Application of classification algorithms to predict flight delays

Alexandre Crivellari

Claudio

Gabriele

1Università degli Studi di Milano – Bicocca - Data Science [FDS01Q]

# Summary

***keywords***:

# Index

Summary 1

Introduction 1

Data Exploration and Visualization 2

Preprocessing 3

Modelling 3

Performance Evaluation 4

Conclusions and Future Research 5

References 6

# Introduction

With air travel becoming increasingly affordable to the average person, air traffic has witnessed an exponential growth. To this day, around 115,000 daily flights are scheduled around the world [1]. The business of the airspace shaped an environment in which respecting strict time timetables has become challenging, with the delay of a flight’s landing affecting the punctuality of the next flight’s takeoff.

Flight delays are a source of customer dissatisfaction. Not only this, they are also extremely costly to airline companies. The estimated cost of this phenomenon in the USA, in 2017, was $32.9 billion.

Given the considerable size of the inefficiencies caused by them, flight delays should be predicted and, if possible, restrained. With this aim, this project applies machine learning algorithms to the [Airline Dataset to Predict a Delay](https://www.kaggle.com/datasets/jimschacko/airlines-dataset-to-predict-a-delay) [3] dataset (retrieved on Kaggle in January 2023), in order to understand if:

1. any kind of prediction is the flight delay is possible with the given features, and
2. there are some features that heavily contribute to the probability of a flight being delayed.

# Data Exploration and Visualization

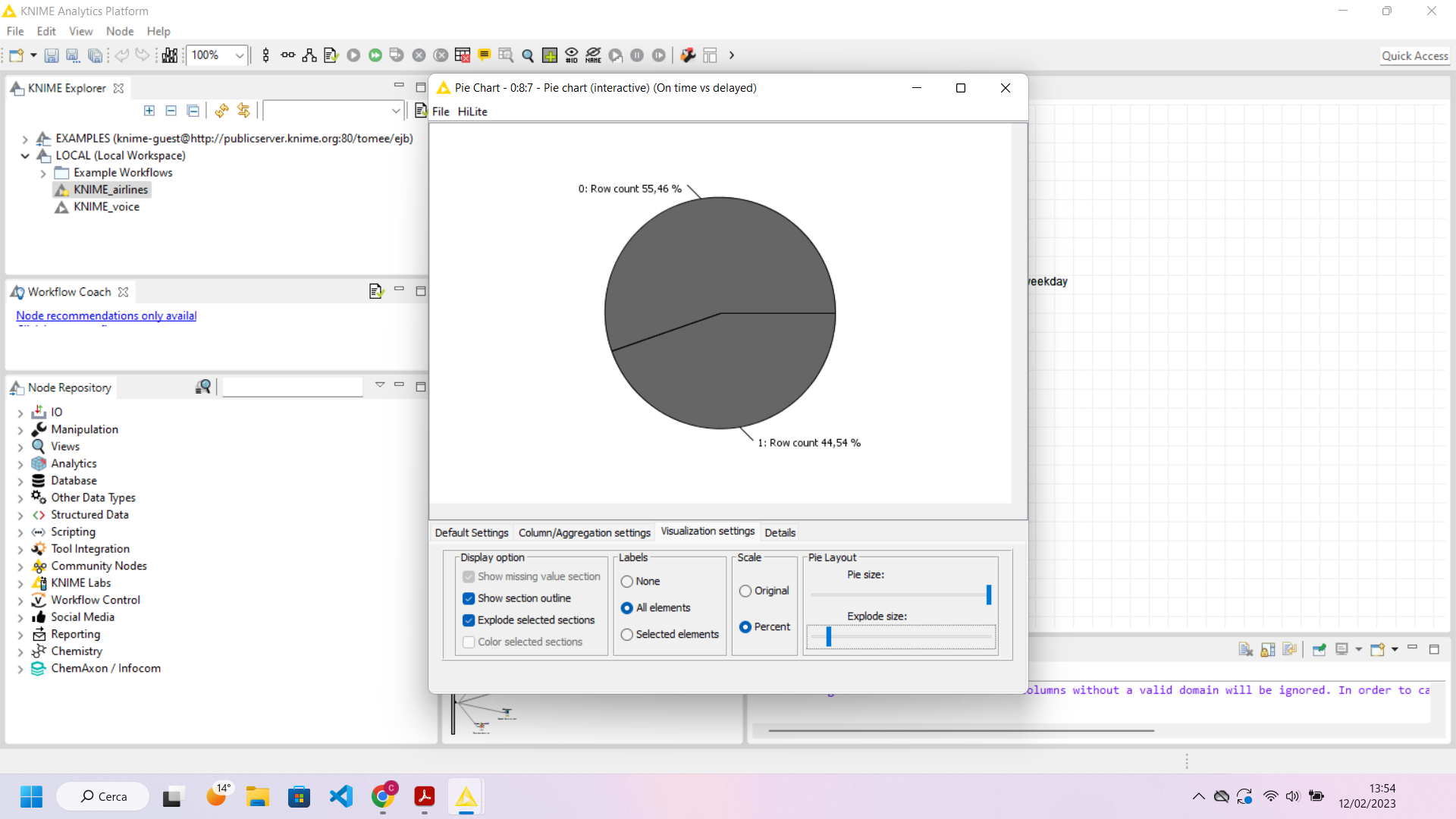
The dataset is composed of 539,383 records and 8 attributes.

