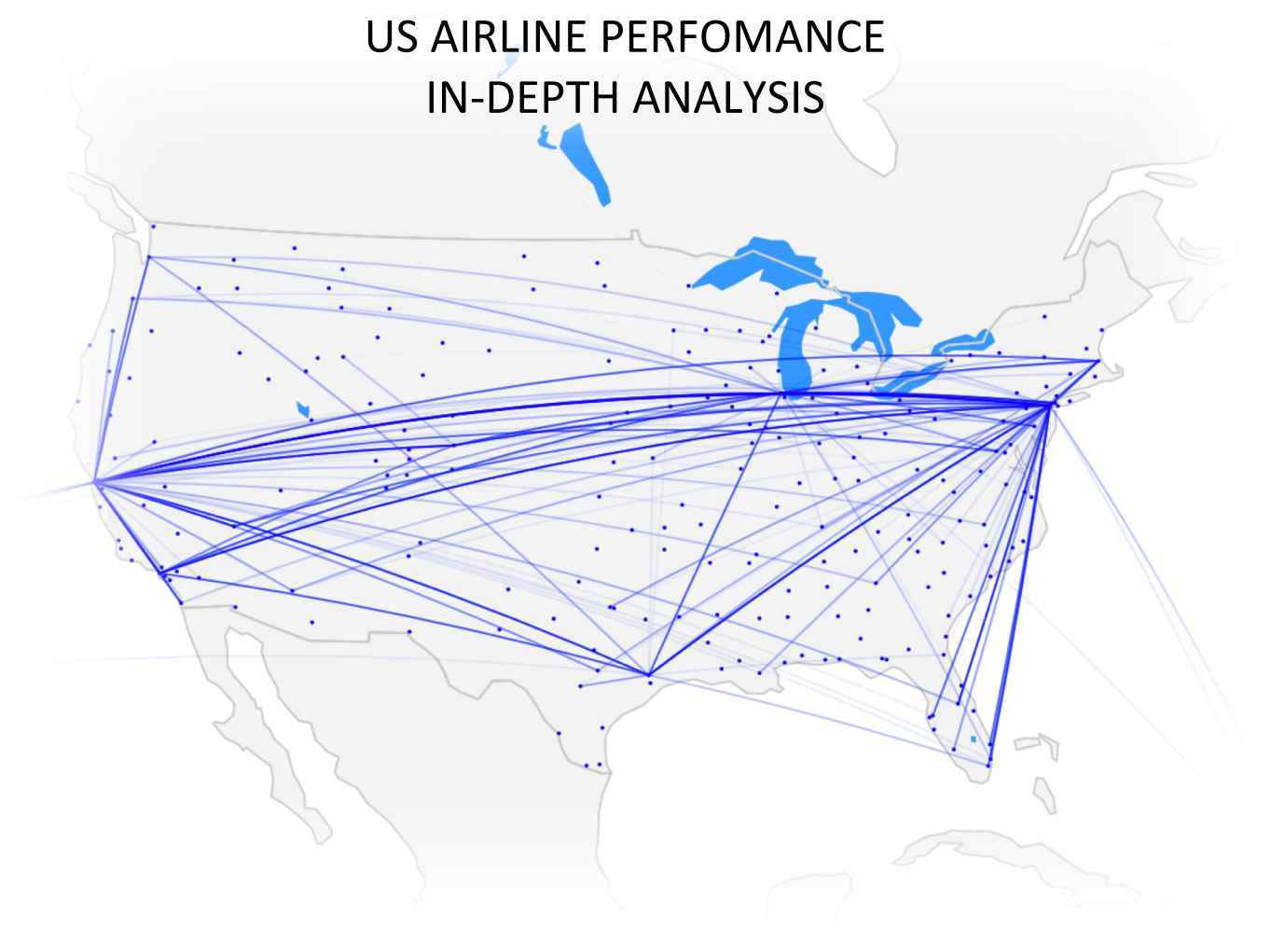
US AIRLINE PERFOMANCE IN-DEPTH ANALYSIS



2017 United Airlines Most Delayed Routes

Dmitriy Kats
May 2019

Table of Contents

1. Modeling

There are many classification algorithms available to predict whether a flight will be delayed or not. Below is a table with ones we'll focus in this analysis.

	Advantages	Disadvantages
Logistic Regression	Can incorporate future data into the model easily. Fast and good for probabilistic approaches.	Not as accurate as random forest. Almost no tuning capabilities.
Naïve Bayes	Super simple and fast.	Can't learn about interactions between features. Not as accurate as some other models.
Random Forest	Easy to implement, use, and tune. Very easy to interpret and explain as this is an ensemble of decision trees. Not a lot of parameters to tune. Very fast and proven to be accurate.	Although not necessary, some tuning is required to achieve optimal performance.
XGBoost	High model performance and execution speed.	A lot of parameters to tune.

