Predicting Delay Times for US Domestic Flights

Eric Chang

November 2015

1 Introduction

US Domestic flights are often delayed, especially after 9/11 with TSA regulations. Passengers do not have a way of knowing whether their flight will be delayed. This project serves to analyze and predict whether a flight will be delayed in advance. The Bureau of Transportation statistics provides a data set on the flights within the continental US from 1987 to 2008.

2 Dataset

In this project I will only used flights from 2002-2008, since Pre 9/11 flight delays should not be as relevant to creating a predictive model on modern flight delays. I also only imported the flight records from one datasource in the BTS, since this was the most relevant for tracking flight delays. I began by importing this flight data into a SQLite database, but with over 40 million records, even simple queries were very slow. I truncated the data by only selecting 10% from flights from each year.

2.1 Data columns

The raw data from BTS contained the following columns:

Year	2002-2008
Month	1-12
DayofMonth	1-31
DayOfWeek	(Monday) - 7 (Sunday)
DepTime	actual departure time
CRSDepTime	scheduled departure time
ArrTime	actual arrival time
CRSArrTime	scheduled arrival time
UniqueCarrier	unique carrier code
FlightNum	flight number
TailNum	plane tail number
ActualElapsedTime	in minutes
CRSElapsedTime	in minutes
AirTime	in minutes
ArrDelay	arrival delay
DepDelay	departure delay
Origin	origin IATA code
Dest	destination IATA code
Distance	in miles
TaxiIn	taxi in time, in minutes
TaxiOut	taxi out time in minutes
Cancelled	was the flight cancelled?
CancellationCode	reason for cancellation
Diverted	1 = yes, 0 = no

2.2 Truncating Data

After examining the data, I noticed that there were a large number of invalid records, so I truncated some invalid data.

1. Negative elapsed flight time
2. Very long elapsed flight time (¿20 hours)
3. Missing elapsed flight time
4. Arrival times outside 00:00-23:59 range
5. Departure times outside 00:00-23:59 range

This truncated dataset turned out to contain 4639040 flights.

2.3 Statistics

Basic statistics on raw data:

Total Flights	4763460
Cancelled Flights	111432
Percent Cancelled	2.34%
Time Range	Jan 2002 - Dec 2008

Averages on truncated data (time in mins):

Actual Elapsed Time	132.674
Air Time	109.735
Arrival Delay	10.355
Departure Delay	11.515

Some interesting findings:

- Longest: EWR to HNL, 2007, 13h56m
- Shortest: ACV to CEC, 2003, 22m
- Farthest: EWR to HNL, 4962 miles
- Closest: LGA to JFK, 11 miles
- Most popular origin: ATL
- Least popular origin: OGD
- Most popular dest: ATL
- Least popular dest: PIR

3 Predictive Task

In order to create a predictor for predicting whether a flight will be delayed, I will be using a model with the features Year, Month, Day of Week, Departure Time. Creating a SVM out of these features will require that the features be categorized, so I will create a binary vector for each of these features. I will also add the average departure delay for the carrier as another feature. After creating a SVM model, I will have subsets of data for training, validation, and testing, each with size 50000. The training set will be for fitting the model, the validation set for optimizing the C parameter, and finally the testing set forproducing results. For the baseline, I will be sorting all the Carriers by their rates of delays, and predicting yes if the Carrier has a high rate of delays.

3.1 Preparing data

In addition to invalid records, there are also records that may not be relevant for creating a predictive model. I also removed the rows that fell into one of these categories:

1. Departed more than 2 hours early
2. Delayed more than 24 hours
3. Departure delayed until next day.

4 Model

I picked the model with features that seemed relevant. With each successive year, there could be more demand for flights, which may result in slower service by staff. I also took into account the day of the week, since weekends may be more popular days for booking flights, and employees

may not be as productive on Fridays. The different months will result in different weather conditions, which should definitely have an impact on the flight delays. Departure time may also be relevant since peak hours will often be more delayed than the less popular hours. I also tried using some other features such as rounding the carrier's average delay and origin airport. However, these did not seem to make any improvement over the existing model. Adding these features significantly increased the size of the feature vector(over 300 airports), while the results were more or less the same.

4.1 Predicting delays

This section covers the predictions of whether or not a flight will be delayed. The accuracy of this model can be measured using the Hamming Loss. Initially I defined a flight to be late if its departure time was over 15 minutes later than scheduled. However, the results of the predictor on the 15 minutes turned out to be rather trivial, so I created another predictor for predicting whether a flight would be at least 1 minute late.

