**Capstone milestone report draft**

By Will Bishop

Updated January 20, 2017

**What is the problem? Who is the client?**

The problem

To predict the on-time performance of a flight based on:

* Origin and destination cities
* Airline
* Time of year, time of day, day of week
* Aircraft type and number of seats

These are the variables I have identified as likely to hold predictive power, although there are others in the data set.

The dependent variables are various measures of arrival delay, the most basic being simply the number of minutes between the scheduled and actual arrival times, as well as a dummy variable simply indicating whether the flight was delayed by 15 minutes or more. There are also five variables that break down the number of minutes of delay into different causes: carrier delay, weather delay, National Air System delay, security delay, and late aircraft delay (meaning that the previous flight using the same aircraft arrived late). However, these variables for the separate types of delay are missing for a large number of observations, so I have focused on the total delay time and the 15-minute dummy.

The client

An example client is Google and its Google Flights application. The idea is that when helping travelers decide on the best flights, this kind of app could consider the probability of a significant delay as well as more traditional metrics such as price, comfort, and itinerary duration. A difference in expected delay time may offset some of the difference in price depending on the traveler’s priorities.

**What important fields and information does the data set have?**

The data come from the Bureau of Transportation Statistics’ On-Time Performance table: <http://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=236&DB_Short_Name=On-Time>. They have data on every flight of major U.S. airlines in monthly tables. The latest available data was from May 2016 at the time I started the project, and the data seem to go back to at least 1990. However, using all the data from 1990 would not only crash my computer, it would not be very useful because the airline industry and characteristics of airports have changed so dramatically since 1990. Instead I am using a five-year sample from June 2011 to May 2016. Using all the flights, even for only five years, would be too large a data set for my computer to handle, so I’m using a 5% random sample.

Important fields that the data set includes are:

* Origin and destination airports
* Airline
* Time of day, week, and year

**What are some limitations of the data set?**

* It does not include the type of aircraft, number of seats, or number of passengers on the plane. I thought it would be interesting to see whether full and empty flights had different delay rates. The load factor data does not seem essential for the goal of the project, since the purpose of an app like Google Flights is to help passengers pick flights well in advance. A regression model based on the variables mentioned above would presumably encode the probability of a full flight and any effect that would have on delay time. However, the type of plane is often known in advance, so it could theoretically be a factor worth modeling.
* It does not include weather data, so we can’t answer questions about micro-level effects of specific weather patterns. Again, this variation should be encoded in a model that includes origin and destination cities, with the possible exception of flights booked on short enough notice that it is possible to predict weather beyond seasonal patterns.

**What kind of cleaning and wrangling did I do?**

I had to download the data one month at a time, then turn it into Pandas DataFrames and concatenate them all. I used a function that took advantage of similarities in the files’ names, which may not have been the most efficient way but that part only needed to be done once—the five-year sample is saved on my computer now.

For the most part, the data came in a clean format. A few cleaning steps I did include:

* Merged on a data set that contained airline names. The original files had only IATA codes for airlines, but BTS provides a separate crosswalk between codes and names.
* Merged on a data set that included the names of “markets” rather than cities. This allows analysis by metropolitan area as well as individual airports, which is interesting, though not strictly necessary for the goal of the project.
* Created variables that count the number of departures by city and market, and the number of arrivals by city and market. (This was mainly for descriptive statistics purposes.)
* Merged on a data set that includes airport latitudes and longitudes. Again, any variation in delay times caused by these factors should be accounted for in the airport variable itself. However, I wanted to test a hypothesis that bad-weather seasons were a less important factor than busy-travel seasons in predicting delay times.
* Grouped data by month in order to create a time-series plot and see whether an overall time trend in delay times exists. (This has been a frustrating step and taken more time out of the project than I hoped!)

