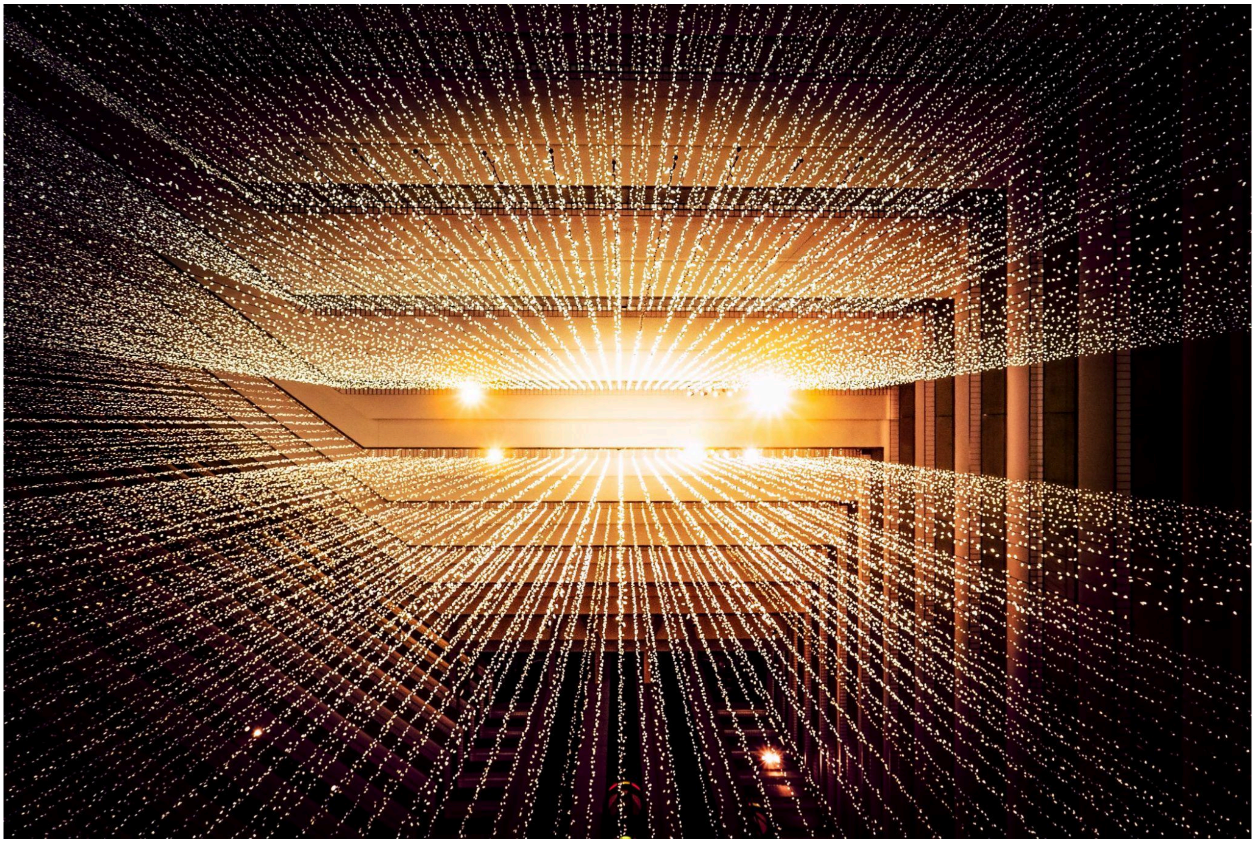
# A Novel Method for Signal Processing Inspired Stock Prediction



By Aaradhyaa Gyawali, Tara Kapoor, Jake Young, and Michael Young

## Non-Technical Executive Summary

The motivation behind our project is to understand underlying patterns between various data sources to predict airline stock values. Specifically, we sought to find a delay between when a "signal" is seen in one data stream and when another responds. In practice, this is difficult due to the large amounts of noise in this type of data. To approach this problem, we devised a novel algorithm for matching large changes in dependent and independent variables across a lag interval, filtered the data according to the results of this algorithm, and then tested it as a proof of concept with multiple linear regression. Although further testing is needed, this method's ability to produce a linear regression model with a large proportion of statistically significant coefficients indicates that our algorithm is quickly and effectively peeling back the noise in our data and selecting significant features.

## Technical Exposition

### Introduction

With the premise that stocks respond to signals from other data streams, we set upon the task of isolating these signals for better prediction. If you can only determine these signals after they affect a stock, they are likely not valuable. However, in the real world, it seems likely that a signal would be delayed in the time it takes for it to have an effect on the stock price, a feature that makes parsing out the true signal more valuable but also more difficult. With the lag interval between when a signal is detectable and when it has an effect being unknown, a signal processing approach to stock prediction is particularly difficult even without the confounding noise of the real world.