The eight attributes and their description are the following:

1. **Airline**: a string code identifier for each commercial airline company;
2. **Flight**: identifying number of the flight;
3. **AirportFrom**: a string code identifier for each departure airport;
4. **AirportTo**: a string code identifier for each arrival airport;
5. **DayOfWeek**: day of the week of departure. This attribute can take values in a [1-7] range;
6. **Time**: the time of departure, expressed in minutes elapsed since midnight (i.e.: a Time value of “60” corresponds to 01:00 a.m., “120” corresponds to 02.00 p.m., and so on).
7. **Length**: the flight length duration, in minutes.
8. **Delay**: whether the flight was on time (0) or delayed (1). This is the **class attribute**.

The dataset focuses on a subset of American internal flights. In other words, non-American, international flights are outside the scope of this analysis.

Delayed flights, in the dataset, account for 44,54% of the total. **The dataset is thus rather balanced under this respect**. Regardless, further care will be needed when evaluating the performance of classifiers. In fact, the class of interest, i.e. flights that are delayed, is slightly less numerous than the other class, which could lead to biased interpretations of accuracy measures.



A preliminary, visual exploration of the dataset yielded a set of expectations regarding what could be the possible outcomes of a wider analysis. It was spotted that Monday, Wednesday, and Sunday tend to be the days of the week in which the percentage of delayed flights is the highest, with 46.76%, 47.08%, and 45.35% of flights not arriving on time, respectively. It must be noted that all daily averages are concentrated around the overall mean, 44.54%.

