The Spies Among Us

Predictive Methods of Data Mining

2021/2022
2nd Semester

Ana Francisca Dias m20211085
Ana Rita Silva m20200899
André Oliveira m20211253
Raquel Jessen Machado m20211092

Professors:
Roberto Henriques
Carina Albuquerque
Lara Oliveira
Carlotta Lehman
Ana St. Aubin

Index

Introduction and Methodology

Within the scope of the Predictive Methods of Data Mining course, we were proposed to carry out a project taking on the role of data scientists.

The basis of the project is set on the assumption that the USA president needs to obtain a solution for finding the spies responsible for steeling information without compromising the privacy of USA citizens. As the minister of defence already, based on previous data, discovered that there are some groups of people that are more likely to be spies than others, our team was hired to predict which citizens should be placed under close surveillance using a set of pre-selected variables accessible from the last year's citizen database. Previously detected cases of espionage have been meticulously registered in the general citizen database, where all citizens are anonymously registered for statistical purposes, and which is updated yearly. As data scientists, your team is asked to analyse and transform as needed the data available and apply different models to answer the defined question in the most accurate way.

The dataset given to the team contains the following variables:

Table 1 - Description of each variable.

Variable	Description
ID	-
ID_ORIGINAL	The unique identifier of the citizen
Gender	The gender of the citizen
Age	The age of the citizen
Area_Residence	Area of residence
Household_Income	The household income
Household_Size	Number of people in the citizen's household including himself
Foreign_Citizenship	If the citizen has foreign citizenship
Frequent_Transfer	If the citizen is a frequent traveller (travels more than twice a year)
Social_Person	If the citizen is seen as social
Cellphone_Usage	Cell phone usage level of the citizen
Occupation	The type of occupation the citizen has (if employed the type of employer is mentioned – self-employed, government, private or public company)
Military_Service	If the citizen has completed military service
Political_Participation	Political participation level (considering activities such as voting in elections, starting and participating in petitions, etc.)
Satisfaction_Level	The satisfaction level with the standard of living in the country
Spy	If the citizen has been identified as a spy

Exploration and Understanding

The purpose of this chapter is to apply a combination of numerical and visualization techniques to summarize the data and provide understanding of the dataset. This section includes the types of variables attributes or variables that makeup the dataset, treatment of missing values and outliers and statistical analysis.

According to the information received, there are 8 000 citizens registered in the train dataset and 493 citizens in the teste dataset. The type of variable and its quantity are shown in table 2:

Table 2 - Type and quantity of each variable in test and train dataset.

Variable	Type of variable	Train Dataset Quantity	Test Dataset Quantity
ID	ID	8 000	493
ID_ORIGINAL	ID	8 000	493
Gender	Nominal	8 000	493
Age	Nominal	8 000	493
Area_Residence	Nominal	7 924	485
Household_Income	Continuous	8 000	457
Household_Size	Nominal	7 670	476
Foreign_Citizenship	Binary Symmetric	7 862	486
Frequent_Transfer	Binary Symmetric	7 924	490
Social_Person	Binary Symmetric	7 924	485
Cellphone_Usage	Nominal	8 000	493
Occupation	Nominal	7 876	489

Military_Service	Binary Symmetric	7 924	485
Political_Participation	Nominal	7 876	489
Satisfaction_Level	Ordinal	7670	476

Exploring the test and train dataset variables, it was possible to conclude some aspects in each dataset.

Variable: Gender

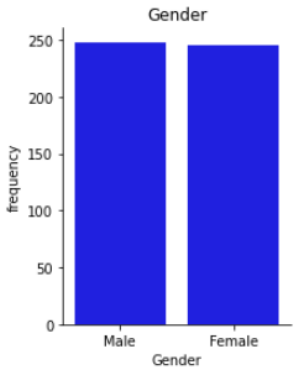

Figure 1 - Bar chart Foreign Gender Data test

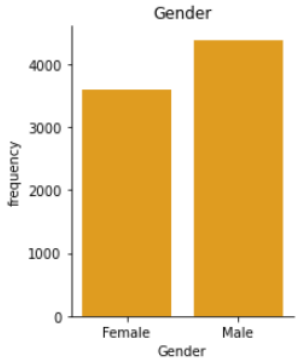

Figure 2 - Bar chart Foreign Gender Data train

In the test and train dataset the percentage of male and female citizens is about 50% each (figure1 and figure 2), which reveals that the variable "gender" won't be a good variable for the predictive model.

Variable: Occupation

The variable occupation characterizes the citizen employment according to the following categories: "Private company", "Government", "self-employment", "Public company, "Student"" or "Nothing"

As figure 3 and 4 show that both test and train datasets present that 70% of the citizens occupation is in a private company as the remaining citizens are divided into the remaining categories with hardly any difference.

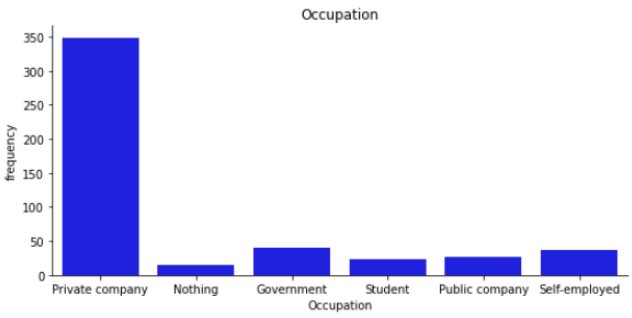

Figure 3 - Bar chart Occupation data test

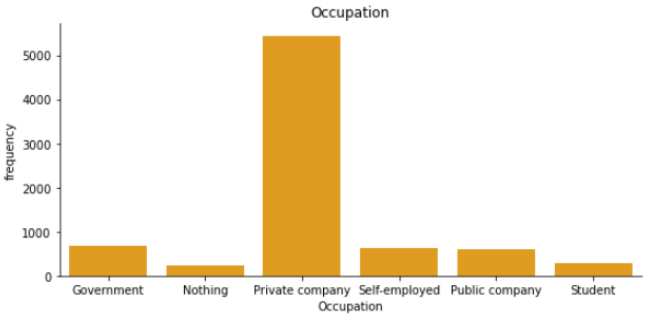

Figure 4 - Bar chart Occupation data train

Variable Political_Participation

The variable Political_Participation characterizes the citizens level of political participation in 4 categories: “No involvement”, “Some involvement”, “Strong involvement” and “Unknown” considering activities such as voting in elections, starting, and participating in petitions, and other political activities.
As the figures 5 and 6 show both in the test and train dataset, have around 70% of the citizens characterized as having no political involvement (36% and 39% in the test and train dataset respectively) or the political participation unknown (34% and 31% in the test and train dataset respectively).

