
# To develop a predictive model for the energy of Solar Flares using Machine Learning and To find the correlation between Sunspot Area and related parameters

*Team SOHO*

**Internal Guide:** Mr.Sundar

**External Guide:** Dr. Parshati Patel

**Society for Space Education Research and Development (SSERD)**

# Team Members

Adhitya Shreyas SP, RA1811019010006, SRM Institute of Science and Technology, Kattankalathur

Amaria Bonsi Navis. I, 170046, Holy Cross College Nagercoil

Anisha, 18582042, Kalindi College New Delhi

Ankit Kumar Mishra, 11703983, Lovely Professional University Punjab

Apeksha Mahesh Phadte, 201810848, PES RSN College of Arts and Science Goa University, Goa

Arvindh E. Prasad, RA1811019010001, SRM Institute of Science and Technology, Kattankalathur

Athuliya A, 721017101011, Nehru institute of technology, Coimbatore

Kritika Joshi, 19MS0053, IIT(ISM) Dhanbad

Megha Madhusudhan, RV18S1183, NMKRV College for Women, Bangalore

Prateek Boga, 122017009, SASTRA University Thanjavur, Hyderabad

Priyanka Kasturia, 18567021, kalindi college, Delhi

R. Aparna, RA1811019010031, SRM Institute of Science and Technology, Chennai

Renuka Velu, 17PHY28, Bhaktavatsalam Memorial College, Chennai

# Acknowledgements

We'd like to firstly express our utmost gratitude to SSERD for giving us this unique opportunity of a research internship which has enhanced each and every one of us in both, an individualistic and collective sense due to the immense exposure, knowledge and fun we've had due in the short duration of a month. The SSERD team has been extremely helpful by providing us a user-friendly and safe platform where communication between us was made easy. The constant support we received from Ms. Nikhitha, Mr. Sujay Sreedar, Mr. Komal Kedarnath and Mr. Mahesh have been a great source of motivation and driven us to accomplish our goal. The weekly update meetings that were conducted helped us to analyse our weekly progress.

The technical talks that were conducted twice a week helped us gain knowledge and has taught us valuable lessons in terms of career growth. We are also extremely thankful to our mentors Mr. Sundar and Dr. Parshati Patel for their valuable inputs in every step of the way. Their guidance and patience with us throughout has been supremely helpful for this was a learning process for all of us. Mr. Sundar has never once hesitated to help us with our queries irrespective of how late at night it may be and we are so very thankful for that. Mr. Pavan Kumar has also been kind enough to help us whenever we had doubts and wanted to verify our proceedings. We are grateful for each and every person that has made this possible and are looking forward to make use of this whole experience to grow in our careers!

# Abstract

**We have studied the long term variation of various solar parameters and aimed at making a predictive model for solar flares using parameters that influence solar flare activity, and also to find a correlation between different parameters of Coronal Mass Ejections (CMEs). The analysis of the predictive model was performed an R-square value of 0.79 and RMSE value of 2.9653 was obtained which pertains to the fact that the model fits. A detailed correlative study has been performed using the monthly data among a variety of solar activity parameters like CME width, Number of Solar Flares, Linear Speed of CMEs, Solar Flare Index, 2nd order Initial and Final Speeds of CMEs and Total Solar Irradiance. The Correlation analysis was performed by studying the Spearman coefficient obtained by plotting and curve fitting graphs for various CME parameters with respect to Sunspot Number ranges and Magnetic Flux differences. It was concluded that the relation between the parameters and the Sunspot Area was an inverse relation where the coefficient values with the MF difference were negative and for the parameters that didn't yield a negative coefficient was an effect of magnetic flux and the anomalies observed were due to lack of data.**

# Contents


**1 Introduction** **1**

1.1 Aims and Objectives . . . 1

1.1.1 **Objective I:**To develop a predictive model for the energy of solar flares with the help of Machine Learning. . . . 1

1.1.2 **Objective II:** To find an inverse correlation between the sunspot area and solar activity. . . . 2

**2 Background & Literature Overview** **3**

2.1 Magnetic Field of the Sun . . . 3

2.2 Sunspots . . . 4

2.3 Solar Flares . . . 4

2.4 Coronal Mass Ejections . . . 5

2.5 Total Solar Irradiance (TSI) . . . 6

2.6 Machine Learning . . . 6

2.7 Curve fitting . . . 7

2.8 Scatterplot Matrix . . . 8

2.9 Ensembled Methods . . . 8

**3 Methodology & Data Collection** **10**

3.1 Objective I: To develop a predictive model for the energy of solar flares using machine learning . . . 10

3.2 Objective II: To find an inverse correlation between the sunspot area and solar activity 20

**4 Results & Discussion** **27**

4.1 Objective I: To develop a predictive model for energy of solar flares using Machine Learning . . . 27

4.2 Objective II: To find the correlation between sunspot area and related parameters . . . 35

4.2.1 Number of Solar Flares v/s SSA: . . . 35

4.2.2 Solar Flare Index v/s SSA: . . . 36

4.2.3 Total Solar Irradiance v/s SSA: . . . 37

4.2.4 CME Width v/s SSA: . . . 38

4.2.5 Linear speed v/s SSA: . . . . . 39
4.2.6 2nd order Initial Speed v/s SSA: . . . . . 40
4.2.7 2nd order Final Speed v/s SSA: . . . . . 41

**5 Conclusions** **43**
5.1 **Objective I:**To develop a predictive model for the energy of Solar Flares using Machine Learning . . . . . 43
5.2 **Objective II:**To find the correlation between Sunspot Area and related parameters . . 43
5.3 FUTURE SCOPE . . . . . 44
5.3.1 **Objective I:**To develop a predictive model for the energy of Solar Flares using Machine Learning . . . . . 44
5.3.2 **Objective II:**To find the correlation between Sunspot Area and related parameters 44

**6 References** **45**
6.1 Background & Literature survey . . . . . 45
6.2 Data Collection . . . . . 47
6.3 Other references . . . . . 50

**Index** **51**

# 1

# Introduction

The involvement of a strong magnetic field at localized regions (called active regions) in the atmosphere of the sun gives rise to different dynamic and spatially confined phenomena. These are sunspot groups, faculae, plages, filaments or prominences (when viewed on the solar disk or the solar limb respectively), flares, coronal mass ejections (CMEs), Solar winds, coronal loops, Solar Energetic Particles (SEPs), etc. The active regions are the areas of strong magnetic fields where the bundles of these field lines extend above the photosphere and form into the loops in the solar atmosphere. Thus, the sun displays a few or all phenomena in the active regions.

One such solar activity is the sunspots, which are the temporary dark spots on the solar photosphere, from which, most of the solar flares and Coronal Mass Ejections originate. It is reasonable to study the sunspot parameters like sunspot numbers(SSN) and sunspot area(SSA) as they are crucial in the study of solar activities. On the other hand, Solar flares are the intense burst of the radiations due to the sudden release of the energy stored in the magnetic fields, and the Solar Flare Index (SFI) is one of the considered parameters to get the rough measurement of their energy.

The cause of these phenomena is the magnetic field. But, there are different parameters such as kinetic energy of CMEs, shear angles, differential rotations, solar cycle, flux densities, irradiance, CME speed parameters, etc. that are essential to look upon for studying these phenomena. Python programming, MATLAB, Supervised Machine Learning, and Ensemble modelling were useful in data analysis, graphs plotting, curve fitting and developing a predictive model.

## 1.1 | Aims and Objectives

### 1.1.1 | **Objective I:** To develop a predictive model for the energy of solar flares with the help of Machine Learning.

Solar flares are the most well-known phenomenon, which is a part of the 11-year solar activity cycle with increasing and decreasing the number of sunspots on the Sun. These sunspots can be tens of thousands of kilometres across. The number of sunspots peak during the solar maxima and are generally closer to the Sun's equator.

These solar flares are accompanied by huge amount of high energy proton and electron ejections exceeding the normal solar winds. It is part of the solar weather and at their peak, could harm satellites due to its excessive magnetic field interactions. Solar flares Index of solar flares depends on various parameters but the most important and easily accessible are Magnetic flux density, sunspot number and sunspot area.

Linking all these parameters would create a better understanding of Solar flares and would create a new perspective to the study of the Sun. The data from this was taken and put into machine learning algorithms to acquire desirable results.

### 1.1.2 | **Objective II:** To find an inverse correlation between the sunspot area and solar activity.

The cause of solar activity in sun is its complex magnetic field and phenomena relating to it i.e. loop formations, magnetic reconnection etc.

Solar activities like solar flares, coronal loops, CMEs are directly related to sunspot formations. The hypothesis is based on a very basic properties of magnetic field lines.

- They don't intersect each other
- The magnetic strength is stronger where the lines are near to each other.

This suggests that for a sunspot with a lesser area, the field lines will be confined to a smaller region which means that the magnetic field strength at that region will be higher.

Hence, the possibility of occurrence of Solar activities are more if the SSA is less. To understand this phenomenon, observation of one particular sunspot has to be done, which is difficult. Overall solar activity and magnetic field strength can be observed and analysis can be done. However, there are other factors affecting the magnetic field strength. At an instance, if number of sunspots are different, the strength is bound to vary. Hence, instances of same/similar SSN are to be considered.

Similarly, even with the same SSN values, the magnetic flux from a particular sunspot might vary. Thus, instances are to be refined further to ones with same/similar magnetic flux. At last, the parameters of solar activity are to be compared with SSA of refined time periods with same SSN and same magnetic flux.

# 2

# Background & Literature Overview

## 2.1 | Magnetic Field of the Sun

The presence of the electric current within the surface of the sun generates the magnetic field, which further causes various activities such as sunspots, solar flares, coronal mass ejections, solar winds, etc. at the solar surface in a regular cycle of 11-years, called the solar cycle. Comparable to the bar magnet, the sun's magnetic field has two poles that flip at the peak of each solar activity cycle.

Plasma, the main constituent of the sun, a gas-like state of matter in which electrons and ions are separated, creating a super-hot mix of charged particles. When charged particles move, they naturally create magnetic fields, which in turn have an additional effect on how the particles move. The plasma in the sun, therefore, sets up a complicated system of cause and effect in which plasma flows inside the sun, churned up by the enormous heat produced by nuclear fusion at the center of the sun, thus creating the sun's magnetic fields.

The differential rotation, which is the rotation of the solar surface at different speeds depending on the latitude, increases the complexity of the solar magnetic field. Differential rotation (DR) is a powerful generator of magnetic fields, and therefore, a key ingredient in stellar dynamo models. The surface DR of the sun has been known for a long time from the tracking of sunspots.

The strongest magnetic field regions are in sunspots, reaching field strengths of B =2000–3000 G. Active regions and their places comprise a larger area around sunspots with average photosphere fields of B $\approx$ 100–300 G, containing small-scale pores with typical fields of B $\approx$ 1000 G. The background magnetic field in the quiet Sun and coronal holes have a net field of B 0.1–0.5 G, while the absolute field strengths in resolved elements amount to B = 10–50 G.Therefore, understanding the magnetic field is extremely important to predict the solar activities and to develop relations among them.

## 2.2 | Sunspots

Sunspots are the dark regions with the intense magnetic field and low temperature, found in the photosphere, are known to the humans since the middle of the fourth century, coined by Theophrastus. They are the temporary structures evolved on the Sun, and their population varies according to the eleven-year cycle of solar minimum and maximum. Their growth, survival, and decay have been explained by different theories and mechanisms.

Sunspots accompany secondary phenomena such as coronal loops, prominences, and reconnection events with the intense magnetic field, and usually appear in pairs of opposite magnetic polarity. Most solar flares and coronal mass ejections originate in magnetically active regions around visible sunspot groupings. Sunspots areas and sunspots numbers are the parameters of the sunspots that are useful and measurable quantities to study the sunspots that accompany secondary phenomena such as coronal loops, prominences, and reconnection events.

It is believed that the rotation of the sun causes distortion in the magnetic field. These distortions cause magnetic areas to break through the photosphere, resulting in the sunspots. Although Alfven's theory of sunspot is not accepted yet, but preferred in many cases, and explains many properties of spots, such as their tendency to occur in pairs of opposite magnetic polarity, the duration of the solar cycle, the reversal of polarities in each new cycle, the progression of spots towards the equator during a cycle, etc.

Most solar flares and coronal mass ejections originate in magnetically active regions around visible sunspot groupings. The Sunspots numbers, which have been recorded for several centuries, used as a proxy for describing the level of solar activity. Sunspot Number, as an index, can be defined on a daily basis. But, because of the large day-to-day variation, they are usually averaged over longer periods, and the most common being the monthly and the yearly average. The sunspot number varies smoothly, charting the progress of the solar cycle when averaged over a year.

## 2.3 | Solar Flares

Solar flares are the intense burst or explosion on the sun's atmosphere, caused due to sudden release of the free energy stored in twisting magnetic fields by magnetic reconnection (Petschek 1964), and usually observed in close proximity to a sunspot group. They can last from minutes to hours and are also sites where particles (electrons, protons, and the heavier ones) are accelerated.

The study of solar flares is necessary as they're the most energetic explosions in the solar system, and they can have a direct effect on Earth's atmosphere. It can also be useful in understanding other cosmic events as the energy release process is similar.

A solar flare can be observed by the emergence of photons (or light) it releases at most every wavelength of the spectrum. The primary ways in which the flares are monitored are in x-rays and optical

light. In the corona, where solar flares occur, the magnetic field structure is controlled by the photospheric magnetic field. Therefore, it is expected that the photospheric magnetic parameters have some strong relationships with the flare parameters. Flares occur in active regions around sunspots, where intense magnetic fields penetrate through the photosphere to link the corona to the solar interior. The same energy releases may produce coronal mass ejections (CMEs).

There are several studies, which has analyzed the relations between the flare and the sunspot properties. A study for the probability of flare eruptions in terms of the sunspot parameters (increase and decrease in the area, spot class, and others) was done by Giovanelli (1939). Also, Mayfield and Lawrence (1985), in their work, reported that the flare index derived from H$\alpha$ emission is proportional to the magnetic flux of active regions.

Solar flare index (SFI), an important parameter to predict the probability of solar flares to occur, and can be defined as the rough measurement of the total energy emitted by the flares. Kleczek (1952) introduced the quantity **FI = it** to quantify the daily flare activity over 24 hours per day and named it 'flare index' (FI). In this relation, **i** represents the intensity scale of importance and **t** the duration of the flare in minutes.

Solar flare index can also be expressed as the energy of the flares per unit area. FI is one of the best indicators of activity variations in the chromosphere. It is of value as a measure of the short-lived (minutes to hours) activity on the Sun. The comparison of FI with these similar indices should indicate how well they correlate, and this will be useful to model the temporal variations of solar activity.

## 2.4 | Coronal Mass Ejections

One of the phenomena occurred due to the Solar Magnetic field, in which a significant amount of plasma releases from the sun's corona. CMEs most frequently originate from active regions on the Sun's surface, such as groupings of sunspots, and are associated with solar flares.

A plausible important cause behind the generation of CMEs is magnetic reconnection. When two oppositely charged magnetic fields are brought together, rearrangement takes place, and immediately after this rearrangement, the energy stored in the oppositely directed magnetic field lines is released. This sudden release of energy is assumed responsible for solar flares which causes the CMEs.