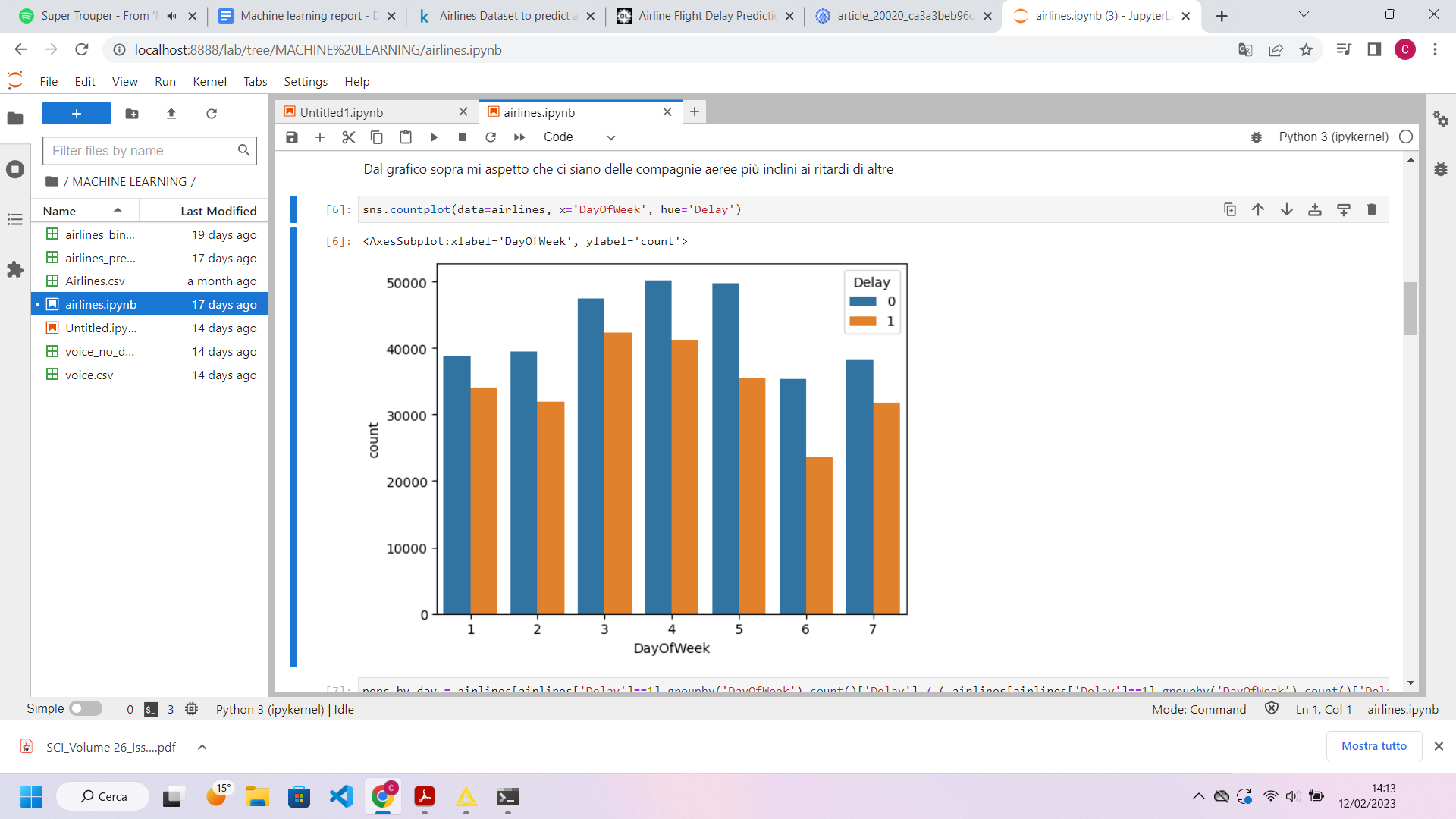


Figure 1. Delay distribution over the day of the week, from Monday to Sunday

In addition to this, some airline companies are more prone than others to experience a flight delay. In particular, those who travel with SouthWest Airlines (WN) and Continental Airlines (CO) will, more likely than not, arrive at their destination with some delay.

