Dependency Analysis of Solar Flares and Prediction of Solar Activity

Project Report for Indian Institute of Technology, Bombay - DS203: Programming for Data Science (2022)

Ravi Kumar
Dept. of Aerospace Engineering
Indian Institute of Technology, Bombay
Mumbai, Maharashtra
210010052@iitb.ac.in

Shravya Suresh
Dept. of Engineering Physics
Indian Institute of Technology, Bombay
Mumbai, Maharashtra
210260046@iitb.ac.in

Saurabh Prajapati
Dept. of Aerospace Engineering
Indian Institute of Technology, Bombay
Mumbai, Maharashtra
210010057@iitb.ac.in

***Abstract*—The sun is the ultimate source of energy for most life on earth. It warms the atmosphere and supplies energy use to grow. But the sun sometimes releases huge bursts of electrified gases into space. These bursts are called coronal mass ejections (CMEs) or solar flares. A solar flare is a tremendous explosion on the Sun that happens when energy stored in 'twisted' magnetic fields (usually above sunspots) is suddenly released. When they are directed towards the Earth, they can either wreak havoc or generate auroras, the spectacular atmospheric displays also known as "northern lights". The goal of this project is to determine whether there is a correlation between solar flare activity and the solar sunspot cycle, using historical data. We analyse the duration and energies of solar flares detected over a period of time and draw graphical depictions of their trends. We then train machine learning models to predict future solar flare. The data is sourced from the Laboratory for Experimental Astrophysics, Ioffe Institute and Kaggle.**

I. Introduction

At over 1.4 million kilometers (869,919 miles) wide, the Sun contains 99.86 percent of the mass of the entire solar system: well over a million Earths could fit inside its bulk. The total energy radiated by the Sun averages 383 billion trillion kilowatts, the equivalent of the energy generated by 100 billion tons of TNT exploding each and every second.

But the energy released by the Sun is not always constant. Close inspection of the Sun's surface reveals a turbulent tangle of magnetic fields and boiling arc-shaped clouds of hot plasma dappled by dark, roving sunspots.

Once in a while — exactly when scientists still cannot predict — an event occurs on the surface of the Sun that releases a tremendous amount of energy. This energy is released in the form of a solar flare or a coronal mass ejection, an explosive burst of very hot, electrified gases with a mass that can surpass that of Mount Everest. Coronal mass ejections can not only put on a spectacular light show, they can also wreak havoc with earth-orbiting satellites and sometimes even ground-based electrical systems.

Sunspots are another solar phenomenon that have a much longer history of scientific study than CMEs. Sunspots were first discovered by Galileo Galilei in 1612, who made regular observations of features on the surface of the sun, which moved as the sun rotated. Sunspots consist of concentrations of strong magnetic flux. They usually occur in pairs or groups of opposite polarity that move in unison across the face of the Sun as it rotates.

The solar sunspot cycle has been observed for hundreds of years, a long time span compared to a human life, but not even an eye blink compared to the life of the sun (4.5 billion years, and slowly counting). Nevertheless, at least at this point in time, the sunspot cycle appears to be a robust phenomenon. A question that immediately jumps to mind is "What about solar flares? Do their numbers rise and fall like the sunspot cycle?"

The existing research on this dependence piqued our curiosity and motivated us to take up this topic. We decided to study the dependency and relationships of solar flares on sunspots and any surface activity on the sun. This led us to analyse correlations between solar activity and solar flares over time. We utilised measurements like duration of the flare, energy and time of occurrence to derive conclusions.

Our data primarily comes from two sources. One dataset has been taken from the Konus-Wind Solar Flare Database while the other dataset has been borrowed from Kaggle. We have linked together these data sources.

In this project we have performed the following operations:

- We have extensively pre-processed and cleaned both datasets to concise them down to their context, making them easy to use and understand
- We have analysed the correlation between various variables to understand how one factor affects another
- We have plotted the distribution of energy of solar flares and analysed the frequency count of flares of different energy ranges over a period of time
- We have trained two machine learning models on both the datasets and compared their relative performance in predicting future solar activity

Our key goal was to identify a dependable relationship between sunspots and solar flares to help make accurate predictions of the same. However, with the need of significantly more data and longer periods of processing to make accurate

predictions, this analysis is a basic representation of the higher work going on in this field.