CMEs are responsible for large–scale changes in the corona that have fundamental implications for the evolution of the magnetic flux of the Sun. Therefore, we can expect a good correlation between the various observed physical parameters of CMEs and the other tracers of solar activity such as sunspots number and sunspot areas. Due to the magnetic field, CMEs' parameters like acceleration, mass, Kinetic energy, width, linear speed, and second-order initial and final speed are affected too.

The characterization of CMEs depends on three speeds: (1) the linear speed (2) initial quadratic speed, and (3) the final quadratic speed when the CMEs reach a height of 20 solar radii. The acceleration of a CME can be positive, negative, or close to zero, meaning CMEs speed up, move with constant speed or slow down.

## 2.5 | Total Solar Irradiance (TSI)

The TSI variation is an important parameter for the understanding of the solar internal structure and solar-terrestrial relationships. It can indicate a secular change that might be associated with subtle changes in the solar radius, which may be related to a pulsating solar core. Short-term changes of TSI during the solar activity cycles are expected to be due to the luminosity changes connected with the temperature fluctuation of the solar surface, and may also be due to the redistribution of the solar radiation by sunspots and active region population.

Measurements of TSI vary over a range of periodicities, and most of the observed changes are probably associated with other solar activities, e.g., sunspot numbers, solar flares, solar diameter, solar neutrino fluxes, low order acoustic p-modes, etc. Estimations of solar magnetic field contribution to total solar irradiance variations show that magnetic features at the solar surface account for over 90% of the irradiance variations on a solar rotation time scale and at least 70% on a solar cycle time scale.

**For the research project, the knowledge of the computation skills and predictive models were also essential. Therefore, the applications of Supervised Machine Learning and Python programming were implemented, the theoretical concepts behind the Scatterplot Matrix, MATLAB, Curve Fit Models, Linear and Non-linear Regressions, Regression Forest Method, and Ensembled Modeling were also understood.**

## 2.6 | Machine Learning

Machine learning involves a computer to be trained using a given data set, and use this training to predict the properties of a given new data. For example, a computer can be train by feeding it 1000 images of cats and 1000 more images other than that of a cat, and train so that the computer identifies the corresponding image correctly. When the computer observes a new image, it should be able to differentiate between cat's and other picture from the above training. The process of training and prediction involves the use of specialized algorithms. The training data is feed to an algorithm, and it uses the training data to give predictions on a new test data.

At most fundamental level, machine learning is categorized to three major kinds: supervised, unsupervised, and reinforcement learning. Supervised learning involves modeling the relationship between measured features of data and some label associated with the data. Once this model is determined, it can be used to apply labels to new and unknown data. This is further subdivided into classification tasks and regression tasks: in classification, the labels are discrete categories, while in

regression, the labels are continuous quantities.

Regression analysis consists of a set of machine learning methods that allow us to predict a continuous outcome variable (y) based on the value of one or multiple predictor variables (x). Briefly, the goal of regression model is to build a mathematical equation that defines y as a function of the x variables.

## 2.7 | Curve fitting

Regression analysis is a form of predictive modeling technique, which investigates the relationship between a dependent and independent variable. In the regression, curve fitting is a mathematical tool that examines the relationship between one or more predictors (independent variables) and a response variable (dependent variable), intending to define the best fit model of the relationship. The trend in the data is captured and allows us to make predictions of how the data series will behave in the future.

Whenever the scattered data is fit to a straight line, then it is called linear regression, or the linear curve fit. A straight line equation is obtained, having two constant parameters, called slope and intercept. But, it is not necessary to get much accuracy in the linear relationship for specific data sets, so it is better to look for the polynomial curve fits in those cases. Polynomial curve fitting is used to fit the data to the graph of a polynomial function.

For linear relationships, the mean of the dependent variable always changes by a specific amount for a unit increase in the independent variable. But, the change in the dependent variable associated with a one-unit shift in the independent variable varies based on location in observation space for a curved relationship.

Whenever the data is curve fitted, there are a few related values such as RMSE, SSE, R-square, and adjusted R-square associated with the curve that describes the errors, relationship between dependent and independent variables, and accuracy of the curve fit, which are helpful in statistical analysis of the graph. Thus, these quantities are the measure of the goodness of the fit. Apart from that, the equations can be obtained for a predictive model for the dependent variable.

RMSE or Root Mean Square Error is the standard deviation of the residuals (prediction errors). Residuals are a measure of distance of data points from the regression line; RMSE is a measure of spread of the residuals. In other words, it tells you degree of concentration of the data around the line of best fit. While SSE is the sum of squares due to errors, measures the total deviation of the response values from the fit to the response values. It is also called the summed square of residuals and is usually labeled as SSE.

R-square, also known as the coefficient of determination is the proportion of the variance in the dependent variable that is predictable from the independent variable(s). It is a statistic used in the

context of statistical models whose main purpose is either the prediction of future outcomes or the testing of hypothesis, on the basis of other related information. It is also a measurement of the closeness of the fit to the data points and the values ranges from 0 to 1. Therefore, it can be said that the R-square is the measurement of the accuracy of the fit. It is also known as the square of the correlation coefficient.

Correlation coefficient is defined as the statistical measure of the strength of the relationship between the relative movements of two variables. The values range between -1.0 and 1.0. There are several types of correlation coefficients, but the one that is most common is the Pearson correlation (r). This measures the strength and direction of the linear relationship between two variables. Another type is the Spearman's rank correlation coefficient, which is a measure of the relationship between two variables that can be described by a monotonic function. The sign of the Spearman correlation indicates the direction of association between X (the independent variable) and Y (the dependent variable).

## 2.8 | Scatterplot Matrix

A scatter plot matrix is a grid (or matrix) of scatter plots used to visualize bivariate relationships between combinations of variables. Each scatter plot in the matrix visualizes the relationship between a pair of variables, allowing many relationships to be explored in one chart. A scatter plot matrix is composed of a grid of mini-plots and one larger preview plot that shows selected mini-plot in more detail. A scatter plot matrix is made up of three or more Numeric fields. A scatter plot will be created for every pairwise combination of variables.

## 2.9 | Ensembled Methods

It is a meta-algorithm which combines smaller or weaker models working on different subsets of main dataset to produce better predictions with less errors. The goal of ensemble methods is to combine the predictions of several base estimators built with a given learning algorithm in order to improve generalizability (robustness) over a single estimator. It can be categorized into its two major types: Parallel Ensemble Learning and Sequential Learning.

In the Parallel Learning, all base learners are running parallel. Bragging is one such method that puts data into 5 different bags and put random data points and then the model is trained and retrained. But in the Sequential Learning, base Learners are ran on a sequential basis so that every next model is an improvement over the previous one. Boosting is one such model, which takes a subset, trains, finds errors, and emphasises on those errors in the next base learner along with random data points from main data set. There are mainly Adaptive Boosting, Gradient Boosting and XG boosting for the modeling.

Adaptive Boost has sequence of week learners whose results are combined by weighted average. Gradient Boosting trains many models in a gradual, additive and sequential manner. The major difference between Adaptive Boost and Gradient Boosting Algorithm is how the two algorithms

identify the shortcomings of weak learners (eg. decision trees). While the Adaptive Boost model identifies the shortcomings by using high weight data points, gradient boosting performs the same by using gradients in the loss function.

We used LS Boost Optimizable Ensemble learning with 5 fold cross validation, i.e., we applied Ensemble method of type LS boost (Comes under Gradient Boosting), and running it 5 different times on data (each of these base learners worked on different subsets of data) and produced the predictions.

# 3

# Methodology & Data Collection

The work began with the background reading and collective study of the sun's interior, solar atmosphere, solar probes, and solar activities such as Coronal Mass Ejections, Solar Flares, Solar energetic particles, and High-speed solar wind streams, and the factors which influence solar activity. The core of all these activities was found to be the magnetic field followed by other factors such as sunspots number and area, tilt angle, speed parameters, number of flares, flare index, flux densities, etc. So, to align the objectives with the studies and data availability, the parameters were selected.

## 3.1 | Objective I: To develop a predictive model for the energy of solar flares using machine learning

Understanding machine learning played an important role in the work. The team first studied upon various types and methods and upon discussion came to a standpoint of using reinforced learning. However, first the data was arranged and analysed. Since the data exists between different timelines, the dataset was shortened and cut down to get equal amounts for comparison.
For training the predictive model, data for different solar flare parameters like Solar Flare Index (SFI), number of Solar Flares, Sunspot Area (SSA), Sunspot Number (SSN) were collected. However, the data was in a raw form i.e. unprocessable form. Thus, the data was later neatly arranged in Excel. Some datasets were in 'DAT' format which was then put in Excel sheets for easier data handling. If the datasets were on a daily basis, it was averaged month-wise by using the below mentioned code.

---

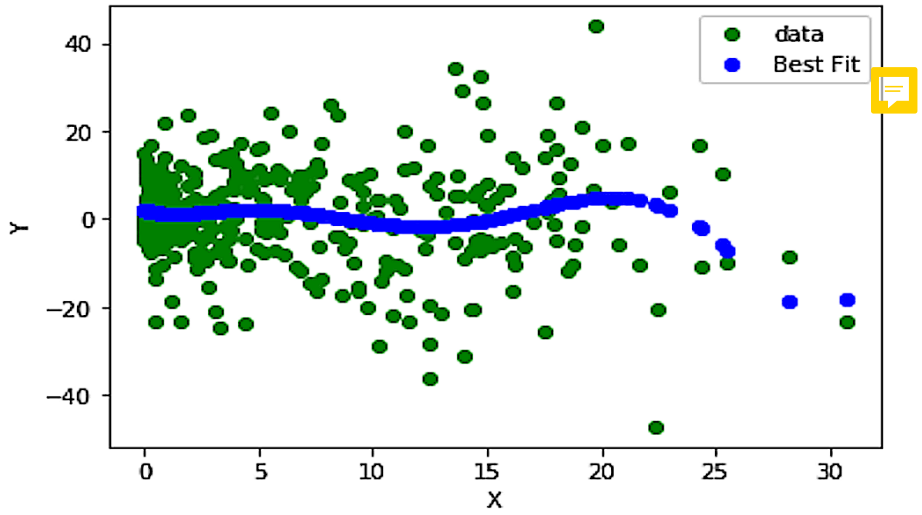
import pandas as pd

$fulldata = pd.read\_excel(r'C:\Users\SPAS\Documents\Courses\_April2020\SSERD\Magnetic\_flux\_density\_data.xlsx')$

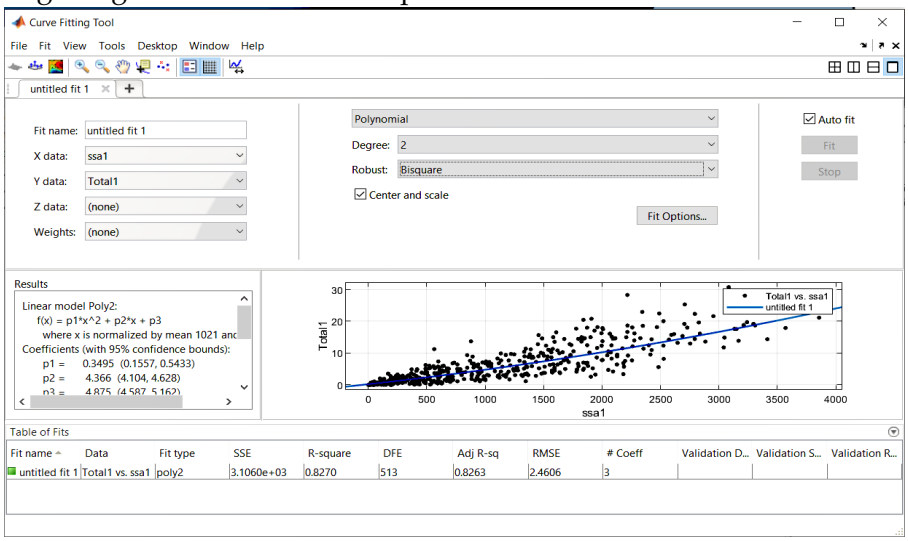
df1 = pd.DataFrame(fulldata, columns= ['dated'])
df2 = pd.DataFrame(fulldata, columns= ['MFD'])

```
datedarr = df1.values
magfielddens = df2.values

cnt=1
sum1=0

file = open('mfd_avg.txt', 'a')
for i in range(1,len(datedarr)):
if(datedarr[i]==datedarr[i-1]):
if(magfielddens[i-1][0]!=100):
sum1+=magfielddens[i-1][0]
cnt+=1

else:
sum1+=magfielddens[i-1][0]
avg=sum1/cnt
file.write(str(avg))
file.write('\n')
sum1=0
cnt=1


final_list = []
for num in datedarr:
if num not in final_list:
final_list.append(num)

# print(final_list)
file = open('mfd_avg_date.txt', 'a')
for num in final_list:
arrOfStr = str(num).split();
res = "";
for a in arrOfStr:
res += a[2:len(a) - 2] + " ";
file.write(res)
file.write('\n')
file.close()

print("end")
```

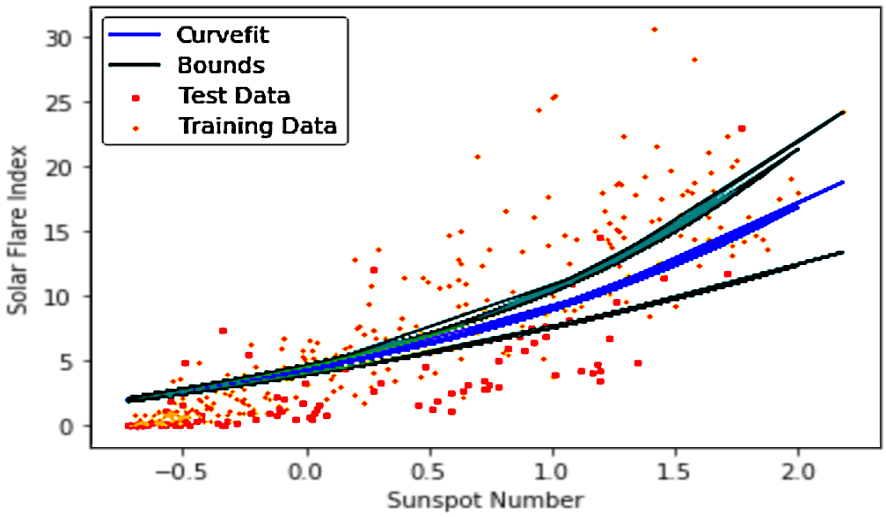
The data for SFI was taken and plotted against other parameters and curve fitted using python. Different curves were tried to curve-fit it to the maximum accuracy. One of the results and its corresponding code are shown in the pictures below. Unfortunately, the results were not satisfactory.



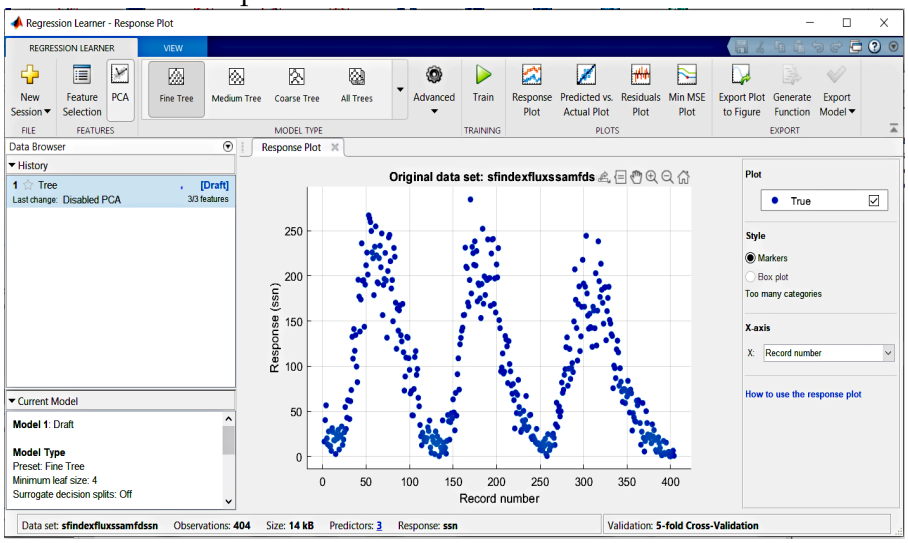

To make the curve fitting better and more accurate, team switched to MATLAB. Using Curve Fit toolbox on MATLAB, the data was curve fitted in much better way. The versatility of MATLAB helped in getting better results. A sample of which is shown below.




