STA 325: Final Project Report

Calleigh Smith, Hannah Bogomilsky, Hugh Esterson, Maria Henriquez, Mariana Izon

November 22, 2020

Introduction

The importance of air travel in the United States is unparalleled, connecting distant parts of the country with the aviation industry's hallmarks of efficiency, safety, and reliablility. This mode of transportation offers citizens to conduct business, visit loved ones, and travel for pleasure, and the number of Americans flying is widely on the climb. In fact, in 2019, U.S. airlines carried a staggering 925.5 million passengers, a record-setting number, and a healthy increase of 4.1% over the previous year. However, what is the most prominent complaint from these 925.5 million clients? Perhaps unsurprisingly, the answer is delayed flights.

Simply, flight delays brings into question a given airline's devotion to efficiency and reliability, and when such efforts are not met, disgruntled passengers are sure to become an issue. Arrival delays do not occur all too seldomly, with 19.95% of flights incurring arrival delays in 2019, according to the Department of Transportation's Bureau of Statistics. A slew of research has also shown that flight delays, and the ensuing negative reactions by passengers, have consequential effects for all involved, affecting customers' airline choice, as well as their spending habits at a given airport. Thus, it is in the best interest of all parties (customers, airlines, and airport management) to ensure that the maximum number of flights are completed without delay. This goal, of course, is not realistically achieveable, 100% of the time. Yet a model in which to predict arrival delays could benefit all parties involved, offering a better understanding of the duration of any delay and allowing customers and providers to plan to optimize the situation at hand for their collective benefit.

With this thought in mind, our group has taken on the task of using machine learning methods to form a model that accurately predicts arrival delays for real-world flights. By focusing on a popular flight route, within a specific interval of time, our team hopes to accurately predict arrival delays, while also [keeping in mind] the interpretability of available predictors, which could help in an explanation in the primary factors of flight delays. Using publicly-accessible data, the team also aims to provide findings that are readily reproducible and interpretable for all audiences, whether it be fellow passengers or airline executives.

In order to form such a model, we will use various model-building techniques, including multiple linear regression, generalized additive models, and tree-based regression. Informed by statistical measures of goodness-of-fit, variable selection, diagnostic checks, and in-depth exploratory data analysis, the project will help to develop a choice of a specific model across the viable options. Specifically by comparing relative error metrics across the different types of models, a final machine learning model will be fully explained and interpreted, weighing the relative pros and cons of each statistical decision. Future directions of the project will also be discussed, hoping to draw generalized, yet accurate, conclusions from our dataset and model to the large-scale topic of flight delays across the American aviation industry.

Data

Data Background & Cleaning

The data used within this final project originates from the United States Department of Transportation's Bureau of Transportation Statistics. Specifically, the team has downloaded the publicly-accessible, government data from their Airline On-Time Performance[link: https://www.transtats.bts.gov/Tables.asp?DB_ID=120]

database, using a subset of the data entitled "Reporting Carrier On-Time Performance." This portion of the database records all relevant data for all non-stop flights of major U.S. airlines. It is updated monthly, dating back to 1987, and includes a plethora of informative variables.

The Bureau's website allows for a direct download of the dataset for a given month and year by means of a .CSV file. For purposes of this project, the team opted to choose January 2020 as our time period of interest. Several considerations were involved in this decision, including the choice of a recent month that was not severely affected by the COVID-19 pandemic. Thus, the data collected from this month will not showcase the drastic and devastating effect that the pandemic has had on air travel traffic. We also chose to focus on a specific non-stop route within this month of data. Since four of our give group members came from either New York or California, we chose to view flights originating from New York's John F. Kennedy Airport (JFK) and arriving in California. While there were initially 10 such routes that departed from JFK and arrived in the state of California, we again narrowed our focus to those flights bringing passengers to San Francisco International Airport (SFO) and Los Angeles International Airport (LAX). This choice was made due to the fact that these two airports were the only two that had flights serviced by each carrier in the dataset, with JFK-SFO and especailly JFK-LAX being among the busiest domesitc air routes. This inital sorting was completed locally within Microsoft Excel and Numbers, before being uploaded to RStudio as a .CSV file.