II. Prior Work

A keen interest in studying clestial bodies and astronomical phenomena has always been there since time immemorial. A significant amount of research and calculative analysis has been done on solar activity. The RHESSI (Reuven Ramaty High Energy Solar Spectroscopic Imager) Mission utilised NASA's RHESSI solar flare observatory. Its primary mission was to explore the physics of particle acceleration and energy release in solar flares. Besides this, the Laboratory for Experimental Astrophysics, Ioffe Institute has been consistently collecting and analysing solar activity. NASA too has been working extensively in this field.

III. Datasets and Methodology

This project deals with analysing the dependence of solar activity on coronal mass ejections between the years 2002 and 2018. We primarily used data concerned with solar flares, energy, radial offset and timings for all solar flares detected during the given period of time.

A. Datasets

RHESSI Mission Dataset: We have sourced a large dataset on solar flares from this database by NASA, through Kaggle. It is an elaborate dataset that provides information about the time of occurrence, duration, peak counts and energy of solar flares that were delected between 2002 and 2018. We used this dataset to analyse relationships between different pairs of variables and draw conclusions regarding the same. We also split this data into training, validation and testing data, and trained machine-learning models on it to test the accuracy of the models in predicting future solar activity.

KW-Sun: Konus-Wind Solar Flare Database: This is a smaller dataset than the one above, and has been taken from the said database of the Laboratory for Experimental Astrophysics, Ioffe Institute. It provides concise information about the time of occurence, class of the flare, duration and peak timings that were delected between 1994 and 2022. We used this dataset as well to draw conclusions based on the number of flares observed in each year. Following similar suit as above, we split this data into training, validation and testing data, and trained the same machine-learning models on it to test their accuracy in predicting future solar activity.

B. Data Pre-Processing

- We began the project with preliminary preparation of the data for analysis and prediction. We started by cleaning our data and reducing it to the crucial and relevant information required. To do this, we identified and eliminated those columns from our datasets that did not have sufficient data to draw conclusions. In doing so, we only retained the columns that provided abundant information while discarding the ones that did not.
- We also identified and dropped those columns and rows that had excessive null value (NaN) entires. The presence of NaN values hinders the process of data analysis by contributing to faulty readings and unsuccessful interpretation and visualisation. Hence, they were immediately discarded.
- Some datatypes of the columns of the filtered datasets were changed to be compatible with the type of data contained in them. For example, the date in both datasets was converted into the suitable 'datetime' format.
- Columns were also renamed for easier readibility. Renaming columns helps to better convey the importance of the data contained in them, allowing quicker accessibility as well.
- Finally, the columns of both datasets were segregated into discrete and continuous variables. Identifying the type of variable represented by a column helps in performing the appropriate data analysis and operation on the data. Discrete and continuous data need to be processed differently; for example, string data cannot be used in calculating mean square error. Thus, classifying the variables as discrete and continuous helps to identify the type of operations and analysis to perform on them.

Having pre-processed the data, we then performed exploratory data analysis on them.

IV. Data Visualisation

A. Variable distribution

The datasets used by us are significantly large. So we first identified what mode of analysis we were to use to make sense of the processed data. We began with studying the frequency variation of the discrete variables of the RHESSI dataset.

