Machine Learning Engineer Nanodegree

Capstone Project

Gabriel Augusto

March 7th, 2019

I. Definition

Project Overview

The group of Journalism Globo made a study of flights delay from more than 2.8 million flights in Brazil revealing that approximately 20% of the entire scheduled flights had a delay [1]. Airline delays cost usually billions of dollars per year having many causes for example extreme weather, late-arriving, security.

Weather is one of the main cause of airline delays having for about 40% of total delay minutes [2]. Many company, like Google [3], have interests on studying flights delays caused by problematic weather condition, helping airlines companies to schedule solutions for delay or informing flight status for passengers.   .

In this project i analyzed 7 airport flights in Brazil from 2016 to 2017 :

SBCT,Afonso Pena

SBGL,Aeroporto Internacional Do Rio De Janeiro/Galeao

SBGR,Guarulhos - Governador Andre Franco Montoro

SBMN,Eduardo Gomes

SBPA,Salgado Filho

SBSP,Congonhas

SBCF,Tancredo Neves

The idea was to create a model that predict if a flight will be canceled or will have a delay using METeorological Aerodrome Report (METAR) information to train the system and test with other METAR of forecast predictions.

Problem Statement

The main goal in this project is to create a model that is able to predict if a flight is going to have a delay or if it would be canceled following the steps below:

1. Get METAR data combined with flight data from the region that is being study.
2. Keep only data that contains flights concluded with success or that had been canceled or delayed because bad meteorological conditions.
3. AtrasoVoo data for a best model creation.
4. Execute multiple machine learning algorithms treatening imbalanced data (the number of flights with no problem is much higher than cancelled or delayed flights)
5. Check results for the executed machine learning algorithms

The expected results are a high hit rate for testing data treatening false positive and true negative as well.

Metrics

Fbeta Score (explicar a escolha do Fbeta score)

The metric chosen was FBeta Score, to define it is necessary to know the concepts of Precision and Recall that follows the equations below:

$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$ and $Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$

Using Precision and recall we obtain the Fbeta Score, multiplyed for $(1 + \beta)$, which $\beta\ ,\ 0 \leq \beta \leq 1$:

$F\beta = (1 + \beta) \star \frac{precision * recall}{precision + recall}$

For β in the range of 0.0 to 0.5 gives weight to Recall

For β in the range of 0.5 to 1.0 gives weight to Precision.

And create a confusion matrix to help understanding the results.

II. Analysis

(approx. 2-4 pages)

Data Exploration

The dataset of this project were created merging two different datasets.

The first one was obtained from Iowa Environmental Mesonet containing METAR information of brazilian airports in 2016 [4]. The second dataset was obtained from a Kaggle competition (BrFlights2) containing flights tracked by the National Civil Aviation Agency (ANAC) [5], in Brazil, from January 2015 to August 2017.

The datasets were combined by the rounded hour and the respective airport having at the end all information from Metar and information from ANAC about the status of the flight (if a flight was delayed or cancel and it's departure and arrival time).

The first one contained these columns:

Airport and Hour that were used to merge with the second dataset.

tmpf: Air Temperature in Fahrenheit, typically @ 2 meters

dwpf: Dew Point Temperature in Fahrenheit, typically @ 2 meters

relh: Relative Humidity in %

drct: Wind Direction in degrees from north

sknt: Wind Speed in knots

alti: Pressure altimeter in inches

mslp: Sea Level Pressure in millibar

vsby: Visibility in miles

gust: Wind Gust in knots

skyc1, skyc2, skyc3, skyc4 -> sky coverage, related with visibility skyl1, skyl2, skyl3, skyl4 -> sky level, related with visibility

The second dataset contained four columns:

Airport, Hour used to combine with the first dataset.

Codigo.Justificativa contained the name of flight status

Situacao.Voo saying if the flight was Ok or canceled

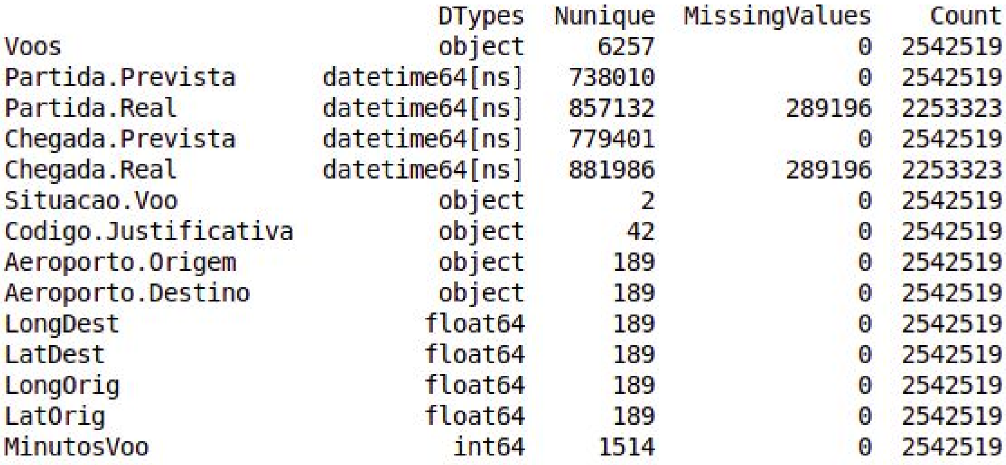
Estimative of time of departure (Partida.Prevista) and arrival (Chegada.Prevista) and the real departure (Partida.Real) time and arrival (Chegada.Real), and with them was created the column AtrasoVoo with the delayed minutes of a flight if the flight got early than expected were considered 0 and if the flight was canceled it was considered 100000 (to treat the missing values for arrival time when the flight is canceled)

In many columns as tmpf… there is the letter M that means not relevant time so it’s set to 0.

