Summer Training Report

On

GRID FREQUENCY PREDICTION USING MACHINE LEARNING MODELS AND TIME SERIES ANALYSIS

Submitted in partial fulfillment of the requirements for the completion of one month's summer internship/training [ART 355]

Name: **Eshaan Gupta**
Enrollment number: **14519051622**

Under the supervision of

Mr. Anupam Lakhanpal

UNIVERSITY SCHOOL OF AUTOMATION AND ROBOTICS
GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY
EAST DELHI CAMPUS, SURAJMAL VIHAR, DELHI- 110032

RELIANCE

Sasan Power Limited
CIN: U40102MH2006PLC190557

2nd Floor, Reliance Centre
19, Walchand Hirachand
Marg, Ballard Estate
Mumbai 400 001
Tel: +91 22 30327000
Fax: +91 22 30327699
www.reliancepower.co.in

Certificate No: VT-FY-24-25-16

Date: 31.07.2024

CERTIFICATE

(To whom so ever it may concern)

This is to certify that **Mr. Eshaan Gupta**, S/o **Mr. Ajay Kumar Gupta**, student of **B. Tech. in Artificial Intelligence & Machine Learning** from **University School of Automation and Robotics, GGSIPU, EDC, New Delhi** has successfully completed his 30 days industrial training in our organization. He completed his training **since 01.07.2024 to 31.07.2024** at **Sasan Ultra Mega Power Plant (6X660MW)**.

During the above period, we found him sincere and hard working.

"WE WISH HIM ALL THE BEST FOR HIS FUTURE CAREER."

Excellent!
Keep it up...
You are simply "Brilliant"

AVP-L&D
Reliance Power
Sasan

Training Head/In-charge

SASAN POWER LIMITED
M.P.

Declaration

I hereby declare that the Summer Training Report entitled "Grid Frequency Prediction using Machine Learning Models and Time Series Analysis" is an authentic record of work completed as requirements of Summer Training (ART 355) during the period from **1st July 2024** to **31st July 2024** in University School of Automation and Robotics/CDAC/NIC/DRDO/PEC/etc under the supervision of **Mr. Nishant Kumar Gupta, Additional Vice President (L&D), Reliance Power, Sasan.**

EshaanGupta

ESHAAN GUPTA
(14519051622)
(Signature of student)
(Name of Student)
(Enrollment Number)

Date: 19-09-2024

Additional Vice President (L&D) Reliance Power Sasan

(Signature of Supervisor)

MR. NISHANT KUMAR GUPTA
(Name of Supervisor)

Date: 19-09-2024

(Anupam Lalchandani)
(Mentor)

Acknowledgement

I would like to express my deepest gratitude to all those who contributed to the successful completion of my internship and the preparation of this report.

Firstly, I am extremely grateful to **Sasan Thermal Power Plant, Reliance Power Limited** for providing me with this wonderful opportunity to intern at their esteemed organization. I would like to thank **Mr. Nishant Kumar Gupta**, my internship mentor, for his invaluable guidance, constant encouragement, and constructive feedback throughout the duration of my internship. His expert knowledge and patience have been instrumental in enhancing my understanding of the field.

I would also like to extend my heartfelt thanks to **Devendra Pratap Singh** for his support and camaraderie, which made my experience both educational and enjoyable.

My sincere thanks to the **University School of Automation and Robotics, GGSIPU, EDC**, for providing the framework that made this internship possible. The academic support and resources from the institution have significantly contributed to my professional growth and learning.

Lastly, I am grateful to my family and friends for their unwavering support and encouragement throughout this journey.

This internship experience has been a rewarding and enlightening chapter of my academic life, and I am deeply thankful to everyone who made it possible.

Eshaan Gupta

About Company

I completed my summer internship/vocational training at the **Sasan Thermal Power Plant**, which operates under **Reliance Power Limited**. My internship began on **1st July 2024** and concluded on **31st July 2024**. The Sasan Power Project is situated on the land of the villages – Sidhi Khurd, Tiyara and Harharwa, in the Waidhan tehsil of Singrauli district of Madhya Pradesh state which is often referred to as the "Power House of India" due to the concentration of thermal and hydroelectric power plants in the region.

The Sasan Ultra Mega Power Project (UMPP) is the world's largest integrated power generation and coal mining venture and serves as Reliance Power's flagship project. It comprises six units, each with a generation capacity of **660 MW**, making the total capacity **3,960 MW**. This project is designed to supply reliable and low-cost electricity to over **42 crore** people across seven Indian states for the next **25 years**.

Sasan is located approximately **900 kilo meters from Delhi**. Due to the remote location and the presence of numerous power plants in the area, connectivity to the region is limited. Only one train travels to Sasan weekly. The nearest major city is **Varanasi**, which is around **200 kilo meters away**. Despite the logistical challenges, the area boasts a low population density, resulting in minimal noise pollution and offering stunning scenic views of the natural landscape.

This unique combination of industrial power and natural beauty provided a fascinating backdrop to my internship experience, where I gained insights into large-scale energy production and management.

Fig 1: Central Control Room of the power plant

Fig 2: Units at Sasan Thermal Power Plant

Fig 3: Geographical location of Sasan Thermal Power Plant

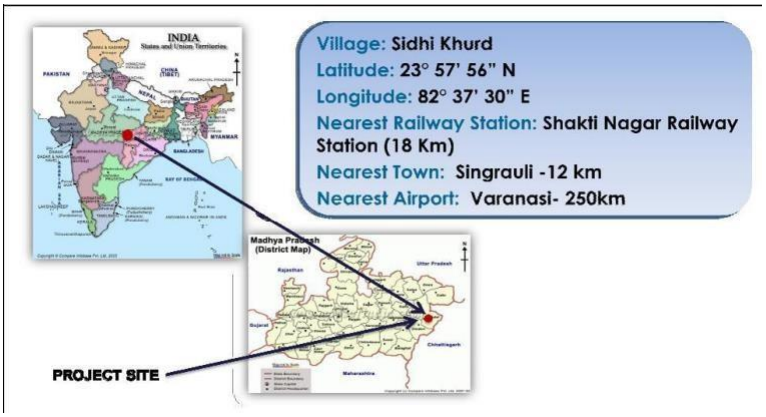

Fig 4: Location of Sasan

Table of Contents

List of Figures

Abbreviation and Nomenclature

Abbreviation

AR	Auto Regressive
MA	Moving Average
ARMA	Auto Regressive Moving Average
ARIMA	Auto Regressive Integrated Moving Average
SARIMA	Seasonal Auto Regressive Integrated Moving Average
ACF	Auto Correlation Function
PACF	Partial Auto Correlation Function
NTPC	National Thermal Power Corporation
AC	Alternating Current
ADF	Augmented Dickey Fuller
SVM	Support Vector Machine
KPSS	Kwiatkowski–Phillips–Schmidt–Shin
AIC	Akaike Information Criterion
BIC	Bayesian Information Criterion
HQIC	Hannen-Quinn Information Criterion

Chapter 1

Abstract

I undertook my summer internship at the **Sasan Thermal Power Plant**, a coal-based power generation facility under **Reliance Power Limited**, which plays a critical role in India's energy production. The plant operates using steam generated by burning coal, a process that has remained a dominant method for electricity generation despite significant technological advancements in recent centuries. Globally, coal continues to be a major energy source, contributing to around 60% of total power generation.

My internship project focused on the prediction of **grid frequency** using **Machine Learning models**, with the goal of analysing and forecasting frequency variations based on historical data. The dataset provided for this project spanned three years, consisting of grid frequencies, blocks, and dates. Each day was divided into 96 blocks of 15-minute intervals, leading to an extensive dataset of approximately **116,800 rows**. Grid frequency, the rate at which power is transmitted through the grid, ideally remains at **50 Hz** in India, although it can fluctuate based on power demand and supply imbalances. In some other countries, this frequency reaches up to 60 Hz.

In addition to working on the data-driven aspect of the project, I had the opportunity to visit the coal mines linked to the thermal power plant. This experience offered me a unique perspective on the entire energy production process, from raw material extraction to electricity generation. I observed the control centre responsible for managing coal mining operations and witnessed how **artificial intelligence** and **machine learning** technologies could potentially be integrated to enhance the efficiency and overall performance of such large-scale industrial systems. This experience provided me with valuable insights into the role of engineering and modern technologies in optimizing industrial processes.

The primary task in my project was to analyse the dataset, identify patterns in the historical grid frequencies, and utilize these patterns to build a machine learning model capable of predicting future grid frequencies. This was my first experience working with such a vast dataset, which required me to perform several crucial steps, including **data cleaning**, **data extraction**, and time series analysis. To gain a deeper understanding of these techniques, I sought out learning resources through YouTube videos, e-books, and online courses on Udemy. Equipped with this knowledge, I applied various machine learning algorithms to the data to forecast grid frequency.

Upon further discussions with my supervisor, it became clear that while predicting grid frequency was valuable, the power plant could not immediately act on these predictions. Power plant operations, particularly turbine management, are sensitive processes, and abrupt shutdowns or restarts based on frequency fluctuations could cause damage to the hardware, which includes multi-crore turbine equipment. As a result, while the model's predictions were accurate, they were only useful for precautionary measures rather than real-time operational decisions.

Despite this limitation, I do not consider the project a failure. Instead, I view it as a significant step forward in my learning journey. Through this experience, I developed essential skills in **time series analysis**, machine learning model development, and data management. I also gained a deeper appreciation of the engineering challenges involved in managing large-scale industrial operations, where even a small error could lead to major consequences. Additionally, I learned the importance of maintaining strong work ethics and professional responsibility, which are crucial for personal and professional growth.

In conclusion, this internship was an enriching experience that not only enhanced my technical knowledge but also provided me with insights into the practical application of AI and machine learning in industrial settings. It reinforced my understanding of how engineering can make a significant impact on a national scale and how important it is to approach such work with diligence and precision.

Chapter 2

Introduction

Sasan Thermal Power Plant is a critical power generation facility and one of the pioneers of thermal power production in India. The plant comprises six units, each generating 660 MW of electricity. While the basic concept behind the operation of the plant is straightforward—using coal to produce steam for power generation—the execution is highly engineered and meticulously monitored by professional engineers. The plant is in an area with several other power plants, including an NTPC facility with three units. Coal transportation from the mining site to the power plant is carried out through a 15-kilometer-long conveyor belt, ensuring continuous and efficient coal supply without delays, which is essential for maintaining operational efficiency.

