Assignment Report - Group: 119

Group members: Xumou Zhang, Yichen Chen, Zhuoxi Kuang, Jing Jia, Xinyue Meng

Assignment cover sheet

Unit of Study: QBUS6810

Assignment name: Classification Project: Marketing Analytics

DECLARATION

We the undersigned declare that we have read and understood the University of Sydney Student Plagiarism: Coursework Policy and Procedure, and except where specifically acknowledged, the work contained in this assignment/project is our own work and has not been copied from other sources or been previously submitted for award or assessment.

We understand that failure to comply with the Student Plagiarism: Coursework Policy and Procedure can lead to severe penalties as outlined under Chapter 8 of the University of Sydney By-Law 1999 (as amended). These penalties may be imposed in cases where any significant portion of my submitted work has been copied without proper acknowledgement from other sources, including published works, the internet, existing programs, the work of other students, or work previously submitted for other awards or assessments.

We realize that we may be asked to identify those portions of the work contributed by each of us and required to demonstrate our individual knowledge of the relevant material by answering oral questions or by undertaking supplementary work, either written or in the laboratory, in order to arrive at the final assessment mark.

Project team members				
Student Name	**Student ID**	**Participated**	**Agree to share**	**Signature**
1. Xumou Zhang	510317173	**Yes**/No	**Yes**/No	
2. Yichen Chen	510138903	**Yes**/No	**Yes**/No	Yichen Chen
3. Zhuoxi Kuang	510089546	**Yes**/No	**Yes**/No	Zhuoxi kuang
4. Jing Jia	500686407	**Yes**/No	**Yes**/No	Jing Jia
5. Xinyue Meng	510286363	**Yes**/No	**Yes**/No	

1. Abstract

This assignment provides two separate datasets but with some level of similarity. Our tasks are first to implement five machine learning models using the knowledge we have learned in the lecture to evaluate the situation in each dataset, and then analyze and find more insights of the common area in these two datasets.

2. Introduction

2.1 Introduction of the Project

We have two main tasks in this project. The first task is to analyze the given two datasets using the machine learning techniques we have learned in the class. The entire procedure should be implemented on both datasets. After we get the result from both datasets, we will try to answer the question given in task 2. The question is: "what types of customers are more responsive to marketing campaigns?" Also, we will analyze the insights of these two datasets based on the results we got from task 1.

2.2 Overview of the Methodology and Main Results

The procedures of machine learning should go as follows: exploratory data analysis, feature engineering, model hyperparameter tuning and finally model evaluation and comparison. We can get the results or make predictions based on the model we have built and tuned.

3. Problem Formulation and Objectives

We are focusing on doing classification with the respect to the given datasets in this project since we chose the classification option instead of the regression option. Therefore, in our project, our research question would be: Can we accurately predict if the customer is going to respond to the marketing campaigns or not with given datasets? Respectfully, our objective would be trying to build up models using the bank and the store dataset and predict the yes or no to our research question.

4. Data Understanding

4.1 Overview of the Datasets

4.1.1 Bank Dataset

The bank dataset is saved in a CSV file, it has 16 columns and 29387 entries: including 2 continuous, 5 discrete, 5 categorical, 3 binary, 1 response. After deleting duplicate values, the bank dataset is 16 columns and 29380 entries. Continuous: age, balance: average yearly balance. Discrete: day: day of contact, campaign: number of contacts performed during this campaign for this client; pdays: number of days passed by after client was the last contact from a previous campaign; previous: number of contacts performed before this campaign for this client, id. Categorical: job: type of job, (categories: blue-collar, management and other); marital: marital status, (categories: married, single and other); education: level of education (categories: secondary, tertiary and other); contact: contact communication types (categories: cellular, unknown and other); month: day of contact (categories:'Jan', 'Feb'...... 'Dec'); poutcome:

outcome of a precious marketing campaign (categories: failure, unknown, success and other). Binary: default: does the client have credit default or not (yes/no); housing: does the client have a housing loan or not(yes/no); loan: Does the client have a personal loan or not (yes/no). Response:subscribed: does the client subscribe to a term deposit?

4.1.2 Store Dataset

The fashion store dataset is stored in a 4.9 MB CSV file containing 21740 rows and 48 columns of data. Each row of dataset represents a customer of the fashion store and its corresponding 48 related features. What we want to study is to predict whether customers will respond to email promotions and find out the important factors that affect customer response rate. The 48 variables contained in the dataset include 31 continuous variables, such as MON (total net sales), AVRG (average amount spent per visit), OMONSPEND (amount spent in the past month), etc.; 11 discrete variables, such as FRE (number of purchase visits), PROMOS (number of marketing promotions on file), DAYS (number of days the customer has been on file), etc.; 3 binary variables, such as CC_CARD (credit card user), VALPHON (valid phone number on file), WEB (web shopper); 2 categorical variables, such as ZIP_CODE and PC_CALC20 (brand); 1 response variable, RESP, which is the response result of the customer to the promotion. Through data exploration, it can be found that the 21740 customers included in the dataset have a promotion response rate of 16.61%.

4.2 Exploratory Data Analysis

The main purpose of the EDA in the project is to explore the data before we make any assumptions for the project. That been said, the procedures we will follow in the EDA section would be shown as follows:

First, checking the data types for each feature in the dataset; then checking the distribution of the label data. After that, we will be focusing on the univariate analysis and the bivariate analysis by creating the data visualization plots for the related data. In the end, we will need to create the correlation matrix for the dataset to check the basic correlation between different features before the feature engineering.

4.2.1 EDA of Store Dataset

There are 48 columns or 48 different features in the store dataset, in which, one is the response or label feature, 41 of them are the useful numerical data, 5 of them are the useful categorical data, and the last one is called "ZIP_CODE" or the useless nominal data in this case.

Next, we check the distribution of the label data, the data visualization chart is presented in the Appendix as Figure 1. It is obvious to see that the RESP yes or 1 is significantly lower than the number that RESP no or 0 in our case. This observation gives us an idea of what would be like in the result if we have enough input data and predict the output set.

After that, we implemented the Univariate analysis. We separate the data into categorical and numerical data for the univariate and bivariate analysis. The idea of the univariate analysis is similar to the observation we did above to check the distribution of a single feature and if there is any potential outlier that may or may not need to be processed in the later feature engineering. The majority of the features in the dataset are shown as a right-skewed distribution similar to the AVRG feature shown in Figure 2 left, and there is only one feature called DAYS shown as the left-skewed in the chart which is presented as the Figure 2 mid, and one feature called GMP shows more or less of a normal distribution as the Figure 2 right.

The categorical features on the other hand also show a similar story shown as Figure 3 in the appendix, most of the categorical data distributed fairly extreme to the one end so that the numerical difference between both ends is very large.

The next part is to perform the bivariate analysis. The idea of the bivariate analysis is to show the relationship between two different columns. In our case, we want to find out the basic relationship between each feature and the output RESP variable.

Both categorical and numerical data show a similar idea shown as Figure 4 in the appendix. Most data have a relationship like the FRE and POUTERWEAR features, and less than 25% of features show a similar relationship with the output variable to the PROMOS feature. In conclusion, with the increase in the numerical value of most of the features, the customers would either change their minds and respond yes or no change their minds and stick with no for an answer.

