SUNSPOT CYCLE PREDICTION: EXPLORING DATA-DRIVEN AND MACHINE LEARNING APPROACHES

MSc Astronomy Thesis

By:

Daisy Rani Boro

DEPARTMENT OF ASTRONOMY , ASTROPHYSICS AND SPACE ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

May, 2025

SUNSPOT CYCLE PREDICTION: EXPLORING DATA-DRIVEN AND MACHINE LEARNING APPROACHES

A THESIS

Submitted in partial fulfillment of the
requirements for the award of the degree
of

Master of Science

by

Daisy Rani Boro

DEPARTMENT OF ASTRONOMY, ASTROPHYSICS AND SPACE ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

MAY 2025

INDIAN INSTITUTE OF TECHNOLOGY INDORE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **Sunspot Cycle Prediction: Exploring Data-Driven and Machine Learning Approaches** in the partial fulfillment of the requirements for the award of the degree of **MASTER OF SCIENCE** and submitted in the **DEPARTMENT OF ASTRONOMY ASTROPHYSICS AND SPACE ENGINEERING, Indian Institute of Technology Indore**, is an authentic record of my own work carried out during the time period from July-2023 to May-2025 under the supervision of Dr. Amit Shukla , Associate Professor , DAASE, IIT Indore.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other institute.

Daisy Rani Boro

13-05-2025

Signature of the student with date
(Daisy Rani Boro)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

13-05-2025

Signature of the Supervisor of
M.Sc. thesis (with date)
(Dr. Amit Shukla)

Daisy Rani Boro has successfully given her M.Sc. Oral Examination held on **05-05-2025**

Amit Shukla

Signature(s) of Supervisor(s) of MSc thesis
Date: 13-05-2025

Manoneeta Chakraborty

Programme Coordinator, M.Sc.
Date: 19/05/2025

Manoneeta Chakraborty

Convenor, DPGC
Date: 19/05/2025

HoD, DAASE
Date:

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to all those who have supported and encouraged me throughout the course of this work.

First and foremost, I am deeply grateful to my supervisor, Dr. Amit Shukla, for his consistent guidance, encouragement, and insightful feedback, which were invaluable in shaping the direction of this thesis.

I would also like to thank the faculty members and staff of the Department of Astronomy Astrophysics and Space Engineering, IIT Indore, for providing the academic environment and resources necessary for my research.

A special thanks to Prasad for his constant help with coding and technical support, especially during challenging phases of the project. I genuinely appreciate his patience and willingness to assist at any time.

I am also thankful to my wonderful lab mates — Hari, Aman, Chandhan, Ayush, Shradha, Sushmita and Priyanka — for their support, camaraderie, and for making the lab environment enjoyable and motivating.

I extend my warmest thanks to my friends —Vishruth, Vijay, Parth, Gitaj, Amar, Devesh, Aryan, Annie, Anushka, Ashutosh, Anand, and Navanit for their constant moral support, brainstorming sessions, and for making this journey a memorable one.

Finally, I am forever grateful to my family for their unwavering support, love, and motivation, without which this endeavor would not have been possible.

ABSTRACT

The prediction of solar phenomena, such as sunspot activity, plays a critical role in understanding the solar cycle and its impact on space weather. This thesis explores the application of time series analysis techniques to forecast sunspot numbers during the solar minima of the 25th solar cycle, using data from the Sunspot Index and Long-term Solar Observations (SILSO). Five predictive models were evaluated: ARIMA, SARIMA, Random Forest, XGBoost,and LSTM with a focus on assessing their accuracy and robustness in predicting sunspot counts.The classical models, ARIMA and SARIMA, exhibited higher error values compared to machine learning approaches, with ARIMA recording a Mean Absolute Error (MAE) of 57.60 and a Root Mean Squared Error (RMSE) of 70.98, while SARIMA showed similar performance. These results suggest that classical models struggle to capture the underlying patterns of sunspot data. In contrast, machine learning models, particularly Random Forest, significantly outperformed the classical methods. The Random Forest model with a lag of 7 produced the lowest MAE (15.04) and RMSE (19.35), making it the most effective model in this comparison. Although XGBoost also demonstrated strong performance, increasing the lag from 7 to 12 led to a slight increase in errors, possibly due to overfitting. Additionally, a deep learning model, LSTM, was tested. The LSTM model with a lag of 7 yielded an MAE of 15.82 and RMSE of 20.91, but its performance worsened when the lag was increased to 12, highlighting the need for careful tuning and optimization in deep learning models.Overall, the study demonstrates that machine learning models, especially Random Forest with a lag of 7, provide superior predictive performance over traditional approaches. This work highlights the importance of data preprocessing, model selection, and optimization in time series forecasting, contributing to the broader field of solar physics and space weather prediction.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The study of sunspot activity is crucial for understanding the dynamic behavior of the Sun and its impact on space weather. Sunspots, which are regions of intense magnetic activity, exhibit patterns that evolve over time and reveal important information about solar cycles. This thesis focuses on analyzing sunspot numbers using time series methods, utilizing data from the Sunspot Index and Long-term Solar Observations (SILSO) database. By applying advanced techniques for trend extraction, seasonal adjustment, and forecasting, the project aims to uncover underlying periodicities and improve the predictive modeling of solar activity. A comprehensive understanding of these variations not only enhances our knowledge of solar physics but also contributes to better forecasting models for space weather phenomena affecting Earth.

1.1 Historical Background

Sunspots have been observed for centuries, becoming one of the earliest recorded phenomena in solar astronomy. Ancient Chinese astronomers documented dark spots on the Sun as early as 364 BCE .[3]. In these early observations, sunspots were often interpreted as omens or supernatural signs rather than natural features of the Sun.

A major shift in understanding occurred in the early 17th century with the invention of the telescope. Pioneering astronomers such as Galileo Galilei, Thomas Harriot, and Christoph Scheiner independently studied the Sun through telescopes, identifying and documenting the presence of dark spots on its surface . [4, 5]. Galileo notably argued that these spots were on the Sun itself, challenging the prevailing Aristotelian belief in the perfection and immutability of celestial bodies.

Systematic observations expanded in the 18th and 19th centuries. In 1843, Samuel Heinrich Schwabe discovered the roughly 11-year cycle of sunspot activity, [6], a finding that became fundamental to the study of solar behavior. Building on this, Rudolf Wolf introduced a quantitative measure of sunspot activity, now known as the Wolf Sunspot Number, allowing for the

statistical study of solar variability over long periods . [7].

Advancements continued with the emergence of solar spectroscopy. In 1908, George Ellery Hale detected strong magnetic fields within sunspots, establishing a direct connection between sunspots and solar magnetism . [8]. His discovery demonstrated that sunspots are regions of concentrated magnetic activity where the surface temperature is lower than that of the surrounding photosphere, leading to their darker appearance.

Today, sunspots are understood as key indicators of solar magnetic dynamics and variability. Modern observations, using both ground-based telescopes and space missions, have significantly enhanced our knowledge of sunspot formation, evolution, and their impact on space weather and Earth's environment.

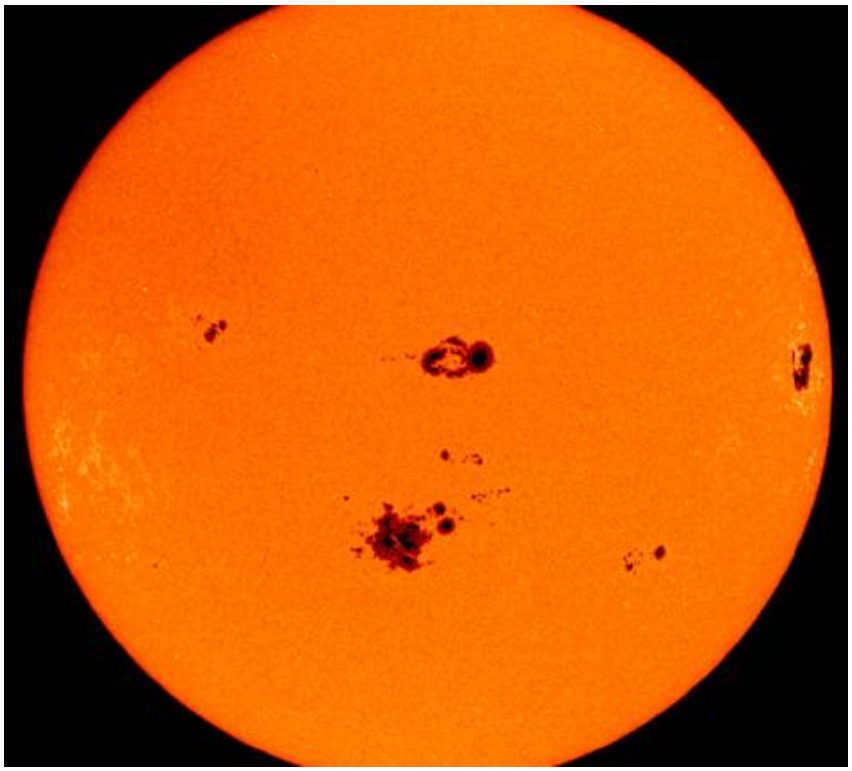

Figure 1.1: Sunspots on the surface of the sun.Source: NOAA Weather Service

1.2 Solar Activity and Sunspots

Solar activity, driven by the complex magnetic processes within the Sun, exhibits variability across a range of timescales. One of the most well-known manifestations of solar activity is the appearance of sunspots, which are temporary regions of reduced surface temperature caused by magnetic field flux. The number of sunspots observed on the solar surface fluctuates in a quasi-periodic manner, most prominently following an approximately 11-year solar cycle .[9] Monitoring and understanding sunspot patterns is crucial, as solar activity influences space

weather, which can impact satellite operations, communication systems, and terrestrial infrastructure [10].

1.3 Importance of Studying and Analyzing Sunspots

Understanding and predicting the behavior of sunspots is vital for forecasting solar activity and mitigating its effects on modern technological systems. Sunspot records serve as crucial proxies for solar irradiance variations and play an important role in long-term climate studies. Moreover, sunspots are key to studying the solar dynamo and internal magnetic field processes, offering insights into the mechanisms driving solar variability. Active sunspot regions often give rise to solar flares and coronal mass ejections, which can severely impact satellites, power grids, and communication infrastructures on Earth. Historically, periods of low sunspot activity, such as the Maunder Minimum, have been linked to significant climatic events like the Little Ice Age, highlighting the Sun's influence on terrestrial climate systems. Consequently, developing robust methods to analyze sunspot data is essential for advancing space weather forecasting, safeguarding technology, and deepening our understanding of solar physics.

1.4 Contemporary Research and Observations

Modern telescopes like the *Daniel K. Inouye Solar Telescope (DKIST)* and satellites like *SOHO* and *SDO* have provided high-resolution observations of sunspot morphology and dynamics. [11] These instruments, combined with advanced computational models and machine learning approaches, have significantly enhanced sunspot and solar cycle prediction accuracy.[12].

Sunspot observations also guide comparative studies of starspots on other solar-type stars, extending our understanding of stellar magnetism. [13].

1.5 Challenges in Sunspot Time Series Analysis

The temporal behavior of sunspots forms a classic example of a non-stationary time series, characterized by underlying trends, periodic components (seasonality), and random fluctuations. Traditional statistical models often struggle with non-stationary data unless preprocessing steps are employed. Effective techniques to remove trends and isolate periodic behavior are thus crucial to enhance the reliability of subsequent modeling and prediction efforts.

1.6 Solar Energetic Particles

Solar Energetic Particles (SEPs) are high-energy charged particles, predominantly protons, electrons, and heavier ions, that originate from the Sun during solar eruptive events. These particles are primarily accelerated through two key mechanisms: magnetic reconnection in solar flares and shock waves driven by coronal mass ejections (CMEs). Once accelerated, SEPs can travel through the interplanetary medium, sometimes reaching Earth and impacting space-borne and ground-based technologies.

Understanding SEPs is crucial for several reasons. First, their sudden and intense fluxes can pose serious radiation hazards to astronauts and spacecraft electronics. Second, SEPs offer insights into particle acceleration and transport processes in astrophysical plasmas. Moreover, their study contributes to space weather forecasting, helping to mitigate the potential risks posed by solar storms to satellites, aviation, and power grids.

The temporal and spatial distribution of SEPs is influenced by a variety of factors, including the structure of the solar magnetic field, the dynamics of the solar wind, and the characteristics of the interplanetary medium. As such, analyzing SEP events involves an interdisciplinary approach, combining observations from space missions, theoretical modeling, and statistical time series analysis.

1.7 Time Series Analysis :

Time Series Analysis is a powerful statistical technique focused on understanding data points that are indexed in time order. Unlike traditional statistical methods, which often assume independence among observations, time series methods account for the inherent temporal dependence between successive data points . A typical time series can be decomposed into several components, including a long-term trend, seasonal variations, cyclical patterns, and random noise. Analyzing these components separately helps to build accurate predictive models and uncover underlying mechanisms driving the observed data. Methods such as autocorrelation analysis, Fourier transforms, and decomposition techniques are often used to extract meaningful insights. In astrophysics, time series analysis finds extensive application, particularly in studying phenomena that exhibit periodic or quasi-periodic behavior over time . In this project, the focus is on analyzing sunspot activity recorded over several decades. By applying time series techniques, we aim to characterize patterns in solar activity, remove trends and seasonal effects, and develop models capable of forecasting future sunspot numbers. Understanding these patterns not only enriches our knowledge of solar physics but also has broader implications for space weather prediction, satellite operation planning, and the study of solar-terrestrial interactions.

Time series Components:

In time series analysis, any dataset observed over time can be decomposed into four fundamental components:

- Trend
- Seasonal
- Cyclic
- Irregular

Trend:

The **trend** component reflects the long-term movement in the data, indicating a general increase, decrease, or stability over an extended period.