Fig 5: Mr. Nishant Kumar Gupta (left)

2.1 Safety Induction

During my initial days at the power plant, I underwent a safety induction where I was briefed on the dos and don'ts to ensure personal safety and the safety of others. I was made fully aware of the safety protocols and informed about designated safe locations to reach in case of an emergency. Safety is of paramount importance in the power plant environment, and any carelessness can lead to severe injury or even fatal consequences. As a mandatory safety measure, no one was permitted to enter the plant without wearing safety shoes and helmets. Each person working at the plant was issued a helmet and a pair of safety shoes, and failure to comply with these rules could result in serious disciplinary action.

2.2 About the project

For my project, I was assigned the task of predicting grid frequency using machine learning models. Grid frequency refers to the rate at which the alternating current (AC) in the electrical grid oscillates and serves as a key indicator of the balance between electricity supply and demand. In India, the ideal grid frequency is 50 Hz, meaning the current reverses direction 50 times per second. In some other countries, such as the United States, the standard grid frequency is 60 Hz. When there is an imbalance between power generation and consumption, the grid frequency fluctuates. For example, when power generation exceeds demand, the frequency increases, and when demand exceeds supply, the frequency decreases. Maintaining a stable grid frequency is essential for the safe and efficient operation of electrical equipment, as deviations from the ideal frequency can lead to equipment damage, reduced performance, and, in extreme cases, blackouts. Therefore, power utilities continuously monitor and adjust electricity generation to keep the frequency within an acceptable range.

The dataset provided for my project consisted of approximately 116,800 rows and 3 columns, which contained data on grid frequency over time. This was the first time I worked with such a large dataset. To develop the machine learning model, the first step was to clean and preprocess the data. I used Jupyter Notebook for this purpose. Jupyter Notebook is an open-source, web-based interactive environment that allows users to create and share documents containing live code, equations, visualizations, and narrative text. It supports a variety of programming languages, but Python is the most used. Jupyter Notebook is widely used in data science, machine learning, academic research, and education due to its flexibility, ease of use, and ability to seamlessly integrate code with documentation and visualizations.

2.3 Time Series Analysis

Time series analysis is a statistical technique used to analyse sequences of data points, typically collected, or recorded at successive points in time. It involves understanding the underlying structure and patterns in the data to make predictions or gain insights into future behaviour. Time series data often exhibits trends, seasonal patterns, and cyclical fluctuations, distinguishing it from other types of data. Common applications include stock market predictions, weather forecasting, economic modelling, and demand forecasting in energy or utilities. Techniques like ARIMA (Auto Regressive Integrated Moving Average) and SARIMA (Seasonal ARIMA) are often used to model and predict future values based on historical data. The goal of time series analysis is to uncover patterns that can be used for making accurate forecasts and informed decisions.

The dataset I worked with was structured into 96 blocks, with each block representing a 15-minute interval throughout the day. This level of granularity allowed for a detailed look at how grid frequency fluctuates over time, offering insights into both short-term variations and longer-term trends. Time series analysis was applied to the data due to the presence of clear seasonal patterns. These patterns likely arose from variations in electricity demand and supply, which can be influenced by factors such as time of day, weather conditions, and seasonal energy usage trends.

For instance, during certain times of the year—such as summer—electricity demand may increase significantly due to widespread use of air conditioning, while demand might decrease during cooler months. Additionally, daily cycles of high demand during peak hours (such as mornings and evenings) and lower demand during off-peak hours were evident in the dataset. These repeating seasonal and cyclical patterns made time series analysis the appropriate method for understanding and predicting future grid frequencies. By analysing the data over time, we could capture these trends and incorporate them into the model, allowing for more accurate predictions of grid frequency behaviour.

Since time series analysis was a completely new topic for me, I took the initiative to complete a course on Udemy to gain valuable insights and practical knowledge. The course helped me understand the foundational concepts and techniques necessary to work with time series data effectively. In addition to Udemy, I also explored numerous YouTube videos, research papers, and books to further expand my understanding. These resources provided me with a comprehensive view of the subject, enabling me to apply the concepts and methodologies

accurately in my project. This self-driven learning approach allowed me to better tackle the complexities of time series analysis and successfully implement it in my work.

2.4 Seminar on Time Series Analysis

During my internship at the Sasan Thermal Power Plant, my supervisor, Mr. Nishant Kumar Gupta, asked me to present a seminar on time series analysis to the other interns. The purpose of the seminar was to explain my project, the challenges I faced, and the approach I was taking to solve the problem. I started by introducing the problem statement—predicting grid frequency using machine learning models—and explained how I decided to approach this issue.

In the seminar, I provided an overview of time series analysis, focusing on the key concepts and models used in the field. I discussed the basics of ARIMA (Auto Regressive Integrated Moving Average), AR (Auto Regressive), MA (Moving Average), and ARMA (Auto Regressive Moving Average) models, which are essential for modelling time-dependent data. Additionally, I introduced concepts like ACF (Auto Correlation Function) and PACF (Partial Auto Correlation Function), explaining how they help in understanding the relationship between data points over time. The seminar was a great opportunity for me to share my learning and provide insights into how time series analysis can be applied to solve real-world problems in power generation.

2.5 Visit to Coal Mines

During my internship, I had the privilege of visiting one of the most fundamental and crucial areas of the Sasan Thermal Power Plant—the coal mines. Located approximately 20 kilo meters from the power plant, the coal is transported via a conveyor belt to ensure a seamless and efficient supply. At the mining site, I had the opportunity to meet the director in charge, who provided valuable insights into the mining operations, emphasizing their focus on quality and efficiency rather than just quantity. Their primary goal is to extract coal in the most efficient manner possible to maintain high energy production levels.

The director also took me on a tour of the control centre, where I observed how a dedicated team of engineers monitors and controls the entire mining process. They use Power BI, an application that tracks real-time data and displays it through visual aids like graphs and charts. The team keeps a constant check on crucial metrics, such as the amount of coal mined and the specific areas being worked on, to ensure that everything runs smoothly.

Interestingly, the director shared one of the challenges they were facing—diesel theft. To combat this issue, RFID tags were installed on each dumper. These tags provide real-time information such as location, fuel usage, remaining diesel, mileage, and the driver's identity. If any discrepancies arise, the team can immediately reach out to the specific dumper from their control room, ensuring transparency and operational efficiency.

One of the ongoing challenges at the mines is optimizing the use of solar-powered batteries. The team is looking for a device capable of predicting cloud cover so that they can adjust the power load on the batteries accordingly. I was asked to collaborate with other interns on this project, exploring potential solutions.

Overall, the visit provided a deep insight into how the plant prioritizes the extraction of high-quality coal. Low-grade coal results in reduced power generation, which can lead to significant losses for the company. Their approach of focusing on quality over quantity was evident in every aspect of their operations, from mining practices to the technology used to ensure efficiency.

Fig 6: Layer of coal deep inside the Earth

Fig 7: Mobile Signal Tower Battery

Fig 8: RFID Tag storing information about battery health

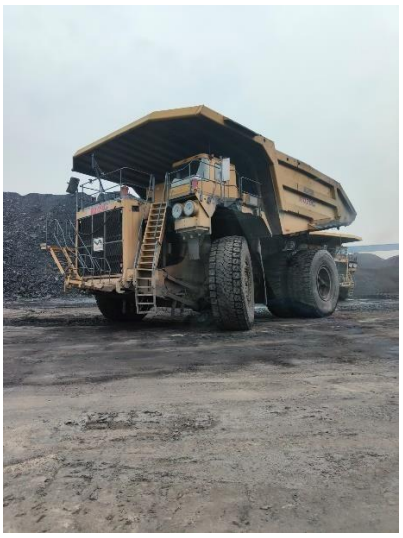

Fig 9: Dumper at the mining site

Fig 10: Our team that visited coal mining site

Fig 11: Visit at the coal mine

2.6 Coal transportation and fly ash deposition

After my visit to the coal mines, I was asked to explore how coal is transported and stored from the mines to the furnace. The system is impressively designed, utilizing three distinct routes to transfer coal. Each unit is supported by seven storage silos, where a specific level in the bunkers is meticulously always maintained. Once the coal reaches the plant, it is finely ground and then passed into the furnace to ensure maximum efficiency and utilization.

Fig 12: Demonstration of how coal is transported

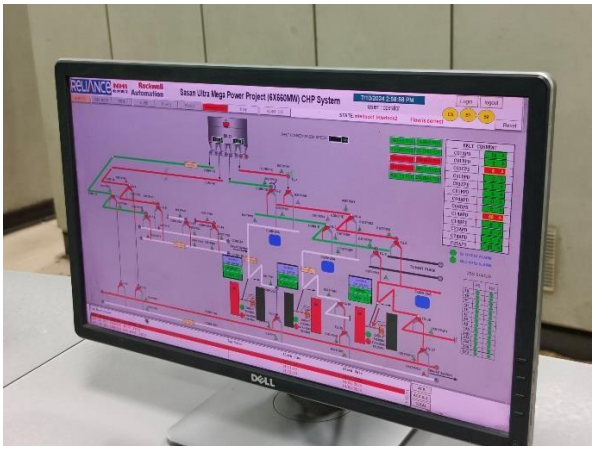

Fig 13: Control system used to monitor coal level

Fig 14: Silos where coal is stored beside the furnace

After the coal is burned, the remaining material is processed for disposal. About 40% of the byproduct is fly ash, while the rest is regular ash. These ashes are mixed with water to create a slurry, which is then sold to produce fly ash bricks. These bricks offer several advantages—they are not only stronger and more durable but also more affordable and lightweight compared to conventional materials.

Fig 15: Conversion of coal ash to slurry

This process demonstrates the ingenuity of engineers who optimize every aspect of coal utilization and the business acumen of entrepreneurs who capitalize on waste products, turning them into profitable ventures. It highlights how technology and industry can work together to create sustainable and economically beneficial solutions.

