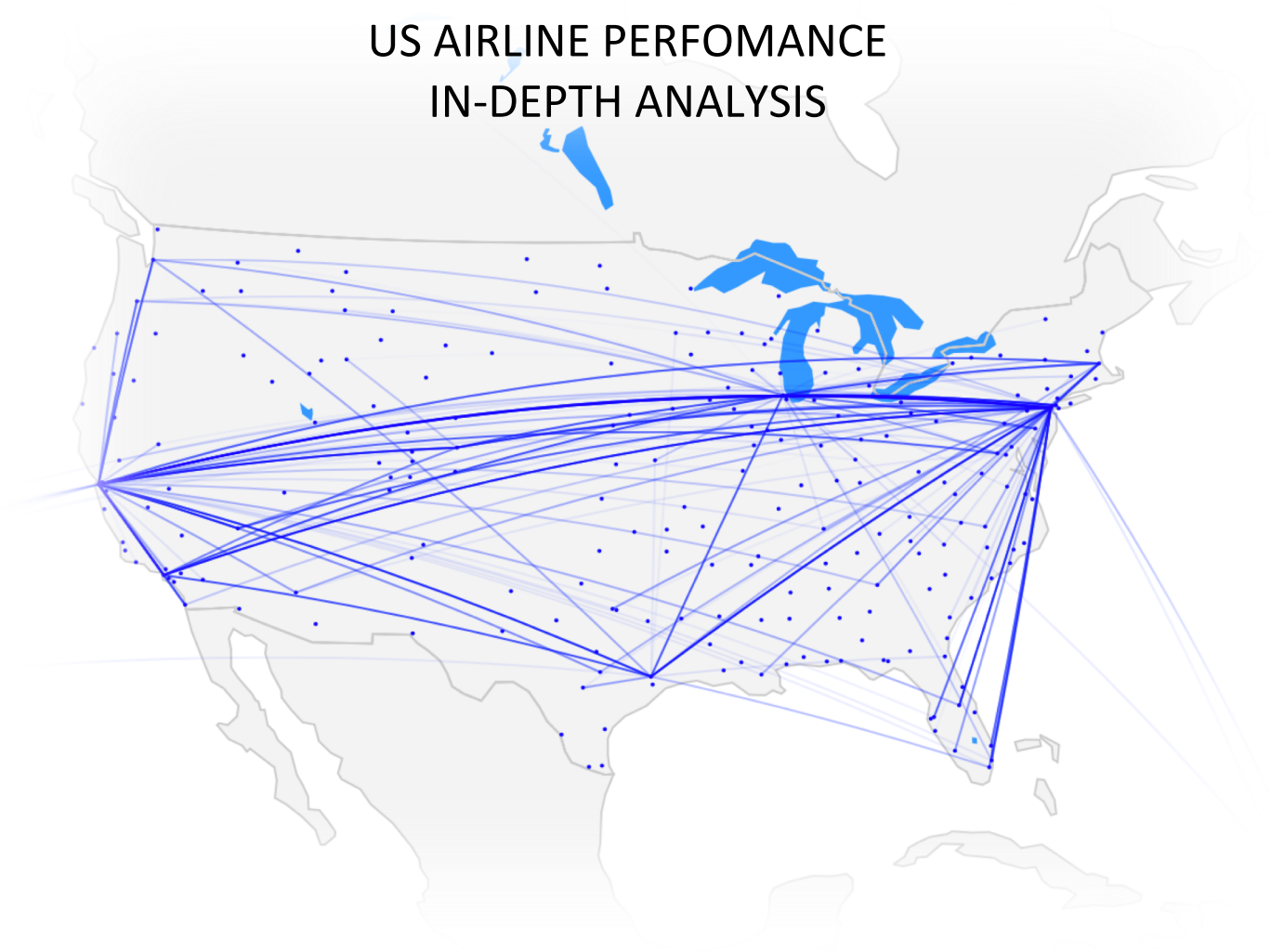
US AIRLINE PERFOMANCE
IN-DEPTH ANALYSIS



2017 United Airlines Most Delayed Routes

Dmitriy Kats
May 2019

Table of Contents

1. Datasets

Loading Data
https://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=311

Complete Datasets
https://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=236