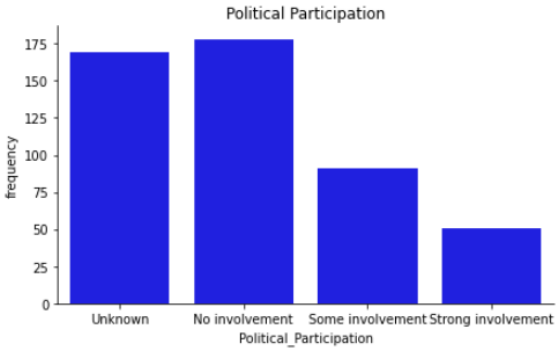

Figure 5 - Bar chart Political Participation data test

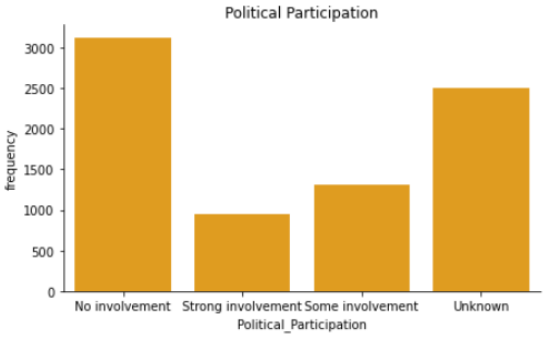

Figure 6 - Bar chart Political Participation data train

Variable: Area_Residence

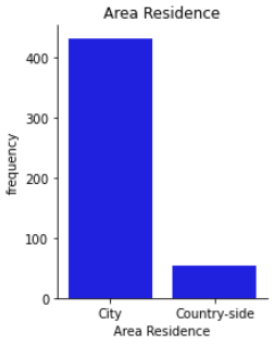

Figure 7 - Bar chart Area Residence Data test

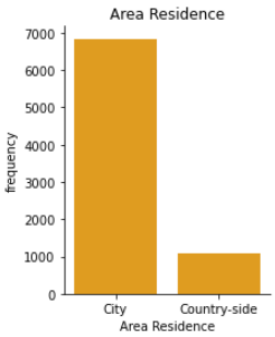

Figure 8 - Bar chart Area Residence Data train

The variable Area Residence characterizes the citizens by their home address in 2 categories: “City” or “Country-Side”. As the figures 7 and 8 show both in the test and train dataset show that around 87% of the citizens registered have their home address in the city.

Variable: Frequent traveler

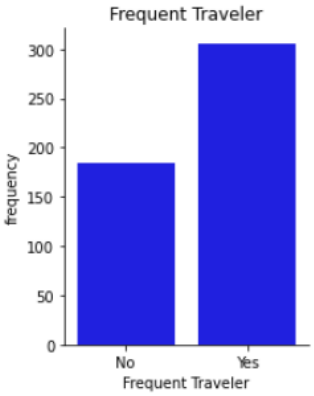

Figure 9 - Bar chart Frequent Traveler data test

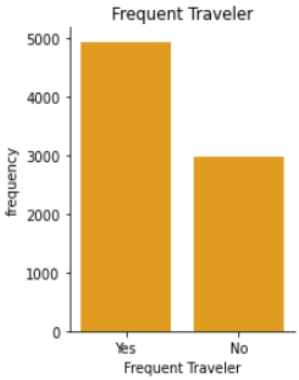

Figure 10 - Bar chart Frequent Traveler data train

The variable Frequent_Traveler identifies if the citizen is a frequent traveller or not. As the figures 9 and 10 show, 55% to 58% of citizens, in both test and train datasets, are frequent travellers.

Variable: Cellphone usage

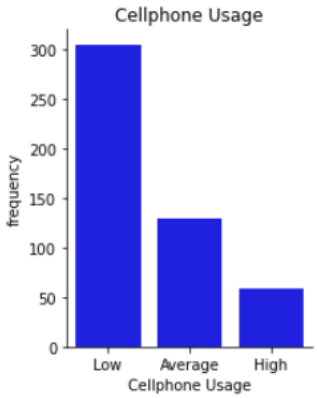

Figure 11 - Bar chart Cellphone Usage data test

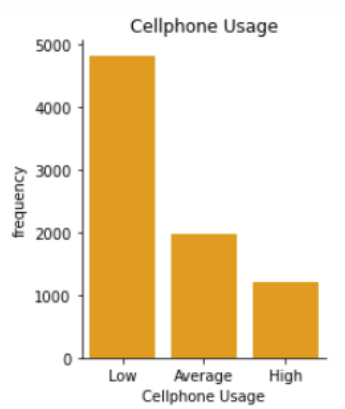

Figure 12 - Bar chart Cellphone Usage data train

The variable cellphone_usage identifies how often the citizen uses their cell phone. This variable is categorized in 3 levels: Low, Average and High.
As the figures 13 and 14 show, the majority of citizens has a low level of cell phone usage (~60%), followed by average (~25%).

Variable: Household size

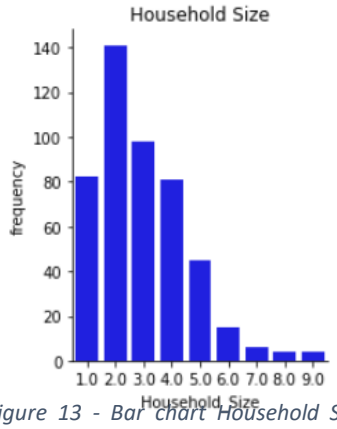

Figure 13 - Bar chart Household Size data test

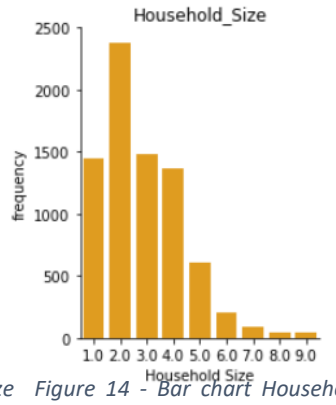

Figure 14 - Bar chart Household Size data train

The variable household_size identifies the number of people in the citizen's household including himself. As the figures 13 and 14 show, majority of the registered citizens has a between a 1 and 4 household size (~86%).

