Flight Delay Prediction

Muhammad Muneeb Afzal (mma525)

Introduction

Before going into data analysis, firstly the given topic was researched thoroughly through social science research papers which discussed different reasons that lead to flight delays (Sternberg, 2017) (Schaefer, 2001). Given this background, it was easier to know whether a feature in the data is relevant for our prediction or not. Furthermore, the distribution of each of the attributes was carefully analyzed before training.

Results and Discussion

Feature Selection, Encoding and Creation

It must be noted that before selecting the final features, many combinations of features were tried and tested before the best combination was selected.

The removed or irrelevant features are discussed as follows. Firstly, UID was deleted as it serves no purpose. FL_NUM was deleted as it is a unique number and independent for flights. Therefore, it won't affect the flight delay. ORIGIN_CITY_MARKET_ID, DEST_CITY_MARKET_ID, DEST, ORIGIN, DEST, DEST_CITY_NAME and DEST_CITY_NAME, were deleted because for these features there were around 250 unique values (hot encoding would lead to too many features). It is unlikely that such specific information will help us in generalization.

We included the ORIGIN_STATE_ABR and DEST_STATE_ABR because they are much broader attributes than cities and might help us in generalization (50 attributes were hot encoded for each). Also, research papers indicated that the origin and destination place will affect the delay time (each location has different populations and facilities at the airport) (Sternberg, 2017). However, these two attributes were discarded after using a validation set. It must be noted that ORIGIN and ORIGIN_CITY_NAME are redundant as they can be inferred from each other. Similarly, ORGIN_CITY_NAME and ORIGIN_STATE_ABR are also redundant. The feature CRS_DEPARTURE_TIME was tested as we deemed it relevant since there might be more people say in the airport in the evening (hours were hot encoded). However, it was discarded after using the validation set.

The selected attributes are discussed as follows. DAY_OF_WEEK was hot encoded because we might have more or less flights on weekdays (more flights on weekends). From FL_DATE, only the MONTH feature was created because we conjectured that different months and seasons could lead to different flight delays (people might be traveling more in some months than the other). Or people may be traveling more in the summer breaks. More people mean that there will more congestion and security checks leading to flight delays. The month feature was then hot encoded.

TAXI_OUT, TAXI_IN AND ACTUAL_ELAPSED_TIME were included as there are clearly relevant. Note that DISTANCE and ACTUAL_ELAPSED_TIME are highly correlated since long travel distance implies more elapsed time. Therefore, only ACTUAL_ELAPSED_TIME was used. Note that we follow the same procedure in the test dataset also so that the columns are identical in both the training and test datasets.

This selection and creation meant that we had 46 attributes at the end. While performing the algorithms, PCA was used to reduce the number of attributed to lower values such as 10 or 15. However, the final selected model gave better results without the application of PCA.

Outlier Removal

The distribution of the target variable ARR_DELAY was analyzed. It had a mean of 4.13 with a standard deviation of 45.38. Thus, we removed the few examples with more than 150min delay. The new number of training example was 4834 compared to the original 4911.

Algorithms

Previously, many methods such as kNN, SVM, fuzzy logics and random forest have been used to predict flight delay (Gopalakrishnan, 2017) (Robello, 2014) (Lu, 2008). We decided to use Neural Nets and Regression for this problem.

Neural Nets

Using Pytorch, a fully connected neural net was implemented. Different combinations of the number of hidden layers, number of hidden units, learning rates and different optimizers such as SGD and Adams were used. Note that number of features used in the results below is 46. Some combinations are shown below:

Combination	**MSE**
5 Hidden Units	1172
10 Hidden Units	1171
12 Hidden Units	1171
15 Hidden Units	1174

After trying many different techniques such PCA and varying the tunable parameters, we were not able to achieve satisfactory results. Therefore, we decided to try Regression.

Regression

Since we were not able to achieve impressive results using neural networks, we focused on Regression. The best model at the end was created using Lasso regularization.

PCA Dimensionality reduction

Note that for all the iterations and combinations tried, we first decreased the dimensions using PCA. However, the results indicated that it did not help in decreasing the MSE (Mean Square Error). Firstly, Multivariable Linear Regression was performed which yielded an MSE of 1039.

In order to check whether polynomial regression would perform better, we also tried polynomial degree 2 and 3. The results are shown in the table.

Table 1 Shows a comparison in MSE between Multilinear and Polynomials of degree 2 and 3.

Model	Training MSE
Multi-linear	1039.04
Polynomial (degree 2)	958.64
Polynomial (degree 3)	630.82

The table above shows that MSE decreases as we increase the degree. It seems that using a higher degree is doing better. However, to check for overfitting, we used 10-fold cross validation. The cross-validation results show that for some of the folds, the test error was extremely high (exponential) indicating overfitting. In fact, the cross validation of multi linear regression also showed extremely high values for some fold alluding towards overfitting. The cross-validation results for Multilinear are in the table below.

