

**CIS - 600 Intro To Machine Learning**

***Final Project***

Anurima Swarup (853430232)

Nirmit Patel (992323441)

Manoj Kumar Gundapaneni (239719638)

Purushothama Rajanna (770079902)

Table of Contents

[Introduction: 3](#_Toc134134991)

[Initial Prediction: 3](#_Toc134134992)

[Final Prediction: 8](#_Toc134134993)

[Conclusion: 19](#_Toc134134994)

[Reference: 20](#_Toc134134995)

# Introduction:

* As part of the final project in our Introduction to Machine Learning class, we were provided with flight data for United Airlines departing from four different airports: Newark (EWR), Washington (IAD), Denver (DEN), and Chicago (ORD), all of which were destined for Syracuse (SYR) Airport.
* Our objective was to predict the flight status for each flight, which we classified as "early" if it arrived more than 10 minutes ahead of schedule, "on-time" if it arrived within plus or minus 10 minutes of the scheduled time, "late" if it arrived more than 10 but less than 30 minutes late, and "severely late" if it was more than 30 minutes late.
* The project consisted of two stages: initial prediction and final prediction. To accomplish these, we were provided with two different datasets containing flight data for the periods of April 12-15 and April 21-24, respectively. We had no restrictions on the data we could use to train our model or on the type of machine learning model we could employ.

# Initial Prediction:

* For our initial prediction we were given a .csv file with the following data : Date, Day, Origin Airport, Flight Number, Arrival Time and a blank column Status. Our goal is to predict the status of these flights.

**Approach:**

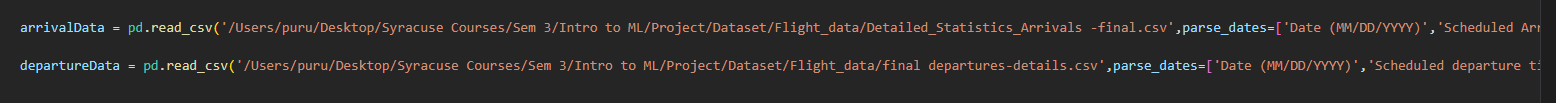
* For our initial prediction we went for a very simple approach of just using historical flight data for our train data without adding any additional data to it.
* Our thought process behind this approach was to find just how accurate of a prediction we can get just by using the historical flight data without introducing any additional factors to our train data.
* For this approach we decided to use a machine learning model to predict the delay based on the historical data, whatever delay we get out of it we then categorized them into late, severely late, on-time and early.

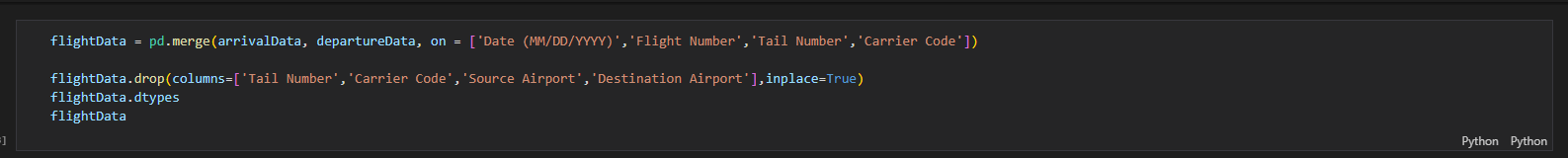
**Data Gathering and Analysis:**

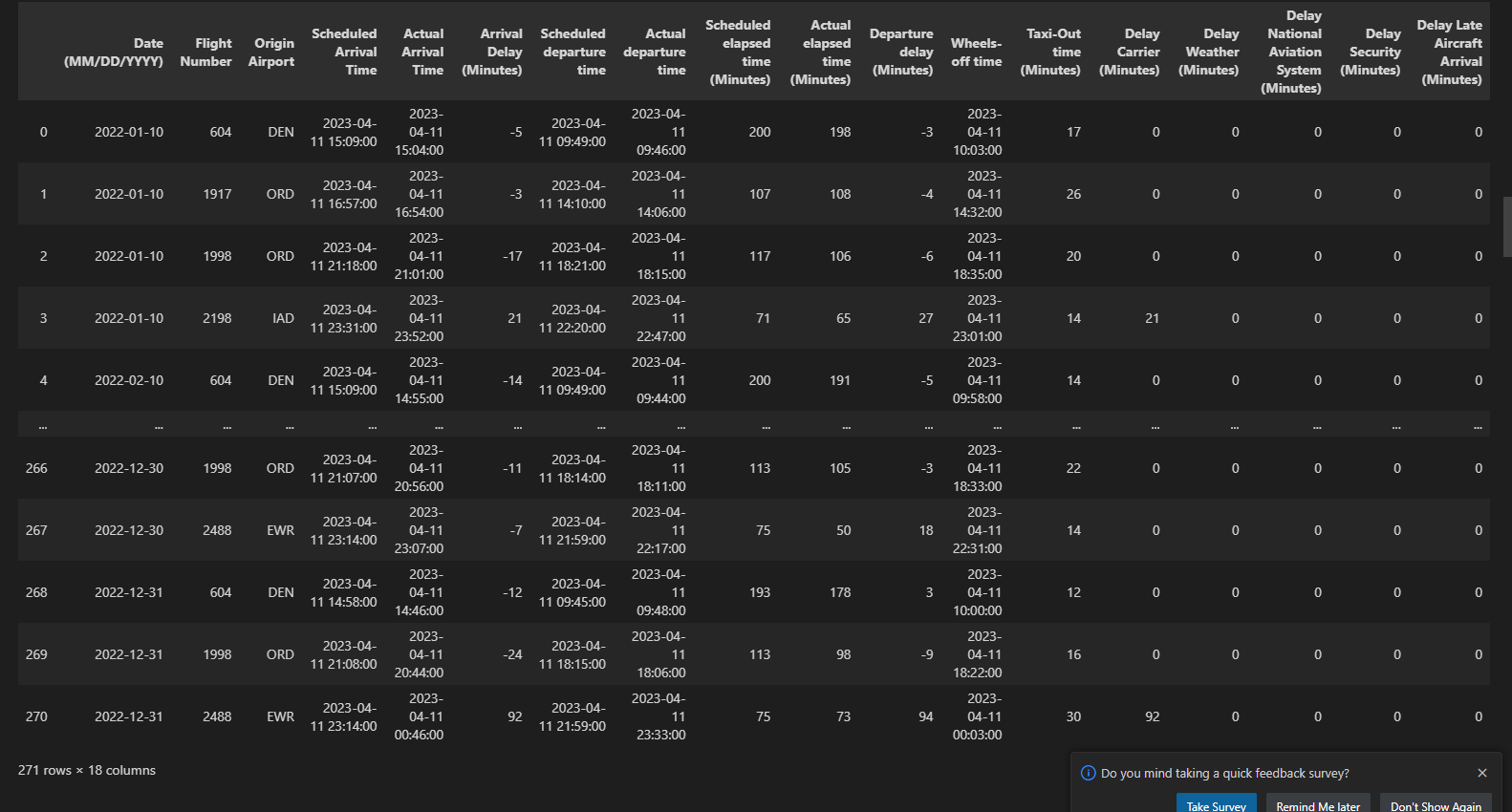
* We obtained information from the Bureau of Transportation Statistics (BTS), a federal body that gathers and disseminates statistics on airline performance and operations.
* We began by obtaining data for departures from four places (Washington, Denver, Newark, and Chicago) and arrival statistics of Syracuse.
* Aside from the data suggested by the professor, we gathered a few more metrics (which are useful in making accurate predictions).
* We integrated arrivals and departures based on date, origin airport, and flight number after cleaning the data. This process assisted us in removing any inconsistencies in the data.
* To train the model, we hot encoded the origin airports and scaled the data.
* We used a linear regression model for training and got a score of 0.89.
* Test predictions are moderately off. Mean absolute error is 13.9 and Error ratio is 7.22
* Finally, we exported the predictions into a CSV file.

