WARSAW UNIVERSITY OF TECHNOLOGY

FACULTY OF MATHEMATICS AND INFORMATION SCIENCE

Project 2 Report

Advanced Machine Learning

Karina Tiurina (335943)
Nikita Kozlov (317099)
Róża Klimek (329533)

supervisor
mgr Anna Kozak

Warsaw 2024

Introduction

The aim of this report is to describe conducted experiments on the task, compare the results, as well as explain the method chosen in the best solution.

1 Dataset exploration and preparation

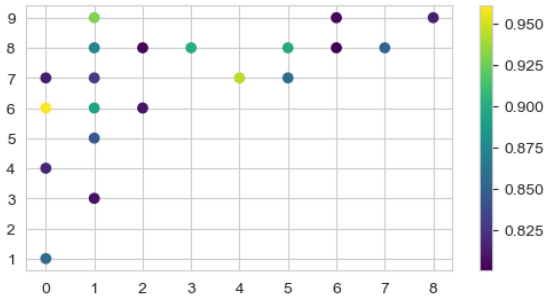

Figure 1: Features with high correlation in the dataset

The dataset has been explored and prepared using the following steps:

1. Missing values were checked - there are no missing values in the dataset
2. Class balance was checked - the dataset has 2 classes, they are balanced
3. Correlation in the dataset was checked - the dataset has 9 correlated features from 0 to 8, they were removed
4. VIF of features was checked - no features with high VIF were found
5. Dataset features were normalized

2 Feature Selection

2.1 Initial feature selection

We have selected a handful of different feature selection methods to try and find the most important features the dataset has that influence the target variable. For each feature selection method we have run the selection $iters = 50$ times and, after collecting all results, we chose features that have appeared in at least 90% of all iterations per each method.

Below is the table of the results of each feature selection method we chose to explore:

Method	Selected features (from 0 index)
RFECV with logistic regression estimator	200, 203, 204, 206, 210, 214, 215, 218, 220, 225, 227, 228, 240, 249, 250, 252, 253, 259, 266, 273, 277, 281, 283, 285, 288, 291, 296, 303, 306, 308, 309, 316, 317, 321, 322, 323, 324, 327, 328, 335, 339, 351, 356, 357, 360, 363, 369, 380, 387
Lasso	403
Boruta	8, 100, 101, 102, 103, 104, 105
mRMR	105, 64, 131, 136, 100, 102, 24, 103, 104, 359, 101, 29, 266, 57, 39, 241, 351, 155, 335, 442
ReliefF	105, 102, 101, 321, 100, 289, 283, 374, 254, 311, 380, 103, 339, 303, 315, 211, 462, 338, 208, 200
Random Forest	100, 101, 102, 103, 104, 105, 403
XGBoost	100, 101, 102, 103, 104, 105
Common features for non-linear methods	100, 101, 102, 103, 104, 105

Table 1: Feature selection methods and their selected features

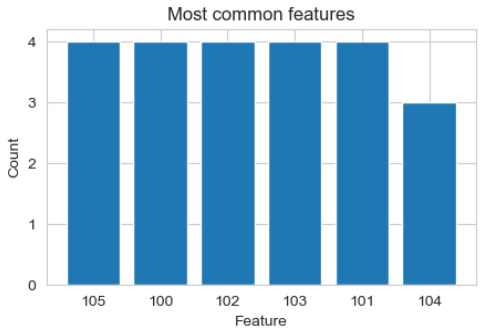

Figure 2: Most common features in the dataset

The following conclusions can be made about the results:

1. Linear feature selection methods, like LinearRegression estimator or Lasso, provide unstable results choosing different amount and indices of features.
2. Recursive feature elimination methods are unusable on scale with the dataset as their speed is very low for the dataset.
3. Non-linear feature selection methods tend to choose identical set of features (100-105) and some variations. Only results of these methods were chosen to be used in the next stages.

2.2 Models initial evaluation

After we have collected all feature selection methods results, most common features and a union of all results we have decided to "score" the selection of different models against our feature selections.

We have split the training dataset into 4000 training instances and 1000 validation instances. For additional results stability we have run each algorithm 10 times. For evaluation, we have used the scoring function proposed by the project guidelines, with 2 modifications:

1. The scoring was initially scaled to match the amount of positive instances in the validation dataset.
2. The punishment for feature use was nullified for this stage only.