In skyc1, skyc2, skyc3, skyc4 the categories are changed for numbers (0-8) as follows

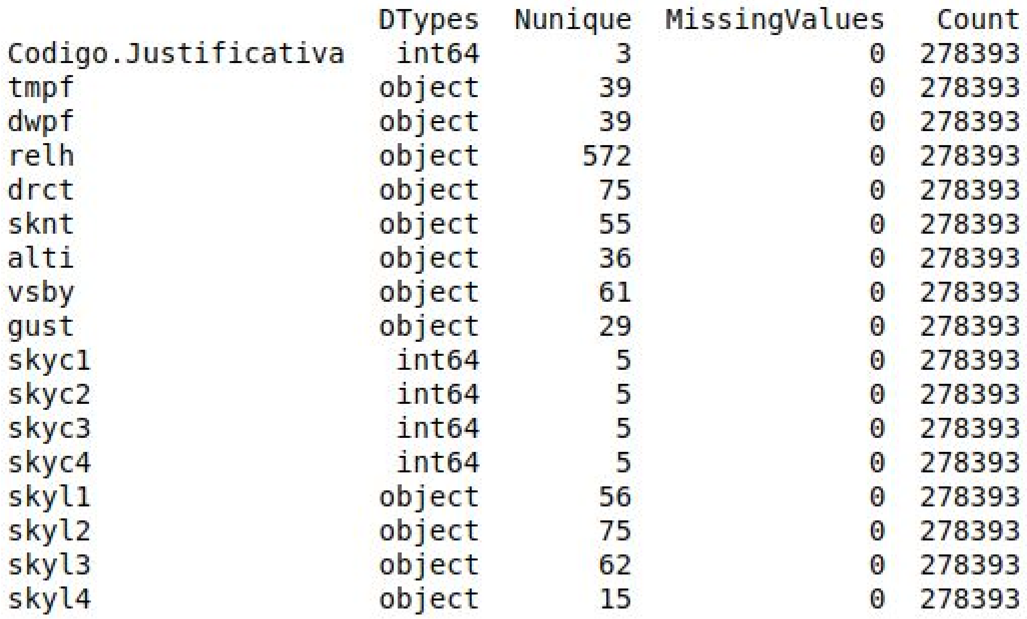
M = 0, FEW = 2, SCT = 4 , BKN = 6, OVC: 8, VV= : 8, /// = 8, NSC=0, NCD= 2 (according to https://www.skybrary.aero/index.php/Meteorological_Terminal_Air_Report_(METAR))

and skyl1, skyl2, skyl3, skyl4 has letter M as well, but in this case means 10000.



```
                              DTypes  Nunique  MissingValues    Count
Voos                          object     6257              0  2542519
Partida.Prevista      datetime64[ns]   738010              0  2542519
Partida.Real          datetime64[ns]   857132         289196  2253323
Chegada.Prevista      datetime64[ns]   779401              0  2542519
Chegada.Real          datetime64[ns]   881986         289196  2253323
Situacao.Voo                  object        2              0  2542519
Codigo.Justificativa          object       42              0  2542519
Aeroporto.Origem              object      189              0  2542519
Aeroporto.Destino             object      189              0  2542519
LongDest                     float64      189              0  2542519
LatDest                      float64      189              0  2542519
LongOrig                     float64      189              0  2542519
LatOrig                      float64      189              0  2542519
MinutosVoo                     int64     1514              0  2542519
```

The dataset in the end of merging contains the columns below:



```
                       DTypes  Nunique  MissingValues   Count
Codigo.Justificativa    int64        3              0  278393
tmpf                   object       39              0  278393
dwpf                   object       39              0  278393
relh                   object      572              0  278393
drct                   object       75              0  278393
sknt                   object       55              0  278393
alti                   object       36              0  278393
vsby                   object       61              0  278393
gust                   object       29              0  278393
skyc1                   int64        5              0  278393
skyc2                   int64        5              0  278393
skyc3                   int64        5              0  278393
skyc4                   int64        5              0  278393
skyl1                  object       56              0  278393
skyl2                  object       75              0  278393
skyl3                  object       62              0  278393
skyl4                  object       15              0  278393
```

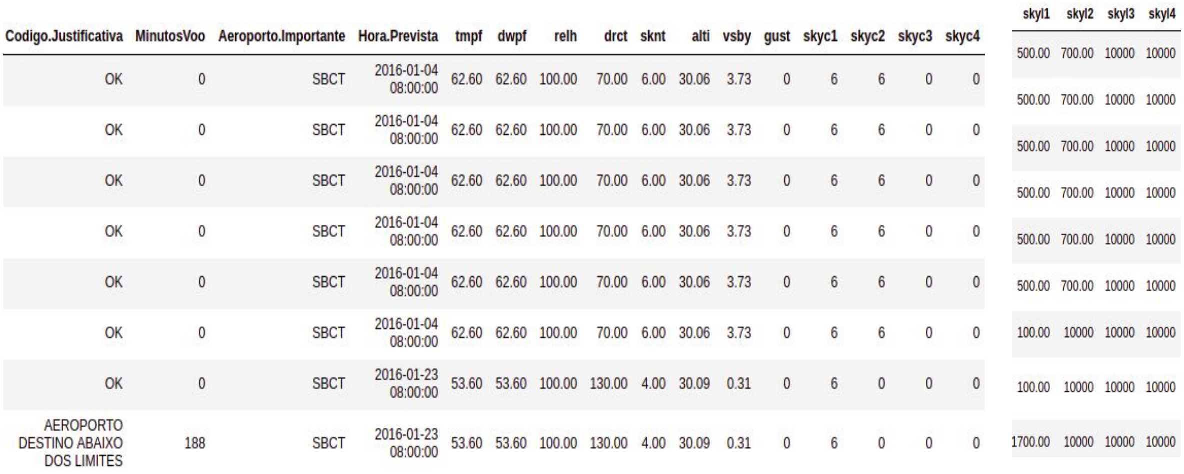
DTypes: shows the type of variable.

