LONG-TERM DEPOSIT SUBSCRIBING PREDICTION FOR A PORTUGUESE MARKETING CAMPAIGN

Ye Shenghua G2406019J
Ding Juncheng G2404083G
Qi Yu G2302682J
Liu Yiting G2405198F
Mai Xiyang G2404171F
Xiang Fenghao G2405076C

Group 14
Msc in Analytics
School of Physical and Mathematical Sciences
Nanyang Technological University, Singapore

October 24, 2024

ABSTRACT

The study of increasing marketing efficiency has been a real issue in recent years. The Internet has revolutionized how people obtain and process information, leading to the gradual decline of traditional marketing methods influence on the public. So, the shift from traditional marketing methods to data-driven approaches, leveraging machine learning techniques to optimize marketing resources, is necessary. Through this report, we aim to increase the accuracy of long-term deposit subscribing prediction and contribute valuable insights into customer behavior and marketing efficiency, offering practical implications for future banking campaigns.

1 Introduction

Mass marketing and direct marketing is two main approaches to advertising and promotion. However, with an abundance of products and a competitive landscape, traditional mass promotion tactics are losing their impact - the response rate is less than 1% (Ling and Li 1998).As mass marketing becomes less effective, banks, financial services, and other companies are shifting towards direct marketing strategies. They now focus on targeting customers with personalized offers for specific products and services, leveraging data insights to better meet individual needs and preferences. This approach enhances customer engagement and fosters stronger relationships(Petrison, Blattberg, and Wang 1997). While direct marketing has become a significant application of data mining. Companies widely use data mining techniques to identify prospective customers for new products by analyzing purchasing data. Through predictive modeling, they can assess the likelihood of a customer responding to a promotion or offer, enabling more targeted and effective marketing campaigns(Ayetiran and Adeyemo 2012).

In marketing, data used to predict subscription responses often exhibits a binary imbalance, as the number of customers who respond positively to a promotion is typically much smaller than those who do not. This imbalance poses a challenge for standard learning algorithms, as it can significantly compromise their performance. The algorithms may become biased towards the majority class, making it difficult to accurately predict positive responses(He and Garcia 2009). Furthermore, the minority class, which represents the positive responses, is often of greatest interest from a learning perspective and misclassifying this class can lead to significant costs(Elkan 2001). E.g., The goal of the classifier in this report is to accurately identify customers who are likely to subscribe to long-term deposits, which belong to the minority class. Misclassifying this group can lead to missed opportunities, as potential subscribers may not receive targeted offers, resulting in lost revenue. Therefore, the classifier must be designed to prioritize the correct identification of the minority class to maximize the effectiveness of marketing campaigns.

A previous study on predicting bank marketing campaign outcomes was conducted by Moro, Laureano, Cortez, et al. (2011). They utilized the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology to develop a predictive model aimed at improving the effectiveness of bank deposit subscription campaigns. However, they did not adequately address the issue of imbalanced datasets, especially in data processing and performance evaluate. Our study focuses on that using more scientific data mining methods to enhance binary classifiers performance in this imbalanced dataset.

The remainder of the report is organized as follows: section 2 works on introducing the data set and primary analysis of features; section 3 focuses on explaining the iterations of developing the binary classifier; section 4 explains how we process the data set, train and test classifiers; section 5 displays the result and analyzing the features based on data mining methods; section 6 is the conclusion containing insights and feature work.

Group contribution is attached in Appendix B.

Code and dataset are public on Github (Long-term-deposit-subscribing-prediction) for reference.

2 Data set

The dataset is derived from a Portuguese marketing campaign aimed at promoting bank deposit subscriptions through phone call. It has complete information on the personal information of bank clients and the contact performance of bank clients before, during, and after the bank marketing campaign. Number of Instances: 45211 without null values and duplicated values. Number of Attributes: 16 predictor variables and 1 response variable. Attribute information is attached in Appendix A.

2.1 Response variable

Figure 1 indicates the data set for binary classification is imbalanced: the no label has a count of 39922, and the yes label only has a count of 5289.

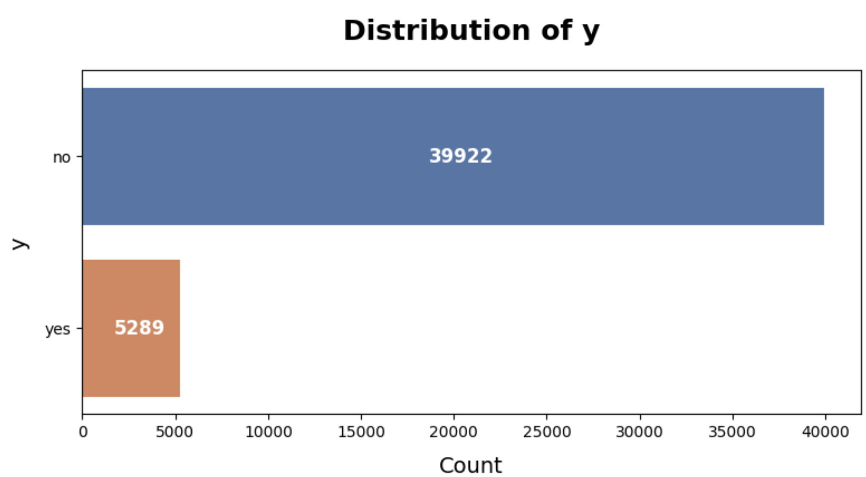

Figure 1: The distribution of response variable: y

2.2 Categorical variables

The data set has 9 categorical variables including job, marital, education, default, housing, loan, contact, month, and poutcome. Their distributions by target is shown in Figure 2.

