Analysis of delayed flights

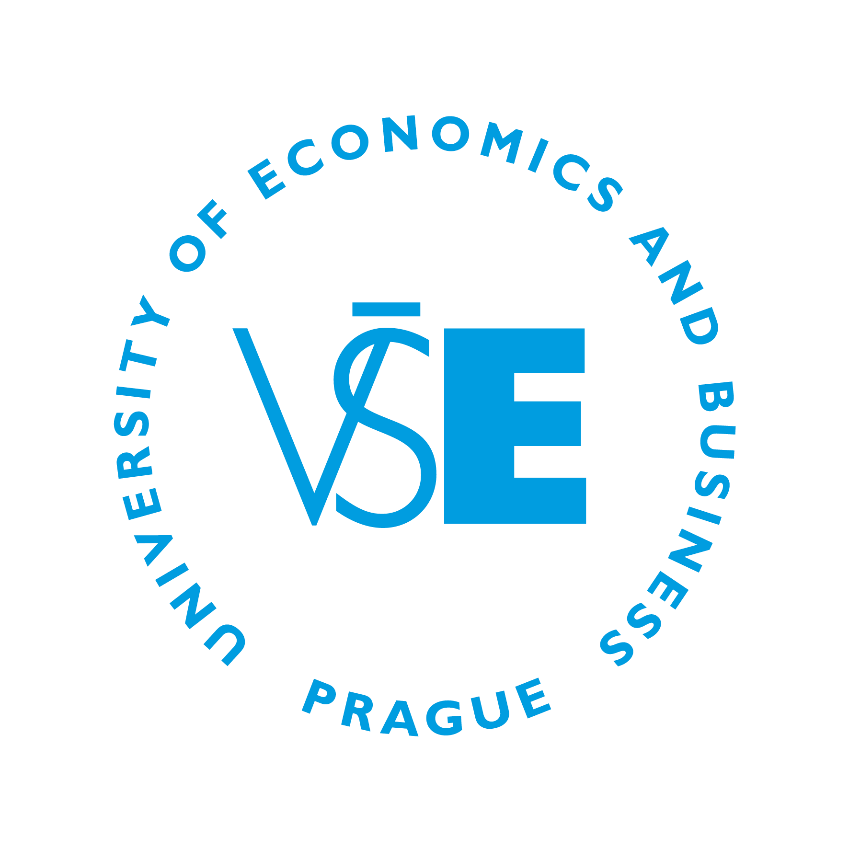
4IZ171 Strojové učení 1

Bílková Helena, bilh00

Lauer Michal, laum02

Tomášková Tereza, tomt04

Vinkled Vladimír, vinv05



Contents

[Introduction 3](#_Toc133163069)

[Supervised learning 5](#_Toc133163070)

[Data preparation 5](#_Toc133163071)

[Approach to training 6](#_Toc133163072)

[Decision tree 6](#_Toc133163073)

[Random forest 6](#_Toc133163074)

[Evaluation 6](#_Toc133163075)

[Threshold 7](#_Toc133163076)

[Explanation 8](#_Toc133163077)

[Single prediction 8](#_Toc133163078)

[Unsupervised learning 9](#_Toc133163079)

[Data preparation 9](#_Toc133163080)

[K-means 9](#_Toc133163081)

[Hierarchical clustering 11](#_Toc133163082)

[Evaluation 13](#_Toc133163083)

[Conclusion 13](#_Toc133163084)

[Requirements 13](#_Toc133163085)

[Figures 14](#_Toc133163086)

[Tables 14](#_Toc133163087)

# Introduction

We analyzed a dataset about flights and their delays. The dataset can be freely found on [kaggle.com](https://www.kaggle.com/datasets/ulrikthygepedersen/airlines-delay).

There are eight different columns:

* *Flight* - Flight ID,
* *Time* - Time of departure (minute of the day the flight departed at),
* *Length* - Length of Flight (in minutes),
* *Airline* - Airline ID,
* *AirportFrom* - Which airport the flight flew from,
* *AirportTo* - Which airport the flight flew to,
* *DayOfWeek* - The day of the week of the flight, and
* *Class* - Delayed (1) or not (0).

The flight ID is not unique, as one flight can be represented multiple times. We wanted a unique identifier for each row and added an *ID* column. The first five columns are shown in Table 1.

