.89	10.2	11.1	6.7	10.1	6.9	6.9	333.958	6.837	231.111	1	413.543	31.823
12	January 01, 2019	08:00 AM	Periodic	Close	135.19	137.09	138.37	33.86	33.85	33.92	95.75	5.6	6	4.9	5.6	4.8	4.9	333.965	6.837	231.114	1	413.549	31.823
13	January 01, 2019	09:00 AM	Periodic	Close	184.95	190.03	187.94	33.32	33.64	33.38	96.62	2.2	2.6	2.9	2.4	3	3	333.975	6.837	231.116	1	413.559	31.823
14	January 01, 2019	10:00 AM	Periodic	Close	176.45	176.18	177.05	33.66	33.84	33.64	96.34	2.6	2.8	3.4	2.7	3.4	3.4	333.985	6.837	231.119	1	413.570	31.823
15	January 01, 2019	11:00 AM	Periodic	Close	163.78	166.68	165.62	33.61	33.77	33.62	95.95	2.7	2.8	3.3	2.8	3.2	3.3	333.995	6.837	231.121	1	413.580	31.823
16	January 01, 2019	12:00 PM	Periodic	Close	161.43	165.4	166.26	33.75	33.88	33.67	96.23	2.8	3	3.5	2.7	3.5	3.4	334.003	6.837	231.125	1	413.590	31.823
17	January 01, 2019	01:00 PM	Periodic	Close	157.26	160.12	161.74	33.6	33.76	33.62	95.23	2.9	2.9	3.5	2.8	3.6	3.6	334.013	6.837	231.127	1	413.600	31.823
18	January 01, 2019	02:00 PM	Periodic	Close	182.69	186.65	186.09	33.09	33.22	33.1	96.38	1.3	1.4	2.5	1.4	2.4	2.4	334.023	6.837	231.130	1	413.610	31.823
19	January 01, 2019	03:00 PM	Periodic	Close	190.02	196.13	194.87	33.06	33.32	33.05	96.12	1.1	1.1	2.6	1.1	2.6	2.6	334.033	6.837	231.133	1	413.621	31.823
20	January 01, 2019	04:00 PM	Periodic	Close	214.57	220.04	220.42	32.74	33.05	32.78	96.62	0.5	0.6	2.1	0.5	2.2	2.2	334.044	6.837	231.136	1	413.631	31.823
21	January 01, 2019	05:00 PM	Periodic	Close	254.4	260.35	259.12	32.13	32.51	32.2	97.26	0.5	0.5	1.7	0.5	1.7	1.7	334.057	6.837	231.139	1	413.645	31.823
22	January 01, 2019	06:00 PM	Periodic	Close	262.5	267.98	267.32	32.14	32.57	32.23	97.67	0.6	0.6	1.7	0.6	1.8	1.8	334.072	6.837	231.143	1	413.660	31.823
23	January 01, 2019	07:00 PM	Periodic	Close	248.88	251.3	250.11	32.43	32.89	32.53	97.7	0.8	0.9	1.9	0.8	1.9	1.9	334.086	6.837	231.146	1	413.675	31.823
24	January 01, 2019	08:00 PM	Periodic	Close	237.06	239.36	240.19	32.75	33.21	32.87	97.82	1	1.1	1.9	1.1	2	2	334.099	6.837	231.149	1	413.689	31.823
25	January 01, 2019	09:00 PM	Periodic	Close	227.76	229.82	230.31	33.26	33.67	33.31	97.41	1.1	1.3	2.1	1.3	2.4	2.3	334.113	6.837	231.152	1	413.701	31.823

Figure 3.1: Original Data

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

	A	B	C	D	E	F	G	H	I	J
1	Date	Time_24_h	I1	I2	I3					
2	1-Jan-19	0:00	141.43	143.19	143.35					
3	1-Jan-19	1:00	124.01	123.9	124.6					
4	1-Jan-19	2:00	114.84	114.41	115.01					
5	1-Jan-19	3:00	107.81	107.5	107.43					
6	1-Jan-19	4:00	103.54	100.19	102.38					
7	1-Jan-19	5:00	107	105.53	107.16					
8	1-Jan-19	6:00	114.84	111.94	116.03					
9	1-Jan-19	7:00	113.19	112.45	114.44					
10	1-Jan-19	8:00	135.19	137.09	138.37					
11	1-Jan-19	9:00	184.95	190.03	187.94					
12	1-Jan-19	10:00	176.45	176.18	177.05					
13	1-Jan-19	11:00	163.78	166.68	165.62					
14	1-Jan-19	12:00	161.43	165.4	166.26					
15	1-Jan-19	13:00	157.26	160.12	161.74					
16	1-Jan-19	14:00	182.69	186.65	186.09					
17	1-Jan-19	15:00	190.02	196.13	194.87					
18	1-Jan-19	16:00	214.57	220.04	220.42					
19	1-Jan-19	17:00	254.4	260.35	259.12					
20	1-Jan-19	18:00	262.5	267.98	267.32					
21	1-Jan-19	19:00	248.88	251.3	250.11					

Figure 3.2: columns to process

A plot for the data after interpolating it.

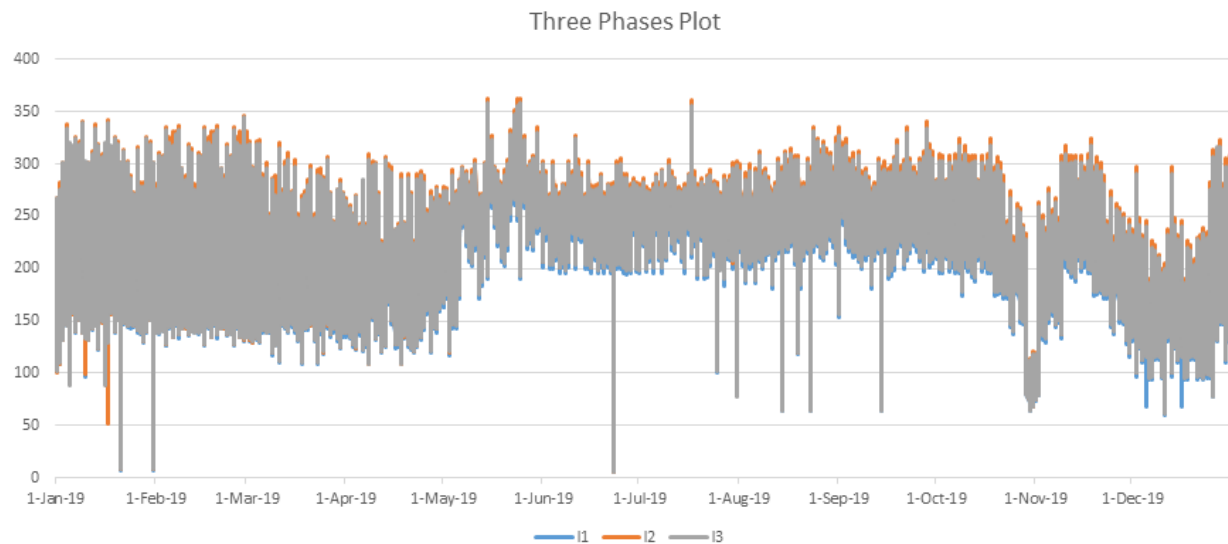


Figure 3.3 Data before interpolating

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.

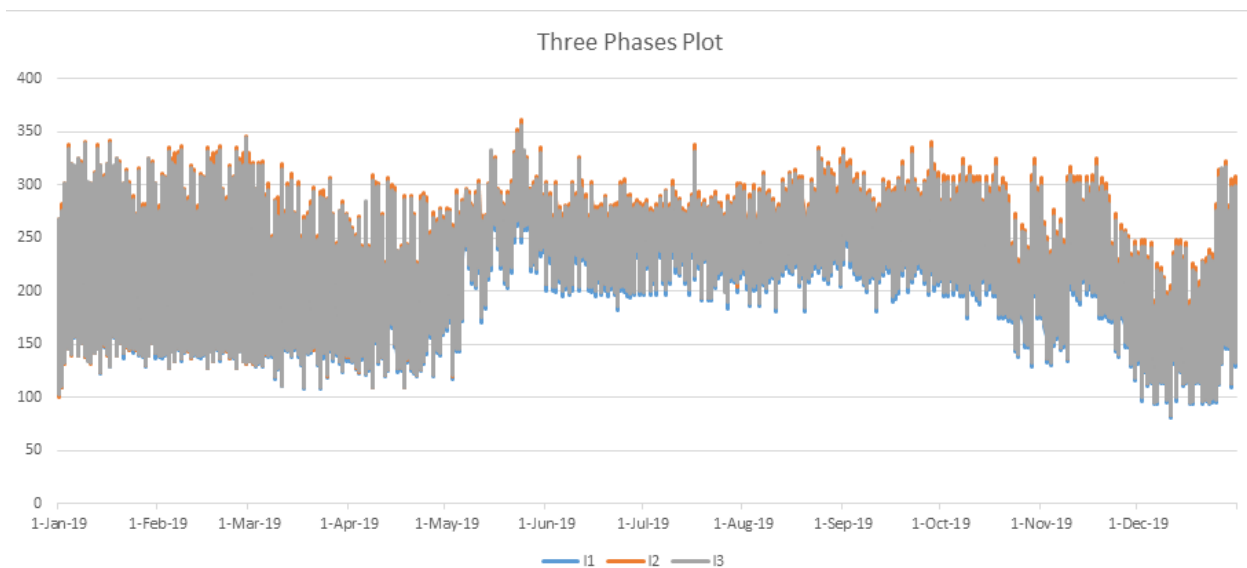


Figure 3.4: Data after interpolation

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.

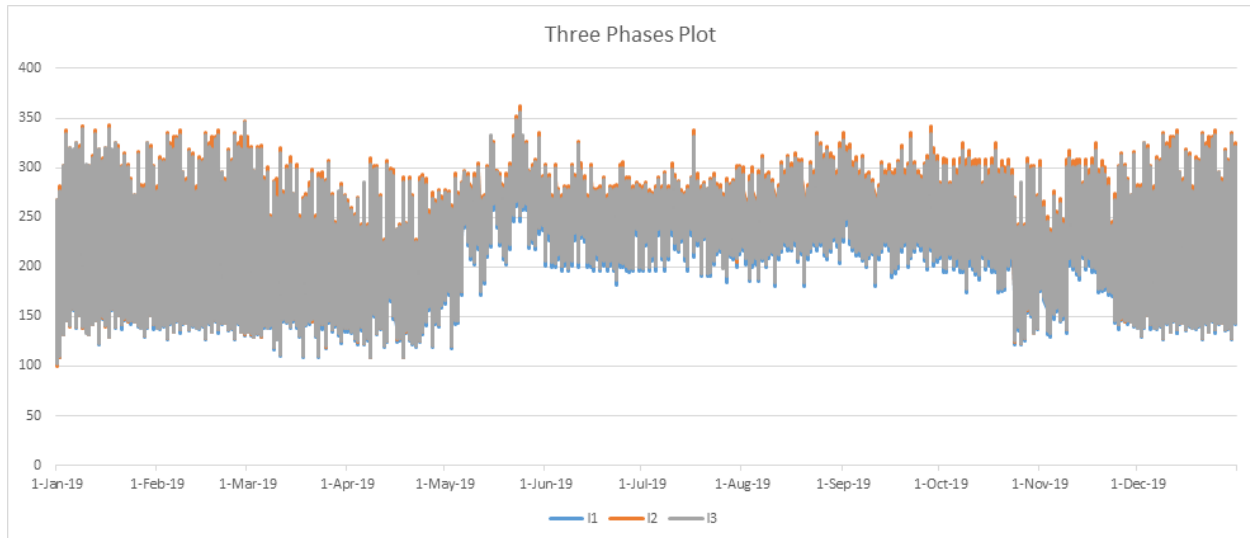


Figure 3.5: data after manipulation

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.