Seasonal:

The **seasonal** component captures patterns that repeat at regular intervals due to systematic calendar-related effects, such as monthly, quarterly, or yearly variations.

Cyclic:

The **cyclic** component represents fluctuations that occur over longer, variable periods, typically influenced by economic, environmental, or natural cycles, and unlike seasonality, their durations are not fixed.

Irregular:

Finally, the **irregular** component, often termed the residual or noise, consists of random, unpredictable variations that cannot be attributed to the other components. Understanding and separating these components allows for more accurate modeling, forecasting, and interpretation of complex time-dependent data.

1.8 Scope of This Study

In this study, we focus on preprocessing sunspot number data by first removing the long-term trend and then modeling the seasonal component. The dataset utilized in this work is sourced from the SILSO (Sunspot Index and Long-term Solar Observations) database, which provides high-quality historical records of sunspot numbers [14]. Our objective is to develop a clear and

interpretable methodology that captures the essential periodic features of solar activity, laying the groundwork for more advanced predictive models in future research.

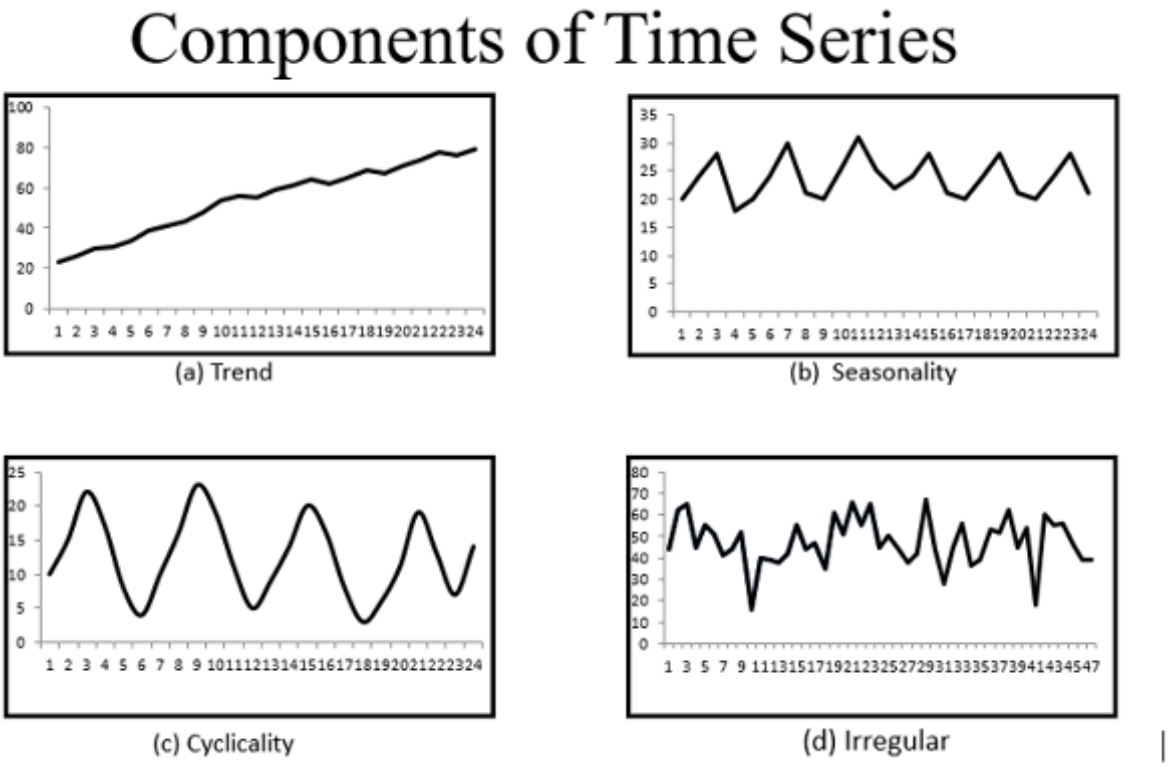

Figure 1.2: Components of Time Series. Source: Analytics Vidhya

Chapter 2

Theoretical Background

2.1 Formation and Nature of Sunspots

Sunspots are temporary phenomena on the Sun's photosphere that appear as dark patches compared to the surrounding regions. Their darkness is not due to an absence of light, but because they are significantly cooler than the surrounding photospheric material . [15]. While the typical temperature of the photosphere is about 5,800 K, the core of a sunspot may be as cool as 3,800 K.

The formation of sunspots is tied to the Sun's magnetic field. Beneath the surface, differential rotation causes magnetic field lines to become twisted and tangled. When the magnetic field becomes highly concentrated, it inhibits convective heat transport from the solar interior to the surface, causing localized cooling and sunspot formation . [16, 17].

Sunspots typically appear in pairs or groups with opposite magnetic polarities. These bipolar regions represent the two ends of a magnetic loop extending into the corona. Magnetograms reveal that sunspot regions are among the most magnetically active areas on the Sun .[18].

2.2 Physics Behind the Formation of Sunspots

Sunspots are formed due to the interaction between plasma flows in the solar convection zone and magnetic fields generated by the solar dynamo. The physics can be described using the framework of **magnetohydrodynamics (MHD)**.

2.2.1 Magnetic Pressure and Inhibition of Convection

Sunspots appear dark because the strong magnetic fields inhibit convective motion, reducing the energy transported from the solar interior to the surface. The pressure in a magnetized plasma is given by:

$$P_{\text{tot}} = P_{\text{gas}} + \frac{B^2}{2\mu_0} \tag{2.1}$$

where:

- P_{tot} is the total pressure,
- P_{gas} is the gas (thermal) pressure,
- B is the magnetic field strength,
- μ_0 is the permeability of free space.

In a region with strong magnetic field (like a sunspot), the magnetic pressure increases, requiring the gas pressure to decrease to maintain equilibrium. This causes a drop in temperature, hence the darker appearance.

2.2.2 Magnetohydrostatic Equilibrium

The vertical force balance in the solar atmosphere under magnetic fields can be written as:

$$-\nabla P + \rho \vec{g} + \frac{1}{\mu_0}(\nabla \times \vec{B}) \times \vec{B} = 0 \tag{2.2}$$

Here,

- ∇P is the pressure gradient,
- $\rho \vec{g}$ is the gravitational force,
- $\frac{1}{\mu_0}(\nabla \times \vec{B}) \times \vec{B}$ is the Lorentz force.

This equation describes how magnetic fields modify hydrostatic equilibrium, leading to sunspot structure.

2.2.3 Magnetic Buoyancy and Flux Tubes

Sunspots arise from magnetic flux tubes that are generated at the base of the convection zone. These tubes become buoyant and rise due to the Parker instability .[16].

The buoyant force per unit volume on a magnetic flux tube is:

$$F_b = -\nabla P_{\text{ext}} + \nabla P_{\text{int}} = g(\rho_{\text{ext}} - \rho_{\text{int}}) \tag{2.3}$$

The condition for rise is:

$$\rho_{\text{int}} < \rho_{\text{ext}}$$

which can occur because the magnetic field reduces the internal gas pressure, and hence the density.

2.2.4 Induction Equation and Dynamo Action

The evolution of magnetic fields in a conducting plasma is governed by the MHD induction equation:

$$\frac{\partial \vec{B}}{\partial t} = \nabla \times (\vec{v} \times \vec{B}) + \eta \nabla^2 \vec{B} \tag{2.4}$$

where:

- $\vec{v}$ is the plasma velocity,
- η is the magnetic diffusivity.

The first term represents the generation and advection of magnetic fields (from differential rotation and convection), while the second term represents diffusion. In the solar interior, the competition between these terms gives rise to the solar dynamo and the cyclical emergence of sunspots.

2.2.5 Convective Suppression

In normal convection, the energy transport is given by:

$$F_{\text{conv}} \propto \rho c_P v_T \Delta T \tag{2.5}$$

where:

- ρ is the density,
- c_P is the specific heat at constant pressure,
- v_T is the turbulent velocity,
- ΔT is the temperature difference.

In regions of high magnetic field, v_T is suppressed, reducing F_{conv}, leading to a local temperature drop.

2.3 Structure of Sunspots

A sunspot consists of two main regions: the *umbra* and the *penumbra*. The umbra is the central, darkest region where the magnetic field is strongest and nearly vertical. Surrounding it is the penumbra, a lighter, filamentary region where the magnetic field is weaker and more inclined . Figure2.1. [15].

Advanced observations, especially from instruments aboard the Solar Dynamics Observatory (SDO) and Hinode satellite, have shown sunspots to be complex and dynamic. Their magnetic field structures evolve over time and are often associated with explosive events such as solar flares and coronal mass ejections . [19, 20].

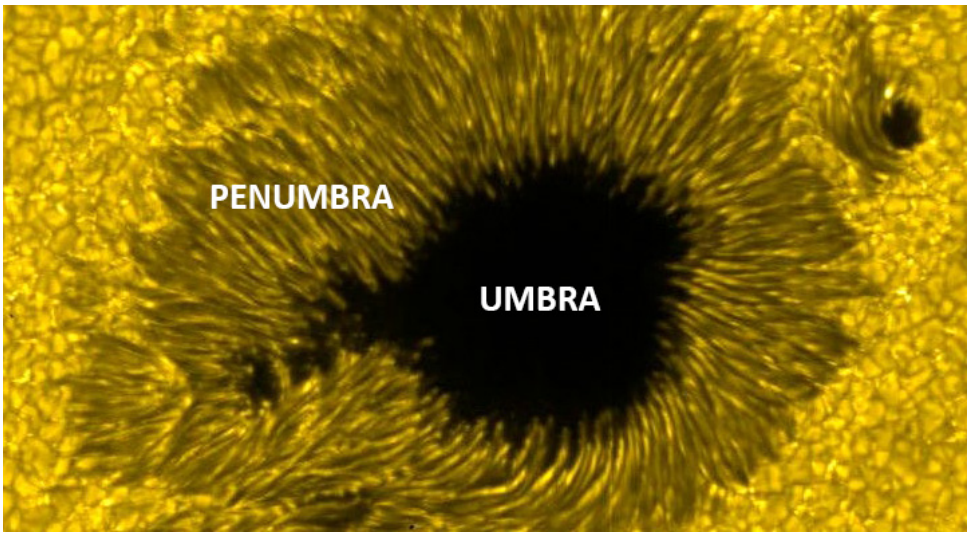

Figure 2.1: Sunspot Structure.Source: JAXA

2.4 The Solar Cycle and Sunspot Activity

Sunspots exhibit an approximately 11-year cycle of activity known as the **solar cycle**, which corresponds to the periodic reversal of the Sun's magnetic field .[9]. Sunspots initially emerge at high latitudes and migrate toward the equator as the cycle progresses, following **Spörer's Law**.

Sunspot numbers rise to a **solar maximum** and then decline to a minimum. This behavior is quantified using the **Wolf sunspot number**, a long-standing metric combining both spot groups and individual spots . [21].

The magnetic polarity of sunspot pairs also follows **Hale's Law**, reversing every 11 years, resulting in a full 22-year magnetic cycle . [22] These patterns are essential indicators of the Sun's magnetic dynamics.

2.5 Theoretical Background of Solar Energetic Particles

Solar Energetic Particles (SEPs) are highly energetic ions and electrons emitted by the Sun during transient solar phenomena such as solar flares and coronal mass ejections (CMEs). These events release large amounts of energy, accelerating particles to near-relativistic speeds. The

study of SEPs is fundamental to understanding solar-terrestrial interactions, particle acceleration in astrophysical plasmas, and the dynamics of the heliosphere.

Two principal mechanisms are widely accepted for SEP acceleration: **magnetic reconnection** in solar flares and **diffusive shock acceleration** at CME-driven shocks [23, 24]. Flare-associated SEPs are generally characterized by rapid onset times and are confined near the solar surface, while CME-associated events can persist for several days and propagate across broad heliospheric regions.

The propagation of SEPs through the interplanetary medium is governed by the structure of the solar magnetic field, particularly the Parker spiral [25], and by interactions with magnetic turbulence [26]. Particle transport involves processes such as pitch-angle scattering, adiabatic deceleration, and magnetic mirroring. These effects shape the observed intensity-time profiles and energy spectra at various locations in the heliosphere.

Understanding SEP dynamics is not only a matter of scientific interest but also has practical implications for space weather forecasting and astronaut safety. The radiation environment created by SEPs can significantly impact spacecraft systems and human activities in space [27].

Modern missions such as SOHO, STEREO, Parker Solar Probe, and Aditya-L1 provide in-situ measurements of SEPs, offering valuable insights into the acceleration sites, composition, and transport properties of these particles [28, 29].

Chapter 3

Methodology

3.1 Data Collection

The primary data used in this study is sourced from the **Sunspot Index and Long-term Solar Observations (SILSO)**, maintained by the Royal Observatory of Belgium. The dataset includes monthly and yearly averaged sunspot numbers, covering multiple solar cycles. These data serve as a quantitative measure of solar activity and provide a basis for statistical modeling and time series forecasting.

In addition to the sunspot data, this study incorporates Solar Energetic Particle (SEP) data obtained from NASA's OMNIWeb database. This dataset provides in-situ measurements of high-energy protons and ions detected near Earth, compiled from multiple spacecraft including ACE, WIND, and others. The OMNI data includes parameters such as proton fluxes across various energy channels, typically recorded on an hourly or daily basis. These measurements offer valuable insight into transient solar events and particle acceleration processes, enabling a comparative analysis with sunspot activity and supporting advanced time series modeling of solar phenomena.

3.2 Data Preprocessing

Before applying any models, the raw data undergoes several preprocessing steps to ensure its suitability for time series analysis:

- **Stationarity Check:** The Augmented Dickey-Fuller (ADF) test is performed to check for stationarity, a key assumption for many time series models.
- **Trend Removal:** Differencing is employed to eliminate long-term trends, enhancing the stationarity of the time series.

