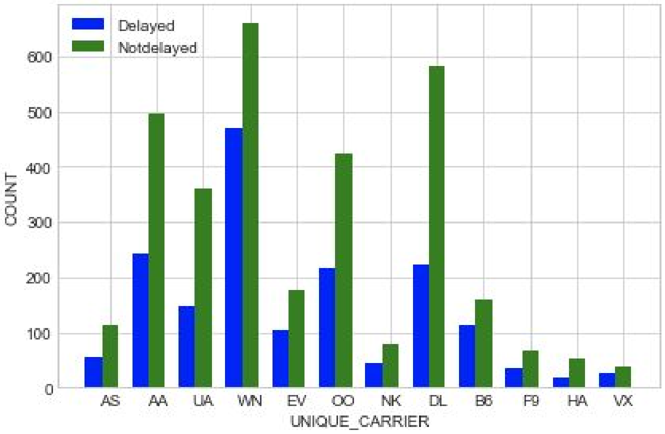
Machine Learning

Assignment 5

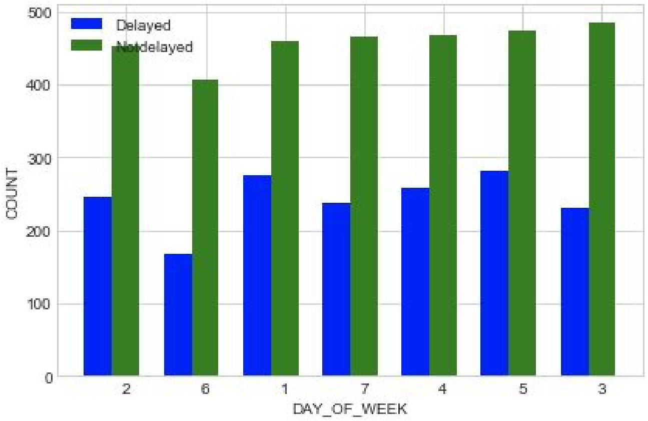
Geetesh Nikhade (gpn218), Rahul Keshwani (ryk248)

Understanding Data

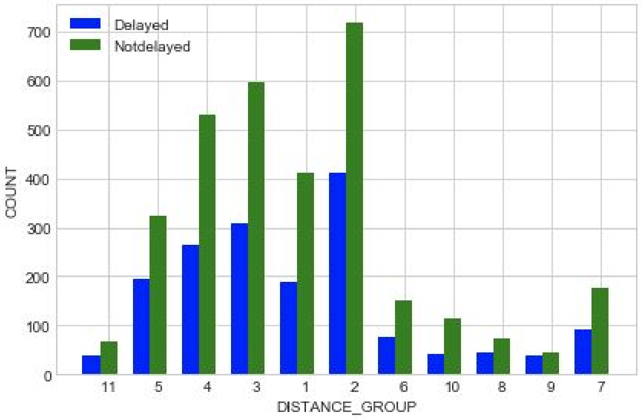
1. Wrote a Lambda function to calculate the number of NaN values in our given train data set. We found out that the column FIRST_DEP_TIME has a lot of these Null values (4882 to be precise i.e. 99.4%), and hence decided to drop it.
2. Calculating the unique values of every column in the train dataset gave us a good idea of the columns that have less unique values. We used this data to decide which columns can we perform One Hot encoding on, in the later stages of this assignment.
3. To further understand the data, we proceeded to plot various graphs:
    - The graph below shows Delayed and On-time flights according to the carriers. This shows us that the delays are carrier related, and the carrier column data should be used while predictions.



    - The graph below shows Delayed and On-time flights based of different days of the week. (Legend: Blue -Delayed, Green - Non-delayed)



- The graph below shows Delayed and On-time flights based on different distance groups. This graph shows us that there are fewer delayed flights on a few distance groups, as compared to others (Legend: Blue -Delayed, Green - Non-delayed).



Data Preprocessing

1. Using Python's replace and typecasting functions, removed a comma (,) from Distance column, and converted it into Numeric format from String format
2. We converted the FL_DATE column into a Datetime format, using Python's inbuilt datetime.strptime function
3. Performed Binning on the column CRS_DEP_TIME, converting it into hours

4. Dropped column AIRLINE_ID, as AIRLINE_ID and CARRIER correspond to same value in the dataset.
5. Dropped column FIRST_DEP_TIME, as in the data understanding stage, we had found out that it has 4882 NULL values.

Feature Extraction

1. Converted the US states to US Regions, for the columns ORIGIN_STATE_ABR and DEST_STATE_ABR. Our initial research showed us that that the states do not adversely affect the delays. Converting states to regions and applying One Hot Encoding on regions was a better option.
2. Performed One Hot Encoding on UNIQUE_CARRIERS and the above created ORIGIN_REGIONS and DESTINATION_REGIONS columns as these were string values and we would need numeric values for performing regression.
3. Performed One Hot Encoding on DAY_OF_WEEK as the numeric values of days would incorrectly act as weights during training.
4. Implemented a function to calculate days until an US holiday, and added a new column that lists days until closest holiday. We think the flight delays might be affected by the dates of the nearest holidays, assuming the number of flights increase during the holiday seasons
5. Extracted month from the FL_DATE column, and added that as a new column. Also performed One Hot Encoding on this new column as it numeric values representing months.

Training Models and Predicting Data

Linear Regression

1. After splitting our Pre-processed data into 70% train and 30% test, we used this training data set to train a Linear Regression model in the Python's scikit learn library.
2. Since predicting the Arrival delay model is a regression problem, the first model that came to our mind was Linear Regression model. We tried training and testing the above mentioned split dataset with Linear regression models of degree 1 and 2.

The results of degree 2 regression model were very bad as compared to the model with degree 1, and hence we settled with Linear Regression model of degree 1.

3. The results on our 30% validation set with Linear Regression with degree 1 are : RMSE: 45.07

Decision Tree

1. Similar to the steps in above model, we split the data set into the 70% train and 30% validation set, and used the this training set while training the Decision tree model implemented in Scikit Learn.
2. We tried different combinations of values of parameters max_depth and min_samples_leaf. Our observation was that increasing the max_depth affected the accuracy, but it looked like it was overfitting the data. Hence, the final parameter values that we settled on were max_depth=3 and min_samples_leaf=25
3. The results on our 30% validation set are: RMSE: 45.3

Selected Model & Cross Validation

The final model that we selected was Linear Regression. We settled with this model as Decision Tree model was giving us poor accuracy as compared to the Linear Regression. We tried further cleaning the data but the Root Mean Squared Error remained around 45.07.

Python Packages Used

- Scikit-learn
- Pandas
- Numpy
- Math
- Datetime
- Matplotlib