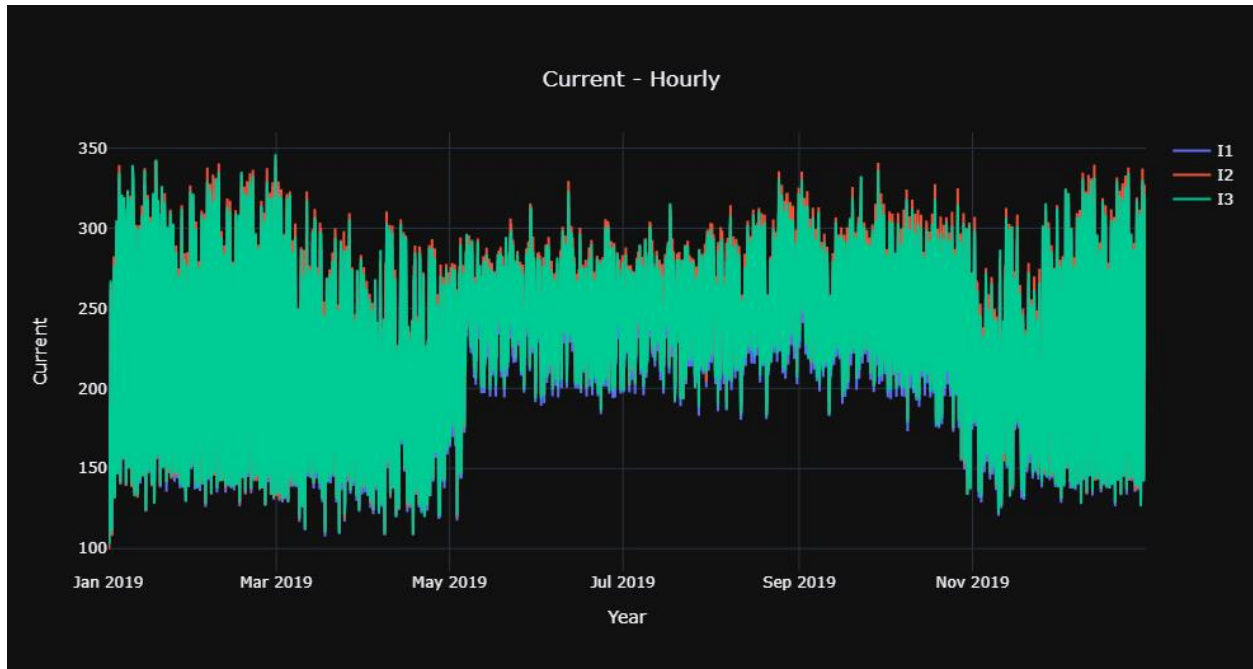


Figure 3.6: Hourly data plot.

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

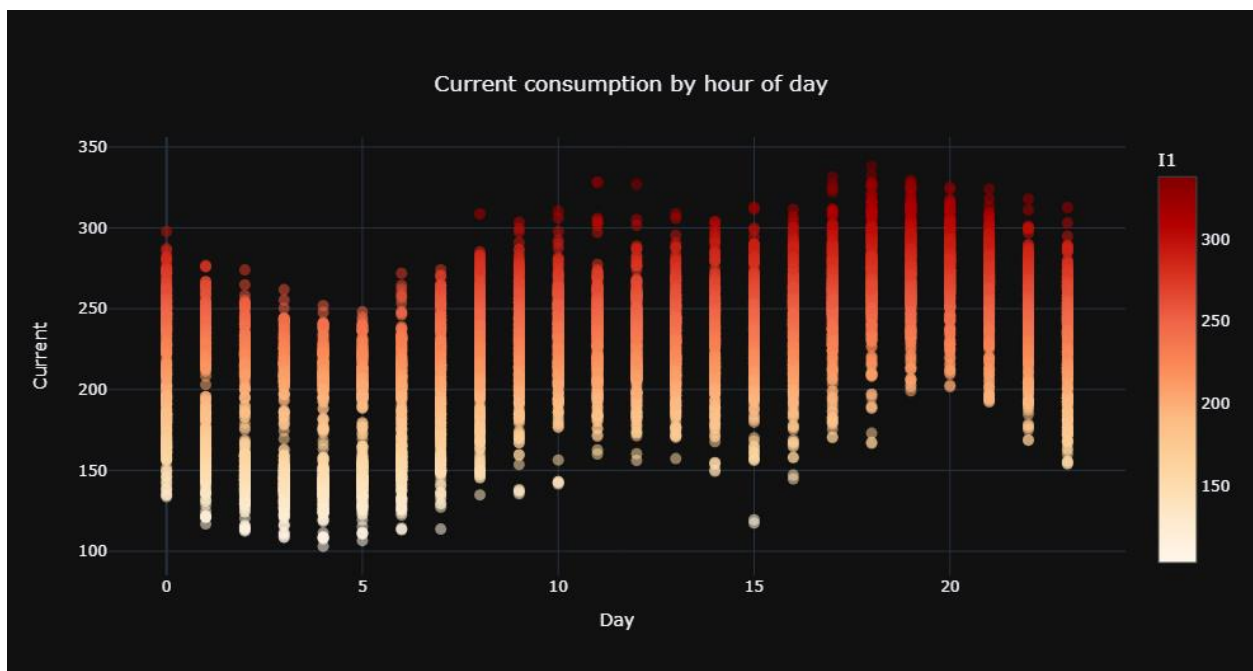


Figure 3.7: Hour of the day plot.

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.

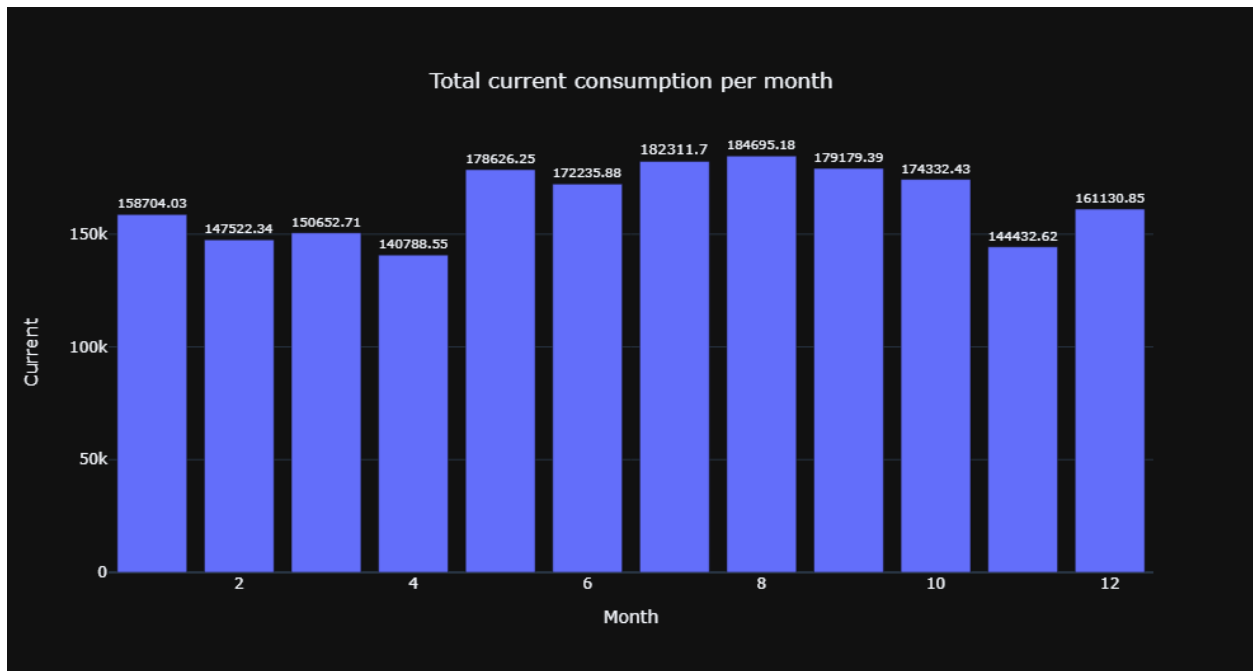


Figure 3.8: Monthly plot.