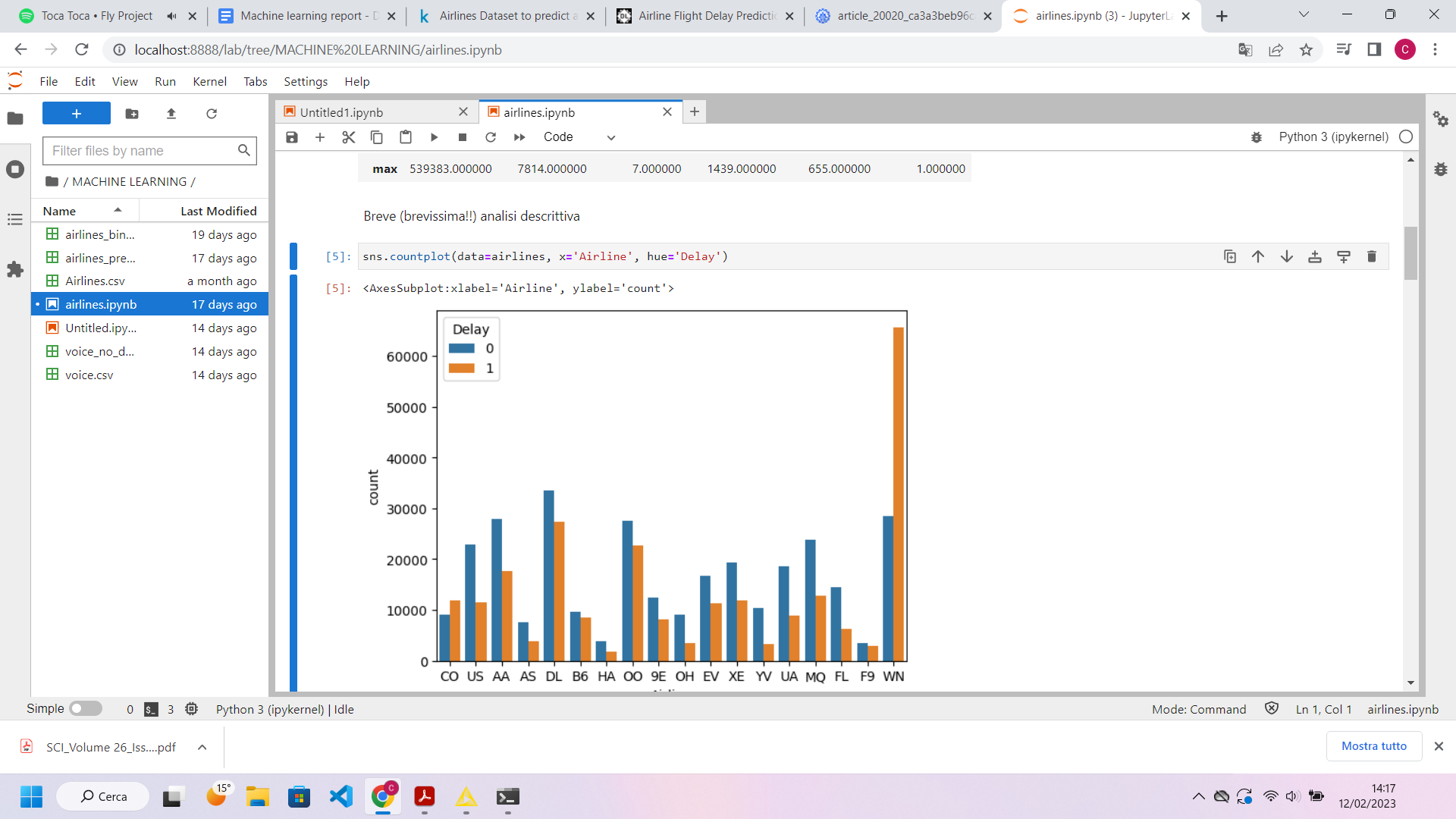


Figure 2. Delay distribution over the Airline company.

Lastly, flights scheduled for times between 850 and 1300 (that is, scheduled between 02:00 p.m. and 9:00 p.m.) are more often delayed than not, which could probably induce that the less “busiest hours” will be more correlated to a flight landing on time.