Chapter 3

Literature Survey

During my internship at the Sasan Thermal Power Plant under Reliance Power Limited, I undertook a project focused on predicting grid frequency using machine learning techniques. The electrical grid in India operates at a nominal frequency of 50 Hz, and deviations from this value can have significant implications for the efficiency and safety of the power system. My data showed that grid frequency fluctuated minimally, ranging from 49.9 Hz to 50.1 Hz, indicating a well-regulated grid system. Despite this tight range, predicting frequency is crucial for ensuring grid stability, especially during high-demand or fault conditions. In this review, I will explore the use of time series analysis techniques and machine learning models such as ARIMA, SARIMA, and their underlying components to predict grid frequency.

3.1 Time Series Analysis

Time series analysis involves analysing sequential data points over time to identify underlying patterns and make future predictions. In grid frequency prediction, time series data is ideal since the frequency is recorded at regular intervals, providing a natural sequential dataset.

3.2 Stationarity in Time Series

A critical concept in time series analysis is stationarity, which refers to a time series whose statistical properties such as mean, variance, and autocovariance are constant over time. Stationary series are easier to predict, and most predictive models assume the data is stationary. To achieve this, we often apply transformations such as differencing, normalizing, or detrending. The Augmented Dickey-Fuller (ADF) test is widely used to test the stationarity of a series. During my internship, I conducted this test and transformed the dataset when necessary to meet the stationarity condition, a prerequisite for many models like ARIMA.

3.3 Autocorrelation and Partial Autocorrelation Functions

The Autocorrelation Function (ACF) and the Partial Autocorrelation Function (PACF) are essential tools in time series analysis for identifying the lagged relationships within the data. The ACF provides the correlation of the time series with its own past values, while the PACF isolates the direct correlation of the series with a particular lag. During the data analysis, I used these functions to determine the order of autoregressive (AR) and moving average (MA) components in the ARIMA and SARIMA models. The ACF and PACF plots were instrumental in helping me select the appropriate lag terms for building an effective predictive model.

3.4 Time Series Models

Several models can be used to analyse and forecast time series data. Among them, ARIMA and SARIMA stood out as highly effective tools for predicting grid frequency during my internship project.

3.5 Autoregressive (AR) Model

The AR model is based on the idea that the current value of a series can be explained by its previous values, which are called lags. The AR(p) model involves the prediction of the variable of interest based on a linear combination of past p values. This model assumes that the influence of older observations diminishes over time, which is often seen in grid frequency data.

3.6 Moving Average (MA) Model

In contrast to AR models, the MA model uses past forecast errors in a regression-like model. In the MA(q) model, the time series is expressed as a linear function of q previous error terms. I found this useful when modelling noise components within the frequency data.

3.7 ARMA (Autoregressive Moving Average) Model

The ARMA model combines both AR and MA models to account for both past values and past forecast errors. It is especially useful when the data is stationary and the noise or shocks to the

system persist over time. I found the ARMA model effective when dealing with the short-term dynamics of grid frequency fluctuations.

3.8 ARIMA (Autoregressive Integrated Moving Average) Model

The ARIMA model generalizes the ARMA model by adding a differencing step to deal with non-stationary data. ARIMA (p, d, q) consists of three components:

- p: the number of lag observations in the autoregressive model

- d: the number of times the series needs to be differenced to achieve stationarity

- q: the size of the moving average window

In my project, I used ARIMA to predict grid frequency by tuning these parameters. Differencing the data (d) helped in removing trends and making the data stationary, while the ACF and PACF plots guided the choice of p and q. The model was able to predict grid frequencies with reasonable accuracy by capturing the underlying patterns in the time series data.

3.9 Seasonal ARIMA (SARIMA) Model

The SARIMA model extends ARIMA by including seasonal components to capture periodic behaviour in the data. This was particularly useful in the context of grid frequency, which might show seasonal trends due to fluctuations in power demand during different times of the year. SARIMA (p, d, q) (P, D, Q, s) involves adding terms for seasonal autoregression (P), differencing (D), and moving average (Q), with s representing the length of the seasonal cycle.

Though my dataset did not exhibit strong seasonal patterns, SARIMA proved beneficial in improving model accuracy during specific periods of high demand, such as peak summer months when air conditioning loads significantly affect grid frequency.

3.10 Rolling Mean and Rolling Average

A rolling mean is used to smooth out short-term fluctuations and highlight longer-term trends in time series data. I applied this technique to identify trends in grid frequency data by averaging over a sliding window of data points. This helped reduce noise and visualize the underlying trends more clearly.

3.11 Random Walk and White Noise

A random walk is a stochastic process where the next value is determined by the current value plus a random step. In time series, random walks are often observed when the data shows no discernible pattern or trend. On the other hand, white noise refers to a series of random, uncorrelated data points with a constant mean and variance. These concepts helped me differentiate between patterns in the data and random fluctuations that could not be predicted.

3.12 Model Selection Criteria

3.12.1 Akaike Information Criterion (AIC)

The Akaike Information Criterion (AIC) is a measure used to compare different statistical models. It balances model complexity with goodness of fit by penalizing models with more parameters. In my internship, I used AIC to evaluate and select the optimal model, ensuring that the model was not overfitting the data.

3.12.2 Bayesian Information Criterion (BIC) and Hannan-Quinn Information Criterion (HQIC)

The Bayesian Information Criterion (BIC) and Hannan-Quinn Information Criterion (HQIC) serve similar purposes to AIC but apply a stronger penalty for models with additional parameters. These criteria were useful in validating the robustness of my predictive models, especially when there was a risk of over-parameterization.

3.12.3 Data Normalization and Z-Score

To enhance the performance of machine learning models, I normalized the data. Normalization scales the data to a standard range, while Z-Score normalization specifically transforms the data such that its mean becomes zero and its standard deviation one. Normalizing the frequency data

ensured that each feature contributed equally to the model's performance, avoiding bias toward variables with larger scales.

3.13 Future Work

The analysis and models built during my internship at Sasan Thermal Power Plant provided valuable insights into grid frequency fluctuations. While the data showed only minor deviations from the nominal 50 Hz, the predictive models demonstrated that even small variations could be effectively captured using time series analysis. I successfully identified patterns and applied forecasting models like ARIMA and SARIMA, showcasing their potential for predicting grid frequency with precision.

However, after discussing my findings with the control room head, I learned that while predictions are valuable, the ability to act on them is limited in practice. Adjusting power units in response to predicted frequency variations can risk hardware failure, making real-time operational changes difficult. Although this limitation means that prediction alone cannot directly influence grid operations, my project still provided important theoretical insights and practical knowledge about the dynamics of grid frequency management.

Ultimately, the project was a valuable learning experience that highlighted the strengths of predictive models while also emphasizing the importance of operational constraints. It opened new avenues for future work, such as focusing on preventive strategies rather than real-time adjustments, and laid the foundation for further research on integrating predictive analytics with grid management in ways that align with operational safety. While real-time adjustments may not always be feasible, understanding frequency trends is still crucial for long-term grid stability, making my project a success in providing key insights for future advancements.

Chapter 4

Problem Statement

4.1 Background

The electrical grid operates at a nominal frequency, typically 50 Hz in India, to ensure the stable and efficient distribution of power. Maintaining this frequency within a narrow range is essential for the safe operation of electrical equipment and the overall reliability of the power grid. However, due to fluctuations in power demand, supply variations, and grid disturbances, the frequency can deviate from this ideal value. These deviations, although minor (in India, they range between 49.9 Hz and 50.1 Hz), can result in inefficiencies, damage to equipment, and in extreme cases, grid failures.

Traditionally, grid operators rely on manual interventions and control systems to balance supply and demand and maintain grid frequency. However, these approaches have limitations, particularly as the grid becomes more complex with the integration of renewable energy sources like solar and wind power, which are inherently variable. Predicting these fluctuations in grid frequency using machine learning models can provide grid operators with early warnings and enable them to take preventive measures to avoid frequency excursions.

4.2 Problem

Current methods of grid frequency management are primarily reactive rather than proactive, with operators making real-time adjustments to generation and load based on current conditions. These adjustments often come with delays and are limited by the capabilities of the grid infrastructure. The lack of real-time frequency prediction limits the ability of operators to anticipate disturbances and prevent them from affecting the grid.

The challenge is to develop a machine learning-based prediction model that can forecast grid frequency fluctuations with high accuracy. By predicting these fluctuations in advance, grid operators can take preventive action, such as ramping up or down generation or managing demand through load-shedding or demand response. This is particularly crucial in modern grids, where the increasing penetration of renewable energy sources creates more frequent and unpredictable variations in frequency.

Chapter 5

Description of various Training Module

5.1 Pandas:

Pandas is an essential Python library for data manipulation and analysis, particularly well-suited for handling structured datasets. In this project, Pandas allowed me to manage and manipulate the large dataset of grid frequencies, which had over 116,800 rows spread across three columns: date, block, and frequency. Pandas was instrumental in reading the dataset from a CSV file into a Data Frame, which is a tabular data structure. It enabled me to easily perform tasks like handling missing data, grouping data by date or block, resampling the data into hourly or daily averages, and slicing the dataset to extract specific time periods. The library's powerful data manipulation functions like `groupby()`, `merge()`, and `pivot()` were frequently used to transform and aggregate the data to make it more suitable for time series analysis.

5.2 Numpy:

Numpy, short for Numerical Python, is the core library for numerical and matrix computations in Python. In my project, Numpy was essential for handling large arrays of grid frequency data. It helped perform efficient mathematical operations such as calculating moving averages, sums, and differences between grid frequency values over time. Since time series analysis often involves working with large datasets, Numpy's array structure (ndarray) allowed me to perform these operations with much greater speed and efficiency than native Python lists. Moreover, Numpy's random number generation and statistical functions were helpful when I needed to simulate or test certain time series behaviours during exploratory analysis.

5.3 Matplotlib:

Matplotlib is a versatile plotting library in Python that allowed me to visualize the grid frequency data over time. By creating time series plots using Matplotlib, I was able to detect underlying patterns in the data, such as trends, cycles, and outliers. This visualization was crucial for understanding how grid frequency fluctuates over days, weeks, and months. I also used Matplotlib to generate plots of the residuals after fitting my ARIMA model, which helped in

diagnosing the model's performance and identifying any remaining patterns in the error terms. Plot types like line charts, histograms, and scatter plots provided clear insights into the dataset, making it easier to present my findings in the seminar.