4.1.1 Baseline - 15 minutes

For the baseline, I looked at the historical flights for the carrier. If more than 50% of the recorded flights departed after the expected departure time, predict True; else, predict False. This predictor had a Hamming Loss of 0.2388

4.1.2 Predictor - 15 minutes

To create a predictor, I looked over the available features and picked some that seemed viable. I used a SVM with the month, day of the week, departure time, and carrier average. Each month had 12 features indicating which month the flight was in, DayOfWeek had 7 features, and departure time had 24 features. This simple predictor yielded a hamming loss of 0.2126 on the same randomized validation set. It turns out that this predictor ended up just predicting False for every flight, since the SVM will predict 0 if the probability is less than 50% or 1 if greater than 50%. Since the actual probability of flights being 15 minutes late is rather low, the predictor came up with low probabilities for each prediction and picked 0 for all flights in the validation set.

4.1.3 Baseline - 1 minute

I also tried defining a flight to be considered late if its departure was at least 1 minute later than scheduled. For the baseline, I ranked the carriers by their delay rates, and picked the lower half to predict False, and the higher half to predict True. This baseline produced a delay of 0.4568

4.1.4 Predictor - 1 minute

The predictor for this used the same features as the 15 minute predictor, and had a hamming loss of 0.3848. The hamming loss turned out much higher than expected, but given that only around 20% of flights are more than 15 minutes late, trivial predictors could produce accurate results in the 15 minute case. In this predictor, I optimized the model by removing features until it nothing could be removed without making the model worse. I also optimized the model by using the validation set to pick the optimal C which turned out to be 100 [1]. However, when comparing with the baseline in the 1 minute delays, it is significant since now closer to half the flights are delayed, and trivial predictors would not perform as well in this case.

4.2 Predicting delay times

Instead of predicting yes or no to indicate whether a flight will be delayed, we can also try to predict how delayed a flight will be using regression.

4.2.1 Baseline

For a baseline, I predicted the Departure delay for a flight by predicting average of that airline for the same year. This baseline resulted in a MSE of 1114.99

4.2.2 Predictor

I first attempted to create a simple linear regression model:

$$DelayTime = \alpha + (averagecarrierdelay)\beta$$

which yielded a MSE of 1087.86. I also tried to use logistic regression with the same categorical variables as predicting delays. However, with linear regression, I had to categorize the resulting set, which led to inaccurate predictions. The MSE of the logistic regression model turned out to be 1145.95. This turned out to be unsuccessful, and adding more features to the linear regression model was difficult due to most of the features being categorical and did not change the results much.

5 Literature

This dataset originated from RITA in the Bureau of Transportation Statistics. The BTS was established to administer data collection, analysis, and reporting for different means of transportation [1]. This data that I used was originally to track flights and create reports on what percentage of flights arrived ontime. There have been various studies in the past that involve finding the cheapest way to buy air tickets, including, Skiplagged which was a website to find cheaper flights by taking into account the destination as a layover. [2] More relevant studies include those that try to predict flight whether a flight will be delayed. While researching some similar studies, I found that several groups in Stanford's CS229 course did a study on this as well, except they also took into account the weather at the relevant cities, as well as whether the previous flight had been delayed [3]. They also tried different methods of making predictions, such as Naive Bayes, Random Forests, and GLMs. This allowed them to produce more accurate results, since weather and previous flights are both key factors to determining whether a flight would be delayed. I tried to avoid using real time data such as these, since I wanted to try making predictions in advance, rather than the day of the flight. Other groups have also had high prediction rates due to most of the flights having no delays. I did not find any studies on predicting the actual delay times to compare my Predicting Delay Times with.

6 Results

My 15-minute model was able to outperform the baseline since most of the flights were not delayed. However, even though my 15 minute model was rather trivial, it performed better than one model which took into account historical weather during similar date and times. However, when comparing to models which used real-time features, my model was significantly worse. It seems that using features such as the month, year, and departure time are not nearly as im-

portant as realtime features such as weather and previous flight status. In my test using the 1-minute model, my model did outperform the baseline, which shows that my model was not trivial, but could still be improved with better feature selection. The strength of my model, however, is that it does not require real-time knowledge of the weather and previous flight to make predictions.

References

[1] https://www.rita.dot.gov/bts/about

[2] https://skiplagged.com

[3] http://cs229.stanford.edu/projects2012.html