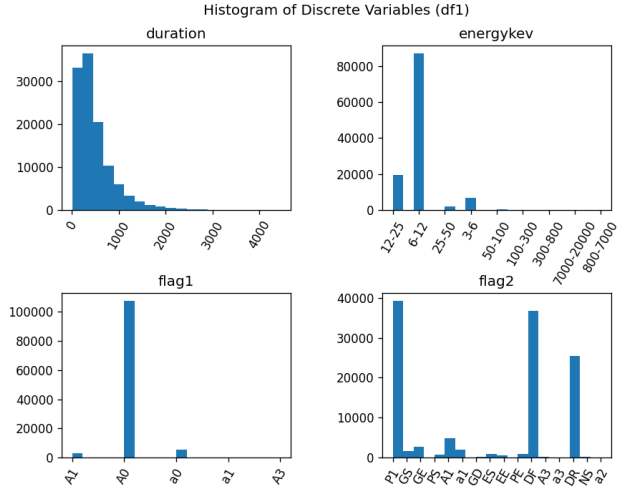

Fig. 1. Frequency distribution of discrete variables

The frequency distribution of the duration with number of flares shows that most of the solar flares that were detected lasted for less than 1000 seconds of time. Most of them lasted for about 500 seconds. Very few solar flares lasted for nearly 2000 seconds. The energy distribution reveals that over 80000 solar flares had energy ranging between 6kev to 12 kev, after

which came 12 kev to 25 kev, 3 kev to 6 kev, and very few ranging between 25 kev and 50 kev. The bottom two graphs categorise the types of flares observed under flag.1 and flag.2.

Moving on to the Konus-Wind dataset, we plotted a histogram of variable classes for the dataset. This represents the number of flares belonging to each unique class included in the dataset.

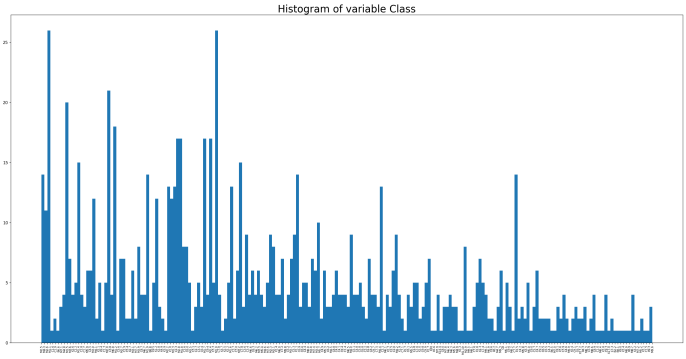

Fig. 2. Histogram of variable classes

B. Variable correlation

We could study the correlation properties of the RHESSI dataset only, and not the Konus-Wind dataset since most variables in the latter dataset were of object datatype.

To study correlation, we first plot a scatter matrix for the RHESSI dataset.

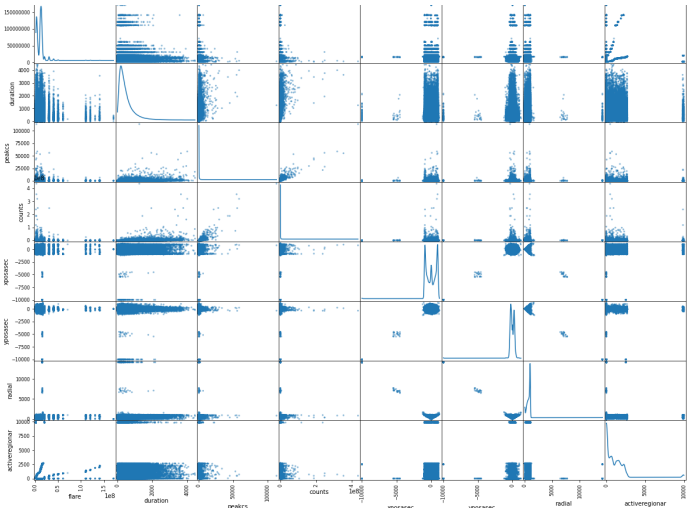

Fig. 3. Scatter Matrix of RHESSI dataset

We plotted the correlation heatmap for the same dataset. In order to generate this heatmap, we have taken care to shift the colour map with its lightest shade centred at one. This ensures that positive and negative values are visually symmetric.

From this matrix, we conclude the following:

- Most pairs of variables are nearly non-correlated, which could lead to higher accuracy in predictions.