One approach to circumvent this problem is to feed all known data into some form of neural network and hope it finds a pattern in the noise. However, this is impractical for our task because convolutional neural networks that are designed to train on ultra-large datasets, ideally with hundreds of features per input vectors and thousands of input vectors, are not optimized for the vectors that we created of a few dozen data points per day of stock data. Although it is theoretically possible that enough computer time would produce a neural network that "finds" the underlying patterns of the data-generating process without overtraining, deep learning without adequate feature selection is more of a shot in the dark (and somewhat of a black box algorithm even if it does perform well) than a data scientist would hope to be taking.

## Our Algorithm

With this in mind, we attempted to generate an algorithm that would be able to find consistent signal lags between an outcome variable and an independent variable. To achieve this, we set upon this process: multiply the square of every data point in the outcome variable with the square of every data point in the independent variable, then sum all products of outcome variable instances separated by 1 day from the independent variable, by 2 days, by 3 days, and so on. The theory behind this idea is that if there is a consistent relationship between these two variables, then there will be some instances where a large signal in that data stream is followed by a large response in the outcome variable and their product will be very large compared to the random noise in the two data streams. Summing the result and then selecting the largest sums would then result in finding the true signal lag, and then filtering only the instances of the independent variable corresponding to that lag to use for training of a model would allow a researcher to shorten the time for training required while maintaining or improving the efficacy of the model.

This algorithm could be represented mathematically as

$$\sum s_i^2 * \begin{bmatrix} a_i^2 & a_{i-1}^2 & a_{i-2}^2 & a_{i-3}^2 \\ b_i^2 & b_{i-1}^2 & b_{i-2}^2 & b_{i-3}^2 \\ c_i^2 & c_{i-1}^2 & c_{i-2}^2 & c_{i-3}^2 \end{bmatrix},$$

where S represents the stock value, and a, b, and c denote independent variables. The resultant matrix contains all of the sums, the indices of the largest of these should represent the true lag between a data stream's true signal and a stocks' response.

There is also an indication that the magnitude of the value at that location has some correlation with the amount of information or the strength of the correlation between an outcome variable and a time-displaced independent variable. With ideas for the future of optimizing prediction algorithms according to the relative sizes of these values, in our code, we referred to these as weights. Since this project was time limited and served as a proof of concept, we discarded much of the information held in these values, and only selected the eight highest indices with which to train linear regressions.

## Methodology

### Data Cleaning & Preprocessing

To ensure that the data is suitable for further exploratory data analysis, we cleaned and preprocessed the following datasets:

A. Weather

Entries in the weather dataset with erroneous data values (e.g. temperatures of 999.9, visibilities of 999999 m, etc.) were dropped from the dataset. Moreover, the cloud_status variable was dropped completely because it only had missing values. As a result, we focused on temperature, wind speed, and visibility. For ease of later analysis, we created six regions (northwest, southwest, central north, central south, northeast, and southeast) and created daily averages of temperature, wind speed, and visibility data in each of the aforementioned regions.

B. Events

The date variable for the Events dataset was standardized to a YYYY-MM-DD format. Additionally, we utilized GeoPy's geocoders to query a list of airports within a 100-mile radius of the event cities based on latitude and longitude.

C. Stocks

Finally, we augmented the stocks dataset with percent change since the closing price of the previous day. This is because raw stock value is not the best indicator of companies' performance due to variations in the number of outstanding shares.

**Data Exploration (EDA)**

After cleaning the data, we conducted a survey of the available datasets to explore any intriguing findings.

A. Fares

Approximately 37% of the entries in this dataset were not associated with an airline. Moreover, we found and plotted the total distance traveled by all passengers of a particular airline per quarter which can be seen in Figure 1. We believed that this aggregated distance data may be related to operational efficiency and consumer preferences, which may be useful for later analysis.