Variable: Foreign citizen

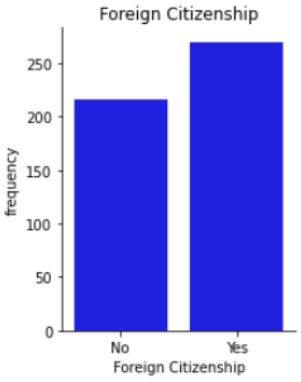

Figure 16 - Bar chart Foreign Citizenship Data test

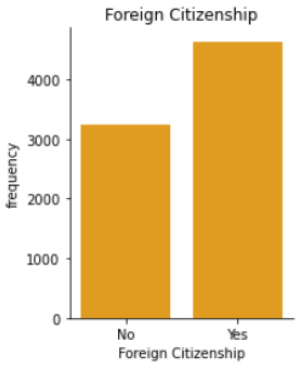

Figure 15 - Bar chart Foreign Citizenship Data train

As figure 15 and 16 show, there are ~55% of citizens with a foreign citizenship registered in both train and datasets.

Variable: Age

Both test and train datasets contain registrations of citizens between 18 and 88 years old. Analyzing the age variable, the figures 17 and 18, it possible to identify that there is a higher count of citizens with ages between 25 and 50 years old.

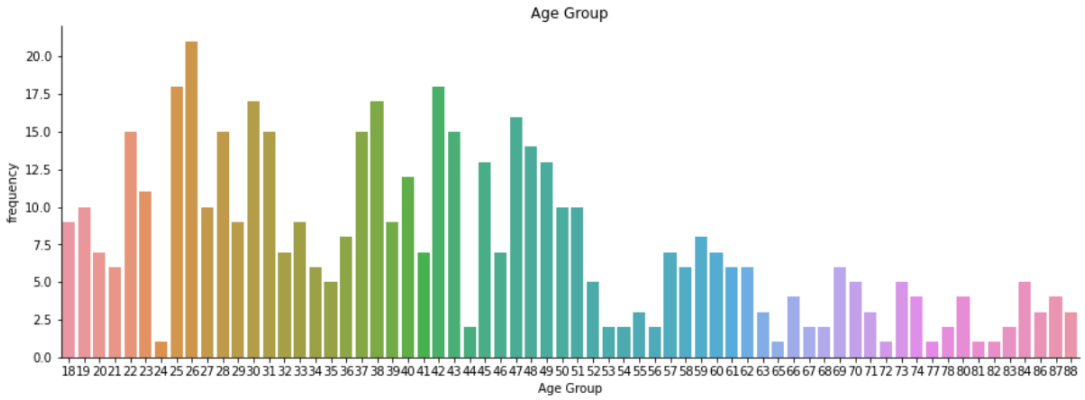

Figure 17 - Bar chart age Data test

Train

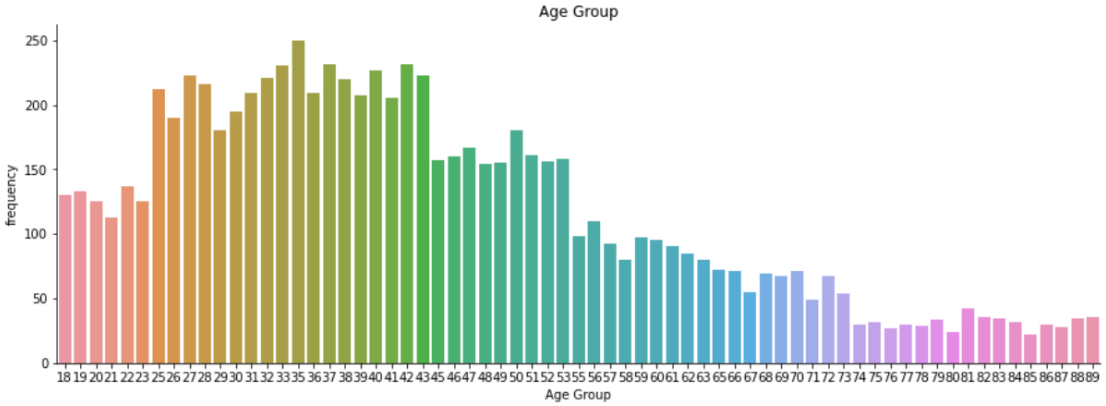

Figure 18 - Bar chart age Data train

Variable: Household_Income

The train and test data show a different in the values of the household_Income variable. As in the train dataset the majority of the registered citizens as a household income between 7.000€ and 8. 261€, in the test dataset the majority of the citizens has a household income between 0€ and 1.261€. This difference values between the datasets can lead to a faulty prediction model.

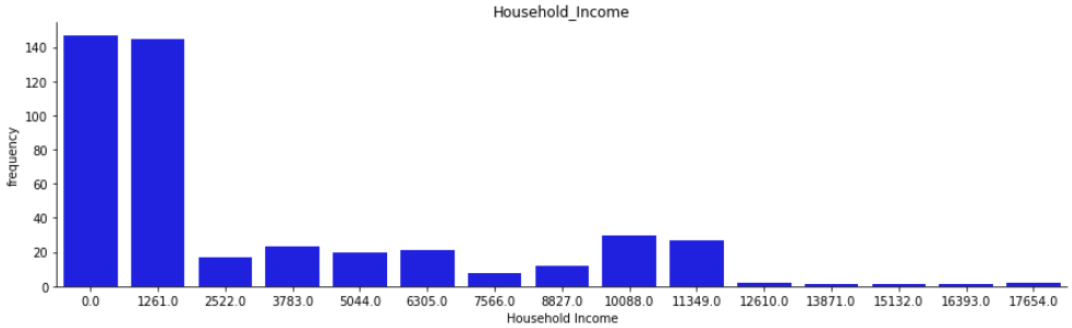

Figure 19 - Bar chart Household Income data test

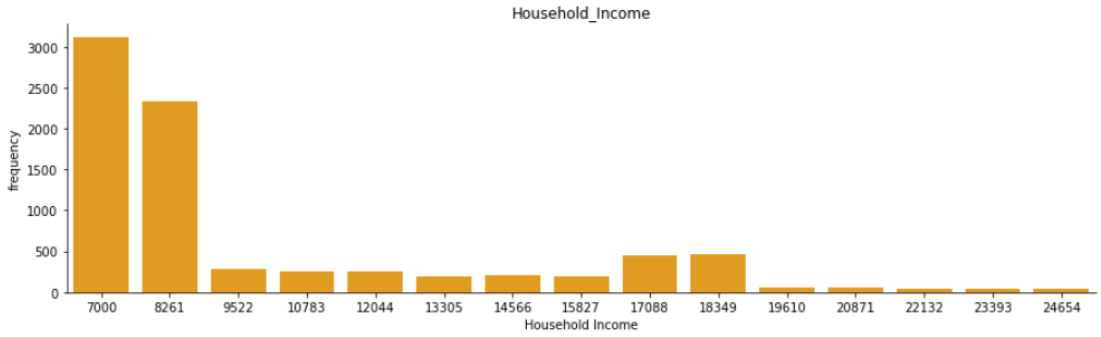

Figure 20 - Bar chart Household Income data train

Variable: Satisfaction Level

The variable satisfaction level represents the level of satisfaction of the citizens with the standard of living in the country. This variable is categorised between 9 levels of satisfaction.
As figures 21 and 22 demonstrate there is a slight difference between the datasets.
In the train set the majority of citizens has a satisfaction level of 1 as in the test dataset the majority of the citizens has a satisfaction level of 2. However, both datasets show that in average, the citizens have a low level of satisfaction of standard living conditions.