4.2.2 EDA of Bank Dataset

Similar to the Store dataset, we first check the distribution of the output label data which indeed also shows the same situation as above where the majority of the output is pointing to no rather than yes for the answer. The chart is shown in the Figure 5 left in the appendix.

Next, we implement the univariate analysis and the bivariate analysis. After we check each of the features, we find that the majority of them are distributed right-skewed, similar to the balance feature shown in Figure 5 mid in the appendix. There is only one exception in the dataset which is the day feature shown as somewhere normally distributed as the third chart shown in Figure 5 right in the appendix. As for the categorical data, we can see from Figure 6 that the majority of data are still contained in a single option rather than uniformly distributed.

Jumping to the bivariate analysis of the bank dataset, similarly, we conduct the charts to show the relationship between the features and the output variables, the plots we got for the numerical data is in Figure 7.

The bivariate charts show a similar situation that the most output is either change their mind from no to yes while the numerical value of the data is increased, and others just stay with no as the final result. The categorical data also shows that the proportion of saying no is much larger than saying yes despite more detailed classification in the charts shown in Figure 8.

4.2.3 Correlation Matrix:

In the end, we also compute the correlation matrix for the numerical features for both datasets; the correlation matrices are presented in Figure 9 in the appendix.

5. Feature Engineering

5.1 Data Cleaning

Data cleaning includes handling missing values, duplicate values and outliers.

5.1.1 Missing values

As shown in Figure 10 in the appendix , there are no missing values for both datasets.

5.1.2 Duplicate values

By checking the duplicate values, it can be seen in Figure 11 in the appendix that there are 7 duplicate values for the bank dataset and there are no duplicate values for the store dataset. So just removing these values.

5.2 Encoding

In the dataset we may contain data of categorical types. To build a model, we need to have all features which are integer data types. So, it is quite important to encoding data. We used three types of encoding, which are one-hot encoder, label encoder and ordinal encoder.

- One-Hot Encoder: Each category is mapped with a binary variable only containing either 0 or 1. 0 means the absence, and 1 represents the presence of that category.
- Ordinal Encoder: it is different from One-Hot Encoder, since it just converts features to integers, such as 1,2,3,4.
- Label Encoder: it is used to convert labels to 0 and 1.

In our assignment, Ordinal Encoder is applied in the LightGBM model, One-Hot Encoder is used in other models and Label Encoder is used in the label “subscribed”.

5.3 Feature Selection

There are three methods for feature selection, including filter, embedded and wrapper. In our assignment, embedded and wrapper methods were used.

- Wrapper methods: It is a greedy search algorithm that attempts to find the best subset of features by repeatedly selecting features based on model performance until it finally reaches the desired number of features to select. Forward feature selection, backward feature elimination, recursive feature elimination are some common examples of this method.
- Embedded methods: The embedded method combines the advantages of the filter and wrapper methods. It depends on algorithms which have their own feature selection methods, such as tree-based models, LASSO and RIDGE regression.

We used the Random Forest algorithm as the base model to select the number of features.

For the bank dataset, as shown in Figure 12 of appendix, it is the Embedded method showing the roc_auc_score of different threshold values of feature importance, and it is on a downward trend. When the threshold value is equal to 0, the score is the highest, indicating any features do not need to be removed. This result, also shown in Figure 13 of appendix, is for the wrapper method, suggesting the performance is best when all features are used.

For the store dataset, as seen in Figures 14 of appendix, the result is different from the bank dataset. when the threshold value equals about 0.03, the score is the highest. In this case, the features left is those with feature importance greater than approximately 0.03, and finally only 8 features are selected.

5.4 Feature Scaling

The reason why doing feature scaling is that if there is a huge difference in range, for example a few ranges in the thousands and a few ranges in the tens, it assumes that the numbers with the larger ranges are somehow superior. As a result, these more important numbers start to play a more decisive role when training the model. Feature Scaling contains Normalization and Standardization. In our assignment, the Normalization method was used to scale features.

Normalization: It rescales the feature to the range of [0,1] by subtracting the minimum value of features, then dividing by the range.

$$x' = \frac{x - x_{mean}}{x_{max} - x_{min}}$$

Figure 15: General Equation of Normalization

Standardization: It makes the mean of each feature is 0 and the standard deviation is 1.

$$Z = \frac{X - \mu}{\sigma}$$

Figure 16: General Equation of Standardization

6. Methodology

6.1Classification Model Selection and Introduction

In this project, we will be implementing 5 classification models in total as requested. Each model will be used in the bank dataset and the store dataset to compute the final results. The five models we chose are the decision tree classifier, random forest classifier, logistic regression classifier, LightGBM classifier and the stacking classifier. In addition, the benchmark model is the logistic regression model. As requested, we will only introduce the best three models out of these five, but we will compute and present the results of all models in the later sections.

6.1.1 LightGBM Classifier

LightGBM is an upgraded version of XGBoost. It overcomes drawbacks of XGBoost and to be able to speed up the training of GBDT models without lower performance, some optimizations are as follows.

- Histogram-based decision tree algorithm.
- Gradient-based One-Side Sampling (GOSS): Using GOSS reduces the number of data instances with only small gradients so that only the remaining data with high gradients can be used when calculating the information gain, saving a lot of time compared to XGBoost traversing all feature values.
- Exclusive Feature Bundling (EFB): Using EFB, many mutually exclusive features can be bundled into one feature, thus achieving dimensionality reduction.

- Leaf-wise leaf growth with depth restrictions: Most tree-based models use an inefficient level-wise decision tree growth strategy, which brings about a lot of unnecessary cost. While LightGBM uses a leaf-wise algorithm with a depth limit.
- Supporting categorical features.

6.1.2 Random Forest Classifier

Random forest classification is an ensemble algorithm, which constructs many decision trees to build a forest. Using the bootstrap method to generate B bootstrap samples from the train set. Each time to split a node, Random selects a subset of “k” variables from original “p” variables. Selecting the best variable and split point among k variables. And then split the node into two child nodes. Continue to repeat this process, until you reach minimum node size. Finally using majority voting to get the final prediction on random forest. The reason why we choose Random Forest is because it can decrease the correlation between trees. When the number of trees increases, it is less likely to overfit.

6.1.3 Stacking Classifier

Stacking classification algorithm is an ensemble method to solve the classification problem. The basic idea of the stacking classifier is to implement multiple models as the base models and use the outputs from these models as the inputs for the meta model to compute the result. Similar to the neural network structure, the stacking classifier can have more than one layer of models which could form a pyramid or building structure similar to the figure shown below:

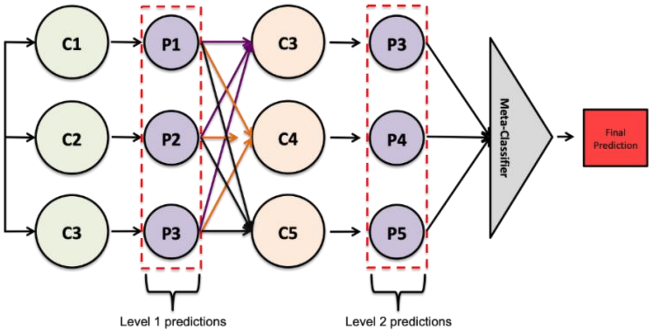


Figure 17: Graphic Description of the Stacking Classifier

Anyway, in our project, we are going to just use a one layered basic stacking classifier since the more complex multi-layer stacking classifier would take a really long time for the next part of the hyperparameter tuning section.