3.3 Time Series Decomposition

To better understand the underlying structure of the sunspot data, classical decomposition techniques are used to separate the series into:

- **Trend Component:** Captures long-term upward or downward movement.
- **Seasonal Component:** Captures the repeating patterns due to solar cycles.
- **Residual Component:** Represents the noise or irregular fluctuations after removing trend and seasonality.

This decomposition allows for targeted modeling of individual components and improves the overall accuracy of forecasts.

3.4 Forecasting Models:

To capture both linear and non-linear patterns in sunspot activity, statistical and machine learning models was employed. The models used are briefly described below:

- Statistical Model(ARIMA and SARIMA)
- Machine Learning Model(Random Forest and XGBoost)
- Deep Learning Model(LSTM)

3.4.1 ARIMA and SARIMA models:

ARIMA model:

ARIMA stands for Autoregressive Integrated Moving Average. It is a statistical model for time series analysis. It is the combination of auto-regressive (AR), moving average(MA), and Integrated(I) components to model temporal dependencies and ensure stationarity in the data [30] .

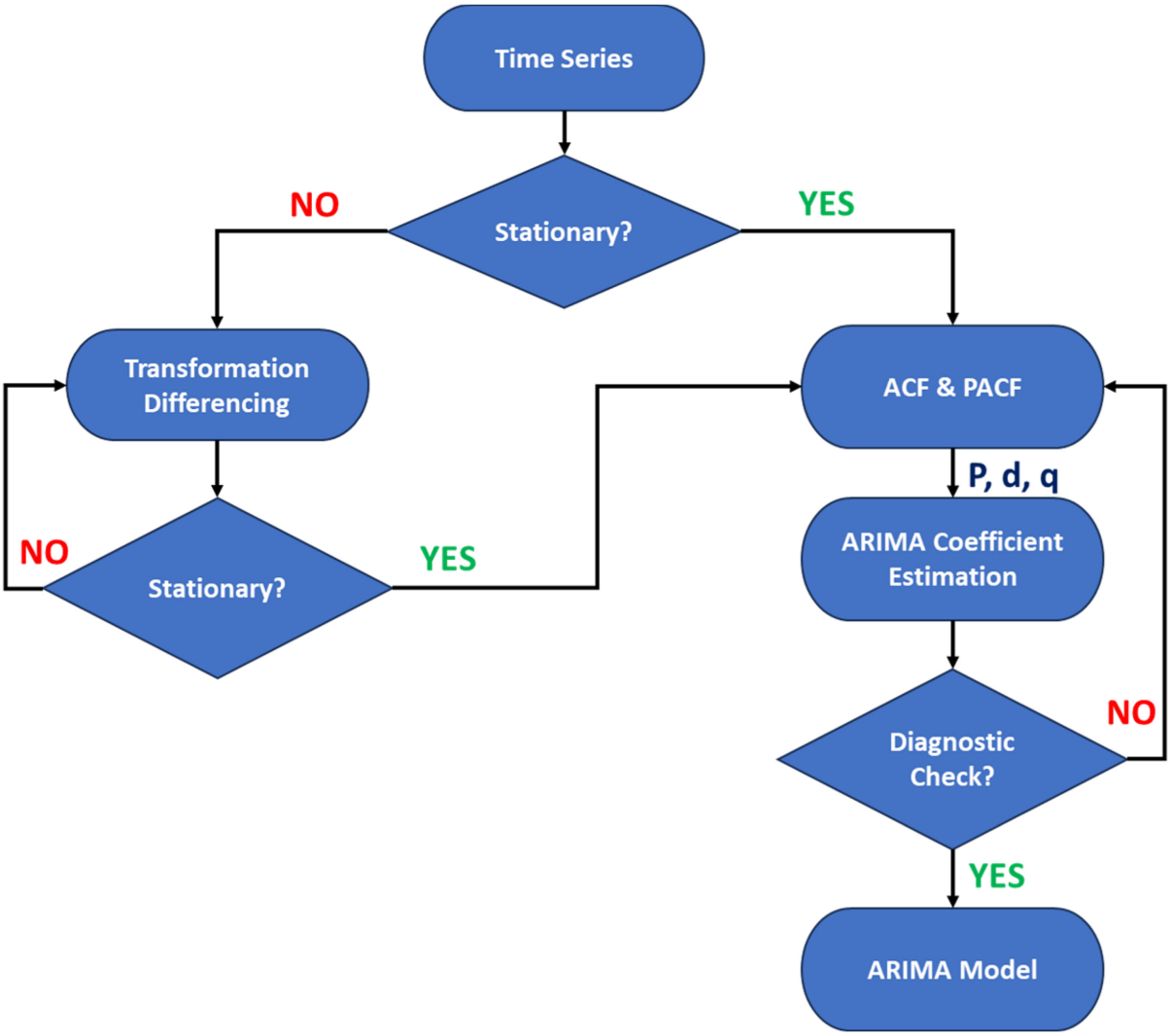

Figure 3.1: ARIMA Flowchart. Source: Sage Journals

Autoregressive (AR):

The AR part predicts the current value of a time series by looking at its past values. Essentially, it tries to explain how the past behavior of the series influences its current behavior.

Mathematically it is given by,

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \cdots + \phi_p X_{t-p} + \varepsilon_t$$

where:

- X_t: Current value of the time series.
- ϕ_i: Autoregressive coefficients.
- ε_t: White noise (random error term).

Integrated (I) or Differencing:

Differencing part makes the time series stationary by removing trends or seasonal effects.

First order differencing:

$$Y_t = X_t - X_{t-1}$$

Moving Average (MA):

The MA part explains the current value of the series by looking at how past random errors have influenced it. Mathematically,

$$X_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}$$

where:

- ε_t: Current error term (white noise).
- θ_i: Moving average (MA) coefficients.

The parameters of ARIMA model are p,q and d which represents AR order, Differencing order and MA order respectively

SARIMA model:

Seasonal AutoRegressive Integrated Moving Average SARIMA extends the ARIMA model by adding seasonal terms to account for periodic trends in the data. The model is specified by two parts: the non-seasonal (ARIMA) part and the seasonal (SARIMA) part. The general form of the SARIMA model is expressed as:

$$\Phi_p(B) \cdot (1 - B^s)^D \cdot Y_t = \Theta_q(B) \cdot \varepsilon_t$$

Where:

- Y_t is the observed time series at time t,
- $\Phi_p(B)$ is the seasonal autoregressive part (SAR),
- $(1 - B^s)^D$ represents the seasonal differencing operation (to handle periodicity),
- $\Theta_q(B)$ is the seasonal moving average part (SMA),
- ε_t is the error term at time t,
- B is the backshift operator,

- s is the number of periods in a season (e.g., 12 for monthly data with yearly seasonality),
- p, d, q are the orders of the autoregressive, differencing, and moving average terms, respectively, for the non-seasonal part of the model,
- P, D, Q are the corresponding seasonal orders of the AR, differencing, and MA terms, respectively.

  The SARIMA model is typically denoted as SARIMA$(p,d,q)(P,D,Q,s)$.

  where, p, d, q are the non-seasonal parameters for autoregressive, differencing, and moving average terms, P, D, Q are the seasonal parameters for autoregressive, differencing, and moving average terms, s represents the length of the seasonal cycle.

The seasonal differencing term, $(1-B^s)^D$, removes the seasonal patterns in the data, making the series stationary.

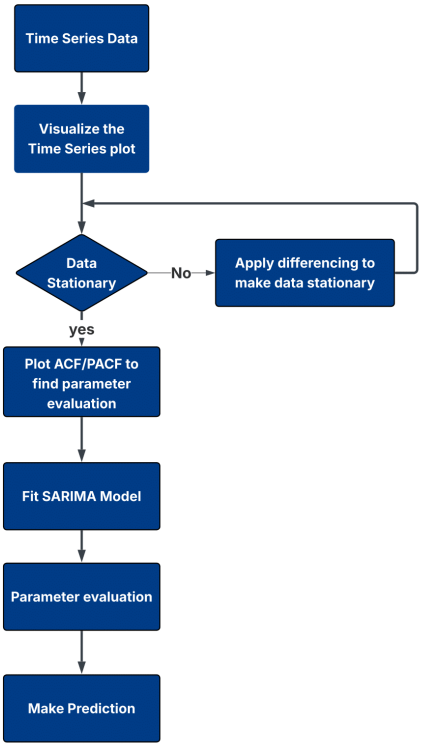

Figure 3.2: Flowchart of SARIMA

Seasonality in SARIMA

The core strength of SARIMA lies in its ability to model seasonality. Seasonality refers to patterns that repeat at fixed intervals, such as annual, monthly, or weekly cycles. For example,

in the case of sunspot activity, solar cycles repeat approximately every 11 years, which can be modeled as a seasonal component in the SARIMA model.

SARIMA accounts for seasonality through its seasonal autoregressive (SAR) and seasonal moving average (SMA) components. The seasonal autoregressive term captures the relationship between a current observation and previous observations separated by a seasonal period, while the seasonal moving average term captures the seasonal error patterns.

Parameter Estimation and Model Fitting

SARIMA model parameters (p,d,q,P,D,Q,s) are estimated using methods such as maximum likelihood estimation or conditional sum of squares. To select the optimal values for the model's parameters, diagnostic tools like Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots are used to identify the appropriate lags for the AR and MA components, both for the non-seasonal and seasonal parts of the model.

The Box-Jenkins methodology is often employed to fit SARIMA models, which involves: 1. Identifying the appropriate model by examining the ACF and PACF plots, 2. Estimating the parameters using an iterative method (e.g., maximum likelihood), 3. Diagnosing the residuals to ensure that the model is correctly specified and no further patterns remain in the residual errors.

Forecasting with SARIMA

Once the SARIMA model is fitted, forecasting future values of the time series can be performed. The forecast is generated by using the model to predict the future values based on the past observations and the estimated parameters.

The forecasted value $\hat{Y}_{t+h}$ for h-steps ahead is given by:

$$\hat{Y}_{t+h} = \Phi_p(B) \cdot (1 - B^s)^D \cdot Y_t + \Theta_q(B) \cdot \varepsilon_t$$

Where:

- h is the forecast horizon,
- $\hat{Y}_{t+h}$ is the predicted value for h-steps ahead.

The other terms represent the same components as described earlier.

Forecasting with SARIMA can account for both short-term fluctuations and long-term seasonal trends, making it suitable for modeling cyclic phenomena like sunspot activity.

3.4.2 Random Forest for Time Series Forecasting

Random Forest is a supervised machine learning algorithm that belongs to the family of ensemble methods. It is widely known for its ability to provide accurate predictions, particularly when dealing with datasets that are noisy, high-dimensional, or have complex interactions between variables. In the context of solar data analysis, especially for forecasting tasks involving sunspot numbers, Random Forest serves as a non-linear alternative to traditional statistical models.

In this chapter, I introduce the fundamental working of the Random Forest algorithm, its strengths in handling time series data, and how I have applied it to forecast solar sunspot activity using the SILSO dataset.

Concept and Structure of Random Forest

Basic Idea

Random Forest operates by building multiple decision trees and combining their predictions. Each tree is trained on a different subset of the data, and the final result is usually taken as the average (for regression) or the majority vote (for classification). By averaging many trees, the model reduces the risk of overfitting and improves generalization.

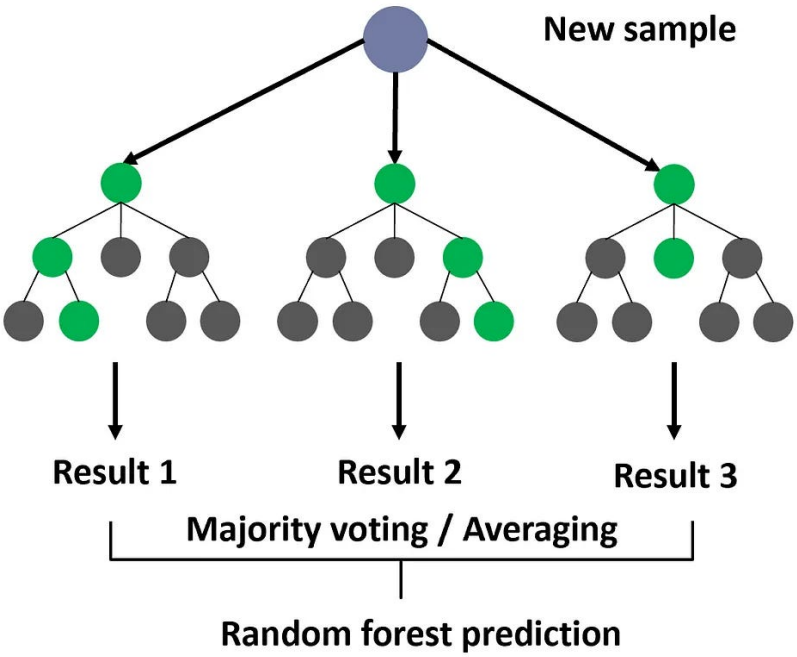

Figure 3.3: Random Forest Flowchart.Source: Medium

Bootstrap Sampling and Random Feature Selection

Each tree in the forest is trained using a random sample of the training data (with replacement), known as bootstrap sampling. Additionally, rather than considering all features at each split, a random subset of features is selected. This randomness is what gives Random Forest its name and helps it perform better than individual decision trees.

Regression with Random Forest

In this project, where the task involves predicting continuous values (e.g., monthly sunspot numbers), Random Forest is used in its regression form. It estimates future values by averaging outputs from several decision trees trained on different views of the data.

Implementation Strategy

To prepare the data for Random Forest modeling, the following steps were taken:

Preprocessing

The sunspot dataset was normalized or scaled to ensure consistent input

Feature Engineering

Lag variables were created to help the model understand temporal dependencies. For instance, the sunspot number at time t might depend on values at $t-1, t-2, \ldots, t-n$.

