Using Machine Learning to Predict the Density of Solar Winds

Nora Kane[1] and Dr. Yin Kwong Lee[2#]

[1]Lynbrook Senior High School
[2]University of Chicago
[#]Advisor

ABSTRACT

Solar winds are a variable flow of ions originating from the Sun's corona that have the ability to travel to Earth and disrupt its magnetic field, leading to unpredictable space weather storms which can disturb communication systems and power grids. Prediction and therefore preparation for these solar winds can ensure stability in everyday life; however, an accurate predictor for solar wind features has yet to be developed. Previous models for solar wind prediction have been inhibited by incomplete datasets, correlated features, and over-simplification of solar winds. In this study, a linear regression model was fit to predict the density of solar winds using machine learning. The greater the density of a certain solar wind event, the greater the chance of collision with Earth's magnetosphere, and therefore ramifications on Earth, due to the presence of more charged particles per unit space within the solar wind. The dataset used in this study was taken from a public NASA/NOAA dataset which included 1,048,575 solar wind events. After data preprocessing was performed, the working dataset was composed of 792,587 solar winds events. The model was then trained using the Scikit-Learn Python library to determine a linear regression equation. Following three different training strategies, the model had a relatively low MAE and RMSE score, demonstrating low variance between the true and predicted values of solar wind density. In the future, this model could be applied to real-time input data to warn against concerningly high solar wind density which may negatively impact life on Earth.

Introduction

Solar winds, variable flows of fully ionized particles from the Sun, are the chief drivers of Earth's geomagnetic activity (Dobrijevic, 2022). Up to 70% of geomagnetic activity at Earth during a solar minimum and 30% of geomagnetic activity at Earth during a solar maximum can be attributed to the effects of solar winds (Reiss et al., 2016). Additionally, solar winds can emit large amounts of radiation which can be harmful to astronauts and space equipment, and cause space weather storms which can disrupt satellites, communications, and power grids ("Affects of the Solar," 2019). The origin of these solar winds can be traced back to areas of the solar corona that contain a relatively low magnitude of X-rays. Also, differences in magnetic flux through the Sun can lead to solar winds being released from the Sun's surface (Roberts et al., 2020). Solar wind can mainly be defined by its density, speed, and temperature. Recently, researchers have implemented machine learning techniques to better understand and predict the effects of specific solar winds due to their variable nature.

Past models such as the Wang-Sheeley-Arge (WSA) and Empirical Solar Wind Forecast (ESWF) models have been extensively researched for their applications in solar wind prediction. The WSA model predicts solar wind speed using the magnetic field expansion factor of the Sun, while the ESWF model uses the size and location of coronal holes to predict solar wind speed. These models have suffered from incorrectly estimating predicted values to be closer to the mean values of the training data, arbitrary categorizations of solar winds, and failure in analyzing complex data. Additionally, they can accurately predict solar wind properties close to the Sun but fail to extend these predictions to when the wind is close to Earth. In order for a model to significantly improve the current understanding of space weather and how it impacts life on Earth, prediction models must be able to overcome the above states challenges and accurately output a result that can be applied to near-Earth locations.

Using data from NASA and NOAA Satellites, this research aims to create a model that remedies the above-mentioned issues with past models. Using the Matplotlib Python library, models such as linear regression, lasso and ride regression, and elastic net regression were explored.

Dataset

To train this model, a dataset from the National Oceanic and Atmospheric Administration (NOAA) was used. This dataset, created with the help of the National Aeronautics and Space Administration (NASA), is comprised of 1,048,575, each with 15 columns of features. The features and their respective physical meaning are described in Table 1. This data was recorded using ACE and DSCOVR satellites, both of which orbit around a point, called the L1 point, in a relatively constant position with respect to the Earth as the Earth revolves around the sun. Thus, the large size of this dataset and variable position of the recording equipment makes this dataset suitable for training a machine learning algorithm. The first five rows of the dataset are shown in Table 2.

Table 1. Table of Each feature's symbol in the dataset and physical meaning.

