Project 2

Hubert Bujakowski
Mikołaj Gałkowski
Julia Przybytniowska

June 2024

1 Introduction

The goal of the project was to build a predictive model to identify customers likely to take advantage of a bank's marketing offer, balancing high accuracy and parsimony. Given the dataset of 5000 training records with 500 anonymized variables, our task was to train a model to predict which customers in a test set would respond positively. The company's campaign capacity is limited to 1000 customers, necessitating precise and efficient model predictions.

2 Model and Method Selection

We selected the following models for comparison: `Random Forest`, `Support Vector Machine (SVM)`, `XGBoost`, `CatBoost` and `LightGBM`.

We compared these models using the full set of features and evaluated various feature selection methods, experimenting with different threshold values to determine the optimal number of features to include. The feature selection methods included: `Random Forest Feature Importance`, `Recursive Feature Elimination (RFE)`, `XGBoost Feature Importance`, `SelectKBest with F-classif` and `SelectKBest with mutual_info_classif`.

Evaluation

To evaluate the models, we used a custom scorer function designed to balance accuracy and variable cost, maximizing the net financial gain for the company.

Scorer Function Components

In our experiments, we employed cross-validation to ensure comparable results. To standardize our findings, we scaled the predictions as if there were 1,000 potential clients for the 1,000 offers.

Accuracy Reward:
- €10 for each correctly identified customer who takes advantage of the offer.
- Calculation: Total Reward = €10 × Number of Correct Predictions

Variable Cost:
- €200 for each variable used in the model.
- Calculation: Total Cost = €200 × Number of Variables Used

Score:
- The overall effectiveness of the model.
- Calculation: Score = Total Reward - Total Cost

From the initial experiments, we identified which model and method performed the best. We then conducted a grid search on this selected model and method to optimize performance with the chosen features. Following this, we calibrated our classification model to ensure that the predicted probabilities accurately represented the true distribution. This calibration allowed us to reliably select 1000 clients to whom we should propose the offer.

3 Experiments

Before conducting our experiments, we performed preprocessing steps such as dropping highly correlated features (those with a Pearson Correlation coefficient greater than 0.8) and applying MinMax scaling.

Next, we established thresholds for the number of features to be used, ranging from 1 up to 50 (excluded). We determined that using more than 50 features would be unnecessary, as it would incur a penalty cost. Specifically, using 50 features would cost us €10000 (€200 per feature × 50 features), which equals the best possible earning from targeting the proper clients (1000 clients × €10 per client = €10000). Experiments were conducted with 10-fold cross-validation. Table 1 shows the results from these experiments. As we can see, all top 20 experiments used `Random Forest Feature Importance` as the feature selection method. The best performing model was `Random Forest` with a mean score of 5867.24 and a standard deviation of 340.599. The second best was `SVM` with a mean score of 5831.12 and a lower standard deviation of 251.325.

Model name	Feature Selection Method	No Features	Mean Score	Std Score
Random Forest	Random Forest Feature Importance	6	5867.24	340.599
SVM	Random Forest Feature Importance	6	5831.12	251.325
LightGBM	Random Forest Feature Importance	6	5779.05	370.178
Random Forest	Random Forest Feature Importance	7	5751.31	268.989
Random Forest	Random Forest Feature Importance	5	5706.57	266.59
Random Forest	XGBoost Feature Importance	7	5691.34	213.911
SVM	Random Forest Feature Importance	7	5671.21	243.37
SVM	XGBoost Feature Importance	7	5643.23	348.501
Random Forest	XGBoost Feature Importance	6	5606.83	265.905
SVM	Random Forest Feature Importance	8	5587.42	287.586
CatBoost	Random Forest Feature Importance	6	5570.63	321.697
LightGBM	Random Forest Feature Importance	5	5566.62	340.393
SVM	Random Forest Feature Importance	5	5566.49	298.1
LightGBM	XGBoost Feature Importance	7	5546.8	343.988
CatBoost	Random Forest Feature Importance	5	5534.28	362.008
Random Forest	Random Forest Feature Importance	8	5531.18	221.124
Random Forest	XGBoost Feature Importance	8	5523.26	325.195
LightGBM	Random Forest Feature Importance	7	5494.83	283.061
SVM	XGBoost Feature Importance	6	5466.78	324.042
Random Forest	Random Forest Feature Importance	9	5415.52	272.847