5.4 Seaborn:

Seaborn is a higher-level visualization library built on Matplotlib that provides a more aesthetically pleasing and easier-to-implement interface for statistical plotting. In my project, I used Seaborn to create advanced visualizations such as heatmaps and correlation matrices, which helped me explore the relationships between different blocks of time and how grid frequency varied across seasons. The ability to create these detailed and attractive visualizations made it easier to interpret and present the results. Seaborn's integration with Pandas also made it convenient to directly pass Data Frames for plotting, making the visualization process smoother.

5.5 Statsmodels.tsa.stattools:

The `statsmodels.tsa.stattools` module contains various statistical tests and tools for time series analysis. This was a critical component in my project, particularly for diagnosing stationarity in the grid frequency data, which is a key prerequisite for most time series models. The module provides access to tests like the Augmented Dickey-Fuller (ADF) test and the KPSS test, which I used to check whether the grid frequency data had a constant mean and variance over time. Additionally, the module provided tools to compute the autocorrelation function (ACF) and partial autocorrelation function (PACF), which helped in identifying the appropriate lags for my ARIMA model. Understanding these lags was crucial for building an accurate model.

5.6 AD Fuller (Augmented Dickey-Fuller Test):

The Augmented Dickey-Fuller (ADF) test, part of the `stattools` module, was key to determining the stationarity of the time series. Stationarity means that the statistical properties of the data (like mean, variance, and autocorrelation) remain constant over time, which is a fundamental requirement for many time series models. The ADF test helped me assess whether the grid frequency data needed differencing (removing trends and seasonality) to become stationary. A p-value from the ADF test below a certain threshold (e.g., 0.05) indicates that the data is stationary.

In my project, applying this test helped me decide whether transformations or differencing were required before modelling the data.

5.7 From Scipy.stats.distributions import chi2:

Chi-squared distribution is commonly used in hypothesis testing, and I used it for model diagnostics in my project. Specifically, after fitting the ARIMA model, I needed to ensure that the residuals (the difference between the actual and predicted values) followed a normal distribution with zero mean and constant variance. The chi-squared distribution helped me assess whether the residuals were white noise, meaning they contained no additional information that the model failed to capture. If the residuals passed this test, it indicated that the model was well-fitted to the data.

5.8 Statsmodels.tsa.seasonal import seasonal_decompose:

Seasonal decomposition was used to break down the grid frequency time series into its constituent components: trend, seasonality, and residuals. This was important for understanding how much of the variation in the grid frequency was due to underlying trends (such as long-term shifts in demand), seasonal patterns (like daily or weekly cycles), and random noise. By visualizing and analysing these components separately, I gained insights into the overall structure of the data. This process also helped me decide whether to use a seasonal ARIMA (SARIMA) model to capture the recurring patterns in the data.

5.9 Statsmodels.tsa.arima.model import ARIMA:

ARIMA, short for Auto Regressive Integrated Moving Average, was the core model used for predicting grid frequency. ARIMA models are widely used for forecasting time series data by capturing relationships between past observations and future values. In my project, I applied the ARIMA model to forecast grid frequency based on historical data. The model combines three components: autoregression (AR), differencing to make the series stationary (I for Integrated), and moving averages (MA). By selecting the appropriate AR, I, and MA terms based on the ACF and PACF plots, I built a model that could predict future grid frequencies with a high degree of accuracy. ARIMA was particularly useful because it allowed for flexible modelling of both short-term dependencies (via autoregression) and noise (via moving averages) in the data.

This comprehensive use of libraries and tools gave me a holistic understanding of how to handle large time series datasets, build predictive models, and validate those models through rigorous statistical testing. Each module played a crucial role in different phases of my project, from data cleaning and transformation to visualization and model building.

Chapter 6

Methodology adopted

Since time series analysis was a completely new domain for me, I initially struggled with understanding how to approach the problem. To bridge this knowledge gap, I extensively researched online resources, including articles, tutorials, and documentation on time series analysis. I quickly learned that plotting the data to observe its patterns would be a good starting point. When I initially plotted the raw grid frequency data, however, it appeared highly noisy and did not reveal any discernible patterns or trends. This marked the beginning of a more rigorous, structured approach to my analysis.

6.1 Design of Experiment

6.1.1. Introduction to Grid Frequency Prediction

The power grid's frequency stability is crucial for the reliable operation of any electrical system. The grid frequency in India is ideally maintained at 50 Hz, and any deviations from this can lead to inefficiencies or damage to electrical equipment. Predicting grid frequency helps utilities maintain this balance between power supply and demand.

My project focused on predicting grid frequency fluctuations using machine learning techniques. Given the critical importance of grid stability, the dataset I was provided with spanned 1217 days and was recorded at 15-minute intervals, leading to a large dataset of 96 data blocks per day. Each block represented the frequency at a particular moment. While this seemed straightforward, the real challenge lay in uncovering hidden trends amidst the highly fluctuating data, a task that required both domain knowledge and advanced data analysis techniques.

6.1.2. Initial Exploration and Challenges with Raw Data

The initial step in my experimental design was a granular exploration of the raw dataset. Given the large dataset with approximately 116,832 rows (1217 days × 96 blocks per day), my first objective was to visualize this data to understand its structure and patterns. To achieve this, I plotted the grid frequency for a few consecutive days.

6.1.2.1 Plotting Raw Data

When plotting the raw data, my expectation was to observe some recurring patterns or trends in the frequency values. However, despite multiple visualization attempts, the results were disappointing. The dataset appeared overwhelmingly noisy, with no obvious trends or repeating patterns. Each data point seemed disconnected, and it was clear that the inherent variability in grid frequency made it difficult to identify any straightforward insights.

6.1.2.2 Nature of Noise in Frequency Data

The noise in this dataset stemmed from the nature of the grid frequency, which is sensitive to immediate fluctuations in power generation and consumption. This sensitivity creates constant short-term variations that, when viewed over long periods, obscure the longer-term trends and seasonal cycles that are more meaningful for predictive analysis. Therefore, before attempting any predictive modelling, it was necessary to preprocess and clean the data to reduce this noise and enhance its underlying structure.

6.1.3. First Attempts with Machine Learning Models

Having recognized the challenge of noise in the dataset, I decided to apply some basic machine learning models. The models I initially explored included multivariate linear regression, Support Vector Machines (SVM), and decision trees.

6.1.3.1 Multivariate Linear Regression

Linear regression is one of the most common machine learning models for predictive tasks. It attempts to model the relationship between independent variables (time blocks in my case) and a dependent variable (frequency). The idea was that if there was any linear relationship between the time of day and grid frequency, linear regression would be able to capture it. However, the raw dataset did not show any meaningful linear relationships. The error rates were high, and the model failed to generalize, likely because grid frequency is influenced by complex, non-linear factors that this simple model could not capture.

6.1.3.2 Support Vector Machines (SVM)

Next, I attempted to apply SVM, a model designed to classify data points by finding a hyperplane that separates different classes in a dataset. Since my task involved continuous numerical prediction, I adapted the SVM model for regression (SVR). However, the results were similarly disappointing. The SVR model was not able to fit the data due to the noise and lack of distinct structure in the raw frequency data.

6.1.3.3 Decision Trees

As a last attempt with simple models, I employed decision trees. Decision trees work by splitting the data into subgroups based on certain decision rules, eventually forming a tree structure that can be used for predictions. This model seemed more promising as it could handle non-linear relationships better than regression models. However, decision trees tend to overfit on noisy data, and that was exactly what happened in this case. The model captured too much noise, leading to poor generalization on unseen data.

6.1.4. Data Preprocessing and Aggregation Strategy

Realizing that machine learning models were struggling due to the raw data's noisy nature, I shifted focus to a more structured data preprocessing approach. Preprocessing is a critical step in any data-driven project, especially in time series analysis where noise can mask the trends.

6.1.4.1 Aggregation of Data

The first preprocessing step I implemented was the aggregation of the frequency data into larger time intervals. Instead of working with the raw 15-minute intervals, I grouped consecutive 15-minute blocks into 1-hour intervals by averaging the frequency values for each hour. This drastically reduced the dataset from 96 blocks per day to 24 blocks, representing the average frequency for each hour of the day. This not only reduced the noise but also allowed for better visualization of long-term patterns across the days.

6.1.4.1.1 Advantages of Aggregation

By aggregating the data, short-term fluctuations were smoothed out, revealing longer-term trends. The averaging process acted as a noise filter, allowing the underlying behaviour of the grid frequency over the course of a day to become more apparent. This method is particularly effective in time series analysis where cyclical patterns can be hidden by high-frequency noise.

6.1.4.1.2 Further Aggregations for Different Intervals

In addition to the 1-hour aggregation, I applied similar aggregation strategies for different intervals, including half-hour, 15-minute, and 45-minute intervals. Each aggregation level allowed me to analyse the dataset at different resolutions. Shorter intervals provided more detailed views of the data, while longer intervals revealed broader trends.

6.1.4.2 Normalization and Z-Score Calculation

While aggregation helped reduce noise, I further enhanced the data's interpretability by normalizing the frequency values. I calculated the Z-scores for the frequency data, which standardizes the values by measuring how far they deviate from the mean in terms of standard deviations. The Z-score formula is given by:

$$z = \frac{x-\mu}{\sigma}$$

By converting the raw frequency values into Z-scores, I effectively normalized the data and made it easier to compare different intervals on the same scale. I also multiplied the Z-scores by 100 to amplify their values for better visualization.

6.1.4.3 Plotting the Normalized Data

Once the Z-scores were calculated, I plotted the normalized data for various time intervals (hourly, half-hourly, 15-minute, and 45-minute intervals). At this stage, I noticed emerging

patterns in the dataset. For example, the hourly frequency data showed consistent peaks during certain times of the day, reflecting the daily cycles of power demand. Similarly, the smaller time intervals (15 minutes and 30 minutes) revealed more localized patterns that were previously hidden in the noise.