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

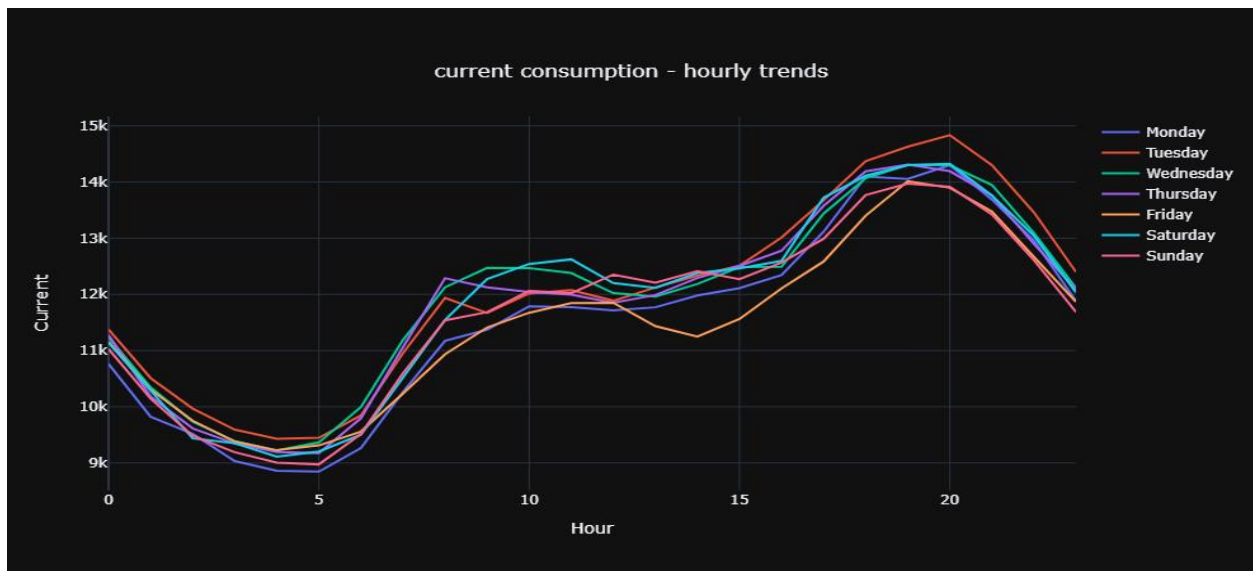


Figure 3.9: Day of week plot

3.3 Prediction

Two models were 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

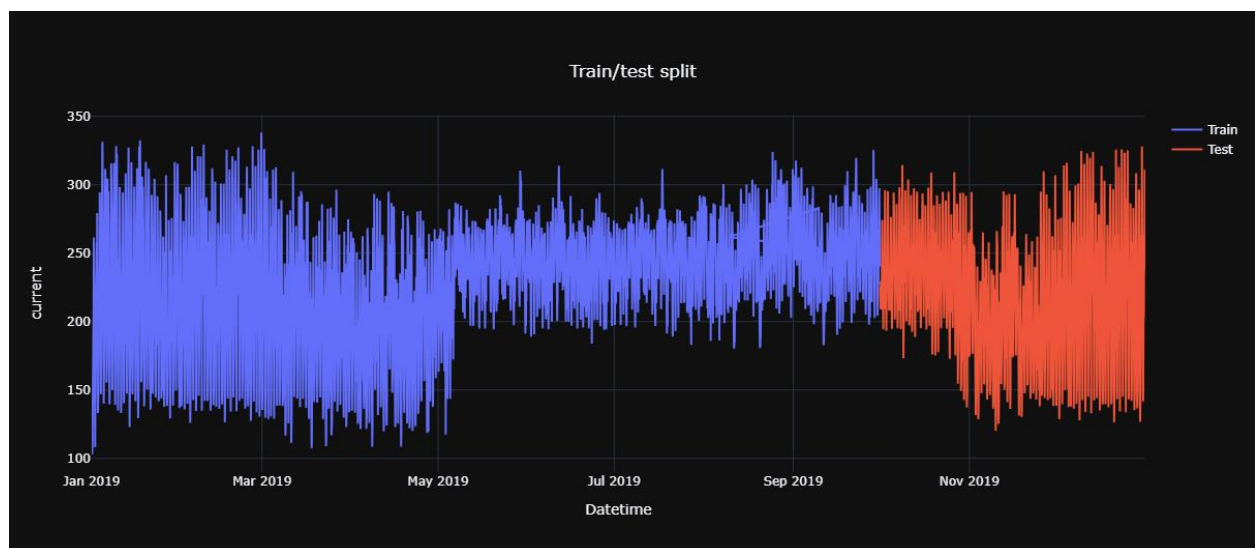


Figure 3.10: 9/3(months) train and test.

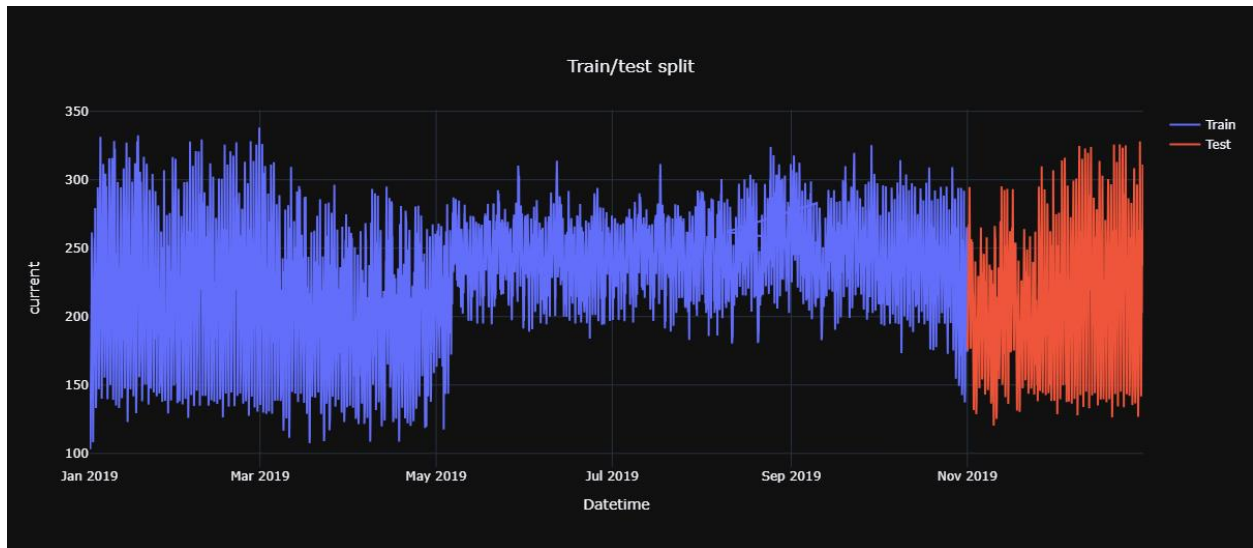


Figure 3.11: 10/2(months) train and test.

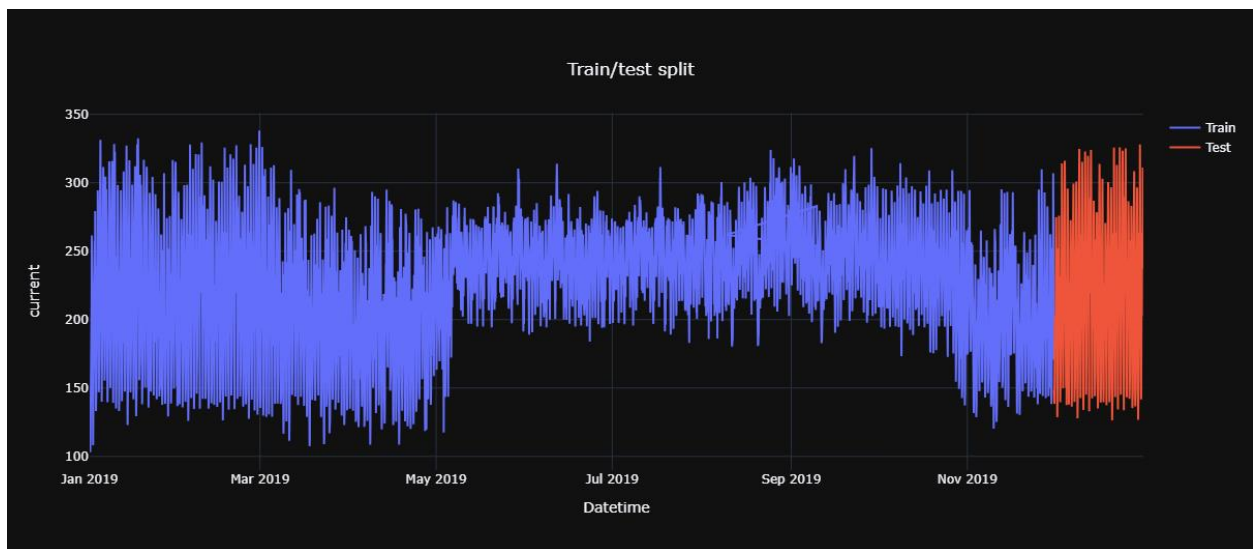


Figure 3.12: 11/1 (months) train and test.

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

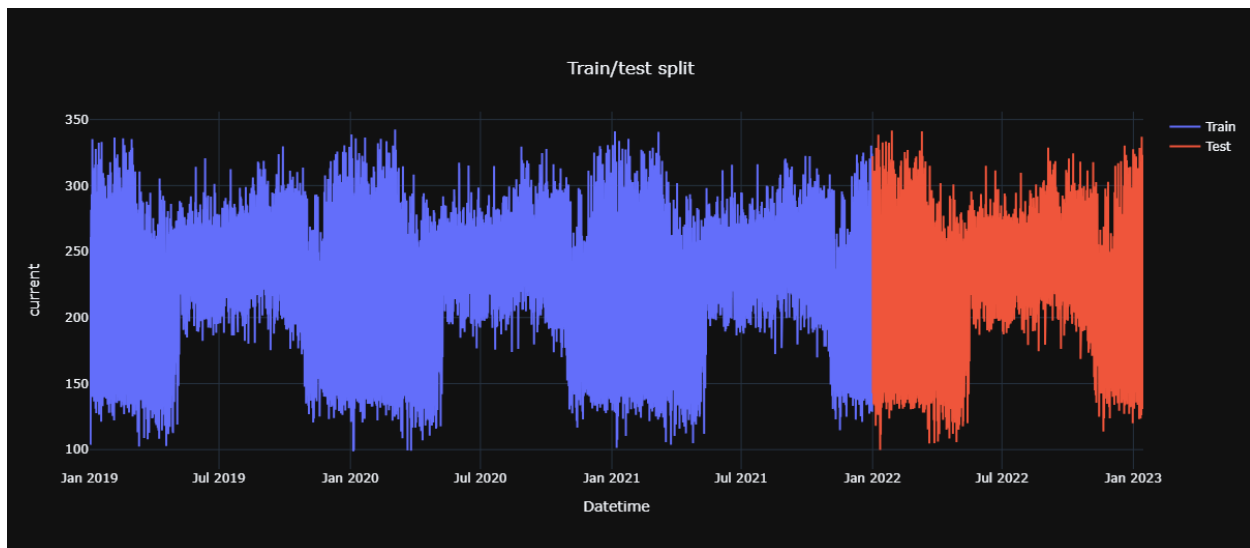


Figure 3.13: 36/12 (months) train and test.

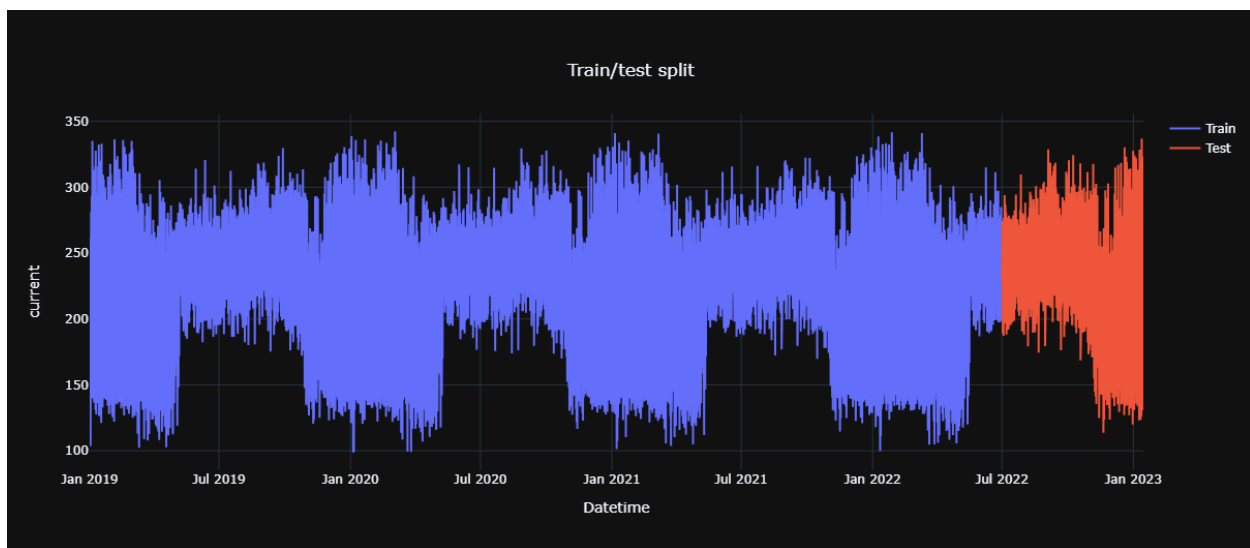


Figure 3.14: 42/6 (months) train and test.