6.2 Hyperparameter Tuning Procedures

Hyperparameter optimization is essential to enhance the performance of models. We used two hyperparameter tuning tools which are GridSearchSV and the optuna library. Importing Grid Search with 10-fold cross-validation to find the optimized value of hyperparameter. Cross-validation with 10-fold is commonly used in machine learning. Using cross-validation can avoid overfitting, it has low variance and computational efficiency. Grid search is exhaustive search methods to search every specified value of a parameter. The other tool is optuna which is used to tune LightGBM model.

6.2.1 LightGBM

The tuning parameters are divided into four categories.

- Parameters that affect the structure and learning of the decision tree, such as, max_depth, num_leaves and min_data_in_leaf.
- Parameters that affect the speed of training, such as early_stopping_round
- Parameters that improve performance, like n_estimators and learning_rate.
- Parameters to prevent overfitting, such as lambda_l1, lambda_l2, bagging_fraction and feature_fraction.

I split the bothdatasets into the train set and valid set and the hyperparameters I set are as below. Then I implemented ten trials, as seen in Figure 18(Take the bank dataset as an example). When the number of trials is greater than four, the best score of this model is almost unchangeable, at about 0.79. The hyperparameters tuned finally are shown in Tables 1 and 2. The roc_auc_score for the bank dataset on the validatation set of Kaggle is 0.81043.

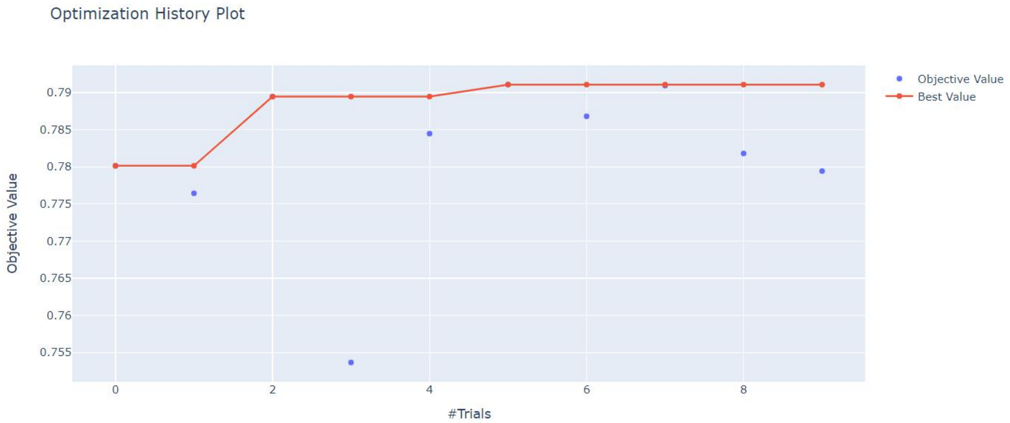


Figure 18: The scores of different trials

Table 1: Bank dataset

Hyperparameter	Value List	Tuned Value
learning_rate:	loguniform(1e-4, 1e-1)	0.014
lambda_l1:	loguniform(1e-8, 10)	0.123
max_depth:	int(3, 16, step=1)	4
num_leaves	int(2, 50, step=1)	28
feature_fraction	uniform(0.4, 1)	0.952
bagging_fraction	uniform(0.4, 1)	0.818
bagging_freq	int(1, 7, step=1)	7

Table 2: Store dataset

Hyperparameter	Value List	Tuned Value
learning_rate:	loguniform(1e-4, 1e-1)	0.009
lambda_l1:	loguniform(1e-8, 10)	1.629
max_depth:	int(3, 16, step=1)	3
num_leaves	int(2, 40, step=1)	8
feature_fraction	uniform(0.4, 1)	0.823
bagging_fraction	Uniform(0.4, 1)	0.854
bagging_freq	int(1, 7, step=1)	3

6.2.2 Random Forest Classifier

Table 3: Bank dataset

Hyperparameter	Value List	Tuned Value
n_estimators:	[200,300,400]	300
max_features:	[6,7,8,9]	8
max_depth:	[5,10,15,20]	10
calss_weight:	[“balanced”,”balanced_subsample “]	balanced

Then we fit the model with the optimized value of the parameter on the train set and get the accuracy of the Random forest on the validation set, AUC: 0.78, training set AUC: 0.85.

Table 4: Store dataset

Hyperparameter	Value List	Tuned Value
n_estimators:	[300,400,500]	400
max_features:	[6,7,8,9]	8
max_depth:	[3,4,5,6,7]	5
calss_weight:	[“balanced”, ”balanced_subsample “]	balanced

Then we fit the model with the optimized value of the parameter on the train set and get the accuracy of the Random forest on the validation set, AUC: 0.85, training set AUC: 0.86.

6.2.3 Stacking Classifier

We implemented the simple structured stacking classifier for both datasets. In the bank dataset, we used one random forest classifier and one logistic regression classifier as the base models and another logistic regression classifier as the meta model. As for the store dataset, we implemented one random forest classifier, one decision tree classifier and one gradient boosting classifier as the base models and used a logistic regression classifier as the meta model. The hyperparameters we are going to tune in this section are fundamentally the hyperparameters of each base and meta models. As we discussed in the earlier section, the performance of the stacking classifier basically relies on the performance of each model and their coordination, thus it is important to tune them well in this section. The model below shows the details about the tuning part of our stacking classifier

Table 5: Bank and Store datasets

Dataset	Model	Hyperparameter	Value List	Tuned Value
bank	RF	n_estimators max_depth	[300,350,400] [5,10,15,20]	400 10
bank	LR	C	[1,3,5,7]	7
store	RF	max_features	[4,8,16]	8
store	DTree	max_leaf_nodes	[35,45,55]	55
store	GB	learning_rate	[0.001,0.01,0.1]	0.01

7. Results

7.1Model Results and Evaluation

In the bank and store dataset, the machine learning algorithms are Model Stacking, LightGBM, Random Forest, Decision Tree and Logistic Regression. After tuning hyperparameters on the training data set by using 10-fold cross-validation method. And then, we evaluate model performance on a test set, that can ensure algorithms meet requirements of business goals.

The metric is quality measurement to measure the accuracy of machine learning algorithms. In the bank dataset, the metrics we use are training set AUC, best CV score and kaggle testing score. In the store dataset, the metrics we use are AUC (training and test set), accuracy (test set), Recall (test set), Precision (test set), and F1 score (test set). Model Results are shown on below tables.