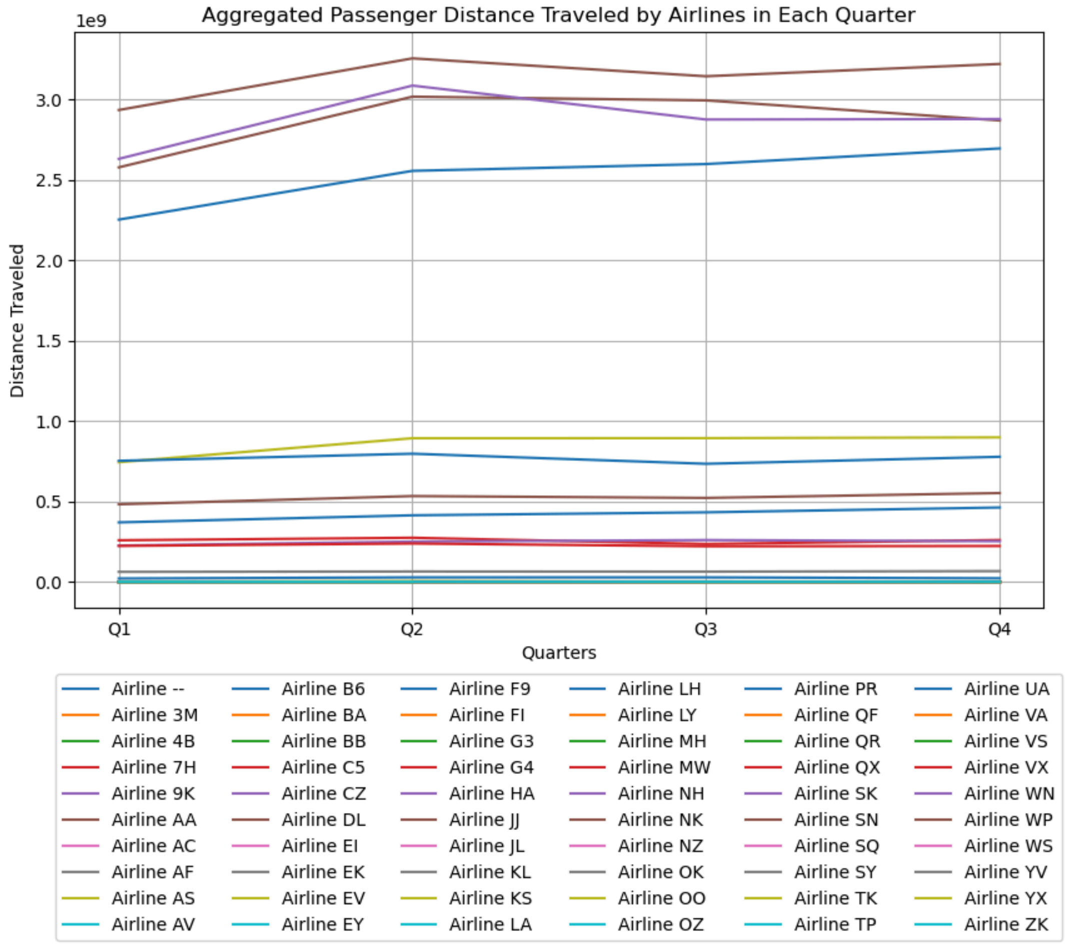


Figure 1: Aggregated Passenger Distance Traveled by Airlines in Each Quarter

## B. Events

After plotting a time series (Figure 2) discretized by week for the number of events occurring nationwide, we can see that the number of events is typically higher during the summer months. Further analysis can be conducted to determine the relationship between event occurrence and weather.