6.1.5. Conclusion and Future Work

The methodology I adopted was a step-by-step process that gradually refined the data and revealed meaningful patterns. By applying data aggregation, normalization, and time series analysis techniques, I was able to overcome the challenges posed by the noisy raw dataset. While the initial machine learning models did not yield satisfactory results, the preprocessing steps laid the foundation for more advanced time series forecasting models, such as ARIMA, which could be applied in future work.

Moving forward, further optimization could involve incorporating external variables such as weather data or national holidays to improve the predictive accuracy of the models. Additionally, more sophisticated machine learning models like recurrent neural networks (RNNs) could be explored for long-term forecasting of grid frequency.

In summary, this project demonstrated the importance of data preprocessing and exploratory analysis in uncovering hidden patterns in time series data. It also highlighted the limitations of basic machine learning models when applied to highly noisy datasets, reinforcing the need for specialized time series analysis techniques.

6.2 Hardware and Software used

For this project, I utilized a combination of hardware and software resources to perform the analysis and implement machine learning models. The hardware setup included a standard workstation equipped with an Intel i5 processor and 8 GB of RAM, which was sufficient for processing the dataset and executing various machine learning algorithms. On the software side, Python was the primary programming language used throughout the project, and Jupyter Notebook provided an interactive environment for writing and running Python code. For data manipulation and preprocessing tasks, including cleaning, aggregation, and resampling, I relied on the Pandas library, while Numpy was employed for efficient numerical computations, such as Z-score calculations and array manipulations. Data visualization was accomplished using

Matplotlib and Seaborn, which helped generate insightful plots of both the raw and processed data. Time series analysis was conducted using the Statsmodels library, which facilitated statistical tests like the Augmented Dickey-Fuller (ADF) test for stationarity. Additionally, machine learning models such as multivariate regression, SVM, and decision trees were implemented using the Scikit-learn library to explore various predictive approaches.

6.3 Optimization

In the methodology for optimizing the dataset and applying machine learning models, the initial focus was on data preprocessing and aggregation. The dataset, comprising grid frequency values recorded at 15-minute intervals across 1217 days, required careful handling to reduce the noise inherent in such large and granular time series data. The first optimization step involved aggregating these 15-minute intervals into larger time blocks to simplify the dataset and highlight any latent patterns that could be obscured by short-term fluctuations.

6.3.1. Data Preprocessing and Aggregation

I began by aggregating every 4 consecutive 15-minute blocks into hourly intervals, effectively reducing the dataset size from 96 rows per day to 24 rows per day. Each new row represented the average frequency value for a specific hour of the day. This aggregation process helped smooth out short-term variations in grid frequency, which were largely due to momentary fluctuations, and allowed for a more focused analysis of long-term trends and patterns. After aggregating the data into hourly intervals, I further grouped these hourly blocks across all the days in the dataset. The result was a new dataset consisting of 24 rows, each corresponding to the average frequency for a specific hour across the entire 1217-day period. This reduction in dataset size not only facilitated easier visualization of frequency patterns but also enabled the computational resources to be used more efficiently.

To enhance the depth of the analysis, I repeated this aggregation process for different time intervals, including half-hour, 45-minute, and 15-minute intervals. By doing so, I could compare and analyse how grid frequency trends varied depending on the level of granularity. The 45-minute and half-hour intervals offered a middle ground between the 15-minute and 1-hour blocks, providing a balanced view of short-term and long-term trends. Through this iterative process of data aggregation, I was able to visualize distinct patterns that had been obscured by the granularity of the raw data.

6.3.2. Normalization

Following the aggregation process, the next key step was normalization. This involved calculating the Z-score for each data point within the dataset. The Z-score is a standard statistical measure that represents how many standard deviations a data point is from the mean of the dataset. In this case, normalization allowed me to standardize the dataset and convert the raw frequency values into a more interpretable scale. Normalizing the data helped to reveal patterns and trends that were not immediately obvious in the raw frequency data.

Moreover, by multiplying the Z-scores by 100, I could amplify these normalized values, making it easier to detect anomalies and outliers. The scaling helped improve the clarity of the visualized data and brought attention to significant deviations in grid frequency during specific hours of the day. This transformation was crucial in making the data patterns more discernible and in aiding the subsequent stages of analysis.

6.3.3. Data Flow Diagram

The overall data flow for this project began with the raw dataset, which consisted of grid frequency values recorded in 15-minute intervals over 1217 days. The first step was preprocessing, where I aggregated these values into larger time blocks (e.g., hourly, half-hour, 45-minute, and 15-minute intervals). After aggregation, the data was normalized through Z-score calculation, which standardized the values and allowed for easier detection of patterns and anomalies.

Once the data had been pre-processed and normalized, I visualized it using various plotting techniques, which helped uncover patterns that were previously hidden in the raw data. Despite my attempts to apply machine learning models like multivariate regression, SVM, and decision trees, the noisy nature of the dataset made these approaches ineffective. Instead, time series decomposition, combined with Z-score normalization, proved to be the most effective method for identifying the underlying trends and seasonal patterns in the grid frequency data.

In summary, this process of aggregation, normalization, and iterative analysis was crucial in optimizing the dataset and revealing patterns that were not apparent in the raw data. Although machine learning models were initially applied, they did not yield accurate predictions due to the irregular and noisy nature of the dataset. The combination of time series decomposition and statistical normalization, however, provided valuable insights into the behaviour of grid frequency over time.