Table 1 Modeling Techniques

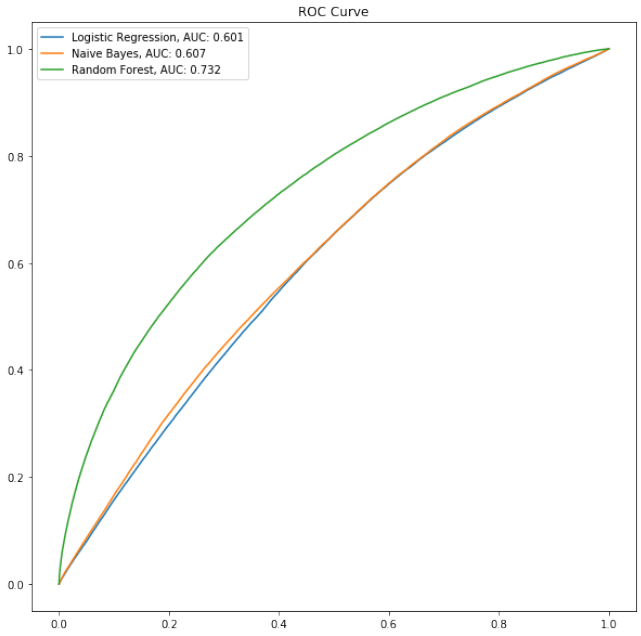
We started with the fastest and easiest implementation of Logistic Regression and Naïve Bayes classifiers. An 80/20 test/train split was utilized initially. We don't have a concern about not having enough training data as the there are over 2M samples. Additionally, since there is a class imbalance in the dataset, only ~18% of flights tend to be delayed, we want to make sure this is taken into account. The goal is to be able to predict as many delayed flights accurately as possible, but also without sacrificing on-time predictions.

We'll start with minimal data manipulation and no feature engineering to get a baseline. Encoding categorical values and limiting the data to top 20 airports and top 5 airlines by volume, we get our first results:

	Quarter	Month	DayofMonth	DayOfWeek	Reporting_Airline	Origin	Dest	ARR_HOUR	DEP_HOUR	STATUS
25	2	5	6	6	4	19	39	19	18	0
26	2	5	6	6	4	19	39	9	8	0
27	2	5	6	6	4	19	40	15	9	0
28	2	5	6	6	4	19	66	18	14	0
29	2	5	6	6	4	19	69	14	13	0

Table 2 DataFrame Used for Initial Model

Initial Results:



Logistic Regression		Actual		
		Positive	Negative	
Predicted	Positive	401,109	-	401,109
	Negative	90,673	-	90,673
		491,782	-	

Naïve Bayes		Actual		
		Positive	Negative	
Predicted	Positive	401,109	-	401,109
	Negative	90,673	-	90,673
		491,782	-	

Random Forest		Actual		
		Positive	Negative	
Predicted	Positive	399,344	1,765	401,109
	Negative	86,270	4,403	90,673
		485,614	6,168	

Figure 1 Initial ROC Curve and Confusion Matrices

ROC AUC does not look great and initial predictions don't look too promising. Both Logistic Regression and Naïve Bayes models did not predict any delayed flights, and Random Forest did relatively well, given the raw data. Let's focus on the Random Forest model and see if we can make improvements by engineering additional features, data balancing, hyper parameter tuning, and cross validation.

2. Feature Engineering and Data Balancing

2.1. Feature Engineering

In order to improve our model, we'll add extra features to our dataset.
Flight number / plane tail number were added back into the dataset. Note that this is not an engineered feature but provides historical significance to the model if certain flight numbers are historically delayed.

1. Weather data: we added 5-hour prior to departure precipitation sum, temperature at origin, 5-hour prior to departure average visibility, wind speed at origin.
2. Flight congestion: added a count of number of departing and arriving flights during the scheduled departure hour
3. Load factor: added monthly passenger loading factor for a given flight
4. Historic delay frequency: added monthly average delay for a given flight number

2.2. Imbalanced Data

Since our flight data has only ~20% of flights delayed, the model tends to favor on-time arrivals. This can be seen in the initial results where our Random Forest model only predicted ~4k delayed flights out of 90k. In order to deal with this bias, we'll introduce a weight factor to the model. We'll assign a weight of 3 to the delayed flights and a weight of 1 to the on-time flights, then we'll adjust the weight of delayed flights to 5 and see how our predictions improve.