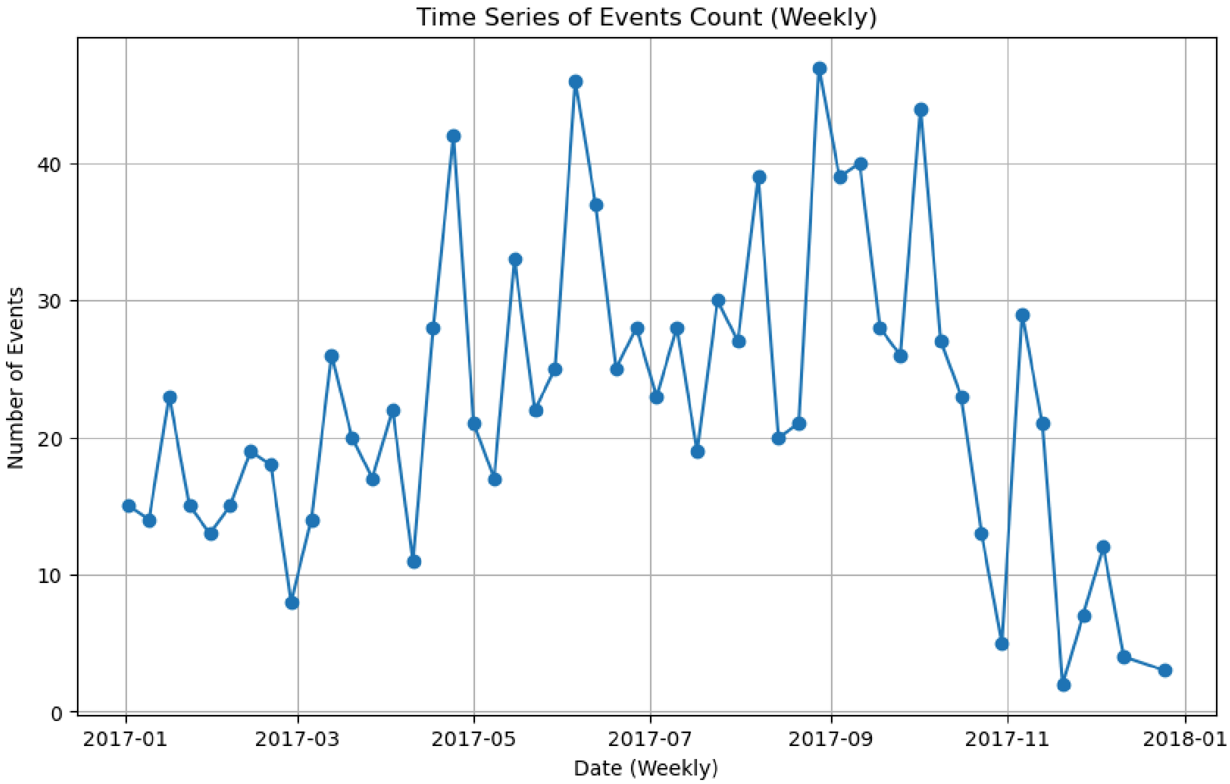


Figure 2: Time Series of Events Count (Weekly)

Additionally, we created choropleths depicting the number of events by state. This was created for the year overall (Figure 3) and by quarter (Figure 4) and revealed popular event locations such as California and New York. The choropleths also show that the events dataset did not contain events from ten states.