Weather:
https://mesonet.agron.iastate.edu/ASOS/

Airline Info (automatic download)
https://www.transtats.bts.gov/Download_Lookup.asp?Lookup=L_UNIQUE_CARRIERS

Airport Coordinates:
https://drive.google.com/file/d/1bMVXqd8Tm30RwmYLBgKk8RMr786IuDoK/view?usp=sharing

FAA OPSNET:
https://aspm.faa.gov/opsnet/sys/Delays.asp

2. Modeling

Our first approach is going to be a classification model to predict whether a flight will be delayed or not. There a number of algorithms available for use, we will focus on the following:

	Advantages	Disadvantages
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.	
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.
XGBoost	High model performance and execution speed.	A lot of parameters to tune.

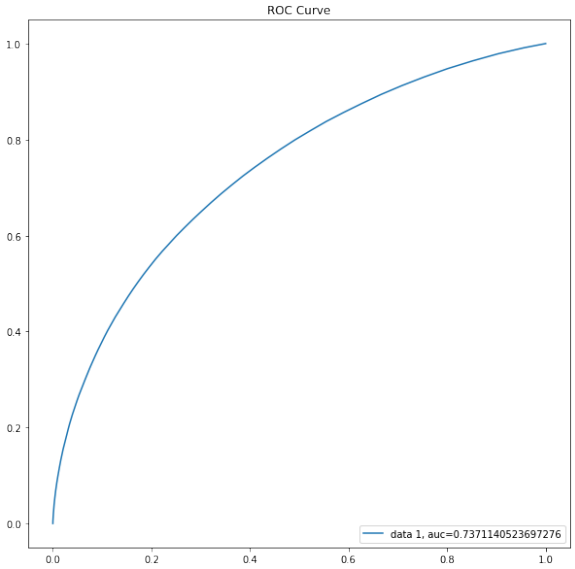
2.1. Classification

We'll initially approach this problem using RandomForestClassifier with an 80/20 test/train split to predict if flight is delayed or not, along with the probability. Random Forest was chosen for its relatively fast training algorithm and accuracy. It is also relatively simple to tune and use, and since there are a large number of trees, overfitting tends to be less of an issue.
We start our run with minimal data, only including the features that will be available at the time of purchasing the ticket: date, carrier, origin, destination, and time. Original run is on the entire 2017 dataset, with no feature engineering, only binning departure and arrival hour blocks:

	MONTH	DAY_OF_MONTH	DAY_OF_WEEK	OP_CARRIER	ORIGIN	DEST	STATUS	ARR_HOUR	DEP_HOUR
0	3	21	2	2	39	81	0	11	10
1	3	21	2	2	145	39	0	16	14
2	3	21	2	2	39	146	0	13	12
3	3	21	2	2	39	236	0	9	8
4	3	21	2	2	110	26	0	13	10

Table 1 DataFrame Used for Initial Model

Using n_estimators=25, Initial results were favorable, as the AUC is significantly higher than chance, at 0.737 with and accuracy of 82.5%. This is not bad for first run but leaves room for improvement.

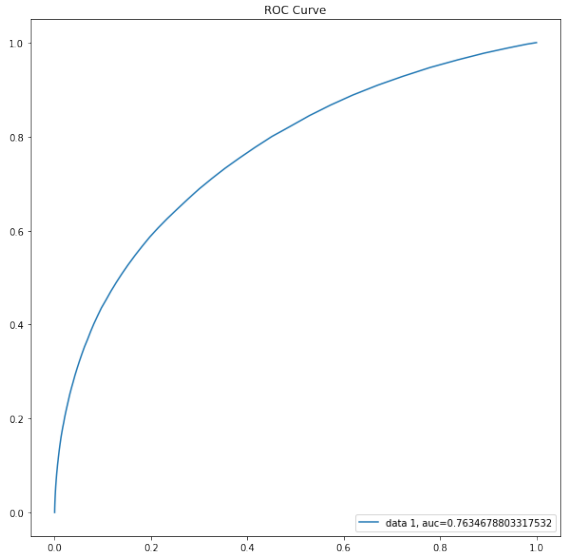


		Actual		
		Positive	Negative	
Predicted	Positive	874,265	36,085	910,350
	Negative	158,869	46,663	205,532
		1,033,134	82,748	

Figure 1 ROC Curve and Confusion Matrix for Initial Run

Interpreting the confusion matrix, we can see that our predicted delays were much lower than actual, 46k vs 205k, respectively. This may not be a bad thing as this is a more conservative approach to predicting. However, we also didn't predict all on-time flights accurately, only 874k out of 1.02M were predicted accurately to be on-time.