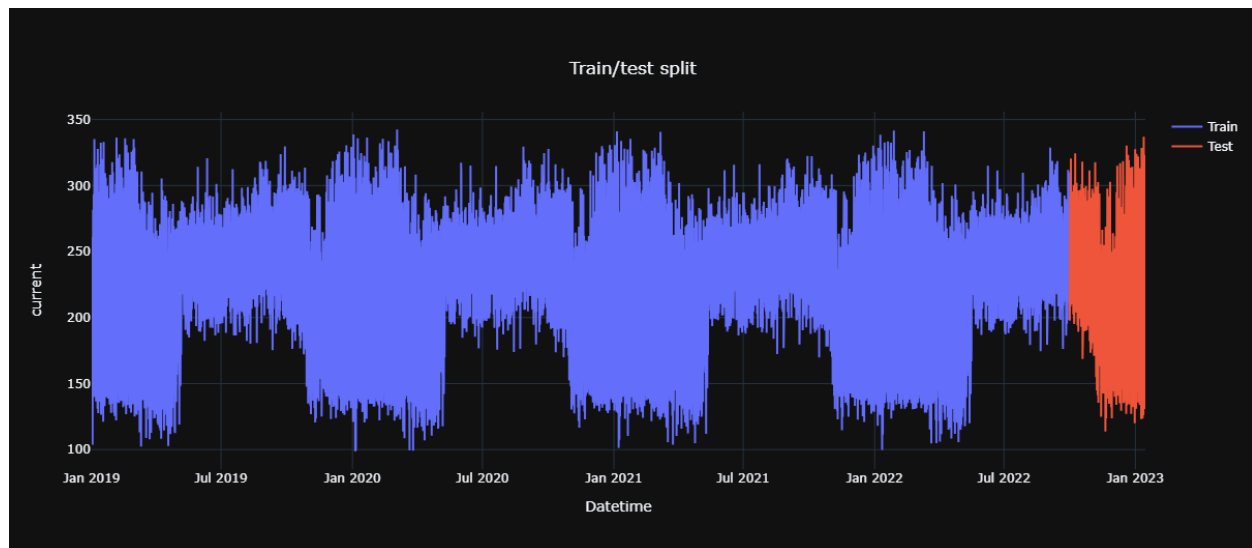


Figure 3.15: 45/3 (months) train and test.

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:

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

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 were trained and tested using Prophet model implementation in python.

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

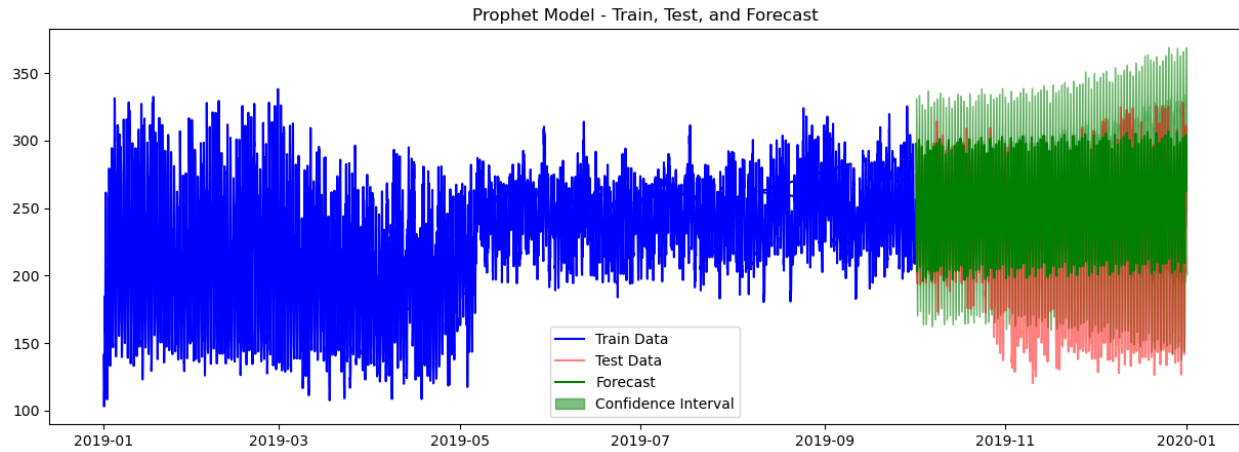


Figure 3.22: 9/3 (months) train/test and forecast using prophet.

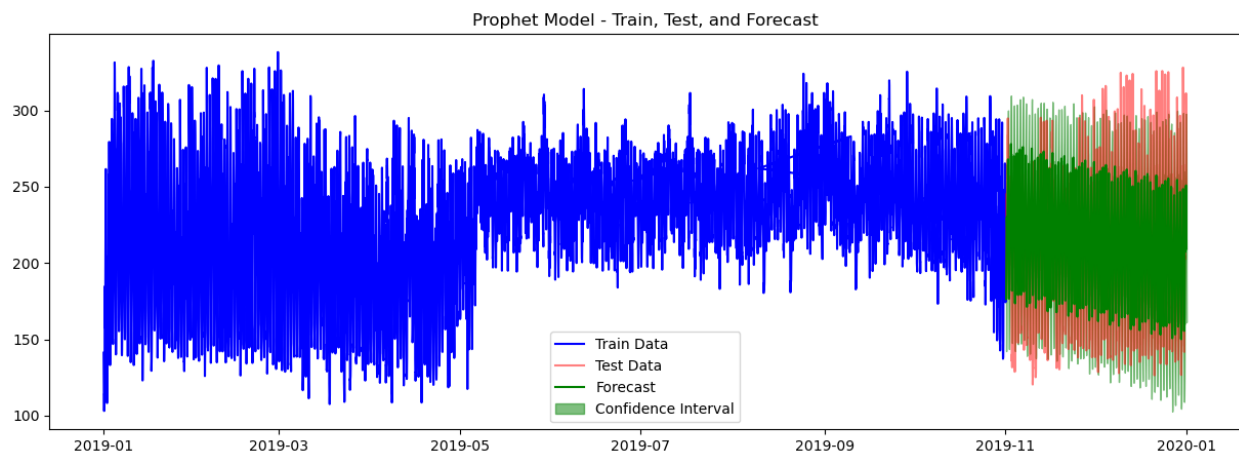


Figure 3.23: 10/2 (months) train/test and forecast using prophet.

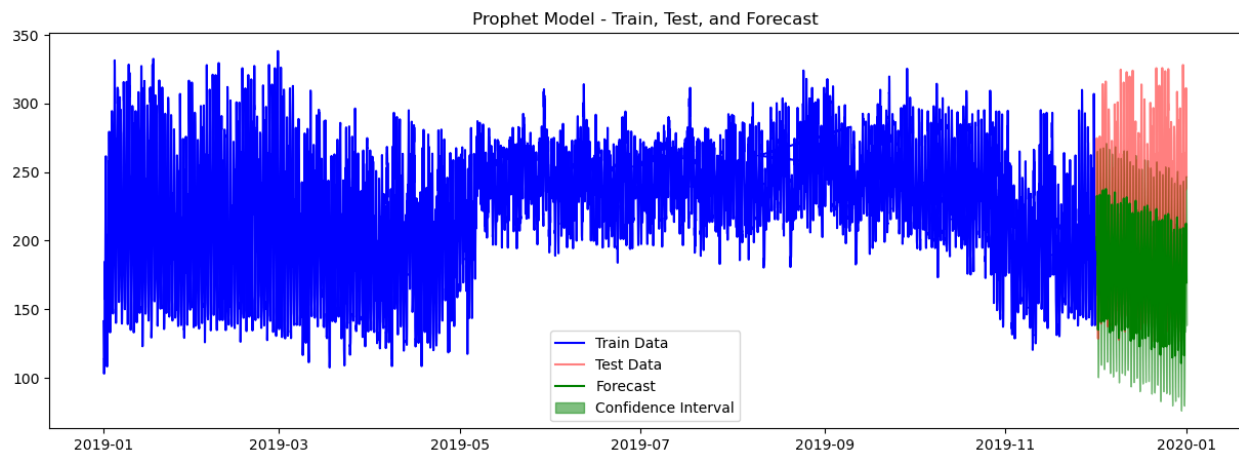


Figure 3.24: 11/1 (months) train/test and forecast using prophet.

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

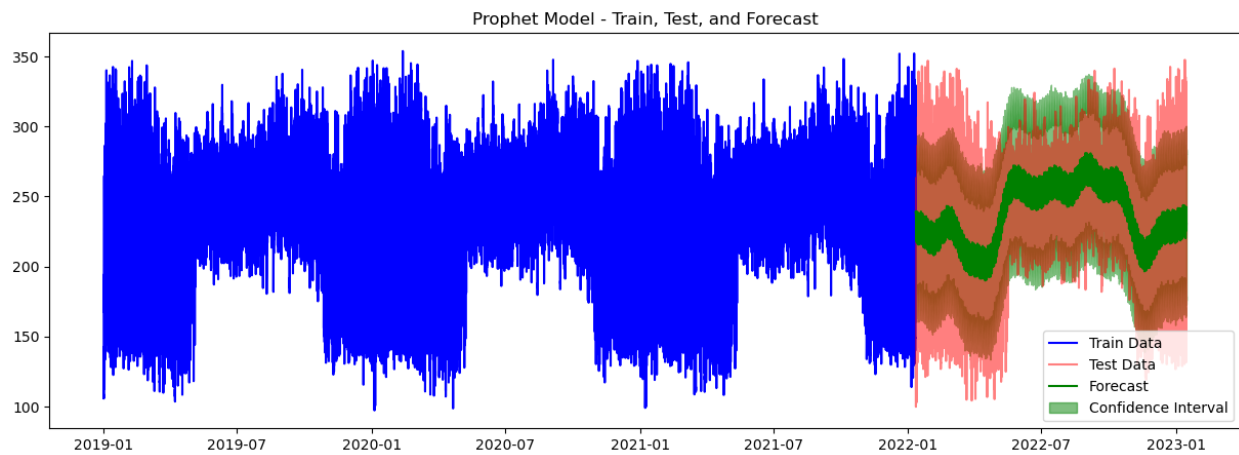


Figure 3.25: 36/12 (months) train/test and forecast using prophet.