Generally, the variables included in the dataset fall under a few holistic categories. Firstly, there are certain time-based variables, including $DayofWeek$, $DayofMonth$, and scheduled departure (CRS_DEP_TIME) and arrival times (CRS_DEP_TIME). Of course, route-based information is included with the $ORIGIN$ and $DEST$ variables. In our case, the origin was JFK for each observation, while the destination varied between SFO and LAX. Flight-based statistics, such as the reporting airline, departure delay, and taxi time, both prior to departure ($TAXI\_OUT$) and upon arrival ($TAXI\_IN$), offered additional information on each flight. For this project, four U.S. mainline carriers are represented, namely American (AA), Delta (DL), Alaska Airlines (AS), and JetBlue (B6). Finally, several variables corresponded with delay times for any of five reasons: carrier delays, weather delays, National Air System delays, security delays, or late aircraft delays. Finally, as suggested by our aforementioned modeling objective, the arrival delay ARR_DELAY acts as our response variable within this project.

Data Transformations

Within R, the data was cleaned once again. Here, errant NA values were removed and the *dplyr* package allowed for more in-depth filtering. The *mutate()* function was also used to change certain variables. One such example were the delay-based variables, which we decided to mutate into a categorical predictor that listed the type of delay that a given flight might have experienced. Changes such as these ensured that the chances of multicollinearity were reduced, as the original dataset reported delay times that, additively, could have been used to nearly exactly predict the arrival delay, thus, leaving the modeling objective with no true predictive learning potential.

An initial exploratory data analysis also uncovered a few predictors that suggested some transformations to ensure normality for linear regression. Two such predictors were the taxi times, both prior to departure and upon arrival. The departure taxi time ($TAXI\_OUT$) reports the time from pushback from the departure gate to the time of "wheels up" upon takeoff, while the arrival taxi time ($TAXI\_OUT$) would denote the time from "wheels down" upon landing to parking at the arrival gate. A histogram of these two predictors showed fairly significant rightward skew. This result is not necessarily unexpected, as monumental ground delays or other issues might lead to several extreme values in towards the upper tails. Nonetheless, for matters of transformations, we chose to explore the usage of a log-transformation, yielding histograms that exhibited distributions that much more closely resembled a normal spread.

It is also important to note the manner in which the response variable is reported within the original dataset. Flights arriving prior to their scheduled time yield a negative value of ARR_DELAY, while flights that are late post a positive value. As such, our model will aim to predict both flight delay times, yet also the duration by which a given flight might be early.

Like the taxi times, a similar skew was found in both the predictor of departure delay and the response of arrival delay. For the response variable, we chose to undergo a Box-Cox transformation. However, this

transformation proved to be somewhat challenging as the numerical response variable of ARR_DELAY includes both positive and negative values. As such, we chose to first transform the variable, increasing each value by a set amount, ensuring that the minimum value of the variable was non-negative. From here, a Box-Cox transformation was performed.

Other interesting findings from our inital exploratory data analysis included finding empirical means and histograms of several predictors and the response. Additionally, from the days of week and days of month histograms and exploratory data analysis, we interestingly discovered that Saturday was an overall slow day in terms of numbers of flights, with a histogram of the number of flights across the days of the month of January 2020 showing the same trend.

Methods

Multiple Linear Regression

In order to understand some of the underlying relationships in our data, we began our modeling with various iterations of multiple linear regression. Our first approach was to fit a model with all of our predictors, including the log-transformed predictors, $TAXI\_OUT$ and $TAXI\_IN$. We performed model selection using AIC as the criterion to get rid of some insignificant predictors, including DAY_OF_WEEK and CRS_ARR_TIME. We then used ANOVA testing to test various interaction terms based on our prior knowledge of these variables and found the following relationships significant in this log-transformed MLR model: airline carrier ($OP\_CARRIER$) & destination ($DEST$), time to taxi in (log_TAXI_IN) & destination ($DEST$), and time to taxi out (log_TAXI_OUT) and departure delay (DEP_DELAY).