Each curve fit was used to understand the plots and the relation between the variables. Initially, the aim was to get a better Root Mean Square Error (RMSE) value but with time, focus was shifted to the R-square value instead. Training and testing of data was done in python, once again. However, the results were lacking expected accuracy and precision. The team relied upon MATLAB once again.




The data was hence arranged into a single excel sheet and imported into MATLAB. The MATLAB Regression Learner was used where the software receives the data and applies various regression models to it. The regression learner contained all the required tools to build the model very precise and accurate. The picture of the feature is shown below.




Individually, the output of the models was satisfactory. However, insatiable to get more accuracy, the team applied ensemble method to the data. The model was enhanced and errors went down to a great extent. The code and its corresponding output was recorded and is displayed below.

## 1) Generated code from Regression Learner App:

```
function [trainedModel, validationRMSE] = trainRegressionModel(trainingData)
% [trainedModel, validationRMSE] = trainRegressionModel(trainingData)
% returns a trained regression model and its RMSE. This code recreates the
% model trained in Regression Learner app. Use the generated code to
% automate training the same model with new data, or to learn how to
% programmatically train models.
%
```

```
% Input:
% trainingData: a table containing the same predictor and response
% columns as imported into the app.
%
% Output:
% trainedModel: a struct containing the trained regression model. The
% struct contains various fields with information about the trained
% model.
%
% trainedModel.predictFcn: a function to make predictions on new data.
% validationRMSE: a double containing the RMSE. In the app, the
% History list displays the RMSE for each model.
%
% Use the code to train the model with new data. To retrain your model,
% call the function from the command line with your original data or new
% data as the input argument trainingData.
%
% For example, to retrain a regression model trained with the original data
% set T, enter:
% [trainedModel, validationRMSE] = trainRegressionModel(T)
%
% To make predictions with the returned 'trainedModel' on new data T2, use
% yfit = trainedModel.predictFcn(T2)
%
% T2 must be a table containing at least the same predictor columns as used
% during training. For details, enter:
% trainedModel.HowToPredict

% Auto-generated by MATLAB on 28-Jul-2020 17:35:47


% Extract predictors and response
% This code processes the data into the right shape for training the
% model.
inputTable = trainingData;
predictorNames = 'mfd_avg', 'SSA', 'ssn';
predictors = inputTable(:, predictorNames);
response = inputTable.SFIndex;
isCategoricalPredictor = [false, false, false];

% Train a regression model
% This code specifies all the model options and trains the model.
```

```
template = templateTree(...
'MinLeafSize', 1, ...
'NumVariablesToSample', 3);
regressionEnsemble = fitrensemble(...
predictors, ...
response, ...
'Method', 'LSBoost', ...
'NumLearningCycles', 10, ...
'Learners', template, ...
'LearnRate', 0.2480844467477807);

% Create the result struct with predict function
predictorExtractionFcn = @(t) t(:, predictorNames);
ensemblePredictFcn = @(x)
predict(regressionEnsemble, x);
trainedModel.predictFcn = @(x)
ensemblePredictFcn(predictorExtractionFcn(x));

% Add additional fields to the result struct trainedModel.RequiredVariables = 'SSA', 'mfd_avg', 'ssn';
trainedModel.RegressionEnsemble = regressionEnsemble;
trainedModel.About = 'This struct is a trained model exported from Regression Learner R2019b.';
trainedModel.HowToPredict = sprintf
('To make predictions on a new table, T, use: \n yfit = c.predictFcn(T) \nreplacing "c" with the name of the variable
that is this struct, e.g. "trainedModel". \n \nThe table, T, must contain the variables returned by:
\n c.RequiredVariables \nVariable formats (e.g. matrix/vector, datatype) must match the original training data. \n
Additional variables are ignored. \n \nFor more information, see <a href="matlab:helpview
(fullfile(docroot, "stats", "stats.map"), "appregression_exportmodeltoworkspace")">
How to predict using an exported model</a>.');

% Extract predictors and response
% This code processes the data into the
right shape for training the
% model.
inputTable = trainingData;
predictorNames = 'mfd_avg', 'SSA', 'ssn';
predictors = inputTable(:, predictorNames);
response = inputTable.SFIndex;
isCategoricalPredictor = [false, false, false];

% Perform cross-validation
```

partitionedModel = crossval(trainedModel.RegressionEnsemble, 'KFold', 5);

% Compute validation predictions validationPredictions = kfoldPredict(partitionedModel);

% Compute validation RMSE
validationRMSE = sqrt(kfoldLoss(partitionedModel, 'LossFun', 'mse'));

30/7/20 7:06 PM MATLAB Command Window 14 of 14

```
>> [trainedModel, validationRMSE] = trainRegressionModel(trainingData)

trainedModel =

  struct with fields:

           predictFcn: @(x)ensemblePredictFcn(predictorExtractionFcn(x))
     RequiredVariables: {'SSA'  'mfd_avg'  'ssn'}
    RegressionEnsemble: [1×1 classreg.learning.regr.RegressionEnsemble]
                 About: 'This struct is a trained model exported from Regression↙
Learner R2019b.'
          HowToPredict: 'To make predictions on a new table, T, use: ↵  yfit = c.↙
predictFcn(T) ↵replacing 'c' with the name of the variable that is this struct, e.g.↙
'trainedModel'. ↵ ↵The table, T, must contain the variables returned by: ↵  c.↙
RequiredVariables ↵Variable formats (e.g. matrix/vector, datatype) must match the↙
original training data. ↵Additional variables are ignored. ↵ ↵For more information,↙
see How to predict using an exported model.'


validationRMSE =

    3.1326

>> yfit = trainedModel.predictFcn(T2)

yfit =

    0.7419
```

## Code for same model without using Regression Learner App:

```
load('sfinewopt.mat')
X = [mfd_avg SSA ssn];
Y = SFIndex;
rng default
Mdl = fitrensemble(X,Y,...
   'Method','LSBoost',...
   'Learner',templateTree('Surrogate','on'),...
   'OptimizeHyperparameters',{'NumLearningCycles','MaxNumSplits','LearnRate'},...
   'HyperparameterOptimizationOptions',struct('Repartition',true,...
   'AcquisitionFunctionName','expected-improvement-plus'))
loss = kfoldLoss(crossval(Mdl,'kfold',5))
Mdl2 = fitrensemble(X,Y,...
   'Method','LSBoost',...
   'Learner',templateTree('Surrogate','on'));
loss2 = kfoldLoss(crossval(Mdl2,'kfold',5))
Mdl3 = fitrensemble(X,Y);
loss3 = kfoldLoss(crossval(Mdl3,'kfold',10))
```

Output-

```
        >> code_without_app
|=====================================================================================================
=====================|
| Iter | Eval   | Objective:  | Objective   | BestSoFar   | BestSoFar   | NumLearningC-|   LearnRate | MaxNumSplits |
|      | result | log(1+loss) | runtime     | (observed)  | (estim.)    | ycles        |             |              |
|=====================================================================================================
=====================|
|    1 | Best   |      2.8621 |      8.6406 |      2.8621 |      2.8621 |          383 |     0.51519 |            6 |
|    2 | Best   |      2.7014 |     0.46159 |      2.7014 |      2.7099 |           16 |     0.66503 |           13 |
|    3 | Best   |      2.5302 |     0.92495 |      2.5302 |      2.5313 |           33 |      0.2556 |          350 |
|    4 | Accept |      4.2993 |     0.34115 |      2.5302 |      2.5304 |           13 |   0.0053227 |            8 |
|    5 | Accept |      2.8378 |      1.8638 |      2.5302 |      2.5305 |           68 |     0.99821 |          385 |
|    6 | Accept |      2.6973 |     0.31369 |      2.5302 |      2.5307 |           10 |     0.13171 |          295 |
|    7 | Best   |      2.4165 |     0.29034 |      2.4165 |      2.4176 |           10 |     0.28637 |          364 |
|    8 | Accept |      2.4795 |     0.28574 |      2.4165 |      2.4296 |           10 |     0.35491 |          371 |
|    9 | Accept |      2.4493 |     0.28383 |      2.4165 |      2.4387 |           10 |     0.25623 |          136 |
```

| 10 | Best | 2.357 | 0.25126 | 2.357 | 2.3924 | 10 | 0.27012 | 8 |
| 11 | Accept | 2.4201 | 0.24155 | 2.357 | 2.4199 | 10 | 0.26632 | 1 |
| 12 | Accept | 3.6867 | 10.334 | 2.357 | 2.4197 | 450 | 0.0010008 | 72 |
| 13 | Accept | 2.5278 | 13.068 | 2.357 | 2.4171 | 500 | 0.027803 | 211 |
| 14 | Best | 2.3473 | 9.4441 | 2.3473 | 2.3524 | 499 | 0.046688 | 1 |
| 15 | Accept | 3.6761 | 0.26784 | 2.3473 | 2.3527 | 11 | 0.043683 | 2 |
| 16 | Accept | 2.5469 | 13.285 | 2.3473 | 2.3653 | 500 | 0.092423 | 384 |
| 17 | Accept | 2.5713 | 9.5351 | 2.3473 | 2.4259 | 499 | 0.048562 | 4 |
| 18 | Accept | 4.4025 | 0.23602 | 2.3473 | 2.4268 | 10 | 0.0010049 | 3 |
| 19 | Accept | 2.5428 | 10.15 | 2.3473 | 2.4255 | 497 | 0.0040187 | 8 |
| 20 | Best | 2.3061 | 9.6017 | 2.3061 | 2.3305 | 499 | 0.0090502 | 3 |

| Iter | Eval result | Objective: log(1+loss) | Objective runtime | BestSoFar (observed) | BestSoFar (estim.) | NumLearningC-ycles | LearnRate | MaxNumSplits |
|---|---|---|---|---|---|---|---|---|
| 21 | Accept | 2.459 | 12.068 | 2.3061 | 2.3636 | 499 | 0.0093261 | 155 |
| 22 | Accept | 2.3464 | 10.092 | 2.3061 | 2.3479 | 498 | 0.0096835 | 1 |
| 23 | Accept | 2.3481 | 9.8439 | 2.3061 | 2.3447 | 500 | 0.010916 | 1 |
| 24 | Accept | 2.3879 | 9.5912 | 2.3061 | 2.3554 | 498 | 0.0086048 | 1 |
| 25 | Accept | 2.3263 | 9.2733 | 2.3061 | 2.3465 | 496 | 0.012019 | 1 |
| 26 | Best | 2.2731 | 9.1628 | 2.2731 | 2.3295 | 494 | 0.011746 | 1 |
| 27 | Accept | 2.3571 | 9.6821 | 2.2731 | 2.3319 | 500 | 0.014501 | 1 |
| 28 | Accept | 2.4199 | 9.0396 | 2.2731 | 2.3388 | 497 | 0.0081132 | 1 |
| 29 | Accept | 2.3651 | 9.1128 | 2.2731 | 2.3412 | 498 | 0.013798 | 1 |
| 30 | Accept | 2.6574 | 1.778 | 2.2731 | 2.3422 | 94 | 0.019379 | 1 |

---

Optimization completed.

MaxObjectiveEvaluations of 30 reached.

Total function evaluations: 30

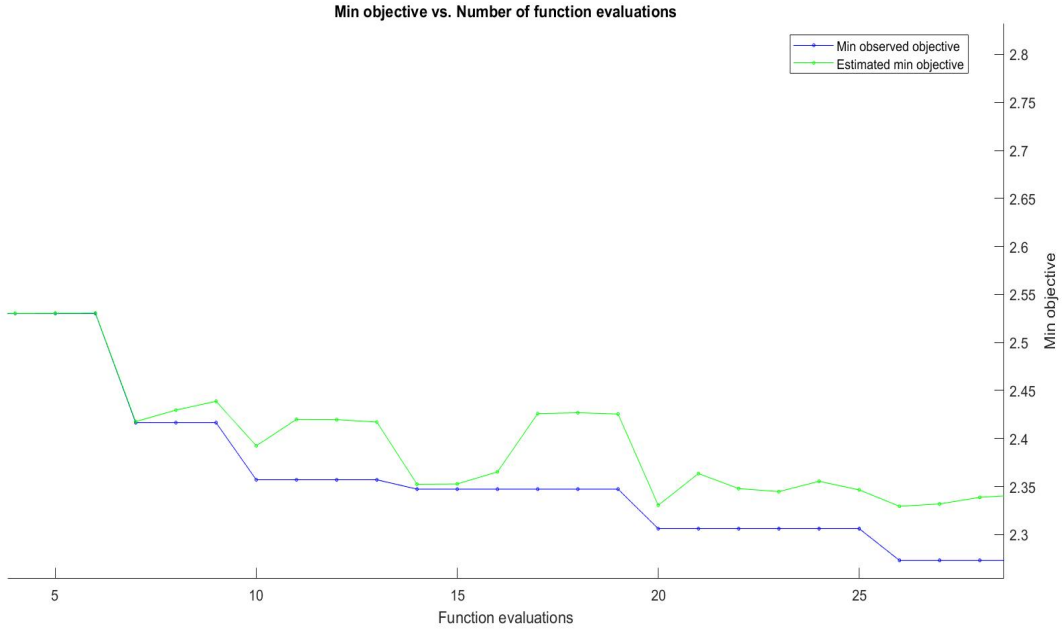
Total elapsed time: 214.99 seconds.

Total objective function evaluation time: 179.4644

Best observed feasible point:

```
    NumLearningCycles    LearnRate    MaxNumSplits
    _________________    _________    ____________

          494            0.011746          1

Observed objective function value = 2.2731
Estimated objective function value = 2.3422
Function evaluation time = 9.1628

Best estimated feasible point (according to models):
    NumLearningCycles    LearnRate    MaxNumSplits
    _________________    _________    ____________

          500            0.010916          1

Estimated objective function value = 2.3422
Estimated function evaluation time = 9.4136

Mdl =
  classreg.learning.regr.RegressionEnsemble
                 ResponseName: 'Y'
        CategoricalPredictors: []
           ResponseTransform: 'none'
            NumObservations: 387
  HyperparameterOptimizationResults: [1×1 BayesianOptimization]
                NumTrained: 500
                  Method: 'LSBoost'
              LearnerNames: {'Tree'}
         ReasonForTermination: 'Terminated normally after completing the requested number of training cycles.'
                  FitInfo: [500×1 double]
          FitInfoDescription: {2×1 cell}
             Regularization: []
  Properties, Methods

loss =  9.2569
loss2 =  16.7055
loss3 =  15.5417
```

### Output for hyperparameter optimization:




The recorded results were analysed and compared with the standard values to ensure proper model building.