Table First five rows of the analyzed dataset

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Flight | Time | Length | Airline | AirportFrom | AirportTo | DayOfWeek | Class |
| 0 | 2313 | 1296 | 141 | DL | ATL | HOU | 1 | 0 |
| 1 | 6948 | 360 | 146 | OO | COS | ORD | 4 | 0 |
| 2 | 1247 | 1170 | 143 | B6 | BOS | CLT | 3 | 0 |
| 3 | 31 | 1410 | 344 | US | OGG | PHX | 6 | 0 |
| 4 | 563 | 692 | 98 | FL | BMI | ATL | 4 | 0 |

We will analyze the *Class* variable, e.g., whether or not the flight has been delayed. The first thing we have checked is whether the data is balanced.

A picture containing text, screenshot, plot, line

Description automatically generated

Figure Balance of variable Class

The absolute counts are displayed in Figure 1. We can see that the data is balanced, and one class does not heavily outweigh the other one.

For variables *Time* and *Length*, we created a histogram showing the distribution of these variables.

A picture containing screenshot, plot, text, diagram

Description automatically generated

Figure Histogram of variables Time and Length

In Figure 2, *Time* does not have almost any tails, and all the data is close to the mean. The variable *length* has a high peak and is positively skewed. We also created a bar chart of all Airlines, as seen in Figure 3.

A picture containing text, screenshot, diagram, plot

Description automatically generated

Figure Bar chart of variable Airline

There are 18 different airline companies in total. The most frequent airline in the dataset is WN (Southwest Airlines), while the least-flown airline is HA (Hawaiian Airlines). Variables *AirportFrom* and *AirportTo* contain a lot of unique airports, and no clean and pretty graph would display them all. Furthermore, we do not think a graph with 50+ categories would give us additional information. Finally, we looked at the *Day of week* variable (Figure 4).

A picture containing text, screenshot, font, number

Description automatically generated

Figure Bar chart of variable Day of week

We can see that most flights are flown on Wednesday, Thursday, and Friday. The last flights are taken on Sunday. This makes sense for us, as people usually take a long weekend and fly out during the last few days of a working week.

# Supervised learning

## Data preparation

We transformed our *AirportFrom* and *AirportTo* variables into one variable called *Route*, which represents from where to where an airport has flown. For example, if a flight flies from Atlanta to Houston, the route for this flight is „ATL -> HOU“. This creates 4 190 different routes.

As random forests and decision trees in the *sklearn* library work mainly with numbers, we must trencode all variables into numeric format.

The next step is splitting the data into train and test samples. We do this because we do not want to overfit our models, and we can evaluate them on new data. This also helps us get more precise final metrics. We created two different split samples. In the first one, the test sample makes up about 30 % of all data, while the second makes up about 95 %.

## Approach to training

When training the models, we use a grid search method to find the best hyperparameters for our data. Then, to find the best model, we used K-fold cross-validation with three folds.

## Decision tree

To train an optimal model, we fine-tuned the maximum depth of each tree. The optimal depth for our tree is 10. When we used the smaller training sample (5 %), the resulting max depth was 5.

## Random forest

For random forests, we fine-tuned the number of estimators and the max depth of trees. The result tells us that the decision tree should contain 25 trees, and the tree depth is 10. When the random forests were optimized using the smaller training sample, the number of trees was optimized at 200, and the maximum depth of each tree was 10.

## Evaluation

Since the target variable (*Class*) is well-balanced, we use accuracy to asses the four models we created. The results are:

1. Accuracy Score Random forest (train 70 %): 66.46 %
2. Accuracy Score Decision tree (train 70 %): 64.7 %
3. Accuracy Score Random forest (train 5 %): 69.95 %
4. Accuracy Score Decision tree (train 5 %): 65.27 %

We see that all models are relatively close in terms of their accuracy. The best-performing model is a random forest with an accuracy of about 70 %. Since this model is considered the best, a classification report is shown only for the random forest. All resulting statistics are shown in Table 2.

Table Classification report on the random forest with 5 % of training data

|  |  |  |
| --- | --- | --- |
| Statistic | Value for Class = 0 | Value for Class = 1 |
| Precision | 0.69 | 0.72 |
| Recall | 0.83 | 0.53 |
| F1-score | 0.75 | 0.61 |

Another statistic that we looked at is the Reciever Operating Characteristics (ROC) curve and tha Area under curve (AUC) associated with it. We compared all four built models to asses which one is the best.