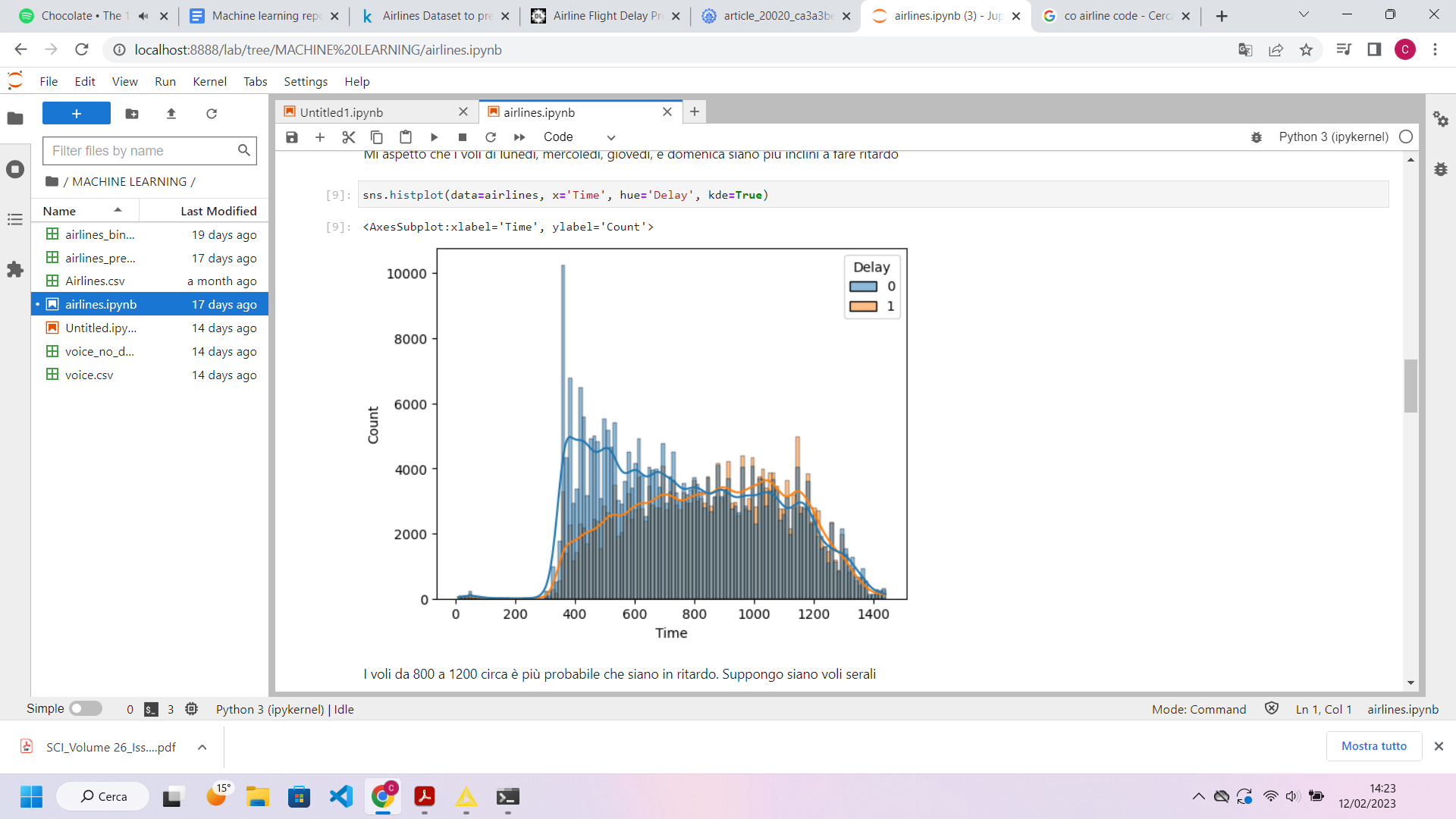


Figure 3. Delay distribution over the time of the day.

In light of this, **it could be expected that these attributes will contribute to great extent to the prediction of whether a flight will experience a delay**.

# Preprocessing

To prepare the dataset for the analysis, a set of preprocessing activities were performed on the original dataset.

First and foremost, **all features should be converted to their correct data types**, with particular attention to Flight, DayOfWeek and Delay being transformed into strings (as they do not make any sense as numerical features).

For the purposes of improving the algorithm, **dimensionality reduction techniques were employed**. Using domain knowledge, it would seem unreasonable that the Flight feature (again, the identification number for any flight) could be a good predictor of weather a flight could arrive in time or not. By using feature selection (a Backward Feature Elimination algorithm, trained on a Naïve Bayes Learner), this hypothesis has been confirmed. A feature filter revealed that the performances of the model did not significantly decrease after the removal of the attribute “Flight”, which was therefore excluded from the analysis.

To further reduce the granularity of the information in the dataset, and to make it much more interpretable, **the attribute Time has been discretized**. This was done because the attribute could take 1439 possible values, a factor that would have hampered the intelligibility of results. In particular, it was divided into eight categories of equal width. The original attribute was then replaced by the discretized one.