## 3.2 | Objective II: To find an inverse correlation between the sunspot area and solar activity

Concerning to the correlation aspect of our objective, three different CMEs' speed parameters, Total Solar Irradiance (TSI), and CMEs width and solar flares parameters like Solar Flare Index (SFI), Number of flares (monthly) were taken into consideration. At first, the data for the chosen parameters was collected from various credible sources. However, the data was in raw form i.e. not in processable form (.dat or .txt format). Hence, the data was to be converted to excel for further processing. Some of the data was as daily data which was to be averaged to monthly for further use. This was done using a python code.

### Averaging code

```
import pandas as pd

fulldata = pd.read_excel (r'C:\Users\SPAS\Documents\Courses_April2020\SSERD\CMEavg.xlsx')

df1 = pd.DataFrame(fulldata, columns= ['Date2'])
df2 = pd.DataFrame(fulldata, columns= ['Acceleration'])
```

```
datedarr = df1.values
central = df2.values cnt=1
sum1=0

file = open('cme_avg.txt', 'a')

for i in range(1,len(datedarr)):
if(datedarr[i]==datedarr[i-1]):
if(central[i-1][0]!=100):
sum1+=central[i-1][0]
cnt+=1

else:
sum1+=central[i-1][0]
avg=sum1/cnt

file.write(str(avg))
file.write('\n')

sum1=0
cnt=1

file.close()
print("end")
```

---

The objective entails development of a correlation between the mentioned parameters and SSA without the effects of SSN and Magnetic Flux changes as explained before. Hence, it became important for the team to do a 2-stage refining on the existing data to get the data corresponding to same/similar SSN and Magnetic Flux. A python code was used to categorize all the epochs (time periods) which had SSN values in the ranges of 1-5, 6-10 and 11-15.

## Code to sort the data to SSN ranges

---

```
import pandas as pd
import datetime

fulldata = pd.read_excel (r'C:\Users\SPAS\Documents\Courses_April2020\SSERD\daily1.xlsx')

df1 = pd.DataFrame(fulldata, columns= ['dated'])
df2 = pd.DataFrame(fulldata, columns= ['sunspotno'])
```

```
datedarr = df1.values
sunspotno = df2.values
print(datedarr[1])
print(sunspotno[1])


final_list = []
for num in sunspotno:
if num not in final_list:
final_list.append(num)
final_list.sort()
cnt=0
max1=0
sum1=0
for key in final_list:
if(key>max1):
max1=key
sum1+=key
cnt+=1
print('Average= ')
print(sum1/cnt)

print(str(datedarr[1]))
string = str(datedarr[1])

arrOfStr = string.split();
res = "";
for a in arrOfStr :
res += a[2:len(a) - 2] + " ";
print(res)

string2 = '09/02/2017'
print(string2)
format_str = '%d/%m/%Y' # The format datetime_obj = datetime.datetime.strptime(res, format_str)
print(datetime_obj.date())
```

In the 2nd stage of processing i.e. for attaining data with same amount of magnetic flux, team encountered a hurdle. The data for magnetic flux was not available. Hence, the data for magnetic flux density and SSA were multiplied for the corresponding epochs and the magnetic flux data was attained. Upon observation, the magnetic flux values were not same for any of the epochs. Thus, it became inevitable for the team, but to categorize the epochs in the ranges of Magnetic Flux differ-

ence. A program was coded to achieve the same.

## Magnetic flux code

```
import pandas as pd

fulldata = pd.read_excel (r'C:\Users\SPAS\Documents\Courses_April2020\SSERD\Magnetic_flux_density_data.xlsx')

df1 = pd.DataFrame(fulldata, columns= ['dated'])
df2 = pd.DataFrame(fulldata, columns= ['MFD'])

datedarr = df1.values
magfielddens = df2.values

cnt=1
sum1=0

file = open('mfd_avg.txt', 'a')
for i in range(1,len(datedarr)):
if(datedarr[i]==datedarr[i-1]):
if(magfielddens[i-1][0]!=100):
sum1+=magfielddens[i-1][0]
cnt+=1

else:
sum1+=magfielddens[i-1][0]
avg=sum1/cnt
file.write(str(avg))
file.write('\n')
sum1=0
cnt=1


final_list = []
for num in datedarr:
if num not in final_list:
final_list.append(num)

# print(final_list)
file = open('mfd_avg_date.txt', 'a')
for num in final_list:
```

```
arrOfStr = str(num).split();
res = "";
for a in arrOfStr:
res += a[2:len(a) - 2] + " ";
file.write(res)
file.write('\n')
file.close()

print("end")
```

---

The range of Magnetic flux difference (MF difference) was considered and not Magnetic flux due to the fact that the MF difference is the affecting factor in the hypothesis. Multiple ranges were chosen to understand the trend of parametric changes with MF difference as well. The next task for the team was to get the data for the corresponding refined epochs. Programming was used here as well.

## Data extraction code

---

```
import numpy as np
import scipy.optimize
import pandas as p
import matplotlib.pyplot as plt
import xlwt
from xlwt import Workbook


epochdata = p.read_excel
(r'D:\PRATEEK\SSERD\Data\Ready to process\Epoch_TSI(m).xlsx')
epochs = p.DataFrame(epochdata, columns= ['Epochs'])
MY = p.DataFrame(epochdata, columns= ['MY'])
area = p.DataFrame(epochdata, columns= ['Area'])
par= p.DataFrame(epochdata, columns=['TSI'])

Epochs = epochs.values
my = MY.values
Area = area.values
Par=par.values

i=0
j=0
k=0
l=0
```

```
wb=Workbook()
sheet1=wb.add_sheet("Sheet 1")

for i in range(0,len(Epochs)) :
if Epochs[i]=="100/100":
k+=2
l=0
else:
for j in range(0,len(my)):
if (Epochs[i]==my[j]):

sheet1.write(l+1,k,str(Area[i][0]))

sheet1.write(l+1,k+1,str(Par[j][0]))
l+=1

wb.save('Epoch_TSI_6-10.xls')
print("end")
```

At the end of this stage, the data for all the parameters and its corresponding SSA values was available and ready to be plotted. The data was read in the program and was plotted. Further, it was curve-fitted with an inverse equation (i.e. $y = m/x + b$ ). For all the graphs, the correlation coefficient was also found. Unfortunately, the results were completely off the hypothesis.
Upon introspection, the team found the mistake. There was a human error in the 2nd stage of the refining. Instead of using the refined data from the first stage, the original data was used and hence the results were flawed. The results further improved after a talk with Mr. Pavan Kumar who suggested us to explore the possibilities of different equations for curve-fitting. The equation of curve fit was changed to a general equation (i.e. $y = (m/x^a + n/x^b + c)^d$)

## Plotting and curve-fitting code

```
 import math
import numpy as np
import scipy.optimize
from scipy.optimize import curve_fit
import pandas as p
import matplotlib.pyplot as plt
from scipy.stats import pearsonr
from scipy.stats import spearmanr
from scipy.stats import kendalltau
```

```
x, y = np.loadtxt("Book1.csv", delimiter=",", skiprows=1, unpack = True, usecols=[0,1])

i=0
j=i+1

for i in range (0,len(x)-1):
for j in range (i+1,len(x)-1):
if x[i]>x[j]:
temp=x[i]
x[i]=x[j]
x[j]=temp
temp2=y[i]
y[i]=y[j]
y[j]=temp2
j+=1


def test(x, a, b, c, d, m, n, o, q, e ,f):
return (m/x**a + n/x**b + c)**d

param,param_cov= curve_fit(test,x,y, maxfev=1000000000)

residuals = y- test(x, *param)
ss_res = np.sum(residuals**2)
ss_tot = np.sum((y-np.mean(y))**2)
r_squared = 1 - (ss_res / ss_tot)

print(r_squared)

spm, _ = spearmanr(x,y)
kt, _ = kendalltau(x,y)
data=len(x)
plt.plot(x,y,'o', label=data)
plt.plot(x,test(x,*param), label=spm)
plt.xlabel('SSA')
plt.ylabel('SFI')
plt.title('SFI vs. SSA (SSN 1-10, 0-900 MF difference)')
plt.legend()
plt.show()
```

# 4

# Results & Discussion

## 4.1 | Objective I: To develop a predictive model for energy of solar flares using Machine Learning

### Prediction model for Solar Flare:

Firstly, we took 384 datasets in total.The predictors we took are average Magnetic Flux Density, Sunspot Number and Sunspot Area.The response of Solar Flare Index was recorded.

#### LS boost optimizable ensemble regression model

**Cross-validation:** 5 folds validation
The results obtained were:
**RMSE :** 2.9653
**R-squared :** 0.79
**MSE :** 8.7929
**MAE :** 2.0799
**Prediction speed** = 14000 obs/sec
**Training Time :** 193.59 sec

### Model type:

**Preset:** Optimizable Ensemble

### Optimized Hyperparameters

**Ensemble method :** LS Boost
**Minimum leaf size :** 1
**Number of learners :** 10
**Number of predictors to sample :** 3

### Hyperparameter Search Range

**Ensemble method :** Bag, LS Boost
**Number of learners :** 10-500

**Learning rate :** 0.001-1
**Minimum leaf size :** 1-192
**Number of predictors to sample :** 1-3

## Optimizer Options

**Optimizer Bayesian optimization Acquisition function :** Expected improvement per second plus
**Training time limit :** false
**Iterations :** 30

## Feature Selection

All features used in the model, before PCA.
**PCA:** PCA disabled

## SFI vs Magnetic Flux Density:

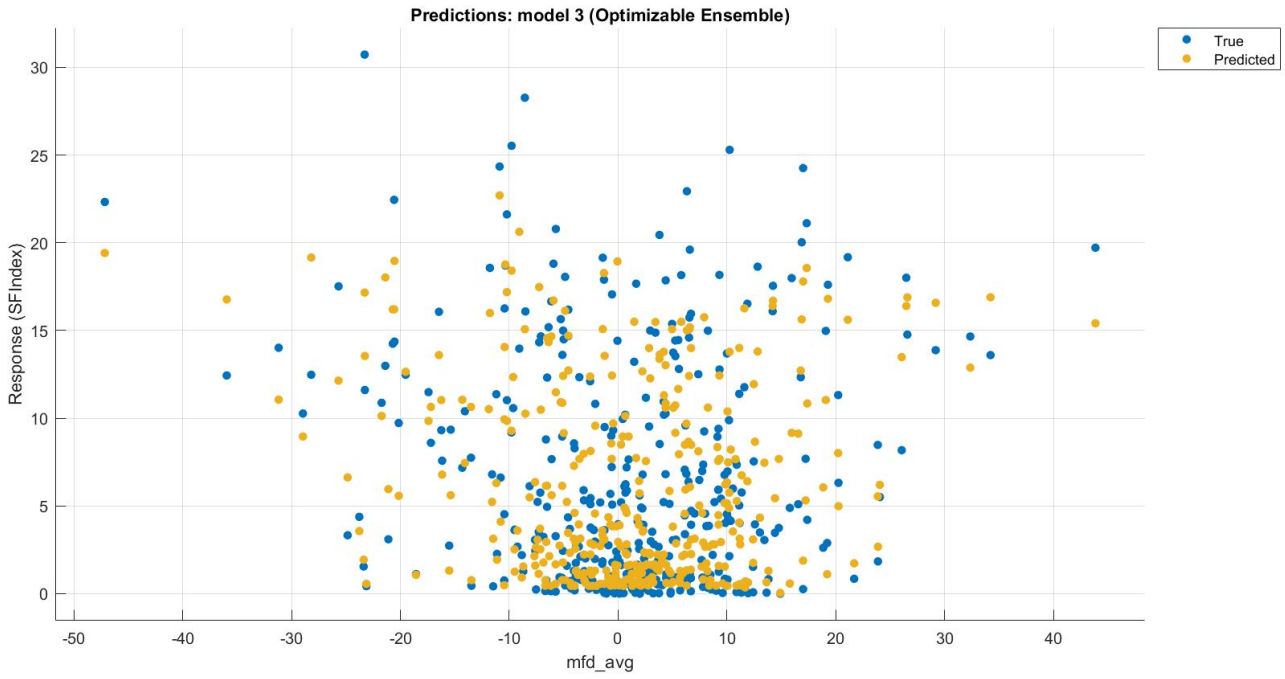



Figure 4.1: Prediction model between SF Index and Average MFD

This is a prediction model between SF Index and Average MFD. It shows us the true and predicted data points.

## SFI v/s Record Number:

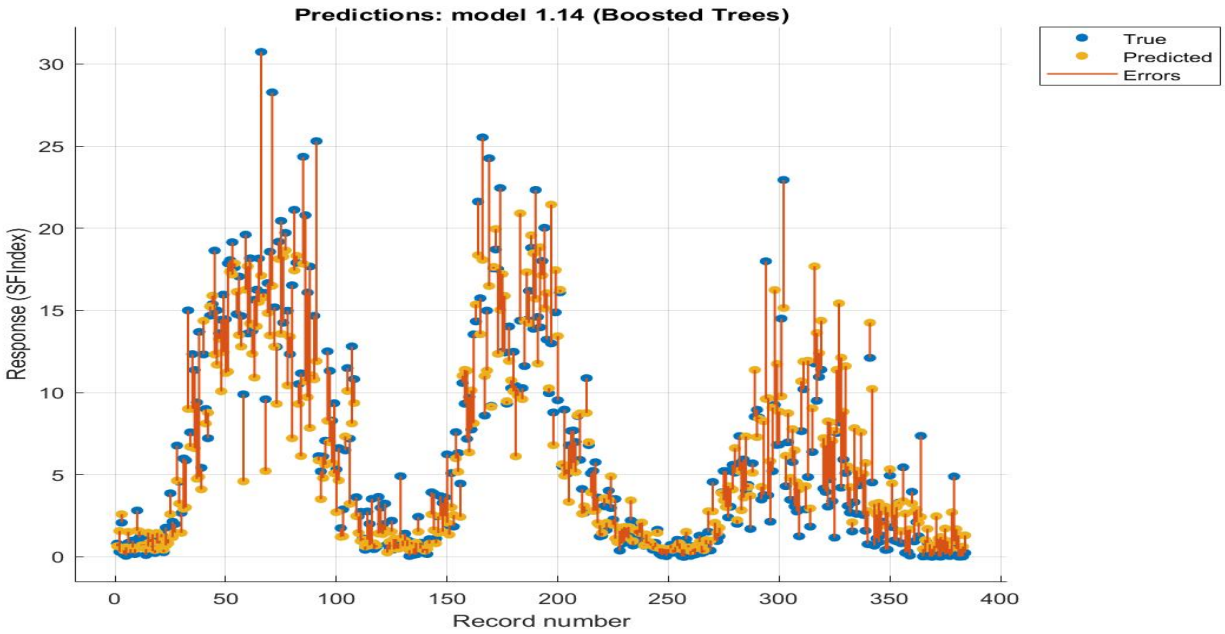



Figure 4.2: Prediction model between SF Index and Record Number.

This is a prediction between SF Index and Record Number. The RED lines show the errors, the BLUE dots show true data points and the YELLOW dots show the predicted points. Record Number allocates the number to the recorded variables of dataset.