Nunique: How many different values the column of the dataset has.

MissingValues: As the name says if there are any null values.

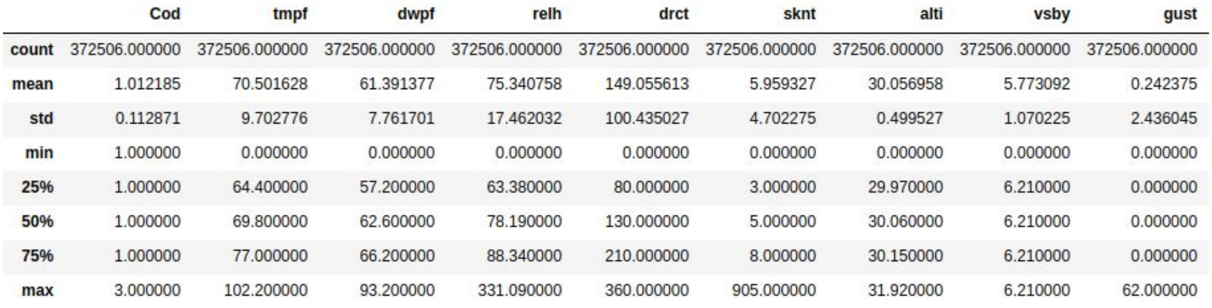
Count: Number of registers of each column

Example of merged dataset:

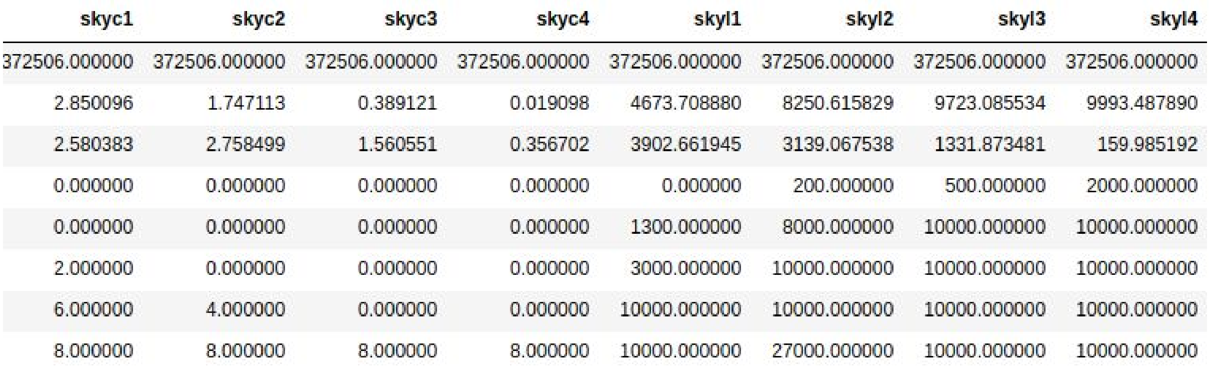


Codigo.Justificativa	MinutosVoo	Aeroporto.Importante	Hora.Prevista	tmpf	dwpf	relh	drct	sknt	alti	vsby	gust	skyc1	skyc2	skyc3	skyc4	skyl1	skyl2	skyl3	skyl4
OK	0	SBCT	2016-01-04 08:00:00	62.60	62.60	100.00	70.00	6.00	30.06	3.73	0	6	6	0	0	500.00	700.00	10000	10000
OK	0	SBCT	2016-01-04 08:00:00	62.60	62.60	100.00	70.00	6.00	30.06	3.73	0	6	6	0	0	500.00	700.00	10000	10000
OK	0	SBCT	2016-01-04 08:00:00	62.60	62.60	100.00	70.00	6.00	30.06	3.73	0	6	6	0	0	500.00	700.00	10000	10000
OK	0	SBCT	2016-01-04 08:00:00	62.60	62.60	100.00	70.00	6.00	30.06	3.73	0	6	6	0	0	500.00	700.00	10000	10000
OK	0	SBCT	2016-01-04 08:00:00	62.60	62.60	100.00	70.00	6.00	30.06	3.73	0	6	6	0	0	500.00	700.00	10000	10000
OK	0	SBCT	2016-01-04 08:00:00	62.60	62.60	100.00	70.00	6.00	30.06	3.73	0	6	6	0	0	100.00	10000	10000	10000
OK	0	SBCT	2016-01-23 08:00:00	53.60	53.60	100.00	130.00	4.00	30.09	0.31	0	6	0	0	0	100.00	10000	10000	10000
AEROPORTO DESTINO ABAIXO DOS LIMITES	188	SBCT	2016-01-23 08:00:00	53.60	53.60	100.00	130.00	4.00	30.09	0.31	0	6	0	0	0	1700.00	10000	10000	10000

A describing stats of each column is showed below, showing the count, mean, standard deviation, the minimum value, maximum value, where 25%, 50% and 75% of data is:



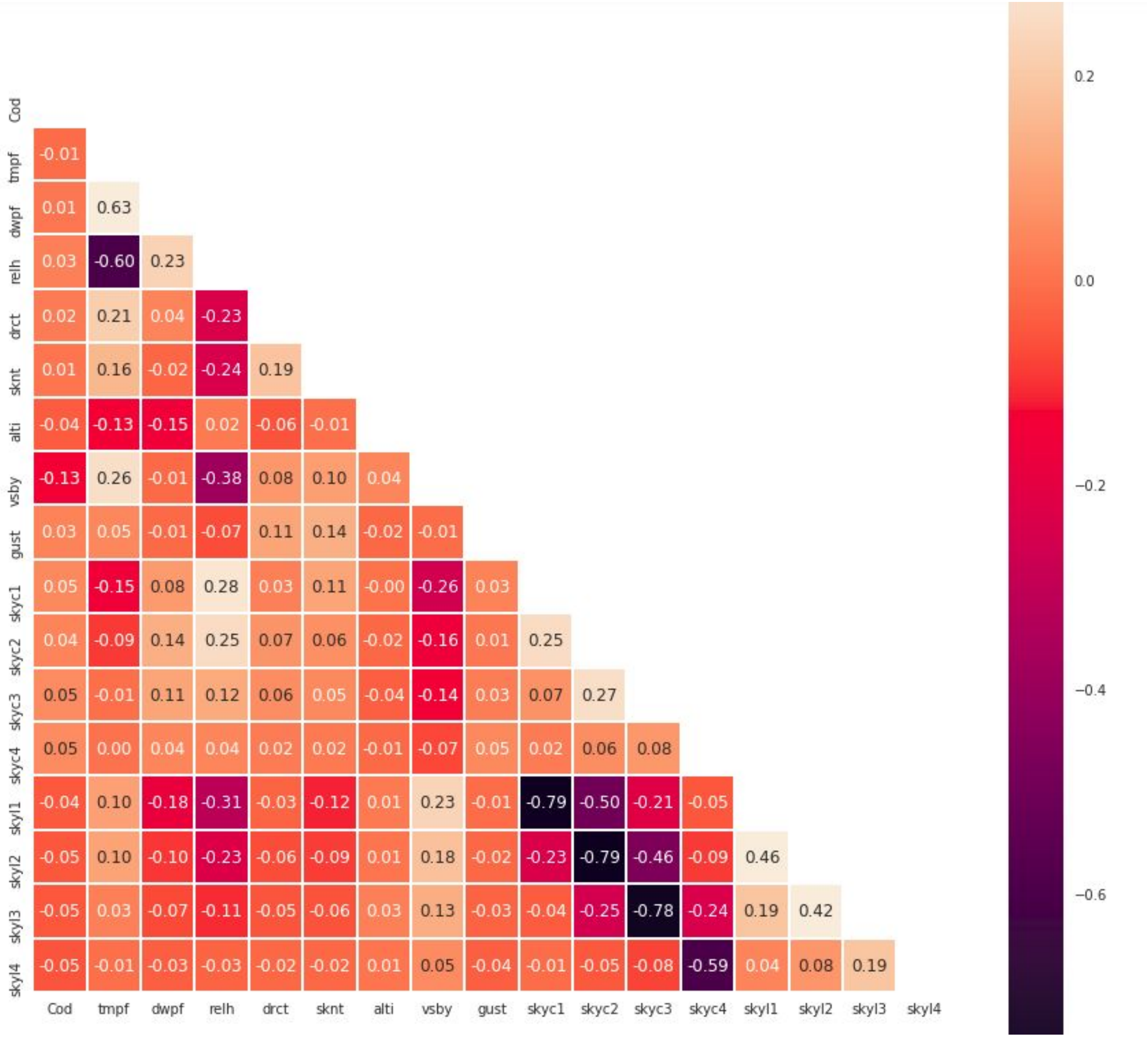
	Cod	tmpf	dwpf	relh	drct	sknt	alti	vsby	gust
count	372506.000000	372506.000000	372506.000000	372506.000000	372506.000000	372506.000000	372506.000000	372506.000000	372506.000000
mean	1.012185	70.501628	61.391377	75.340758	149.055613	5.959327	30.056958	5.773092	0.242375
std	0.112871	9.702776	7.761701	17.462032	100.435027	4.702275	0.499527	1.070225	2.436045
min	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	1.000000	64.400000	57.200000	63.380000	80.000000	3.000000	29.970000	6.210000	0.000000
50%	1.000000	69.800000	62.600000	78.190000	130.000000	5.000000	30.060000	6.210000	0.000000
75%	1.000000	77.000000	66.200000	88.340000	210.000000	8.000000	30.150000	6.210000	0.000000
max	3.000000	102.200000	93.200000	331.090000	360.000000	905.000000	31.920000	6.210000	62.000000



skyc1	skyc2	skyc3	skyc4	skyl1	skyl2	skyl3	skyl4
372506.000000	372506.000000	372506.000000	372506.000000	372506.000000	372506.000000	372506.000000	372506.000000
2.850096	1.747113	0.389121	0.019098	4673.708880	8250.615829	9723.085534	9993.487890
2.580383	2.758499	1.560551	0.356702	3902.661945	3139.067538	1331.873481	159.985192
0.000000	0.000000	0.000000	0.000000	0.000000	200.000000	500.000000	2000.000000
0.000000	0.000000	0.000000	0.000000	1300.000000	8000.000000	10000.000000	10000.000000
2.000000	0.000000	0.000000	0.000000	3000.000000	10000.000000	10000.000000	10000.000000
6.000000	4.000000	0.000000	0.000000	10000.000000	10000.000000	10000.000000	10000.000000
8.000000	8.000000	8.000000	8.000000	10000.000000	27000.000000	10000.000000	10000.000000