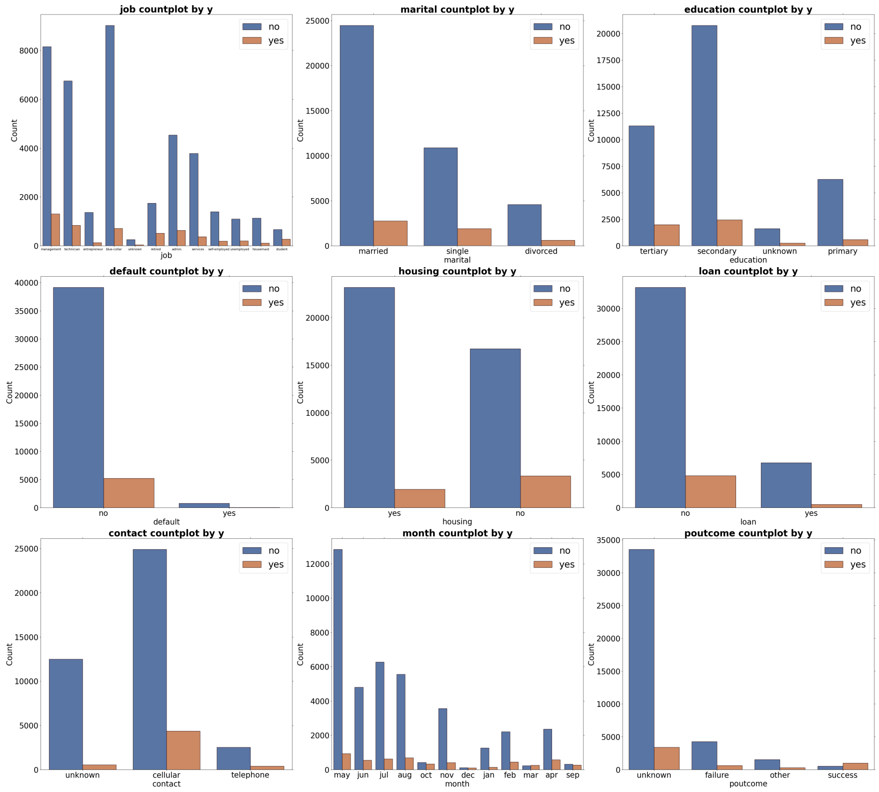

Figure 2: The distribution of categorical variables by target

Job count plot shows that the number of management and technicians who quit the bank is relatively higher than others.
Marital count plot shows the number of married and single who quit the bank is relatively high.
Education count plot shows the number of tertiary and Secondary who quit the bank is relatively high.
Default count plot shows the number of those not having credit in default who quit the bank is relatively high.
Count plot of housing loan shows the number of those not having housing loans who quit the bank is relatively high.
Contact count plot shows the number of cellular contacts that quit the bank is relatively high.
Loan count plot shows the number of not having loans that quit the bank is relatively high.
Month count plot shows the number of quit the bank in the months from May to August is relatively high.
P-outcome count plot shows the outcome of the previous marketing campaign unknown that quitting the bank is relatively high.

2.3 Numerical variables

There are 7 numerical variables and their distributions are shown in Figure 3.