Feature Name in Dataset	Feature's Physical Meaning
bx_gse	Interplanetary-magnetic-field (IMF) X-component in geocentric solar ecliptic (GSE) coordinate (nanotesla (nT))
by_gse	Interplanetary-magnetic-field Y-component in GSE coordinate (nT)
bz_gse	Interplanetary-magnetic-field Z-component in GSE coordinate (nT)
theta_gse	Interplanetary-magnetic-field latitude in GSE coordinates (defined as the angle between the magnetic vector B and the ecliptic plane, being positive when B points North) (degrees)
phi_gse	Interplanetary-magnetic-field longitude in GSE coordinates (the angle between the projection of the IMF vector on the ecliptic and the Earth–Sun direction) (degrees)
bx_gsm	Interplanetary-magnetic-field X-component in geocentric solar magnetospheric (GSM) coordinate (nT)
by_gsm	Interplanetary-magnetic-field Y-component in GSM coordinate (nT)
bz_gsm	Interplanetary-magnetic-field Z-component in (GSM) coordinate (nT)
theta_gsm	Interplanetary-magnetic-field latitude in GSM coordinates (degrees)
phi_gsm	Interplanetary-magnetic-field longitude in GSM coordinates (degrees)
bt	Interplanetary-magnetic-field component magnitude (nT)
density	Solar wind proton density (N/cm^3)
speed	Solar wind bulk speed (km/s)
temperature	Solar wind ion temperature (Kelvin)
source	Denotes if the data was obtained from a DSCOVR ("ds") satellite or an ACE ("ac") satellite

Table 2. A table of the first five rows of the unprocessed dataset used. The table is split into two parts to ensure a large size; however, each event is the same across both sections of the table.

	bx_gse	by_gse	bz_gse	theta_gse	phi_gse	bx_gsm	by_gsm
0	-5.55	3.00	1.25	11.09	153.37	-5.55	3.00
1	-5.58	3.16	1.17	10.10	151.91	-5.58	3.16
2	-5.15	3.66	0.85	7.87	146.04	-5.15	3.66
3	-5.20	3.68	0.68	6.17	146.17	-5.20	3.68
4	-5.12	3.68	0.49	4.62	145.72	-5.12	3.68

	bz_gsm	theta_gsm	phi_gsm	bt	density	speed	temperature	source
0	1.25	11.09	153.37	6.80	1.53	383.92	110237.0	ac
1	1.17	10.10	151.91	6.83	1.69	381.79	123825.0	ac
2	0.85	7.87	146.04	6.77	1.97	389.11	82538.0	ac
3	0.68	6.17	146.17	6.74	1.97	389.11	82538.0	ac
4	0.49	4.62	145.72	6.64	1.77	394.26	94269.0	ac

Methodology

Data Preprocessing

Before training any model, preprocessing was performed on the dataset to ensure accuracy of the model. First, any solar wind sample with missing data was removed, leaving the dataset with more than 750,000 solar wind events. By doing this, the model only considered complete solar wind events during training and was able to accurately analyze the relationships between different features. Next, a correlation heat map, which visualizes how correlated every variable is with every other variable, was generated. This is shown in Figure 1.

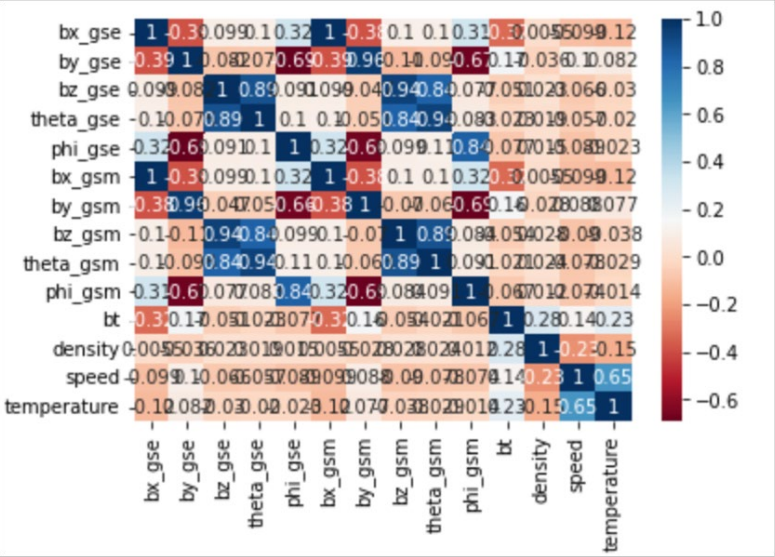

Figure 1. A correlation heatmap showing each feature's correlation with every other feature.