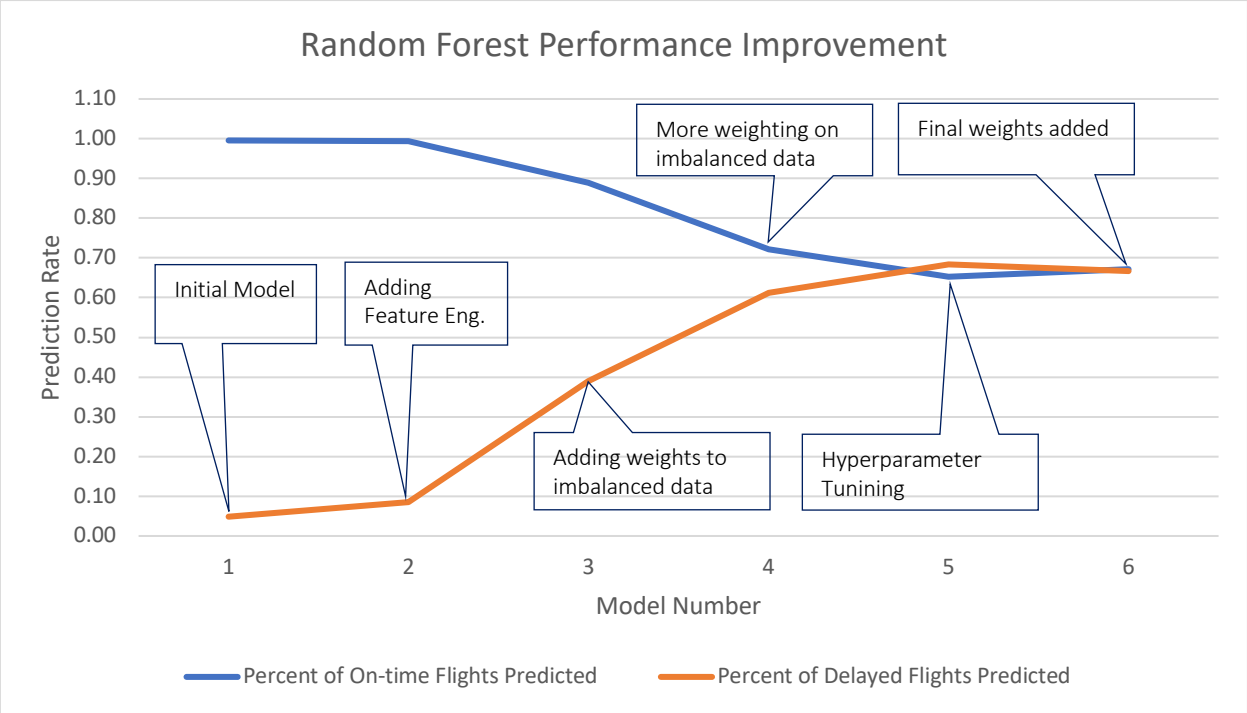


Figure 2 Random Forest Model Prediction Improvements

Engineered Features		Actual		
		Positive	Negative	
Predicted	Positive	398,512	2,597	401,109
	Negative	82,977	7,696	90,673
		481,489	10,293	

Adding Weights to Imbalanced Classes 1:3		Actual		
		Positive	Negative	
Predicted	Positive	356,400	44,709	401,109
	Negative	55,311	35,362	90,673
		411,711	80,071	

Additional Weights to Imbalanced Classes 1:4.5		Actual		
		Positive	Negative	
Predicted	Positive	289,448	111,661	401,109
	Negative	35,185	55,488	90,673
		324,633	167,149	

Tuned Model and Final Weights Added 1:4.85		Actual		
		Positive	Negative	
Predicted	Positive	269,557	131,552	401,109
	Negative	30,175	60,498	90,673
		299,732	192,050	

Figure 3 Confusion Matrices for Improved Random Forest Model

Interpreting the final results, we can calculate some key metrics:
Recall / Sensitivity: When the flight is actually on-time, how often does our model predict that it is on-time?
Recall = 269,557 / 401,109 = **0.67**

Precision: When the flight is predicted to be on-time, how often is the model predicting it to be on-time correctly?
Precision = 269,557 299,732 = 0.90

False Positive Rate: When the flight is actually delayed, how often does it predict that it's not?
False Positive Rate = 131,552 / 90,673 = 1.45

True Negative Rate: When the flight is actually delayed, how often is the model correctly predicting that the flight is delayed?
True Negative Rate = 60,498 / 90,673 = **0.67**

We were able to adjust and tune our model to predict ~67% of flights correctly whether it's delayed or on-time. This was the goal of the project, to maximize prediction rate for both types of flights.

3. Conclusions and Next Steps

In this analysis we aimed to shine a light on flight delays and predict weather a future flight would be delayed or not. The flight data were acquired from US Department of Transportation. We focused only on 2017 data for this analysis.

There are three major contributors to delayed flights, as we found in the EDA, Figure 14: NAS (mostly weather), Carrier delay (e.g. mechanical delay, flight crew, etc.), and Late Aircraft.
We had data on both, departure delays and arrival delays and we chose to focus on the latter, as this is the most important factor for travelers.

In order to improve our chances of prediction, we added a number of other datasets to the flight data. Passenger Load Factor data was added to account for delays due to more flyers. Weather data was included to account for weather delays. And additional features were engineered to account for airport traffic.
Since we wanted to predict whether a flight was going to be delayed or not, a classification algorithm was chosen. A number of methods were evaluated, and Random Forest was ultimately chosen for the reasons described in the Modeling Section.

Although the model performed relatively well, there a number of improvements that can be made. Adding passenger load factor data on an hourly basis, the intuition is that the predicted delayed flights will increase. The other key contribution to delays is Carrier performance, which can be explored via airline data on maintenance records and even age of aircraft based on the tail number and FAA data. Our model was only trained on 2017 data and can be further improved by introducing more training data. And finally, Late Aircraft delays can be traced using the tail number data and origin / destination information to better predict overall delays.