**Are there any other data sets I can find, use and combine to answer the questions that matter?**

I think in terms of the fundamental question the client needs to answer, the BTS data plus the few other data sets I merged onto it have all the necessary variables (although I’m open to suggestions). Load factor data would be interesting, and the BTS does have it, but so far I have only found it aggregated by airline and time period rather than for every individual flight. Data on the type of airplane used would be even more interesting, but seems much less likely to be easy to find.

**Preliminary exploration and findings**

My initial exploration (descriptive statistics only, no modeling) can be found in the Data Story assignment. Some initial findings—which could, of course, change with a more sophisticated model—include:

* Airline name is a very strong predictor of delay. Spirit Air Lines is the least timely airline by a wide margin, both in terms of 15-minute delay rate (30%) and average delay time (14 minutes). Frontier Airlines is second by both measures. On the other end, Alaska, Hawaiian, and Delta rank as the three best airlines by both measures; Hawaiian has the lowest delay rate (8%) while Alaska has the lowest average delay time (-1.4 minutes, i.e., flights take off *before* their schedule departure on average). But we have to be a bit careful here: is Hawaiian Airlines stellar, or is there something else about flying to Hawaii that makes flights tend to be on time?
* Time of day is also a strong predictor, with flights in the afternoon and evening 2-3 times as likely to be delayed as those in the early morning.
* Arrival cities have widely varying delay rates. Among cities with more than 2,000 flights in the database, 15-minute delay rates range from 10% to 26% and average delay times range from -0.2 to 12.3 minutes. Departure cities also vary widely: 15-minute delay rates from 9% to 25% and average delay times from -0.9 to 9.2 minutes. Four of the ten cities with the lowest delay rates are in Hawaii. However, since there are so many different cities, a comprehensive picture of their effect on delay rates cannot be achieved by listing individual cities—we need a regression model.
* Day of week does not appear to be a strong predictor.
* June, July, and December are the months with the highest delay rates, while September, October, and November have the lowest. The differences between months are much smaller than those between airlines, times of day, or cities.
* Flights a few days to either side of major travel holidays (Christmas, New Year, July 4) appear to have high delay rates, while flights on the holidays themselves have low rates; however, there is a good bit of randomness in the data at a level this granular.
* Flights overall don’t appear to have gotten later or earlier over the 5-year period.

Descriptive statistics do not prove anything about causal effects—for instance, the fact that delay rates vary greatly by time of day does not prove that this is the most important causal factor. It is just an initial indicator of where the data might be leading us.

**Approach**

A regression model for this data will require very large numbers of dummy variables and interaction variables to isolate one effect from another. For example, how can we separate the effect of flying on Hawaiian Airlines from that of flying to Hawaii? We need not only a dummy for each Hawaiian airport and for Hawaiian Airlines, but the interaction between the two, so we can see whether flights to Hawaii are earlier on Hawaiian Airlines than other airlines.

I hope to do a linear regression model for delay time and a logistic regression model for the 15-minute dummy. However, every time I’ve attempted to create the required number of dummy variables in a Pandas DataFrame, it’s crashed my computer. Figuring out how to handle these interaction variables has taken up the better part of my past two months. Eventually I settled on a strategy that cuts down the number of interactions to only a few hundred:

* Only use interactions between airlines and airports. These are the ones I am most concerned about, because as stated above, I don’t want my results to “punish” airlines for flying out of bad airports.
* Only take interactions for airports with at least 1.5% of the arrivals or departures in the data set. This is not ideal, as it would be better to find the effect of each airline at each airport, no matter how small. However, restricting to the larger airports is the tradeoff I will have to make between model robustness and computing power.

Besides pure linear regression and logistic regression, I also hope to experiment with a principal components analysis (PCA) linear regression model. This way, there would be fewer variables in the regression. However, this does not solve the computing power problem, as it still takes a lot of memory to transform the variables (including interactions) into principal components. I have been focused on resolving the computing power issues for a long time, and am now ready to get to work on generating model results.