In machine learning, highly correlated data may cause a model to make incorrect predictions, as the effect of each individual variable is clouded. To remedy this, highly correlated features were removed from the dataset, and only one of the two correlated features remained. In this dataset the interplanetary agentic field (IMF) variables in GSE coordinates were highly correlated with each of the IMF variables in GSM coordinates, so each of the IMF variables in GSM coordinates columns were removed from the dataset, leaving only the IMF variables in GSE coordinates and solar wind speed, density, temperature, and overall interplanetary magnetic field magnitude (bt). The source feature was also removed, as this was not a numerical feature and did not provide any useful information regarding the composition of the solar wind. The first five rows of the dataset after these features were removed are shown in Table 3.

Table 3. A table of the first five rows of the dataset after preprocessing.

	bx_gse	by_gse	bz_gse	theta_gse	phi_gse	bt	density	speed	temperature
0	-5.55	3.00	1.25	11.09	153.37	6.80	1.53	383.92	110237.0
1	-5.58	3.16	1.17	10.10	151.91	6.83	1.69	381.79	123825.0
2	-5.15	3.66	0.85	7.87	146.04	6.77	1.97	389.11	82538.0
3	-5.20	3.68	0.68	6.17	146.17	6.74	1.97	389.11	82538.0
4	-5.12	3.68	0.49	4.62	145.72	6.64	1.77	394.26	94269.0

After removing any correlated features, unimportant features were then removed, preventing overfitting of the model. Using a random forest regressor, the relative importance of each feature was plotted. A 0.05 threshold was used for each feature's importance (any variable with a relative importance under this value was removed). The random forest regressor's results are shown in Figure 2. As every feature's importance was above the threshold, no features had to be removed during this step of data pre-processing.

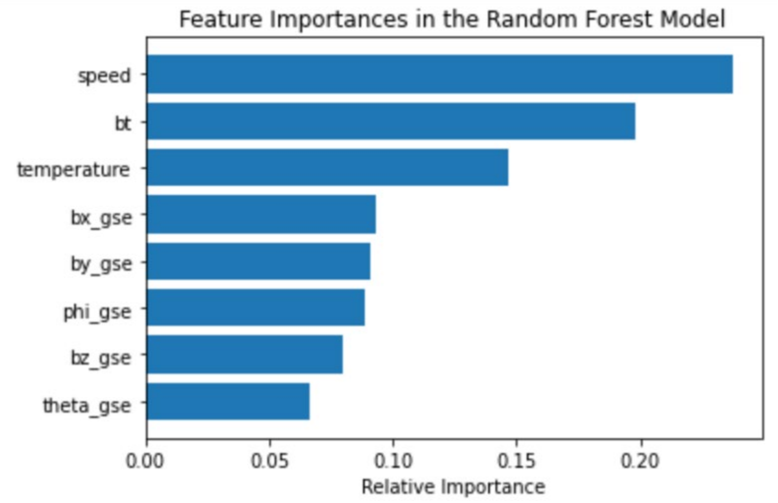

Figure 2. A bar graph showing the relative importance of each feature.

Next, the outliers of each feature were removed from the dataset. In this case, any value of a certain feature below the first quartile of data and above the third quartile of data was removed. This allows for the model to make accurate and streamlined predictions.

The last step in data preprocessing was to standardize the data. During this process, the unit of each value in every feature was removed, allowing future model training to ignore any units and focus on the numerical value of each feature. After this, the data was randomly split into training and testing sets, with 80% of the original dataset becoming the training dataset and 20% becoming the testing dataset.

Model Training

After data preprocessing, models such as linear regression, ridge regression, lasso regression, and elastic net regression were fit to the data. Linear regression uses a simple formula, $Y= A + B_1X_1 + B_2X_2 + B_3X_3 + B_4X_4 + B_5X$ $_5$+ $B_6X_6 + B_7X_7 + B_8X_8$ to determine a predicted numerical value based on weight assigned to each variable. In this case, the coefficient before each variable is the given weight, and the variable itself is the value of each feature for a given solar wind event. Since the model is predicting solar wind density, the Y value obtained from linear regression represents the predicted solar wind density. During model training, the weights which produced the most accurate solar wind density were determined through supervised learning. In this case, the model takes labeled training data and predicts future labels based on the initial inputs. During the learning process, after making a prediction on the training data, the model will be given the expected output and improve its future predictions based on both the given inputs and the given outputs (Delua, 2021).