**Output:**

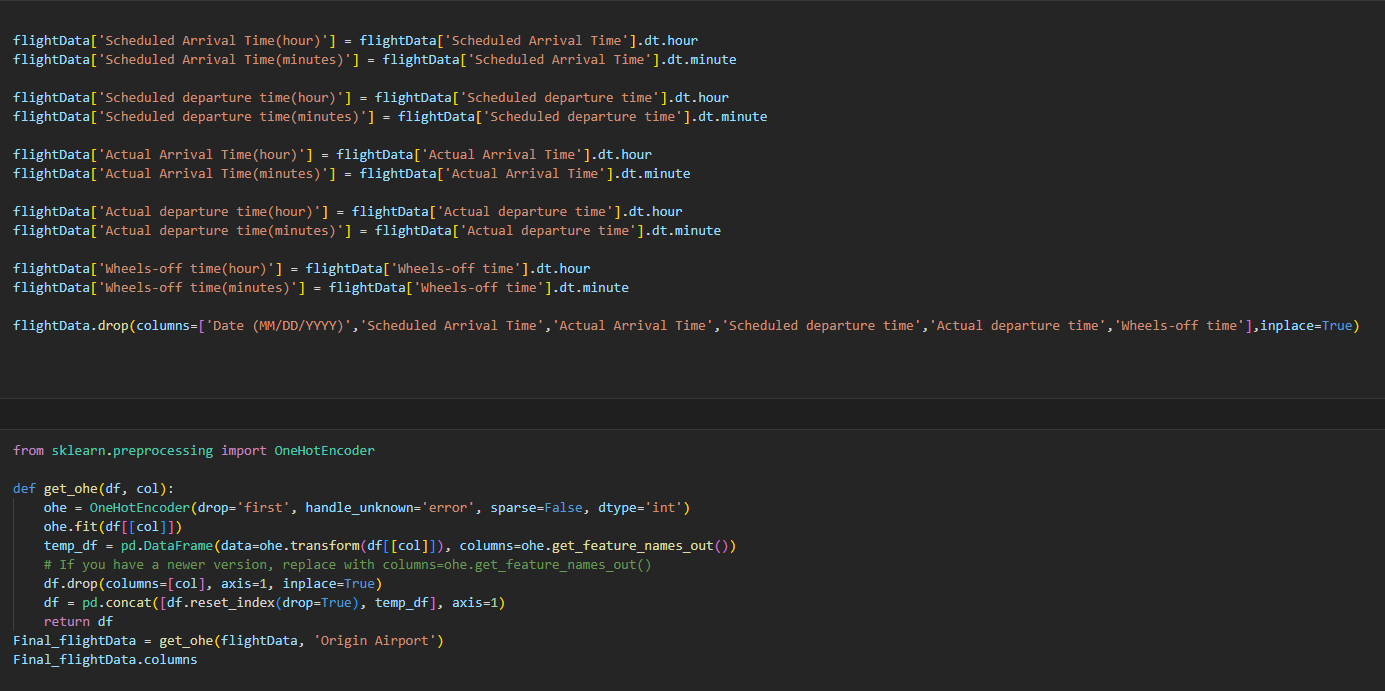
* For our train data we first read arrival and departure csv data into our dataframe and then merged it together to get our final flightData



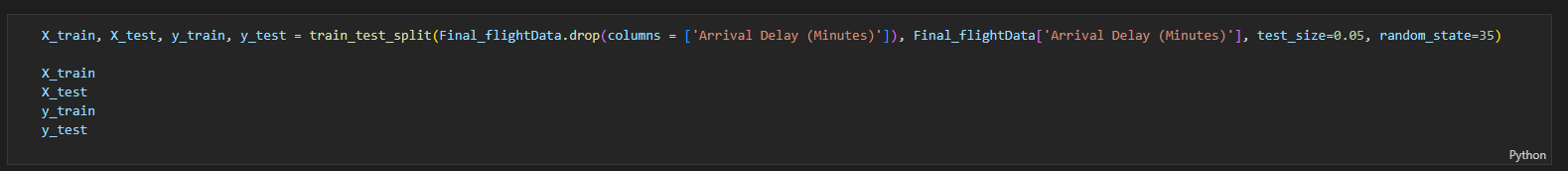


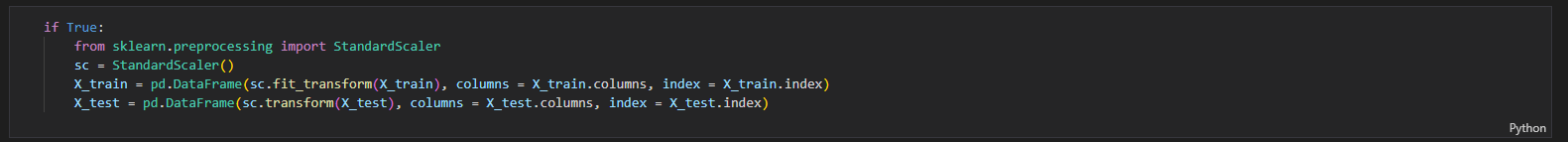


* After removing NaN, we removed unnecessary columns, changed the data type of a few columns into datetime and perform a one hot encoding on “Origin Airport” column