## SFI v/s SSA:

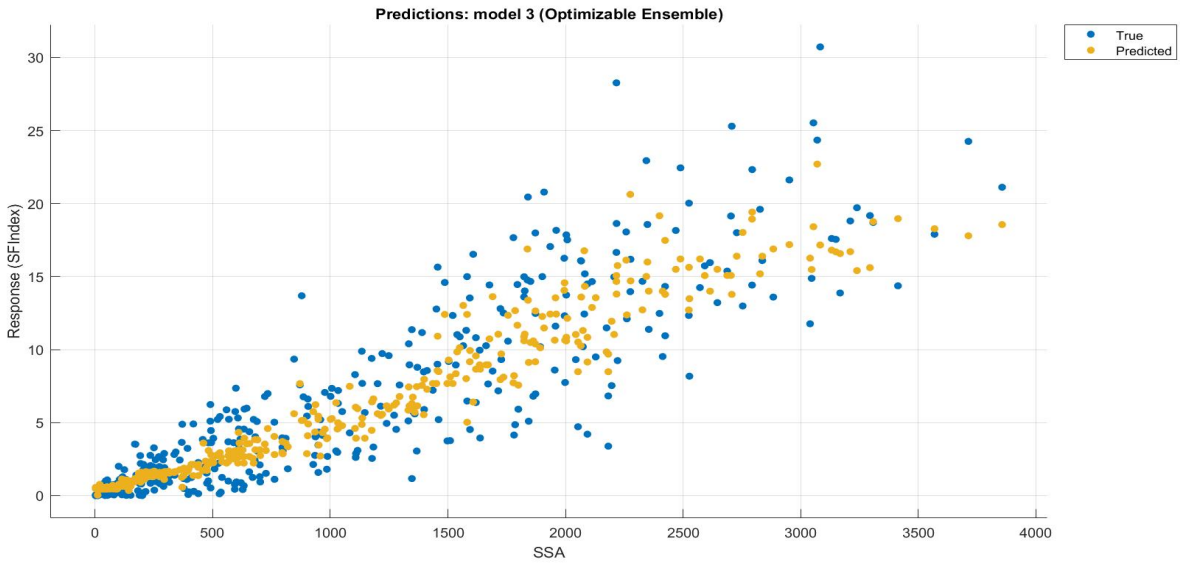



Figure 4.3: Prediction model between SF Index and SSA.

This is a prediction model between SF Index and SSA. We can see the True and Predicted data points represented by the BLUE and YELLOW dots, respectively.

## SF Index v/s SSN:

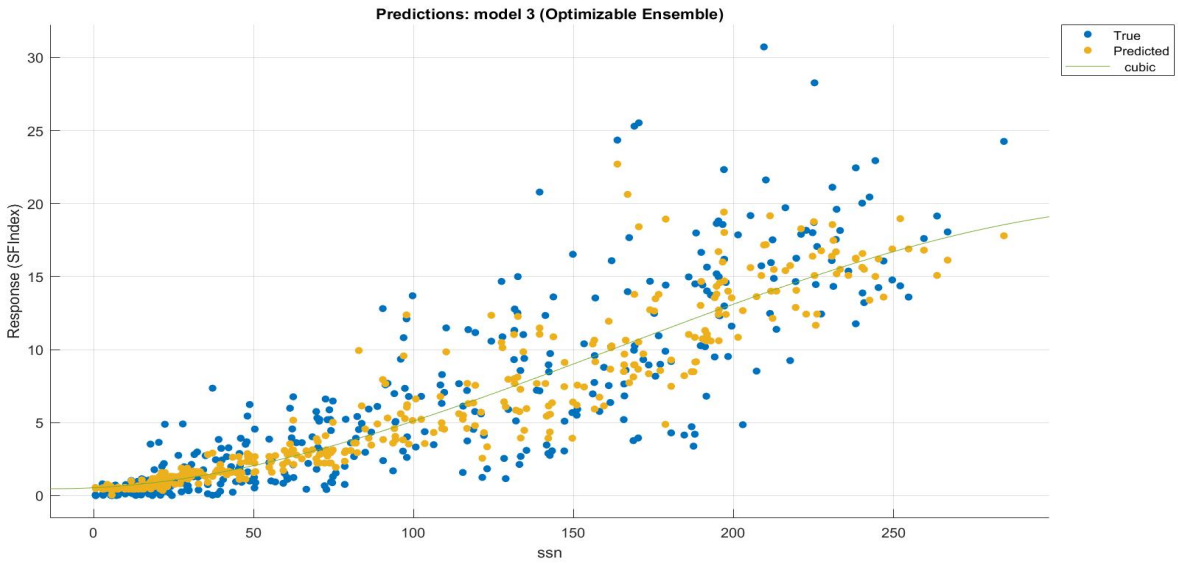



Figure 4.4: Prediction model between SF Index and SSN.

It is the prediction model between SF Index and Sunspot Number(SSN). The GREY line represents the cubic relationship and the BLUE and YELLOW dots represent the True and Predicted data points, respectively.

## Minimum MSE Optimization:

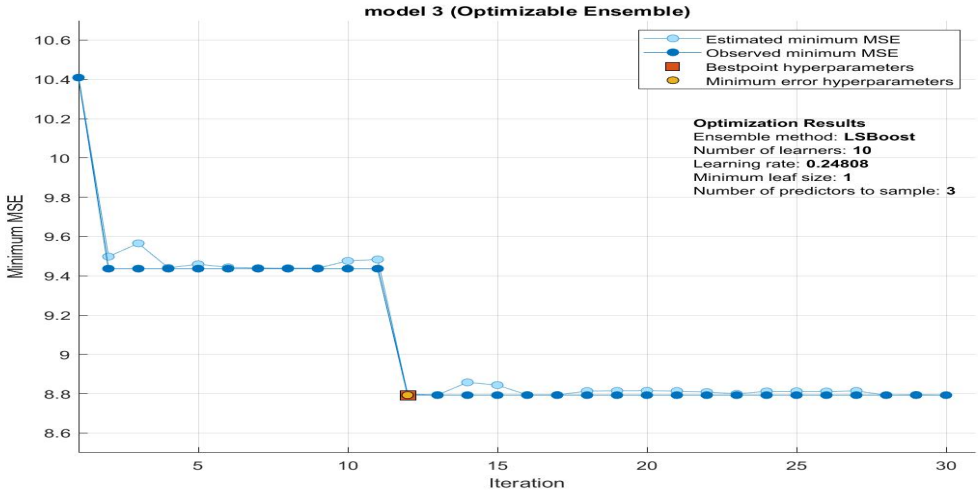



Figure 4.5: Minimum MSE optimization for every iteration of processing or training the dataset(model)

This curve is generated to optimize the minimum MSE for every iteration of processing or training the dataset(model) and find the minimum optimal parameter for MSE(8.79). The error was reduced by optimization. This plot is between minimum MSE and Iteration. This model shows Estimated minimum MSE, Observed minimum MSE, Bestpoint hyperparameters and mimimum error hyperparameters.

## Optimized Ensemble Model between the Predicted response and True response:

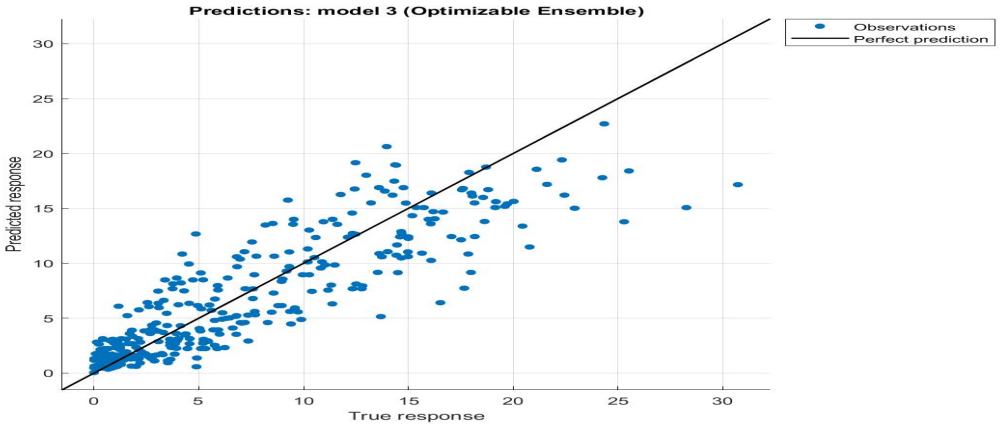



Figure 4.6: Optimized Ensemble Model between the Predicted response and True response

The above is also an optimized ensemble model between the Predicted response and True response. It shows us the observations and the perfect prediction line, represented by the BLUE dots and BLACK line, respectively.

## Residual Plots: Evaluate Model Using Residuals Plot

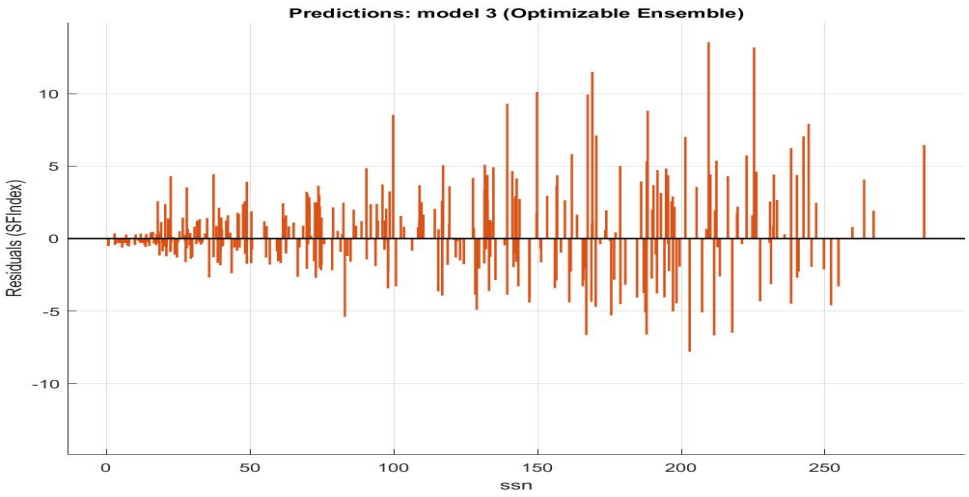



Figure 4.7: Optimized Ensemble Prediction Model between SF Index and SSN

**The residuals plot displays the difference between the predicted and true responses. Usually a good model has residuals scattered roughly symmetrically around 0.**

The above curve is an optimized ensemble prediction model between SF Index and SSN. We can observe that as the graph progresses, it gets wider and towards the end, it becomes narrow again. It is clear that at the ends of the graph, the error is less than that present in the middle portion.

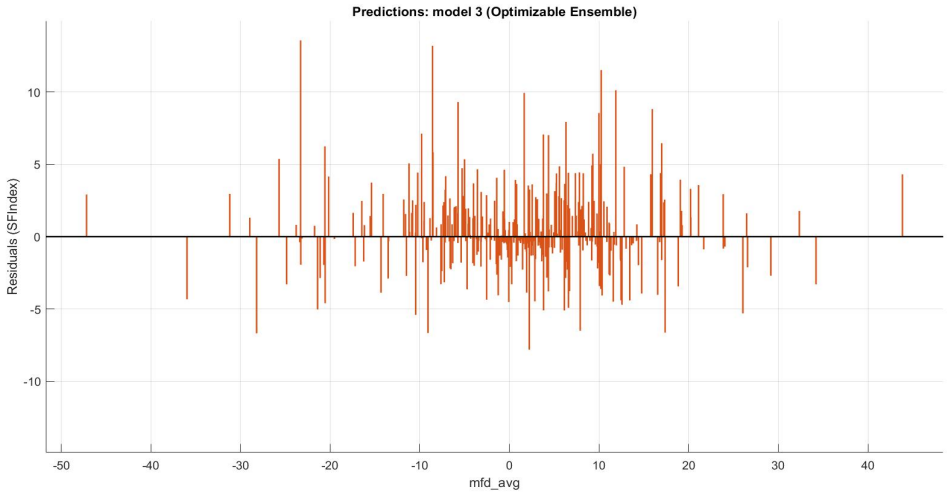



Figure 4.8: Optimized Ensemble Prediction Model between SF Index and Average MFD

This curve is an optimized ensemble prediction model between SF Index and Average MFD. We can observe that as the graph progresses, it gets wider and towards the end, it becomes small again. The error is more in the middle portion than at the ends.

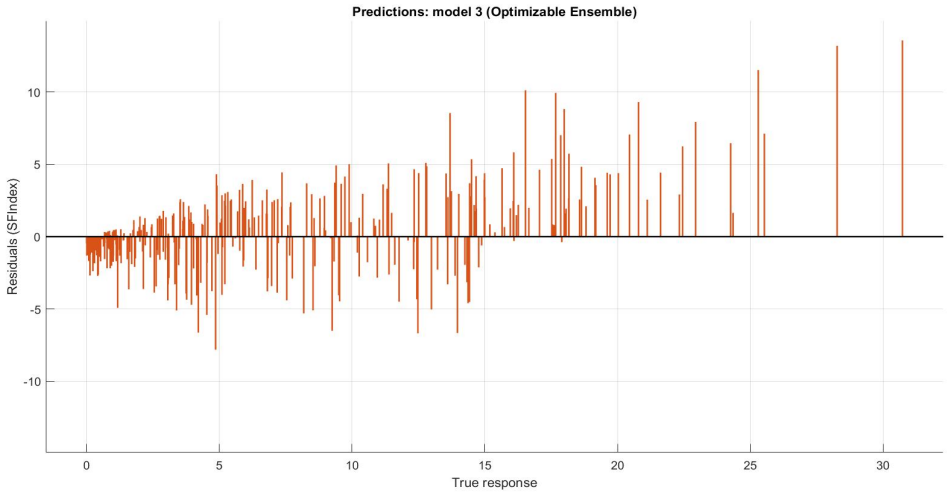



Figure 4.9: Optimized Ensemble Prediction Model between SF Index and True Response

This curve is an optimized ensemble prediction model between SF Index and True Response. As the graph progresses, it gets wider. The error is lesser in the ends as compared to the error in the middle

portion.

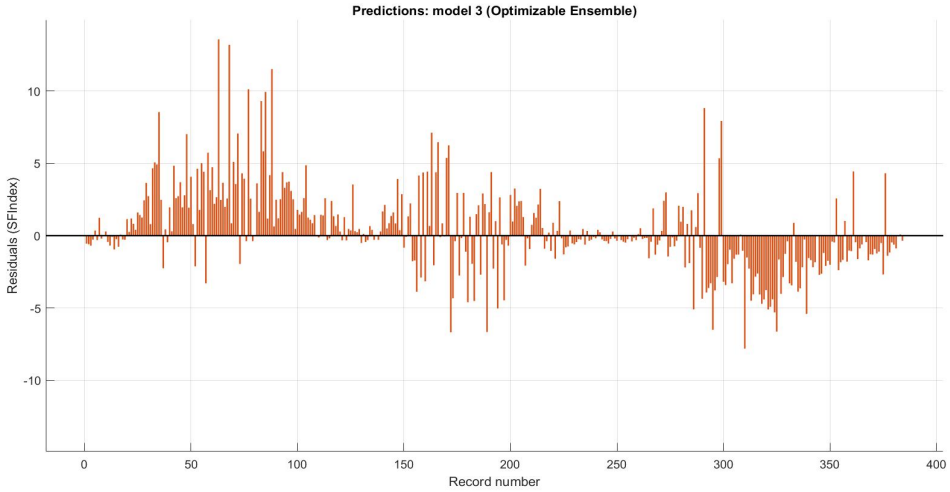



Figure 4.10: Optimized Ensemble Prediction Model between SF Index and Record Number

This curve is an optimized ensemble prediction model between SF Index and Record Number. This is different as it has more error differences at the ends than at the middle. This causes more error at the ends than the middle.