Table 1: Performance of models with different number of features and Feature Selection Methods - only 20 best performing experiments shown (more results available in the Appendix A).

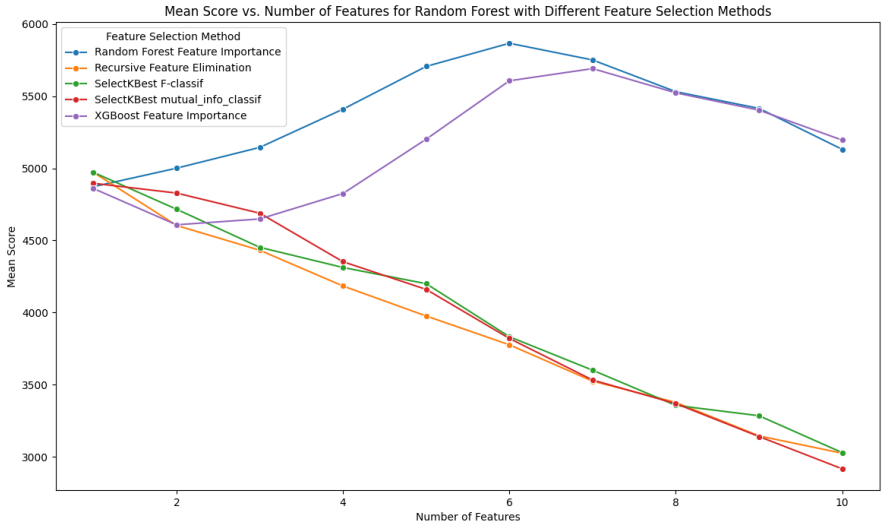

Figure 1: Mean Score vs. Number of Features for Random Forest with Different Feature Selection Methods

In figures 1, 2 shown results focused exclusively on the Random Forest model. It is evident that Random Forest Feature Importance outperforms other feature selection methods. Additionally, the experiment did not need to include more than 10 features, as the results beyond this point were not satisfactory.

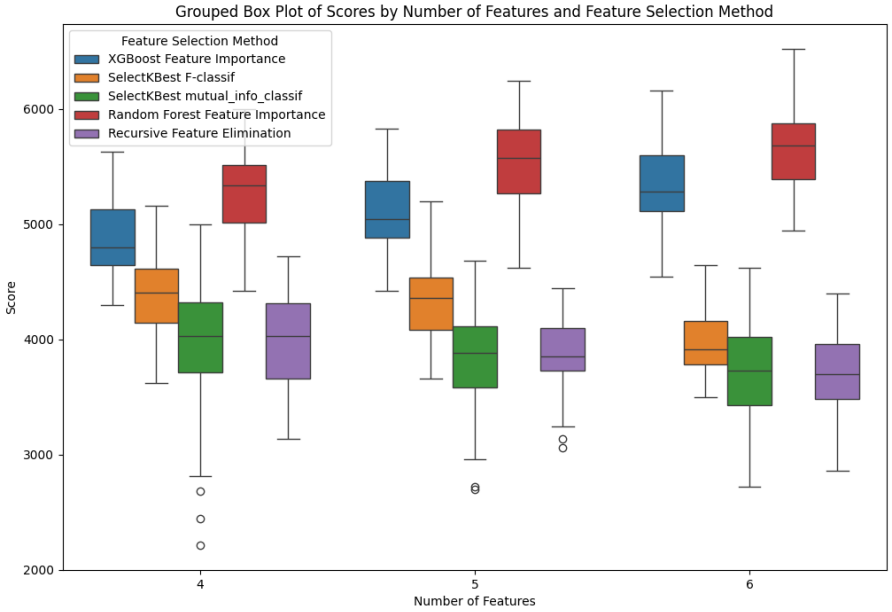

Figure 2: Grouped box plots of scores by number of features and feature selection method only for RandomForest model.