Table 6: Bank Dataset

Model	Training Set AUC	Best CV Score	Kaggle validation Score
Model Stacking	0.83	0.78	0.799
LightGBM	0.83	0.79	0.810
Random Forest	0.85	0.78	0.807
Decision Tree	0.78	0.77	0.784
Logistic Regression **(Benchmark)**	0.77	0.76	0.775

Table 7: Store Dataset

Model	Training Set AUC	Testing Set AUC	Testing Set Accuracy	Testing Set Recall	Testing Set Precision	Testing Set F1
Model Stacking	0.862	0.854	0.853	0.337	0.636	0.441
LightGBM	0.848	0.848	0.772	0.723	0.408	0.521
Random Forest	0.854	0.854	0.739	0.824	0.381	0.521
Decision Tree	0.86	0.840	0.750	0.784	0.387	0.519
Logistic Regression **(Benchmark)**	0.84	0.847	0.710	0.847	0.356	0.501

As shown in the Table 6, the best model with the highest score is LightGBM, followed by Random Forest and Stacking. The difference in AUC between the three algorithms is very small. It is not surprising that the three of them are the best, as they are all ensemble models whose performance is better than other single models in most cases. On the other hand, Logistic Regression has the lowest AUC score of 0.775, since this benchmark model can only capture linear patterns, but for some non-linear relationship which is difficult for Logistic Regression.

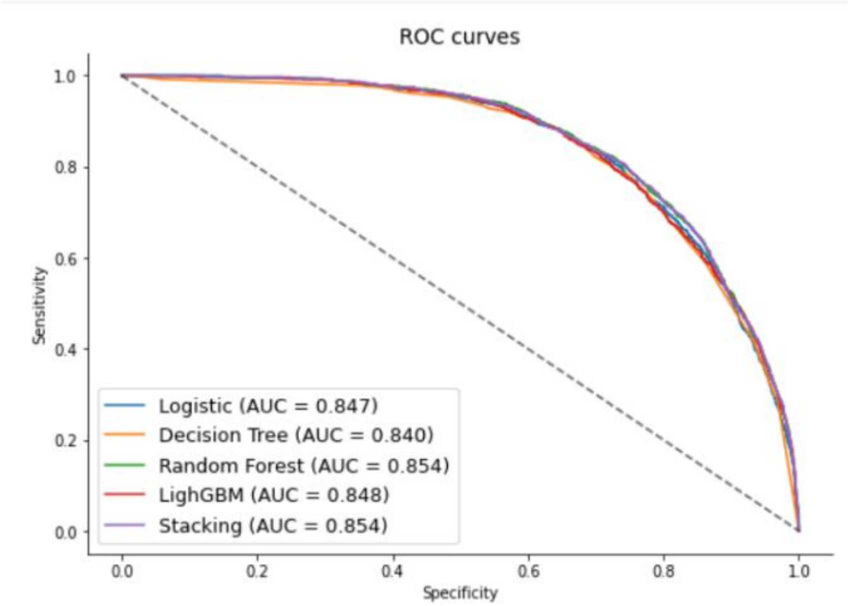


Figure 19: ROC curves of different models

As seen in Figure 19, the AUC score of five algorithms is almost the same at about 0.85. But there are some differences for other metrics.

The accuracy of the Stacking model is the highest at 0.853, as shown in Table 7, but its recall score is the lowest, indicating only approximately 33% people who responded to the promotion were predicted correctly. This result also suggests Precision score is higher than other models because the two metrics are always negatively correlated. While other classifiers, such as Random Forest and Logistic Regression, whose Precision scores all are lower than 0.4, which means only less than 40% people that we correctly identify responding to the promotion out of all the people responding to it. In this case, F1-score plays an important role in evaluating the performance of a model, which is a trade-off of Precision and Recall. From the table above, we can see that Random Forest and LightGBM obtained the highest F1-score of 0.521, while model stacking has the lowest F1-score.

Because this dataset is imbalanced, accuracy may not be a good metric to evaluate a model. In this case, combining all metrics, Random Forest performed the best.

8. Task 2 Discussion

What types of customers are more responsive to marketing campaigns? (Answer to the Task 2)

8.1 SHAP

SHAP Summary plot (in section 8.1.1) shows the information about the importance of variables. This plot lists the variables in descending order by importance.

And if we change the plot type, we can detect the information about positive and negative relationships of the predictors with the target variable (in section 8.1.2).

8.1.1 SHAP summary plot:

Bank (RandomForest):

We can see from Figure 20 that the most important features of the Bank data set include contact, balance, housing, poutcome and so on.

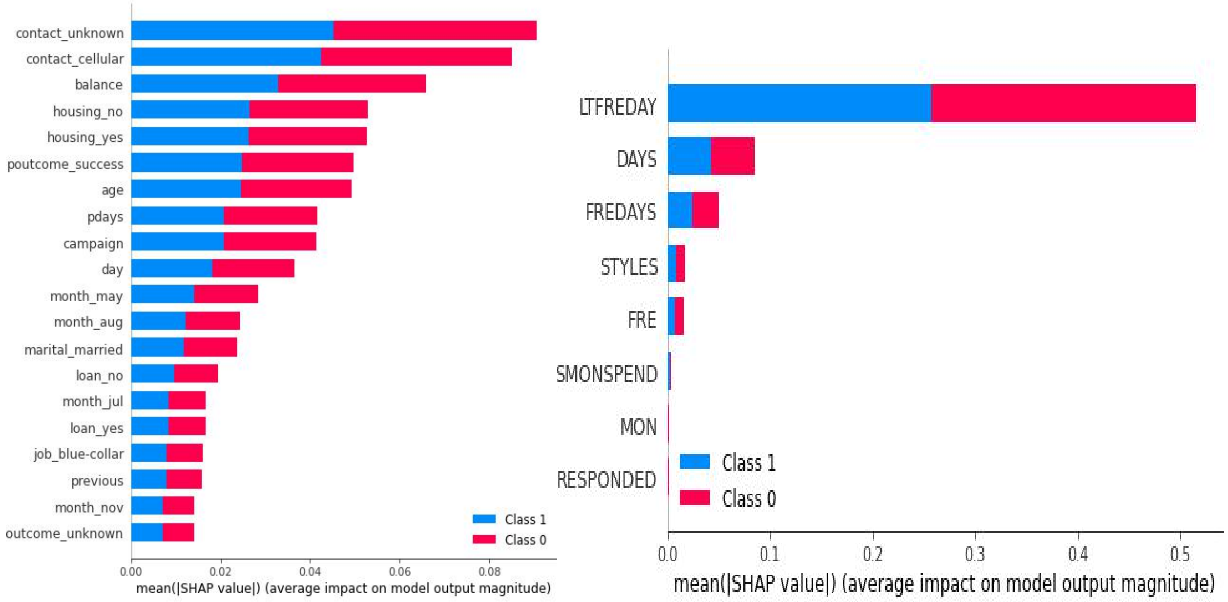


Figure 20 & 21： Feature importance for Bank dataset & Feature importance for Store dataset

Store (RandomForest):

As shown in Figure 21, the most important features of the Fashion Store data set include LTFREDAY, DAYS, FREDAYS and so on.

8.1.2 Another type of SHAP summary plot(Figures 22 and 23):

Bank (RandomForest):

For balance, the result indicates that if the balance is relatively low, the balance feature of this sample is expected to have a negative impact on the probability of success of marketing campaigns. And if the balance is relatively high, the balance feature of this sample is expected to have a positive impact on the probability of success of marketing campaigns.

For housing_no and housing_yes, the result indicates that if someone has housing loans, the housing feature of this sample is expected to have a negative impact on the probability of success of marketing. And if someone does not have housing loans, the housing feature of this sample is expected to have a positive impact on the probability of success of marketing.

For poutcome_success, the result indicates that for someone whose outcome of the previous marketing campaign is success, the poutcome_success feature is expected to have a positive impact on the success probability in the marketing campaign. And if the outcome is not success (fail or unknown), the poutcome_success feature is expected to have a negative impact on the probability of success in the marketing campaign.