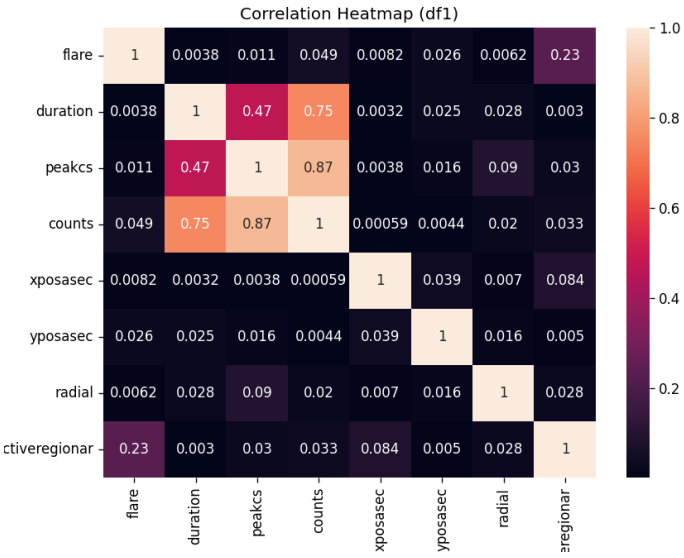

Fig. 4. Correlation Heatmap of RHESSI dataset

- The variables duration, peak.c/s and counts have a higher correlation, indicating their mutual dependence.

We also generated box-and-whiskers plots for each variable of the dataset. To ensure better readability, we first took the natural logarithm of the values stored in each column.

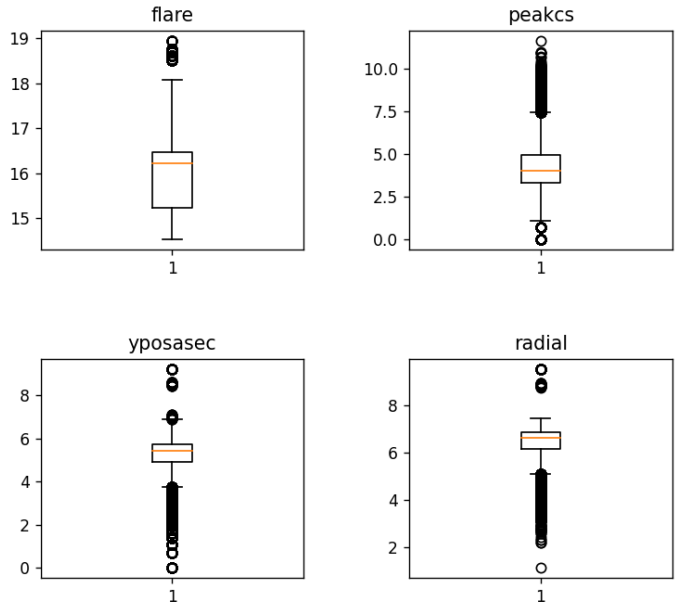

Fig. 5. Box and Whiskers Plot of variables of RHESSI dataset

We conclude the following from the box plots:

- There are a significant number of outliers for each variable of the dataset.
- The median, lower and upper quartiles can be read from the box plots, hence giving us an idea of the distribution of each variable.

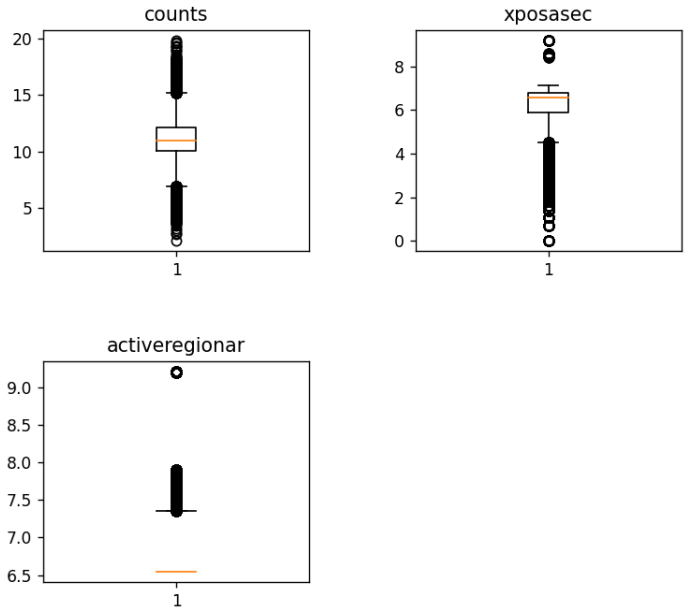

Fig. 6. Box and Whiskers Plot of variables of RHESSI dataset

And finally, we generated a scatter plot of the x and y coordinates of solar flares grouped by energy ranges, giving an elaborate graph.