# Modelling

In this project, four different models were trained and tested, to then elect the best performing one given the data at disposal.

The models employed in this analysis are the following:

1. **Decision Tree**. This model is composed of a set of nodes, branches and leaves that partition the various instances until a decision node is reached. In this analysis, the algorithm starts by analyzing the airline company supplying the flight, and then proceeds by considering all additional attributes.
2. **Random Forest**. This algorithm is a collection of decision trees. Each tree utilizes attributes randomly and independently from other trees. For this reason, random forests typically ensure a level of accuracy higher than that of individual trees, as they build a set of uncorrelated trees.
3. **Gradient Boosting Trees**. This boosting algorithm is a variant of the Decision Trees ensemble methods: it creates a set of different models, and then combines them to obtain better performances.
4. **Naive Bayes**. This probabilistic model is based on Bayes’ theorem. It allows for the computation of the a posteriori probability of an event’s occurrence.

In the first phase, all models were trained using the **holdout method**. The dataset was partitioned in a training and a test set. The former accounted for 67% of the total observations, while the latter for 33%.

The type of sampling used was stratified random sampling, with a binning on attribute “Airline”. This was done to ensure that the respect of the proportions of this attribute between the training and testing partitioning, and the original dataset.

The results of this first training and testing phase are the following.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Classifier* | *Recall* | *Precision* | *F-measure* | *Accuracy* |
| **Decision Tree** | 0.62 | 0.64 | 0.62 | 0.64 |
| **Random Forest** | 0.60 | 0.66 | 0.58 | 0.63 |
| **Gradient Boosting** | 0.62 | 0.64 | 0.61 | 0.64 |
| **Naive Bayes** | 0.61 | 0.64 | 0.61 | 0.64 |

Table 1. Evaluation measures for the models used, considering a 67% partitioning training/test.

To further ensure the robustness of scores, each model was trained and tested using **five-fold cross validation**. According to this method, the dataset was split into five exhaustive partitions, and each model underwent the training phase five times; at each iteration, four partitions of the dataset were employed as training set and one as test set. The final metrics are computed as the mean of the results obtained in each of the five processes, and are reported below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Classifier* | *Recall* | *Precision* | *F-measure* | *Accuracy* |
| **Decision Tree** | 0.62 | 0.64 | 0.62 | 0.6414 |
| **Random Forest** | 0.61 | 0.66 | 0.59 | 0.6365 |
| **Gradient Boosting** | 0.62 | 0.64 | 0.61 | 0.6419 |
| **Naive Bayes** | 0.61 | 0.64 | 0.61 | 0.6358 |

Table 2. Evaluation measures for the models used in combination with a K-Fold Partitioning (5 iterations).

# Performance Evaluation

Before dwelling into any consideration it is important to have a clear set of expectations from this dataset, just by looking at its structure alone. The considerable numerosity of records (over half a million rows, with no missing or incorrect data), along with the scarcity of available features (almost all of them, moreover, of nominal nature) should (and will be) somewhat of a challenge in making any useful prediction. The truth is: that is exactly the case.

First and foremost, **there does not seem to be any significant difference in the predictive power for any of the four models**. The following graphs include the ROC curves and Lift charts for the two better performing models, *Random Forest* and *Gradient Boosted Trees.*

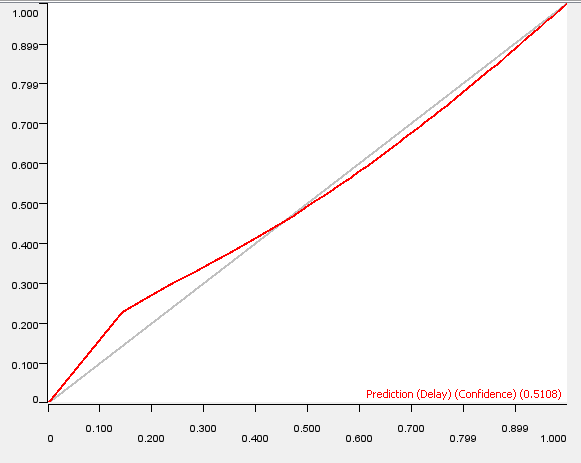


Figure 4. ROC Curve for Random Forest model.