Model Training and Evaluation

The `RandomForestRegressor` from Python's `scikit-learn` library was trained using lagged sunspot values. The model was tested on unseen data, and performance metrics such as RMSE (Root Mean Square Error) and MAE (Mean Absolute Error) were calculated.

Strengths of the Model

- **Versatility:** Works well with both linear and nonlinear data.
- **Feature Importance:** Offers insights into which lag features contribute most to the prediction.
- **Robustness:** Performs well even with outliers or noisy data.
- **Scalability:** Can handle large datasets without a significant drop in performance.

3.4.3 XGBoost Regressor for Forecasting

XGBoost is an efficient and powerful machine learning algorithm based on the gradient boosting framework. It builds an ensemble of decision trees in a sequential manner, where each subsequent tree attempts to correct the errors made by the previous one. XGBoost has become a popular choice for time series forecasting due to its high predictive accuracy and ability to model complex, non-linear relationships in data.

How XGBoost Works

The main idea behind gradient boosting is to iteratively improve the model by minimizing a loss function L. In each iteration, a new decision tree $f_m(X)$ is added to the model to minimize the residual errors made by the previous trees. The prediction is given by the sum of all individual trees, weighted by a learning rate η. The prediction at time t can be expressed as:

$$\hat{Y}_t = \sum_{m=1}^{M} \eta \cdot f_m(X_t)$$

Where:

- $\hat{Y}_t$ is the predicted value for time t,
- M is the total number of trees in the ensemble,
- η is the learning rate (a small constant that controls the contribution of each tree),
- $f_m(X_t)$ is the output of the m-th decision tree for input features X_t at time t.

Each decision tree in the ensemble is trained to minimize the residual error, which is computed using a loss function $\mathscr{L}$. The loss function for regression tasks is typically the Mean Squared Error (MSE), defined as:

$$\mathscr{L} = \sum_{t=1}^{n} (y_t - \hat{y}_t)^2$$

Where:

- y_t is the actual observed value at time t,
- $\hat{y}_t$ is the predicted value at time t.

The key advantage of XGBoost is its ability to incorporate both first-order (gradient) and second-order (Hessian) derivatives, which makes the optimization process faster and more stable compared to traditional gradient boosting methods.

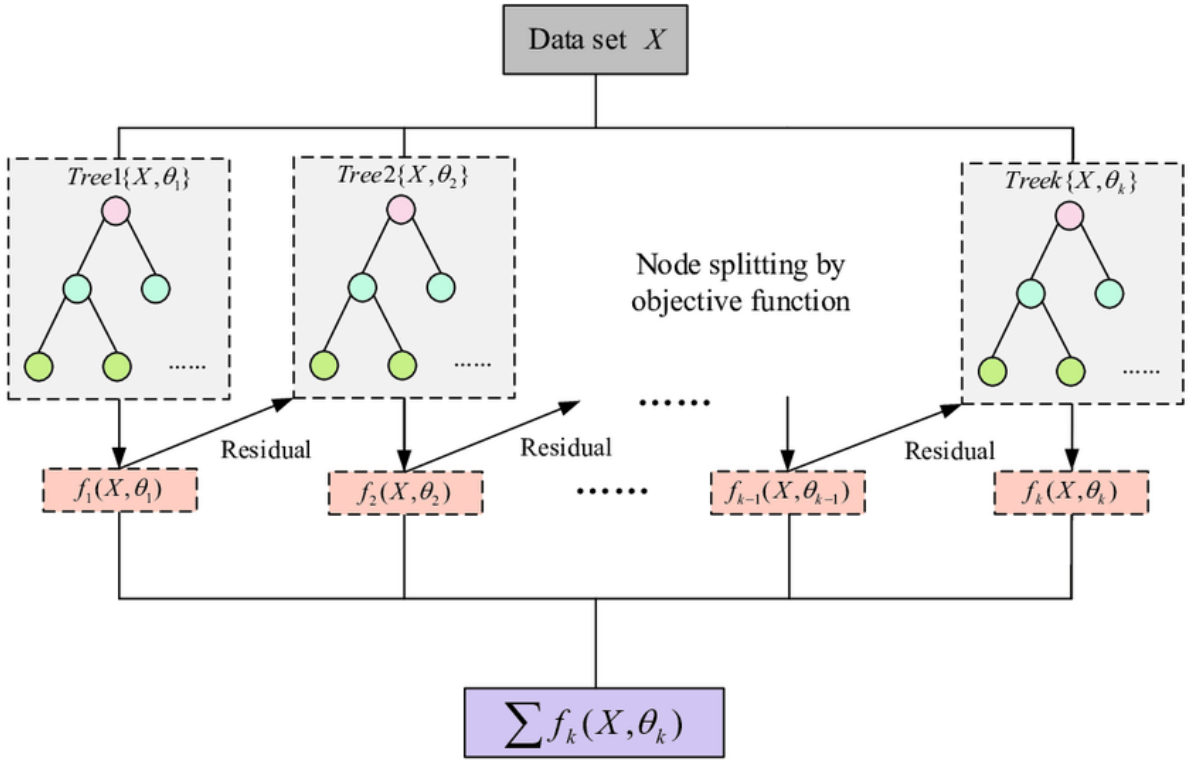

Figure 3.4: Flowchart of XGBoost. Source: [1]

Regularization in XGBoost

XGBoost includes a regularization term to penalize the complexity of the model and prevent overfitting. The objective function in XGBoost is:

$$\mathscr{L}_{\text{final}} = \sum_{t=1}^{n} \mathscr{L}(y_t, \hat{y}_t) + \sum_{m=1}^{M} \Omega(f_m)$$

Where $\Omega(f_m)$ is the regularization term that penalizes the complexity of the model:

$$\Omega(f_m) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

- $T \rightarrow$ number of leaves in the tree
- $w_j \rightarrow$ weight of the j-th leaf
- $\gamma, \lambda \rightarrow$ regularization parameters controlling tree complexity

The regularization term ensures that the trees are not too deep and are kept as simple as possible, which helps to avoid overfitting, especially when dealing with noisy or high-dimensional data.

Hyperparameter Tuning in XGBoost

To optimize the performance of the XGBoost model, hyperparameter tuning is essential. The key hyperparameters include:

- `learning_rate` (η): Controls the contribution of each tree.
- `max_depth`: Maximum depth of each tree, which controls the complexity of the individual trees.
- `n_estimators`: Number of boosting rounds or trees in the ensemble.
- `subsample`: Fraction of training data to sample for each boosting round.
- `colsample_bytree`: Fraction of features to sample for each boosting round.
- `lambda` and `alpha`: Regularization parameters to prevent overfitting.

These parameters are typically tuned using grid search or random search in combination with cross-validation to ensure optimal performance.

Feature Engineering for XGBoost

For time series forecasting, the input features to the XGBoost model are typically engineered using lagged values and moving averages. The key features include:

- Lag features: Past values of sunspot numbers, such as $Y_{t-1}, Y_{t-2}, \ldots, Y_{t-k}$.
- Rolling statistics: Moving averages or rolling windows of past sunspot values to capture long-term trends and smooth out fluctuations.
- Time-based features: For example, year, month, or day of the week, to capture potential seasonal patterns.

These features help the model learn the temporal patterns and dependencies in the data, which are crucial for accurate forecasting.

XGBoost is particularly effective for capturing non-linear relationships and complex interactions in time series data, making it a powerful tool for sunspot forecasting.

3.4.4 Long Short-Term Memory (LSTM) Networks

Long Short-Term Memory (LSTM) networks are a specialized type of recurrent neural network (RNN) designed to model sequences and their long-range dependencies. Originally proposed by Hochreiter and Schmidhuber (1997), LSTMs were developed to overcome the vanishing

and exploding gradient problems that often affect traditional RNNs during training on long sequences [31].

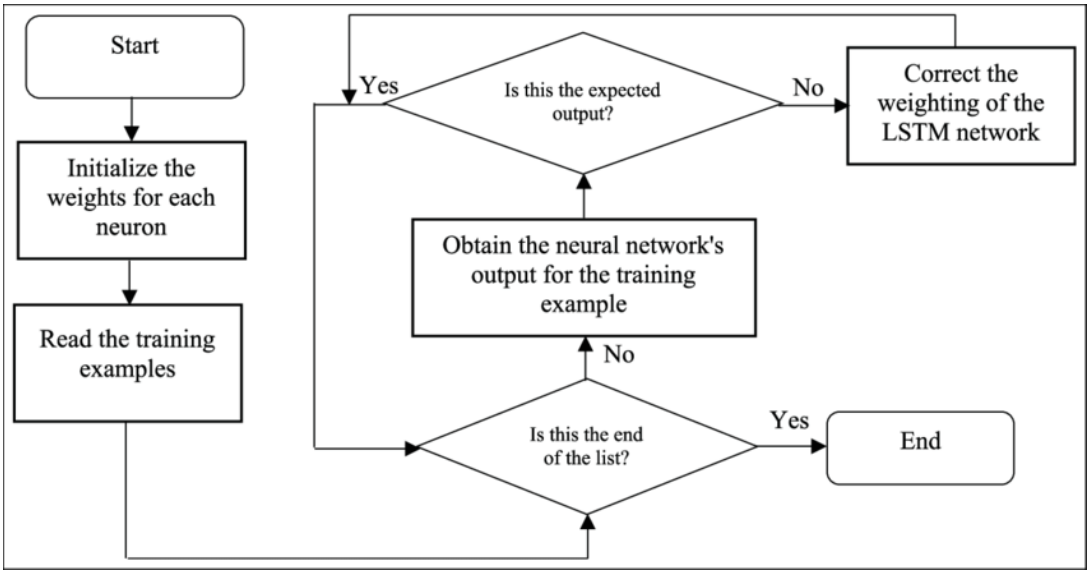

Figure 3.5: Flowchart for LSTM. Source: [2]

Structure of an LSTM Cell

Each LSTM unit consists of a memory cell and three gating mechanisms: the forget gate, the input gate, and the output gate. These gates regulate the flow of information into and out of the memory cell, enabling the network to retain relevant information over extended time intervals.

Let $\mathbf{x}_t$ be the input vector at time t, $\mathbf{h}_{t-1}$ the previous hidden state, and $\mathbf{C}_{t-1}$ the previous cell state. The mathematical operations inside a standard LSTM cell are as follows:

$$\mathbf{f}_t = \sigma(\mathbf{W}_f \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f) \tag{3.1}$$

$$\mathbf{i}_t = \sigma(\mathbf{W}_i \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i) \tag{3.2}$$

$$\tilde{\mathbf{C}}_t = \tanh(\mathbf{W}_C \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_C) \tag{3.3}$$

$$\mathbf{C}_t = \mathbf{f}_t \odot \mathbf{C}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{C}}_t \tag{3.4}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_o \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_o) \tag{3.5}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{C}_t) \tag{3.6}$$

Here, σ denotes the sigmoid activation function, tanh represents the hyperbolic tangent function, and $\odot$ indicates element-wise multiplication. The weight matrices $\mathbf{W}_*$ and bias vectors $\mathbf{b}_*$ are learned during training.

LSTM for Time Series Forecasting

In time series prediction, an LSTM network is trained to map a sequence of previous observations (lag features) to future values. For instance, given input features $X_t = [x_{t-1}, x_{t-2}, \ldots, x_{t-n}]$, the model predicts the next value y_t. During training, the loss function minimized is typically the Mean Squared Error (MSE):

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{3.7}$$

To prevent overfitting and estimate predictive uncertainty, dropout layers can be introduced into the LSTM architecture and activated during inference, following the Monte Carlo Dropout method [32].

3.5 Model Evaluation

To evaluate the performance of the forecasting models, the available dataset is divided into two subsets: a training set and a testing set. The models are trained on the training subset and subsequently assessed on the testing subset to estimate their ability to generalize to new, unseen data.

Two evaluation metrics are employed in this study:

- **Mean Absolute Error (MAE):** MAE measures the average magnitude of the errors between the predicted and the actual values, without considering their direction. It is defined as:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|, \tag{3.8}$$

  where y_i denotes the observed value, $\hat{y}_i$ is the corresponding predicted value, and n is the total number of observations. A lower MAE indicates better model performance .

- **Root Mean Square Error (RMSE):** RMSE provides a quadratic mean of the prediction errors and places a higher penalty on larger errors compared to MAE. It is given by:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}. \tag{3.9}$$

  Due to the squaring of the errors before averaging, RMSE is particularly sensitive to large deviations, making it a suitable choice when larger errors are especially undesirable.

The use of both MAE and RMSE offers a comprehensive evaluation of model accuracy by capturing different aspects of prediction errors. While MAE gives a linear score proportional to the error magnitude, RMSE highlights models that produce a few large errors, thus providing complementary perspectives on model performance.

3.6 Stationarity Testing

In time series analysis, **stationarity** refers to the property that a series' statistical characteristics, such as mean, variance, and autocorrelation, remain constant over time. Ensuring stationarity is a crucial prerequisite for many forecasting models, particularly those based on autoregressive methods like ARIMA and SARIMA . A non-stationary series can lead to misleading inferences and poor predictive performance.

To evaluate the stationarity of the sunspot number time series, both graphical and formal statistical tests were employed. Initially, visual inspection through time series plots, rolling mean, and rolling variance was conducted. While these graphical methods provide preliminary insights, they are often subjective. Therefore, formal statistical tests are necessary to rigorously assess stationarity.

3.6.1 Rolling Mean and Standard Deviation

The rolling mean and rolling standard deviation are commonly used methods for analyzing the local behavior of a time series. These methods allow for a dynamic assessment of the statistical properties of a series, such as its central tendency and variability, over a moving window of time.

The **rolling mean** is calculated by taking the average of the values within a sliding window, which moves across the time series. Mathematically, the rolling mean at time t for a window size of w is defined as:

$$\text{Rolling Mean}_t = \frac{1}{w} \sum_{i=t-w+1}^{t} x_i$$

Similarly, the **rolling standard deviation** measures the local variability of the data within the window. The formula for the rolling standard deviation at time t is given by:

$$\text{Rolling Std Dev}_t = \sqrt{\frac{1}{w} \sum_{i=t-w+1}^{t} (x_i - \text{Rolling Mean}_t)^2}$$

Where x_i represents the data points in the series, w is the size of the moving window, and t refers to the current time point.