In order to improve our model, we'll add extra features to our dataset and we'll use random forest classifier to engineer features and evaluate them prior to using on other classifiers.
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.
We'll also add direction of flight, as flights heading West are typically susceptible to the jet stream and may be subject to more delays. Then we limited the dataset to only include five major airlines and 20 major airports, by volume. And the date features were OneHotEncoded in order to reduce dimensional importance.
Using n_estimators=100, results improved to AUC of 0.763 and Accuracy of 85.8%.

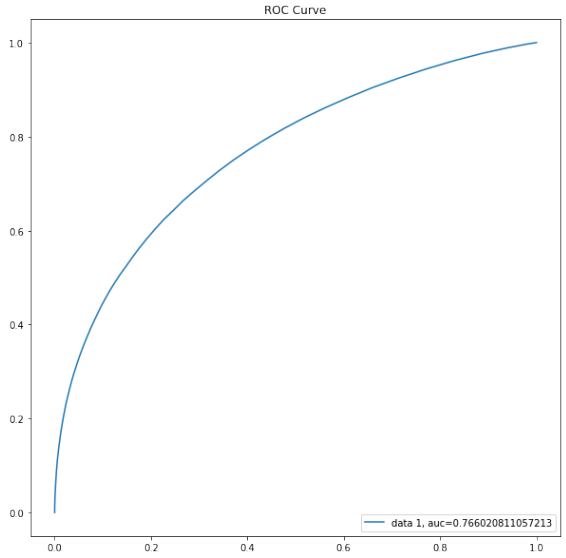


		Actual		
		Positive	Negative	
Predicted	Positive	409,244	5,010	414,254
	Negative	64,382	11,527	75,909
		473,626	16,537	

Again, we see that we under-predicted delayed flights by approximately 65 thousand flights. Note, the totals are different due to reduction in modeled data.

Figure 2 ROC Curve and Confusion Matrix - Run 2

Next, we added precipitation data as a cumulative sum for the past 4 hours prior to scheduled departure. This was done to improve the model due to weather delays. This was not expected to be a large contributor, as weather delays are relatively rare. We also added a congestion component by adding a feature containing total number of scheduled flights for any given hour at a particular airport.
Using n_estimators=100, results improved to AUC of 0.766 and Accuracy of 86.1%.

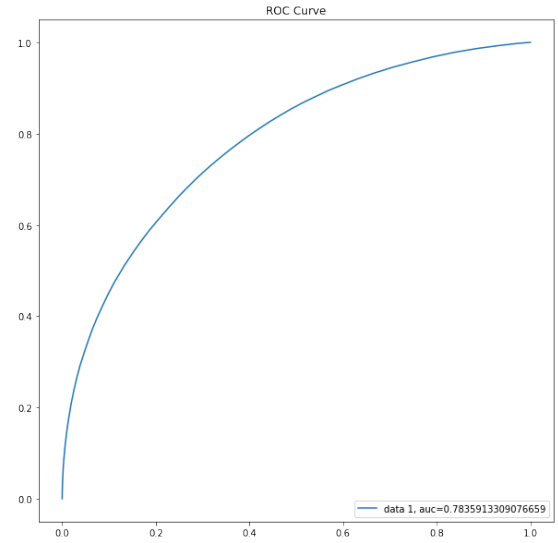


		Actual		
		Positive	Negative	
Predicted	Positive	408,622	5,632	414,254
	Negative	62,604	13,305	75,909
		471,226	18,937	

Again, we see that we under-predicted delayed flights but by a smaller amount. Approximately 62 thousand flights were misclassified as delayed.

Figure 3 ROC Curve and Confusion Matrix - Run 3

Next step was to add historic delay data based on routes and aggregate it on a monthly basis. We know from our EDA that certain airports are prone to delays and we can take advantage of this by creating a feature that takes the combination of origin and destination into consideration. We also added feature scaling and increased the n_estimators to 150. We saw increases in accuracy to 86.1% and AUC to 0.784.