Below are the results for the models on each selection method (bold are the chosen models for the next stages):

Method	Best result and method	Worst result and method
Random Forest	Features 100-105 (6690.229 ± 29.106)	All features (5869.023 ± 103.136)
Gradient Boosting	Features 100-105 (6667.360 ± 88.718)	All features (6232.848 ± 117.533)
Support Vector Machine	Features 100-105 (6860.707 ± 0.000)	All features (3819.127 ± 705.701)
K-Nearest Neighbors	Features 100-105 (6673.597 ± 0.000)	All Features (4490.644 ± 0.000)
QDA	Features 100-105 (6881.497 ± 0.000)	All features (4885.655 ± 0.000)
Multi-Layer Perceptron	F. 100-105, 403 (6216.216 ± 136.012)	All features (4519.751 ± 747.701)
SGD	F. 100-105 (4792.100 ± 175.242)	F. 100-105, 403 (224.532 ± 607.467)
Logistic Regression	All Features (5239.085 ± 0.000)	Features 100-105 (4677.755 ± 0.000)
Naive Bayes	Features 100-105 (6839.917 ± 0.000)	All features (5800.416 ± 0.000)

Table 2: Classification methods initial comparison

2.3 Genetic algorithm feature selection

We have chosen an additional approach for feature selection based on the idea of genetic algorithm, in which "genes" produce next generation using crossover and mutation, out of which the best genes are selected based on tournament for maximizing fitness function.

In our case, we have chosen a union of all results of non-linear feature selection methods as an initial population. Then every gene may either include or exclude a feature from the list. The target for the genes is to maximize the fitness obtained the following way:

1. Training dataset is split into 4000 training instances and 1000 validation instances. For 1000 validation instances max_score is obtained - a number of all positive target instances multiplied by 10.
2. Original scoring from the guidelines is applied (we add 10 points for each good guess and deduct 200 points for each feature used).
3. $fitness = score(customer\_selection)/max\_score$ - this is a target function that genetic algorithm should try to maximize

Below are the feature selection and fitness for each model we have chosen:

Model	Selected features (from 0 index)	Fitness
Random Forest	105, 100, 102, 103, 101, 241, 335, 200	0.5
SVM	105, 100, 102, 101	0.56
KNN	105, 100, 102, 104	0.57
LightGBM	105, 100, 102, 104, 101	0.57

Table 3: Model feature selection results and fitness

Our final feature selection is features 100 to 105, as they are the most common ones selected by every feature selection algorithm.

3 Final models comparison

We have chosen 4 models to participate in the final comparison. For each model we have run it with randomized 4000 - 1000 training splits for 100 iterations to stabilize the score and get variance.

Each model was scored using the algorithm proposed by the guidelines with the following modifications:

1. The scoring was initially scaled to match the amount of positive instances in the validation dataset.
2. The punishment for feature use was scaled according to the following formula: $-200 * customer\_selection\_threshold/1000$. This is done to scale the penalty according to if the score was 10.000 and the amount of persons to select was 1000.

We have also included the scores for feature interaction method, where we append squares of features to the existing selection. This method was added because it has shown to slightly improve scores during our experiments.

The following is a table of scores and accuracies the models have shown on our final feature selection (bold is the model and interactions we have selected for final evaluation with validation dataset).

Method	Score (no interactions)	Score (sq. features)	Accuracy (no interactions)	Accuracy (sq. features)
Support Vector Machine	**5935.68 ± 147.11**	5870.77 ± 133.88	0.72 ± 0.01	0.71 ± 0.01
Random Forest	5696.16 ± 143.39	5833.36 ± 146.30	0.69 ± 0.01	0.70 ± 0.01
Gradient Boosting	5751.99 ± 148.45	5819.00 ± 136.16	0.70 ± 0.01	0.70 ± 0.01
K-Nearest Neighbors	5793.97 ± 140.07	5781.33 ± 141.82	0.55 ± 0.02	0.64 ± 0.02

Table 4: Classification methods initial comparison

Conclusion

In conclusion throughout our experiments we have found that:

1. The dataset is fairly clean, as it has no missing values, it has a little amount of correlated features, it has no features with high VIF and the target variable is balanced.
2. The dataset has a lot of noise and it is not linearly separable as the linear regression models fail to select good features and fail to accurately choose the target variable.
3. Our selection methods, including non-standard approach with genetic algorithm, have shown us that the best selection of features we can base our final answer on is features 101 to 106 (index from 1).
4. Our models performed with an accuracy of 70-72% and score of 5.800 - 6.000. The score does not precisely reflect the outcome of evaluation on the validation dataset, but it gives us an idea of what could our models performance look like.
5. The model we chose to base our evaluation on is SVM with no interactions.

References

[1] Bolón-Canedo, Verónica & Sánchez-Maroño, Noelia & Alonso-Betanzos, Amparo. (2012). A review of feature selection methods on synthetic data. Knowledge and Information Systems. 34. 10.1007/s10115-012-0487-8.

[2] Gad, A. (2023). Pygad: An intuitive genetic algorithm python library. Multimedia Tools and Applications, 1–14.

[3] AlDelemy, Ahmad & Abd-Alhameed, Raed. (2023). Binary Classification of Customer's Online Purchasing Behavior Using Machine Learning. Journal of Techniques. 5. 163-186. 10.51173/jt.v5i2.1226.