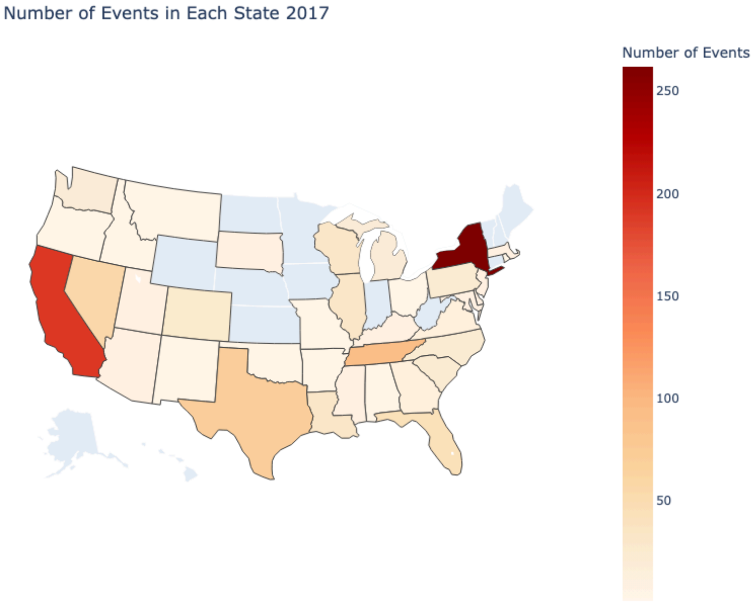


Figure 3: Number of Events in Each State in 2017

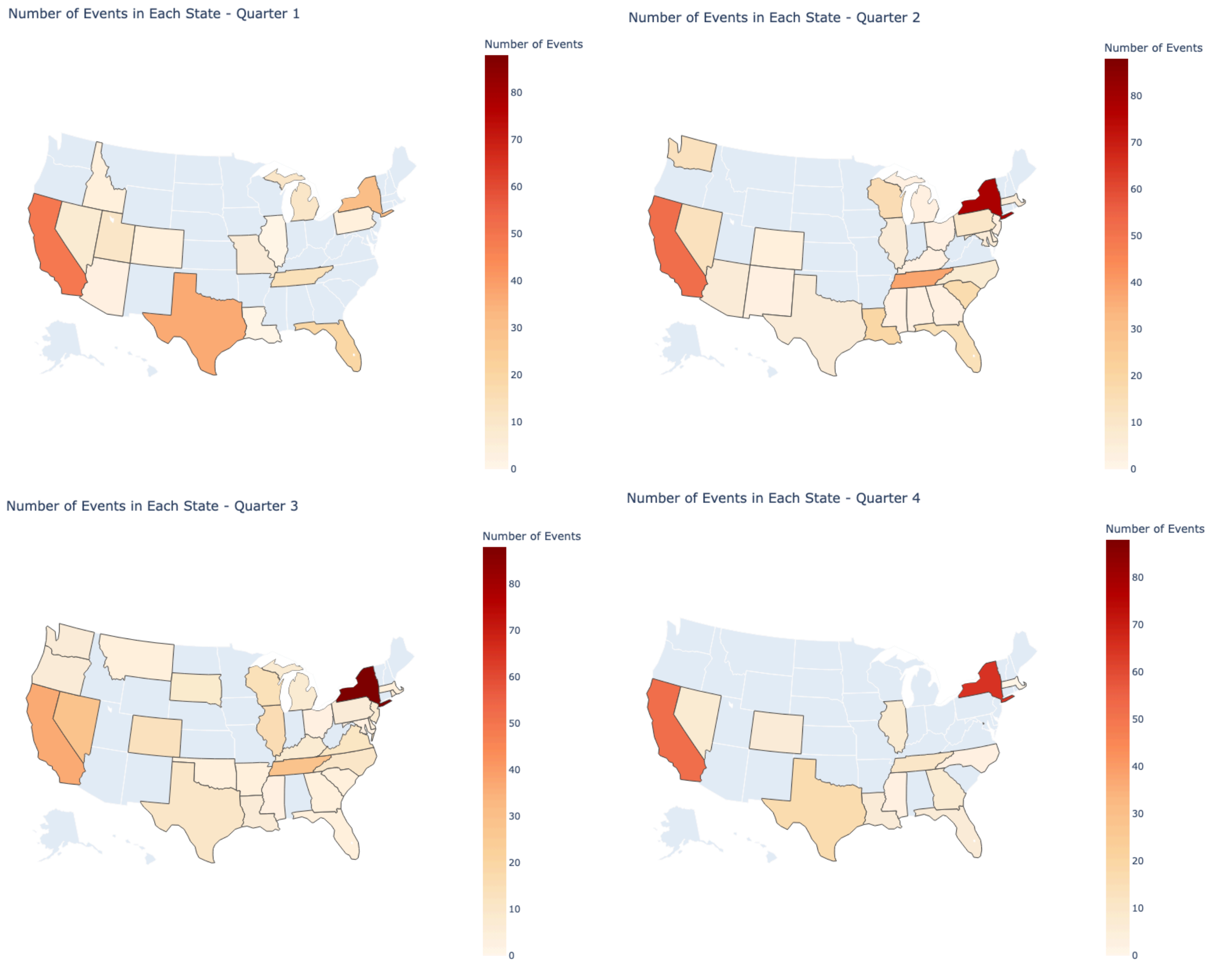

Number of Events in Each State - Quarter 1
Number of Events
80
70
60
50
40
30
20
10
0
Number of Events in Each State - Quarter 2
Number of Events
80
70
60
50
40
30
20
10
0
Number of Events in Each State - Quarter 3
Number of Events
80
70
60
50
40
30
20
10
0
Number of Events in Each State - Quarter 4
Number of Events
80
70
60
50
40
30
20
10
0

Figure 4: Number of Events in Each State by Quarter

### C. Flight Traffic

We primarily focused on canceled flights when exploring flight traffic, as cancellations may adversely affect airlines' operations, customer satisfaction, and financial performance. In the time series plot (Figure 5) of the percentage of canceled flights per week, there are four one-week spikes in cancellations near the beginning of the year and a sustained three-week spike in cancellations around September. These dates are of interest and may aid in further time series analysis.