A picture containing text, line, screenshot, plot

Description automatically generated

Figure ROC and AUC for all four models

Figure 5 displays all resulting curves and areas. We can see that the Random Forest model with 5 % of training data is superior to all other models with AUC equal to 0.77.

## Threshold

We have created a cost matrix to identify what threshold is optimal for our business case. To pick the best threshold, we need to optimize the results of our Random Forest against the cost matrix displayed in Table 3.

Table Cost matrix

|  |  |  |
| --- | --- | --- |
| Actual date vs. Predictions | Prediction of Class = 1 | Prediction of Class = 0 |
| Actual Class = 1 | 0 (True positive) | 5 (False negative) |
| Actual Class = 0 | 1 (False positive) | 0 (True negative) |

We set that if we predict that the flight is not delayed when it is, this false negative costs us five imaginary points. On the other hand, if we predict that the flight will be delayed when it is not, the false positive costs us one point. After optimization, the optimal threshold is 0.23, costing us 260 332 points. If the prediction is below 0.23, we say the flight is not delayed. If the prediction is greater than 0.23, the flight should be delayed.

## Explanation

Based on the best random forest, the most important variables in deciding whether a flight will be delayed are *Airline* and *Time* (Figure 6).

A picture containing text, screenshot, font, line

Description automatically generated

Figure Feature importance

## Single prediction

To further test our model, we created a made-up flight, and we will see whether it will be delayed. The flight is described in Table 4.

Table Made-up flight

|  |  |
| --- | --- |
| Property | Value |
| Flight | 15 |
| Time | 710 |
| Length | 655 |
| Airline | 4 |
| AirportFrom | 96 |
| AirportTo | 128 |
| DayOfWeek | 4 |
| Route | 1045 |

The best Random forest we picked predicted that his flight would be delayed with a probability of 0.74.

# Unsupervised learning

## Data preparation

We selected 27 618 flights operated by a UA (United Airlines) for clustering methods. As both k-means and Hierarchical clustering require the data to be numerical, we chose only the *Time* and *Length* variables. It did not make sense for us to recode categorical variables into numbers because the distance between each category is different and has no meaning.

We then normalized both variables so the mean is zero and the standard deviation is one. This will normalize the distances and variance in different units in both variables.

## K-means

The first method that we used is k-means. To tune the number of clusters, we will try k from 1 to 10. We will also try two different algorithms – elkan and lloyd. The resulting numbers are shown in Table 5.

Table Overview of Inertia for k-means

|  |  |  |
| --- | --- | --- |
| Number of clusters | Algorithm used | Inertia score |
| 10 | elkan | 5223.869916 |
| 10 | lloyd | 5226.245132 |
| 9 | lloyd | 5751.023296 |
| 9 | elkan | 5751.116140 |
| 8 | lloyd | 6521.023600 |
| 8 | elkan | 6521.302605 |
| 7 | elkan | 7530.910170 |
| 7 | lloyd | 7531.052582 |
| 6 | elkan | 8770.378576 |
| 6 | lloyd | 8785.740332 |
| 5 | elkan | 10539.942084 |
| 5 | lloyd | 10540.832915 |
| 4 | lloyd | 13066.798176 |
| 4 | elkan | 13066.853798 |
| 3 | lloyd | 19480.001853 |
| 3 | elkan | 19480.001853 |
| 2 | lloyd | 33826.930111 |
| 2 | elkan | 33827.080245 |
| 1 | lloyd | 55238.000000 |
| 1 | elkan | 55238.000000 |

We will use a Scree plot for both algorithms to assess the best combination. Based on Figure 7, we can say that both algorithms perform very similarly. The optimal number of clusters is 3 or 4.

A picture containing text, plot, diagram, line

Description automatically generated

Figure Scree plot for k-means

The combination we will use is elkan’s algorithm and 4 clusters. Plotted data, colored by its assigned cluster, are shown in Figure 8.

A picture containing text, map, diagram, colorfulness

Description automatically generated

Figure Data colored by the results from k-means

The data is clearly split, mainly because we use only two variables displayed in a 2d space. There is no missing variability, and the clusters are split and visible. Clusters split the data into four groups, each with a specific combination of short/long flights. Interestingly, some flights are short but long, and flights are fast but long.

We tried to see if the clusters split the flights meaningfully regarding whether they had been delayed. Two plots were created – one shows the delayed flights upfront, and the second one shows not delayed flights upfront. Both graphs can be seen in Figure 9.

