CornellEngineering
Operations Research
and Information Engineering

ORIE 4741 - LEARNING WITH BIG MESSY DATA

Will My Flight Get Delayed due to Weather?

Alex Schmack, as2968
Jackson Montijo, jhm343
Kais Baillargeon, kpb52

DECEMBER 10TH 2019

Contents

Abstract

Flight delays and cancellations cost U.S. passengers over $1.1B a year[1]. This project attempted to build a model that would accurately predict whether a flight would be delayed and then classify the severity of a delayed flight based on data from the Bureau of Transportation Statistics and the National Oceanic and Atmospheric Administration. The final user of the model could use information as simple as their plane ticket and the daily weather forecast from a phone app. This report presents the different models built to make this prediction. Hinge Loss, Random Forests, and Gradient Boosting Machines were used to try and classify flights as delayed, while Quadratic and Ordinal Regression models to predict the severity of delay were explored. The results show that the best performing classifier is the XGBoost Model and predicts duration of delay with a k-fold Cross-Validation score of .931.

1. Data Analysis

1.1 Background

The aim of this project is to train a model that predicts the duration of flight delay based on flight information and short-term weather forecasts, such as can be found on a plane ticket and phone weather forecast. Being able to predict delay duration is an important tool for businesses who heavily rely on frequent air travel. Meetings can be rescheduled in advance to accommodate potential delays, departure dates can be changed to reduce the risk of cancellation, and airline reliability in current weather conditions can inform passengers which ticket to purchase.

1.2 Our Dataset

The datasets used in this project are “On-Time : Reporting Carrier On-Time Performance (1987-present)” published by the Bureau of Transportation Statistics (BTS) and “Daily Summaries Station Details” published by the National Oceanic and Atmospheric Administration. These organizations are both government agencies with strict data policies, ensuring high data quality with no entries missing or corrupted.

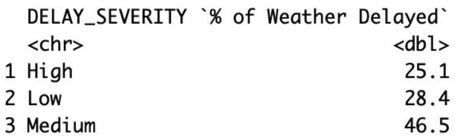
We chose daily summaries to remain consistent with the applicability of the model. If our client is trying to predict the probability of a flight getting delayed in the coming week for example, only daily weather forecasts would be available, so it would make sense to use that data as the features in our model.

From the first dataset, we took 2018 flight data and filtered the entries for the airports of Atlanta, Los Angeles, Chicago, Dallas, Denver, New York (JFK), San Francisco, Seattle, Las Vegas, and Orlando= - the top 10 busiest US airports by passenger traffic. We then joined this with daily airport summaries from the second dataset, to obtain daily weather for each flight. When merged, the full dataset has 413,565 entries and 33 features.

Of the 33 features, 2 are ordinal, 18 are discrete, and 13 are continuous. They include flight statistics - such as departure date and airline - and weather conditions - such as wind, temperature, and significant weather events such as fog or thunderstorms.

1.3 Data Statistics

Our dataset is very unbalanced with only 3,818 flights delayed due to weather out of 413,565, representing just 0.92% of the total dataset. To understand these weather delays in more detail, we created a simple weather delay severity categorization of "Low" for 0-15 minutes, "Medium" for 15-60 minutes, and "High" for 60+ minutes. The proportion of total weather delayed flights within these categories.



```
  DELAY_SEVERITY `% of Weather Delayed`
  <chr>                           <dbl>
1 High                             25.1
2 Low                              28.4
3 Medium                           46.5
```

Table 1: Delay Severity

Table 1 shows that Low category delays do not make up the majority, with High and Medium delays combined representing 71.6%. This means that typically a flight delayed due to weather experiences significant delays, which will help our model identify highly disrupted flights more accurately.

We went on to incorporate hourly weather data from the same source into our models. We extracted the numeric values from this dataset and used a correlation plot to visualize which variables could influence the delay duration due to weather.