6.4 Snapshots of results obtained

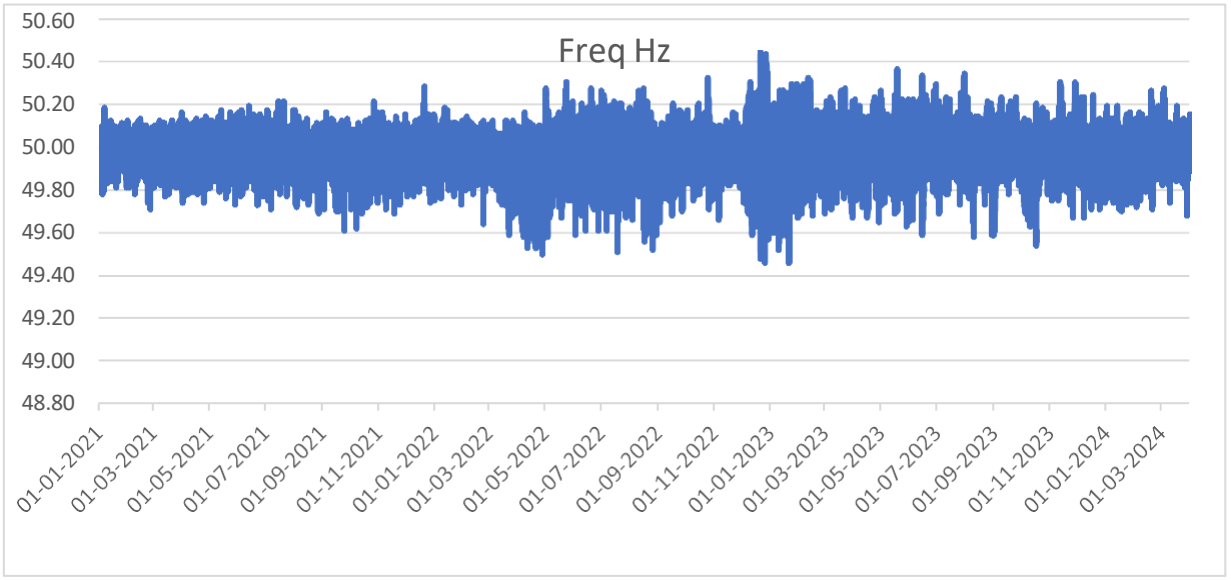

Fig 16: Plot of grid frequency vs Dates

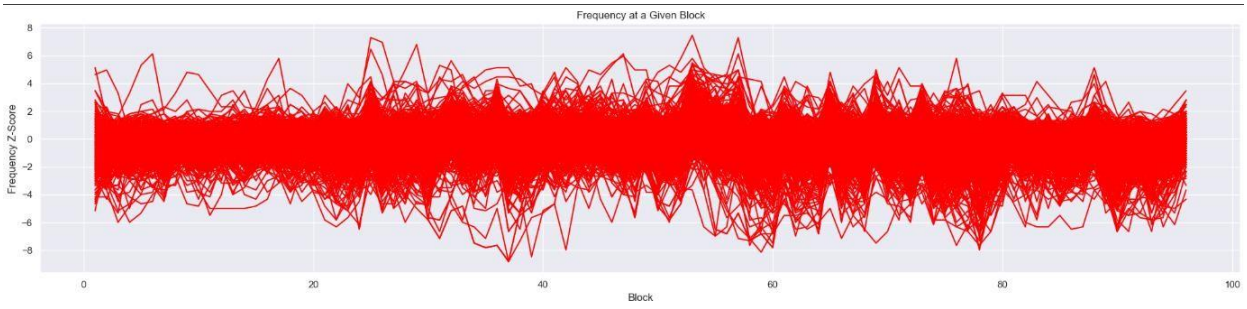

Fig 17: Plot of grid frequency vs blocks

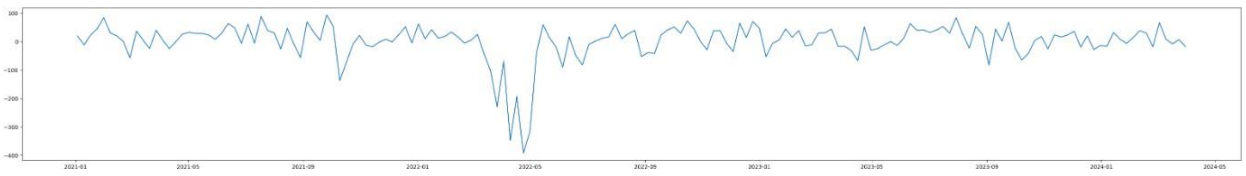

Fig 18: Plot of January grid frequency

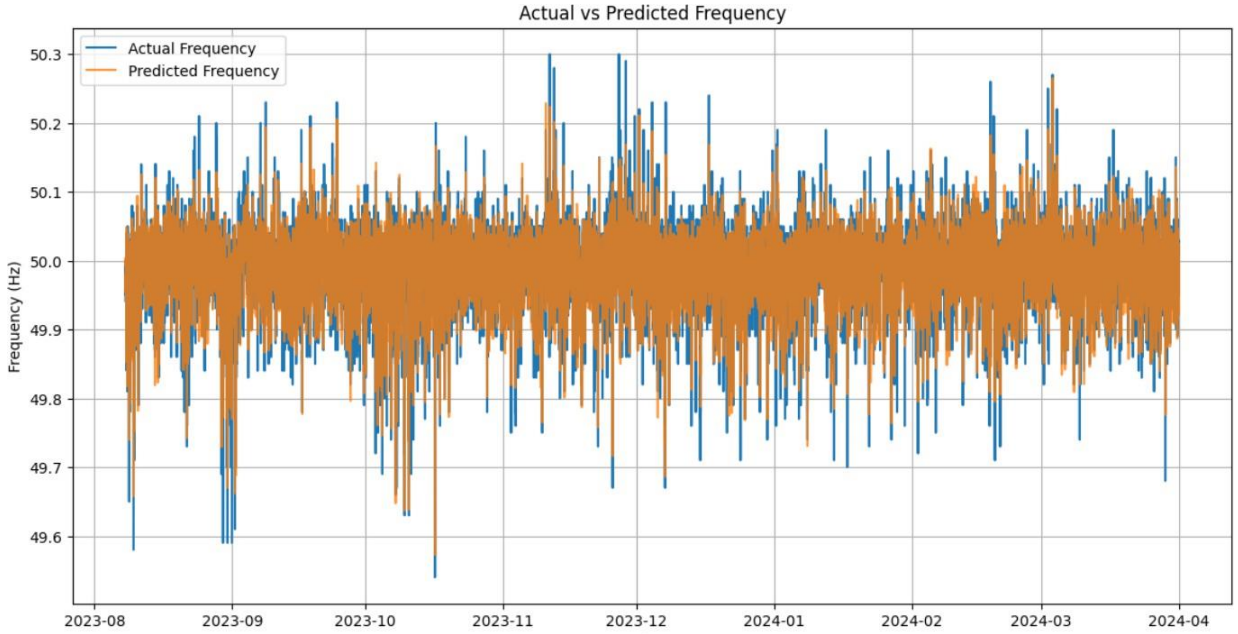

Fig 19: Actual frequency and Predicted frequency by random forest

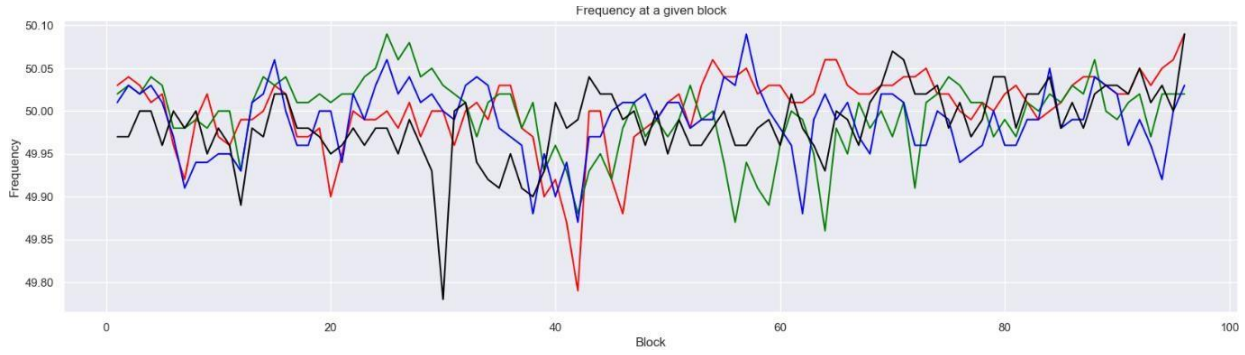

Fig 20: Grid frequency vs blocks for 3 days

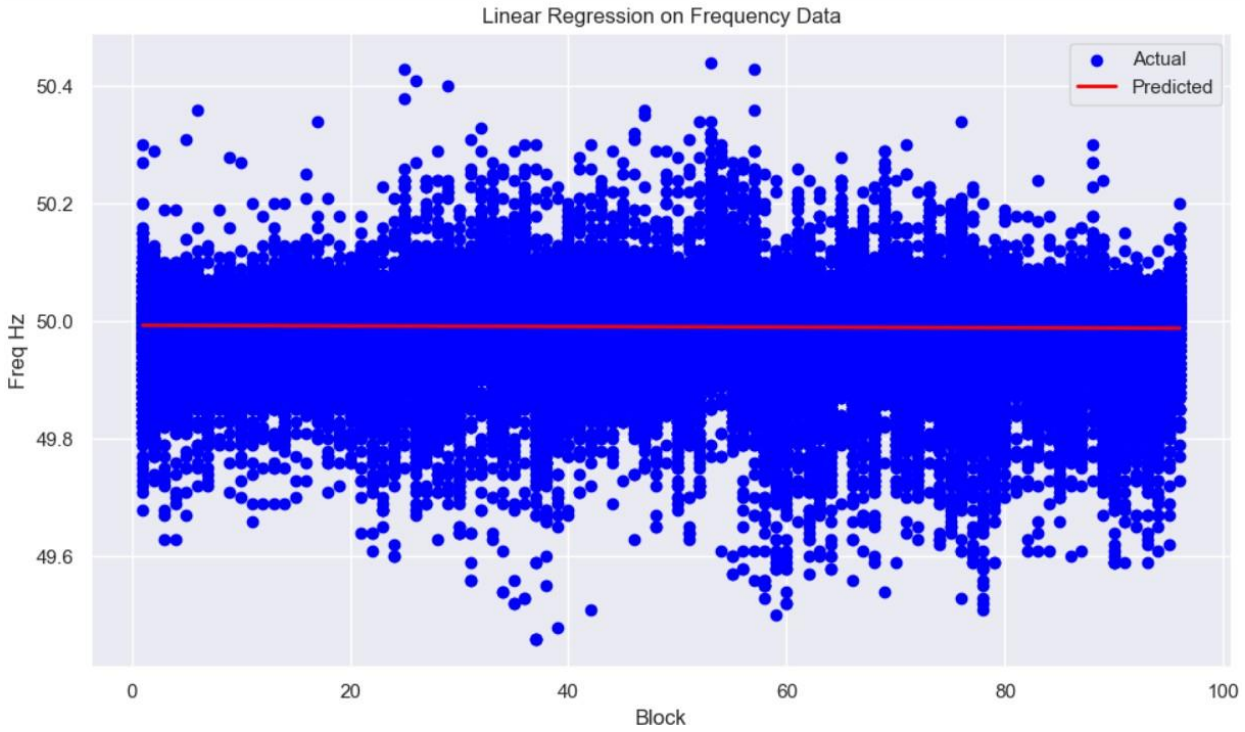

Fig 21: Linear regression on frequency data

Fig 22: SVM Regression on Frequency data

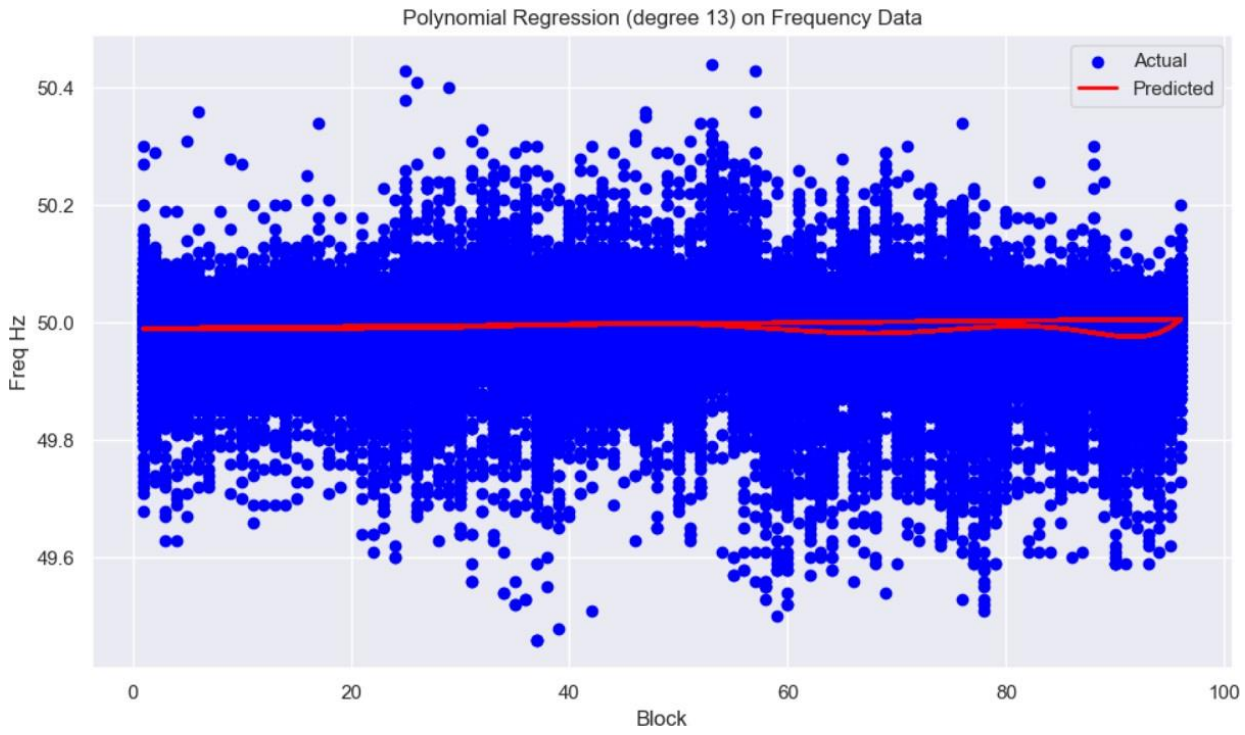

Fig 23: Polynomial regression (degree 13) on frequency data

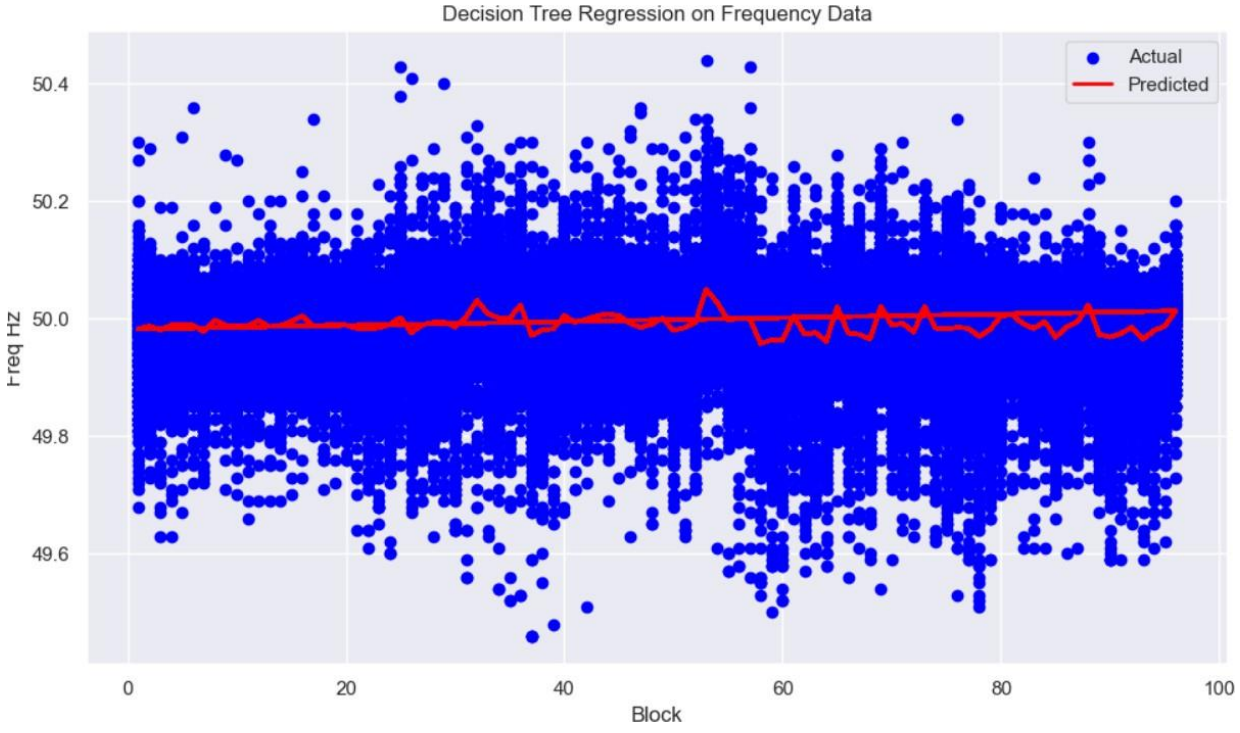

Fig 24: Decision tree regression

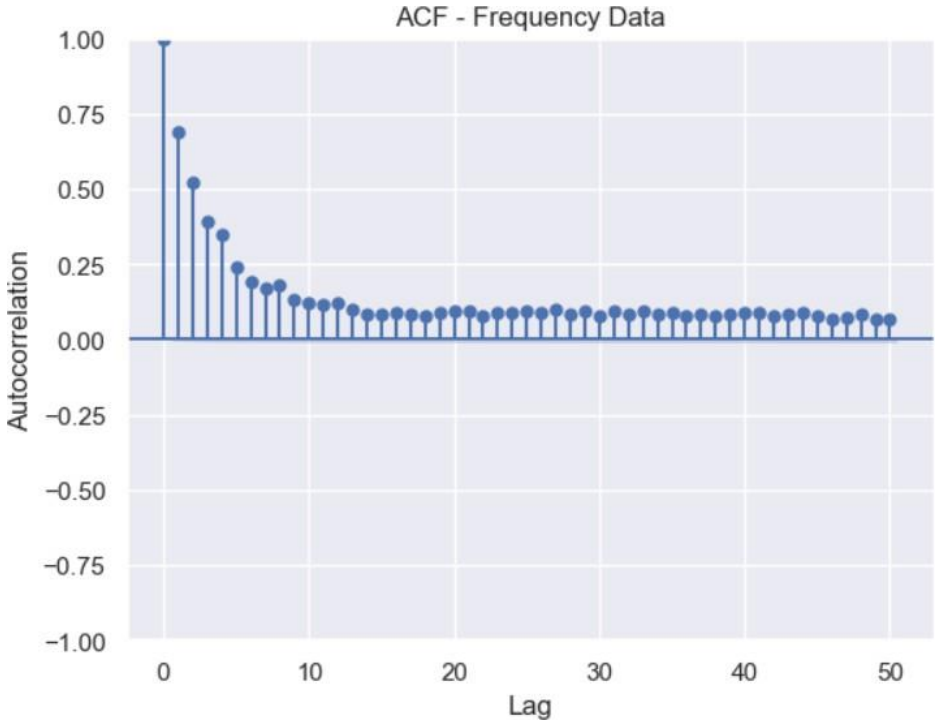

Fig 25: ACF plot for frequency data

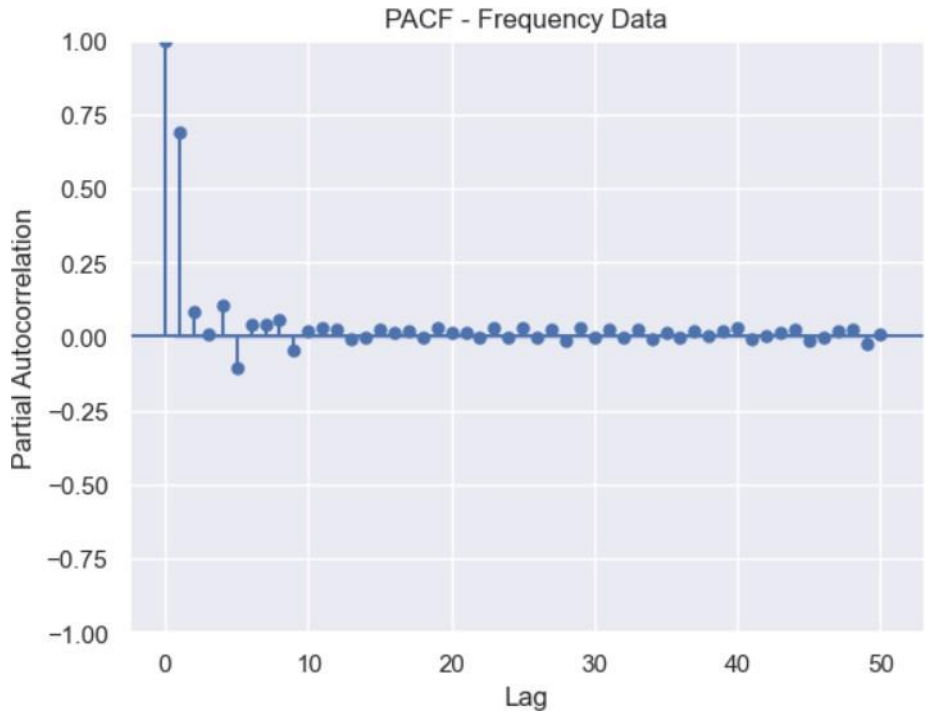

Fig 26: PACF plot for frequency data

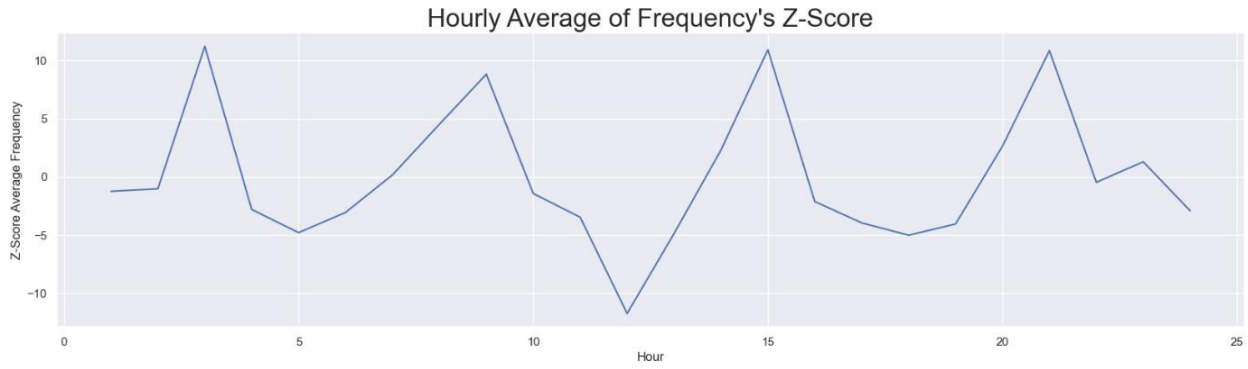

Fig 27: Hourly average of frequency's Z-Score

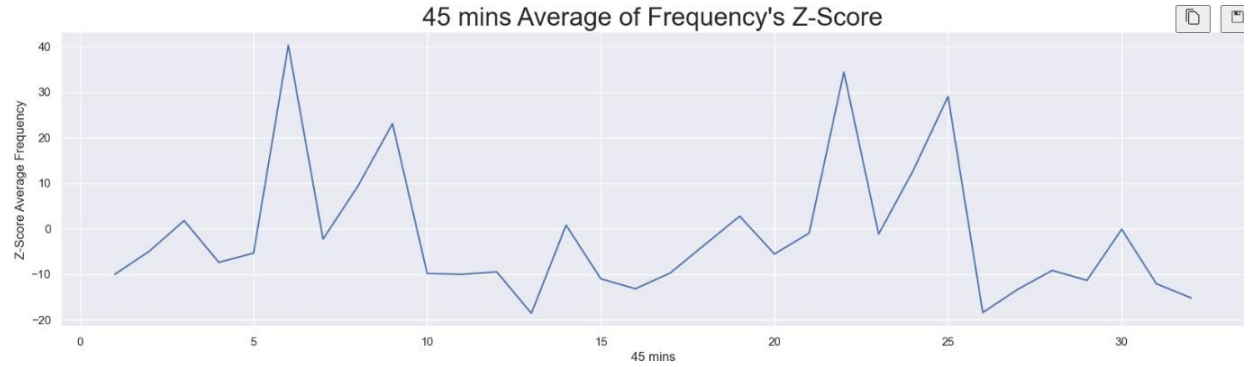

Fig 28: 45 mins average of frequency's Z-Score

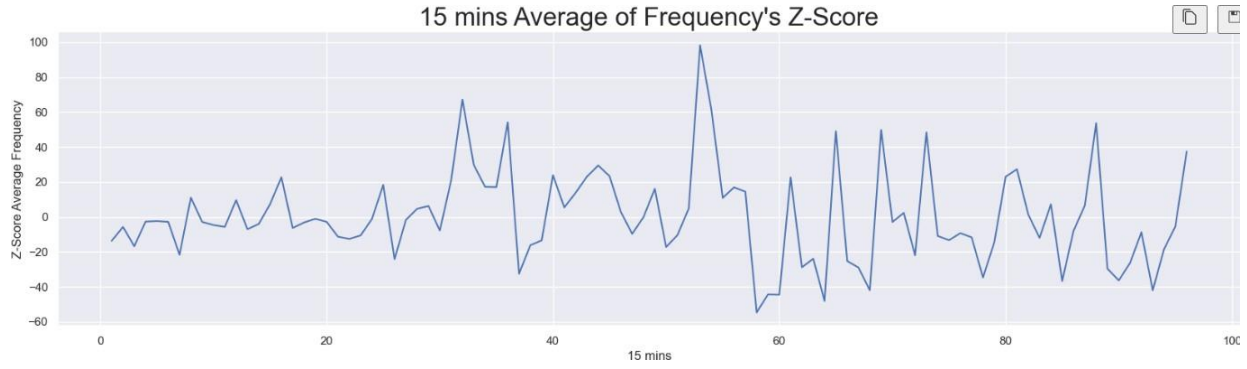

Fig 29: 15 mins average of frequency's Z-Score

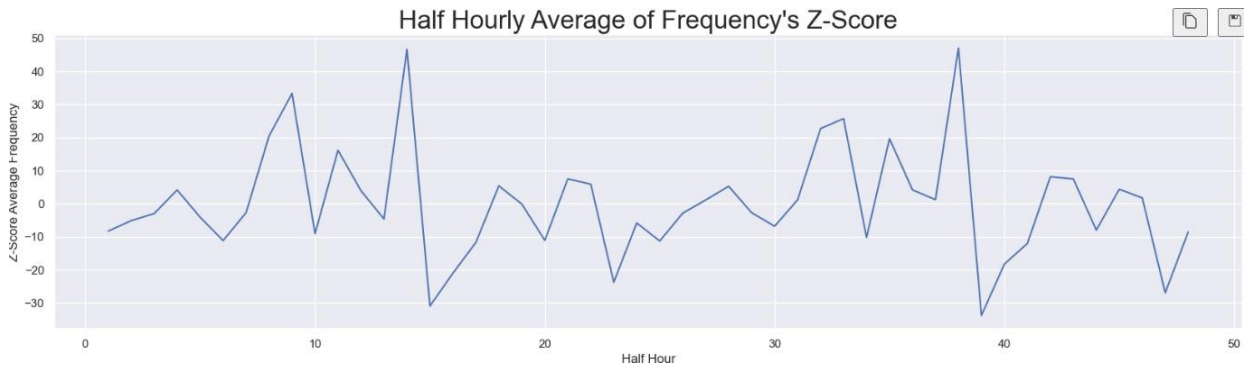

Fig 30: Half hourly average of frequency's Z-Score

Chapter 7

Results and Discussion

The analysis of the hourly average frequency data involved several stages, including data preprocessing, model application, and performance evaluation. The preprocessing steps were critical in transforming the raw data into a more manageable form. Initially, the data, comprising 15-minute intervals, was aggregated into hourly averages to simplify the analysis and reduce short-term fluctuations. This aggregation was performed by combining four 15-minute blocks into one-hour intervals. Subsequently, different time intervals—half-hourly, 45-minute, and 15-minute—were analysed to explore the impact of granularity on pattern detection.

The normalization of the data through Z-score calculations proved to be a significant step in highlighting patterns that were previously obscured by noise. By converting the frequency values into Z-scores and amplifying them for better visualization, some recurring trends and seasonal patterns emerged. However, these patterns were not consistently stable across different time intervals. This variability in pattern consistency was a key observation during the analysis.

The application of ARIMA and SARIMA models provided further insights into the data's temporal characteristics. The ARIMA model, configured with parameters (1, 1, 1), was applied to the training dataset and tested on a separate test set. The performance of the ARIMA model, evaluated using mean squared error (MSE), indicated that the model could capture some of the data’s temporal trends, but the predictions were not highly accurate. The MSE for the ARIMA model suggested that while the model offered some predictive capability, its performance was limited by the noisy nature of the data.