In order to have a baseline model to compare our choices, we decided to fit a multiple linear regression model with the untransformed predictors and no interactions. Once again, we performed model selection using AIC as the criterion. This time, only DAY_OF_WEEK was found to be insignificant. We also tested out interactions, but surprisingly, none of our interaction effects were found to be significant through ANOVA testing. Therefore, our simple multiple linear model just included the untransformed predictors and no interaction effects.

Finally, we decided to fit our simple multiple linear model with the Box-Cox transformed response variable ARR_DELAY, concluding our iterations of multiple linear regression models. After finding the test MSE for each of our three models - (1) MLR with log transformed predictors ($TAXI\_IN$ and $TAXI\_OUT$), interactions, and no Box-Cox transformed response, (2) MLR without any interactions or transformations, (3) MLR with Box-Cox transformed response, no interactions or transformations on the predictors - we were surprised to find that model (2), the simplest iteration, performed substantially better than the rest.

Moreover, upon looking at the model diagnostics for each of the three models, it appeared that most of the plots looked reasonable for a linear model fit. However, the normal QQ plots for each of the three models showed signs of deviation from normality, prompting us to pursue more cutting-edge machine learning models, such as the GAM and tree-based regression.

Generalized Additive Modeling

After fitting three multiple linear regression models, the next step in our process was to fit general additive models (GAMs) in order to explore a more complex model. In general, generalized additive models are relatively computationally inexpensive tasks, allows for rather easy inference, and has the ability to model highly complex nonlinear relationships. Furthermore, it allows us to explore the individual relationships between each of the predictors and the response variable. Thus, we expected our GAM models to perform well.

We first fit a GAM with categorical variables and a smoothing spline on the numerical variables. After checking the linearity, we found that $TAXI\_IN$ may possibly be linear. We conducted an ANOVA test comparing the original GAM to a model without a smoothing spline on $TAXI\_IN$, and concluded the model with a smoothing spline on $TAXI\_IN$ was a better fit. In addition, DAY_OF_WEEK, $DEST$, and

CRS_ARR_TIME had very high p-values, so we ran another ANOVA test excluding those variables. Based on the results of that ANOVA test, the model removing these variables was a better fit. This model included $OP\_CARRIER$, a smoothing spline on $TAXI\_IN$, a smoothing spline on $TAXI\_OUT$, a smoothing spline on DEP_DELAY, a smoothing spline on CRS_DEP_TIME, and $TYPE\_DELAY$. When checking model diagnostics, it appeared that the plots looked reasonable with normally distributed residuals.

We then fit the same GAM on the Box-Cox transformed response variable. The model diagnostics yielded reasonable plots, again with normally distributed residuals. However, this model had a higher MSE than the GAM fit on the untransformed response variable. Therefore, we concluded that the GAM on the original response variable was a better fit.

Tree-Based Regression

Given what we know about trees, we decided to attempt two of the more cutting edge tree-based regression models: random forests and boosting. For random forest, we used the `randomForest()` function and used the untransformed response and predictors that were considered important in our GAM model. For random forests, the one parameter to tune is `mtry`, which is the number of variables randomly sampled as candidates at each fit. The `randomForest()` function usually chooses a default of `p/3` for regression, but we decided to use cross-validation to find the most appropriate choice for `mtry`, which ended up being 2. Though somewhat computationally expensive, we decided to build 10,000 trees to inform our model. Choosing a large number of trees reduces the variance of each of the fitted trees, which is one of the benefits of the random forest approach.

The second tree-based regression approach we tried was boosting on the same untransformed response and predictor variables. Boosting is beneficial because it allows one to control the interaction depth (`interaction.depth`), number of trees (`n.trees`), and the learning rate of the model (`shrinkage`). In order to find the optimal parameters, we used the `train()` function in the `caret` package, ultimately settling on 3, 150, and 0.1, as determined through cross-validation.

Results

Future Directions

- could expand airports, years, COVID effects, etc.

Notes & References