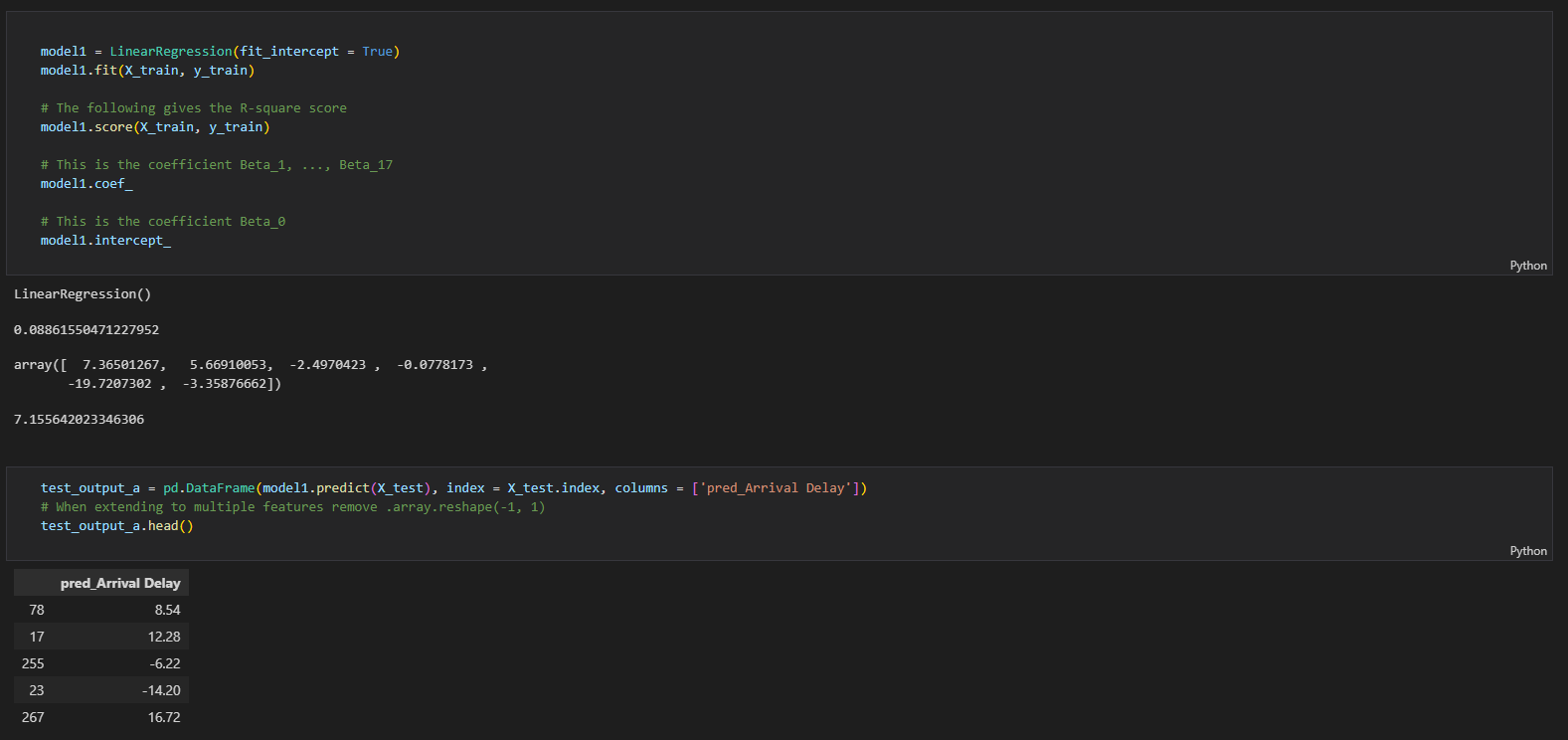


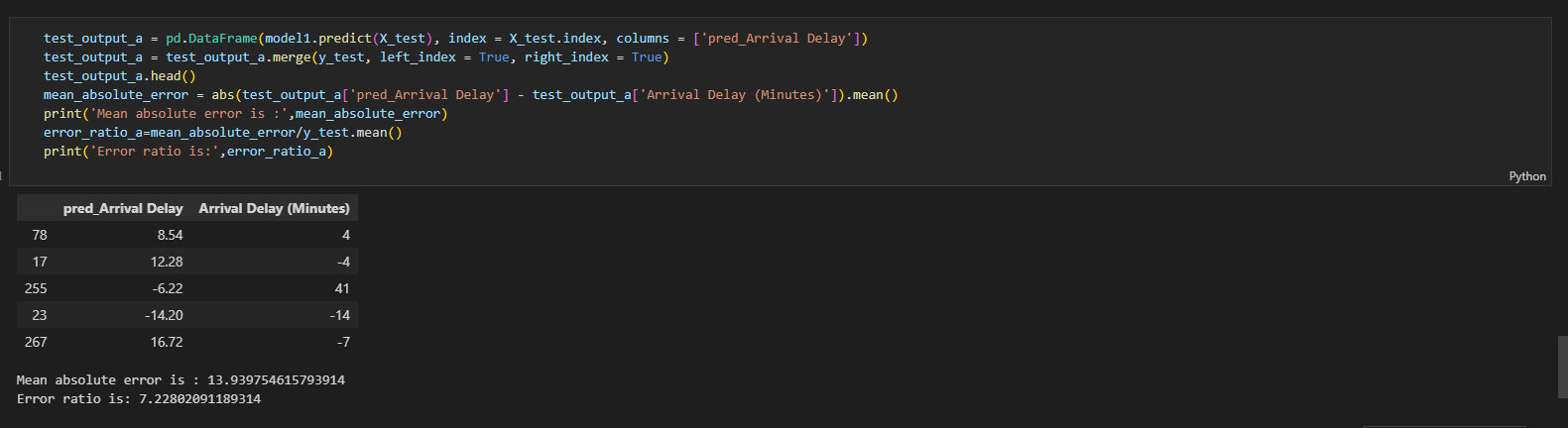
* After that we performed a train test split and used standard scalar on the data so we can fit it into our model.



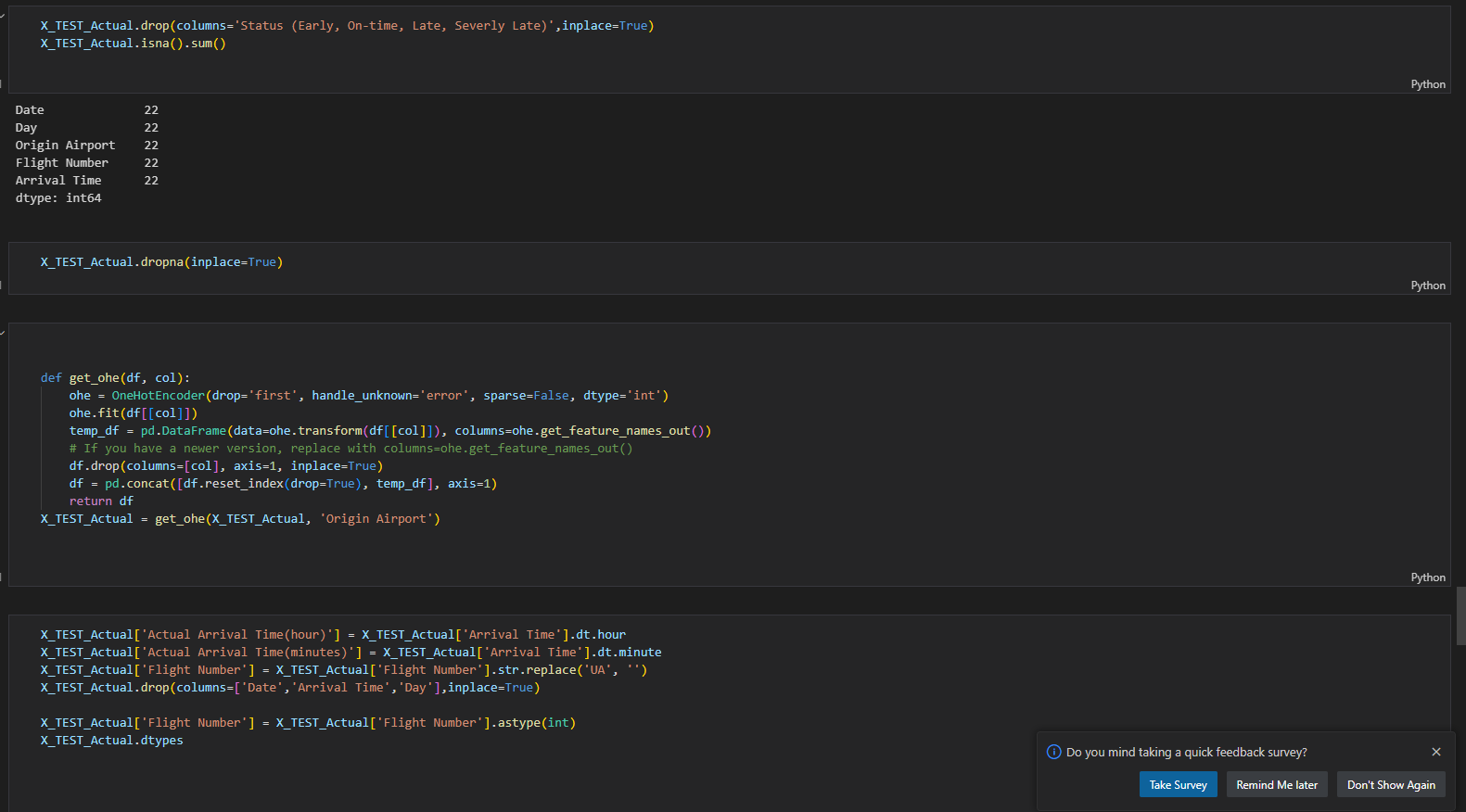


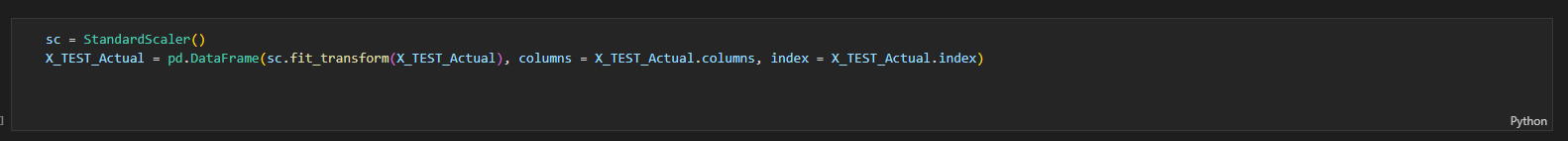
* We use Linear Regression, first we used test data to find out the score of our model.