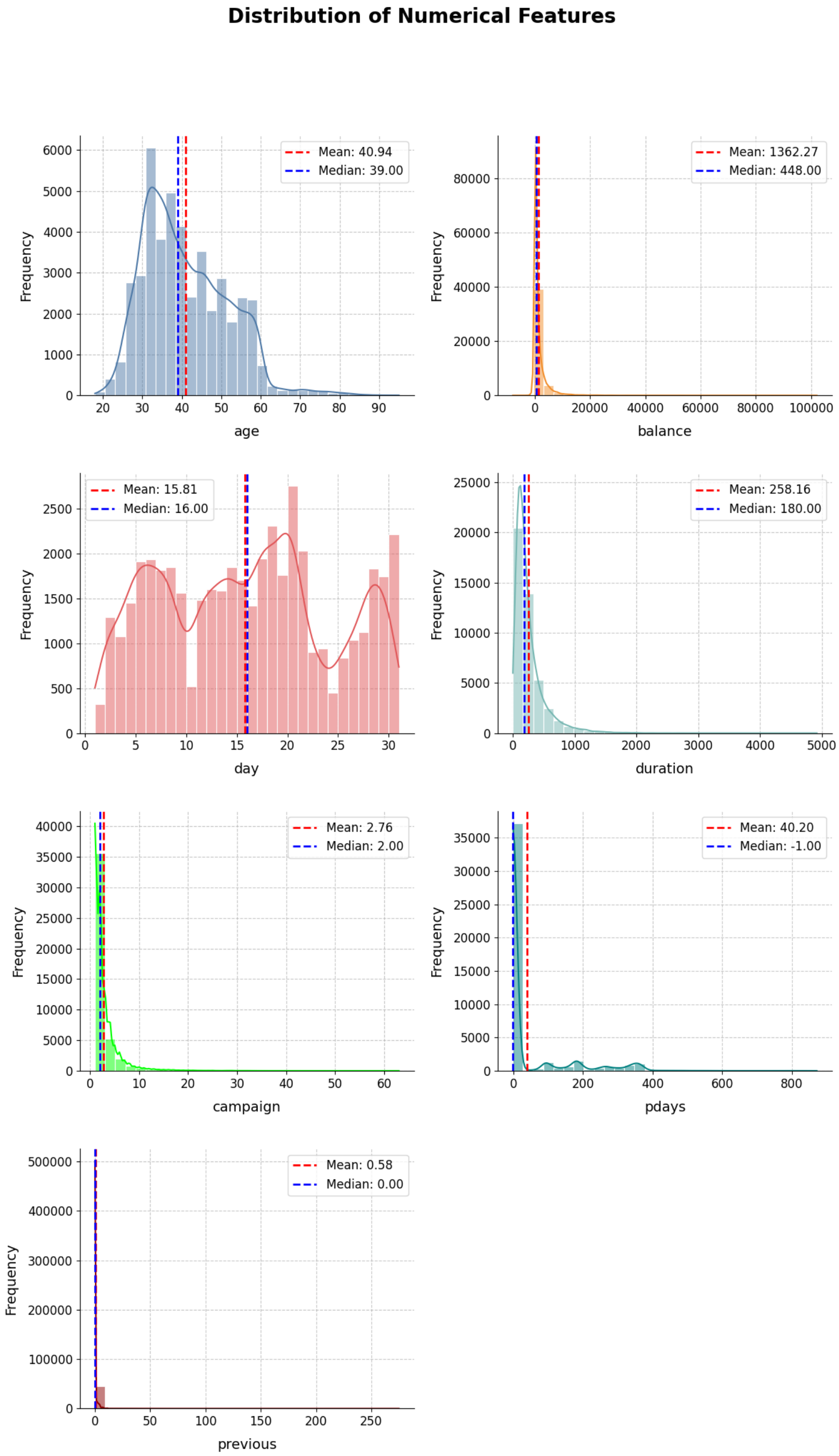

Figure 3: The distribution of numerical variables

From the age, the age distribution is slightly skewed to the right. Most customers are aged between 30 and 50, with fewer customers aged 60 and older.
From the balance, the significant difference between the mean and median indicates that a few customers with very high balances are pulling the mean upwards, while most customers have much lower balances.

From the day, the distribution of the "day" variable has multiple peaks.It suggests that contacts with customers are not evenly distributed throughout the month, with spikes on certain days.
From the duration, most interactions with customers are relatively short, but a few outlier calls skew the average upwards.
From the campaign, most customers were contacted only a few times, The skewness and the difference between the mean and median suggest that there are a few customers who were contacted many more times, pulling the mean higher than the median.
From pdays and previous, we can find most customers had no previous contacts, suggesting that a significant portion of the customer base was being contacted for the first time.

2.4 Correlation Matrix

Figure 4 is the heatmap of the Pearson correlation matrix, we can find that the correlation coefficient of duration is 0.39, which means there is a very strong positive correlation coefficient with last contact duration, meaning that the longer the last contact lasts, the more likely customers are to quit the bank.

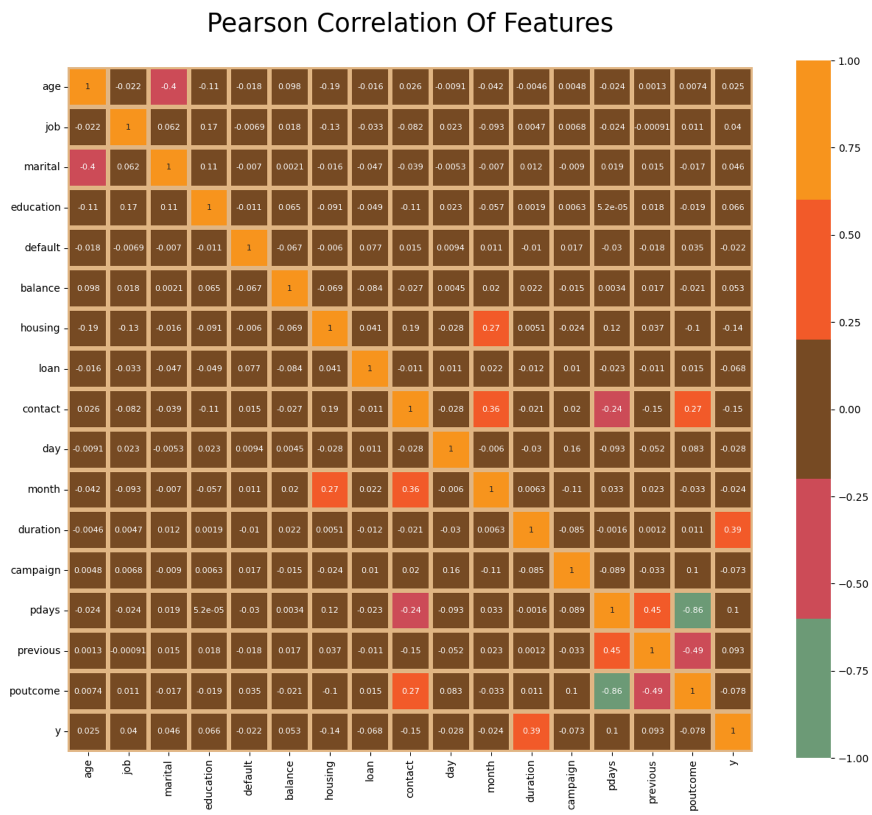

Figure 4: Pearson correlation matrix

3 Methodology

In this section, we explain the development of our machine-learning models over a set of three iterations.

3.1 First iteration: Model reproduction

Moro, Laureano, Cortez, et al. (2011) created data mining models to explain the success of a contact, i.e. if the client subscribes to the deposit. So our first step to study is to reproduce the models they created, including Naïve Bayes (Zhang 2004), Decision Tree (Apté and Weiss 1997), and Support Vector Machines (SVM) (Hearst et al. 1998).