These methods are particularly useful for identifying trends and volatility patterns in TS data. e.g, if the rolling mean shows a gradual increase or decrease, it suggests a potential trend in the series. On the other hand, if the rolling standard deviation shows significant fluctuations, it indicates periods of high variability or instability in the data, which can be important when assessing stationarity or identifying structural breaks.

The primary statistical test used in this analysis was the **Augmented Dickey-Fuller (ADF) test**. The ADF test looks for the presence of a unit root in the series, which would indicate non-stationarity. The null hypothesis (H_0) assumes that the time series has a unit root (i.e., it is non-stationary), while the alternative hypothesis (H_1) suggests stationarity. A sufficiently low p-value leads to the rejection of H_0, implying that the series is stationary .

3.6.2 Augmented Dickey-Fuller (ADF) Test

The Augmented Dickey-Fuller (ADF) test is a widely used statistical test to assess the presence of a unit root in a time series, which would indicate non-stationarity. A time series is said to be **non-stationary** if its properties change over time, and a unit root is one of the key indicators of non-stationarity. The ADF test is an extension of the Dickey-Fuller test that includes lagged differences of the series to account for higher-order serial correlation.

The null hypothesis (H_0) of the ADF test is that the time series has a unit root (i.e., the series is non-stationary), while the alternative hypothesis (H_1) asserts that the time series is stationary. The test is based on the following regression model:

$$\Delta x_t = \alpha + \beta t + \gamma x_{t-1} + \sum_{i=1}^{p} \delta_i \Delta x_{t-i} + \varepsilon_t$$

Where:

- $\Delta x_t = x_t - x_{t-1}$ is the first difference of the time series,
- t is a deterministic trend term (optional),
- x_{t-1} is the lagged value of the series,
- Δx_{t-i} are the lagged differences,
- p is the number of lags included in the model to account for autocorrelation,
- α, β, γ, and δ_i are the model coefficients, and
- ε_t is the error term.

The ADF test statistic is the value of the coefficient γ, which represents the strength of the relationship between the series and its lagged value. If γ is significantly less than zero, the null hypothesis of a unit root is rejected, indicating that the series is stationary.

The test is performed using a t-statistic for γ, and the corresponding p-value is used to determine whether to reject the null hypothesis. A low p-value (typically less than 0.05) indicates that the series is stationary.

3.7 Autocorrelation and Partial Autocorrelation Plots

3.7.1 Autocorrelation Function (ACF)

The Autocorrelation Function (ACF) is a fundamental statistical tool used to measure the degree of correlation between a time series and lagged versions of itself over successive time intervals. Specifically, it calculates the correlation coefficient between observations at time t and those at time $t-k$, where k is the lag. The ACF helps in identifying repeating patterns, trends, and the presence of seasonality in the data. It is computed as the normalized covariance between the series and its lagged counterpart, with values ranging between -1 and 1. A value close to 1 or -1 indicates strong positive or negative correlation at that lag, respectively.

The ACF plot, often called a correlogram, displays these values for different lags and is useful in diagnosing whether a Moving Average (MA) model is appropriate. For example, if the ACF cuts off after a certain lag and becomes insignificant, it suggests an MA process of that order. Moreover, significant spikes at seasonal lags can indicate the presence of seasonal effects in the time series.

Mathematical Definition of ACF

The autocorrelation at lag k, denoted by ρ_k, is given by:

$$\rho_k = \frac{\text{Cov}(X_t, X_{t-k})}{\sqrt{\text{Var}(X_t) \cdot \text{Var}(X_{t-k})}} = \frac{\mathbb{E}[(X_t - \mu)(X_{t-k} - \mu)]}{\sigma^2}$$

where X_t is the time series value at time t, μ is the mean of the series, and σ^2 is the variance.

3.7.2 Partial Autocorrelation Function (PACF)

The Partial Autocorrelation Function (PACF) serves to measure the direct correlation between a time series and its lagged values, after accounting for the influence of intervening lags. While the ACF captures both direct and indirect correlations, the PACF isolates the correlation that is not explained by shorter lags. For instance, the PACF at lag 3 reveals the correlation between x_t and x_{t-3} after removing the effects of x_{t-1} and x_{t-2}. This makes PACF particularly useful in identifying the appropriate number of autoregressive (AR) terms in an ARIMA model.

In practical terms, the PACF is computed using regression techniques that remove the influence of intermediate lags. The PACF plot typically shows a sharp drop (cutoff) at the point

where the AR order should be set. For example, if only the first two lags have significant partial autocorrelations, it suggests an AR(2) model. Like ACF, the PACF plot includes confidence intervals to judge the statistical significance of each lag.

Mathematical Definition of PACF

The partial autocorrelation at lag k, denoted by ϕ_{kk}, is defined as:

$$\phi_{kk} = \text{Corr}(X_t, X_{t-k} \mid X_{t-1}, X_{t-2}, \ldots, X_{t-k+1})$$

This represents the correlation between X_t and X_{t-k} after removing the linear effects of the intermediate lag terms.

The theoretical concepts of ACF and PACF are adapted from standard time series analysis literature [33].

3.8 Preliminary Analysis on Solar Energetic Particles:

In the initial phase of this project, time series analysis was conducted on solar energetic particle (SEP) data. However, the dataset contained a significant amount of missing values, which posed challenges for preprocessing and interpolation. Despite attempts to handle the missing data using standard interpolation techniques, the density and irregularity of the gaps limited their effectiveness.

Forecasting methods such as ARIMA and SARIMA were applied to the available cleaned portions of the dataset. While these models captured some general patterns, the overall forecasting accuracy was poor, likely due to the discontinuities in the data and the complex nature of SEP events. A few plots and error metrics from this phase are shown below:

3.8.1 Handling Missing Data with Time-Based Interpolation

In order to address missing values in the proton flux time series, we employed time-based interpolation. The dataset, which is indexed by date and time, contained gaps primarily due to observational interruptions or data collection issues. To fill these gaps, we utilized time-aware linear interpolation (‘method=’time’‘) provided by the `pandas` library in Python. This method differs from traditional linear interpolation, which interpolates based on the position of rows. Instead, time-based interpolation considers the actual time intervals between data points, ensuring that missing values are estimated in a manner consistent with the time structure of the dataset. This approach is especially suited for time series data with irregularly spaced observations, as it avoids distorting the temporal relationships and produces more accurate and

realistic imputations. As a result, time-based interpolation allowed for continuous and seamless time series modeling without introducing significant discontinuities or biases in the data.

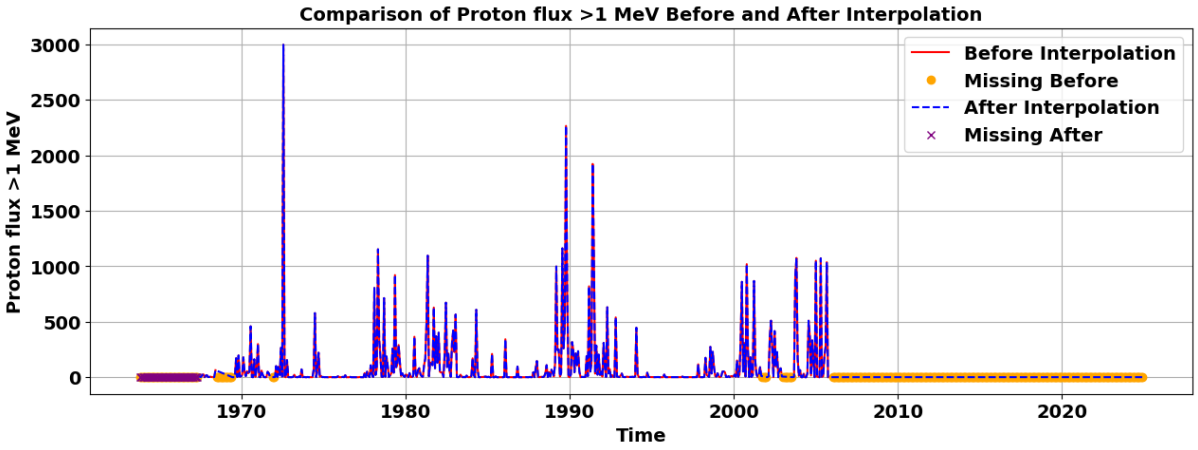

Figure 3.6: Comparison of proton flux values before and after interpolation. The missing values before interpolation are marked in red, while the missing values themselves are highlighted in orange. The interpolated values are shown as a blue dashed line, and the previously missing values after interpolation are indicated with purple 'x'

3.8.2 Prediction of Solar Energetic Particle (SEP) flux using ARIMA and SARIMA Model:

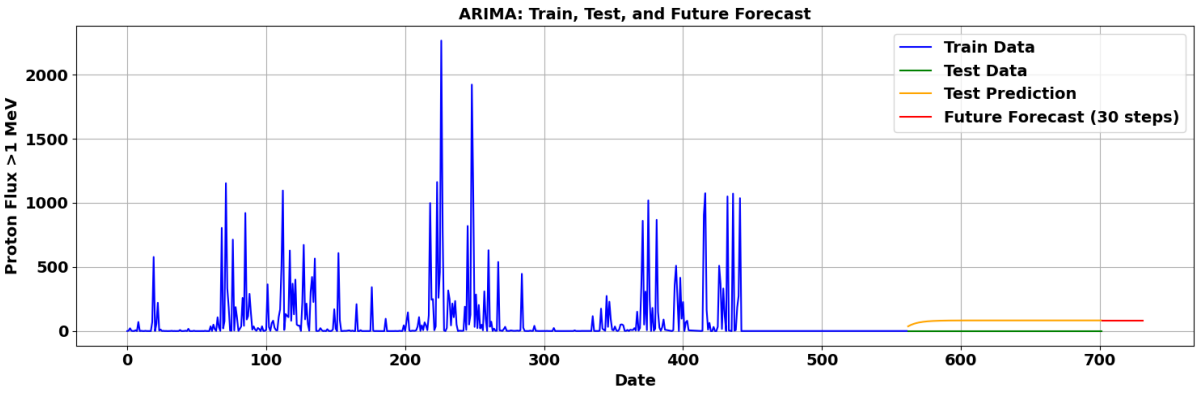

Figure 3.7: Prediction of 27-day averaged Solar Energetic Particle (SEP) flux using the ARIMA(1,0,1) model. The forecast extends 30 steps ahead, corresponding to approximately 810 days (or 2.2 years) into the future. This model captures short-term dependencies but does not account for seasonal variations.

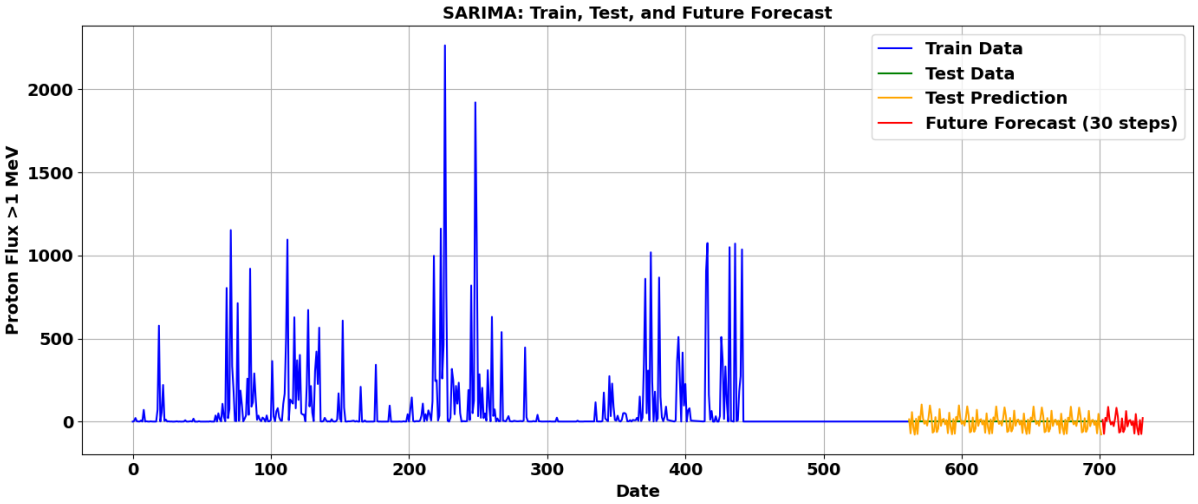

Figure 3.8: Prediction of 27-day averaged Solar Energetic Particle (SEP) flux using the SARIMA(1,0,1)(2,1,1,27) model. The forecast extends 30 steps ahead, corresponding to approximately 810 days (or 2.2 years) into the future. This model captures short-term dependencies but does not account for seasonal variations.

Chapter 4

Results & Discussion

4.1 Stationarity Testing For ARIMA and SARIMA Model:

To check the stationarity of the data, we plotted the rolling mean and rolling standard deviation of the time series using a rolling window of size 11.

4.1.1 Stationarity testing for the Sunspot Data:

In Figure 4.1. The rolling mean and rolling standard deviation is not constant over time and hence showing that our data is not stationary. In the Augmented Dickey-Fuller (ADF) test Figure 4.2. the test statistic was found to be -3.2017, with an associated p-value of 0.0199. The critical values for the ADF test at the 1%, 5%, and 10% significance levels were -3.452, -2.871, and -2.572 respectively.

Since the test statistic is less than the 5% and 10% critical values but greater than the 1% critical value, we reject the null hypothesis of a unit root at the 5% and 10% levels. This suggests that the time series is **stationary at the 5% significance level**, but not at the stricter 1% level.