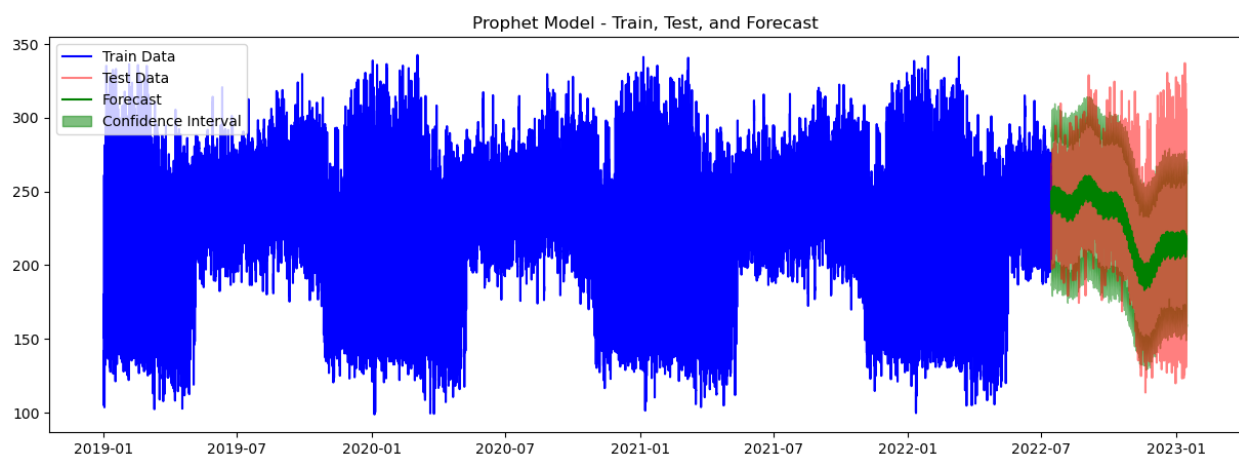


Figure 3.26: 42/6 (months) train/test and forecast using prophet.

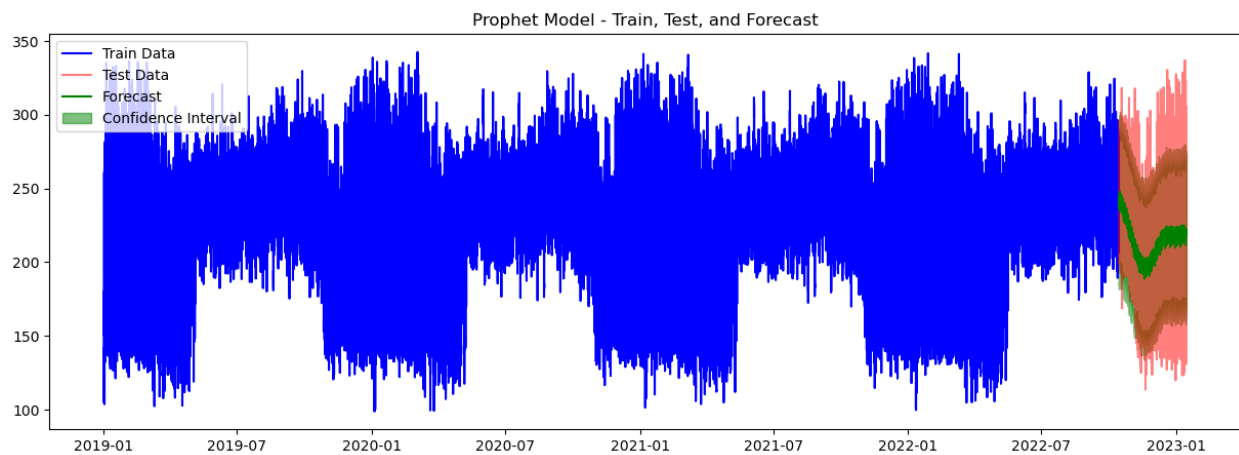


Figure 3.26: 45/3 (months) train/test and forecast using prophet

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:

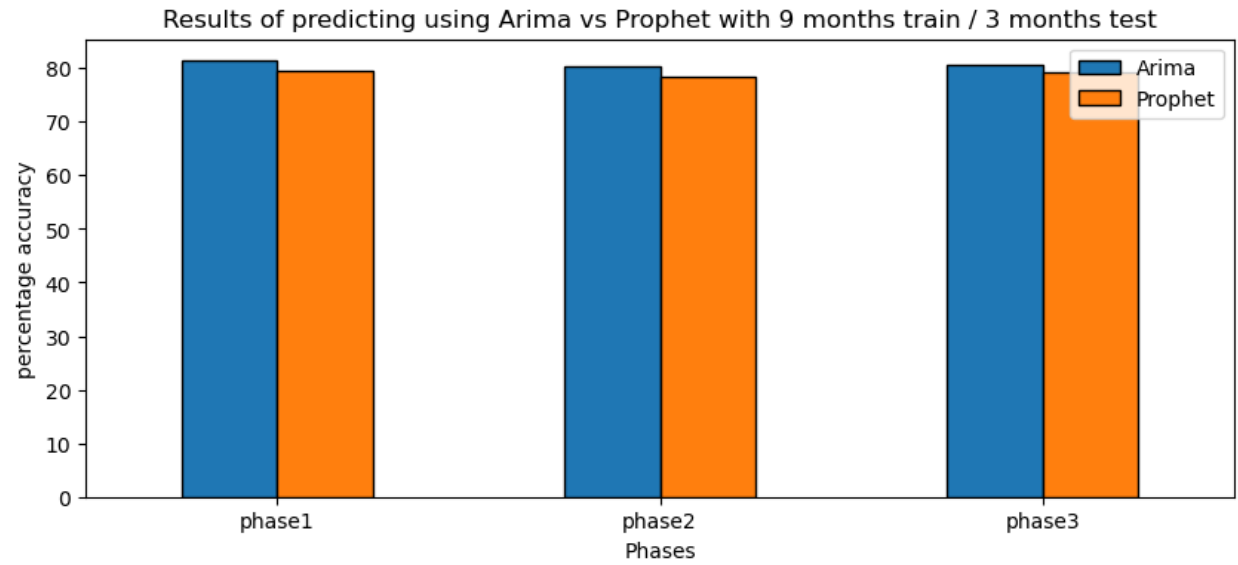


Figure 4.1: ARIMA vs Prophet with 9/3 train and test.

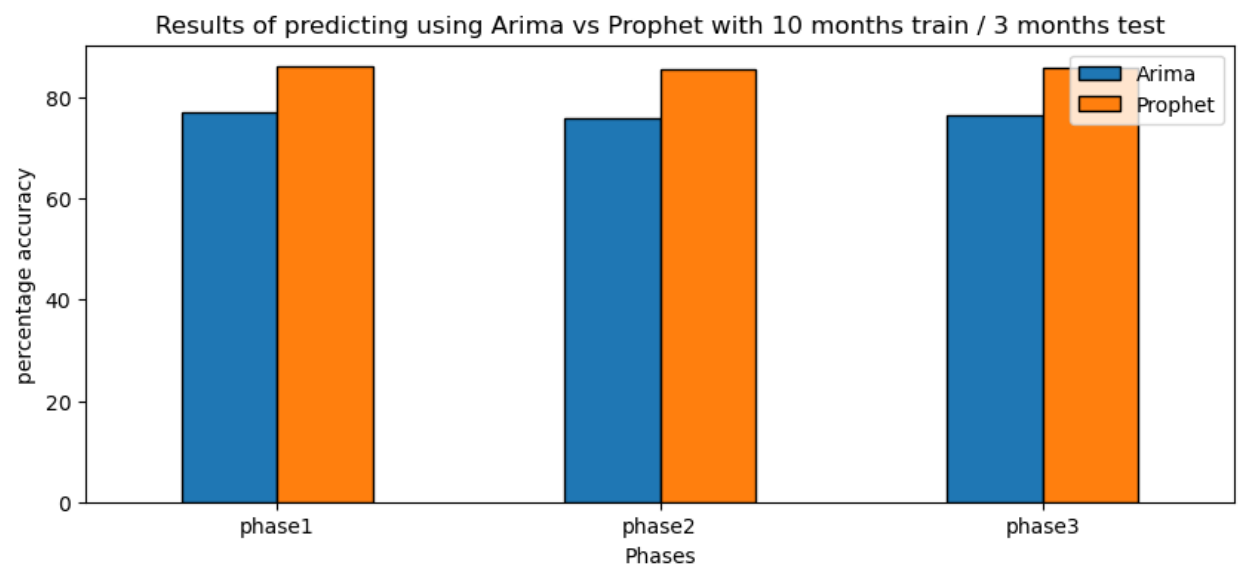


Figure 4.2: ARIMA vs Prophet with 10/2 train and test.

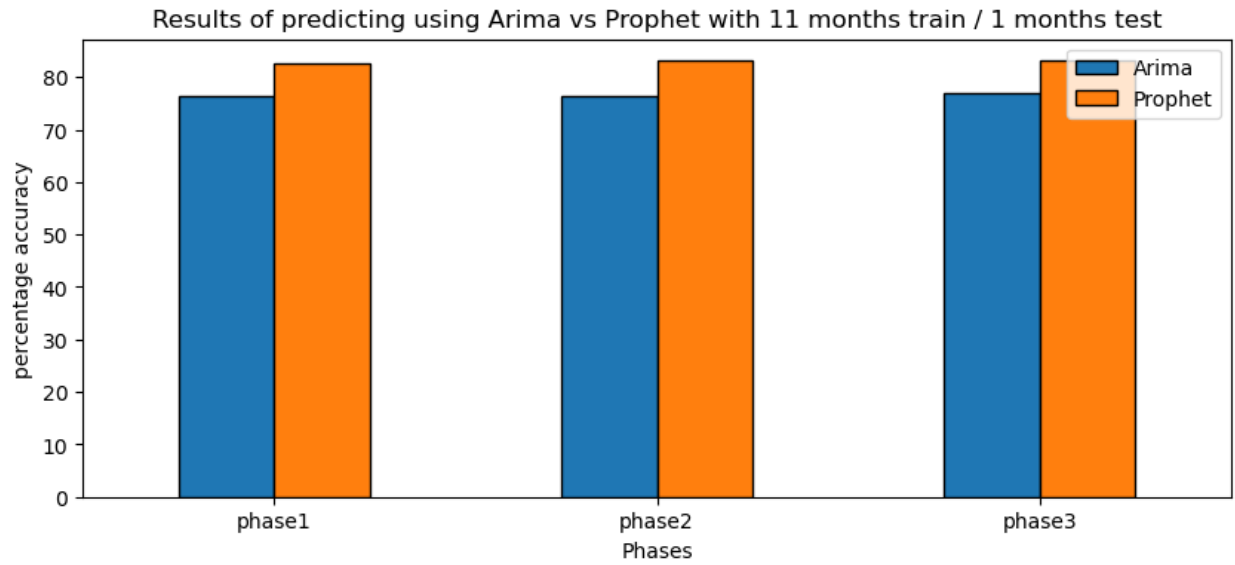


Figure 4.3: ARIMA vs Prophet with 11/1 train and test.