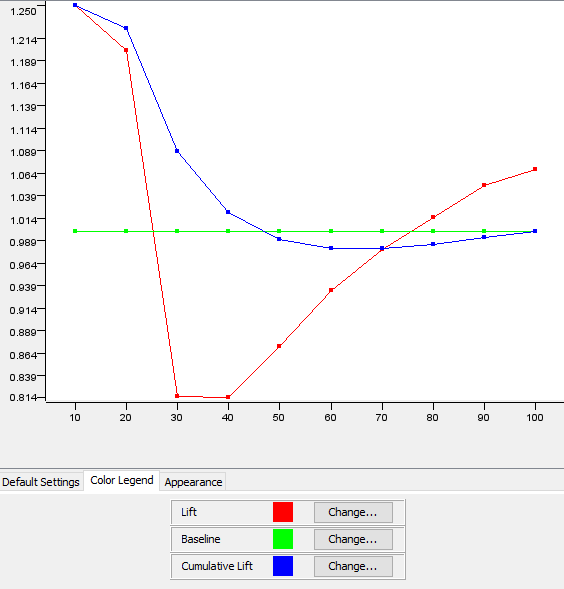


Figure 5. Lift Chart for Random Forest model.

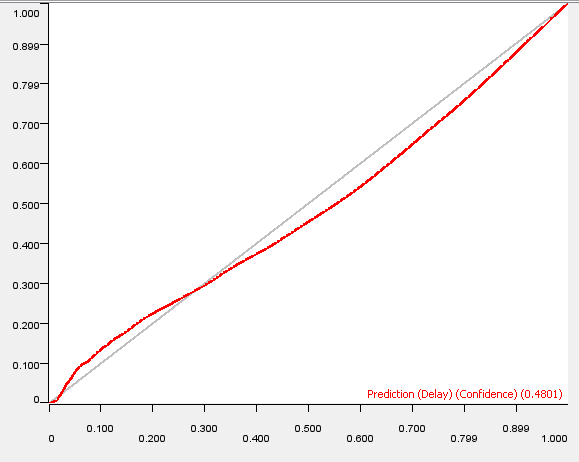


Figure 6. ROC Curve for Gradient Boosted Trees model.

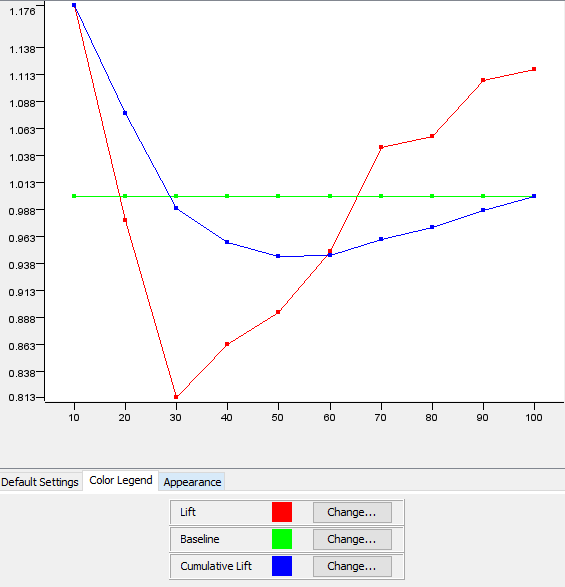


Figure 7. Lift Chart for Gradient Boosted Trees model.

With an **area under the ROC Curve** of **0.536** and **0.5048** (for the Random Forest model and the Gradient Boosted Trees model, respectively), it should be accepted that **no model is inherently good enough in predicting with great confidence** whether a flight is going to be delayed or not.

The **Lift charts** for both approaches confirms this statement (with, again, a more positive outlook in the evaluation of the Random Forest model). It is evident from the Lift chart that the improvement of the model performance seems to be **an almost insignificant improvement over the baseline.** It is vital to underline that a lift factor smaller than 1 means that such portion of the records in the model contains fewer correct predictions that those that would have been made with a random sample.