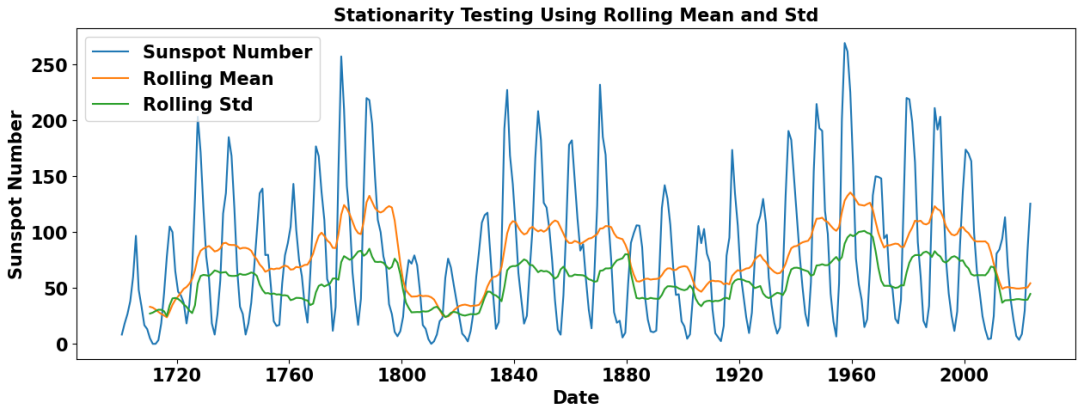

Figure 4.1: Rolling Mean and Standard deviation on the raw Sunspot data. The plot shows the original time series along with its rolling mean and standard deviation. The non-constant behavior of these statistics suggests that the series is likely non-stationary.

```
Test Statistic                  -3.201697
p-value                          0.019887
#lags used                       8.000000
number of observations used    305.000000
dtype: float64
Critical Value 1% : -3.451973573620699
Critical Value 5% : -2.8710633193086648
Critical Value 10% : -2.5718441306100512
```

Figure 4.2: ADF test on the raw Sunspot data. The plot shows the original time series along with its rolling mean and standard deviation. The non-constant behavior of these statistics suggests that the series is likely non-stationary.

4.1.2 Testing for Stationarity in Differenced Sunspot Data

In Figure 4.3, it is evident that both the rolling mean and rolling standard deviation remain relatively constant over time, suggesting potential stationarity in the time series. To confirm this, we performed the Augmented Dickey-Fuller (ADF) test. The test statistic was found to be -14.9857, with a corresponding p-value of 1.14×10^{-27}. The critical values for the test at the 1%, 5%, and 10% significance levels are -3.451, -2.871, and -2.572, respectively.

Since the test statistic is significantly lower than all the critical values, and the p-value is much smaller than the typical significance threshold of 0.05, we can decisively reject the null hypothesis of a unit root. Therefore, we conclude that the time series is **stationary**.

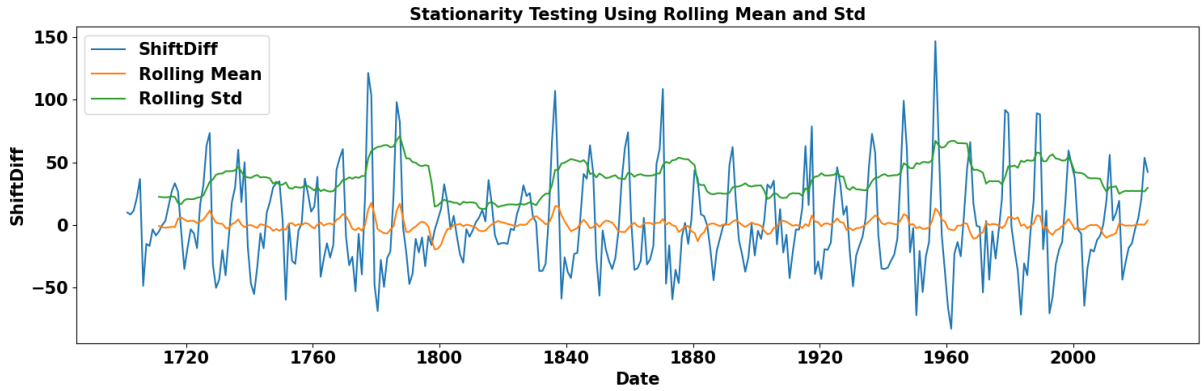

Figure 4.3: Rolling Mean and Standard deviation on the transformed Sunspot data showing stationarity. The plot displays the original series along with its rolling mean and standard deviation.

```
Test Statistic                  -1.498572e+01
p-value                          1.144661e-27
#lags used                       7.000000e+00
number of observations used      3.150000e+02
dtype: float64
Critical Value 1% : -3.451281394993741
Critical Value 5% : -2.8707595072926293
Critical Value 10% : -2.571682118921643
```

Figure 4.4: ADF test on the transformed Sunspot data showing stationarity. The plot displays the original series along with its rolling mean and standard deviation.

4.2 ACF and PACF For ARIMA:

For the ARIMA model, I applied shift differencing to remove non-seasonal trends in the dataset. This transformation stabilizes the mean by subtracting the previous value from the current one. After differencing, I used the ACF and PACF plots to identify the appropriate orders for the AR and MA components. The ACF plot helped determine the MA (q) order by showing the cut-off point, while the PACF plot guided the selection of the AR (p) order by identifying significant lags. This approach allowed the ARIMA model to effectively capture the non-seasonal patterns in the time series.

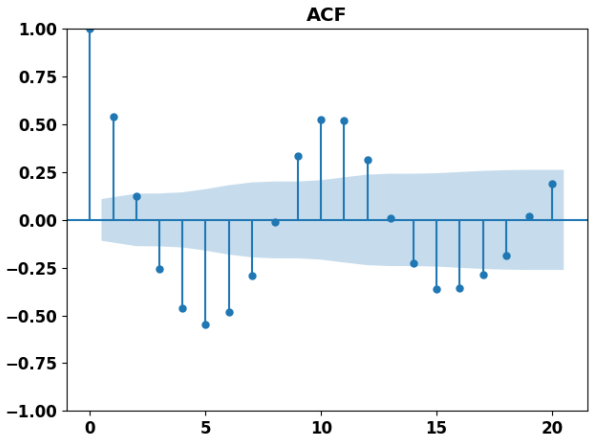

Figure 4.5: ACF plot for ARIMA model taking lag = 20

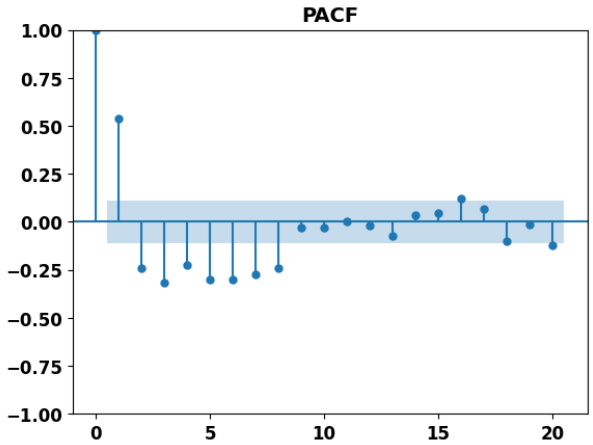

Figure 4.6: PACF plot for ARIMA model taking lag = 20

4.3 ACF and PACF for SARIMA:

In my dataset, seasonality is observed at a period of 108 time points, which suggests a yearly or other cyclical pattern depending on the data's granularity. To account for this seasonality, I applied seasonal differencing with a period of 108. This transformation removes the seasonal component from the data, allowing for more accurate modeling. After seasonal differencing, I analyzed the ACF and PACF plots to identify the seasonal and non-seasonal components of the SARIMA model.

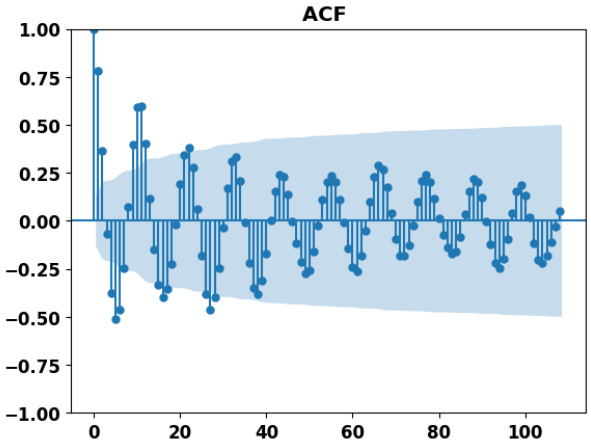

Figure 4.7: ACF plot for SARIMA model taking lag = 108

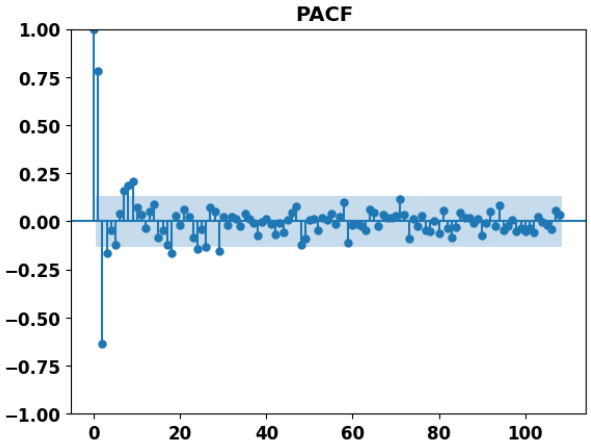

Figure 4.8: PACF plot for SARIMA model taking lag = 108

4.4 ARIMA Model:

We have forecast using ARIMA model taking parameters p = 4, d = 1 and q = 4 .

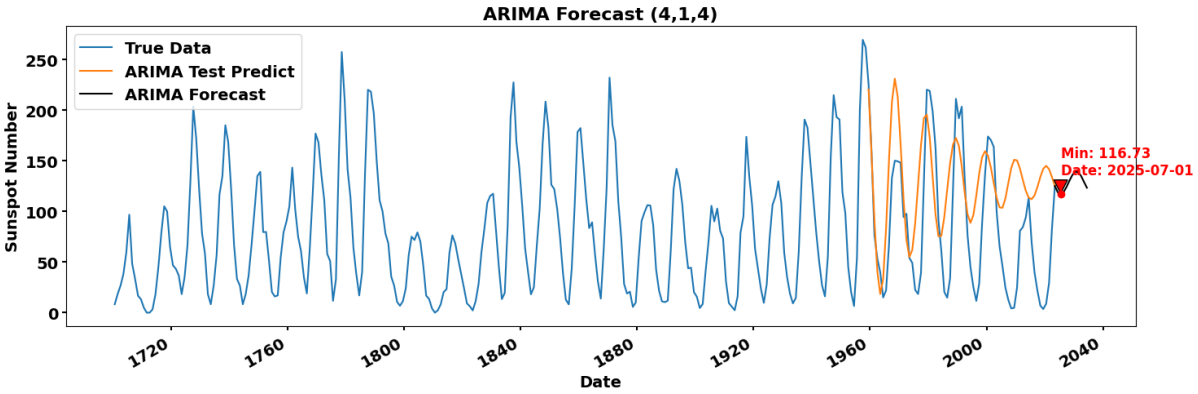

Figure 4.9: Ten-year forecast of sunspot numbers using the ARIMA(4,1,4) model. The plot highlights the predicted solar minimum, indicating the year with the lowest sunspot activity and the corresponding sunspot number.

4.5 SARIMA Model:

We have forecast using SARIMA model taking non-seasonal parameters p = 4, d = 1 and q = 4 and seasonal parameters ,P=1, D=1, Q=2 and s= 108.

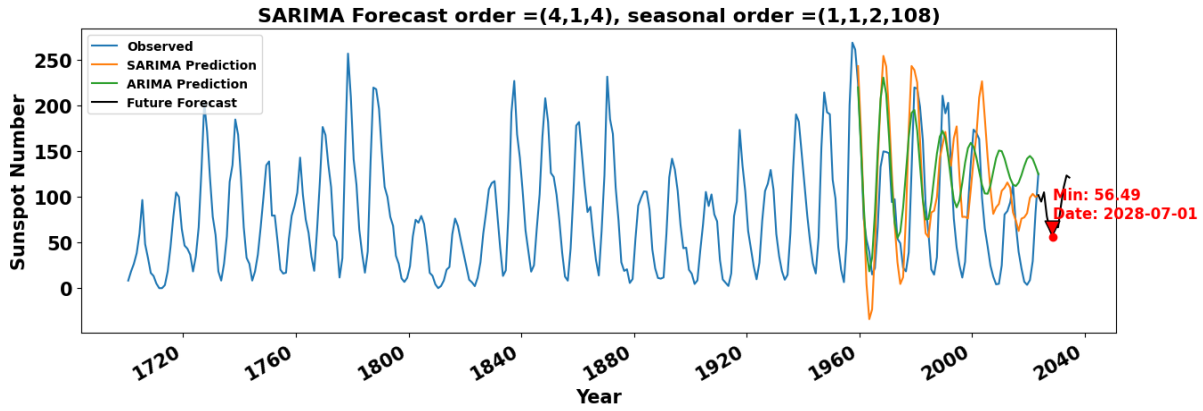

Figure 4.10: Ten-year forecast of sunspot numbers using the SARIMA(4,1,4)(1,1,2,108) model. The plot highlights the predicted solar minimum, showing the year with the lowest sunspot activity along with the corresponding sunspot number.

4.6 Random Forest Regression:

Sunspot numbers during the solar minimum of the 25th solar cycle were predicted using RF model with various lag values.