Figure 21 - Bar chart Satisfaction_Level data test

Figure 22 - Bar chart Satisfaction_Level data train

Initially, the basic descriptive statistics were plotted to get an overall view of the dataset, measuring the central tendency and the distribution of each attribute.

Table 3 - Summary statistics for all variables of the train dataset

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
ID	8000.0	NaN	NaN	NaN	4999.5	2309.54541	1000.0	2999.75	4999.5	6999.25	8999.0
ID_ORIGINAL	8000.0	NaN	NaN	NaN	463456.46225	2585.043875	458982.0	461226.75	463448.5	465716.25	467974.0
Gender	8000	2	Male	4392	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Foreign_Citizenship	7862	2	Yes	4632	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Age	8000.0	NaN	NaN	NaN	43.576625	16.704319	18.0	31.0	41.0	53.0	89.0
Frequent_Traveler	7923	2	Yes	4938	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Cellphone_Usage	8000	3	Low	4815	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Household_Size	7670.0	NaN	NaN	NaN	2.844329	1.527431	1.0	2.0	3.0	4.0	9.0
Spy	8000.0	NaN	NaN	NaN	0.477625	0.49953	0.0	0.0	0.0	1.0	1.0
Satisfaction_Level	7670.0	NaN	NaN	NaN	2.641851	1.711686	1.0	1.0	2.0	4.0	9.0
Occupation	7876	6	Private company	5444	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Political_Participation	7876	4	No involvement	3125	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Social_Person	7924	2	No	5211	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Area_Residence	7924	2	City	6838	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Military_Service	7924	4	Never	7537	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Household_Income	8000.0	NaN	NaN	NaN	9985.575125	4190.830154	7000.0	7000.0	8261.0	12044.0	24654.0

Table 4 - Summary statistics for all variables of the test dataset

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
ID	493.0	NaN	NaN	NaN	9246.0	142.461106	9000.0	9123.0	9246.0	9369.0	9492.0
ID_ORIGINAL	493.0	NaN	NaN	NaN	466423.752535	5122.591784	459005.0	462368.0	465546.0	469232.0	477959.0
Gender	493	2	Male	248	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Foreign_Citizenship	486	2	Yes	270	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Age	493.0	NaN	NaN	NaN	42.432049	16.949769	18.0	29.0	40.0	51.0	88.0
Frequent_Traveler	490	2	Yes	306	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Cellphone_Usage	493	3	Low	305	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Household_Size	476.0	NaN	NaN	NaN	2.955882	1.592134	1.0	2.0	3.0	4.0	9.0
Satisfaction_Level	476.0	NaN	NaN	NaN	2.493697	1.59933	1.0	1.0	2.0	3.0	9.0
Occupation	489	6	Private company	349	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Political_Participation	489	4	No involvement	178	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Social_Person	485	2	No	305	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Area_Residence	485	2	City	432	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Military_Service	485	4	Never	426	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Household_Income	457.0	NaN	NaN	NaN	3123.527352	3948.088253	0.0	0.0	1261.0	5044.0	17654.0

Both household_size and age variables are highly correlated with satisfaction_level variable (figure 23 and 24). Due to this high correlation, producing ratios and combined features of this variables can lead to improvement of the model performance.