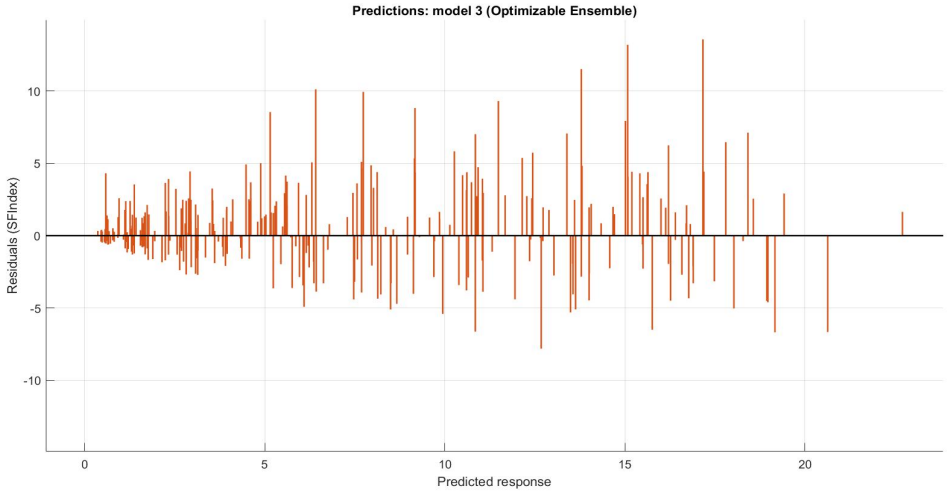



Figure 4.11: Optimized Ensemble Prediction Model between SF Index and Predicted Response

This curve is an optimized ensemble prediction model between SF Index and Predicted Response. We can observe that as the graph progresses, it gets wider and towards the end, it becomes small again. Error differences are more in the beginning.

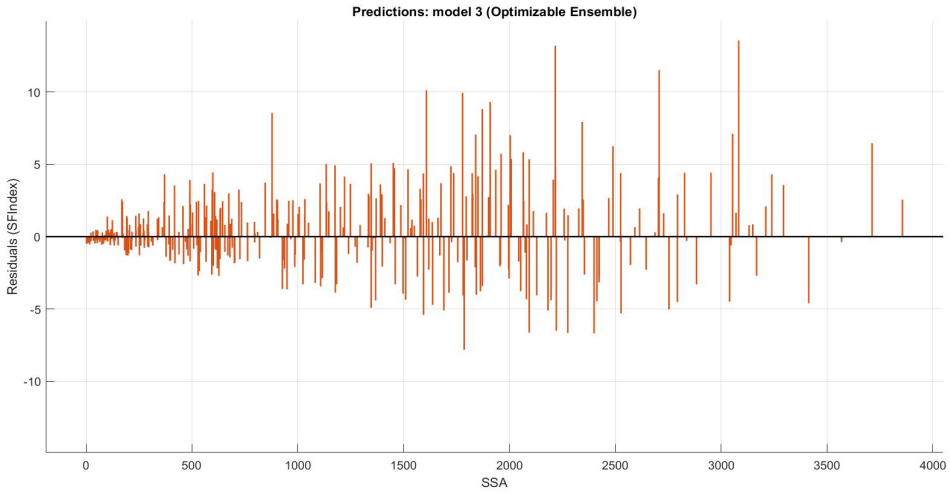



Figure 4.12: Optimized Ensemble Prediction Model between SF Index and SSA

This curve is an optimized ensemble prediction model between SF Index and SSA. We can observe that as the graph progresses, it gets wider. Error differences present in the beginning and the end are less compared to the middle regions.

# Discussions:

1. R-square value of 0.6 or above was the goal of the model and attaining 0.79 as the R-square value shows that the model fits with the data to a fair extent keeping in mind the noise and variation in data.

2. RMSE value in this instance is scale relative. Considering the variance of data and also limitations in data availability, the RMSE value is acceptable and in time, with collection of data, the values can be improved using the same model.

3. Keeping in mind the variations in the data trends, the average magnitude of errors is acceptable. Hence, MAE is in the acceptable range.

4. A comparatively higher value of MSE indicates the highly biased or high variance estimate. This suggests a more refined approach or it is also explained by insufficiency of the data.

5. Since the data was shortened due to excessive refining and combining with the other parameters, the MSE value may be considered to be in the acceptable range. Using ensemble model was to make sure the errors are reduced to the minimum and to optimize the model to the maximum.

6. The cross validation had been done in 5 folds and not more to avoid over fitting of the model. Since the data set was split into 5 models, each of it was used for the training and testing to get the acceptable results. It has also provided much more information on the algorithm performance.

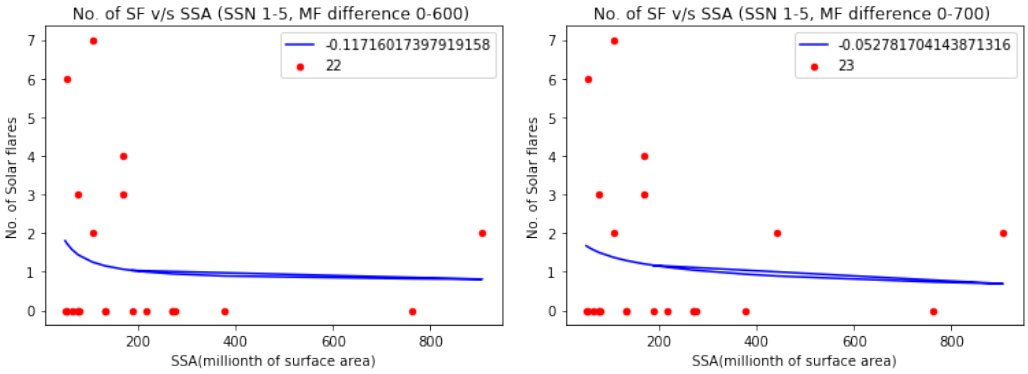
## 4.2 | Objective II: To find the correlation between sunspot area and related parameters

Our objective was to obtain the value of Spearman's coefficient by plotting and curve fitting graphs for various parameters related to CMEs and solar flares with all the possible SSN ranges and magnetic flux differences (MF differences). Since the data for lesser MF difference is not available due to 2-stage refining, the trends are to be observed. If the coefficient values increase as the MF difference range increases, we can extrapolate it back to the lesser MF difference ranges and suggest the corresponding coefficients to be negative. A negative coefficient would mean an inverse relation supporting our hypothesis.

**Graphs were plotted for SSN ranges 1-5 and 6-10 with magnetic flux differences varying from 0-600, 0-700, 0-800. The following were observed:**

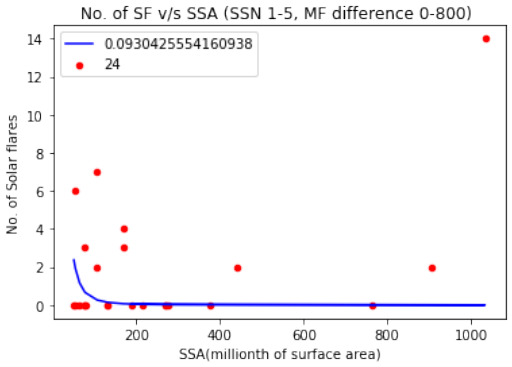
### 4.2.1 | Number of Solar Flares v/s SSA:

#### 4.2.1.1 | SSN 1-5:




a ) MF Difference 0-600

b ) MF Difference 0-700

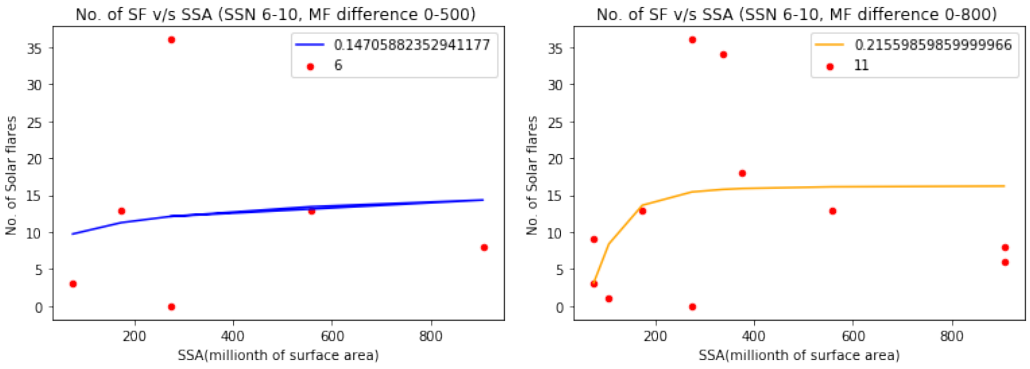



c ) MF Difference 0-800

Figure 4.13: No. Of SF v/s SSA for SSN range 1-5

It is quite evident from the graphs that MF difference affects no. of flares more drastically as compared to ranges of SSN, as Spearman's coefficient value surged from -0.117 to 0.093 over the given range.

#### 4.2.1.2 | SSN 6-10:



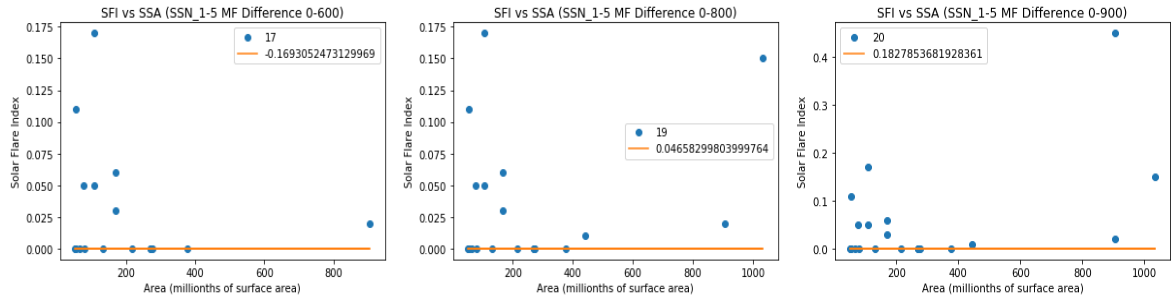

a ) MF Difference 0-500 b ) MF Difference 0-800

Figure 4.14: No. Of SF v/s SSA for SSN range 6-10

Similarly, the correlation coefficient consistently increases from 0.147 to 0.215 as MF difference increase from 0-500 to 0-800 respectively.

### 4.2.2 | Solar Flare Index v/s SSA:

#### 4.2.2.1 | SSN 1-5:

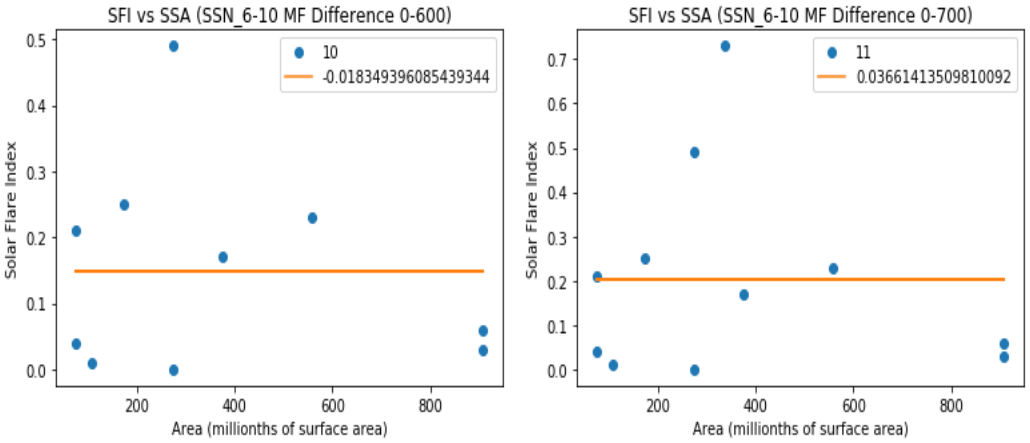



a ) MF Difference 0-600 b ) MF Difference 0-800 c ) MF Difference 0-900

Figure 4.15: SFI v/s SSA for SSN range 1-5

Keeping SSN range constant, the coefficient value gradually increases from -0.169 to 0.182 with increase in MF difference.

#### 4.2.2.2 | SSN 6-10:



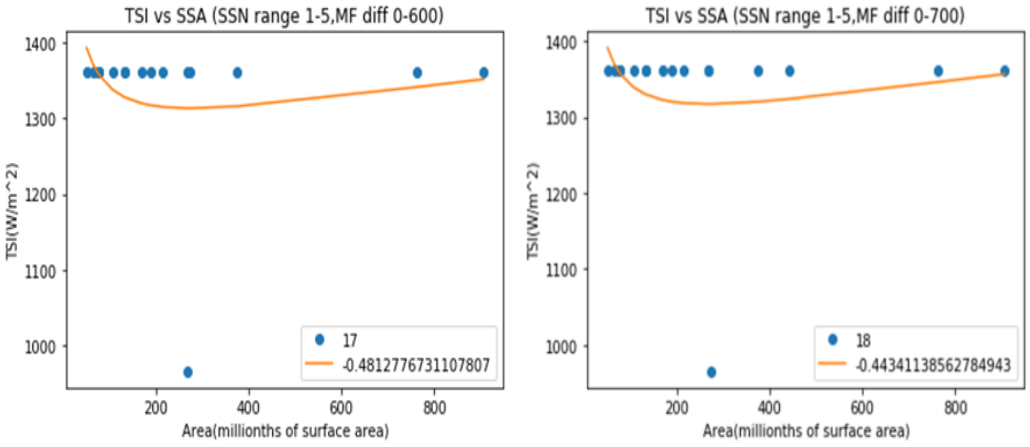

a ) MF Difference 0-600 b ) MF Difference 0-700

Figure 4.16: SFI v/s SSA for SSN range 6-10

The graphs are quite identical with that of SSN range 1-5 by showing a steady increase in coefficient value. The Spearman's coefficient for the MF difference 0-700 increased to 0.0366141 and this value is retained for the higher values of MFD due to lack of data.

### 4.2.3 | Total Solar Irradiance v/s SSA:

#### 4.2.3.1 | SSN 1-5:

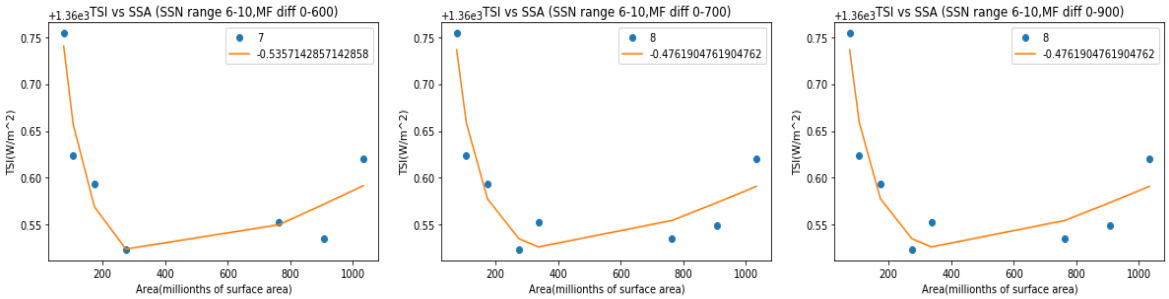



a ) MF Difference 0-600 b ) MF Difference 0-700

Figure 4.17: TSI v/s SSA for SSN range 1-5

The given graphs show the increase in coefficient value from -0.481277 for 0-600 to -0.443411 for 0-700 as MF difference increase.