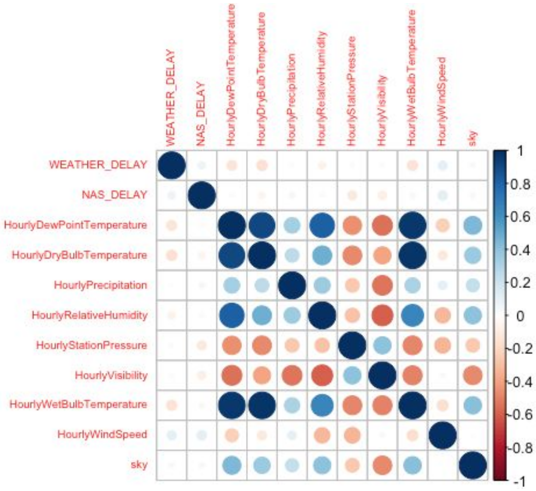


Figure 1: Correlation Visualization

As seen in Figure 1, our dependent variable "WEATHER_DELAY" is related to features describing hourly temperature, wind speed, and humidity. To understand how the mix of positively and negatively correlated values influenced "WEATHER_DELAY", we first built a plot to show how the features HourlyDryBulbTemperature (Air Temperature) and HourlyWindSpeed related to flight delays, coloring points based on their severity.

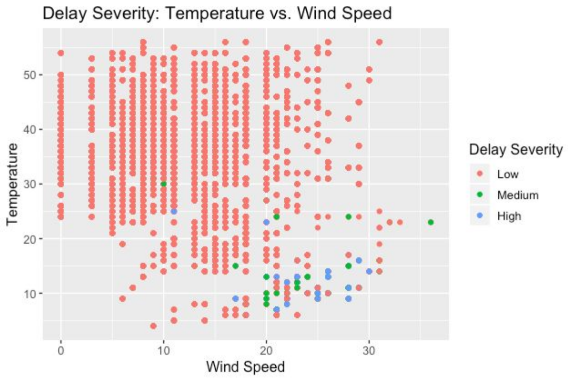


Figure 2: Temperature vs. Wind Speed

Figure 2 shows how as temperature decreases and wind speed increases, flights tend to be classified as High and Medium delay more often. This is logical, as high winds and cold weather conditions are likely to result in winter storm conditions. We did the same for HourlyDewPointTemperature (Temperature at which water vapor will transition to liquid form) and HourlyRelativeHumidity.

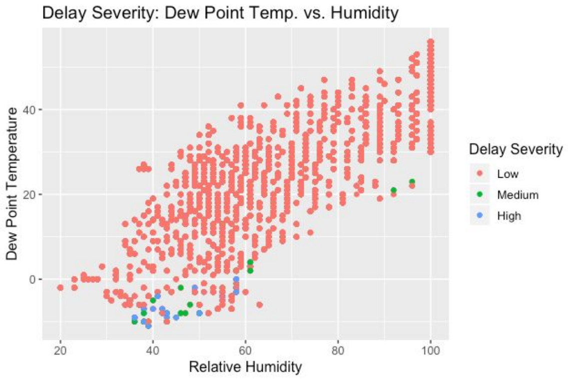
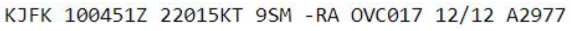


Figure 3: Dew Point Temp. vs. Humidity

Figure 3 shows how as dew point temperature decreases and the relative humidity decreases, flights tend to be classified as High and Medium delay more often. Again, this makes sense as relatively high humidity coupled with a low dew point suggest snowy weather.

1.4 Feature Engineering

Our initial feature vector took all the weather information we had available: maximum and minimum daily temperature, daily precipitation and cloud coverage. Non-numerical weather data comes in the form of a Meteorological Terminal Aviation Routine Weather Report (METAR). This report contains information such as the time, airport observed, wind speed, and visibility.



```
KJFK 100451Z 22015KT 9SM -RA OVC017 12/12 A2977
```

JFK METAR reporting rain and overcast skies

This report also highlights other significant weather events such as rain, snow, mist or fog. We encoded these events using one hot encoding by scanning each METAR string and creating a one-hot vector if each event occured. We also encoded cloud coverage from text format to an ordinal feature with 0 being clear skies and 6 being overcast skies.