Similarly, the SARIMA model was applied with parameters (1, 1, 1)(1, 1, 1, 12), assuming daily seasonality. The SARIMA model aimed to capture both seasonal and non-seasonal components of the data. Although the SARIMA model also provided some useful predictions, it faced similar challenges in accuracy due to the inherent variability in the dataset. The mean squared error for the SARIMA model reflected the difficulty in achieving precise predictions, underscoring the complexity of the data.

7.1. Discussion

The results of this analysis highlight several critical insights and challenges associated with time series analysis and predictive modelling in the context of power frequency data. The preprocessing steps, including data aggregation and normalization, were essential in managing the high variability of the raw data. By aggregating the data into larger time intervals and applying Z-score normalization, clearer patterns emerged, though they were not entirely consistent. This variability in patterns can be attributed to the inherent fluctuations in grid frequency, which are influenced by numerous operational factors.

The ARIMA and SARIMA models provided a framework for understanding the temporal dynamics of the data. The ARIMA model, with its ability to capture autoregressive and moving average components, offered some insights into the short-term trends in the frequency data. However, its predictive performance was constrained by the noisy nature of the data, which made it challenging to derive highly accurate forecasts.

The SARIMA model, with its seasonal components, was designed to capture both seasonal and non-seasonal variations. The model's performance highlighted the importance of accounting for seasonal effects in time series data. However, despite the inclusion of seasonal parameters, the SARIMA model faced similar challenges in accuracy. This indicates that while seasonal modelling can enhance the understanding of periodic trends, it may not fully address the complexities of highly variable datasets.

One of the key takeaways from the analysis is the limitation imposed by the fixed nature of large-scale hardware systems. The predictive models, while theoretically useful, could not directly influence or alter the operational aspects of the power units. This underscores the practical challenge of applying predictive analytics in environments where hardware and operational constraints are rigid.

Additionally, the process of exploring different time intervals and normalization techniques provided valuable insights into the granularity of data analysis. While some patterns were identified, their inconsistency across different intervals suggests that further refinement of models and techniques may be needed to achieve more reliable predictions.

In conclusion, the results of this project reflect the complexity of predictive modelling in the context of industrial power frequency data. The preprocessing and modelling efforts provided useful insights into the data's temporal characteristics but highlighted the limitations in achieving consistent and accurate predictions. The experience gained underscores the importance of adapting analytical approaches to the specific characteristics of the data and the operational

constraints of the system. Future work may involve exploring advanced modelling techniques or incorporating additional contextual factors to enhance prediction accuracy and practical applicability.

Chapter 8

Conclusion