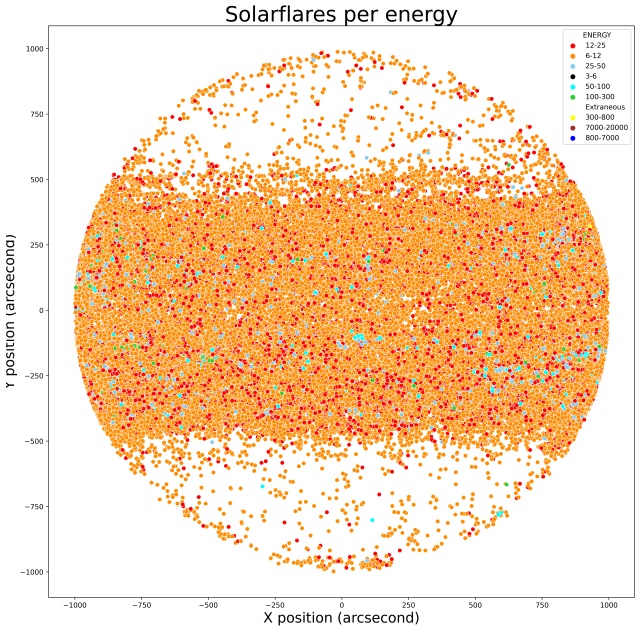

Fig. 7. Scatter Plot of coordinates of each solar flare

This graph represents the location of solar flares grouped according to their energies. As observed earlier, more than 90% of the solar flares have energies lying in the range of 6 kev to 12 kev, represented by orange dots. The other coloured dots represent the miniscule number of solar flares having energies lying in other different ranges.

The graph also shows that while the x coordinates of the solar flares are rather uniformly distributed, the y coordinates of majority of the solar flares lies within the range of -500 arcsecond to 500 arcsecond.

V. Model Selection and Training

A. Long Short Term Memory (LSTM)

Neural Networks are powerful deep learning tools that enables to recognise underlying relationships in a dataset, mimicking the firing of neurons in a human brain. A Recurrent Neural Network (RNN) is one that is a network within a network - it cycles over itself. In these networks, values are in some way dependent on their own earlier counterparts. The output of a particular entity is sent back to the network as the new input of the same entity. Although challenging to ideate on and train, RNNs prove themselves to be extremely powerful tools when applied to tasks relating to spoken and written language. Rather than stopping after an output, the output received in the current iteration is sent back into the model as the new input. This helps in preserving the properties of the previous iterations, thus helping the network to 'remember' some features of the context encountered in the past.