The most important feature engineering we performed was on our output vector. The Department of Transportation classifies delays based on different categories: Carrier Delay, Weather Delay, National Air System (NAS) Delay, Security Delay or Late Aircraft Delay. NAS Delays are those affecting the national airspace and can be anything from an Air Traffic Control strike to non-extreme weather conditions.

However, the FAA does not report the reason attributed to a NAS delay. To produce an output vector with delay time due to weather, we must implement a method to split all NAS delays into NAS delays due to weather and those not due to weather. To achieve this, we used a K-means clustering method on our weather vectors, excluding all features related to airline and airport.

Our clustering methodology involved creating K clusters of flights based on the weather features, and then comparing clusters based on the ratio of flight examples with a NAS delay to the total number of flights in the cluster. We can assume that NAS delays not due to weather occur randomly within our weather vector feature space. Based on this, the clusters with the highest ratio of NAS delays contain the flights with their NAS delays occurring due to weather. According to the Bureau of Transportation Statistics, 55.0% of NAS delays can be attributed to weather[3], so we accepted flight delays from clusters in descending order of NAS ratio, until we had accepted 55% of all NAS delays.

We determined the K amount of clusters needed by searching for an elbow when plotting the distortion scores of the clusters for various Ks.

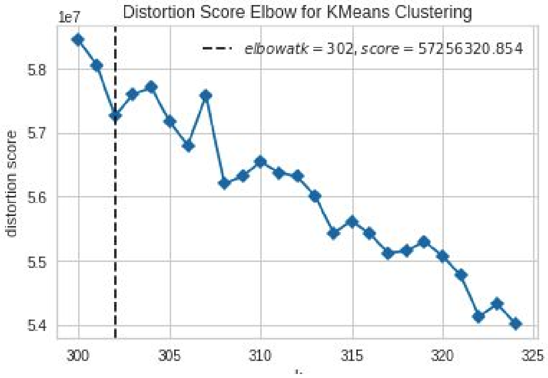


Figure 4: k-means clustering

These results can be seen in Figure 4, where the optimal number k=302 delivers the tightest clusters, without over-separating the data.

2. Model Selection - Delay Classification

To best predict the expected time of a flight delay due to weather, we first classify flights as delayed or on-schedule. Splitting the problem in this way is one method we use to account for the high zero-inflation in our data; there is a large imbalance between examples of delayed flights and examples of on-schedule flights. In addition to this technique, we randomly throw out examples of on-schedule flights in our training dataset to achieve approximately a 50/50 split of on-schedule and delayed examples.

2.1 Stochastic Gradient Descent - Hinge loss

The first technique we used to classify flights based upon weather delay was fitting a Generalized Linear Model (GLM) using the Hinge Loss Function with a L2 regularizer. These were chosen as we are solving a classification problem with regression. We optimized this model using Stochastic Gradient Descent (SGD) due to its speed performance compared to Gradient Descent. We trained this model using all of our engineered features with poor results. We suspected we were overfitting to less important features, such as one-hot encoded airline operator, and then reduced our features to only weather features.

2.2 Random Forest

The second method we used to classify flights is Random Forests. The drawback of this type of model is that the classifications are not easily interpretable: the complex trees used in this method make it hard to understand why a particular flight was classified as it was but are able to see which features were most important. Despite this limitation, Random Forests do produce quality results. Again, we only used our weather features, as Random Forests are known to overfit. We implemented a Random Forest Classifier consisting of 300 random trees of a maximum search depth of 18. Choosing a high number of trees and low maximum search depth can also help alleviate overfitting.

2.3 Gradient Boosting Machine

The final method we used for classification is XGBoost, a parallel tree boosting library implementing machine learning algorithms developed under the Gradient Boosting framework[2]. As it also relies on trees, this method suffers from similar interpretability limitations as the Random Forest. The difference between the two methods is in how they attempt to reduce error:

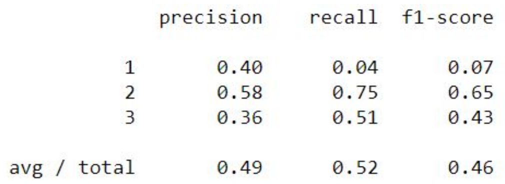
$$error = bias + variance$$

While Random Forests uses large, deep decision trees to maximize the decrease in variance, XGBoost uses boosting methods to create small, shallow trees that maximize the decrease in bias. We used the same feature set as with Random Forest to train our XGBoost model and tuned a classifier with 100 estimators, max tree depth of 6, and minimum child weight of 5. These parameters were 3 of 6 that were tuned to minimize the chances of overfitting while maximizing accuracy.