In this project, significant efforts were made to analyse and model the hourly average frequency data using various statistical and machine learning techniques. The journey began with a comprehensive exploration of the dataset, including an initial attempt to discern patterns through visual inspection and basic machine learning models. The raw data, characterized by its high variability and noise, posed substantial challenges in deriving meaningful insights directly.

Through systematic data preprocessing, including the aggregation of 15-minute intervals into hourly averages and the application of normalization techniques, we successfully reduced the noise and revealed some underlying patterns in the data. This preprocessing step was crucial in simplifying the dataset and making it more amenable to analysis. By grouping and averaging data across different time intervals (hourly, half-hourly, 15-minute, and 45-minute), we could better visualize and understand the temporal trends and seasonal effects.

Despite these improvements, the consistency of the identified patterns was limited. The data exhibited some recurring trends, but they were not stable enough to provide reliable predictions. The inherent complexity and variability of the grid frequency data contributed to the challenges in achieving precise and consistent results. The application of ARIMA and SARIMA models, while providing some insights, highlighted the difficulty of capturing the nuances of such a highly variable dataset.

The limitations of the predictive models underscore an important aspect of industrial applications: the inherent constraints posed by large-scale hardware systems. In this context, while the models offered a theoretical understanding of the data trends, their practical utility was constrained by the fixed nature of unit operations and hardware configurations. Consequently, although the predictive insights could inform theoretical improvements, they could not directly influence or alter the operational aspects of the power units.

In summary, the project demonstrated the value of data preprocessing in enhancing the clarity of patterns within noisy datasets and provided insights into the challenges of predictive modelling in complex industrial contexts. While the models did not yield highly actionable predictions, the process underscored the importance of adapting analytical techniques to the specific characteristics of the data and the constraints of the operational environment. The experience gained from this project contributes to a deeper understanding of the interplay between data analysis, modelling techniques, and practical limitations in large-scale systems.

Chapter 9

References

1. https://otexts.com/fpp2/seasonal-plots.html
2. https://www.udemy.com/course/time-series-analysis-in-python/learn/lecture/16275878?start=15#overview
3. Cryer, J.D., 1986. *Time series analysis* (Vol. 286). Boston: Duxbury Press.
4. Kruse, J., Schäfer, B. and Witthaut, D., 2020. Predictability of power grid frequency. *IEEE access*, *8*, pp.149435-149446.
5. Onsaker, T.L., Nygård, H.S., Gomila, D., Colet, P., Mikut, R., Jumar, R., Maass, H., Kühnapfel, U., Hagenmeyer, V. and Schäfer, B., 2023. Predicting the power grid frequency of European islands. *Journal of Physics: Complexity*, *4*(1), p.015012.