#### 4.2.3.2 | SSN 6-10:



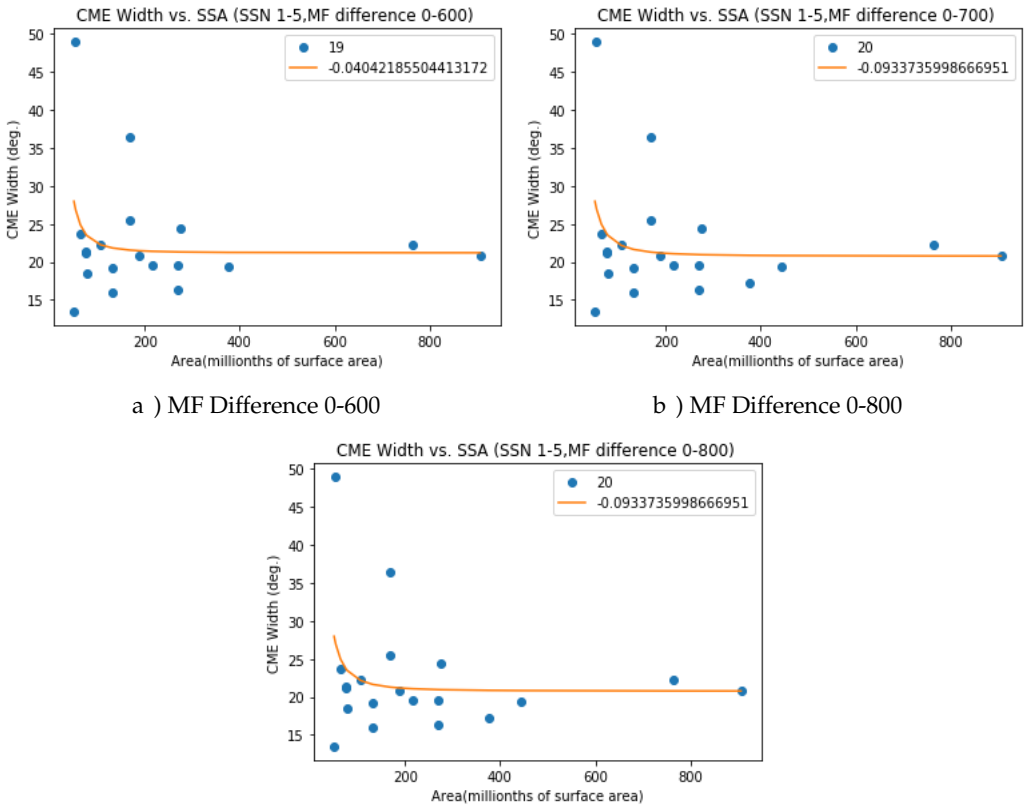

a ) MF Difference 0-600 b ) MF Difference 0-700 c ) MF Difference 0-900

Figure 4.18: TSI v/s SSA for SSN range 6-10

It is clear from the graphs that the Spearman's coefficient increases from -0.535 to -0.476 for the MF differences 0-600 and 0-700. This value remains constant till 0-900 owing to lack of data.

### 4.2.4 | CME Width v/s SSA:

#### 4.2.4.1 | SSN 1-5:



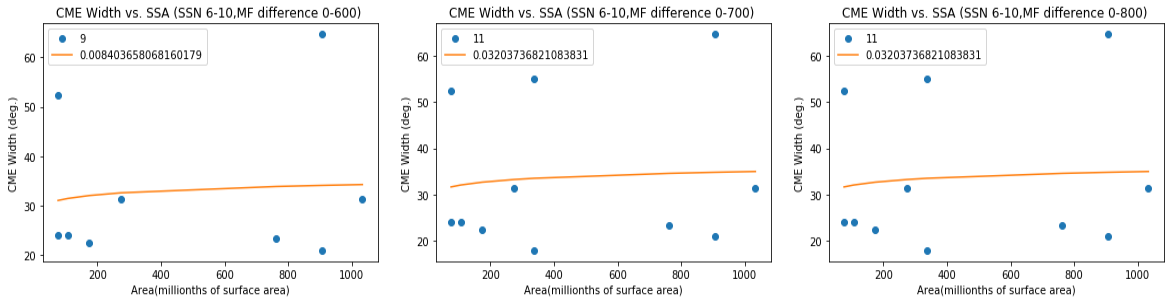

a ) MF Difference 0-600 b ) MF Difference 0-800

c ) MF Difference 0-900

Figure 4.19: CME width v/s SSA for SSN range 1-5

As far as CME Width is concerned, the correlation coefficient decreases from -0.0404 for 0-600 MF difference to -0.0933 for 0-700 MF difference and stays the same for other ranges. This is an anomaly. However, it can be dismissed due to the fact that the coefficient is still negative, clearly indicating an inverse relation between the parameter.

#### 4.2.4.2 | SSN 6-10:



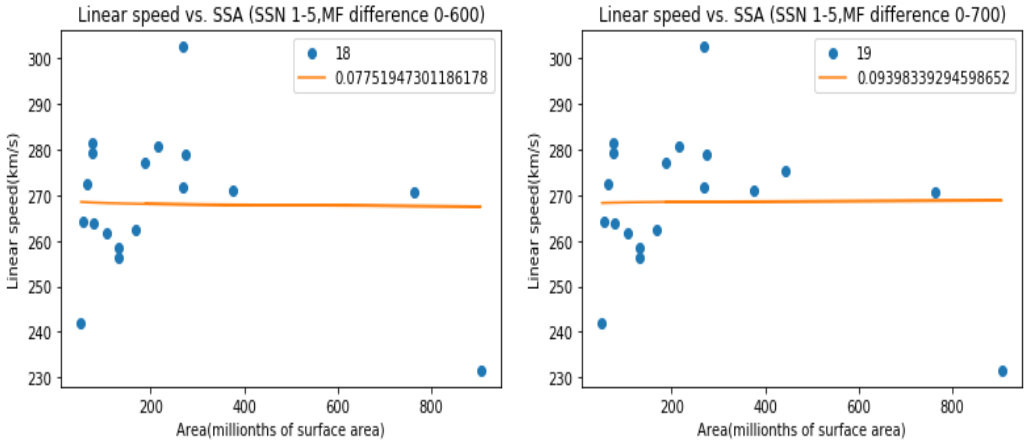

a ) MF Difference 0-600 b ) MF Difference 0-700 c ) MF Difference 0-800

Figure 4.20: CME width v/s SSA for SSN range 6-10

For SSN range 6-10, the coefficient increases from 0.0084 for 0-600 MF difference range to 0.032 for 0-700 MF difference.

### 4.2.5 | Linear speed v/s SSA:

#### 4.2.5.1 | SSN 1-5:

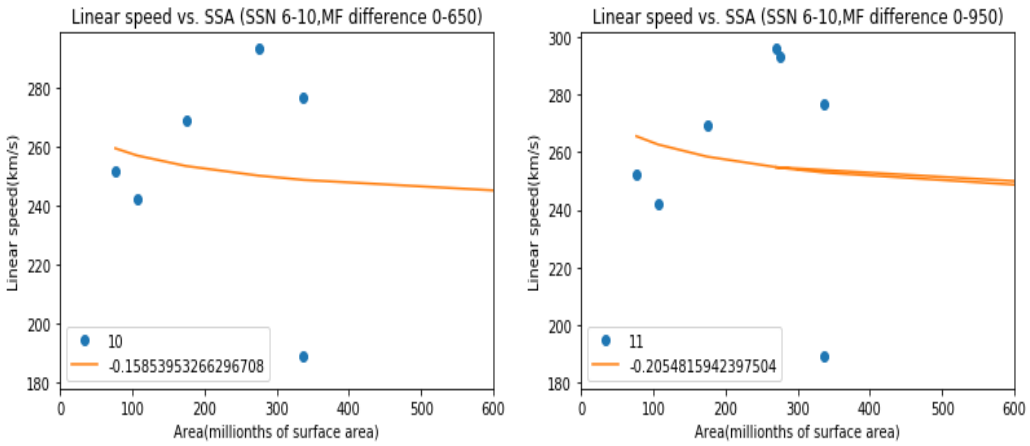



a ) MF Difference 0-600 b ) MF Difference 0-700

Figure 4.21: Linear speed v/s SSA for SSN range 1-5

For CME's linear speeds, as observed from the graphs, the spearman coefficient increases from 0.0775 for MF difference 0-600 to 0.0939 for MF difference 0-700. The coefficient and the curve remain same for the next MF difference ranges due to the lack of data.

#### 4.2.5.2 | SSN 6-10:



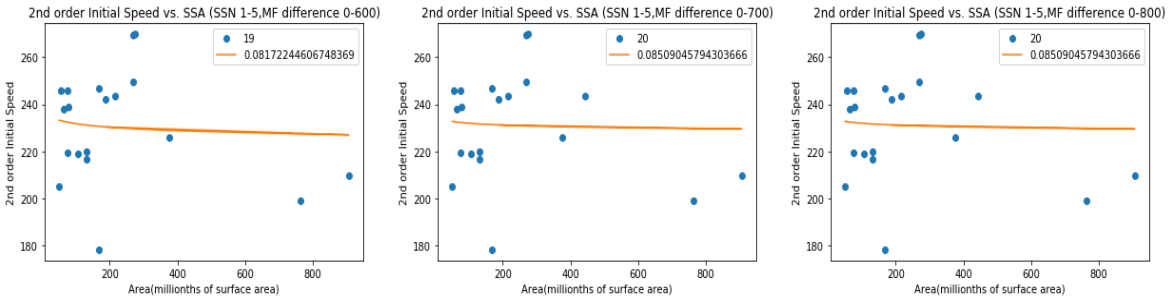

a ) MF Difference 0-650 b ) MF Difference 0-950

Figure 4.22: Linear speed v/s SSA for SSN range 6-10

Similarly, for the SSN range 6-10, we can see the spearman coefficient decreases from -0.15 for 0-650 MF difference to -0.20 for 0-950 MF difference. This is exhibiting an anomalous behaviour. Reasons for this behaviour could be lack of data. The data sets were same for other ranges in between 0-650 and 0-950 MF difference leaving us no room to perform analysis on those.

### 4.2.6 | 2nd order Initial Speed v/s SSA:

#### 4.2.6.1 | SSN 1-5:

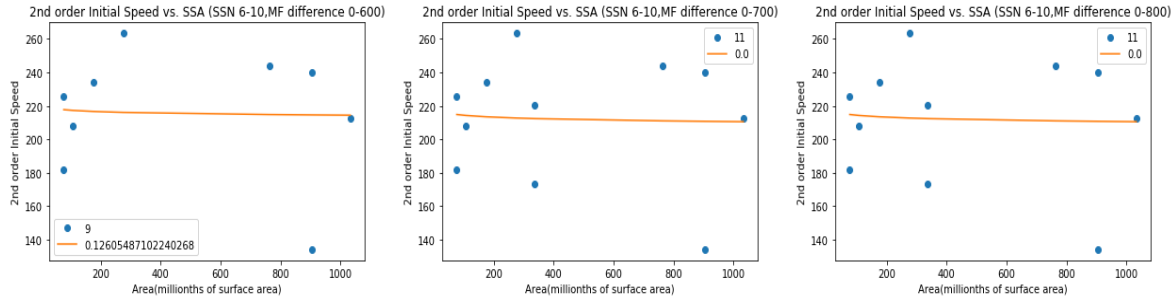



a ) MF Difference 0-600 b ) MF Difference 0-700 c ) MF Difference 0-800

Figure 4.23: 2nd order Initial Speed v/s SSA for SSN range 1-5

From the graph, we can see that there is a small increase in the correlation coefficient from 0.0817224 to 0.0850904. The coefficient remains 0.0850 for the next ranges due to lack of data.

#### 4.2.6.2 | SSN 6-10:



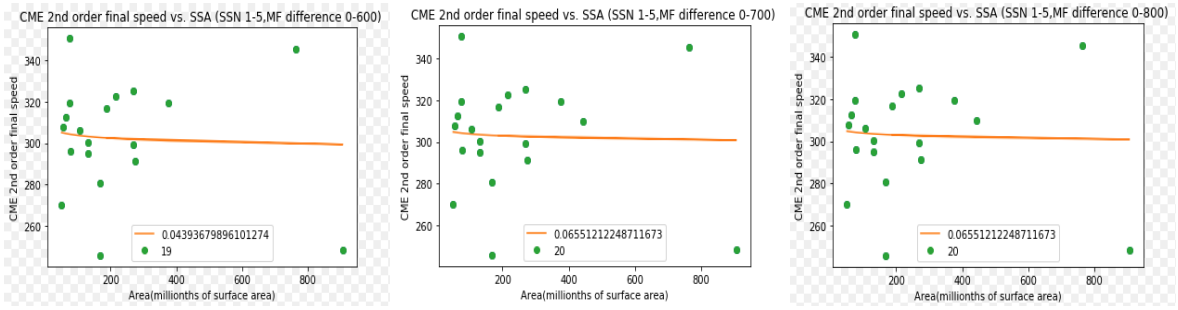

a ) MF Difference 0-600 b ) MF Difference 0-700 c ) MF Difference 0-800

Figure 4.24: 2nd order Initial Speed v/s SSA for SSN range 6-10

Here, we can observe that the correlation coefficient decreases from 0.12605 to 0.0. This is an anomaly which can be due to the lack of data for this SSN range. The magnetic flux might be another factor causing the anomaly.

### 4.2.7 | 2nd order Final Speed v/s SSA:

#### 4.2.7.1 | SSN 1-5:

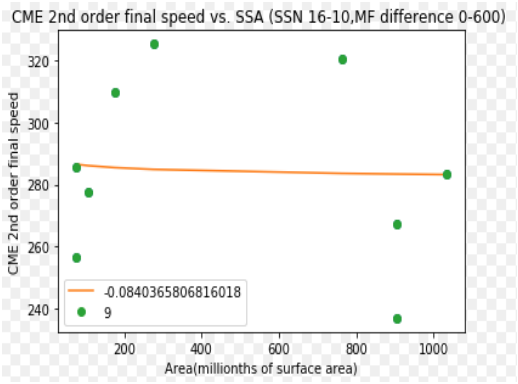



a ) MF Difference 0-600 b ) MF Difference 0-700 c ) MF Difference 0-800

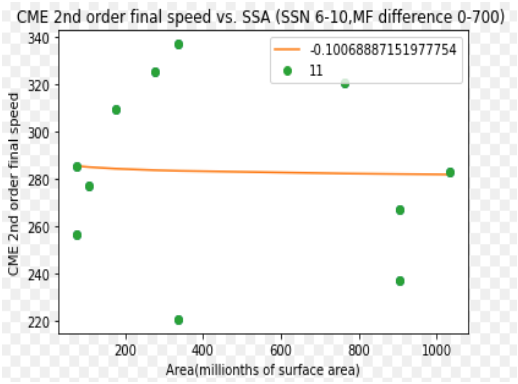
Figure 4.25: 2nd order Final Speed v/s SSA for SSN range 1-5

It can be observed that there is an increase in the value of the coefficient from 0.0439 for 0-600 to 0.0655 for 0-700. The value remains the same for the further ranges due to the lack of data.