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

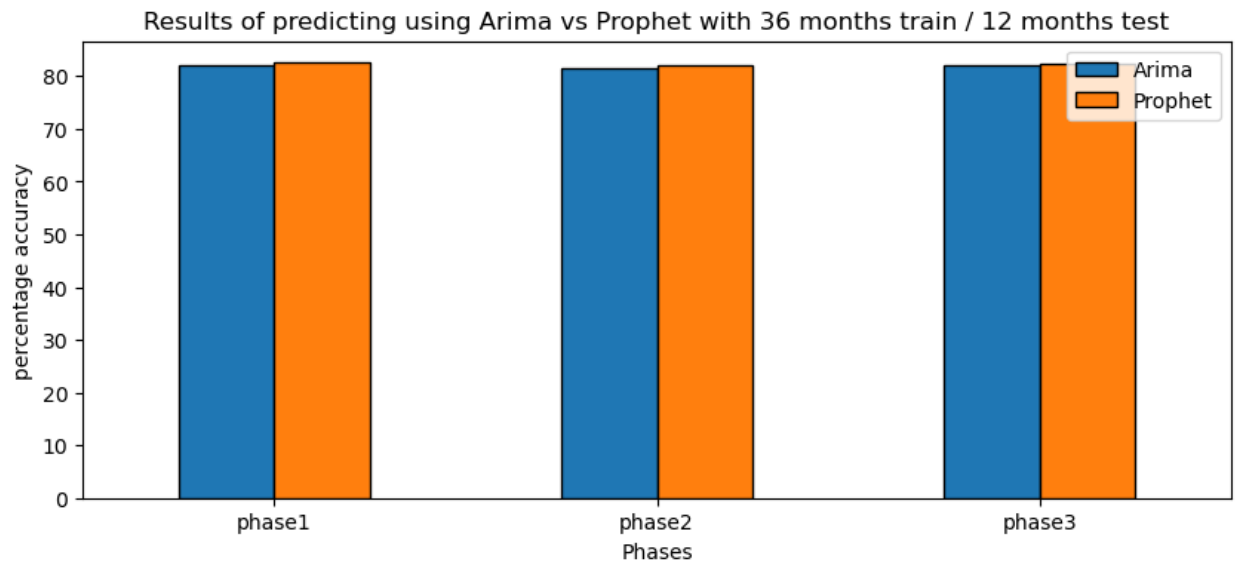


Figure 4.4: ARIMA vs Prophet with 36/12 train and test.

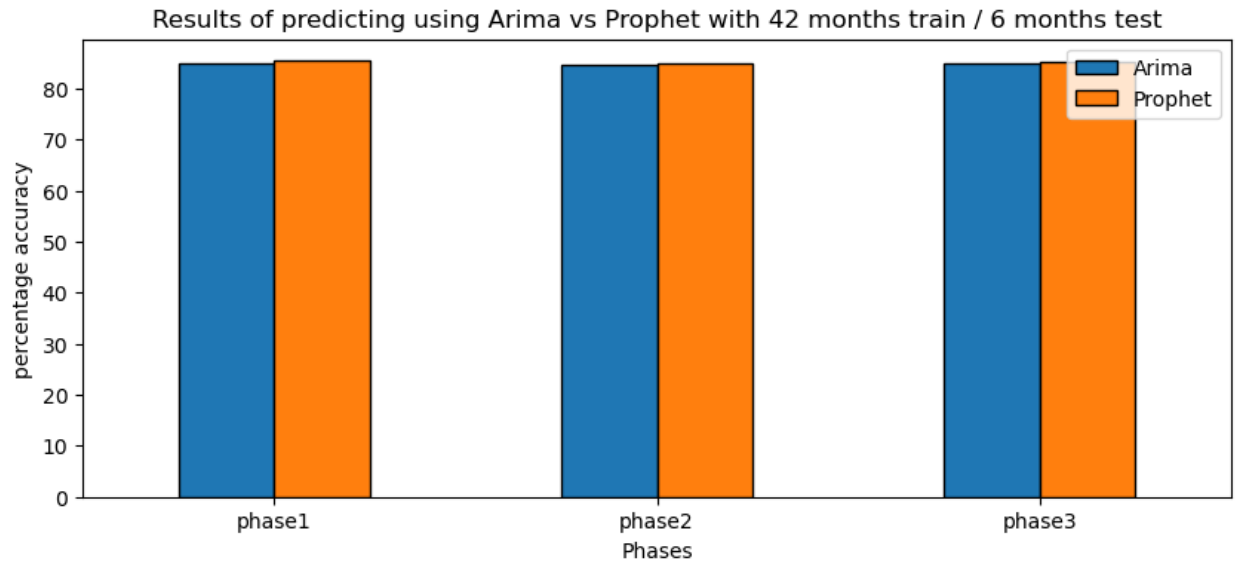


Figure 4.5: ARIMA vs Prophet with 42/6 train and test.

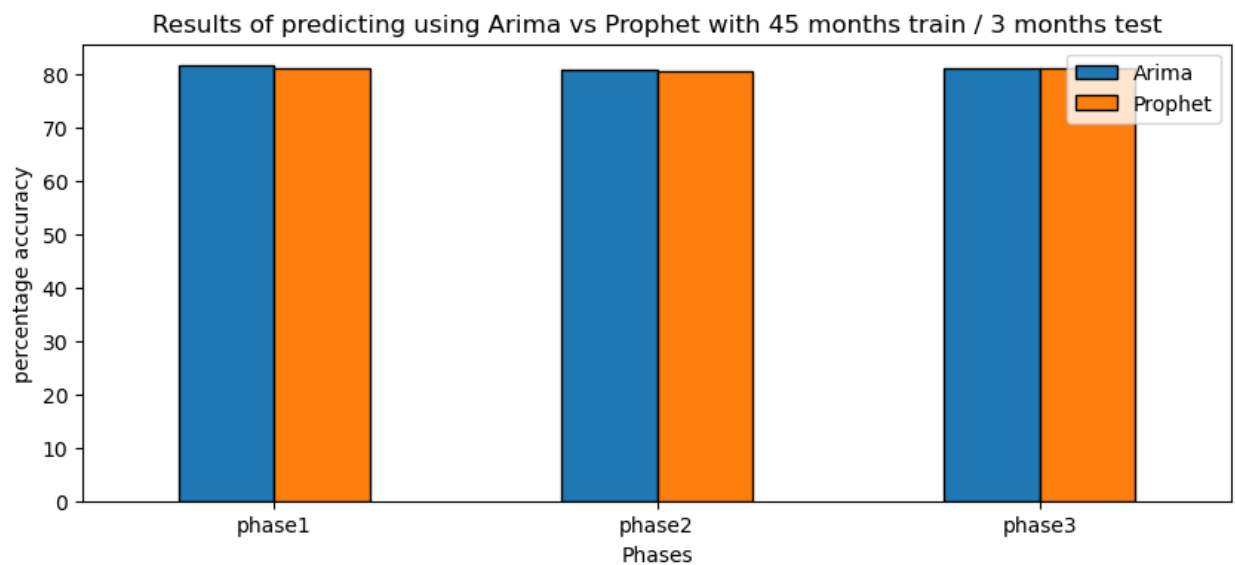


Figure 4.6: ARIMA vs Prophet with 45/3 train and test.

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.

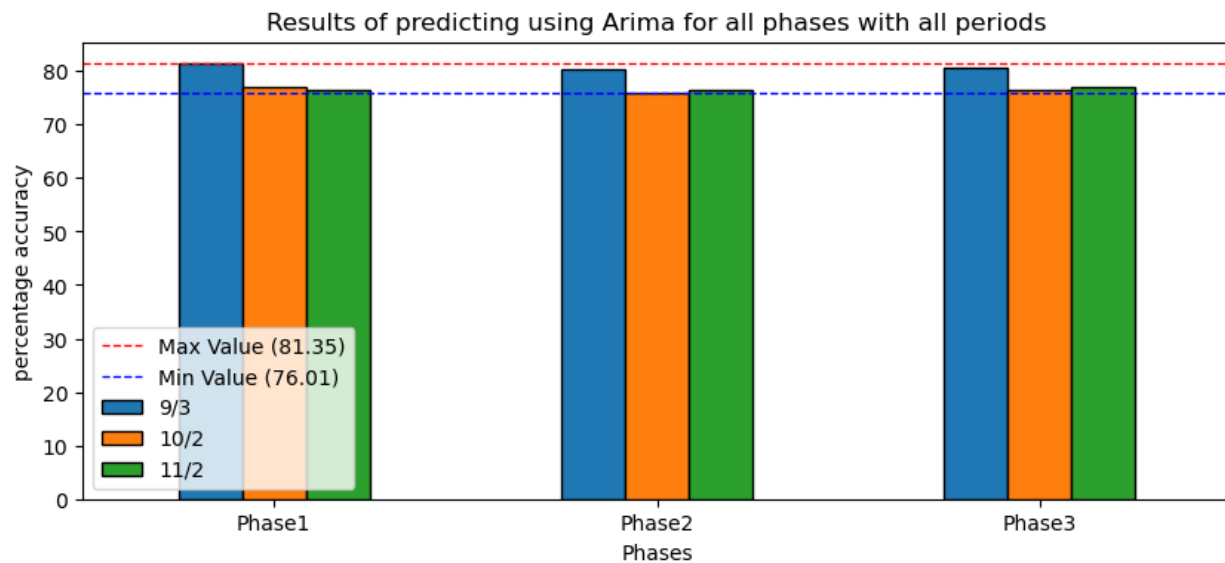


Figure 4.7: ARIMA results with all periods before augmentation.