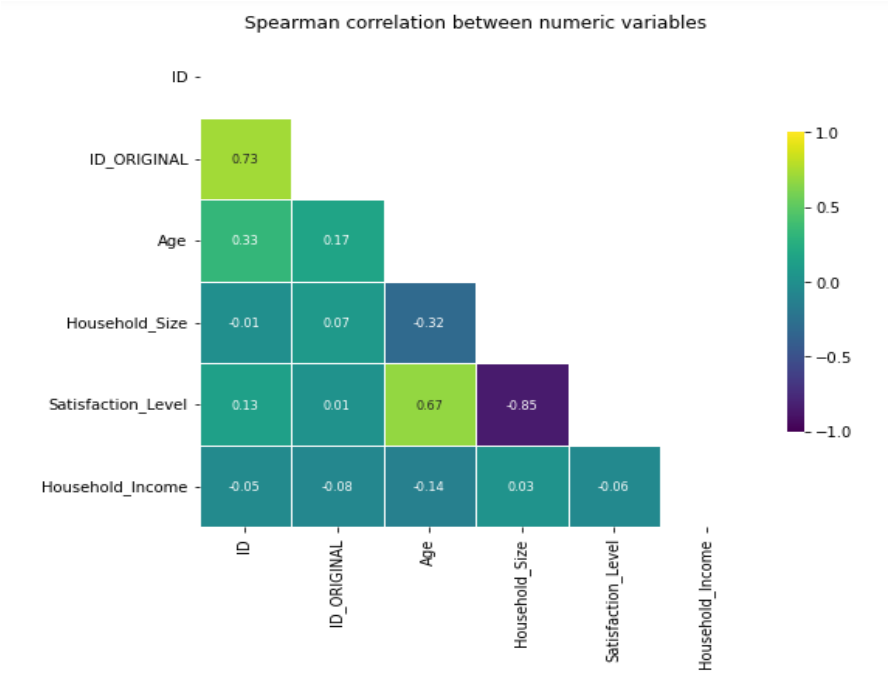

Figure 23 - correlation matrix data test

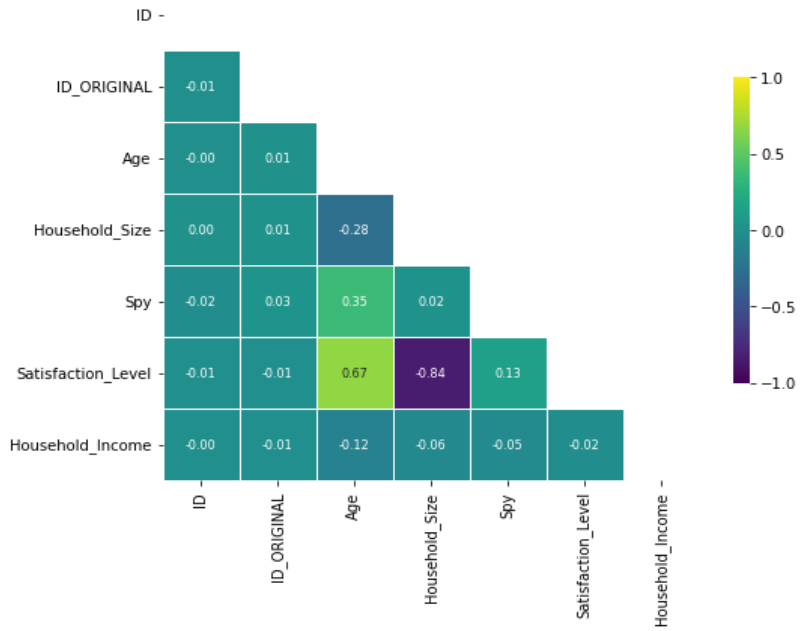

Figure 24 - Correlation matrix data train

Verify data quality

Data quality is a crucial part of the process, as data quality directly impacts the final output. It involves cleaning, filtering, and integrating data, to produce a high-quality dataset to be used on the modelling stage.

Missing values

The preliminary assessment has shown that there are missing values on some of the features of the test and train dataset. Table 5 below, represents the number of missing values in Test and Train Dataset.

Table 5 – Missing Values

Variable	Test Dataset Missing values	Train Dataset Missing Values
ID	0	0
ID_ORIGINAL	0	0
Gender	0	0
Age	0	0
Area_Residence	8	76
Household_Income	36	0
Household_Size	17	330
Foreign_Citizenship	7	138
Frequent_Traveler	3	77
Social_Person	8	76
Cellphone_Usage	0	0
Occupation	4	124
Military_Service	8	76
Political_Participation	8	124
Satisfaction_Level	17	330

Pre-processing

The pre-processing phase is responsible for preparing the data for modelling, and to achieve that we must clean and transform the raw data. We proceeded to clean the data by checking for missing values, duplicates and outliers. Afterwards we performed transformations in the data such as binning, feature selection and scaling.

Data Cleaning

Missing values

In the data understanding and data exploration phase it was possible to detect the existence of missing values on some of the features of the test and train dataset. When analysing in detail the variables with missing values, the full picture and treatment applied can be summarized as present in table 6:

Table 6 - Test and Train Dataset Missing Values : number, percentage and treatment.

Variable	Test Dataset Missing values		Train Dataset Missing Values		Missing Values treatment
	Number	Percentage	Number	Percentage	
ID	0		0	-	Not applicable
ID_ORIGINAL	0		0	-	Not applicable
Gender	0	-	0	-	Not applicable
Age	0	-	0	-	Not applicable
Area_Residence	8	1,62%	76	0,95%	As the variable is categorical, the missing values were replaced by “unknown”

Household_Income	36	7,3%	0	-	The missing values were replaced by "0" being the minimum income salary possible.
Household_Size	17	3,45%	330	4,13%	The missing values were replaced by "1", being the minimum number of people in the household including himself
Foreign_Citizenship	7	1,42%	138	1,73%	As the variable is categorical, the missing values were replaced by "unknown"
Frequent_Traveler	3	0,61%	77	0,96%	As the variable is categorical, the missing values were replaced by "unknown"
Social_Person	8	1,62%	76	0,95%	As the variable is categorical, the missing values were replaced by "unknown"
Cellphone_Usage	0	-	0	-	Not applicable
Occupation	4	0,81%	124	1,55%	As the variable is categorical, the missing values were replaced by "unknown"
Military_Service	8	1,62%	76	0,95%	As the variable is categorical, the missing values were replaced by "unknown"
Political_Participation	4	0,81%	124	1,55%	As the variable is categorical, the missing values were replaced by "unknown"
Satisfaction_Level	17	3,45%	330	4,13%	As the variable is numerical, the missing values were replaced by 0

There are different approaches to handle missing values, such as replacing them with the mean or median/mode for a random value (numerical/categorical variables) or using kNN Imputer (match a point with its closest k neighbours). After testing these options, we concluded that replacing for a constant like "unknown", "0" and "1", was better for the performance of the model.

Duplicates

We checked for duplicate values, but none were found.

Outliers

We proceeded to check the distribution of outliers through boxplots, however, we decided to not remove any outliers because the Kaggle submission required the same number of rows as the original train dataset. Also, comparing the outlier's ratio we concluded that the number of outliers wouldn't affect the performance of the model.