Our next step involved hyperparameter optimization for both RandomForest and SVM classifiers. We selected the top 6 features using the Random Forest Feature Importance method. We then conducted a grid search with 5-fold cross-validation over the parameter grids presented in Tables 2 and 3.

Parameter	Values
n_estimators	[90, 100, 110, 125, 140, 150, 160, 175, 200, 210, 225, 250]
max_features	['sqrt', 'log2']
class_weight	['balanced_subsample', 'balanced', None]
min_samples_split	[2, 3, 4, 5, 6, 7, 8, 10]
min_samples_leaf	[1, 2, 4]

Table 2: Random Forest Parameter Grid

Parameter	Values
C	[0.1, 1, 10, 100]
kernel	['linear', 'poly', 'rbf', 'sigmoid']
gamma	['scale', 'auto']

Table 3: SVC Parameter Grid

Table 4 presents the results of the hyperparameter tuning. We decided to stay with the `RandomForest` classifier.

Model	GridSearch Result
Random Forest	€6123.7275
SVC	€6083.5800

Table 4: GridSearch Results for Random Forest and SVC

In order to accurately select 1000 clients to whom we should propose the offer (predict 1), we decided to perform a calibration process on the fitted `RandomForest` model using cross-validation with `CalibratedClassifierCV`. This calibration aims to produce reliably represented probabilities, which may simulate the underlying distribution of the `y` variable.

4 Conclusion

This project encompassed data preprocessing, model training, variable selection, comparative analysis of feature selection methods, and fine-tuning to develop an optimal predictive model. The primary objective was to deliver accurate and cost-effective predictions to enhance the company's marketing strategy. A key challenge was selecting the appropriate number of features to maximize the company's earnings while minimizing any significant loss in precision.

A Experiment results for all models and feature selection methods - 50 best results