What's more, we made some improvements in data processing and performance evaluation to enhance the professionalism and rationale of the whole process. Please see the section 4 for the detailed information.

3.2 Second iteration: Model development

According to Singhal et al. (2018), the bagging and boosting classifiers have a higher precision score on overpowered positive classes than the standard machine learning algorithms, especially about their robustness against the unbalanced class. So, we decided to use bagging and boosting classifiers.

Random forest (Ho 1995) is the bagging classifier we selected, while for boosting ones we use Xgboost (Chen and Guestrin 2016), LightGBM(Ke et al. 2017), Catboost(Prokhorenkova et al. 2018).

Random Forest: an ensemble learning method that builds multiple decision trees and combines their outputs to improve prediction accuracy and reduce overfitting.

XGBoost: a gradient boosting framework that optimizes model performance through regularization, parallel processing, and efficient handling of missing values, making it suitable for large datasets.

LightGBM: a gradient boosting framework that uses histogram-based algorithms and leaf-wise tree growth to achieve faster training speed and higher efficiency, especially on large datasets with many features.

CatBoost: a gradient boosting algorithm that handles categorical features natively without the need for extensive preprocessing, leading to better accuracy and reduced overfitting in datasets with many categorical variables.)

What's more, Optuna (Akiba et al. 2019)- Python package was used to tun due to the complexity of parameters in new classifiers.

3.3 Third iteration: Model optimization

Although the results in the second iteration are significantly better than in the first, the lengthy time required for tuning parameters remains a concern. We tried to apply particle Swarm Optimization (PSO) to accelerate the process of optimizing the CatBoost algorithm which took the longest time to tun.

3.3.1 PSO

PSO is a heuristic algorithm that simulates the collective behavior of a group, inspired by swarm intelligence (Marini and Walczak 2015). In this approach, particles continuously adjust their positions in the solution space to find the optimal solution."The algorithm updates each particle's velocity and position based on their individual experience and the entire group's experience, gradually approaching the global optimum. As shown in our animation, particles in PSO are randomly distributed at first, and through the iterative process, they gradually move toward the optimal solution. During iterations, particles adjust themselves according to the best positions of both the individual and the group, and eventually, the entire swarm converges toward the global optimal solution. PSO is characterized by its simplicity, strong global search capability, and suitability for solving complex optimization problems.

The general steps are as follows: 1) Initialize the particle swarm (with randomly distributed positions and velocities). 2)Calculate the fitness of each particle (based on the current solution). 3)Update the velocity and position of particles (based on personal best and global best positions). 4)Check the termination condition (e.g., maximum iterations or convergence to the global optimal solution).

3.3.2 Combining PSO with CatBoost

The process of combining Particle Swarm Optimization (PSO) with CatBoost parameter tuning is as Figure 5 shows.

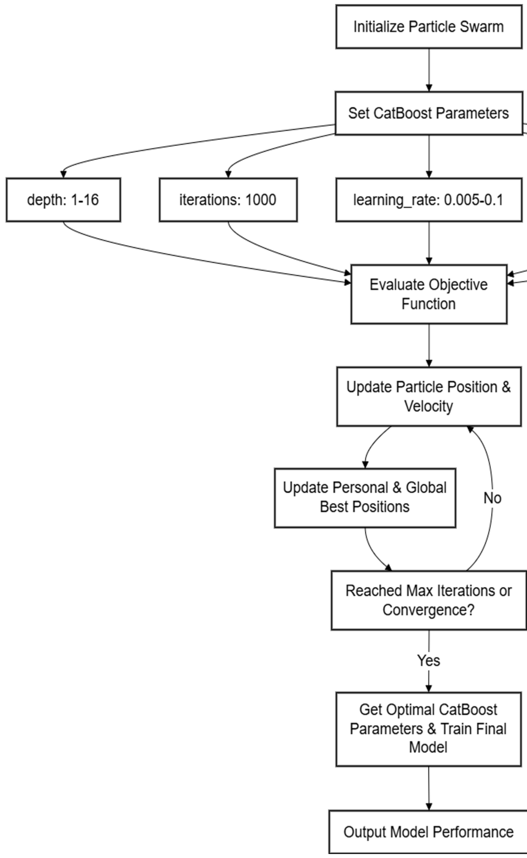

Figure 5: Process of Combining PSO with CatBoost

First, the PSO algorithm simulates swarm behavior. We initialize multiple particles, where each particle represents a combination of CatBoost model parameters. The position and velocity of each particle are randomly assigned, ensuring a broad search in the initial phase.

Next, we define the key parameters for CatBoost. These parameters include the depth of the decision trees (depth), the number of iterations (iterations), the learning rate (learning rate), the subsample ratio (subsample), and the random sampling ratio of features (RSM). Each particle's set of these parameters will influence the model's performance.

Then, we calculate the objective function. Each particle uses its corresponding set of parameters to train a CatBoost model, and its fitness score is calculated based on our chosen evaluation metric, the Matthews Correlation Coefficient (MCC). These scores help us assess the performance of each particle.

Next, the particles update their parameters by adjusting their positions and velocities. This optimization process is influenced by both the particle's own best-known position and the best-known position of the swarm. In other words, each particle draws from its own experience and references the best solution found by the group.

Finally, we check whether the convergence condition has been met. If the stopping criteria are satisfied, such as reaching the maximum number of iterations or achieving a sufficiently stable solution, the iteration process terminates.