Exploratory Visualization

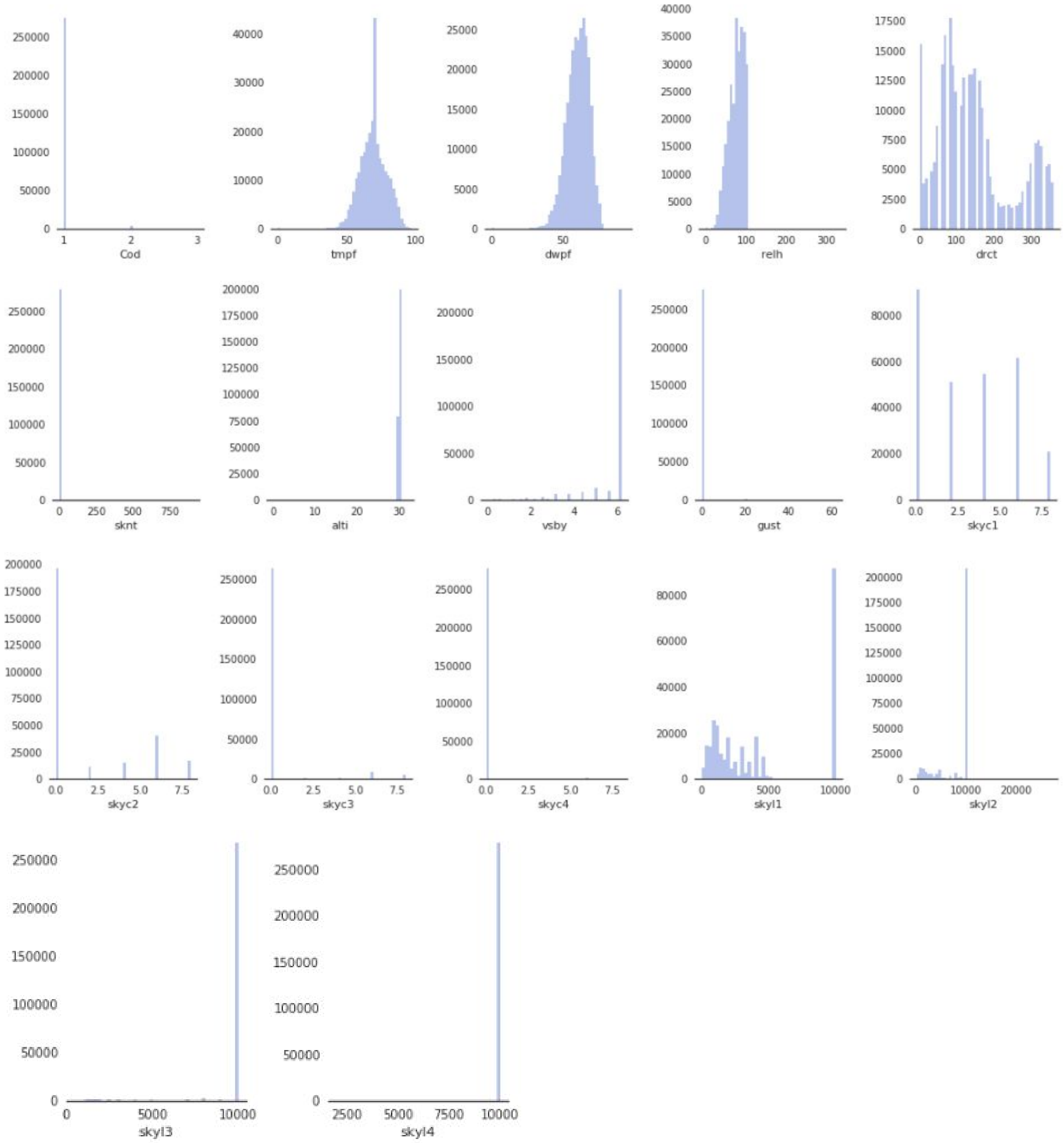
The image below shows the correlation between the columns. Sky coverages (skyc1,skyc2,skyc3,skyc4) are inversely correlated with the sky levels (skyl1, skyl2, skyl3, skyl4); the temperature of the airport are a bit correlated with the dew point. The most important observation is that nothing very correlated with the “Codigo.Justificativa” that we are trying to predict, so probably if there is a solution, starts by the combination of the columns of the dataset.



	Cod	tmpf	dwpf	relh	drct	sknt	alti	vsby	gust	skyc1	skyc2	skyc3	skyc4	skyl1	skyl2	skyl3	skyl4
Cod																	
tmpf	-0.01																
dwpf	0.01	0.63															
relh	0.03	-0.60	0.23														
drct	0.02	0.21	0.04	-0.23													
sknt	0.01	0.16	-0.02	-0.24	0.19												
alti	-0.04	-0.13	-0.15	0.02	-0.06	-0.01											
vsby	-0.13	0.26	-0.01	-0.38	0.08	0.10	0.04										
gust	0.03	0.05	-0.01	-0.07	0.11	0.14	-0.02	-0.01									
skyc1	0.05	-0.15	0.08	0.28	0.03	0.11	-0.00	-0.26	0.03								
skyc2	0.04	-0.09	0.14	0.25	0.07	0.06	-0.02	-0.16	0.01	0.25							
skyc3	0.05	-0.01	0.11	0.12	0.06	0.05	-0.04	-0.14	0.03	0.07	0.27						
skyc4	0.05	0.00	0.04	0.04	0.02	0.02	-0.01	-0.07	0.05	0.02	0.06	0.08					
skyl1	-0.04	0.10	-0.18	-0.31	-0.03	-0.12	0.01	0.23	-0.01	-0.79	-0.50	-0.21	-0.05				
skyl2	-0.05	0.10	-0.10	-0.23	-0.06	-0.09	0.01	0.18	-0.02	-0.23	-0.79	-0.46	-0.09	0.46			
skyl3	-0.05	0.03	-0.07	-0.11	-0.05	-0.06	0.03	0.13	-0.03	-0.04	-0.25	-0.78	-0.24	0.19	0.42		
skyl4	-0.05	-0.01	-0.03	-0.03	-0.02	-0.02	0.01	0.05	-0.04	-0.01	-0.05	-0.08	-0.59	0.04	0.08	0.19	

(Cod is the column Codigo.Justificativa)

The distribution of values from each column are represented below:



Codigo.Justificativa that we are trying to predict have only three values, but we can see that the values are imbalanced. The tmpf, dwpf, relh, drct skyl 1 and 2 shows to have a bigger standard deviation than the other columns.