Model name	Feature Selection Method	No Features	Mean Score	Std Score
Random Forest	Random Forest Feature Importance	6	5867.24	340.599
SVM	Random Forest Feature Importance	6	5831.12	251.325
LightGBM	Random Forest Feature Importance	6	5779.05	370.178
Random Forest	Random Forest Feature Importance	7	5751.31	268.989
Random Forest	Random Forest Feature Importance	5	5706.57	266.59
Random Forest	XGBoost Feature Importance	7	5691.34	213.911
SVM	Random Forest Feature Importance	7	5671.21	243.37
SVM	XGBoost Feature Importance	7	5643.23	348.501
Random Forest	XGBoost Feature Importance	6	5606.83	265.905
SVM	Random Forest Feature Importance	8	5587.42	287.586
CatBoost	Random Forest Feature Importance	6	5570.63	321.697
LightGBM	Random Forest Feature Importance	5	5566.62	340.393
SVM	Random Forest Feature Importance	5	5566.49	298.1
LightGBM	XGBoost Feature Importance	7	5546.8	343.988
CatBoost	Random Forest Feature Importance	5	5534.28	362.008
Random Forest	Random Forest Feature Importance	8	5531.18	221.124
Random Forest	XGBoost Feature Importance	8	5523.26	325.195
LightGBM	Random Forest Feature Importance	7	5494.83	283.061
SVM	XGBoost Feature Importance	6	5466.78	324.042
Random Forest	Random Forest Feature Importance	9	5415.52	272.847
CatBoost	Random Forest Feature Importance	7	5414.71	252.9
Random Forest	Random Forest Feature Importance	4	5409.88	343.376
Random Forest	XGBoost Feature Importance	9	5403.31	288.292
LightGBM	Random Forest Feature Importance	8	5366.84	318.144
SVM	XGBoost Feature Importance	8	5363.07	353.849
XGBoost	Random Forest Feature Importance	5	5334.1	394.425
CatBoost	Random Forest Feature Importance	4	5333.85	350.965
CatBoost	XGBoost Feature Importance	7	5330.7	268.604
CatBoost	XGBoost Feature Importance	6	5330.43	319.834
LightGBM	XGBoost Feature Importance	6	5306.35	301.243
SVM	Random Forest Feature Importance	9	5291.31	262.875
CatBoost	Random Forest Feature Importance	8	5282.89	200.631
SVM	XGBoost Feature Importance	5	5262.14	303.968
XGBoost	Random Forest Feature Importance	6	5238.25	188.62
LightGBM	XGBoost Feature Importance	8	5206.73	243.31
XGBoost	Random Forest Feature Importance	4	5205.8	412.813
Random Forest	XGBoost Feature Importance	5	5201.91	356.373
Random Forest	XGBoost Feature Importance	10	5195.29	246.876
LightGBM	Random Forest Feature Importance	4	5193.56	307.175
CatBoost	XGBoost Feature Importance	8	5170.68	323.193
CatBoost	XGBoost Feature Importance	5	5153.94	302.718
Random Forest	Random Forest Feature Importance	3	5144.92	286.104
XGBoost	Random Forest Feature Importance	7	5142.46	366.044
XGBoost	SelectKBest mutual_info_classif	1	5140.63	180.485
Random Forest	Random Forest Feature Importance	10	5131.31	343.326
XGBoost	Random Forest Feature Importance	8	5110.65	274.246
SVM	Random Forest Feature Importance	4	5093.43	330.15
XGBoost	XGBoost Feature Importance	7	5066.25	330.092
SVM	XGBoost Feature Importance	4	5065.72	334.004
LightGBM	XGBoost Feature Importance	9	5062.94	246.978

Table 5: Performance of models with different number of features and Feature Selection Methods - only 50 best performing experiments shown - in total there has been conducted 1225 experiments (49 thresholds (ranging from 1 up to 49) $\times$ 5 feature selection methods $\times$ 5 models

B Plots representing results for each model and feature selection method

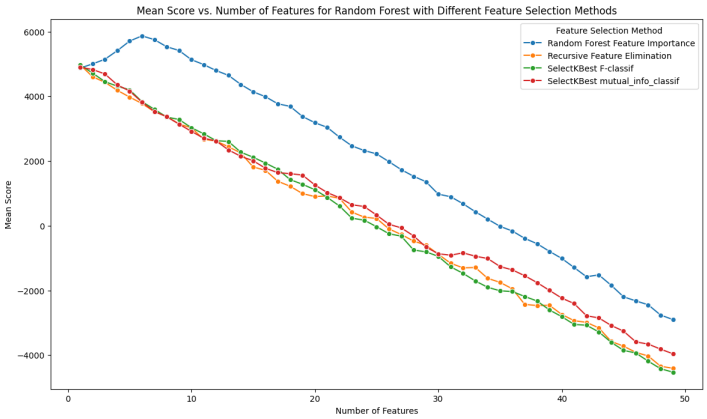

Figure 3: Mean Score vs. Number of Features for Random Forest with Different Feature Selection Methods