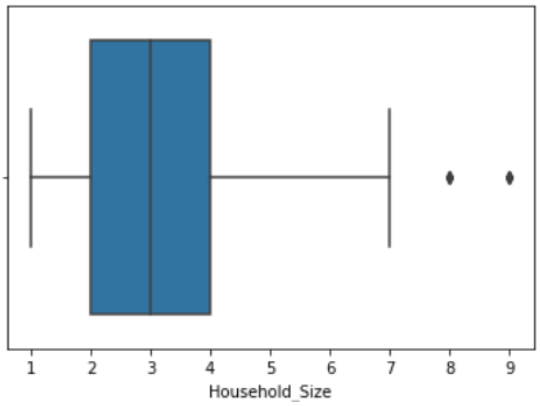

Figure 25 - Household_Size Outliers in Test Dataset

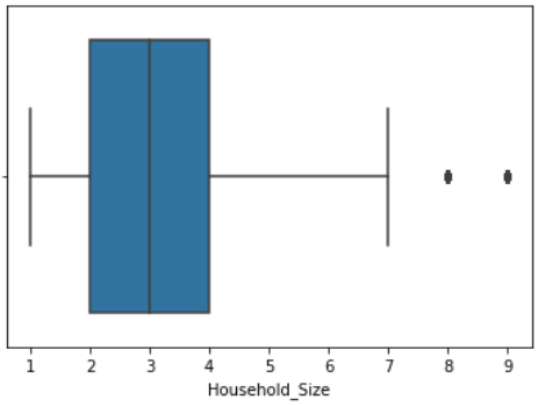

Figure 26 - Household_Size Outliers in Train Dataset

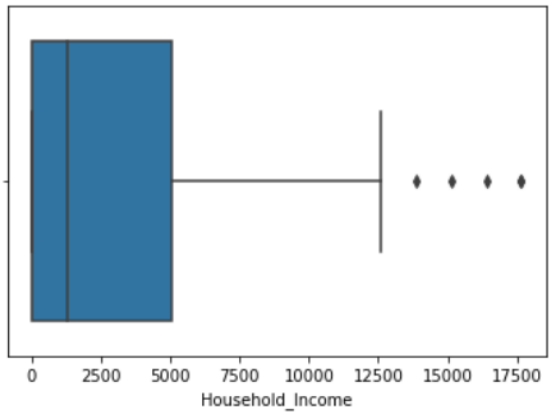

Figure 27 - Household_Income Outliers in Test Dataset

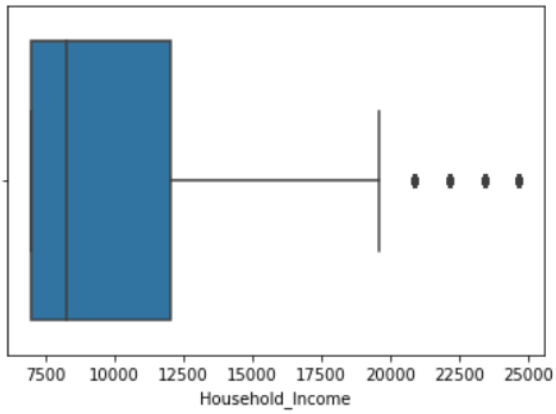

Figure 28 - Household_Income Outliers in Test Dataset

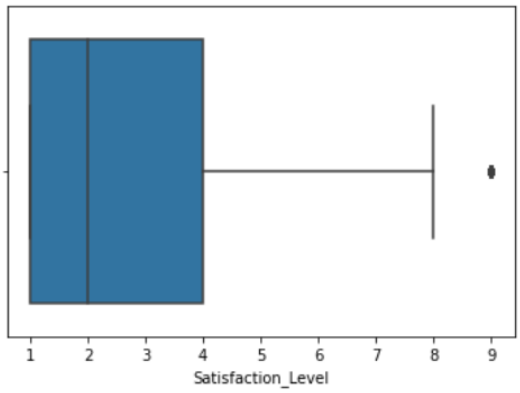

Figure 29 - Satisfaction_Level Outliers Train Dataset.

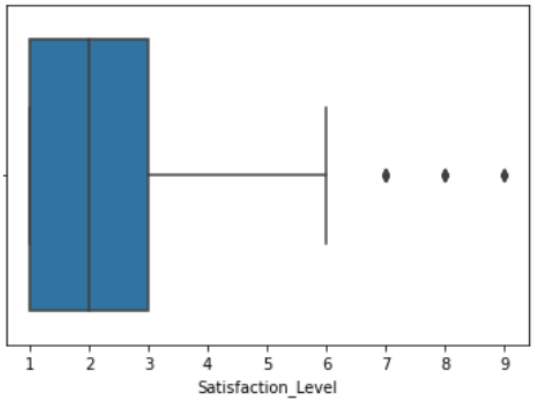

Figure 30 - Satisfaction_Level Outliers Test Dataset.

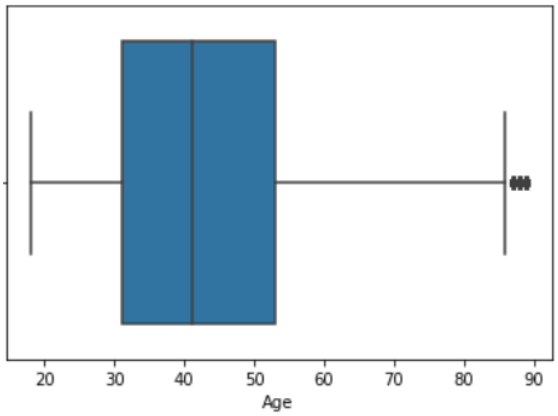

Figure 31 - Age Outliers Train Dataset.

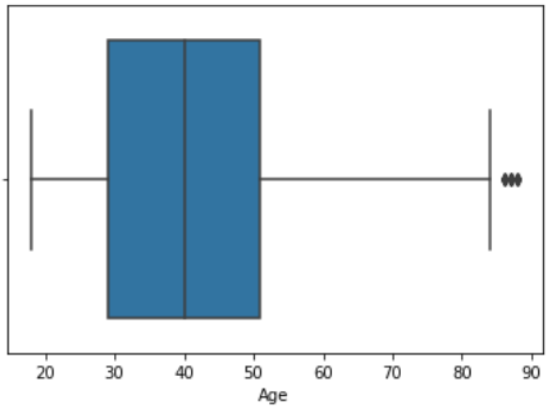

Figure 32 - Age Outliers Test Dataset.

Data Transformation

To prepare the data for the modelling, we opted to do label encoding, a good technique to handle categorical variables. Label encoding replaces categorical values with numerical values.

Binning

We opted to bin the quantitative variables (Age, Household_Income, Household_Size and Satisfaction_Level) in order to partition the numerical variables into bins.