An additional piece of information comes from the decision to test both a Holdout method and a K-Fold partition with five iterations. Data shows that, while computationally very expensive, **the K-Fold partition method does not aid in achieving significantly better results** (again, this is a confirmation of a prior hypothesis, given the numerosity of records available in the dataset). In lieu of this, **it would be advisable to prefer computational speed over accuracy performance** in further optimizing this algorithm, as the number of records available is more than enough in this regard.

# Conclusions and Future Research

After many iterations of fine-tuning of the parameters, a 64% accuracy of the model cannot be defined, *per se*, as a “bad” result in terms of predicting power: the observations are many and the quality of data seems consistent, thus giving this exercise a somewhat improvement from a 50/50% baseline. Nevertheless, with regards to the initial research question, **it is most definitely not advisable to use this dataset** for making strong, confident predictions about the probability for a flight being delayed.

Alas, given the problem at hand, we have asked ourselves what kind of additional features might be useful in understanding the predictability of a flight being delayed. Here, knowledge about the *domain* of the issue at hand could give some insight into making hypotheses about **what additional work might yield better results**. With some additional domain information and a bit of data collection, there are some paths that could potentially significantly improve this algorithm.

First and foremost, an improvement to the dataset might be achieved by collecting the **weather condition** for both the **airports of departure** and the **airports of arrival** for each observation. Heavy rains -or bad weather in general- might be a very valid reason for a flight being delayed (or canceled altogether), making these two features potentially very important in improving our classification model. In addition, **wind speed** and **wind direction**, both -again- **at departure** and at **arrival,** could give additional information to our model as well. This could be very good news, as weather information might be easily accessible from weather records.

Secondly, information about the **airport dimension and capacity** could prove useful as well. *Bigger airports* could, hypothetically, provide more resilience to possible technical issues or *overcrowding* of the lanes, given the physical availability of more lanes or additional personnel. On the contrary, *smaller airports* might be more prone to being a direct cause to a flight delay, especially if any technical difficulty for any given flight results in a “*domino effect” down the lane.* This is not a straightforward option to collect as the previous feature but there are some *“good enough”* proxies that might just come in handy: the **total number of available lanes**, the **number of available gates** or the **square meters of the airport lanes** could give a reasonable hint.

Another important piece of information could very well be **the specific date** in which the flight was made: holiday seasons might give a hint in the level of “crowding” of the airport, again correlating with any specific incidents that might occur because of that. **These additional features mentioned above might potentially yield crucial information**, capable of very significantly improving our model, as demonstrated by other research on the same topic [4].

In conclusion, while not being particularly exhaustive in the predictive power for a flight delay, this intriguing dataset already provides some very interesting starting points for further analysis, the most notable being:

1. While during the late night/early morning the great majority of flights arrive on time, *as the day goes by more and more flights get delayed*.
2. There is some kind of explainability between airports and airlines and the probability of a delay, which means that, most probably, the behavioral organization of efficient and inefficient airports and airlines do have some importance in what we might call “the delay reputation”.

# References

[1] Duggal H, Haddad D.. *Visualising the global air travel industry*, 2021. Retrieved from <https://www.aljazeera.com/economy/2021/12/9/visualising-the-global-air-travel-industry-interactive>

[2] Khaksar H., Sheikholeslami A.. A*irline delay prediction by machine learning algorithms*, 2017. Retrieved from <http://scientiairanica.sharif.edu/article_20020_ca3a3beb96caf169b6f454c703517d0c.pdf>

[3] Kaggle (January 2023). *Airlines Dataset to predict a delay*. Retrieved from https://www.kaggle.com/datasets/jimschacko/airlines-dataset-to-predict-a-delay

[4] Yi Ding, *Predicting flight delay based on multiple linear regression,* 2017 IOP Conf. Ser.: Earth Environ. Sci. 81 012198. Retrieved from <https://iopscience.iop.org/article/10.1088/1755-1315/81/1/012198>