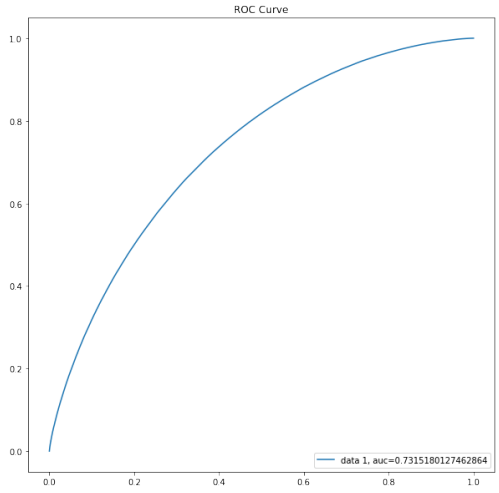
		Actual		
		Positive	Negative	
Predicted	Positive	406,653	7,601	414,254
	Negative	60,396	15,513	75,909
		467,049	23,114	

Again, we see that we over-predicted delayed flights but by a smaller amount. Approximately 60 thousand flights were misclassified as delayed.

Interpreting the 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 = 406,653 / 414,254 = 0.982
Precision: When the flight is predicted to be on-time, how often is the model predicting it to be on-time correctly?
Precision = 406,653 / 466,049 = 0.871
False Positive Rate: When the flight is actually delayed, how often does it predict that it's not?
False Positive Rate = 7,601 / 75,909 = 0.100
True Negative Rate: When the flight is actually delayed, how often is the model correctly predicting that the flight is delayed?
True Negative Rate = 15,513 / 75,909 = 0.204
Our model does not predict delayed flights very accurately and due to the fact that the data is imbalanced, our overall performance doesn't look too bad. However, our ultimate goal is to predict delayed flights more accurately.

These are decent results, but we need to see how the model performs on current/future data. For that we'll train the model on all of 2017 data and test it on 2018 data:

2.1.1. Random Forest Classifier – 2018 Test

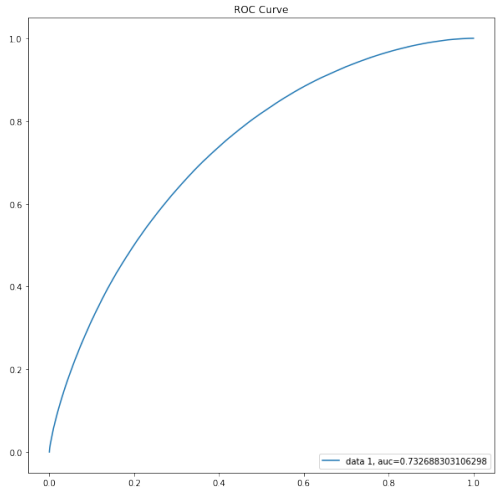


		Actual		
		Positive	Negative	
Predicted	Positive	1,791,259	195,370	1,986,629
	Negative	324,799	148,998	473,797
		2,116,058	344,368	

Sensitivity / Recall	0.902
Precision	0.847
Accuracy	0.789
Misclassification Rate	0.211
False Positive Rate	0.412
True Negative rate	0.314

Actual on-time	1,986,629
Actual delayed	473,797
Correctly predicted on-time	1,791,259
Correctly predicted delayed	148,998

2.1.2. XGBoost Classifier – 2018 Test

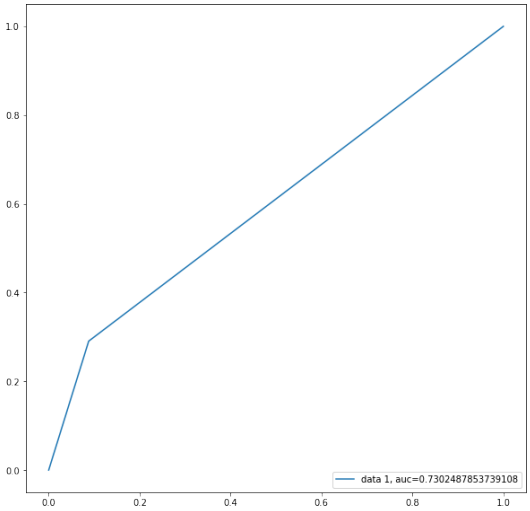


		Actual		
		Positive	Negative	
Predicted	Positive	1,950,793	35,836	1,986,629
	Negative	430,729	43,068	473,797
		2,381,522	78,904	