For age, the result indicates that if the age is medium or relatively low, the age feature is expected to have a negative impact on the probability of success of marketing, but not significant, as there are some low value points that have a positive impact on the output. And if the age is relatively high, the result is not significant as well, as most of them have positive impact on the probability of success of marketing, while some of them have relatively low negative impact on the output.

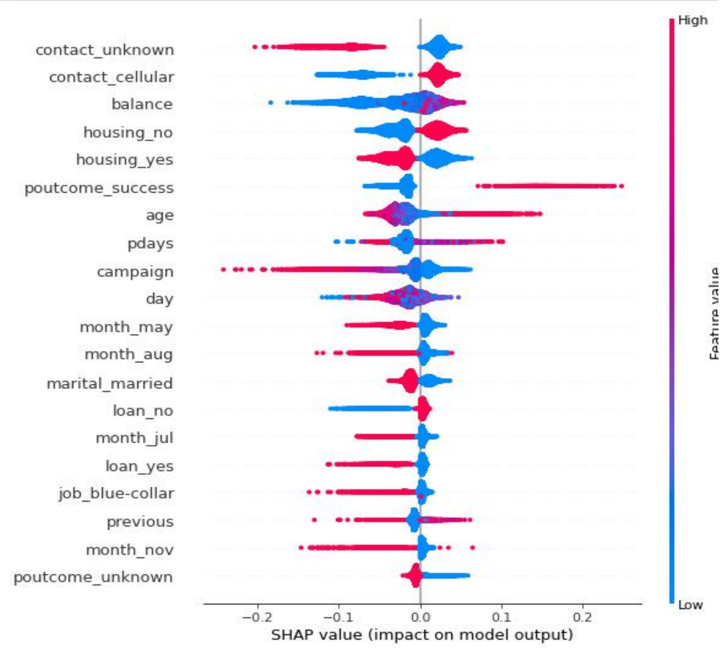


Figure 22: SHAP value for the Bank dataset

Fashion store (RandomForest):

For LTFERDAY (Lifetime average of days between visits), if the value is very high, then this feature is expected to have a negative impact on the output. And if the value of LTFERDAY is not very high, the impact of the feature is not significant (both in positive and negative).

For DAYS (Number of days the customer has been on file), if the value is high, then this DAYS feature is expected to have a positive impact on the probability of success. And if the value is low, the feature seems to have a negative impact on the output.

For STYLES (Total number of individual items purchased by the customer.), if the value is relatively low, then this STYLES feature is expected to have a negative impact on the probability of success of the marketing campaign. And if the value is high, the impact is not significant.

For FRE (Number of purchase visits), the performance is similar to the STYLES, in addition to this, the high value of FRE can have a little positive impact on the output.

For SMONSPPEND (Amount spent in the past six months), if the value is high, then this SMONSPEND feature is expected to have a little positive impact on the success probability of a marketing campaign.

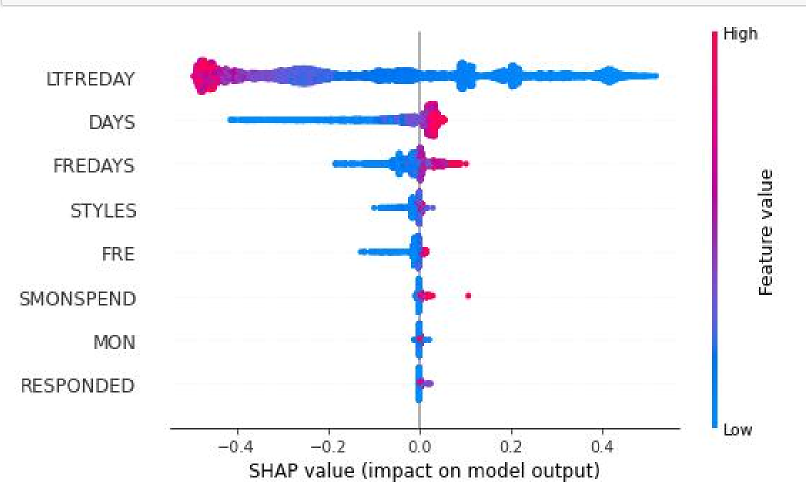


Figure 23: SHAP value for the Store dataset

8.2 What types of customers are more responsive to marketing campaigns?

After data mining and data analysis on bank and store dataset. We found customers who are loyal, high income, and have high purchasing ability are more responsive to marketing campaigns.

After data mining we found that loyal customers are the most valuable asset for a company or a brand. Customer's consumption habits have been developed on a certain company, products, or services. They will repeatedly purchase services and products. If suitable marketing campaigns are organized for this type of clients, there is a high probability of getting a positive response. Also, if the customer participates in previous marketing campaigns, who has a large possibility to be responsive to current and future marketing campaigns.

More importantly, high income customers are more responsive to marketing campaigns, because they could afford goods and services that the marketing campaigns are trying to sell to potential clients. Compared to low-income groups, even they want to be responsive to marketing campaigns, but they cannot bear the cost of goods and services.

Moreover, customers who have high purchasing ability; they are not sensitive to price. This type of customer is willing to spend a lot of money on goods and services. If marketing campaigns attract their interest and satisfy their demand, like pursuit high-end products and services, personality. To some extent induce customer's desire to buy, in order to increase the probability of success of marketing campaigns.

9. Discussion and Conclusion

In task 1 we have successfully implemented five different models in each of the two given datasets with complete pre-processing procedures and final evaluation and comparison of all models. By the comparison we observed that the best model for the bank dataset is the LightGBM classifier with the stacking classifier following slightly behind. As for the store dataset, it is obvious to see that the stacking classifier is the winner in all five models since it has the highest test set accuracy and the test set auc value. In the future, we would like to implement a more complex stacking classifier, for example using the double-layered stacking classifier rather than the simple structured single-layered model we currently use.

In task 2, we use SHAP to extract the most influential features that are related to our business question: which types of customers are more responsive to marketing campaigns? For example, bank dataset the most influential features: contact, balance, housing and poutcome; store dataset the most influential features are LTFREDAY, DAYS, and FREDAYS. On the aspect of business insights, we found customers who are fidelity, have high demand to consume goods and service, and earn high amounts of wage. Those types of customers are more willing to be responsive to marketing campaigns. In this process, we found that the number of features of the store data set were deleted too much in the feature selection section. After feature selection, the remaining features have high similarity, which will lead to homogenized customer characteristics. In this case, we may not be able to summarize enough quantitative insights from the data that address the business question. For future improvement, it's necessary to retain a certain number of features in the feature selection section.

10. Appendix

10.1 Project Management

Project management is a framework that can help team projects achieve the best results. There is no fixed framework model that applies to all teams. Hence, each team needs to choose the appropriate project management methodology according to the actual situation [3]. However, a single project management methodology may have defects or unreasonable points in actual application. In this group project, we implemented a combination of the following project management methodologies.