4.6.1 Random Forest Results for Lag = 7

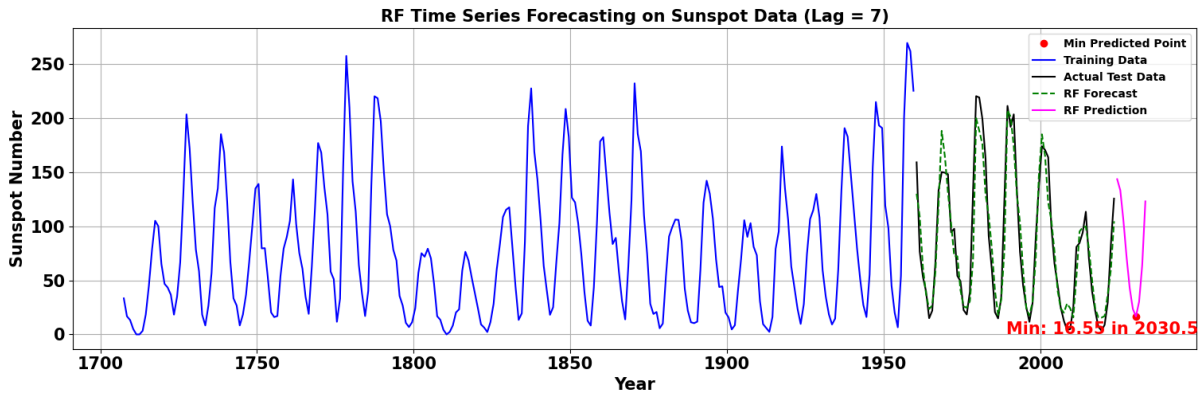

Figure 4.11: Prediction of 10 years of sunspot activity using the Rf model with a lag of 12. The plot illustrates the model's performance in forecasting sunspot activity, with a predicted sunspot number of 16.55 during the solar minima in 2030.5.

4.6.2 Random Forest Results for Lag = 12

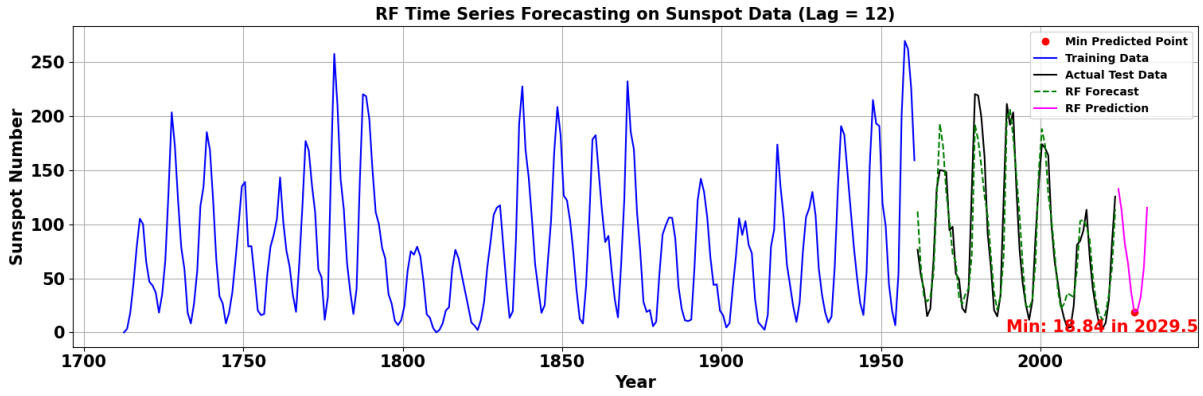

Figure 4.12: Prediction of 10 years of sunspot activity using the Rf model with a lag of 12. The plot illustrates the model's performance in forecasting sunspot activity, with a predicted sunspot number of 18.84 during the solar minima in 2029.5.

4.7 XGBoost:

Sunspot numbers during the solar minimum of the 25th solar cycle were predicted using XG-Boost model with lag 7 and 12 .

4.7.1 XGBoost Results for Lag = 7

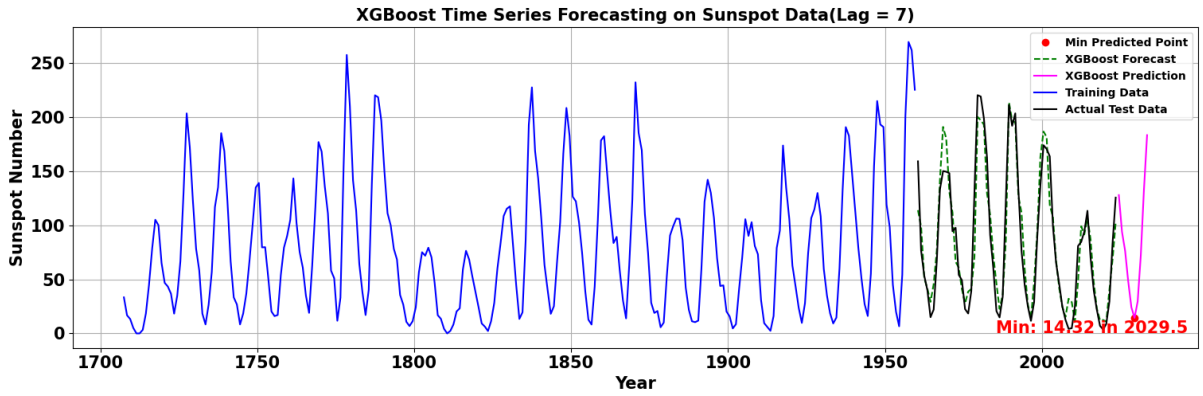

Figure 4.13: Prediction of 10 years of sunspot activity using the XGBoost model with a lag of 7.The plot illustrates the model's performance in forecasting sunspot activity, with a predicted sunspot number of 14.32 during the solar minima in 2029.5.)

4.7.2 XGBoost Results for lag = 12

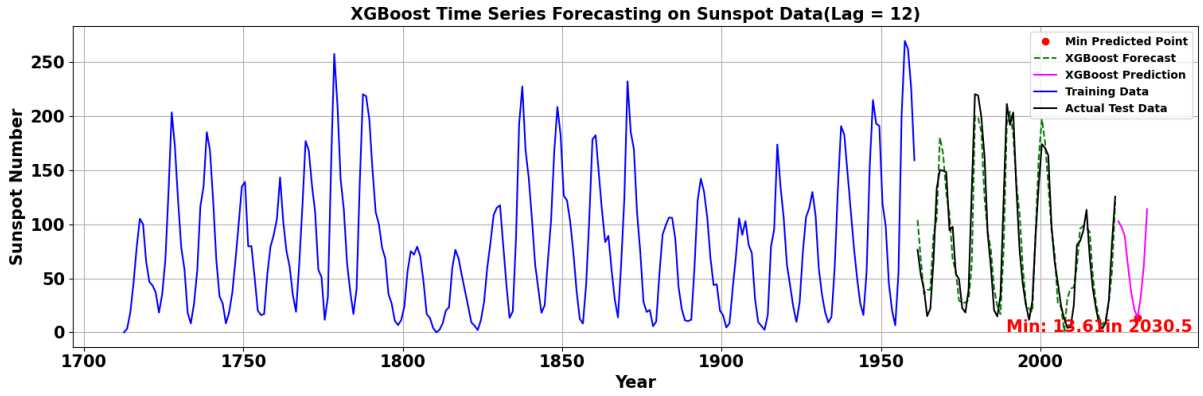

Figure 4.14: Prediction of 10 years of sunspot activity using the XGBoost model with a lag of 12. The plot illustrates the model's performance in forecasting sunspot activity, with a predicted sunspot number of 13.61 during the solar minima in 2030.5.

4.8 LSTM

4.8.1 LSTM Results for Lag = 7

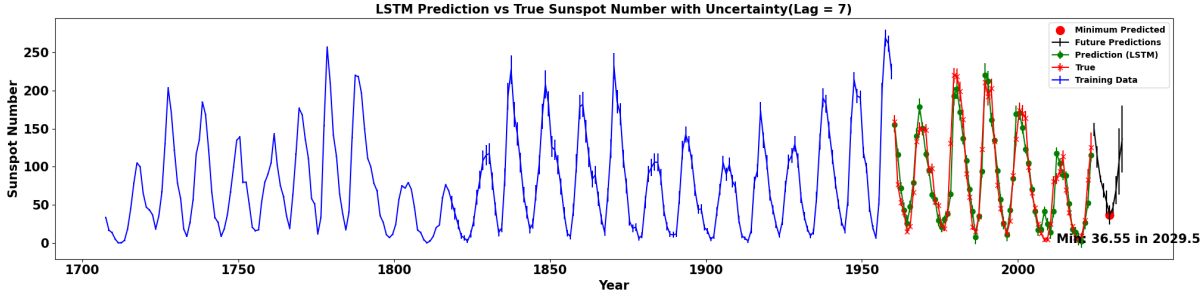

Figure 4.15: Prediction of 10 years of sunspot activity using the LSTM model with a lag of 7. The plot illustrates the model's performance in forecasting sunspot activity, with a predicted sunspot number of 36.55 during the solar minima in 2029.5.

4.8.2 LSTM Results for Lag = 12:

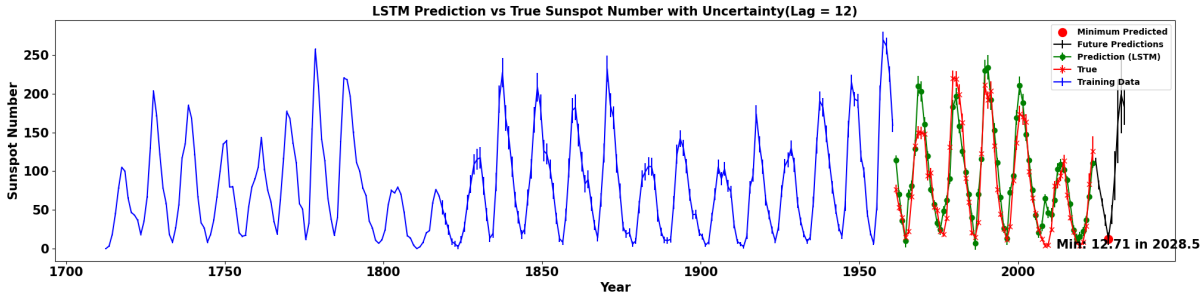

Figure 4.16: Prediction of 10 years of sunspot activity using the LSTM model with a lag of 12.The plot illustrates the model's performance in forecasting sunspot activity, with a predicted sunspot number of 12.71 during the solar minima in 2028.5.

A Random Forest Regressor is used to model and forecast sunspot numbers based on lagged historical data. The primary objective was to capture patterns in solar cycles and predict future trends, with a specific focus on identifying **solar minima**, periods of reduced solar activity.

Upon analyzing the *test dataset*, several historical solar minima were successfully identified, corresponding to well-known periods of low sunspot activity, such as those in 1986.5, 1996.5, 2008.5, and 2019.5. When comparing these with the *predicted values from the Random Forest model*, we observed general alignment in timing and pattern, although some predicted minima showed slight shifts or variations in sunspot count due to the model's smoothing tendencies.

The comparison reveals that while Random Forest captures the **broad structure and frequency of solar cycles**, it tends to **underestimate the depth of minima** and slightly **offset their timing**. This is expected behavior for ensemble models that average over many decision trees and highlights the need for integrating additional physical parameters or nonlinear models to improve temporal precision.

Table 4.1: Comparison of Local Solar Minima: Test Data vs. RF Forecast (Lag = 7)

S.No	Year (Actual Minima)	Actual Sunspot Number	Year (Predicted Minima)	Predicted Sunspot Number
1	1964.5	15.00	1964.5	23.42
2	1976.5	18.40	1976.5	24.66
3	1986.5	14.80	1986.5	17.81
4	1996.5	11.60	1996.5	17.48
5	2008.5	4.20	2007.5	20.09
6	2019.5	3.60	2019.5	15.53

Table 4.2: Comparison of Local Solar Minima: Test Data vs. RF Forecast (Lag = 12)

S.No	Year (Actual Minima)	Actual Sunspot Number	Year (Predicted Minima)	Predicted Sunspot Number
1	1964.5	15.00	1964.5	28.61
2	1976.5	18.40	1975.5	26.36
3	1986.5	14.80	1986.5	21.01
4	1996.5	11.60	1996.5	21.94
5	2008.5	4.20	2006.5	21.72
6	2019.5	3.60	2019.5	11.44

The comparison of local solar minima derived from the test data and Random Forest (RF) forecasts with different lag values reveals interesting insights. While the RF model with lag = 7 generally aligns well with the actual minima years (e.g., 1964.5, 1976.5, 1986.5, 1996.5, 2019.5), it also introduces an extra minimum at 2010.5 not present in the actual data.

In contrast, the RF model with lag = 12 tends to produce slightly higher predicted sunspot values for most minima and introduces different shifts in the timing of some events (e.g., 1975.5 instead of 1976.5 and 2006.5 instead of 2008.5). This indicates that the choice of lag in time series forecasting plays a significant role in identifying the timing and intensity of solar minima.

Overall, both lag values capture the broad trend and some known minima, but there are discrepancies in exact timing and magnitude. These results highlight the importance of optimizing lag selection for improved forecasting accuracy.

Table 4.3: Comparison of Local Solar Minima (Actual vs. XGBoost Forecast) with Lag = 7

No.	Year (Actual)	Actual Sunspot No.	Predicted Year	Predicted Sunspot No.
1	1964.5	15.00	1964.5	29.14
2	1976.5	18.40	1975.5	27.74
3	1986.5	14.80	1986.5	22.64
4	1996.5	11.60	1996.5	13.80
5	2008.5	4.20	2007.5	15.25
6	2019.5	3.60	2018.5 / 2020.5	12.28 / 10.88

To assess the accuracy of the XGBoost model in capturing solar minima, a comparison was made between the actual observed local solar minima and those predicted by the model (Table 4.3). The model was able to identify most minima within close proximity to their true occurrence in time. However, the predicted sunspot numbers during these minima were gen-

erally overestimated. For example, during the well-known minimum of 2008.5 (actual sunspot number 4.20), the model predicted a significantly higher value of 15.25.