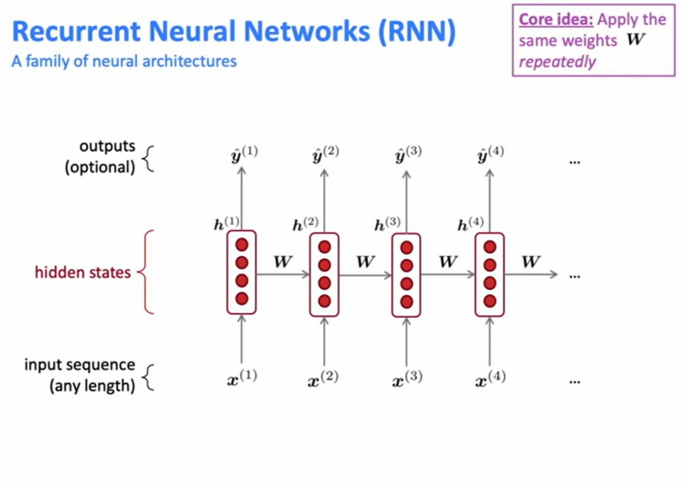

Fig. 8. How an RNN works. Courtesy: Stanford Online

One shortcoming of Recurrent Neural Networks is the issues regarding long-term dependencies. While RNNs perform exceedingly well while relating past context, they can only maintain their precision for a short term. As the length of the recurrence increases, the RNNs start to lose out on context. However, there is a solution to this - a model called Long Short-Term Memory. Long Short-Term Memory networks (LSTM for short) are derived on the basis of RNNs to handle long term dependencies. It tackles the context retention issue in two ways: removing unnecessary information from the context, and adding new information that may possibly be needed in further decision-making.

LSTMs use specialized neural units that use gates to control the flow of information in and out of the network layers.

The gates implement a common sequence and implement the sigmoid activation function, since their aim is to push a binary output of either 0 or 1. As shown in the diagram, the gates are each dedicated for a specific purpose; to 'input' the current context, to 'forget' unnecessary information, and to 'output' the modified context for the next iteration.

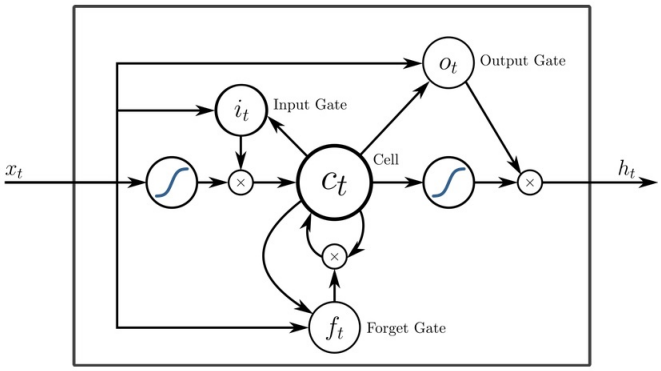

Fig. 9. How LSTM works. Courtesy: Stack Exchange

The LSTM accepts context and an immediately preceding hidden layer as input, and produces the updated context and hidden vectors as output. Thus, through this 'divide and conquer', LSTMs prove themselves more efficient in handling long-term dependencies, since 'remembering' important aspects of the context comes naturally to them due to their architecture.

What makes an LSTM suitable for a time-dependent type of analysis is that it has an internal structure capable of propagating information through long sequences, having a bidirectional context. In this project, data was first processed into training and validation sets to create a dataset based on time windows, and an architecture was designed for our neural network using bidirectional LSTM (long short-term memory), predefined in keras. This cumulative model was then used in training and testing, with the aim of predicting a point in the future given a sequence of data.

B. Autoregression

Autoregression is a time series model that uses observations from previous time steps as input to a regression equation to predict the value at the next time step. Autoregressive models operate under the premise that past values have an effect on current values. This makes the statistical technique popular for analyzing nature, economics, and other processes that vary over time. While several other regression models forecast a variable using a linear combination of predictors, autoregressive models use a combination of past values of the variable.

An AR(1) autoregressive process is one in which the current value is based on the immediately preceding value, while an AR(2) process is one in which the current value is based on the previous two values. An AR(0) process is used for white noise and has no dependence between the terms. In addition to these variations, there are also many different ways to calculate the coefficients used in these calculations, such as the least squares method.

In this project, data was first processed into training and validation sets to create a dataset based on time windows, and both datasets were trained on the autoregression model. Then, their modified mean square errors were compared to conclude the relative superiority of the datasets.

VI. Testing and Predictions

A. Long Short Term Memory (LSTM)

We first plotted histograms of the frequency of solar flares for both the datasets in order to get a rough understanding of what we are working with and choose a model suitably.

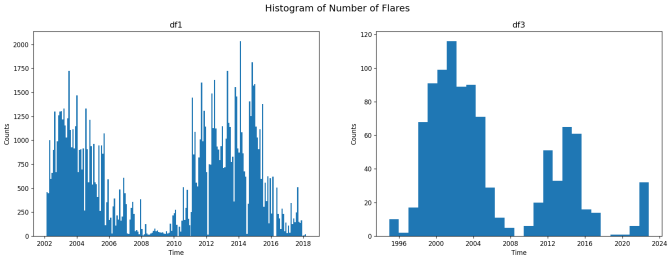

Fig. 10. Histogram of number of flares

Now, we converted the histogram to a line chart to for ease of usability and plotting the results of the model.