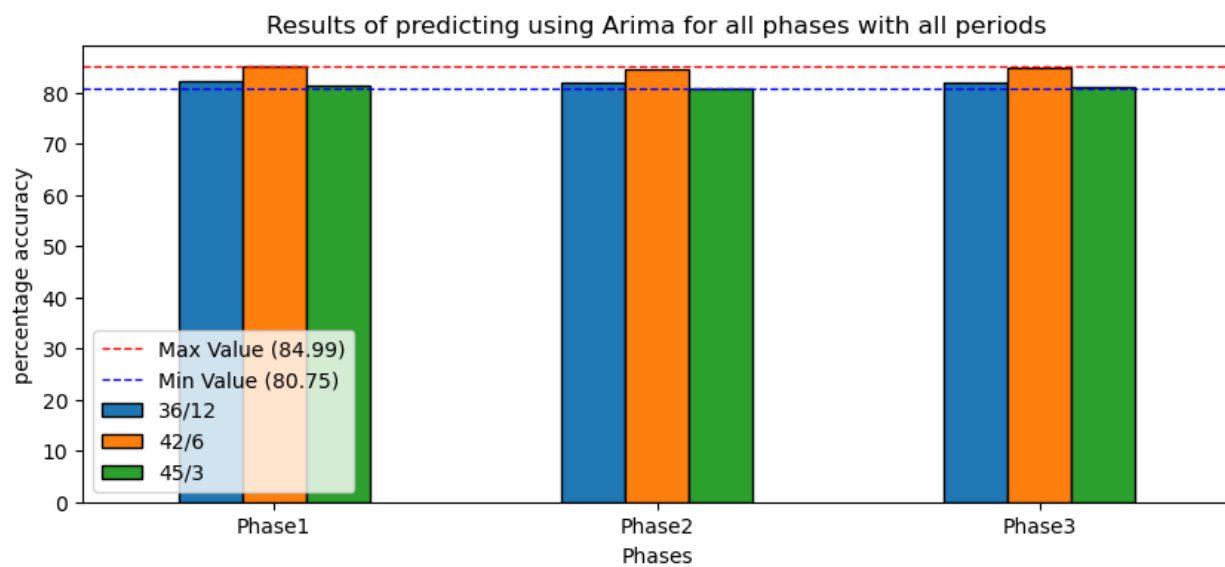


Figure 4.8: ARIMA results with all periods after augmentation.

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.

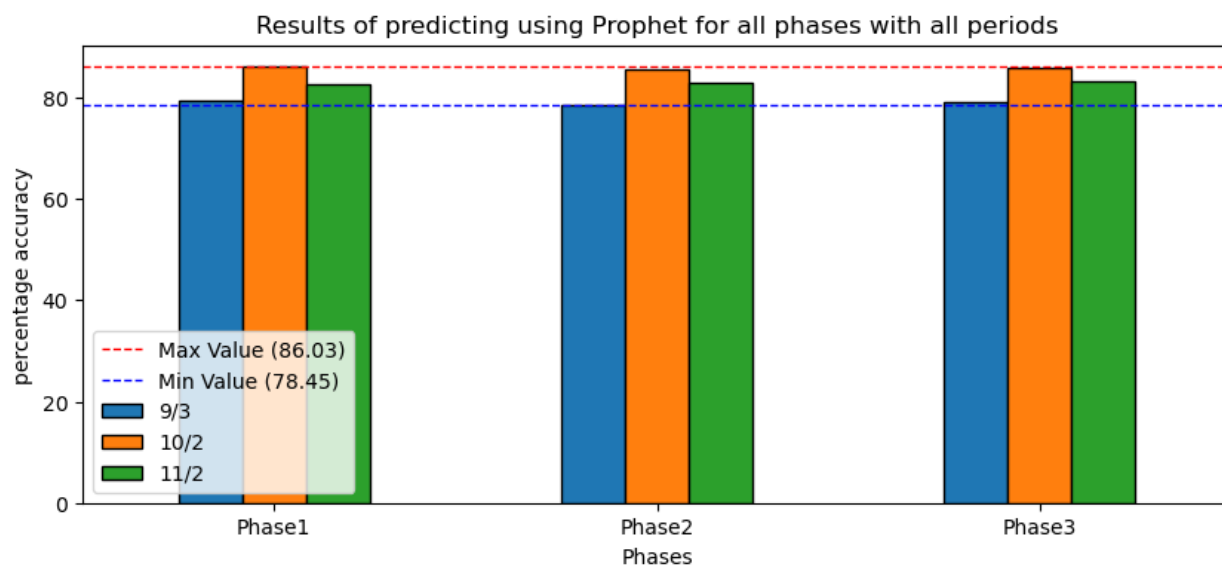


Figure 4.9: Prophet results with all periods before augmentation.

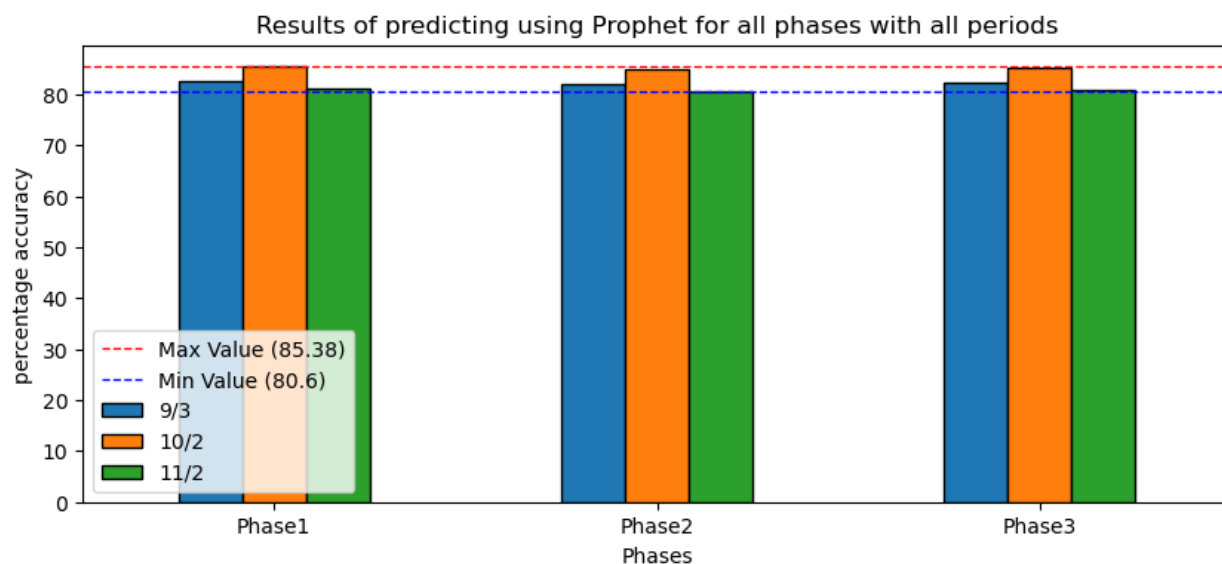


Figure 4.10: Prophet results with all periods after augmentation.

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 0.04 difference between min and max.

Now let's compare periods before and after augmentation.

For ARIMA periods

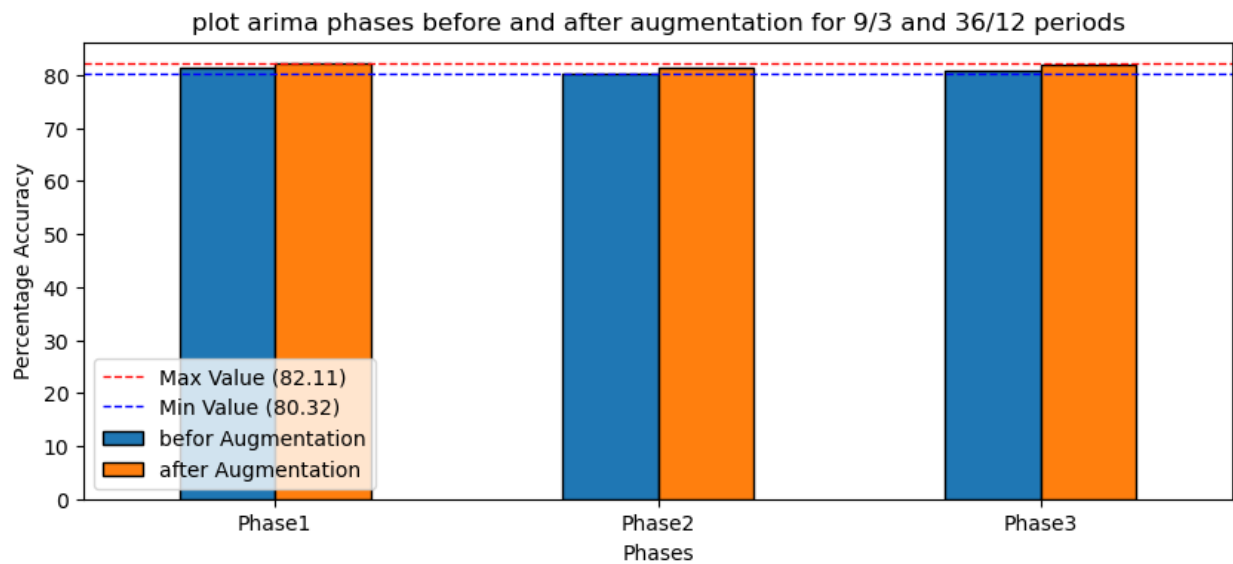


Figure 4.11: ARIMA results with 9/3 and 36/12 periods.

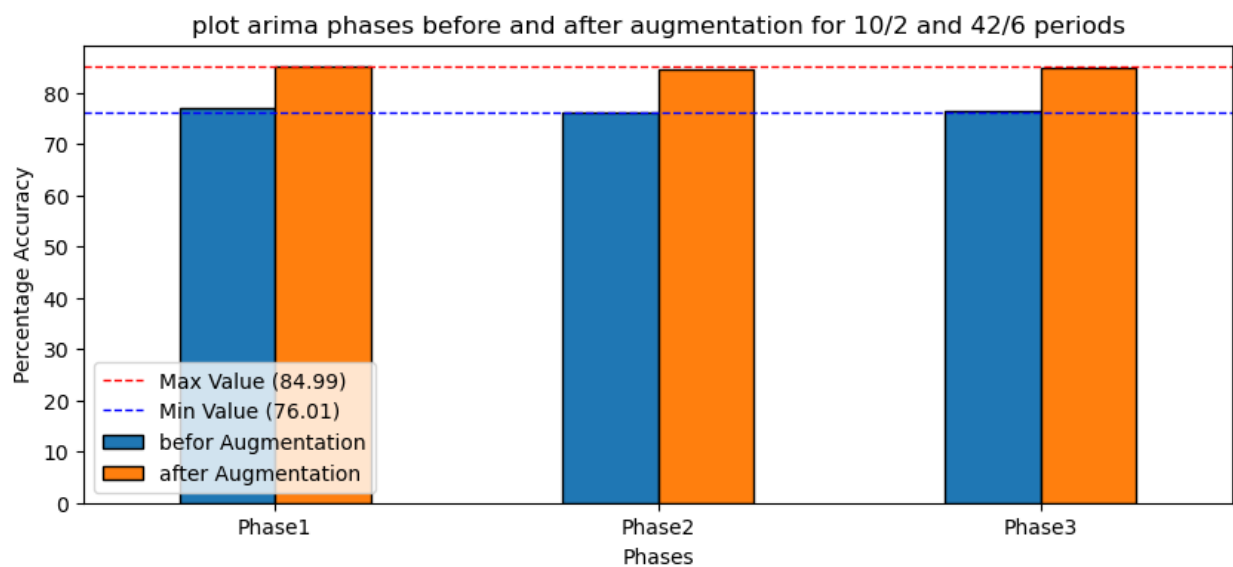


Figure 4.12: ARIMA results with 10/2 and 42/6 periods.