The model also detected multiple nearby points (e.g., 2018.5 and 2020.5) around the 2019.5 minimum, indicating temporal uncertainty in pinpointing the exact minimum. Nonetheless, the predicted values still capture the declining phase of the cycle, showing the model's ability to follow the general trend of solar activity.

Despite slight offsets in timing and amplitude, the XGBoost forecast demonstrates potential for identifying periods of low solar activity, though improvements are needed to enhance amplitude accuracy during solar minima.

Table 4.4: Comparison of Local Solar Minima (Actual vs. XGBoost Forecast with Lag = 12)

No.	**Year (Actual)**	**Actual Sunspot No.**	**Predicted Year**	**Predicted Sunspot No.**
1	1964.5	15.00	1963.5	35.60
2	1976.5	18.40	1976.5	27.44
3	1986.5	14.80	1987.5	16.93
4	1996.5	11.60	1997.5	18.91
5	2008.5	4.20	2007.5	1.00
6	2019.5	3.60	2019.5	6.92

Using an XGBoost regression model with a lag of 12, I forecasted the sunspot numbers and extracted local solar minima from the predicted time series. Table 4.4 presents a comparison between the actual solar minima in the test data and those predicted by the model.

The predicted minima closely match the actual ones in terms of temporal alignment, with most years either matching exactly or being off by just one year. Notably, the 2008.5 minimum (with the lowest recorded sunspot number of 4.20) was successfully captured in 2007.5 with a predicted value of 1.00, indicating a strong dip.

However, the model tends to overestimate the sunspot numbers during most minima, except in 2007.5 where it underestimates. This discrepancy could be due to the smoothing nature of the lag-based autoregressive structure in XGBoost or the influence of preceding high-activity periods.

Overall, the XGBoost model with lag = 12 demonstrates promising ability in tracking the timing of solar minima, though amplitude predictions remain an area for further improvement.

Table 4.5: Comparison of Actual and Predicted Local Solar Minima (LSTM with Lag 7)

No.	Year (Actual)	Actual Sunspot No.	Predicted Year	Predicted Sunspot No.
1	1964.5	15.00	1964.5	25.69
2	1976.5	18.40	1975.5	20.99
3	1986.5	14.80	1986.5	8.11
4	1996.5	11.60	1996.5	10.91
5	2008.5	4.20	2006.5	17.48
6	2019.5	3.60	2020.5	2.72

The comparison of local solar minima from actual sunspot data and LSTM forecasts with a lag of 7 (see Table 4.5) reveals a generally consistent pattern, with some deviations in both the timing and amplitude of the predicted minima. The LSTM model accurately captures the year of minima for 1964.5, 1986.5, and 1996.5, but shows a shift of approximately one year earlier for the 1976.5 minimum (predicted as 1975.5). A notable deviation is observed for the 2008.5 minimum, where the model predicts an earlier minimum at 2006.5 and overestimates the sunspot number significantly (17.48 predicted vs. 4.20 actual). Similarly, the most recent minimum in 2019.5 is predicted slightly later at 2020.5, though the predicted sunspot number (2.72) is quite close to the actual value (3.60). The model performs reasonably well in identifying the timing of the minima but tends to overestimate the amplitude in certain cycles, particularly for deeper minima. These findings suggest that while the LSTM model with lag 7 captures the temporal structure of solar cycles effectively, further refinement may be needed to improve amplitude prediction.

Table 4.6: Comparison of Actual and Predicted Local Solar Minima (LSTM with Lag 12)

No.	Year (Actual)	Actual Sunspot No.	Predicted Year	Predicted Sunspot No.
1	1964.5	15.00	1964.5	10.07
2	1976.5	18.40	1975.5	24.59
3	1986.5	14.80	1986.5	6.87
4	1996.5	11.60	1996.5	12.61
5	2008.5	4.20	2006.5	20.33
6	2019.5	3.60	2018.5	12.07

The comparison of local solar minima between the actual data and predictions from the LSTM model with lag 12 (as shown in Table 4.6) highlights both temporal and amplitude-based discrepancies. The model accurately predicts the year for some minima, such as 1964.5,

1986.5, and 1996.5. However, there are noticeable shifts in other cycles, particularly for the 1976.5 minimum, which is predicted a year earlier at 1975.5. A more significant deviation is observed for the 2008.5 minimum, where the model predicts a minimum at 2006.5, two years early, and overestimates the sunspot number by a wide margin (20.33 predicted vs. 4.20 actual). Similarly, for the most recent minimum in 2019.5, the prediction is one year early and significantly overestimated (12.07 vs. 3.60). These results suggest that while the LSTM model with a lag of 12 can capture general trends and the timing of several minima, it often struggles with precise amplitude forecasting, particularly during deep solar minima. Additional tuning or hybrid models may help improve both the timing and magnitude of minima predictions.

Chapter 5

Summary

This thesis presents a comparative study of classical, machine learning, and deep learning models for forecasting sunspot activity using monthly data from the SILSO dataset. The models evaluated include ARIMA, SARIMA, Random Forest, XGBoost, and LSTM, with different lag values (7 and 12) to capture temporal dependencies.

The classical models (ARIMA and SARIMA) served as baselines and performed poorly in terms of both MAE and RMSE. Machine learning models, particularly Random Forest and XGBoost, significantly outperformed classical models, with Random Forest (lag = 7) achieving the best overall performance (MAE = 15.04, RMSE = 19.35). Deep learning with LSTM showed mixed results, requiring careful tuning and more data to improve accuracy.

All models were evaluated on their ability to predict future sunspot minima, with predicted years and sunspot numbers compared against known solar minima. Machine learning models were more consistent in capturing the timing and amplitude of the cycles, whereas LSTM's performance varied depending on the lag value.

The study concludes that ensemble-based machine learning methods are promising tools for forecasting solar activity and may serve as reliable alternatives to traditional time series models. These results have implications for space weather prediction and solar physics research.

5.0.1 Model Performance Analysis

From Table 4.7, we can draw several insights regarding the performance of different time series forecasting models based on the MAE and RMSE metrics.

- **Classical Models (ARIMA, SARIMA):** The ARIMA and SARIMA models exhibit the highest error values, with ARIMA having an MAE of 57.60 and RMSE of 70.98, and SARIMA performing slightly worse. This indicates that these classical statistical models are less capable of capturing the underlying dynamics of the data compared to other methods.

Table 5.1: Comparison of Model Performance and Solar Cycle Predictions

Model	MAE	RMSE	Sunspot Number	Solar Minimum Year
ARIMA	57.60	70.98	116.73	2025.5
SARIMA	60.67	74.36	56.59	2028.5
XGBoost (Lag = 7)	15.22	20.01	14.32	2029.5
XGBoost (Lag = 12)	16.403	20.40	13.61	2030.5
Random Forest (Lag = 7)	15.04	19.35	16.55	2030.5
Random Forest (Lag = 12)	16.82	21.21	18.84	2029.5
LSTM (Lag = 7)	15.82	20.91	36.55	2029.5
LSTM (Lag = 12)	20.39	25.25	12.71	2028.5

Comparison of model performance metrics for sunspot prediction using different time series approaches. Random Forest with lag 7 achieved the lowest MAE and RMSE, outperforming classical and deep learning models.

- **Machine Learning Models (XGBoost, Random Forest):** Both XGBoost and Random Forest significantly outperform the classical models. Among these, the **Random Forest model with Lag = 7** achieves the lowest MAE (15.04) and RMSE (19.35), making it the best-performing model in this comparison. For both XGBoost and RF, increasing the lag from 7 to 12 slightly increases the error, which may be due to overfitting or the inclusion of noisy lagged features.

- **Deep Learning Model (LSTM):** The LSTM model with Lag = 7 performs reasonably well (MAE = 15.82, RMSE = 20.91), although not better than the best-performing Random Forest model. However, the performance deteriorates when the lag is increased to 12, with errors rising to MAE = 20.39 and RMSE = 25.25. This suggests that deep learning models may require more careful tuning and more data to generalize effectively.

In summary, the results indicate that machine learning models, particularly Random Forest with a lag of 7, offer the best predictive performance for this dataset. Deep learning models like LSTM may not always outperform simpler models, especially without proper optimization.

Future Plan:

With the current models successfully predicting sunspot activity, a possible future direction could be to enhance their robustness and generalizability. This may involve fine-tuning model parameters or incorporating ensemble techniques to improve prediction stability across solar cycles. Additionally, applying these models to other solar or astrophysical time series datasets could help evaluate their versatility and potential for broader use.

Another potential direction is to compare model performance across various datasets to assess their adaptability. Such comparisons could provide insights into the suitability of these forecasting approaches for different types of solar activity data. This line of work may open up possibilities for using similar time series forecasting techniques in broader applications within solar physics, particularly in space weather prediction and related astrophysical research.

Bibliography

[1] Rui Guo, Zhiqian Zhao, Tao Wang, Guangheng Liu, Jingyi Zhao, and Dianrong Gao. Degradation state recognition of piston pump based on iceemdan and xgboost. *Applied Sciences*, 10:6593, 09 2020.

[2] Johana Hernandez, Danilo López, and Nelson Vera. Primary user characterization for cognitive radio wireless networks using long short-term memory. *International Journal of Distributed Sensor Networks*, 14:155014771881182, 11 2018.

[3] D. H. Clark, F. R. Stephenson, and R. Neuhaeuser. Sunspots, solar flares, and aurorae: the historical record. *Astronomy & Geophysics*, 48(4):4.20–4.27, 2007.

[4] Galileo Galilei. *Sidereus Nuncius*. Venice, 1610.

[5] Christoph Scheiner. *Rosa Ursina sive Sol*. Bracciano: Andreas Phaeus, 1630.

[6] Heinrich Schwabe. Sonnenbeobachtungen im jahre 1843. *Astronomische Nachrichten*, 20:495, 1843.

[7] Rudolf Wolf. Numbers of sunspots observed from 1749 to 1850. *Mémoires de la Société des Sciences Naturelles de Neuchâtel*, 1:465–490, 1850.

[8] George Ellery Hale. On the probable existence of a magnetic field in sunspots. *The Astrophysical Journal*, 28:315, 1908.

[9] David H. Hathaway. The solar cycle. *Living Reviews in Solar Physics*, 12(1):4, 2015.

[10] Carolus J. Schrijver and George L. Siscoe. *Heliophysics: Active Stars, their Astrospheres, and Impacts on Planetary Environments*. Cambridge University Press, 2015.

[11] Thomas R. Rimmele, Mark Warner, Stephen L. Keil, et al. The daniel k. inouye solar telescope—observing the sun at highest resolution. *Solar Physics*, 295(12):172, 2020.

[12] Enrico Camporeale. The challenge of machine learning in space weather: Nowcasting and forecasting. *Space Weather*, 17(8):1166–1207, 2019.

[13] Svetlana V. Berdyugina. Starspots: A key to the stellar dynamo. *Living Reviews in Solar Physics*, 2(1):8, 2005.

[14] SILSO World Data Center. International sunspot number monthly bulletin and online catalogue. http://www.sidc.be/silso/, 2023. Royal Observatory of Belgium, Brussels.

[15] Sami K. Solanki. Sunspots: An overview. *Astronomy and Astrophysics Review*, 11(2-3):153–286, 2003.

[16] Eugene N. Parker. Hydromagnetic dynamo models. *The Astrophysical Journal*, 122:293, 1955.

[17] Eric R. Priest. *Magnetohydrodynamics of the Sun*. Cambridge University Press, 2014.

[18] Carolus J. Schrijver and Cornelis Zwaan. *Solar and Stellar Magnetic Activity*. Cambridge University Press, 2000.

[19] Alexander G. Kosovichev and Valentina V. Zharkova. Magnetic energy and helicity budgets in solar flares. *Solar Physics*, 268(1):175–183, 2011.

[20] James R. Lemen, Alan M. Title, David J. Akin, Paul F. Boerner, et al. The atmospheric imaging assembly (aia) on the solar dynamics observatory (sdo). *Solar Physics*, 275(1):17–40, 2012.

[21] Frédéric Clette, Leif Svalgaard, José M. Vaquero, and Edward W. Cliver. Revisiting the sunspot number. a 400-year perspective on the solar cycle. *Space Science Reviews*, 186(1-4):35–103, 2014.

[22] George Ellery Hale, Ferdinand Ellerman, Seth B. Nicholson, and Alfred H. Joy. The magnetic polarity of sun-spots. *The Astrophysical Journal*, 49:153, 1919.

[23] Donald V. Reames. Particle acceleration at the sun and in the heliosphere. *Space Science Reviews*, 90(3-4):413–491, 1999.

[24] Mihir Desai and Joe Giacalone. Large gradual solar energetic particle events. *Living Reviews in Solar Physics*, 13(1):1–54, 2016.

[25] Eugene N. Parker. Dynamics of the interplanetary gas and magnetic fields. *Astrophysical Journal*, 128:664, 1958.

[26] JR Jokipii. Cosmic-ray propagation. i. charged particles in a random magnetic field. *Astrophysical Journal*, 146:480, 1966.

[27] Nathan A. et al. Schwadron. Earth-moon-mars radiation environment module framework. *Space Weather*, 8(10):S00E02, 2010.

[28] Angelos et al. Vourlidas. The first solar observations from the parker solar probe mission. *Nature*, 576(7786):502–505, 2019.

[29] S Seetha and S Megala. Aditya-l1 mission: India's first dedicated solar mission. *Current Science*, 118(3):362–364, 2020.

[30] Khulood Albeladi et al. Time series forecasting using lstm and arima. *International Journal of Advanced Computer Science and Applications*, 14(1), 2023. Accessed: 21 Nov. 2024.

[31] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[32] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pages 1050–1059. PMLR, 2016.

[33] Rob J Hyndman and George Athanasopoulos. *Forecasting: Principles and Practice*. OTexts, 2 edition, 2018.