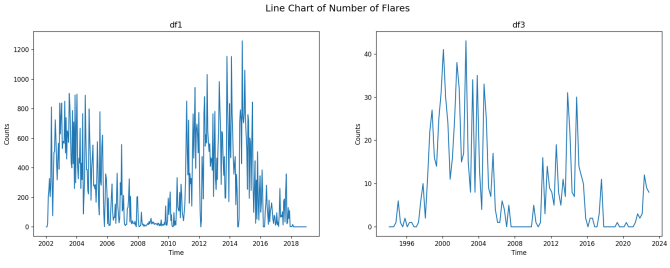

Fig. 11. Line chart number of flares

After implementation of the model using 80% of the data to train and 20% to test, we obtain the following plot.

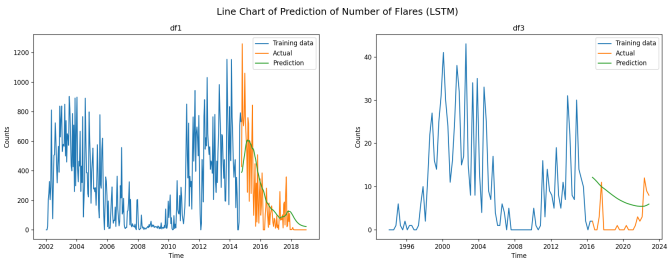

Fig. 12. Result of Long Short-Term Memory model

B. Autoregression

There is a quick, visual check that we can do to see if there is an autocorrelation in our time series dataset. We can plot the observation at the previous time step (t-1) with the observation at the next time step (t+1) as a scatter plot. Pandas provides a built-in plot to do exactly this, called the lag_plot() function. Another quick check that we can do is to directly calculate the correlation between the observation and the lag variable.

We can use a statistical test like the Pearson correlation coefficient. This produces a number to summarize how correlated two variables are between -1 (negatively correlated) and +1 (positively correlated) with small values close to zero indicating low correlation and high values above 0.5 or below -0.5 showing high correlation. Correlation can be calculated easily using the corr() function on the DataFrame of the lagged dataset.

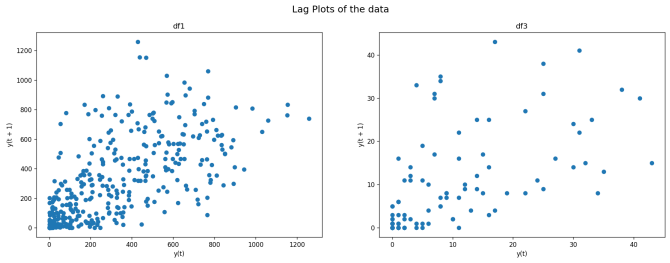

Fig. 13. Lag Plot

We can plot the correlation coefficient for each lag variable. This can very quickly give an idea of which lag variables may be good candidates for use in a predictive model and how the relationship between the observation and its historic values changes over time.Pandas provides a built-in plot called the autocorrelation_plot() function.

The plot provides the lag number along the x-axis and the correlation coefficient value between -1 and 1 on the y-axis. The plot also includes solid and dashed lines that indicate the 95% and 99% confidence interval for the correlation values. Correlation values above these lines are more significant than those below the line, providing a threshold or cutoff for selecting more relevant lag values.

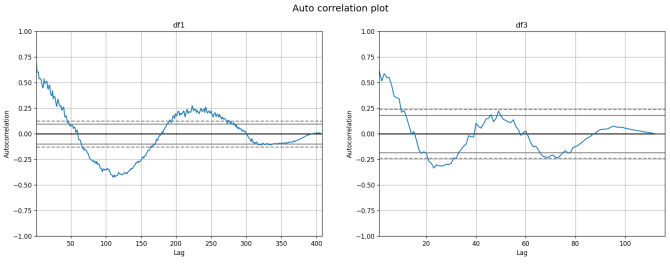

Fig. 14. Auto Correlation Plot

Again, we implementation of the model using 80% of the data to train and 20% to test, we obtain the following plot for both the datasets.