Lasso and ridge regression work similarly to linear regression. The same formula is used for each; however, during the training of a lasso/ridge regression model, features may be penalized for unimportance. While ridge regression simply changes the assigned weights for each value, lasso regression may eliminate an unimportant feature altogether. The testing process for lasso and ridge regression is the same as for linear regression.

Elastic net regression blends the penalization procedures of both ridge and lasso regression. In doing this, elastic net uses two stages of training a model in hopes of making the model more accurate.

Once training of each of the models was completed, confusion metrics were used to determine the effectiveness of the models.

Results

Confusion Metrics

To analyze how well a model performs, a multitude of model metrics such as an MAE score, an RMSE score, and an Adjusted R-Squared were used. An MAE, or mean absolute error, score is the average of differences between each prediction and observation pair a model creates and can be calculated using MAE = $\frac{\sum_{i=1}^{n}|y_i-\hat{y}_i|}{n}$. This score is usually considered to be the magnitude of prediction error. An RMSE score, or root mean square error score, is the average of the differences between each prediction and its corresponding true value squared (Reiss et al., 2016). It can be calculated using RMSE = $\sqrt[2]{\frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{n}}$. This score is typically used in cases where the effect of outliers on the model's results is intended to be maximized. In both of these cases, a model with the best prediction accuracy will produce a low score. Finally, an Adjusted R-Squared is the variance of the dependent variable which the independent variable can explain, and thus only considers the impact of significant variables. It can be calculated using R^2_{adj} = 1 - $\left[\frac{(1-R^2)(n-1)}{n-k-1}\right]$. In this case, a successful model will yield a high Adjusted R-Squared, which would indicate the model's formula accurately represents the relationships between the different variables.

Findings

Table 4. A table showing the Adjusted R-Squared, MAE, and RMSE scores for each of the models trained.

	Adjusted R-Squared	MAE	RMSE
Linear Regression	0.142	1.68	2.17
Ridge Regression	0.142	1.68	2.17
Lasso Regression	0.130	1.71	2.19
Elastic Net Regression	0.088	1.77	2.24

As shown in Table 4, the Adjusted R-Squared, RMSE, and MAE scores were similar for all four types of regression models. This is likely due to data preprocessing eliminating the need for the additional features of ridge regression, lasso regression, and elastic net regression. The low Adjusted R-Squared value indicates the model was not completely able to understand the relationships between different features. This is likely because these relationships are non-linear and cannot successfully be modeled by a linear formula. However, the relatively low MAE and RMSE scores of <5 indicate the model is able to accurately predict the density of solar winds.

Conclusion & Limitations

Overall, the low RMSE and MAE scores indicate this model can successfully predict the density of solar winds. With a mean average error of less than 2, each solar wind can be properly categorized depending on its predicted density as harmful or non-harmful. As a result, the model could be expanded to receive live input data from satellites and warn officials of concerning high solar wind density which might negatively impact life on Earth. During data preprocessing, around 25% of the data was removed. While this still left the dataset with over 750,000 solar wind events, future research could include the augmentation of missing data to minimize the data that must be removed. Additionally, given that linear regression is a simple, high bias and low variance regression formula, future research could utilize more complex regression formulas or neural networks in order to increase the accuracy of the model.

Acknowledgements

I would like to thank my mentor, Dr. Yin Kwong (John) Lee for his guidance throughout this project. I would also like to thank my science research teacher, Mrs. Kathleen McAuley, for her continued support.

References

Delua, J. (2021, March 12). Supervised vs. Unsupervised Learning: What's the Difference? https://www.ibm.com/cloud/blog/supervised-vs-unsupervised-learning

Dobrijevic, D. (2022, July 6). Solar wind: What is it and how does it affect Earth? *space.com.* https://www.space.com/22215-solar-wind.html

Effects of the Solar Wind. (2019, November 24). https://science.nasa.gov/science-news/news-articles/effects-of-the-solar-wind

Reiss, M. A., Temmer, M., Veronig, A. M., Nikolic, L., Vennerstrom, S., Schöngassner, F., & Hofmeister, S. J. (2016). Verification of high-speed solar wind stream forecasts using operational solar wind models. *Space Weather*, *14*(7), 495-510. https://doi.org/10.1002/2016SW001390

Roberts, D. A., Karimabadi, H., Sipes, T., Ko, Y.-K., & Lepri, S. (2020). Objectively determining states of the solarwind using machine learning. *The Astrophysical Journal*, *889*(2), 153. https://doi.org/10.3847/1538-4357/ab5a7a