Algorithms and Techniques

There are many Data Mining approaches for Data Balancing. I chose to use a popular approach named RandomUnderSampler.

Clustering is an Unsupervised Learning Approach. But RandomUnderSampler, only uses the concept of finding cluster centroid (clusters are created encircling data-points belonging to the majority class), as already instances are labelled. The cluster centroid is found by obtaining the average feature vectors for all the features, over the data points belonging to the majority class in feature space.

[4]

For classify the test flights i chose 3 algorithms there are popular in classification problems and easy to use and understand:

1. Random Forest :

   “A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting”. [5] Random forest obtains a class vote from each tree, and then classifies a sample using majority vote.

   $\hat{C}\ rf\ (x) = \text{majority vote}\{\ \hat{C}\ b\ (x)\}\ B$ [2]

2. KNN :

   “Neighbors-based classification is a type of instance-based learning or non-generalizing learning: it does not attempt to construct a general internal model, but simply stores instances of the training data. Classification is computed from a simple majority vote of the nearest neighbors of each point: a query point is assigned the data class which has the most representatives within the nearest neighbors of the point”. [6] To assign a label to the point x, the k-Nearest-Neighbors classifier draws a sphere centered on x enclosing exactly k training points. Afterwards, it examines the label of k training points closest to . Then the label having the largest vote is assigned to the test point x ([7] apud [2])

.

3. Adaboost : An AdaBoost classifier is a meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on difficult cases [8]. AdaBoost is a method of converting weak classifier into highly accurate prediction rule. It learns week learner on weighted example set sequentially and combines weak hypotheses linearly. Produced sequence of classifiers is dependent on the previous one and focuses on the previous one’s errors. Samples that are incorrectly predicted in the previous classifiers are chosen more often or weighted more heavily when estimating a new classifier ([7] apud [2])

Benchmark

To a good benchmark is important to check the Accuracy and the training time is too high for Flight prediction and a metric that help to work on False Positive and the True Negatives especially because they are the cases where probably will make more expensive to airline companies, especially the case that a flight is said to with no delay and it is cancelled. [2] had an accuracy higher than 80% 0 Day Forecast Horizon, its test data was obtained from forecast weather and test data from this report are METAR data that is the real weather conditions not the forecast of it, so its expected a higher accuracy than the one obtained of [2].

And the usage of confusion matrix to check True Positive, False Positive, True Negative and False Negative. So it will be checked the metric Fbeta Score that compares precision and recall that analyze effects of wrong prediction and it will be compared with [2] that had:

recall = 7178/8566 = 83.80% precision = 7178/7256 = 98.92% combined they had FScore 90.73%

III. Methodology

Data Preprocessing

Flights data:

1. Hour to datetime
2. Change name of chosen airport to their respective code example: Guarulhos to SBGR
3. Keep only empty Codigo.Justificativa:

   and the rows that contains:

   AEROPORTO ORIGEM ABAIXO DOS LIMITES, AEROPORTO DESTINO ABAIXO DOS LIMITES, ATRASO DEVIDO RETORNO - CONDICOES METEOROLOGICAS, CANCELAMENTO - CONEXAO AERONAVE/VOLTA - VOO DE IDA CANCELADO - CONDICOES METEOROLOGICAS

   if Codigo.Justificativa is null = 0

   else if Codigo.Justificativa is AEROPORTO ORIGEM ABAIXO DOS LIMITES, AEROPORTO DESTINO ABAIXO DOS LIMITES, ATRASO DEVIDO RETORNO - CONDICOES METEOROLOGICAS is set to 1,

   else = 2

4. Keep only data from flights of chosen airports, Create columns Aeroporto.Importante, Hora.Prevista, Hora.Real if at departure:

   Aeroporto.Importante = Aeroporto.Origem, Hora Prevista = Partida.Prevista e Hora.Real = Partida.Real

   else

   Aeroporto.Importante = Aeroporto.Destino, Hora Prevista = Chegada.Prevista e Hora.Real = Chegada.Real

5. Create column AtrasoVoo that contains the flight delay in minutes if the flight was canceled the it values is set to a huge value (100000), to if want later to make a regression of the delay time.

Airport data:

1. Merge files of chosen airports (one file per airport)
2. Replace values for skycs, skyls and other categorical columns

After these steps:

Merge tables by Station and valid from the second dataset with Aeroporto.Importante and Hora.Prevista from the first dataset. Remove unnecessary columns.

To finish the preprocess was treated the imbalanced data using RandomUnderSampler.

The data was normalized and splitted for training and test data. After that was used gridSearch with cross-validation (cv=10) having the scoring method as the accuracy to help training the data, changing a few hyperparameters trying to get a better result, the n_estimators in Random Forest and in Adaboost and the value of K for KNN.

Implementation

The metrics chosen were accuracy score and Fbeta score that uses precision and recall score as said before. To compare with [2] was chosen ROC curve, but there was a problem because multiclass is not supported for multiclass format is not supported by roc_curves, being more complicated to understand information from multiple roc curves that could be plotted, so to the benchmark was only kept Accuracy and Confusion_matrix and it was added the Fbeta Score.

For KNN the data was normalized. I started using Oversampling techniques as SMOTE, but the time of execution for KNN was too high, so were used the undersample technique of RandomUnderSampler that is popular to treat the imbalanced data for KNN.

Another complication was too find the best way to represent a ROC curve to these cases.