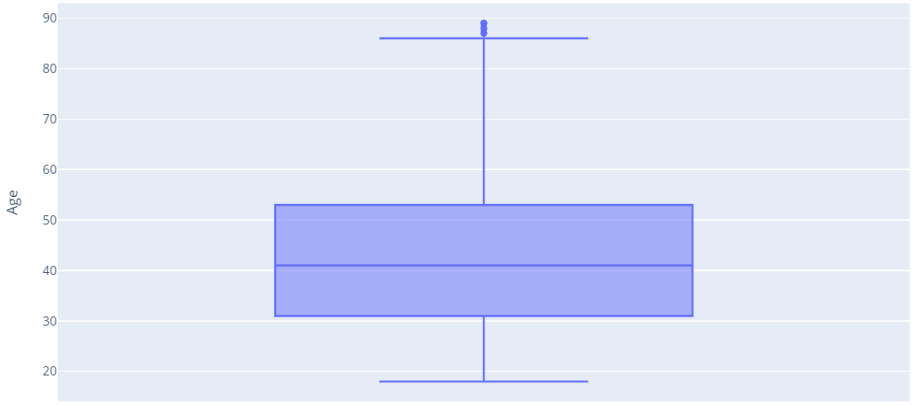

Figure 33 – Age Boxplot

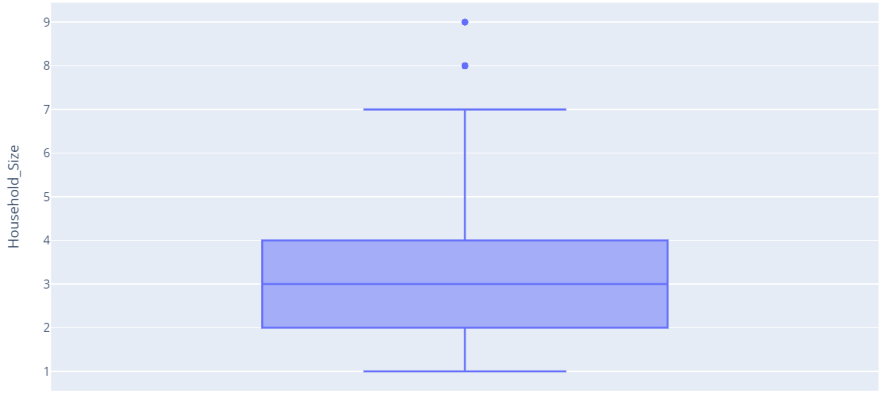

Figure 34 - Household_Size Boxplot

Figure 35 - Satisfaction_Level Boxplot

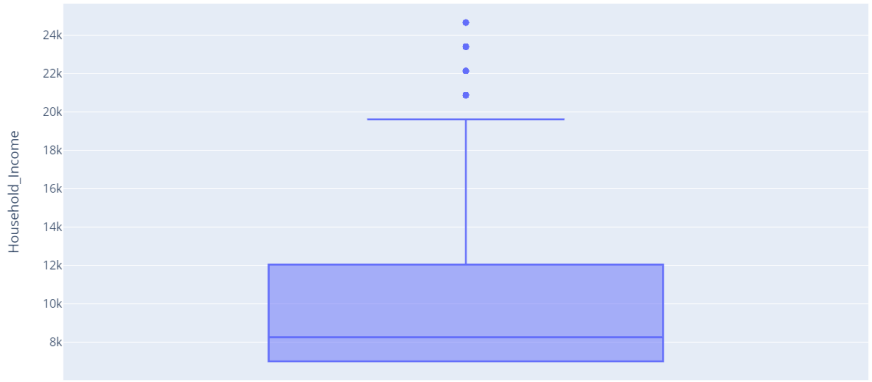

Figure 36 - Household_Income Boxplot

Feature Selection

In supervised learning methods, when preparing a large dataset for training, it is important to select the best features to include, and it is possible to that resorting to feature selection.

Feature selection is a pre-processing tool that is effective and efficient in improving the performance of a machine learning model, especially in cases of high dimensionality.

When dealing with large amounts of features (high dimensionality), learning models tend to overfit and that can reduce the performance of a model. Therefore, feature selection directly selects a subset of relevant features whilst building a model.

Resorting to this has a lot of advantages, such as: improvement of learning performance, increasing computational efficiency, decreasing memory storage, and constructing better models. [1]

The purpose of this step is to reduce the number of input variables to the most useful to the model, in order to predict the target variable, in this case, Spy.

Support Vector-Machine (SVM) and Neural Networks can be affected by not relevant predictors, therefore using irrelevant attributes can decrease the performance of the model.

Linear and logistic regressions can be sensible to correlated predictors, thus dealing with redundancy will help reduce multicollinearity and result in a better predictive model. [2]

SelectKBest

To implement feature selection, we opted for SelectKBest. SelectKBest is a method from feature selection, that will help select the features according to k highest score.

We used SelectKBest to score the features against the target variable using the function f_classify. Since we are dealing with quantitative attributes, f_classify computes the ANOVA F-value between each variable and the target vector. [3]

Analysis of Variance (ANOVA) is a statistical method used to check the means of two or more groups that are significantly different from each other.

Scaling

Due to existing some data mining algorithms affected by differences in the attribute's ranges, it is important to perform scaling before modelling. Differences in ranges can contribute differently to the model and affect its results, for example, variables with a greater variation in their range, can lead to a higher impact on the results. [4]

Since in this dataset we have variables measured at different scales, we performed normalization before modelling. The method we chose is Min-max Normalization and it is one of the most commonly used.

Modelling

Gradient Boosting

The selected model for this project was Light GBM – a fast, distributed, high-performance gradient boosting framework based on decision tree algorithm. [5] Gradient boosting is a type of machine learning boosting, a method for creating an ensemble that is a combination of simple individual models that together create a more powerful new model.

Boosting works by fitting an initial model (e.g. decision tree) to the data. Then a second model is built that focuses on accurately predicting the cases where the first model performs poorly. The combination of these two models is expected to be better than either model alone. The boosting process is repeated many times, as each successive model attempts to correct the shortcomings of the combined boosted ensemble of all previous models. [6]

Gradient boosting relies on the intuition that the best possible next model, when combined with previous models, minimizes the overall prediction error. The key idea is to set the target outcomes for this next model in order to minimize the error. The name gradient boosting arises because target outcomes for each case are set based on the gradient of the error with respect to the prediction. Each new model takes a step in the direction that minimizes prediction error, in the space of possible predictions for each training case. [6]

Hyperparameter Tuning

Hyperparameter tuning is the task of choosing a set of optimal hyperparameters for a machine learning algorithm. A hyperparameter is a model argument whose value is set before the learning process begins, as a result, different models have different hyperparameters to be set. In practice, the aim of hyperparameter tuning is to find the hyperparameters of a given machine learning algorithm that return the best performance as measured on a validation set. [7]

The selected method of hyperparameter tuning for this project was Bayesian Optimization. It is a sequential model-based optimization algorithm that uses the results from the previous iteration to decide the next hyperparameter value candidates. It works by creating a probabilistic model, mapping hyperparameters to a probability of a score on the objective function $P(score|hyperparameters)$. [8] This model is called a surrogate for the objective function and is represented as $P(x|y)$. The surrogate is much easier to optimize than the objective function, therefore, Bayesian methods work by finding the next set of hyperparameters to evaluate on the actual objective function by selecting hyperparameters that perform best on the surrogate function. [8]

This method is efficient because it selects hyperparameters in an informed manner. By prioritizing hyperparameters that appear more promising from past results, it can find the best hyperparameters in less time (fewer iterations) than both grid search and random search methods. [8]

Performance Assessment

In order to avoid data leakage, the test set has only been used to assess the model, while the train set has been split into 5 stratified cross validation folds (as the target feature is unbalanced). Each fold is divided into a train and validation subsets – allowing the gradient boosting model to be trained, while the validation subset is used to allow the Bayesian Optimization algorithm to search for the best hyperparameters, and the SelectKBest algorithm to select the right set of features. The final set of hyperparameters and features is based on the fold that presents better evaluation metric scores. Finally, the model is retrained automatically with all training data.