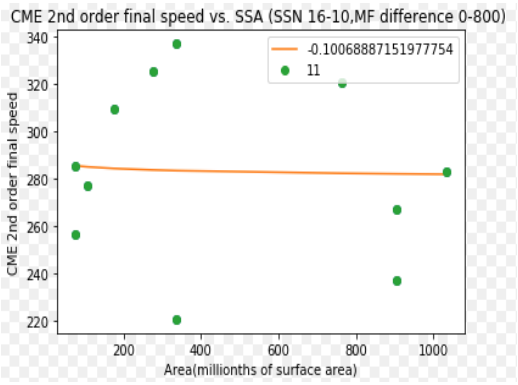
#### 4.2.7.2 | SSN 6-10:




a ) MF Difference 0-600




b ) MF Difference 0-700




c ) MF Difference 0-800

Figure 4.26: 2nd order Final Speed v/s SSA for SSN range 6-10

The graphs clearly suggest that the coefficient decreases here, from -0.0840365 for 0-600 to -0.1006 for 0-700, after which it remains constant. This decrease is again, an anomaly which might be occurring due to the lack of data.

# 5

# Conclusions

## 5.1 | Objective I:To develop a predictive model for the energy of Solar Flares using Machine Learning

1. As we obtained R-square value of 0.79, which is well above the required conditions, we can say that the model fits with the data, while considering the noise and variation in data.
2. The RMSE value obtained (2.9653) is scale relative and acceptable. With more data collection and time, these values can be well improved
3. The average magnitude of errors is acceptable, while considering the variations in data trends. Thus, MAE value (2.0799) is in the acceptable range.
4. A high variance estimate is indicated by a comparatively higher value of MSE (8.7929). This might be explained by the insufficiency of the data or requires a more refined approach.
5. It can be concluded by saying that if more parameters and data sets were available, predictions for SFI could have been more precise
6. This model initially didn't give desirable results, but it was after this that we found out about ensemble learning. We used ensemble learning as the data was very scattered and bad in a few places. Ensemble predictions have helped this model immensely.

## 5.2 | Objective II:To find the correlation between Sunspot Area and related parameters

1. The coefficient values are not negative for all the parameters. The reason is the effect of magnetic flux. In an ideal case, same value of magnetic flux would not affect the results yielding a perfect negative correlation coefficient.
2. It can be concluded that all the graphs which have the coefficient values increasing along with the MF difference can be extrapolated to lower MF difference values and corresponding coefficients can be suggested to be negative. The lower coefficient values would mean a negative coefficient which suggest that the relation between the parameters and SSA is an inverse relation.
3. Some anomalies were also observed. The reason for that might be lack of data and same dataset for multiple ranges. The difference in data sets for different ranges could have helped in understanding

the behavior more efficiently.
4. Out of 7 parameters, 2 ranges of SSN for each parameter i.e. 14 trends, only 4 trends were anomalous. This gives a rough accuracy of 71.428%.

## 5.3 | FUTURE SCOPE

### 5.3.1 | **Objective I:**To develop a predictive model for the energy of Solar Flares using Machine Learning

Prediction of Solar Flares is a very important factor in understanding the space weather. The sudden eruptions of the flares and surges in amount of energetic particles make the satellites and the astronauts vulnerable. The proposed and rendered model to predict the Solar Flare Index (SFI), which is a measure of energy emitted by the flare can be used to foresee the impact of the flare. Hence, giving us time to brace for impact and take necessary precautions. The model can be incorporated with a website and make it open for external use.
From a research perspective, other parameters can be added to the model and increase the accuracy. One of the important parameter can be magnetic field strengths which can be achieved by studying the magnetograms. Further, the method of building such model can be used and applied to other parameters like duration of flares, time of occurrence etc.

### 5.3.2 | **Objective II:**To find the correlation between Sunspot Area and related parameters

The accomplished results can be used as basis for further research. An equation between the parameters and SSA can be formed which can be used to make the results more concrete. This equation would not only relate the parameter with SSA but also the parameter with each other.
With the existing parameters, new parameters like geomagnetic field strengths can be added in order to understand the impact of sun on Earth. Another potential candidate as a parameter can be the magnetic field strength of sun. Magnetograms can be studied and the variation with SSA can help us solidify the correlation.

# 6

# References

## 6.1 | Background & Literature survey


1. How the Sun's magnetic field works:
   https://docs.sunpy.org/en/stable/
2. Understanding the magnetic Sun:
   https://www.nasa.gov/feature/goddard/2016/understanding-the-magnetic-sun2016.
3. Differential rotation:
   https://astronomy.swin.edu.au/cosmos/D/Differential+Rotation
4. Solar Magnetic Field:
   https://www.sciencedirect.com/topics/earth-and-planetary-sciences/solar-magnetic-field
5. Babu Ram Tiwari 'The Solar Flux and Sunspot Number; A long trend analysis'
   https://journals.aijr.in/index.php/ias/article/download/751/172/2018.
6. Sunspots:
   https://en.wikipedia.org/wiki/Sunspot#:~:text=Sunspots%20are%20temporary%20phenomena%20on,field%20flux%20that%20inhibit%20convection
7. What is the Sunspot Number:
   https://www.sws.bom.gov.au/Educational/2/3/3
8. Coronal Mass Ejections:
   https://en.wikipedia.org/wiki/Coronal_mass_ejection#:~:text=A%20coronal%20mass%20ejection%20(CME,be%20observed%20in%20coronagraph%20imagery
9. Onuchukwu Chika Christian 'A statistical analysis of sunspot and CME parameters for the Solar cycle 23.'
   http://medcraveonline.com/PAIJ/PAIJ-02-00103.pdf 2018.
10. K. B. Ramesh 'Coronal Mass Ejections and Sunspots-Solar Cycle Perspective'
    https://iopscience.iop.org/article/10.1088/2041-8205/712/1/L77/pdf 2010.

11. SOHO LASCO CME CATALOG:
https://cdaw.gsfc.nasa.gov/CME_list/catalog_description.htm

12. What is a solar flare:
https://www.nasa.gov/content/goddard/what-is-a-solar-flare

13. Correlations between flare parameters magnetic parameters in Solar Flares:
https://academic.oup.com/pasj/article/61/1/75/1501202

14. Probhas Raychaudhuri 'Total Solar Irradiance Variability and the Solar Activity Cycle'
https://arxiv.org/ftp/astro-ph/papers/0601/0601335.pdf

15. SK Solanki 'How much of the Solar irrandiance variations is caused by the Magnetic Field at the Solar surface?
https://www.sciencedirect.com/science/article/pii/S0273117702002387 2002.

16. Machine learning with Python:
https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_ecosystem.htm

17. Introduction to Machine Learning using Python:
https://www.geeksforgeeks.org/introduction-machine-learning-using-python/#:~:text=To%20Prevent%20It-,Introduction%20To%20Machine%20Learning%20using%20Python,when%20exposed%20to%20new%20data.

18. What is Machine Learning:
https://jakevdp.github.io/PythonDataScienceHandbook/05.01-what-is-machine-learning.html

19. Scatter Plot Matrix:
https://pro.arcgis.com/en/pro-app/help/analysis/geoprocessing/charts/scatter-plot-matrix.htm#:~:text=A%20scatter%20plot%20matrix%20is,be%20explored%20in%20one%20chart.
https://www.itl.nist.gov/div898/handbook/eda/section3/scatterb.htm

20. Curve and Surface Fitting:
https://www.originlab.com/index.aspx?go=Products/Origin/DataAnalysis/CurveFitting

21. What is Root Mean Square Error?
https://www.statisticshowto.com/probability-and-statistics/regression-analysis/rmse-root-mean-square-error/#:~:text=Root%20Mean%20Square%20Error%20(RMSE)%20is%20the%20standard%20deviation%20of,the%20line%20of%20best%20fit.

22. Goodness of Fit statistics:
https://web.maths.unsw.edu.au/~adelle/Garvan/Assays/GoodnessOfFit.html#:~:text=SSE%20is%20the%20sum%20of,accounted%20for%20by%20the%20model.

23. Coefficient of Determination:
https://en.wikipedia.org/wiki/Coefficient_of_determination

24. Correlation Coefficient:
https://www.investopedia.com/terms/c/correlationcoefficient.asp

25. Spearman's rank correlation coefficient:
https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient

26. Ensemble methods:
https://scikit-learn.org/stable/modules/ensemble.html

27. Understanding Gradient Boosting Machines:
https://towardsdatascience.com/understanding-gradient-boosting-machines-9be756fe76ab

## 6.2 | Data Collection

28. SDO Data:
https://sdo.gsfc.nasa.gov/data/

29. Sunspot number:
http://www.sidc.be/silso/datafiles

30. Sunspot number graphics:
http://www.sidc.be/silso/ssngraphics

31. Sunspot numbers and areas:
http://spaceacademy.net.au/asa/solar/ssdata.htm

32. Sunspot data:
https://solarscience.msfc.nasa.gov/greenwch.shtml

33. Comparison of sunspot area data bases:
https://academic.oup.com/mnras/article/323/1/223/1002362

34. Monthly reports and data on Solar flare events took place over a period of time:
https://www.ngdc.noaa.gov/stp/solar/solarflares.html

35. Daily total sunspot number:
http://www.sidc.be/silso/infosndtot

36. Comparison of Sunspot area data:
https://academic.oup.com/mnras/article/323/1/223/1002362

37. Solar activity parameters and data sources:
https://shodhganga.inflibnet.ac.in/bitstream/10603/105499/7/07_chapter%202.pdf

38. Daily 10.7cm Flux values:
https://www.spaceweather.gc.ca/solarflux/sx-5-flux-en.php

39. Monthly 10.7cm Flux values:
https://www.spaceweather.gc.ca/solarflux/sx-5-mavg-en.php

40. Rotational averages of Solar 10.7cm flux:
https://www.spaceweather.gc.ca/solarflux/sx-5-ravg-en.php

41. Solar Flare data:
https://www.ngdc.noaa.gov/stp/solar/solarflares.html

42. Model specification: Choosing the correct regression model:
https://statisticsbyjim.com/regression/model-specification-variable-selection/

43. CME Data:
https://cdaw.gsfc.nasa.gov/CME_list/

44. Solar irradiance:
https://www.nasa.gov/mission_pages/sdo/science/Solar%20Irradiance.html

45. Data for Magnetic flux density values:
http://wso.stanford.edu/#Synoptic

46. CACTus catalogue (CME data):
http://sidc.oma.be/cactus/catalog.php

47. CDAW data catalogue (CME data):
https://cdaw.gsfc.nasa.gov/CME_list/

48. Latest CME detection-updated every 6 hours:
http://sidc.oma.be/cactus/

49. Sunspot area data (full sun and each hemisphere-wise)- (1874-2016):
https://solarscience.msfc.nasa.gov/greenwch.shtml

50. Magnetic field values:
http://wso.stanford.edu/#Synoptic(1975-2020)

51. Sunspot number and area vs time (2007-2020):
http://spaceacademy.net.au/asa/solar/ssdata.htm

52. SSN (1700-2014): https://www.sws.bom.gov.au/Educational/2/3/6

53. Daily, mean, 13-month smoothed, yearly average SSN data and plot (vs. time):
http://www.sidc.be/silso/datafiles#total

54. Relation Between solar wind parameters, coronal mass ejections and sunspot number:
https://www.ijeas.org/download_data/IJEAS0409039.pdf

55. Parameters for data:
http://solarcyclescience.com/forecasts.html#Cycle24Prediction

56. Sunspot Data:
https://solarscience.msfc.nasa.gov/greenwch.shtml

57. Sunspot Region Data:
https://www.ngdc.noaa.gov/stp/solar/sunspotregionsdata.html

58. Comparison of sunspot area data bases:
https://academic.oup.com/mnras/article/323/1/223/1002362

59. Sunspot Number Data:
https://www.ngdc.noaa.gov/stp/solar/ssndata.html

60. Sunspot Numbers:
https://www.sws.bom.gov.au/Solar/1/6

61. Daily sunspot number since 1818:
https://www.datastro.eu/explore/dataset/daily-sunspot-number/export/

62. Inter-cycle variations of a Solar Irradience(Sunspots as a Pointer):
http://www2.mps.mpg.de/dokumente/publikationen/solanki/j71.pdf

63. Royal Observatory Database:
https://solarscience.msfc.nasa.gov/greenwch.shtml

64. Data Clustering of CMEs:
https://iopscience.iop.org/article/10.3847/1538-4365/aab76f/pdf

65. Mean Sunspot Numbers from 1700 to 2015 (yearly):
https://www.sws.bom.gov.au/Educational/2/3/6

66. Daily, mean, 13-month smoothed, yearly average SSN data and plot (vs. time):
http://www.sidc.be/silso/datafiles#total

67. Solar flare index data (1976-2014):
https://www.ngdc.noaa.gov/stp/space-weather/solar-data/solar-features/solar-flares/index/flare-index/-

68. Data from 2002-2016:
https://www.kaggle.com/khsamaha/solar-flares-rhessi?select=hessi.solar.flare.2002to2016.csv

69. Background data sources for solar flares:
https://solarflare.njit.edu/datasources.html

70. Daily, mean, 13-month smoothed, yearly average SSN data and plot (vs. time):
http://www.sidc.be/silso/datafiles

71. Coronal index for years 1939-2008:
https://www.ngdc.noaa.gov/stp/space-weather/solar-data/solar-indices/solar_corona/documentation/readme_solar-corona.pdf

## 6.3 | Other references


72. M. Neugebauer 'Spatial Structure of Solar Wind and Comparisons with solar data and models' https://agupubs.onlinelibrary.wiley.com/doi/epdf/10.1029/98JA007981998.

73. H. Gleisner 'Predicting geomagnetic storms from solar-wind data using time-delay neural networks' http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.57.1366&rep=rep1&type=pdf1996.

74. Emilia K. J. Kilpua 'Magnetic field fluctuation properties of coronal mass ejection-driven sheath regions in the near-Earth solar wind' https://angeo.copernicus.org/preprints/angeo-2020-17/angeo-2020-17.pdf

75. Fredrik Boberg 'Real time Kp predictions from solar wind data using neural networks https://www.sciencedirect.com/science/article/abs/pii/S14641917000001672000.

76. Joseph E. Borovsky 'What Magnetospheric and Ionospheric Researchers Should Know about the Solar Wind' https://www.sciencedirect.com/science/article/pii/S13646826203008822020.

77. Fei Xu 'A new four-plasma categorization scheme for the solar wind' https://agupubs.onlinelibrary.wiley.com/doi/full/10.1002/2014JA020412#:~:text=1.1%20The%20Plasma%20Types%20in,as%20coronal%20mass%20ejections%20%5Bcf. 2014.

78. More about the Sun: http://solar.physics.montana.edu/YPOP/Spotlight/SunInfo/Structure.html

79. Photospheric features: https://solarscience.msfc.nasa.gov/feature1.shtml

80. J. O. Stenflo 'Differential rotation of the Sun's magnetic field pattern': http://adsabs.harvard.edu/full/1989A%26A...210..403S 1988.

# Index

Coronal Mass Ejections, 5
Curve fitting , 7

Ensembled Methods, 8

Machine Learning, 6
Magnetic Field of the Sun, 3

Scatterplot Matrix , 8
Solar Flares, 4
Sunspots, 4

Total Solar Irradiance (TSI), 6