After 10 iterations, PSO optimization identified the best CatBoost model parameters, including a depth of 9, a learning rate of 0.0098, a sample sampling rate of 0.8294, and a feature sampling rate of 0.129. The model achieved its best performance on the Matthews Correlation Coefficient (MCC) with a score of 0.5688, indicating good classification ability, especially in handling imbalanced positive and negative samples. Although the iteration limit was reached, the current combination of parameters has provided a relatively balanced and strong classification performance.

As shown in Figure 6, the model performs exceptionally well in 5-fold cross-validation, achieving an average AUC of 0.9295, indicating strong classification ability. Moreover, the AUC for each fold remains between 0.9247 and 0.9328, demonstrating the model's high stability. The ROC curve is close to the upper left corner, which suggests that the model can accurately predict most positive samples while maintaining a low false-positive rate. This shows that by optimizing the parameters, we have obtained a model with excellent performance across different datasets and strong generalization ability.

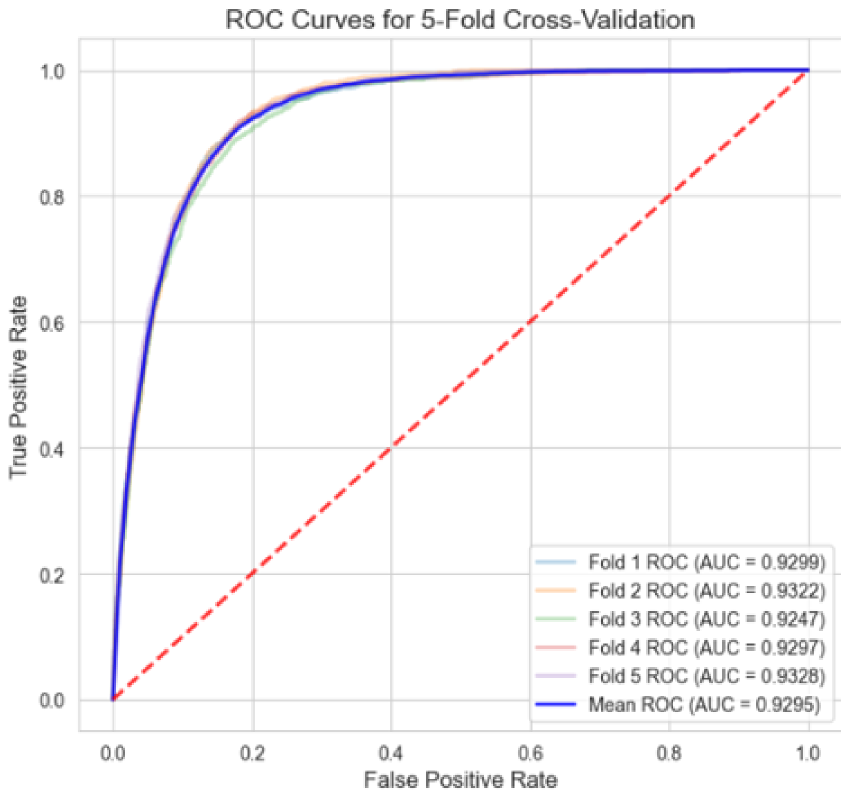

Figure 6: ROC for CatBoost with PSO

4 Experiment process

The training and testing procedure consists of six procedures, including:1) 5-fold cross-validation to split the data set. 2) Scaling numerical features between 0 and 1. 3) Encoding categorical features - label encoder for binary and one-hot encoder for the other. 4) Selecting attributes training the model. 5) Combing sampling to balance the training data set. 6) Putting the processed dataset into models and evaluating the performance.

Figure 7 taking the first fold example describes the specific change process of the training and testing data set in dimensions and numbers of instances.

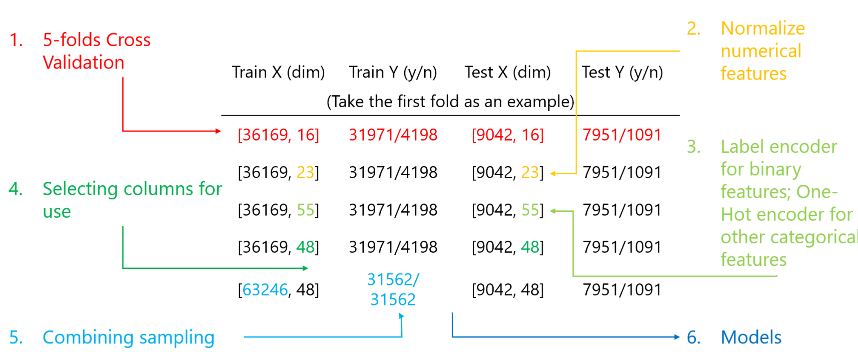

Figure 7: Data processing process

As in the table above, the dataset had 16 columns at first. After we normalized the numerical features, we added 7 new features as columns. Then we applied a label encoder for the binary features and a one-hot encoder for the categorical features. The number of columns increases to 55. The initial, unprocessed features are still being kept, so we only need 48 of them to do our work. The last step is combining sampling. Note that we only do combining sampling on our training set. The testing set should remain unaffected by the sampling methods. After all of these, the processed dataset is placed into our models. Next, all the procedures will be introduced step by step.

4.1 5-folds cross validation

We divide the whole dataset (45211 records) into 5 equal-sized parts(folds). For each fold, we have to make sure that it has all kinds of categorical features, and is allocated a negative target value on average. In our dataset, 45211 records are separated into 5 folds, with about 9042 records in each of them. Iteratively, we choose one of the folds as the testing set and others as the training set.