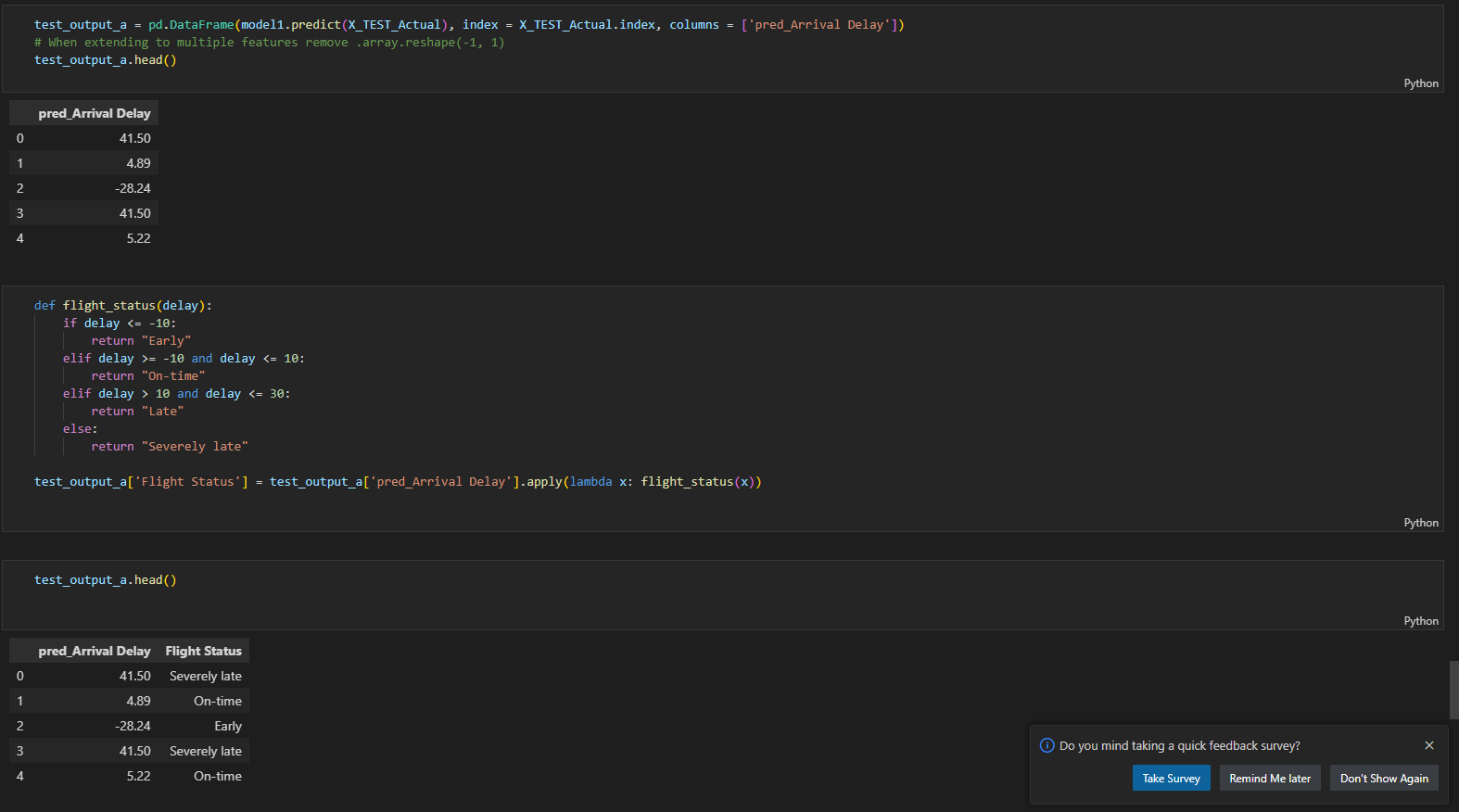


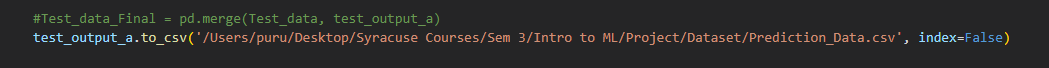
* After checking the score of our model we moved on with our actual test data. We added that data into another dataframe and performed all the necessary scaling operations and one hot encoding to get the data in the same format as our test data that we used for our model.





* Finally, we used our test data to predict our initial prediction, converted that prediction into the status that was asked from the question and created a csv file of our initial predictions.





* According to the ground truth our model predicted 8 out of 32 correctly with the accuracy of 25%

# Final Prediction:

* After our initial prediction we can see that just by using the historical data in this particular situation will not give us very accurate predictions. So, to increase the accuracy of our prediction we have to incorporate multiple datasets.

Approach:

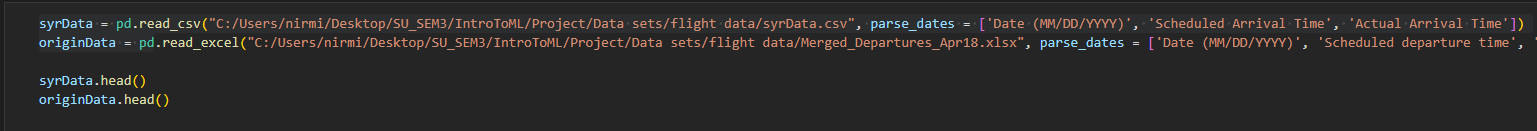
* For our final prediction(21-24) we have considered both flight data and weather data. Because weather is one of the most common causes of flight delays, taking into account both flight data and weather data is a wise approach. This may improve our understanding of the influence of weather conditions on flight schedules and generate more accurate predictions by including weather data.
* We planned to consider the weather on both the arrival and departure city. It is critical to collect information about weather conditions in both the departure and arrival cities when utilizing weather data for prediction. This data assisted us in identifying possible difficulties that may develop during the journey, such as severe rain or thunderstorms, which may cause flight delays or cancellations.
* Our thought about using weather data is getting the reasons behind the delay and interpolate with the timings. After gathering weather data, we evaluated it to see how it will affect the flight schedule. For example, if thunderstorms are expected at the destination airport during the scheduled arrival time, the flight may be delayed or even canceled. If the weather is clear and there are no serious complications, the flight may arrive on time, if not early.
* By merging weather and flight data, we can create a more accurate forecast model that accounts for the specific aspects that might affect the flight schedule. For example, based on current weather conditions, we may utilize meteorological data to interpolate timings and predict the most likely time of arrival or departure.

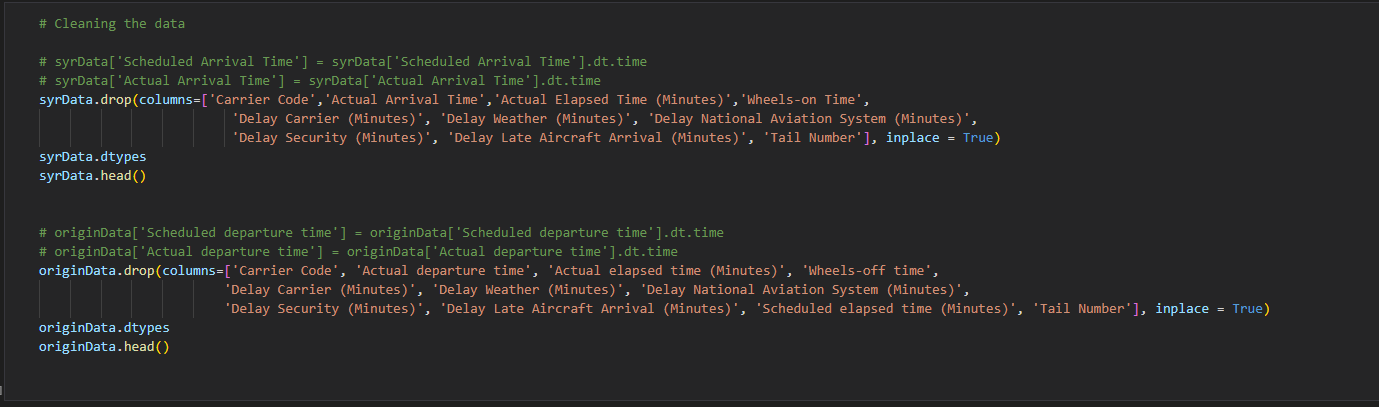
Data Gathering and Analysis:

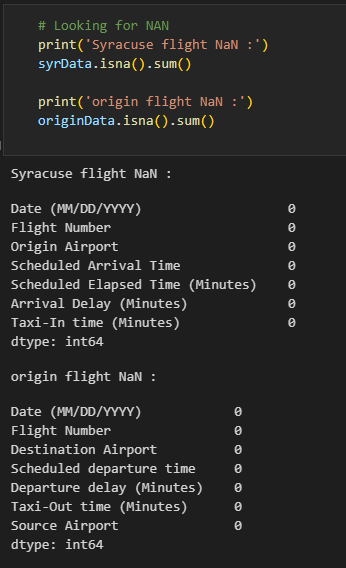
* We obtained information from the Bureau of Transportation Statistics (BTS), a federal body that gathers and disseminates statistics on airline performance and operations. We acquired insights into different elements of airline operations, such as estimated flight delays, cancellations, on-time performance by using data from the BTS. This information has been used to discover trends, patterns, and concerns affecting airline performance, as well as to construct predictive models for anticipating future performance.
* For weather data we used weather bit api to get weather from the past and also forecast data.This information can be used to create predictive models that foresee probable weather-related concerns, allowing us to take proactive efforts to avoid delays and enhance overall flight operations.
* We have used date as a key to merge flight and weather data to create a complete dataset for predicting aircraft delays.Because flight schedules and weather conditions are both time-dependent, using date as a key to merge these datasets is our analysis.
* While using weather data we considered metrics like temp, pressure, wind speed, wind direction,precipitation and others which are key factors.
* Later the extra metrics have been divided into four categories : Cloudy, rain, snow and clear. Which gives us little clarity in data.
* Weather conditions such as cloudy, rain, snow, and clear may all have an affect on flight operations in different ways. Cloudy circumstances, for example, might produce reduced visibility, resulting in aircraft delays or cancellations. Rain and snow can have an effect on airport operations, such as runway conditions and aircraft de-icing needs. Clear weather, on the other hand, may have no effect on aircraft operations.
* We have used the same approach as initial prediction while training the model but this time we have trained multiple models to get the best predictions.
* LinearRegression : 0.12
  + Mean absolute error is 31.06
  + Error ratio is 2.14
* GradientBoostingRegressor
  + Mean absolute error is 32.00
  + Error ratio is 2.20
* Lasso Regression
  + Score: 0.11
  + Mean absolute error is 30.89
* DecisionTreeRegressor
  + We have used cross validation technique to get best prediction but MAE is same for both the results
  + Mean absolute error is 28.2
* RandomForestRegressor
  + Mean absolute error is 30.89
  + 31.44

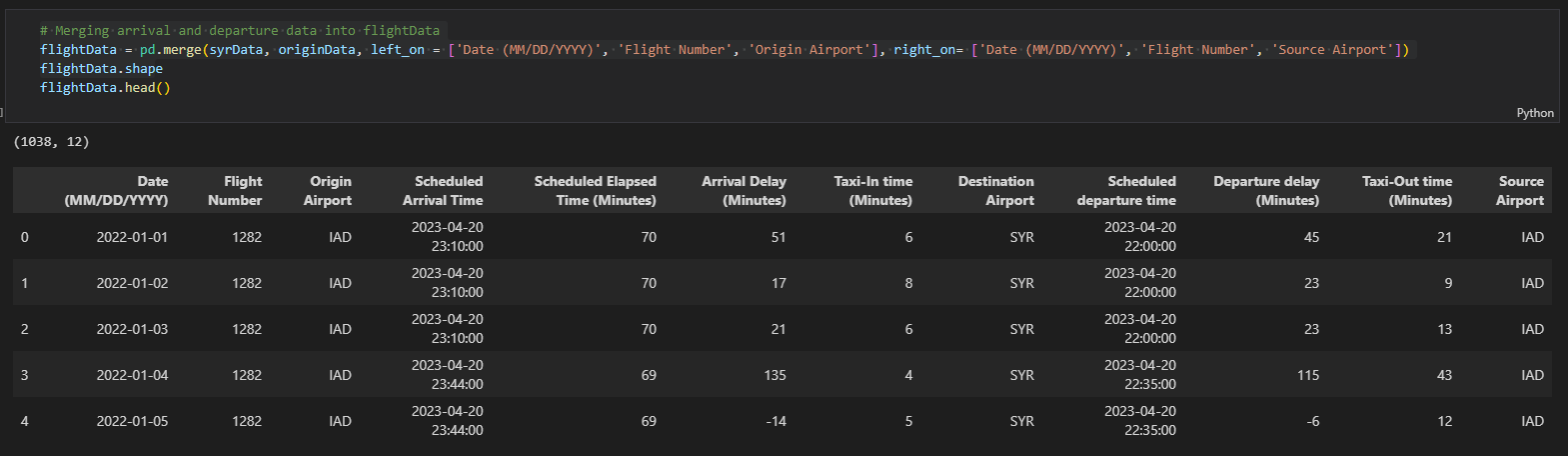
Output:

* We started by adding the flight data for our source airport and destination airport to a dataframe. In this data we had some unnecessary columns, we removed them, dropped NaN rows and merged two datasets on date, flight number and origin airport so we can get our final flight data.