3. Model Selection - Duration

After obtaining good results in our classification model, we then predict the duration of a delay using a regression model. Our training set for this model consisted of all of the delayed flights in 2018 for a total of 23,925 flights. We originally attempted to use a proximal gradient model with quadratic loss and a quadratic regularizer. Given the sparsity of the data, as well as the limitations on the features described earlier in the report, our model was having a hard time predicting an exact delay time, but was approximating the mean fairly well.

We then decided to use an ordinal regression instead, and created three different “buckets” of delayed flights based on delay duration. These were low (0-15 mins), medium (15-45 mins) and high (45+ mins) severity delays. We tested multiple loss functions of our model, specifically hinge, log, modified huber as well as perceptron. Out of all of these, the best model was an Ordinal Hinge Loss SGD Classifier with a quadratic regularizer.



```
             precision    recall  f1-score

          1       0.40      0.04      0.07
          2       0.58      0.75      0.65
          3       0.36      0.51      0.43

avg / total       0.49      0.52      0.46
```

Table 2: Hinge Loss Results

Figure 6 shows that we obtained an average precision of 49% with a recall of 52%. Although the model did not perform fully as expected, we were not surprised, given that there are many more factors that determine the duration length of a flight, some outside the scope of weather metrics. We were still optimistic that the model was more precise with larger delays, meaning that it can predict longer delays better, which is ultimately the interest of our model.

4. Model Results & Comparison

4.1 Delay Classification

To evaluate the performance of our models we examine both the ROC curves and precision/recall matrices. In the context of our model, delay classification precision/recall are the crucial metrics. We want high precision so we can believe our model if it classifies a flight as a delay, but our recall high so we can properly detect as many possible. The two are typically inversely related when there is a large class imbalance. Our results are summarized in Table 3 below.

Precision/Recall	SGD	RT	XGB
On-Schedule	.91/.03	.96/.66	.96/.65
Delay	.07/.96	.12/.63	.12/.65

Table 3: Classification Results

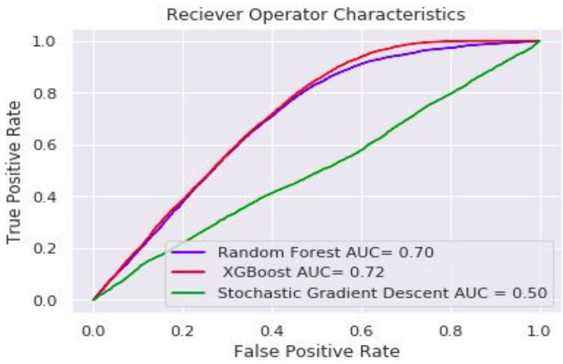


Figure 5: Classification ROC Curves

Although the ROC curve in Figure 5 displays okay results, the precision and recall reveal an issue. We are identifying above 60% of delays in all three models, but with very low precision. We are able to find many delays, but often misclassify on-schedule flights. This is a fault of the data. Because our dataset contains daily weather examples, our model tends to underfit to non-weather features, and thus tends to only classify entire days as delayed or on-schedule. This results in low precision; if an entire day is classified as delayed, there are many incorrectly classified on-schedule flights on that day. It also impacts our recall: if a day is classified as on-schedule, we miss delayed flights on that day. This motivated our decision to try and train a model on hourly data; despite the inconsistent time intervals and large amounts of missing data in these datasets. Using the same three models on the new hourly dataset yielded the results found in Table 4.