4.2 Normalize the numerical features

Numerical features (age, balance, duration, campaign, pdays, previous, and day) are scaled between 0 and 1 based on the range of the training data set. And the name of new attributes appends '_scaled' on the original ones.

4.3 Encode the numerical features

We applied label encoders for binary features (default, housing, loan, y), assigning 'no' to 0, 'yes' to 1, and one-hot encoders to process the categorical features (marital, education, contact, poutcome, month, job). Picking a specific example from our dataset. The feature 'marital' has values: 'single', 'married', and 'divorced'. It was divided into 3 features: 'marital_single', 'marital_married', and 'marital_divorced' by the one-hot encoder.

4.4 Select attributes

48 predictor variables are selected as follows: default, housing, loan, age_scaled, balance_scaled, duration_scaled, campaign_scaled, pdays_scaled, previous_scaled, day_scaled, marital_divorced, marital_married, marital_single, education_primary, education_secondary, education_tertiary, education_unknown, contact_cellular, contact_telephone, contact_unknown, poutcome_failure, poutcome_other, poutcome_success, poutcome_unknown, month_apr, month_aug, month_dec, month_feb, month_jan, month_jul, month_jun, month_mar, month_may, month_nov, month_oct, month_sep, job_admin., job_blue-collar, job_entrepreneur, job_housemaid, job_management, job_retired, job_self-employed, job_services, job_student, job_technician, job_unemployed, job_unknown.

4.5 Combining sampling

Due to the imbalance of the dataset, we use SMOTETomek - combining sampling method to balance the training dataset. SmoteTomek can be referred to as the Smote + Tomek. In the process, we first do oversampling with SMOTE. SMOTE randomly selects samples from the minority class and their nearest neighbors. Then, we do under-sampling with Tomek. Tomek links are pairs of nearest neighbors that come from different classes. For the Tomek links, samples are removed to have cleaner boundaries. By combining these, we have our training sets more balanced and with clearer boundaries.

4.6 Model training and evaluation

The selected training models are introduced in section 3.1, section 3.2,section 3.3

The selected performance evaluation metrics are listed below:

Matthews Correlation Coefficient (MCC) introduced by Matthews (1975), is a measure of the quality of binary classifications, ranging from -1 (total disagreement) to +1 (perfect agreement), and accounts for true and false positives and negatives. So MCC can evaluate performance well even for imbalanced cases. The formula is as follows:

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Receiver Operating Characteristic - Area Under the Curve (ROC-AUC) measures the performance of a binary classifier by plotting the true positive rate (recall) against the false positive rate at various threshold levels. The Area Under the Curve (AUC) represents the overall capability of the model to distinguish between classes. AUC values range from 0 to 1, where 1 indicates a perfect classifier, and 0.5 represents a classifier performing no better than random guessing. It's widely used for evaluating models where the data is imbalanced because it focuses on the model's ability to separate positive and negative classes. The formula is as follows:

$$\mathit{AUC} = \int_0^1 TPR(FPR^{-1}(x))dx$$

Accuracy is the ratio of correctly predicted instances (both true positives and true negatives) to the total number of predictions. It is one of the simplest evaluation metrics and works well when the classes are balanced. However, in the case of imbalanced datasets, accuracy can be misleading as it does not differentiate between the types of errors (e.g., it does not account for false positives and false negatives). The formula is as follows:

$$\mathit{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision Precision measures the proportion of correctly predicted positive observations out of all predicted positives. It is important when the cost of false positives is high, as it focuses on minimizing incorrect positive predictions. A high precision score means fewer false positives. The formula is as follows:

$$\mathit{Precision} = \frac{TP}{TP + FP}$$

Recall Recall (also called Sensitivity or True Positive Rate) measures the proportion of correctly predicted positive observations out of all actual positives. It is important when missing positive cases (false negatives) is costly, and it focuses on minimizing incorrect negative predictions. The formula is as follows:

$$\mathit{Recall} = \frac{TP}{TP + FN}$$

Balanced F-score (F1 score) The F1-score is the harmonic mean of precision and recall, providing a single metric that balances both. It is useful when there's an uneven class distribution or when both precision and recall are important to balance. The F1-score ranges from 0 to 1, where 1 is the best possible score. It's especially effective when false positives and false negatives carry different costs. The formula is as follows:

$$\mathit{F1\text{-}score} = 2 \cdot \frac{\mathit{Precision} \cdot \mathit{Recall}}{\mathit{Precision} + \mathit{Recall}}$$

5 Experimental result and analysis

Table 1 is the results of the machine learning models on imbalanced data using various metrics, and the underline results refer to the best in their metrics.

Table 1: Experimental results

Model	MCC	ROC-AUC	Accuracy	Precision	Recall	F1
Naïve Bayes	0.3529	-	0.8236	0.3475	0.5788	0.4342
Decision Tree	0.4256	-	0.8637	0.4372	0.5769	0.4973
SVM	0.5195	-	0.8801	0.4912	0.6972	0.5763
Random Forest	0.551	0.928	0.8918	0.5278	0.707	0.6043
XGBoost	0.5582	0.9262	0.8994	0.5576	0.6779	0.6118
LightGBM	0.5632	0.9286	<u>0.8999</u>	<u>0.5576</u>	0.6968	0.6194
CatBoost	0.5614	0.9284	0.8959	0.5422	0.7072	0.6138
CatBoost with PSO	<u>0.5688</u>	<u>0.9295</u>	0.8967	0.5429	<u>0.7376</u>	<u>0.6287</u>

Comparing the first iteration models to the second ones, We can conclude that our models significantly outperform the baseline model in every metric.