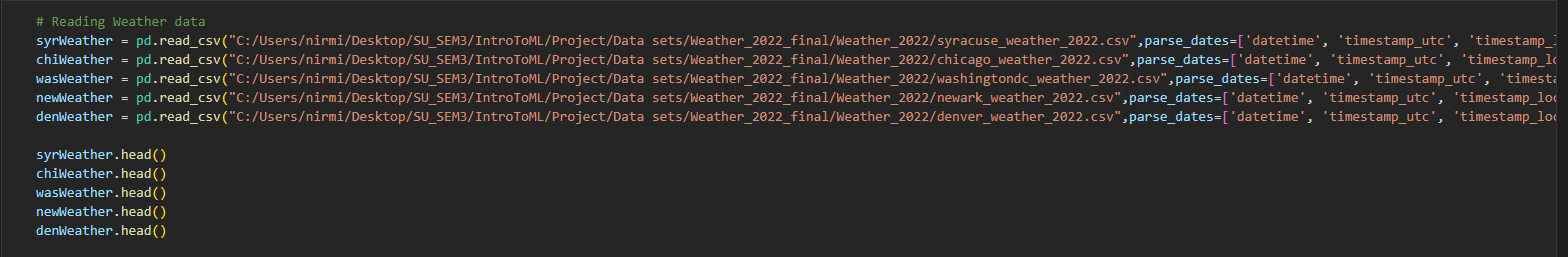


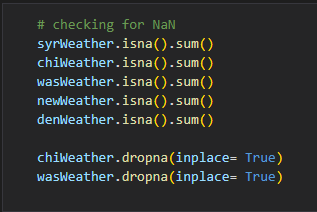


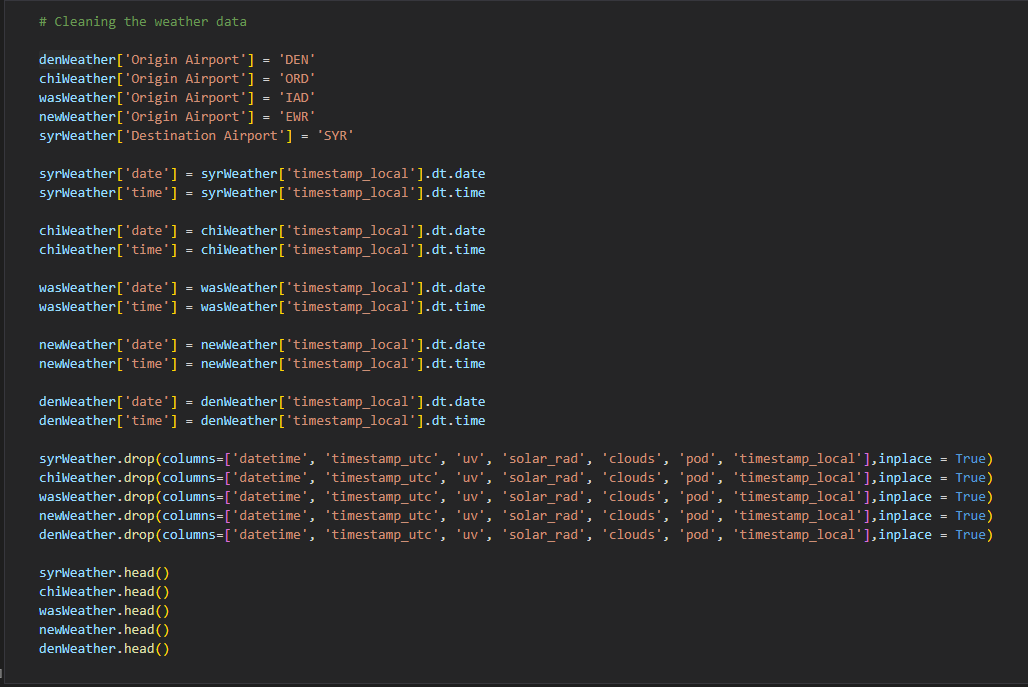


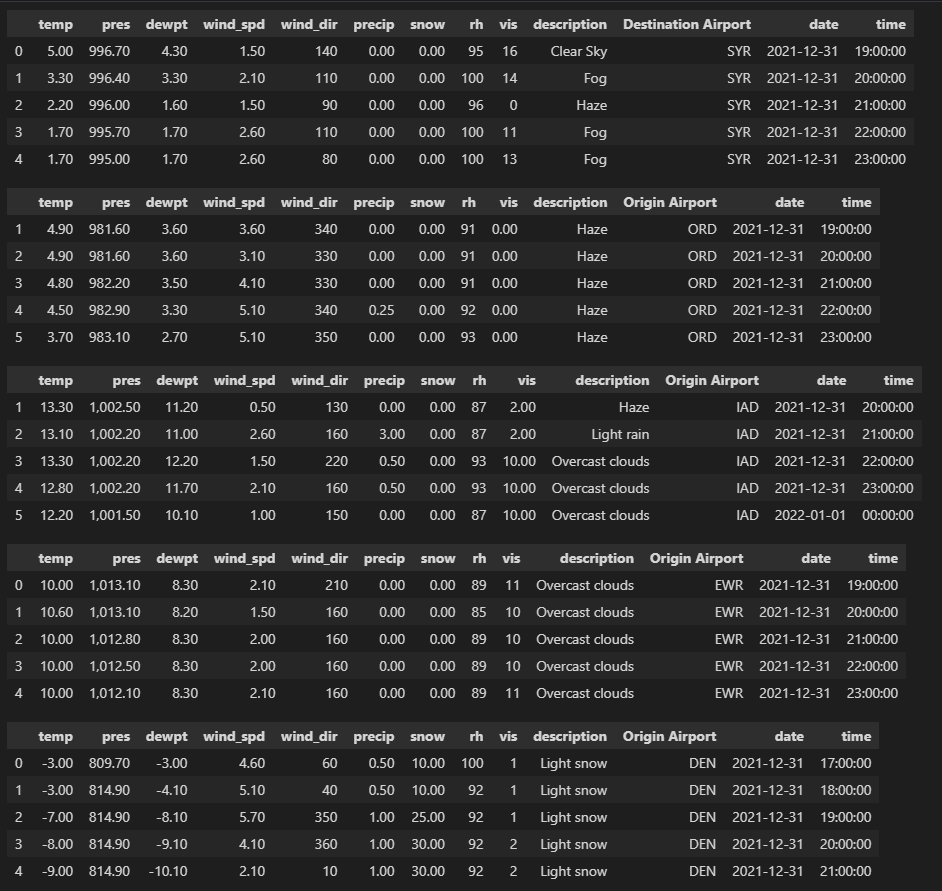


* After getting our flight we started working on getting weather data. We read weather data for Syracuse, Chicago, Newark, Washington D.C. and Denver into dataframes, dropped all unnecessary columns and removed NaNs to get our final weather data.

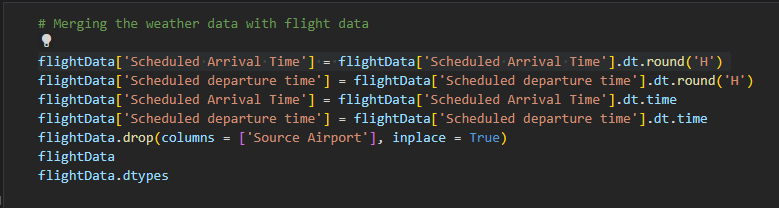




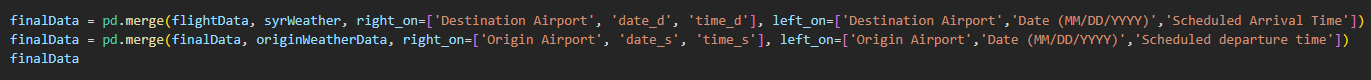


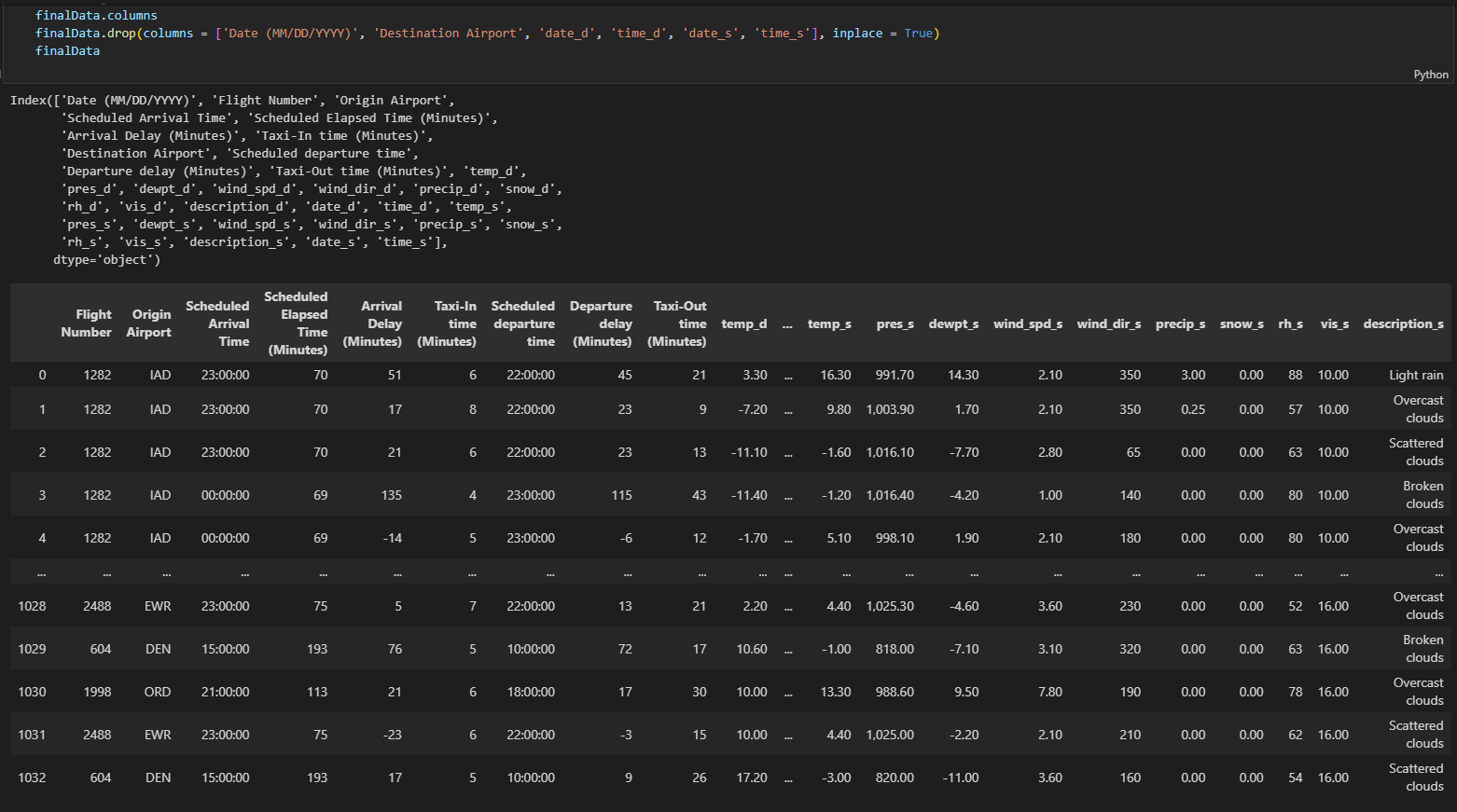


* Finally to get our final data we merged all the data using different keys and performed some cleaning to get our final data.