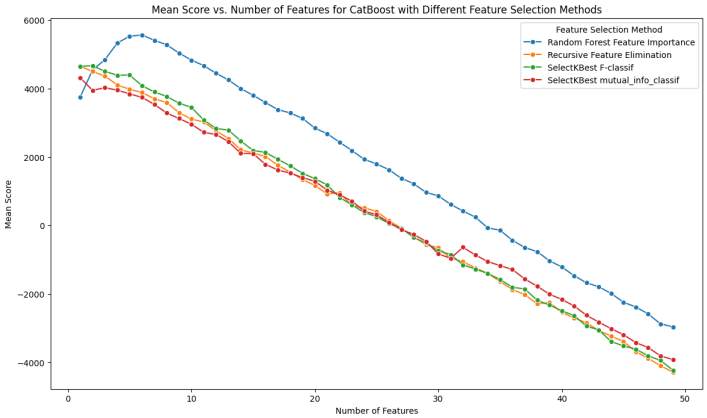

Figure 4: Mean Score vs. Number of Features for Catboost with Different Feature Selection Methods

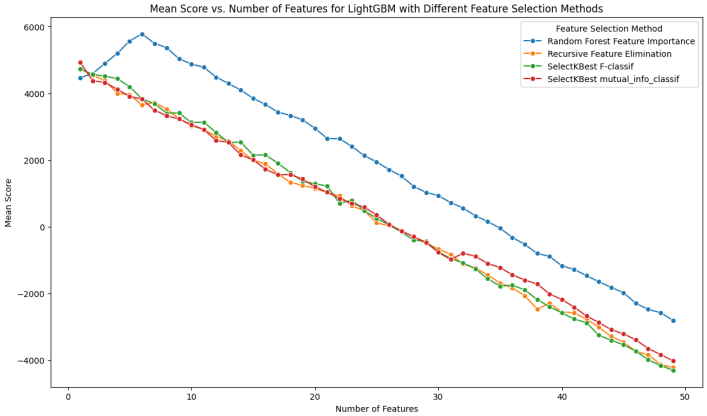

Figure 5: Mean Score vs. Number of Features for LightGBM with Different Feature Selection Methods

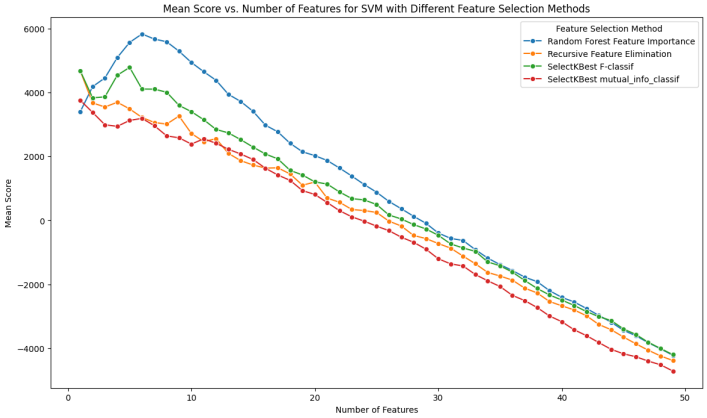

Figure 6: Mean Score vs. Number of Features for SVM with Different Feature Selection Methods

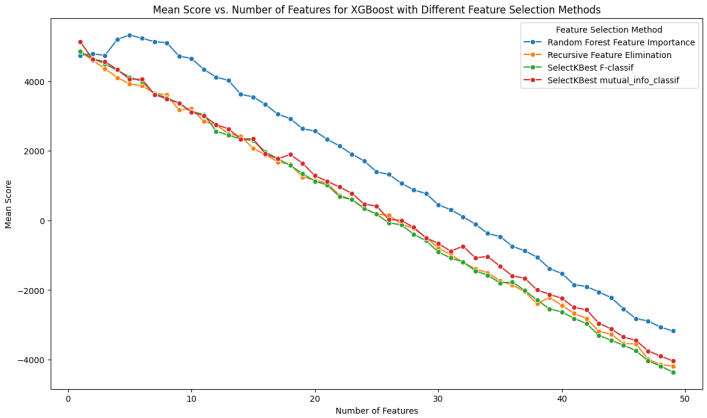

Figure 7: Mean Score vs. Number of Features for XGBoost with Different Feature Selection Methods