Refinement

For trying to refine the results i started trying to use a Undersampling technique, because the training time probably will be smaller than using Oversampling. So to get a better result i used changed the hyperparameter n_neighbours for KNN and n_estimators for RandomForest and Adaboost and tuning these parameters was obtained the best prediction with KNN with n_neighbours = 9 and was used to split data and for classifier algorithms the parameter random state equals to 200.

IV. Results

Model Evaluation and Validation

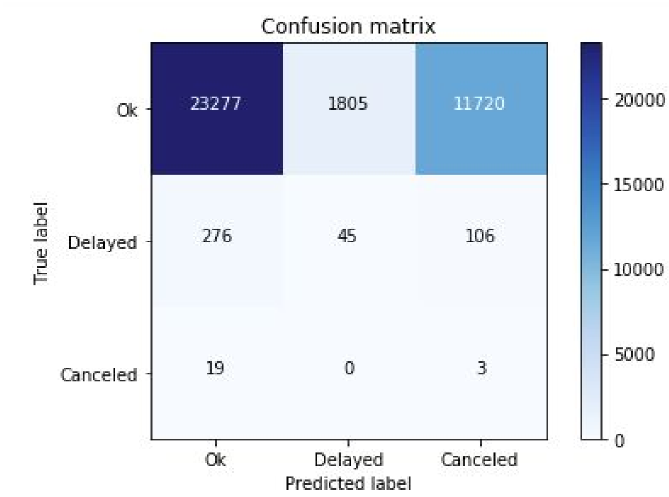
Confusion Matrix:

Was created more one column named Canceled compared to the benchmark one.

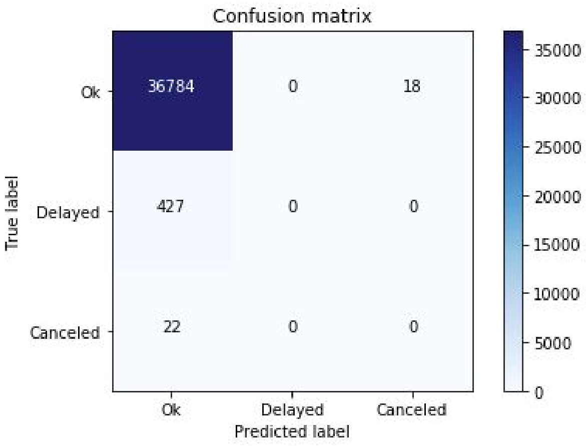
Benchmark:

	Predicted On-time	Predicted Delay
Actual On-time	7178	78
Actual Delay	1388	189

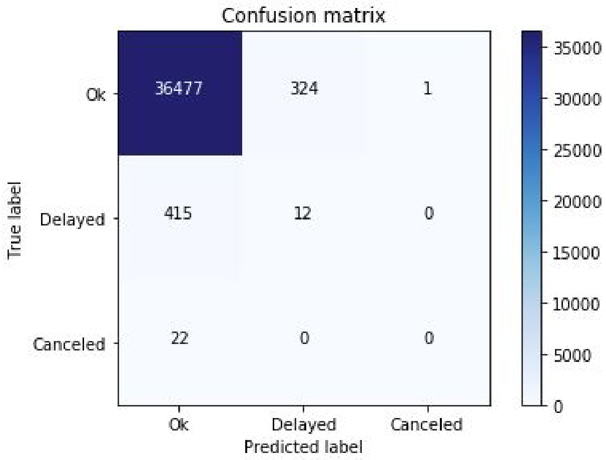
Random Forest:



KNN:



Adaboost:



Accuracy and Fbeta Score:

Benchmark:

recall = 7178/8566 = 83.80%

precision = 7178/7256 = 98.92%

F1 Score = 90.73%

	Random Forest	KNN	Adaboost	Benchmark
Fbeta Score (%)	87.74	97.83	97.72	90.73
Accuracy (%)	62.61	98.74	97.95	80.36

Refinement

The best model was KNN with the followed parameters when it was only changed the n_neighbors:

KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski', metric_params=None, n_jobs=1, n_neighbors=13, p=2,weights='uniform')