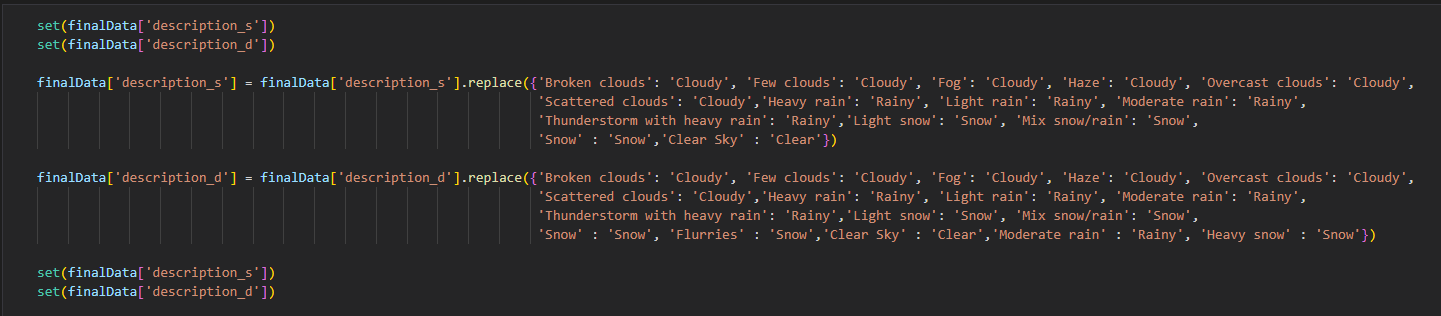


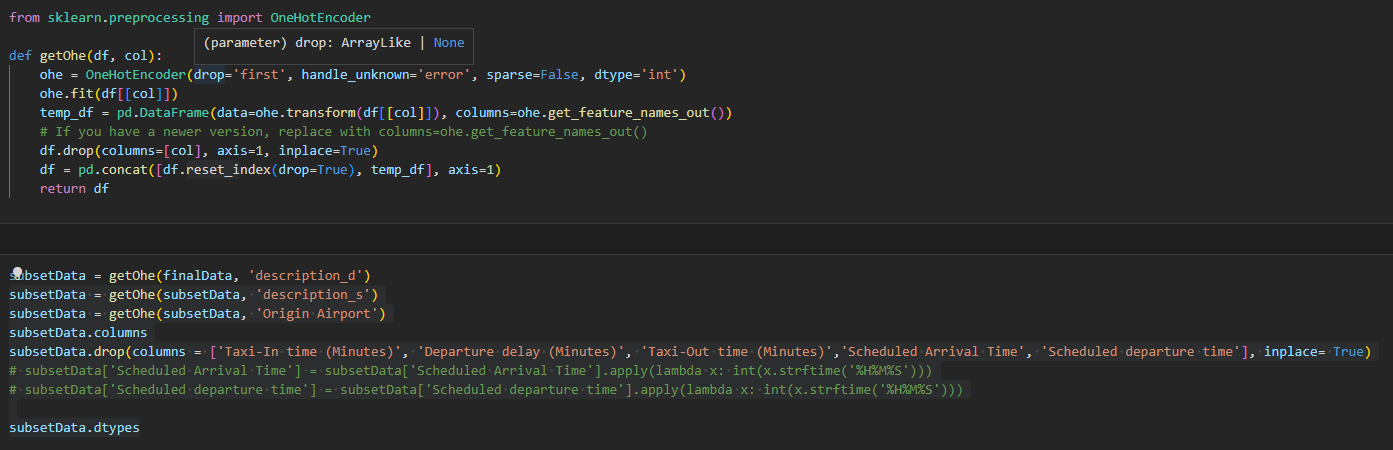




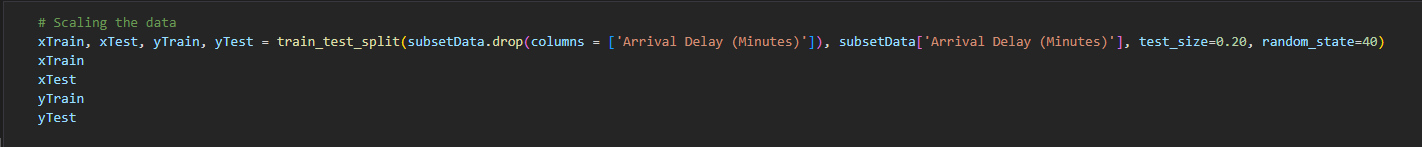


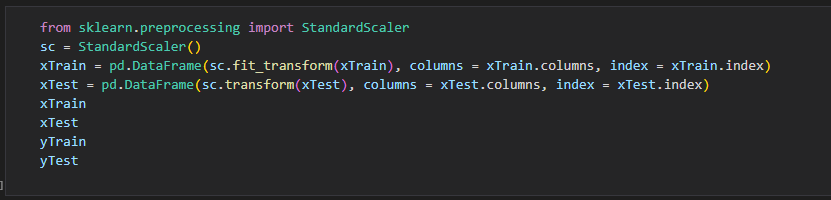
Before getting our test train split, we performed one hot encoding on the origin airport and weather description columns.

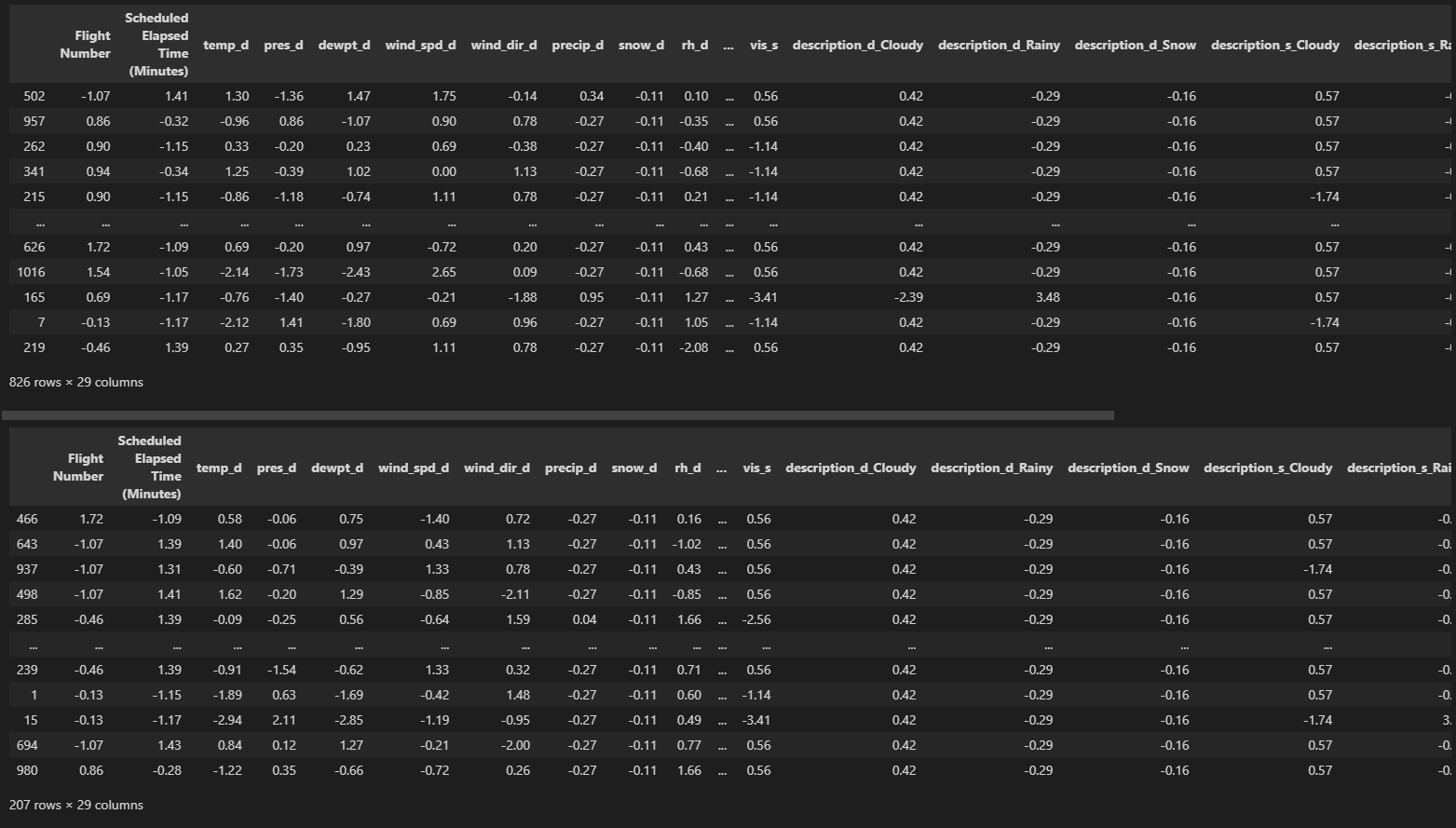


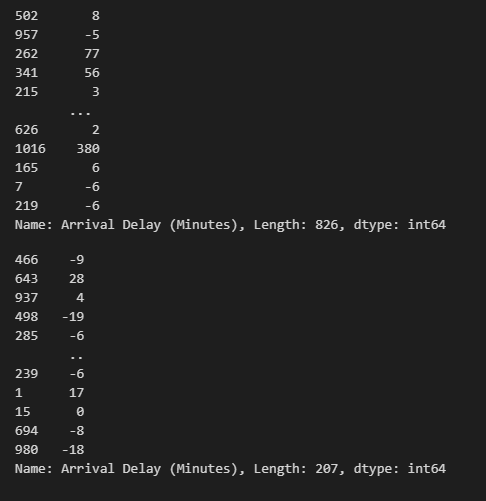


To find the scores of the ML model we split the data into training and testing data and used a standard scaler to scale the data.