1. **Agile Methodology**

In this group project, the agile project management methodology is mainly used. The agile methodology was created in 2001[4]. The core is to split the entire project into multiple tasks. Some tasks can be performed simultaneously and then continuously improved according to requirements. Each member of this group is responsible for group projects in four disciplines at the same time, and the time periods of these four group projects are close. Team members cannot concentrate all their energy and time on a single project and cannot match each other's time in the early stage. Therefore, an efficient and flexible agile methodology has become our preferred project management methodology [5]. To complete the group project more efficiently, we divide the project into multiple stages and distribute them to different team members fairly.

- Preprocessing + EDA + Feature Engineer
  Two of the team members are respectively responsible for data preprocessing, exploratory data analysis and feature engineering tasks for the two datasets, laying a solid foundation for all the team members to understand the full picture of these two datasets and subsequent modeling. Two team members use different methods to complete the same task on the two datasets at the same time, which greatly saves time and allows more time to update and improve later.
- Algorithms + Evaluation
  Each of the five team members selected an algorithm to model, train, and predict the two datasets at the same time. The prediction results of five algorithms using the bank dataset are uploaded to Kaggle. The five algorithms are adjusted repeatedly according to the performance evaluation obtained on Kaggle. At this point, the advantages of the flexibility of agile methodology have been demonstrated. We take the feedback received on Kaggle as customer feedback, make the performance of the model better as the customer's request, then iterate the project process with customer’s request as the goal. Algorithms that need to improve performance will be further studied by the responsible team members. For algorithms which do not need to be improved, the corresponding team members continuously work for other tasks.
- Report
  The five team members wrote reports on their respective contents at the same time and adjusted the report contents according to the changed model contents in a timely manner. Because each team member is familiar with the content of their respective code, everyone can quickly and accurately complete the content of the report. This avoids wasted time and wrong expressions caused by unfamiliarity.

2. Waterfall Methodology [3]

In the final summary process, due to the connection between the codes, we used the waterfall project management methodology to complete the final version of each step in order. Firstly, we determined the methods used in the preprocessing, EDA, and feature engineering. In the second step, we model, train, and evaluate the best model based on the data processed in the previous step. In the end, a complete code file was obtained. In the report part, we summarized according to the code order. Since each team member writes their own content separately, there may be duplication in content. After handling the duplicate content and unifying the format, we got the final and complete report.

10.2 Contribution Statement

In this group assignment, we split the work into 20% partition for each group member so that we are equally contributed to the assignment. As the group member who is responsible for this section, I would say that all members are well-performed in their allocated tasks by the team leader from the beginning to the very end of this project.