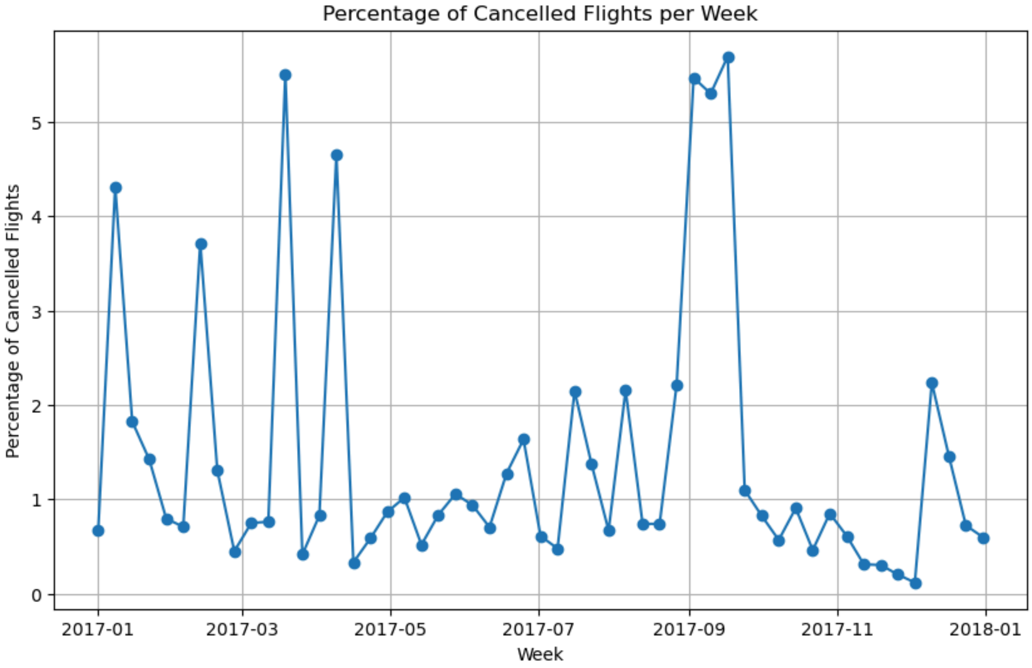


Figure 5: Time Series of Percentage of Canceled Flights by Week

We also wanted to preliminarily explore a potential relationship between the number of flights and the number of events occurring each week. The scatter plot (Figure 6) between the two variables reveals a weak, positive relationship. A simple linear regression fitted on the data yields an adjusted $R^2$ value of 0.206.

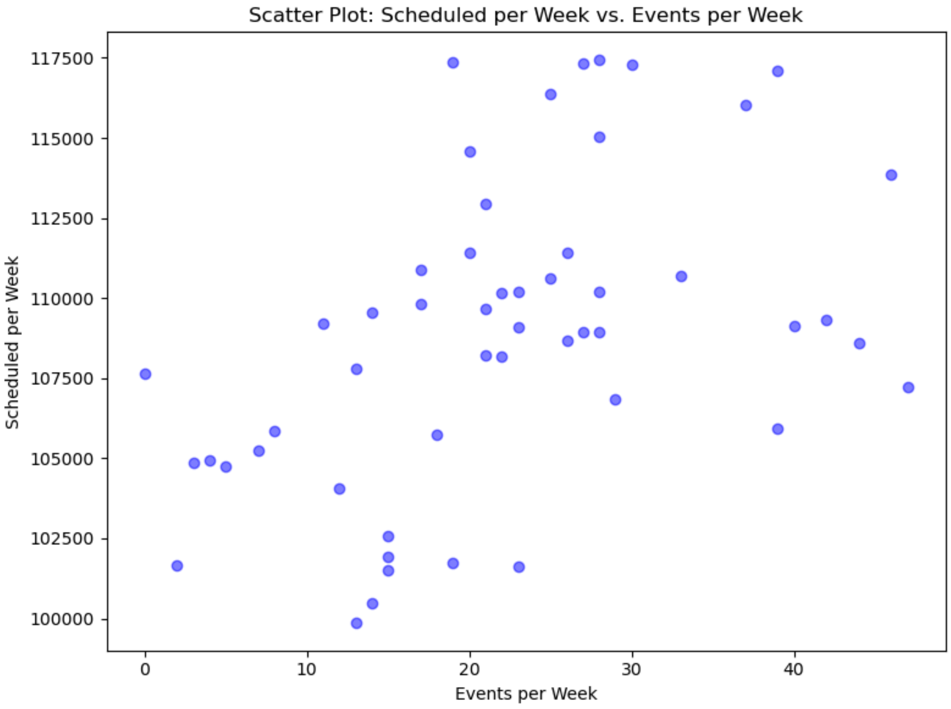


Figure 6: Scatter Plot between the Number of Flights and the Number of Events

D. Weather

The temperature histogram (Figure 7) shows a slightly left-skewed distribution. This may be attributed to the United States' geographic diversity and latitudinal span (northern regions of the United States may experience extremely cold temperatures).

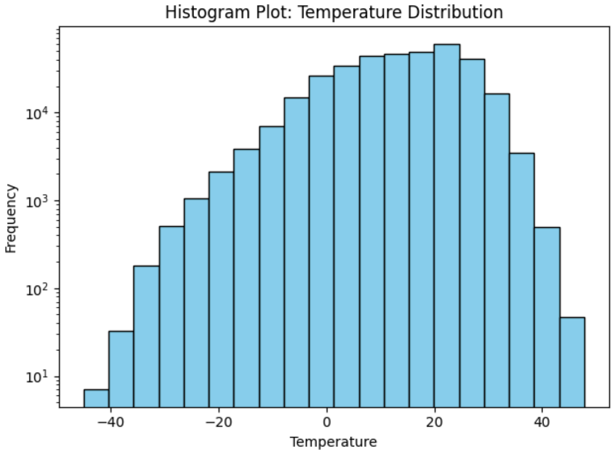


Figure 7: Histogram Plot of Frequency of Temperatures

The visibility histogram (Figure 8) displays a very left-skewed distribution. This may indicate that many airports experience good visibility under typical weather conditions. Moreover, the large number of values at the right end of the histogram indicates the upper limit for visibility.