Table 2 Shows the MSE) for MultiLinear and MultiLinear Lasso Regularized for each of the folds (error of the test fold)

					Test MSE					
Model	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 1
Multi-linear	$4.99*10^{20}$	348.39	479.62	564.65	$2.0*10^{20}$	561.86	441.07	364.47	332.87	4.82*
Mult-linear (Lasso Regularization)	581.1	673.8	645.9	656.9	702.5	722.0	667.0	597.1	590.9	685.1

Thus, we decided to use regularization to overcome this problem. Ridge Regularization and Lasso Regularization were implemented. Indeed, regularization solved the problem of overfitting as shown in the table below:

Table 3 Shows 10-fold Mean test MSE for Multi Linear (without regularization), Ridge Regularization) and Lasso Regularization

Model	10-fold Mean Test MSE
Multi-linear	$1.08*10^{19}$
Mult-linear (Ridge Regularization)	603.2
Mult-linear (Lasso Regularization)	652.2

Different values of alpha (penalty term for regularization) for Ridge and Lasso Regularization were tried. Although Ridge Regression gave us lower MSE, Multi-linear Lasso Regularization was chosen as the final model because through Lasso Regularization, we get the ability to reduce unnecessary attributes. This is illustrated by the Lasso Path plot given below:

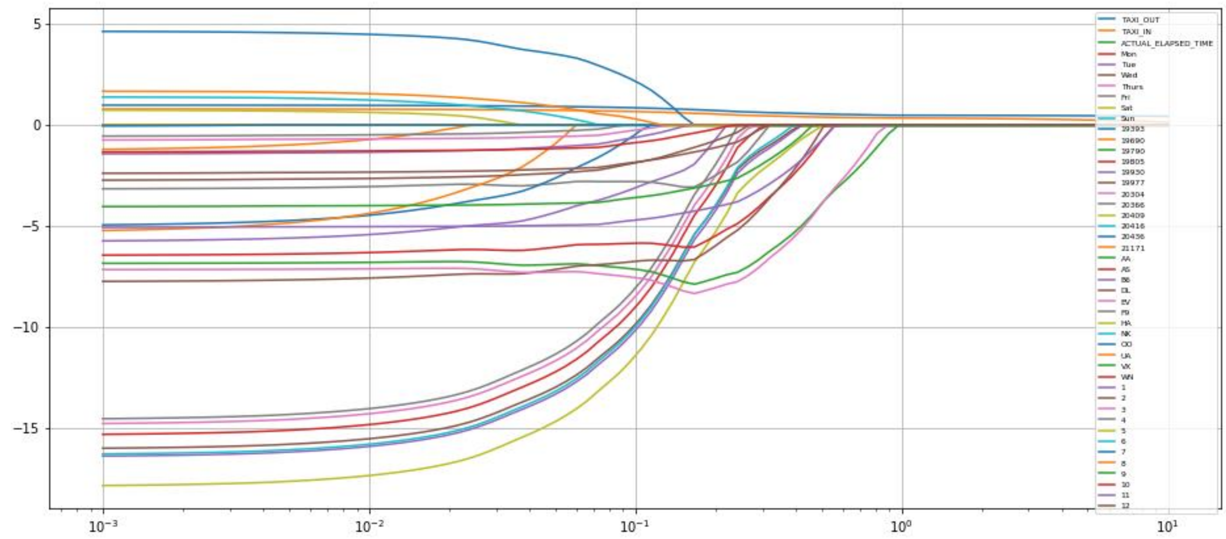

Figure 1 Shows the Lasso Path for the Multi Linear (Lasso Regularization)

The graph above shows that at alpha =1, we can weed out unimportant attributes. Different colored legend shows different attributes. Thus, for our final model, we used Multi Linear Regression with Lasso Regularization.

Note: The code for both implementations (neural network and regression) can be in the accompanying jupyter notebooks.

References

1. Sternberg, Alice, et al. "A Review on Flight Delay Prediction." *arXiv preprint arXiv:1703.06118* (2017).
2. Gopalakrishnan, Karthik, and Hamsa Balakrishnan. "A Comparative Analysis of Models for Predicting Delays in Air Traffic Networks." *USA/Europe Air Traffic Management Seminar*. 2017.
3. Rebollo, Juan Jose, and Hamsa Balakrishnan. "Characterization and prediction of air traffic delays." *Transportation research part C: Emerging technologies* 44 (2014): 231-241.
4. Schaefer, Lisa, and David Millner. "Flight delay propagation analysis with the detailed policy assessment tool." *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*. Vol. 2. IEEE, 2001.
5. Zonglei, Lu, Wang Jiandong, and Zheng Guansheng. "A new method to alarm large scale of flights delay based on machine learning." *Knowledge Acquisition and Modeling, 2008. KAM'08. International Symposium on*. IEEE, 2008