10.3 Figures in EDA and Feature Engineering section

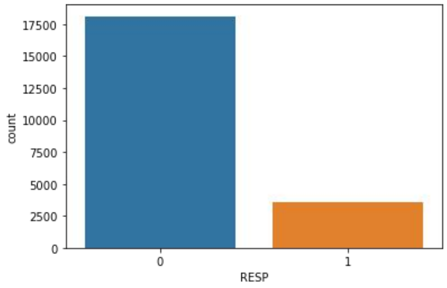


Figure 1: Distribution Chart of RESP Feature

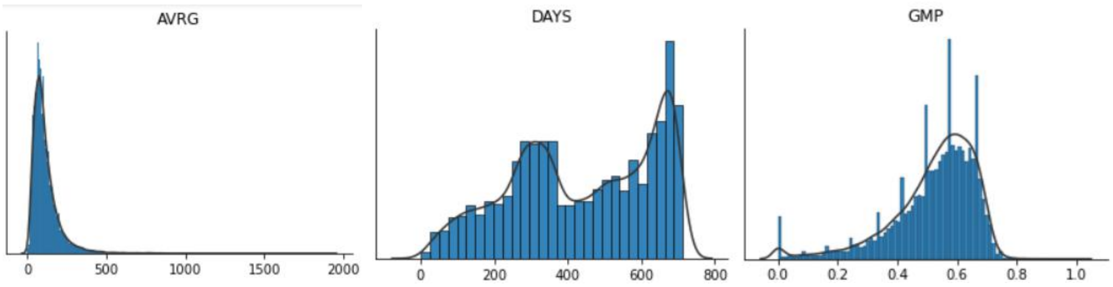


Figure 2: Univariate Chart of AVRG, DAYS, and GMP features

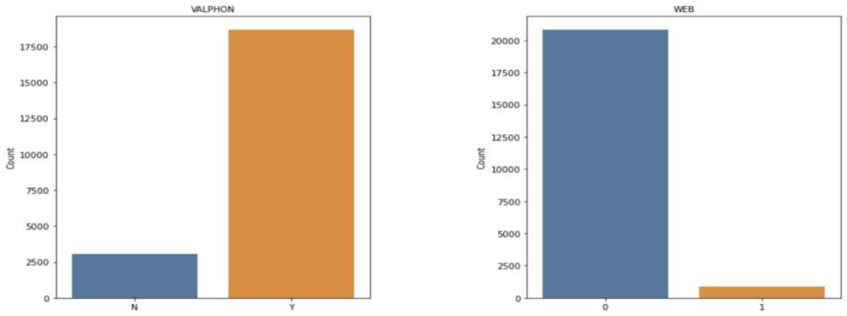


Figure 3: Univariate Chart of VALPHON and WEB features

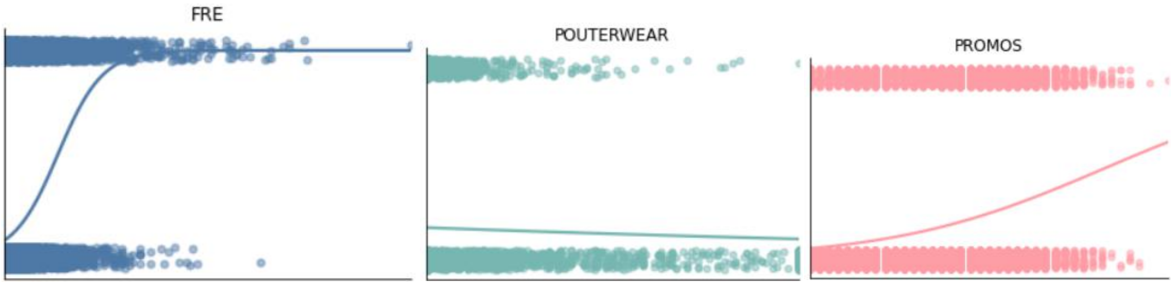


Figure 4: Bivariate Chart of FRE, POUTERWEAR, and PROMOS features

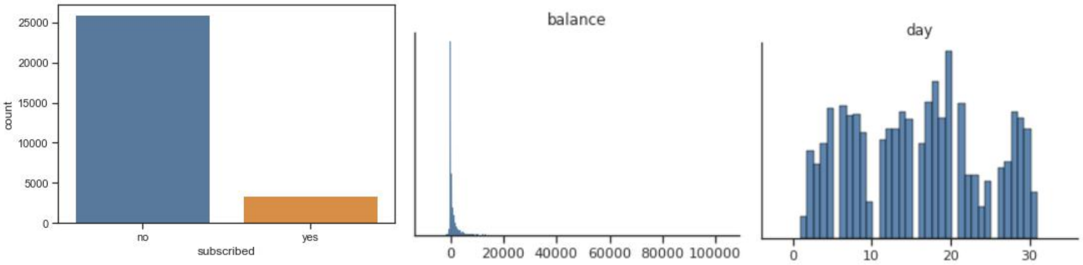


Figure 5: Univariate Chart of Output Subscribed, balance and day Features

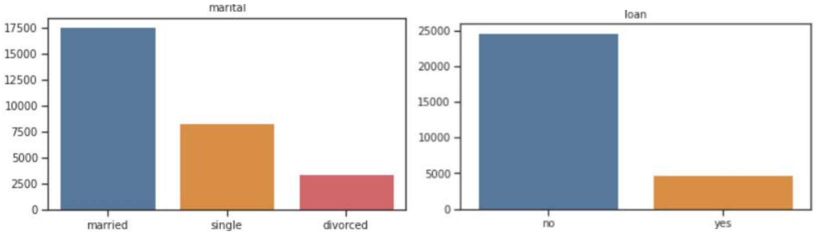


Figure 6: Univariate Chart of marital and loan features

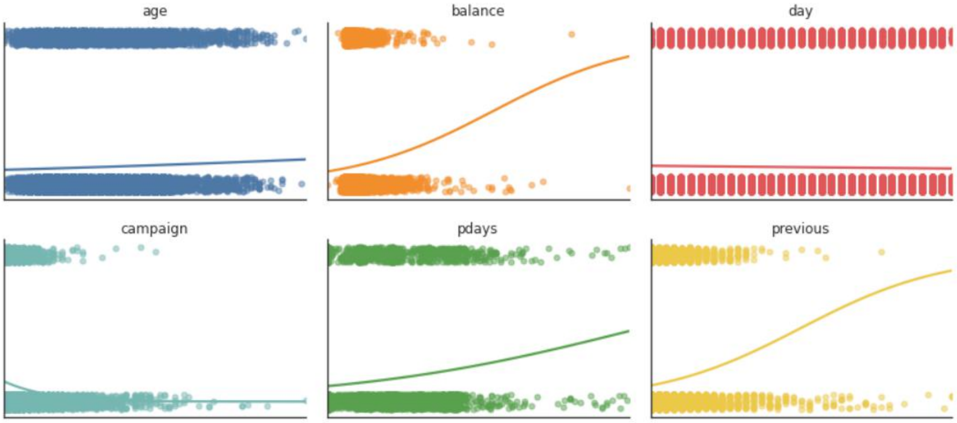


Figure 7: Bivariate Chart of numerical features in Bank Dataset

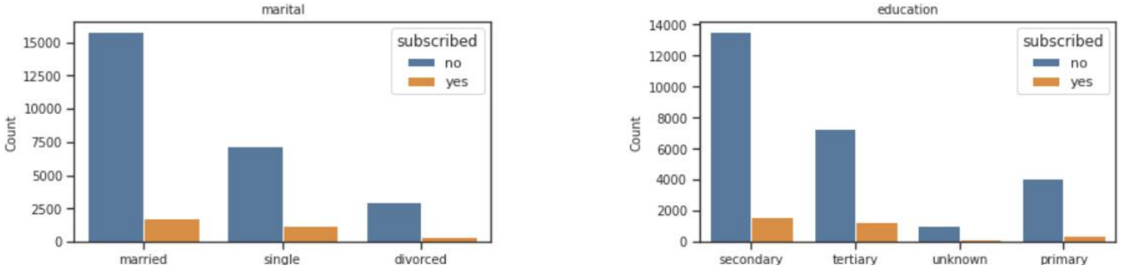


Figure 8: Bivariate Chart of marital and education Features

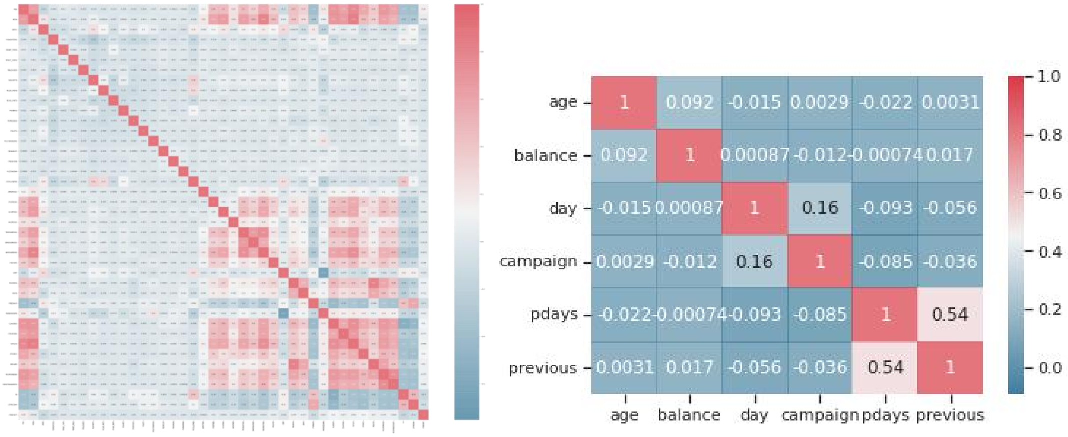
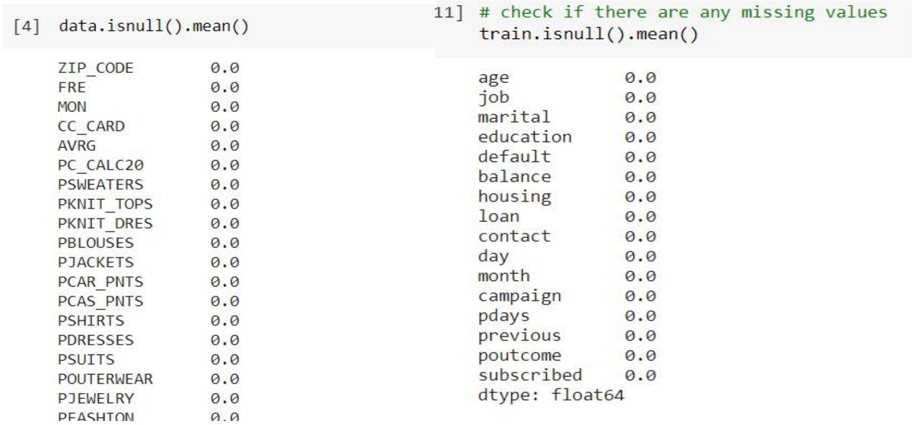