The evaluation metric selected to assess the model's performance was the F1-score, as it combines the precision and recall of a classifier into a single metric by taking their harmonic mean:

$$F1 - score = \frac{2 \times precision \times recall}{precision + recall}$$

In binary classification, precision attempts to assess the proportion of positive identifications that were actually correct, while recall estimates the proportion of actual positives that were identified correctly. The F1-score reaches its best value at 1 and worst value at 0. A low F1 score is an indication of both poor precision and poor recall and the other way around.

When analyzing Figure 37, we can see that the Bayesian Optimization algorithm selected the hyperparameters carefully to avoid overfitting – pruning the maximum depth of the decision trees to reduce complexity.

```
LGBMClassifier(learning_rate=0.01, max_depth=2,
               min_samples_leaf=49, min_samples_split=13,
               n_estimators=193, random_state=16,
               subsample=0.5658606452764399))])]
```

Figure 37 - Best model parameters

The final set of features to model – provided by the SelectKBest algorithm in Figure 38 – has selected 10 columns to model (support = True), including binned variables. The algorithm has filtered variables related with Gender, Household Size, Occupation and Military Service. The inclusion of binned variables has actually increased the model's performance significantly.

	support
Age	True
Household_Size	False
Satisfaction_Level	False
Household_Income	True
CD_Gender	False
CD_Foreign_Citizenship	True
CD_Frequent_Traveler	True
CD_Cellphone_Usage	True
CD_Occupation	False
CD_Political_Participation	True
CD_Social_Person	True
CD_Area_Residence	True
CD_Military_Service	False
CD_Household_Size_BIN	False
CD_Satisfaction_Level_BIN	True
CD_Household_Income_BIN	True
CD_Age_BIN	False

Figure 38 - Features included in the best model.

The model evaluation results in Table 7 show that the model is not overfitting, as the f-score on the train set is lower than on the test set, by a small gap. The model could potentially be underfitting, but when its complexity was increased, it didn't result on a better performance.

Table 7 - Model evaluation f1-score results.

Set	F1-score
Average F1-score Cross-validation	0.73862
Average F1-score Train Set	0.75010
F1-score Train Set by Class ([0 1])	[0.76278 0.73742]
Average F1-score Test Set	0.74956
F1-score Test Set by Class ([0 1])	[0.76388 0.73524]
F1-score Kaggle	0.72164

Table 8 - Model evaluation precision and recall results.

Set	
Precision Train Set by Class ([0 1])	[0.75847458 0.74208861]
Recall Train Set by Class ([0 1])	[0.76714286 0.7328125]
Precision Test Set by Class ([0 1])	[0.75495751 0.74511401]
Recall Test Set by Class ([0 1])	[0.77302393 0.72561459]

The performance on the Kaggle competition set has been worse than on the test set by a wide margin, indicating that the model cannot capture every pattern perfectly. As presented in Table 5, the f1 score of spies (class 1) is always by far lower than in the case of non-spies (class 0), showing that the model cannot define a spy as well as it can distinguish a non-spy.

The root cause of this model behavior can be derived from Table 8, we the values of precision and recall by are presented. The precision values are quite balanced by class, indicating that the model is correctly categorizing the positive identifications, in practice, it is correct around 74% of the time when it predicts a person is a spy. However, the recall values are far lower categorizing spies than non-spies, indicating that the model struggles to correctly identify the actual positives, in other words, it correctly identifies 77% of all non-spies but only 72% of all spies.

Conclusions

During the modelling phase, we we're able to determine, through various attempts (trial and error), that some processes helped the model to be more effective and others didn't.

We concluded that using the Min-max feature scaling was more effective than standard scaling or the use of no scaling at all.

To avoid overfitting, the use of Bayesian Optimization hyperparameter tuning turned out to be effective. Cross validation allowed to improve the model's final score effectively.

We also decided to bin the features in the model and that has proven to be more effective, allowing the gradient boosted decision trees to have more data.

After trying both, we concluded that SelectKBest has revealed to be more effective than Recursive Feature Elimination (RFE) to perform feature selection.

The potential use of correlations to produce ratios and combined features, as we mentioned in the data understanding phase, turned out to not be effective in improving the model's performance, so we discarded it.

Resorting to sampling techniques, such as undersampling or oversampling, to balance uneven datasets has also proven not to be very effective. We considered this happened, mostly because there are many unbalanced features in the dataset – and on the current project this was only tested to balance the target feature (Spy).

Throughout the modelling phase, different models were tested such as kNN, SVC, Naive Bayes, Logistic Regression and Ada Boost, but gradient boosting models were more effective due to class imbalance in the data.

On a final note, we consider that implementing deep learning techniques, such as Neural Networks, could have been a powerful way to improve the final score.

Also, it would be beneficial to include more data from external sources, to classify a spy and therefore improving the model's results.

References

[1] J. Li, K. Cheng and S. Wang, "Feature Selection: A Data Perspective," p. 45, 2017.

[2] "Sklearn.feature_selection.SelectKBest.," [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html. [Accessed 12 06 2022].

[3] "ANOVA F-value For Feature Selection," [Online]. Available: https://chrisalbon.com/code/machine_learning/feature_selection/anova_f-value_for_feature_selection/. [Accessed 12 06 2022].

[4] "Everything you need to know about Min-Max normalization: A Python tutorial," [Online]. Available: https://towardsdatascience.com/everything-you-need-to-know-about-min-max-normalization-in-python-b79592732b79. [Accessed 12 06 2022].

[5] "LightGBM's Documentation," June 2022. [Online]. Available: https://lightgbm.readthedocs.io.

[6] "Gradient Boosting Explained – The Coolest Kid on The Machine Learning Block," [Online]. Available: https://www.displayr.com/gradient-boosting-the-coolest-kid-on-the-machine-learning-block/. [Accessed June 2022].

[7] "Hyperparameter Optimization," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Hyperparameter_optimization. [Accessed June 2022].

[8] W. Koehrsen, "A Conceptual Explanation of Bayesian Hyperparameter Optimization for Machine Learning," Towards Data Science, [Online]. Available: https://towardsdatascience.com/a-conceptual-explanation-of-bayesian-model-based-hyperparameter-optimization-for-machine-learning-b8172278050f. [Accessed June 2022].