Precision/Recall	SGD	RT	XGB
On-Schedule	.94/.40	.92/.84	.91/.97
Delay	.15/.79	.43/.35	.56/.26

Table 4: Classification Results - Hourly

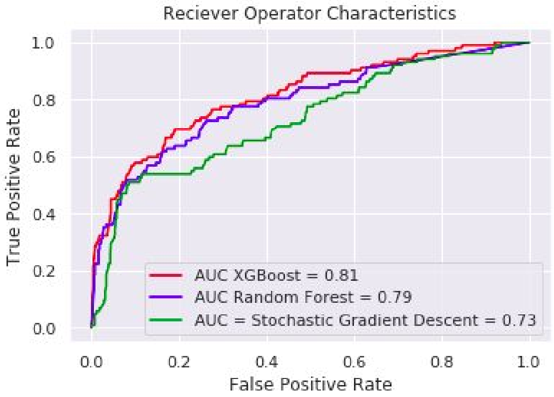


Figure 6: Classification ROC Curves - Hourly

Comparing to the daily-feature results we can clearly see from Figure 6 that the area under the ROC curve is higher for all models using the hourly weather features; we no longer get stuck into daily bins. Examining the precision/recall matrix, we see our precision for delay classifications has risen significantly, although at the cost of recall. The best model to classify flights was XGBoost in both the daily and hourly weather feature cases, achieving the highest recall and precision in the daily case and 56% precision and 26% recall in the hourly case for a total accuracy of 89%.

	SGD	RT	XGB
K-fold Score	.891	.869	.892

Table 5: k-fold validation

As shown in table 5, after validating our models using K-fold cross validation we achieved the highest accuracy of 89.2% using XGBoost.

5. Follow-on Opportunities

5.1 Quality of Data

There were two major limiting factors of our dataset. The first being the lack of a reported delay due to weather. The lack of this data forced us to create our own output vector using our clustering method described in the Feature Engineering section. If this metric was reported by FAA, we expect the quality of model would increase greatly. The second limiting factor of our data is the lack of hourly weather data. We only have access to weather statistics reported on a daily basis. This again severely limits our model; every flight on a given day should not have the same weather features. If this data were tracked, we could join our flight data to weather data by the hour as opposed to day, which would greatly increase our ability to accurately classify and predict the time of weather delays.

5.2 Over-Fitting

We used an 80-20 train-test split on all of 2018's yearly data. We needed at least one full year of data to account for seasonality in our model, and the random 80-20 split was able to capture it.

6. Conclusion

6.1 Conclusion

By using techniques such as one-hot encoding, quadratic loss, and gradient boosting machines, we were able to develop a model to predict flight delays. Our best performing delay classification model using XGboost was able to classify flights fairly well with 89% accuracy.

Taking a classified flight as input, our delay duration model using quadratic loss was able to predict the ordinal severity of delay with 89.2% accuracy.

Although, our model does not predict flight delays with extremely high accuracy and should not be used for extremely time-sensitive travel, it does accurately predict flight delays in the long run. It would be ideal for frequent flyers with a degree of travel flexibility. We strongly believe our model would result in significant savings for clients such as consulting companies, who travel weekly to and from the client site.

Furthermore, our model provides high value to clients conducting wider flight delay cost-benefit analyses. As our predictions are better than a simple guess, we can help calculate more accurate expected costs due to flight disruptions. This is especially useful to clients with regular, keystone events such as trade shows or board meetings to which bad attendance can not be tolerated.

In this context, we believe clients will find our model quite useful due to its ease of use and relative accuracy. All stakeholders, regardless of technical proficiency, can quickly predict results as all input data can be found on a plane ticket and phone weather forecast. We believe this is a strength of our model over more complex ones with inputs from sophisticated sensors or proprietary data sources.

Ultimately, we believe this model can be a useful tool in reducing the over $1.1B that for U.S. passengers' lose to flight delays every year.

Bibliography

1. *"Air Traffic by the Numbers." FAA - Federal Aviation Administration, June 2019, www.faa.gov/air_traffic/by_the_numbers/media/Air_Traffic_by_the_Numbers_2019.pdf.*
2. *"XGBoost Documentation." XGBoost , 2019, xgboost.readthedocs.io/en/latest/.*
3. *"Understanding the Reporting of Cause of Flight Delays and Cancellations." Bureau of Transportation Statistics - U.S. Department of Transportation, www.bts.gov/topics/airlines-and-airports/understanding-reporting-causes-flight-delays-and-cancellations*