Sensitivity / Recall	0.982
Precision	0.819
Accuracy	0.810
Misclassification Rate	0.190
False Positive Rate	0.076
True Negative rate	0.091

Actual on-time	1,986,629
Actual delayed	473,797
Correctly predicted on-time	1,950,793
Correctly predicted delayed	43,068

2.1.3. Logistic Regression – 2018 Test

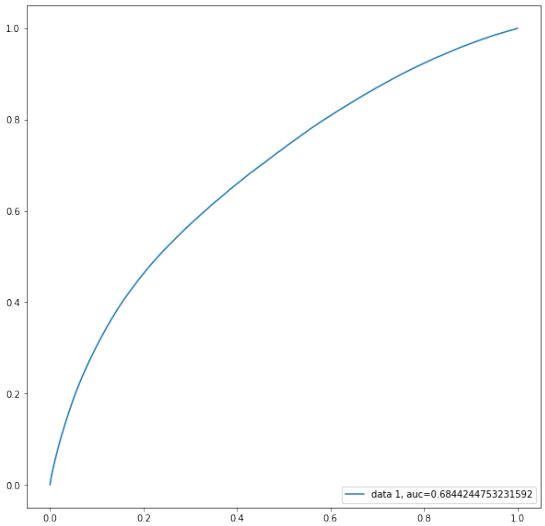


		Actual		
		Positive	Negative	
Predicted	Positive	1,833,460	153,169	1,986,629
	Negative	347,840	125,957	473,797
		2,181,300	279,126	

Sensitivity / Recall	0.923
Precision	0.841
Accuracy	0.796
Misclassification Rate	0.204
False Positive Rate	0.323
True Negative rate	0.266

Actual on-time	1,986,629
Actual delayed	473,797
Correctly predicted on-time	1,833,460
Correctly predicted delayed	125,957

2.1.4. Naïve Bayes – 2018 Test



		Actual		
		Positive	Negative	
Predicted	Positive	1,890,668	95,961	1,986,629
	Negative	387,060	86,737	473,797
		2,277,728	182,698	

Sensitivity / Recall	0.952
Precision	0.830
Accuracy	0.804
Misclassification Rate	0.196
False Positive Rate	0.201
True Negative rate	0.183

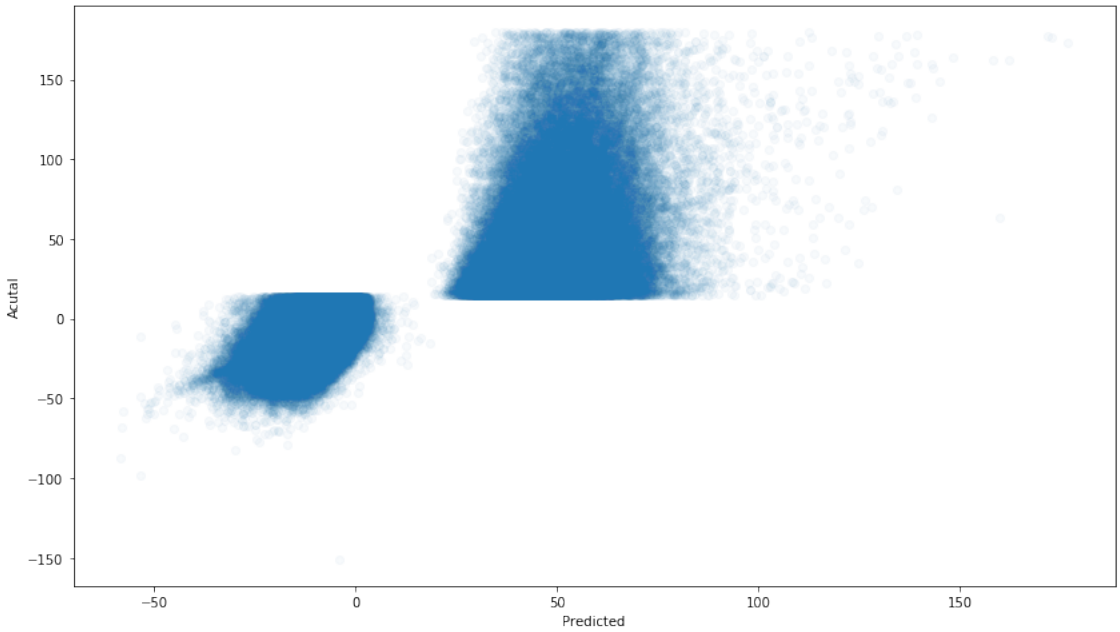
Actual on-time	1,986,629
Actual delayed	473,797
Correctly predicted on-time	1,890,668
Correctly predicted delayed	86,737