Comparing the second iteration models to CatBoost with PSO (third iteration), CatBoost with PSO achieves the highest MCC score, ROC-AUC score, Recall score, and F1 score, indicating a strong performance. While LightGBM also performs well, especially in accuracy and precision, CatBoost with PSO stands out for handling the imbalance better, particularly in recall.

We believe that the reason for such a phenomenon is that accuracy and precision themselves do not directly reflect the model's overall ability to handle imbalanced data, especially in terms of predicting the majority class. LightGBM may perform very well in the majority class. Although CatBoost is better at handling imbalanced data, the efficiency of LightGBM allows it to achieve higher accuracy and precision in certain datasets. This is because accuracy and precision themselves do not directly reflect the model's overall ability to handle imbalanced data, particularly in predicting the majority class. LightGBM may perform very well in the majority class, thus boosting overall accuracy and precision.

5.1 Feature importance analysis

To guide how to allocate marketing resources effectively and identify high-potential clients, We extracted feature importance from Catboost. The result is in Figure 8.

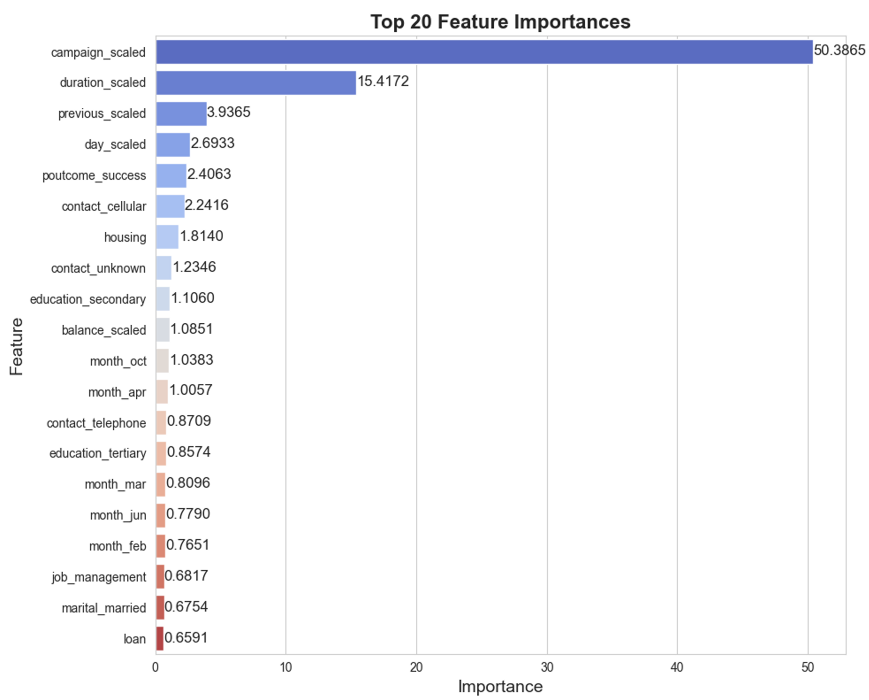

Figure 8: Feature importances

The most crucial features are campaign-scaled and duration-scaled, which have the highest importance scores, indicating their strong impact on the model's accuracy. Historical behavior, such as previous(scaled) and poutcome(success), also significantly affect predictions.

Additionally, customer contact such as contact frequency and call duration are identified as critical factors in determining outcomes. These insights suggest the model heavily relies on features related to customer contact and historical records, highlighting the critical importance of factors such as contact frequency and call duration in the prediction.

6 Conclusion

This project builds on Moro, Laureano, Cortez, et al. (2011) to further explore direct bank marketing activities. Compared to them, the predictive performance of our model is significantly improved, and the pre-processing of the unbalanced dataset and the model performance evaluation methodology that we have adopted are more reasonable.

Our model (Catboost with PSO) achieved the highest score in MCC, ROC-AUC, Recall, F1, indicating that it has an edge in classifying the positive and negative classes more effectively. In addition, we measured Catboost variable importance, and bank managers can use this knowledge to organize campaigns that provide marketing efficiency and reduce project costs.

6.1 Insights

Importance of Data Preprocessing The dataset contains a variety of categorical and numerical features. Through appropriate encoding methods (such as Label Encoding and One-Hot Encoding) and sample balancing techniques, the training effectiveness of the model was greatly enhanced. Additionally, standardizing the numerical features ensured that different features had a consistent scale within the model, preventing biases.

Value of Hyperparameter Optimization In machine learning projects, choosing the right model is important, but using automated optimization tools like PSO and Optuna to adjust hyperparameters can significantly improve model performance. Flexibly selecting suitable optimization methods according to the characteristics of different models is especially crucial.

Model Evaluation and Cross-Validation 5-fold cross-validation ensures the model's stability and generalization across different datasets. Especially for imbalanced datasets, choosing appropriate evaluation metrics (such as MCC and AUC) provides a more accurate assessment of model performance. The balance provided by cross-validation confirms the model's reliability.

Guiding Significance of Feature Analysis By using the feature importance chart provided by the model, it is possible to identify which features contribute the most to the model. This not only helps improve the current model's predictive ability but also provides direction for future feature engineering.

6.2 Future work

In the future directions for further improvements in this group project, we can focus on three key areas: addressing data imbalance issues, enhancing model performance, and utilizing deep learning algorithms based on refined loss functions.