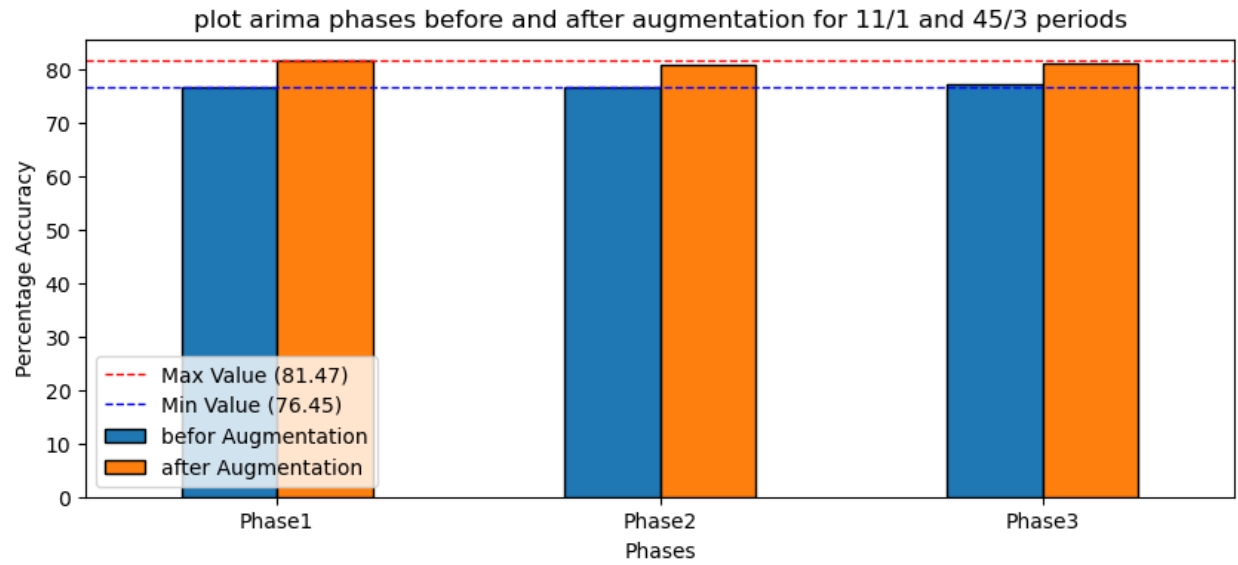


Figure 4.13: ARIMA results with 11/1 and 45/3 periods.

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

For Prophet periods

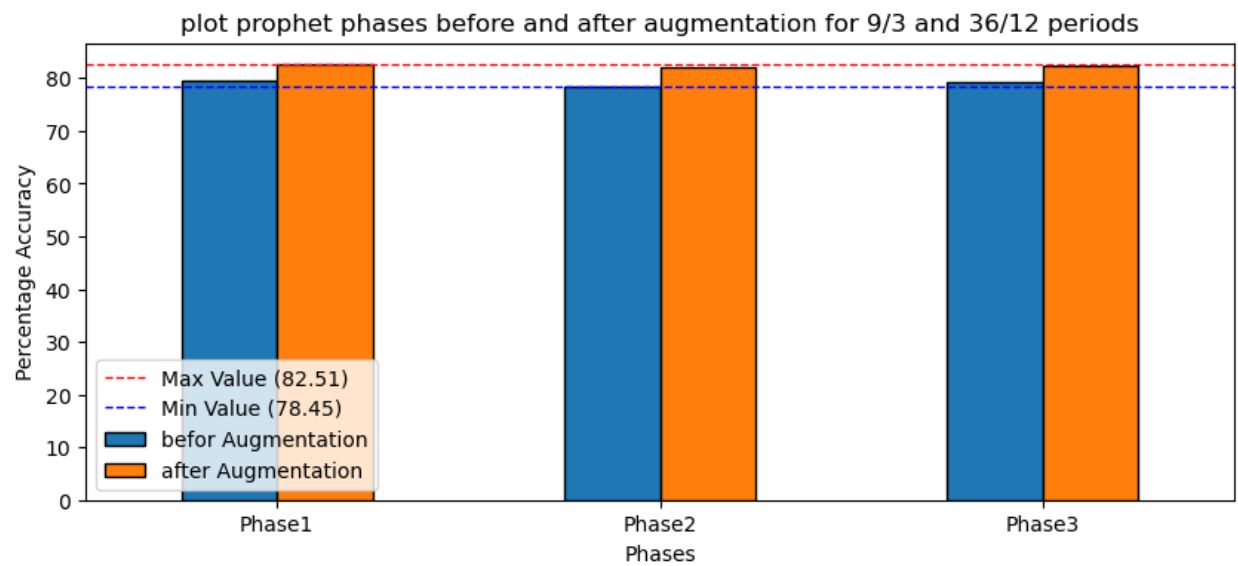


Figure 4.14: Prophet results with 9/3 and 36/12 periods.

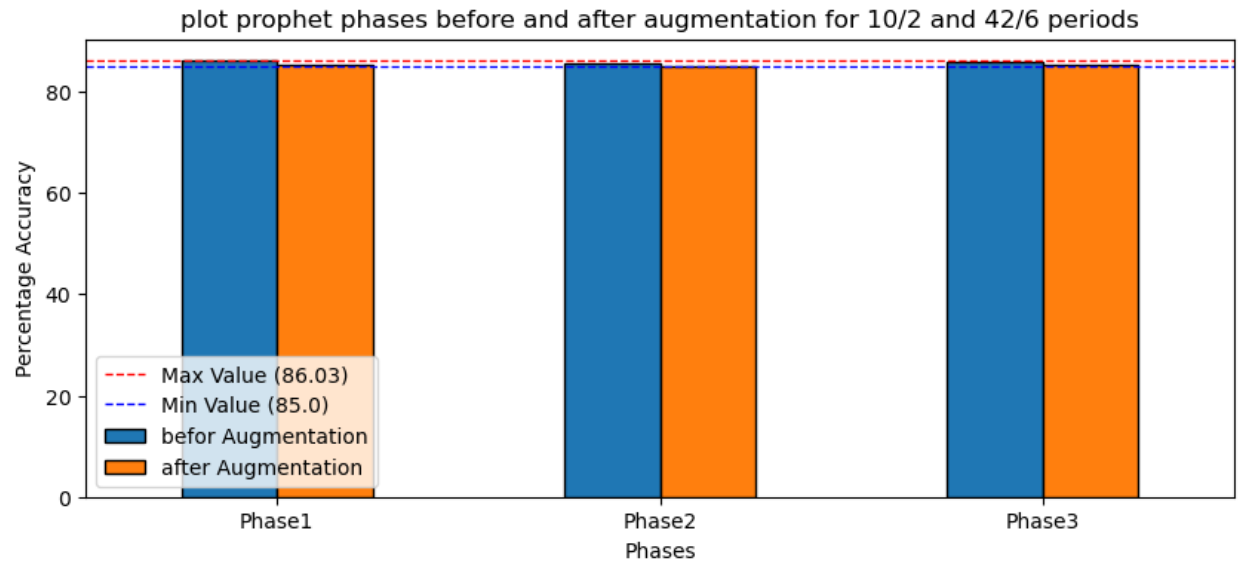


Figure 4.15: Prophet results with 10/2 and 42/6 periods.

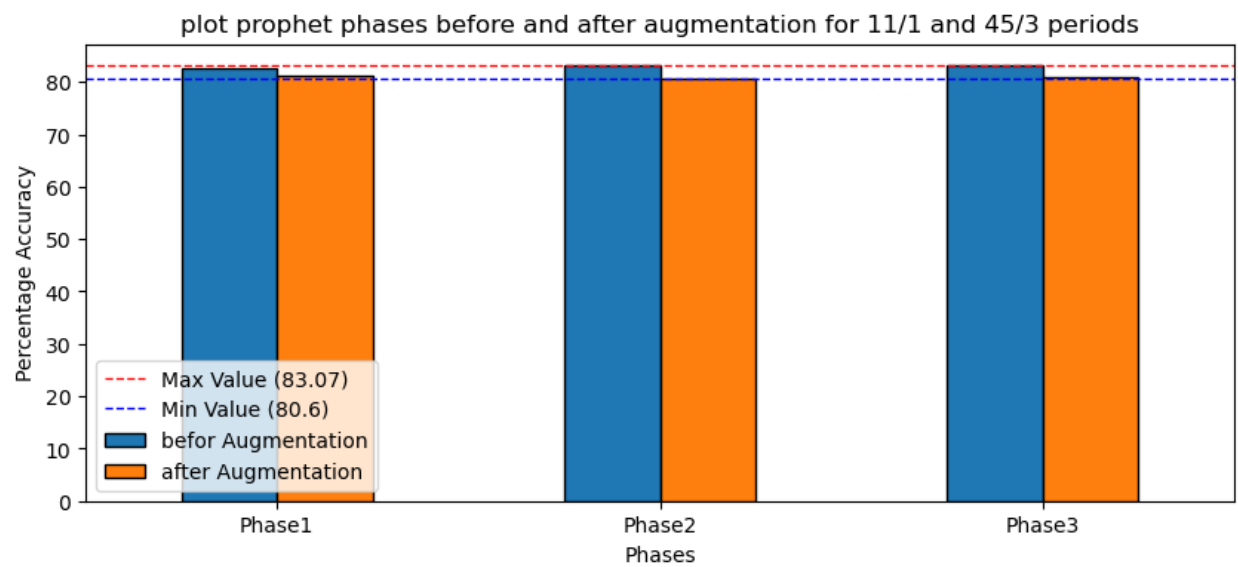


Figure 4.16: Prophet results with 11/1 and 42/3 periods.

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

4.3 conclusion

- The MAPE of Prophet affected with train period increasingly more than ARIMA.
- Clearly the range of the results of ARIMA model where increased after augmentation.
- The 42/6 train and test period had the highest MAPE for both ARIMA and Prophet with augmented data.
- The results for Prophet did not clearly affected with the augmentation.
- 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 Developments

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.

REFERENCES

- 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- Lee Y-S, Tong L-I. Forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming. *Knowl-Based Syst* 2011;24(1):66–72.
- 4- M. Innat, "Preferable tools for machine learning - Python - MatLab - R | Codementor." https://www.codementor.io/@innat_2k14/preferable-tools-for-machine-learning-python-matlab-r-jfozzpphz (accessed Jun. 13, 2023).
- 5- 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.
- 6- Lim B, Arık SÖ, Loeff N, Pfister T. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *Int J Forecast* 2021.
- 7- Chakhchoukh Y, Panciatici P, Mili L. Electric load forecasting based on statistical robust methods. *IEEE Trans Power Syst* 2010;26(3):982–91.
- 8- Pierrot A, Goude Y. Short-term electricity load forecasting with generalized additive models. *Proc ISAP Power* 2011;2011.
- 9- 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.