Figure 9: Correlation Matrix of Store Dataset and Bank Dataset



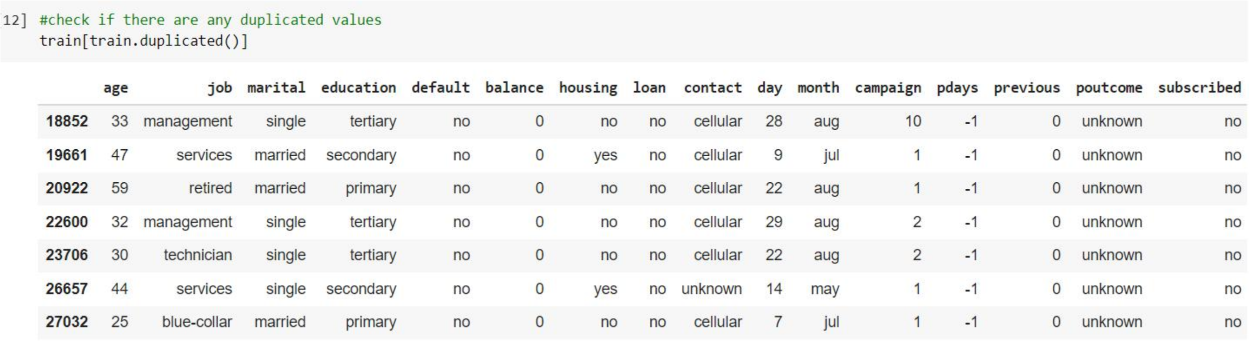
```
[4]  data.isnull().mean()

     ZIP_CODE        0.0
     FRE             0.0
     MON             0.0
     CC_CARD         0.0
     AVRG            0.0
     PC_CALC20       0.0
     PSWEATERS       0.0
     PKNIT_TOPS      0.0
     PKNIT_DRES      0.0
     PBLOUSES        0.0
     PJACKETS        0.0
     PCAR_PNTS       0.0
     PCAS_PNTS       0.0
     PSHIRTS         0.0
     PDRESSES        0.0
     PSUITS          0.0
     POUTERWEAR      0.0
     PJEWELRY        0.0
     PFASHION        0.0
```

```
11]  # check if there are any missing values
     train.isnull().mean()

     age           0.0
     job           0.0
     marital       0.0
     education     0.0
     default       0.0
     balance       0.0
     housing       0.0
     loan          0.0
     contact       0.0
     day           0.0
     month         0.0
     campaign      0.0
     pdays         0.0
     previous      0.0
     poutcome      0.0
     subscribed    0.0
     dtype: float64
```

Figure 10: Outliers Observation for Both Datasets



```
12] #check if there are any duplicated values
    train[train.duplicated()]
```

	age	job	marital	education	default	balance	housing	loan	contact	day	month	campaign	pdays	previous	poutcome	subscribed
18852	33	management	single	tertiary	no	0	no	no	cellular	28	aug	10	-1	0	unknown	no
19661	47	services	married	secondary	no	0	yes	no	cellular	9	jul	1	-1	0	unknown	no
20922	59	retired	married	primary	no	0	no	no	cellular	22	aug	1	-1	0	unknown	no
22600	32	management	single	tertiary	no	0	no	no	cellular	29	aug	2	-1	0	unknown	no
23706	30	technician	single	tertiary	no	0	no	no	cellular	22	aug	2	-1	0	unknown	no
26657	44	services	single	secondary	no	0	yes	no	unknown	14	may	1	-1	0	unknown	no
27032	25	blue-collar	married	primary	no	0	no	no	cellular	7	jul	1	-1	0	unknown	no

Figure 11: Observation of Duplicated Values in the bank dataset

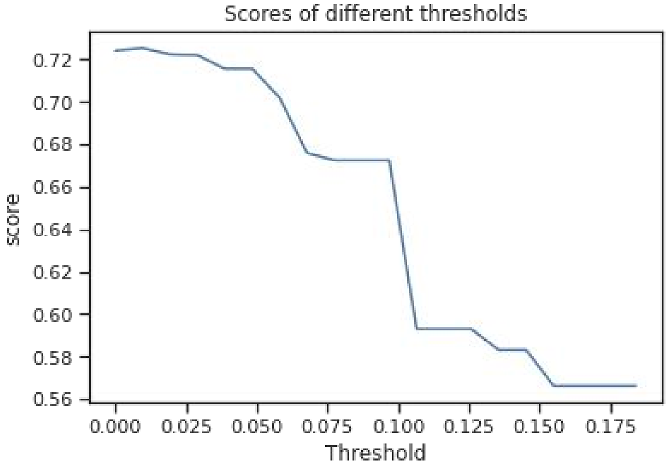


Figure 12: Scores of different thresholds for the bank dataset

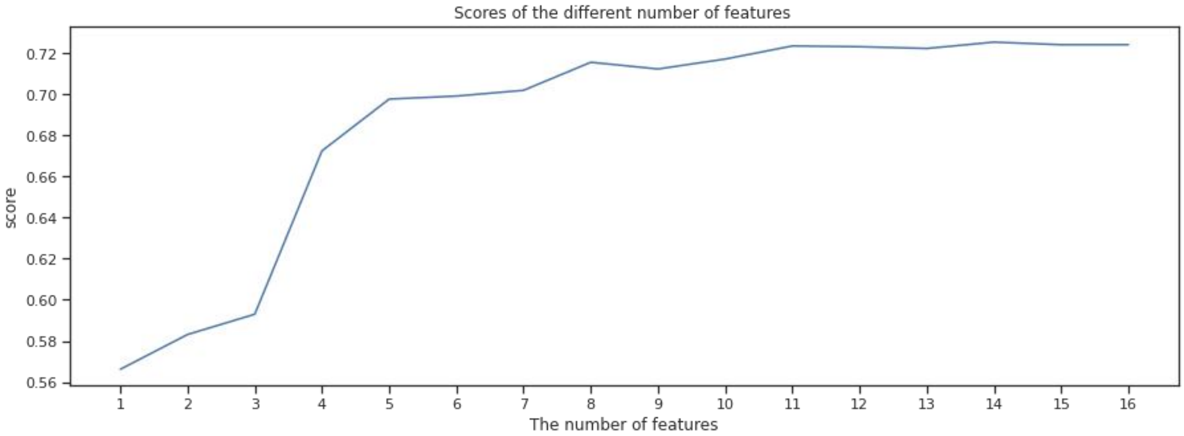


Figure 13: Scores od different number of features for the bank dataset

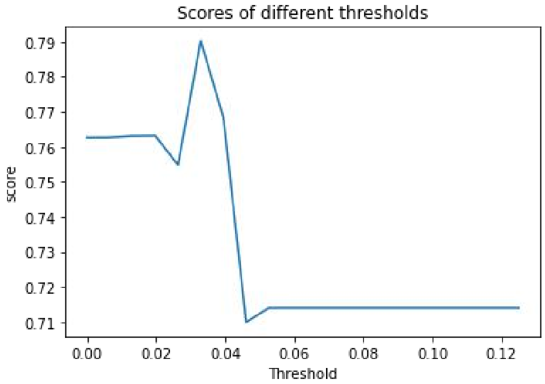


Figure 14 Scores of different thresholds for the store dataset

References

1. “scikit-learn: Machine Learning in Python,” *scikit*. [Online]. Available: https://scikit-learn.org/stable/index.html. [Accessed: 11-Oct-2021].
2. “Python API reference,” *Python API Reference - xgboost 1.6.0-dev documentation*. [Online]. Available: https://xgboost.readthedocs.io/en/latest/python/python_api.html. [Accessed: 21-Oct-2021].
3. “Why is project management important?,” *Project Management Methodologies - Everything You Need To Know*. [Online]. Available: https://www.teamwork.com/project-management-guide/project-management-methodologies/. [Accessed: 11-Nov-2021].
4. “What is Agile Methodology in project management?,” *Versatile & Robust Project Management Software*. [Online]. Available: https://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management/. [Accessed: 11-Nov-2021].
5. “What's the agile methodology and how can it benefit your enterprise?,” *The Agile Methodology and Its Benefits | WAM*. [Online]. Available: https://www.wearemarketing.com/blog/what-is-the-agile-methodology-and-what-benefits-does-it-have-for-your-company.html. [Accessed: 11-Nov-2021].