2.2. Regression

Regression models can be used to predict the actual time of delay in minutes.

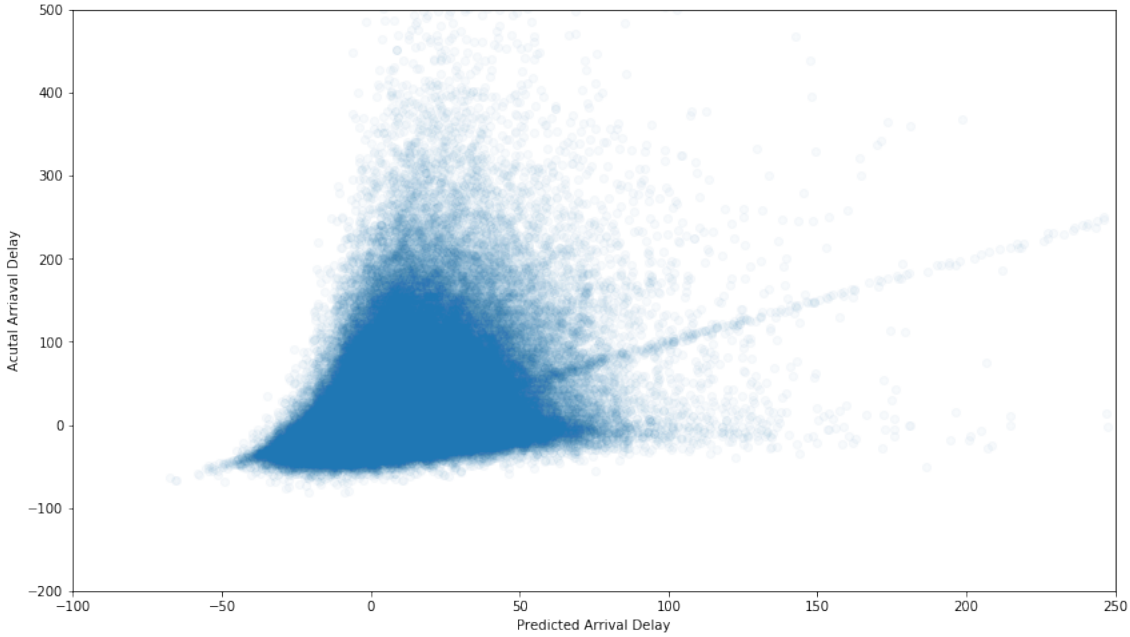
2.2.1. XGBoost Regressor

Root Mean Squared Error: 17.25



2.2.2. Linear Regression

R^2: 0.142
Root Mean Squared Error: 39.49



3. Conclusions and Next Steps

There are three major contributors to delayed flights, as we found in the EDA, Figure 14: NAS (mostly weather), Carrier, and Late Aircraft.
We found a lot of correlations in our data with delayed flights, including certain airports and flight routes, load factor, and weather. However, due to limitations in the load factor data, we were not able to capitalize on the finding. We were only able to obtain monthly averages and our analysis boiled down to hourly variations in flight delays. If we can add load factor data on an hourly basis, the intuition is that the predicted delayed flights will increase. Additionally, weather data only included precipitation information and not wind or other events. This can be further improved with better datasets. 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. And finally, Late Aircraft delays can be traced using the tail number data and origin / destination information to better predict overall delays. Due to the time limitations and the scope of this project, some of these options were not explored at this time.