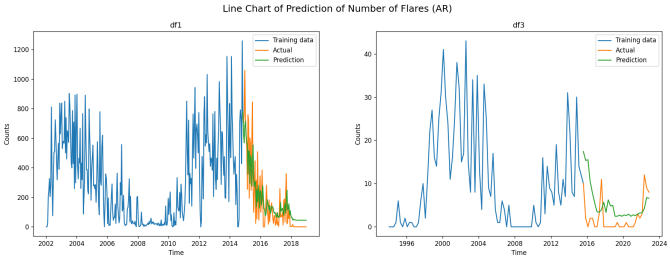

Fig. 15. Results of the Autoregression model

VII. Learning and Conclusions

A. Learning

We now have a much better grasp over the data analysis and visualisation libraries used with python, such as matplotlib, seaborn and pandas. We also learnt different types of machine learning models and understood how to analyse their relative performances. In doing so, we realised the relative usefulness of different machine learning models in different situations or with different types of data. We also noticed the shortcomings of machine learning techniques in predicting complex phenomenon pertaining to humans and climate and recognised that certain events are inherently unpredictable even with a lot of information. We practised how to collaborate over code in real-time with Google Colab. We also got a strong hold of git and Github and used it to manage code effectively. Working in a team taught us how to divide work to not just play to everyone's strengths but also help us improve at those aspects of the project that were slightly foreign to us.

B. Conclusions

Long Short Term Memory

The analysis of 'df1' data-set gives a modified mean squared error of 0.021, whereas, the analysis of 'df3' gives an error of 0.026. This indicates that 'df1' is a better data-set to choose as it's modified mean squared error is smaller than that of 'df3'.

Auto Regression

The analysis and implementation of ML model we get a mean squared of 0.012 for the data-set 'df1' and for 'df3' it works out to be 0.017. This again implies that 'df1' is better data-set to choose.df1 also has a higher correlation than df3 as seen on the Auto Correlation graph.

Comparison of both the models

For the case of 'df1' it produces an error of 0.021 with LSTM model and an error of 0.012 with Auto regression model and in case of 'df3' it produces an error of 0.026 for LSTM and 0.017 for Auto regression both the result concludes that Auto regression is a better model for predicting future based on present data-set.

REFERENCES

[1] Samaha, K. (2021, April 7). "Solar flares from Rhessi Mission." Kaggle. Retrieved November 27, 2022, from https://www.kaggle.com/datasets/khsamaha/solar-flares-rhessi

[2] "KW-Sun: Konus-wind solar flare database." (n.d.). Retrieved November 27, 2022, from https://www.ioffe.ru/LEA/kwsun/

[3] A. D. (2022, November 8). "Time Series Prediction with LSTM in tensorflow." Medium. Retrieved November 27, 2022, from https://towardsdatascience.com/time-series-prediction-with-lstm-in-tensorflow-42104db39340

[4] Fernando, J. (2022, November 9). "What are autoregressive models? how they work and example." Investopedia. Retrieved November 27, 2022, from https://www.investopedia.com/terms/a/autoregressive.asp

[5] Brownlee, J. (2021, September 6). "Autoregression models for time series forecasting with python." MachineLearningMastery.com. Retrieved November 27, 2022, from https://machinelearningmastery.com/autoregression-models-time-series-forecasting-python/

[6] "Correlation of coronal mass ejections with The solar sunspot cycle: Science project." Science Buddies. (n.d.). Retrieved November 27, 2022, from https://www.sciencebuddies.org/science-fair-projects/project-ideas/Astro_p021/astronomy/correlation-of-coronal-mass-ejections-with-solar-sunspot-cycle#objective

[7] Jurafsky, D., &; Martin, J. H. (2014). "Speech and language processing." Pearson.