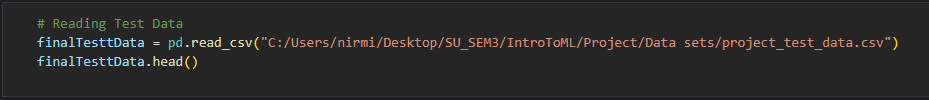


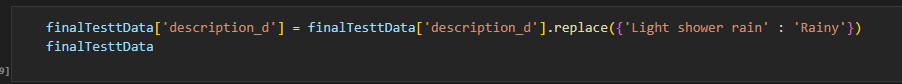


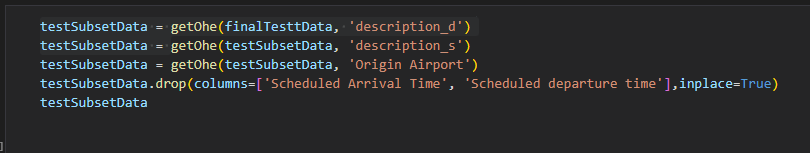


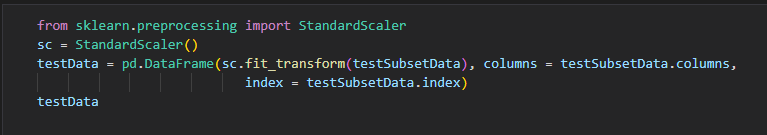


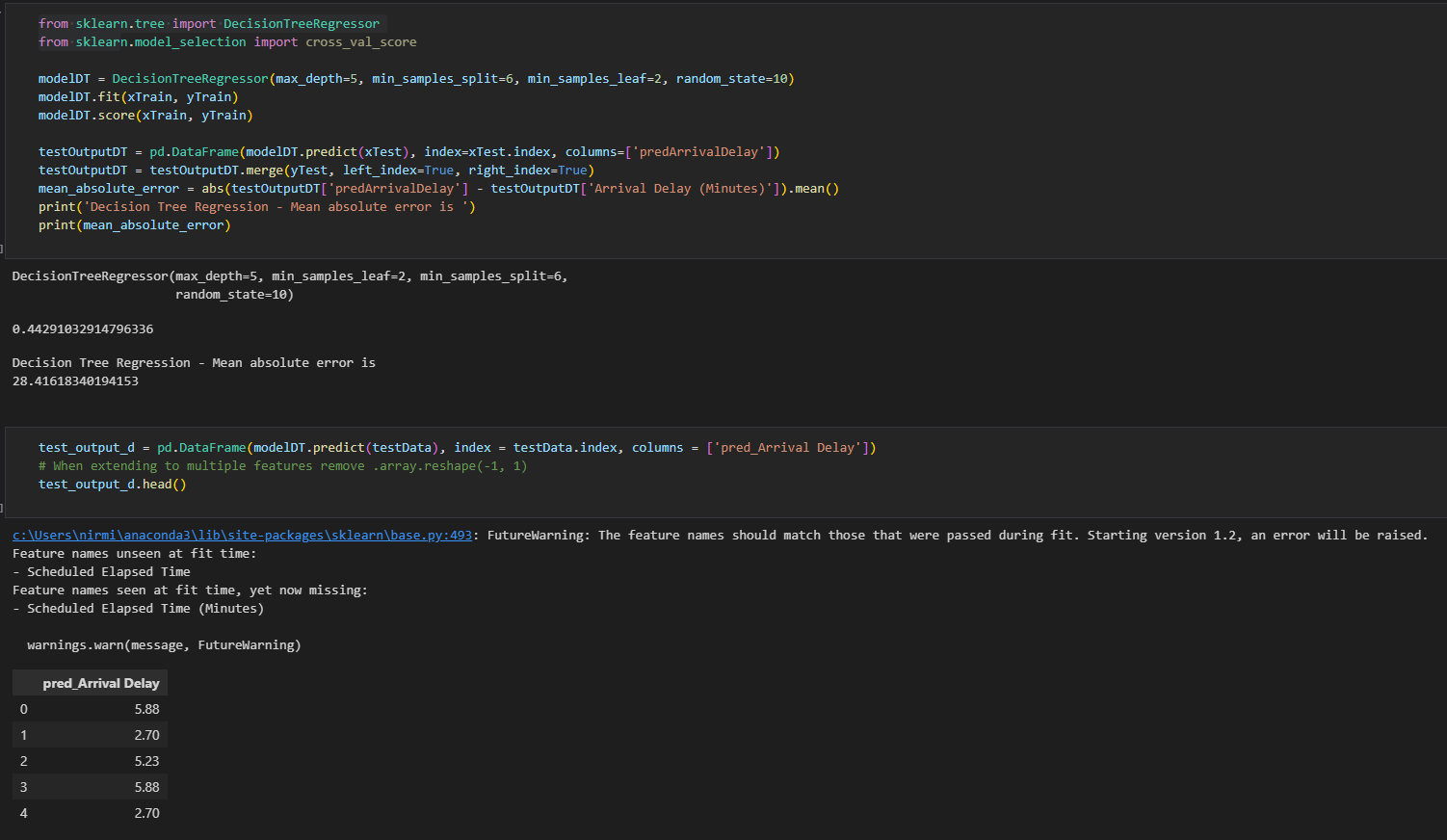
* After getting the data we fitted this data into multiple models to find out the best score and accuracy to get our final predictions.
* According to our data and scores we got the decision tree regressor that gave us the best result in terms of accuracy and score vise.
* To get our final prediction we added test data into a dataframe, cleaned the data up and got it into the same format as our test data. We used the decision tree regressor on our train data to get our final predictions and put it into a csv file.

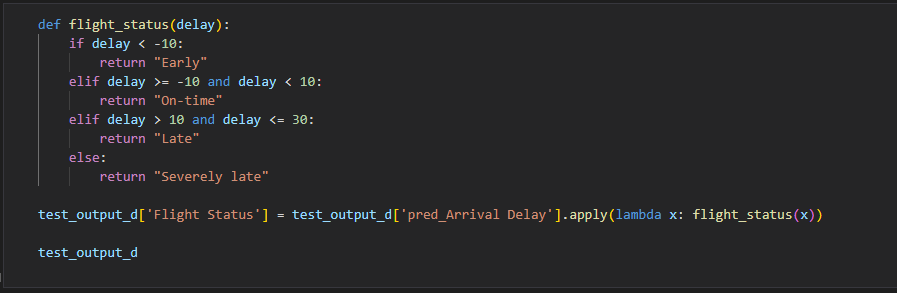












* For our final prediction we got 11 out of 32 predictions correct, that is 34.48% accuracy, that is a clear improvement over our initial prediction that was just 20%.

# Conclusion:

This project gave us an insight into the power of predictive analysis using Machine Learning algorithms like regression. During the course of this project, we employed all our knowledge from our coursework including scaling the data so that they are in similar ranges, handling categorical data as a feature, using one-hot encoding for such features and employing different models and fine tuning their parameters to get the desired results.We were also able to witness the advantages and limitations of using certain models and their effects on accuracy.

The future scope for this project is to analyze other features apart from weather that affect the flight delays like Air Traffic Control delays, technical issues, Security concerns, etc. and predict these flight delays more accurately.

# Reference:

* Flight data: <https://www.transtats.bts.gov/ontime/>
* Weather data: <https://www.weatherbit.io/account/create>
* SKlearn: https://scikit-learn.org/stable/