A picture containing text, screenshot

Description automatically generated

Figure Delayed and not delayed flights within clusters

It is clear that flights, no matter the delay, are evenly distributed across variables *Length* and *Time*. No clustering method on these two variables can split the data well to identify signs of delay.

## Hierarchical clustering

When modeling the data using hierarchical clustering, we used three methods:

* Complete,
* Linkage, and
* Single.

Individual dendrograms are in Figures 11 (complete), 10 (average) and 12 (single). Based on the dendrograms, we also identified the optimal number of clusters based on the maximum distance between them. The optimal number of cluster is

* 3 for complete clustering,
* 3 for average clustering, and
* 1 for single clustering.

A picture containing diagram, rectangle, screenshot, text

Description automatically generatedA picture containing diagram, text, rectangle, plan

Description automatically generated

Figure Dendrogram for average clustering

Figure Dendrogram for complete clustering

A green and black rectangle

Description automatically generated with medium confidence

Figure Dendrogram for single clustering

## Evaluation

All methods are compared in Table 6. The best method for our data is k-means clustering.

Table Comparison of clustering methods

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Specification | Best k | Inertia/WSS |
| K-means clustering | Alkan algorithm | 4 | 13 066 |
| Hierarchical clustering | Complete clustering | 3 | 27 039 |
| Hierarchical clustering | Average clustering | 3 | 39 482 |
| Hierarchical clustering | Single clustering | 1 | 55 238 |

# Conclusion

asdf asdf asfd

## Requirements

Table Requirements for the assignment

|  |  |  |
| --- | --- | --- |
| Learning | Requirement | Result |
| Supervised learning | Attribute of interest and it’s change | x |
| Supervised learning | Comparison of models | ✓ |
| Supervised learning | Cost matrix | ✓ |
| Supervised learning | Evaluation | ✓ |
| Supervised learning | Feature selection | ✓ |
| Supervised learning | Hyperparameter tuning | ✓ |
| Supervised learning | Instance of interest | ✓ |
| Supervised learning | Most important variables | ✓ |
| Supervised learning | New columns | ✓ |
| Supervised learning | Optimized cost matrix | *✓* |
| Supervised learning | Train test split | *✓* |
| Unsupervised learning | Different methods | *✓* |
| Unsupervised learning | Elbow graph | *✓* |
| Unsupervised learning | Hyperparameter tuning | *✓* |
| Unsupervised learning | Interpret clusters | *✓* |
| Unsupervised learning | Rescaling | *✓* |
| Unsupervised learning | Row subsetting | *✓* |

# Figures

[Figure 1 Balance of variable Class 4](#_Toc133162935)

[Figure 2 Histogram of variables Time and Length 5](#_Toc133162936)

[Figure 3 Bar chart of variable Airline 5](#_Toc133162937)

[Figure 4 Bar chart of variable Day of week 6](#_Toc133162938)

[Figure 5 ROC and AUC for all four models 8](#_Toc133162939)

[Figure 6 Feature importance 9](#_Toc133162940)

[Figure 7 Scree plot for k-means 11](#_Toc133162941)

[Figure 8 Data colored by the results from k-means 11](#_Toc133162942)

[Figure 9 Delayed and not delayed flights within clusters 12](#_Toc133162943)

[Figure 10 Dendrogram for average clustering 13](https://vse-my.sharepoint.com/personal/laum02_vse_cz/Documents/Studium/Data/4.%20semestr/4IZ171%20Strojové%20učení%201%20(v%20angličtině)/team/report.docx#_Toc133162944)

[Figure 11 Dendrogram for complete clustering 13](https://vse-my.sharepoint.com/personal/laum02_vse_cz/Documents/Studium/Data/4.%20semestr/4IZ171%20Strojové%20učení%201%20(v%20angličtině)/team/report.docx#_Toc133162945)

[Figure 12 Dendrogram for single clustering 13](#_Toc133162946)

# Tables

[Table 1 First five rows of the analyzed dataset 4](#_Toc133162947)

[Table 2 Classification report on the random forest with 5 % of training data 7](#_Toc133162948)

[Table 3 Cost matrix 8](#_Toc133162949)

[Table 4 Made-up flight 9](#_Toc133162950)

[Table 5 Overview of Inertia for k-means 10](#_Toc133162951)

[Table 6 Comparison of clustering methods 14](#_Toc133162952)

[Table 7 Requirements for the assignment 14](#_Toc133162953)