Given the significant data imbalance in this project, we can apply a combination of oversampling and undersampling techniques such as SMOTETomek. This method not only increases the number of minority class samples but also effectively removes noise data from the majority class, therefore balancing the sample distribution will improve the model's Stability.

To enhance model performance, Using Bayesian optimization algorithms, it is possible to expand the parameter range and explore the parameter space efficiently. Additionally, the model ensemble strategy is another effective solution. By using voting or weighted averaging methods, so multiple strong models can be combined to make predictions. For instance, each model can output a classification result for a sample, and the majority prediction can be taken as the final result. Alternatively, the probabilities of each model predicting the positive class can be averaged, and the final classification can be determined based on a threshold.

Lastly, in terms of deep learning methods with improved loss functions, Focal Loss is worth considering. Focal Loss is specifically designed for imbalanced classification problems, Alpha t balances the effects of positive and negative samples and gamma controlled for the attention of the difficulty sample. Therefore, by adjusting the modulation factor to reduce the impact of majority class samples on the loss. At the same time, combining deep learning algorithms such as Transformer or GANs can further improve the accuracy of minority class classification.

References

Akiba, Takuya et al. (2019). "Optuna: A next-generation hyperparameter optimization framework". In: *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2623–2631.

Apté, Chidanand and Sholom Weiss (1997). "Data mining with decision trees and decision rules". In: *Future generation computer systems* 13.2-3, pp. 197–210.
Ayetiran, Eniafe Festus and Adesesan Barnabas Adeyemo (2012). "A data mining-based response model for target selection in direct marketing". In: *IJ Information Technology and Computer Science* 1.1, pp. 9–18.
Chen, Tianqi and Carlos Guestrin (2016). "Xgboost: A scalable tree boosting system". In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794.
Elkan, Charles (2001). "The foundations of cost-sensitive learning". In: *International joint conference on artificial intelligence*. Vol. 17. 1. Lawrence Erlbaum Associates Ltd, pp. 973–978.
He, Haibo and Edwardo A Garcia (2009). "Learning from imbalanced data". In: *IEEE Transactions on knowledge and data engineering* 21.9, pp. 1263–1284.
Hearst, Marti A. et al. (1998). "Support vector machines". In: *IEEE Intelligent Systems and their applications* 13.4, pp. 18–28.
Ho, Tin Kam (1995). "Random decision forests". In: *Proceedings of 3rd international conference on document analysis and recognition*. Vol. 1. IEEE, pp. 278–282.
Ke, Guolin et al. (2017). "Lightgbm: A highly efficient gradient boosting decision tree". In: *Advances in neural information processing systems* 30.
Ling, Charles X and Chenghui Li (1998). "Data mining for direct marketing: Problems and solutions." In: *Kdd*. Vol. 98, pp. 73–79.
Marini, Federico and Beata Walczak (2015). "Particle swarm optimization (PSO). A tutorial". In: *Chemometrics and Intelligent Laboratory Systems* 149, pp. 153–165.
Matthews, Brian W (1975). "Comparison of the predicted and observed secondary structure of T4 phage lysozyme". In: *Biochimica et Biophysica Acta (BBA)-Protein Structure* 405.2, pp. 442–451.
Moro, Sergio, Raul Laureano, Paulo Cortez, et al. (2011). "Using data mining for bank direct marketing: An application of the crisp-dm methodology". In: *Proceedings of the European Simulation and Modelling Conference-ESM*. Vol. 2011.
Petrison, Lisa A, Robert C Blattberg, and Paul Wang (1997). "Database marketing: Past, present, and future". In: *Journal of Direct Marketing* 11.4, pp. 109–125.
Prokhorenkova, Liudmila et al. (2018). "CatBoost: unbiased boosting with categorical features". In: *Advances in neural information processing systems* 31.
Singhal, Yash et al. (2018). "Review of Bagging and Boosting Classification Performance on Unbalanced Binary Classification". In: *2018 IEEE 8th International Advance Computing Conference (IACC)*, pp. 338–343. DOI: 10.1109/IADCC.2018.8692138.
Zhang, Harry (2004). "The optimality of naive Bayes". In: *Aa* 1.2, p. 3.

A Attribute information

age: (numeric)

job: type of job (categorical: admin., unknown, unemployed, management, housemaid, entrepreneur, student, blue-collar, self-employed, retired, technician, services)

marital: marital status (categorical: married, divorced, single)

education (categorical: unknown, secondary, primary, tertiary)

default: has credit in default? (binary)

balance: average yearly balance, in euros (numeric)

housing: has a housing loan? (binary)

loan: has a personal loan? (binary)

contact: contact communication type (categorical: unknown, telephone, cellular)

day: last contact day of the month (numeric)

month: last contact month of the year (categorical: Jan to Dec)

duration: last contact duration, in seconds (numeric)

campaign: number of contacts performed during this campaign and for this client (numeric, includes the last contact)

pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric, -1 means the client was not previously contacted) previous: number of contacts performed before this campaign and for this client (numeric)

poutcome: outcome of the previous marketing campaign (categorical: unknown, other, failure, success)

B Contribution

Member Name	Percentage of Contributions	Justification
Ye Shenghua	20 %	Team leader
Ding Juncheng	16 %	Data exploration
Qi Yu	16 %	Future work
Liu Yiting	16 %	Highlights
Mai Xiyang	16 %	Experimental study and analysis
Xiang Fenghao	16 %	Training and testing procedure