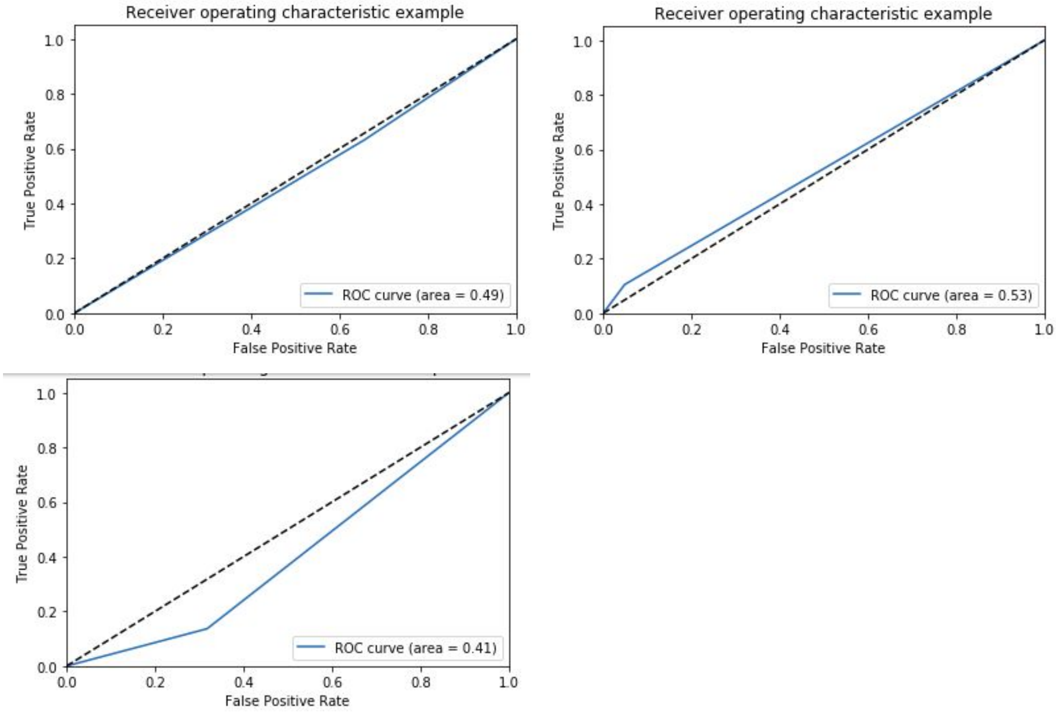
Justification

The best test of course would be with the exact forecast condition and with a dataset bigger containing at least ten years of each airport, but the classification algorithms seems to generalize well, having trained with airports with different weather conditions and with the rounded time. Having a better solution than the benchmark probably of the METAR data and because the weather variation is higher in US than in Brazil (snowing, for example, is rare in Brazil), but the best model create had a FScore approximately 8% higher and Accuracy 18% higher than the one proposed than the one obtained on [2].

V. Conclusion

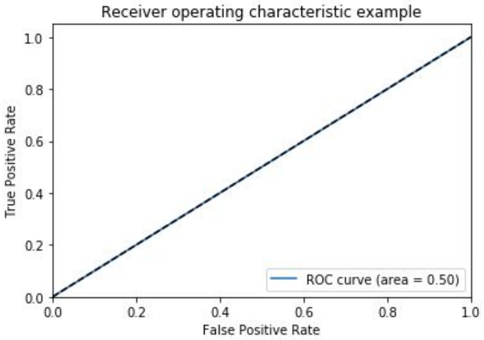
Free-Form Visualization

ROC Curve (the first is [OK, delayed], second [OK, canceled], the third[Delayed, Canceled])
Random Forest:

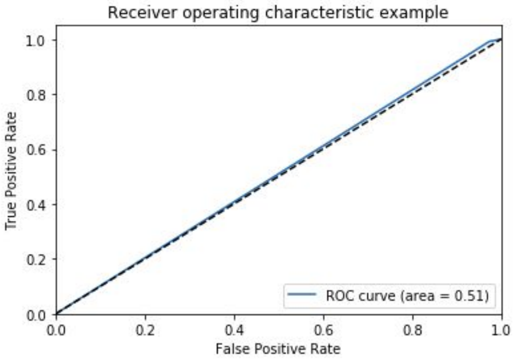


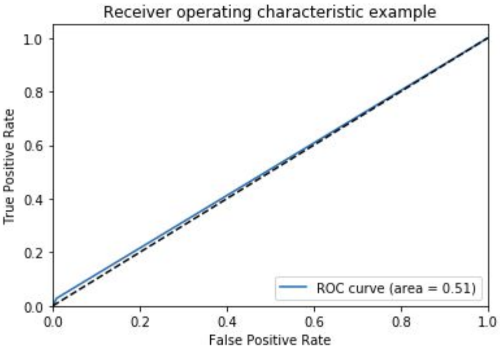
KNN:

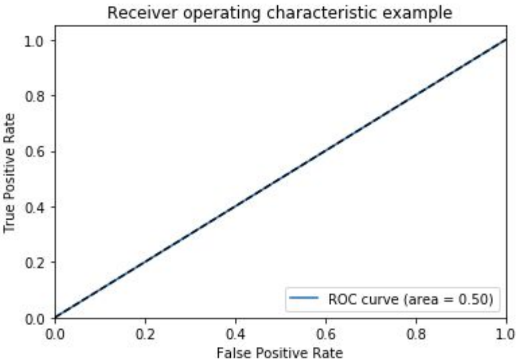
The result were the same for all cases:



Adaboost:







These Roc curves shows a problem because the ROC Curve is too low (close to 0.5), so probably it’s necessary more data to train the model and obtain better results and made the prediction from each airport at time because the different conditions to departure and arrival from each airport.

Reflection

The project needed to merge two datasets and to understand the meaning of each row. The next step was to clean unnecessary column data at the end the problem was to choose how to identify that a flight was delayed or canceled, needing to choose between a regression problem (estimate the time delay) and after a classification (if time is low the flight was Ok, if the time was medium was a delay and if the time was high probably was canceled), so i initially chose only to apply classification algorithms because it would be easy to check input data inconsistency.

Improvement

Improvement that could be made:

1. Test predictions only for one airport each time, because each airport has different conditions compared to others starting from the region and climate, but it would be necessary a dataset containing data of many years.
2. After predicting if a flight is going to be canceled or delayed, make a regression to estimate the delayed minutes of flights.
3. Try to predict the delay using the METAR of an airport hours before the flight arrives.
4. Combine METAR and forecast to only predict flight delay.

V. References

1. https://infograficos.oglobo.globo.com/economia/raio-x-dos-atrasos-dos-voos.html
2. https://ieeexplore.ieee.org/document/7777956
3. https://techcrunch.com/2018/01/31/google-flights-will-now-predict-airline-delays-before-the-airlines-do/
4. https://mesonet.agron.iastate.edu/request/download.phtml?network=BR__ASOS
5. https://www.kaggle.com/ramirobentes/flights-in-brazil/home
6. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
7. https://scikit-learn.org/stable/modules/neighbors.html
8. C. M. Bishop, Pattern Recognition and Machine Learning. Springer, 2006.

9. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html
10. https://stackoverflow.com/questions/48817300/sklearn-plot-confusion-matrix-combined-across-trainingtest-sets (Confusion matrix plot)
11. https://medium.com/greyatom/lets-learn-about-auc-roc-curve-4a94b4d88152 (Multiclass ROC plot)