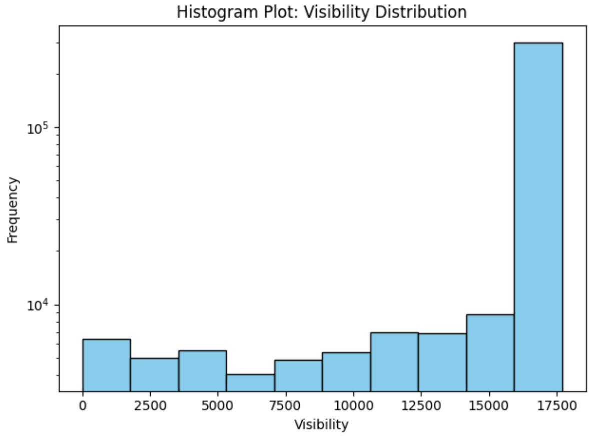


Figure 8: Histogram Plot of Frequency of Visibilities

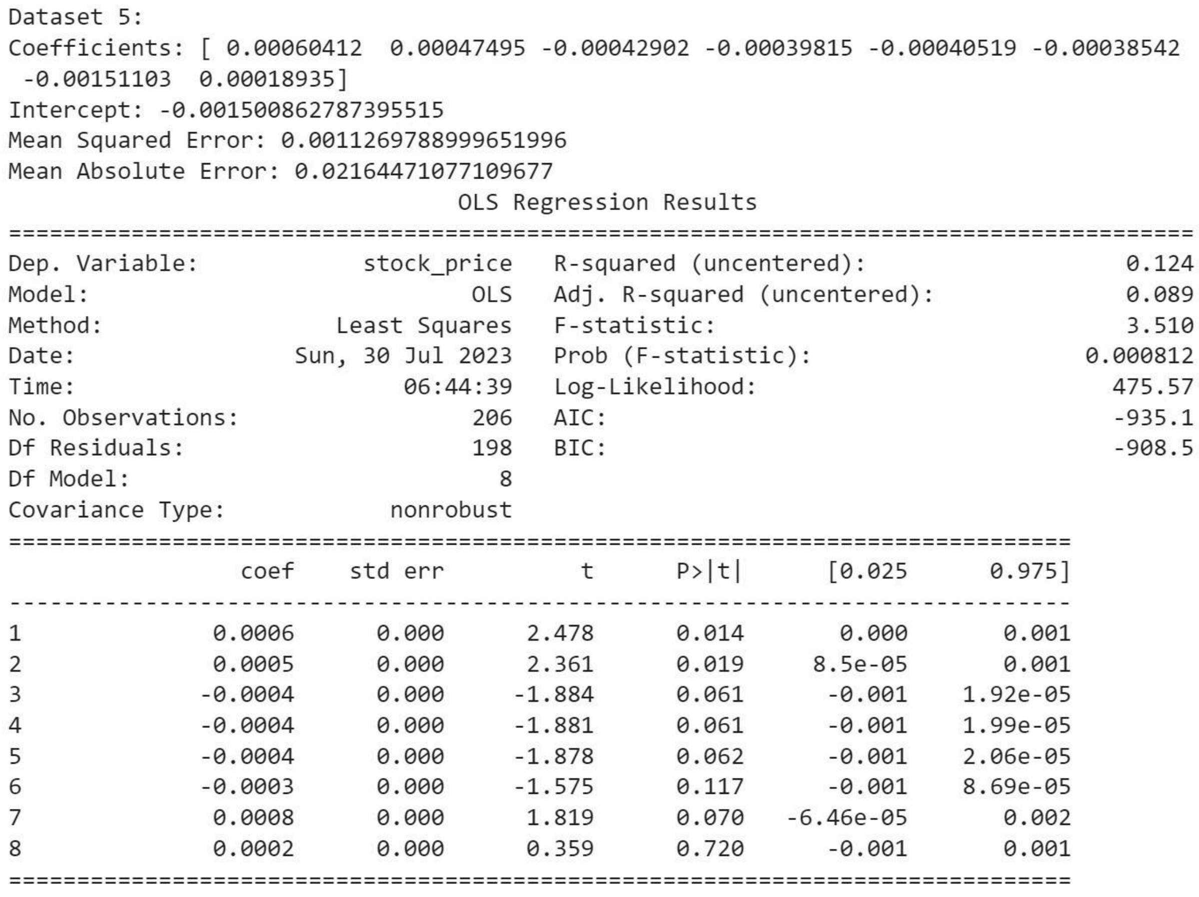
## Modeling

After exploring our various options, we settled on using regionalized weather data to build our linear regression model and test our algorithm's effectiveness. The model was generated on datasets created for the nine airlines with given stock information separately so that the performance of one airline would be assessed independently of the others. The dataset used for the linear regression can be replicated using the following procedure:

1. Classify the weather dataset into six separate regions (northwest, southwest, central north, central south, northeast, and southeast) using the latitude and longitude provided.
2. For each of the regions created above, calculate the average value of each weather variable (temperature, windspeed, and visibility) for airports within the given region every day.
3. Transform the averages into a percent change to (roughly) normalize the different variables.
4. Match the instances of 18 columns of the weather data (three weather variables by six regions) according to date with the percent change in stock prices, created from the stock_price dataset provided.
5. Run the algorithm on each airline stock against the processed weather data.
6. Collect the eight highest values from the result of the algorithm. These values represent the time interval in the different signal streams with the highest correlation.
7. Create a spreadsheet for each airline with nine columns, the first column representing the stock values for 2017 and the rest of the eight values being the highest correlated weather values of varied time displacement, as determined by the algorithm.

**Results**

To evaluate the robustness of the model, we looked into the model performance using Mean Squared Error (MSE). For the linear regression model used for training and testing, the MSE values are low indicating that the model is able to predict the dependent variable quite accurately using the given independent variable.



```
Dataset 5:
Coefficients: [ 0.00060412  0.00047495 -0.00042902 -0.00039815 -0.00040519 -0.00038542
 -0.00151103  0.00018935]
Intercept: -0.001500862787395515
Mean Squared Error: 0.0011269788999651996
Mean Absolute Error: 0.02164471077109677
                                 OLS Regression Results
=======================================================================================
Dep. Variable:            stock_price   R-squared (uncentered):                   0.124
Model:                            OLS   Adj. R-squared (uncentered):              0.089
Method:                 Least Squares   F-statistic:                              3.510
Date:                Sun, 30 Jul 2023   Prob (F-statistic):                    0.000812
Time:                        06:44:39   Log-Likelihood:                          475.57
No. Observations:                 206   AIC:                                     -935.1
Df Residuals:                     198   BIC:                                     -908.5
Df Model:                           8
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
1              0.0006      0.000      2.478      0.014       0.000       0.001
2              0.0005      0.000      2.361      0.019     8.5e-05       0.001
3             -0.0004      0.000     -1.884      0.061      -0.001    1.92e-05
4             -0.0004      0.000     -1.881      0.061      -0.001    1.99e-05
5             -0.0004      0.000     -1.878      0.062      -0.001    2.06e-05
6             -0.0003      0.000     -1.575      0.117      -0.001    8.69e-05
7              0.0008      0.000      1.819      0.070   -6.46e-05       0.002
8              0.0002      0.000      0.359      0.720      -0.001       0.001
==============================================================================
```

More impressively, the P-values calculated for our linear regressions were generally very low, with at least one column having a statistically significant correlation to the stock value in every dataset and some having multiple values under 0.05 and a majority under 0.1.

The R-Squared value of the model ranges from 0.06 to 0.15. This implies that our model is able to explain 6-15% of the change in the dependent variable (stock_price after running the algorithm).

It is important to note that we are using linear regression as a method of testing correlation, not so much as a stock prediction method. Stocks are generally nonlinear, and linear models are not the most effective way to predict them. The end goal and use case of our algorithm is to guide more directly what data gets fed into more complicated modeling methods, particularly deep learning methods. Although we briefly discussed other metrics of the predictive efficacy for our linear model (such as R-Squared), our main goal in using linear regression was to test whether by using the simple transformation of displacing data by a certain number of days we could find strong correlations where preliminary correlation testing failed. To that end, this section of the project was a huge success.

## Conclusion

We have demonstrated the efficacy of our algorithm in determining signal lag by selecting statistically significant features according to the results of our algorithm. The proportion of statistically significant coefficients created shows the promising ability of our algorithm to quickly determine true signals, a difficult task with so much of real-world time-series data being noisy.

With greater time and resources available for this project, we would like to explore many different avenues for applying this algorithm. The most natural progression is to use it to potentially guide the training of neural networks, acting as a coarse filter for data, to save computing time, prevent overtraining, and allow the algorithm to more quickly converge on a true global maximum for predictive accuracy. It would also be worth researching how to use this algorithm in conjunction with non-ML models and to scrape more data out of the resultant “weights” or sums referenced earlier. Work also needs to be done towards normalizing different data streams to make this algorithm less susceptible to preferencing independent variables with higher value distributions but no more actual information than another information stream. Lastly, we can put this concept onto more rigorous academic footing and optimize it for applied use by analyzing our algorithm's performance on computer-generated, synthetic signals with noise.