Member Name	**Percentage of Contributions**	**Justification**
He Shan	20%	Feature engineering, Hyperparameter Tuning
Lyu Yiyang	20%	Exploratory data analysis, Data Preprocessing, Data Visualization
Peng Yichao	20%	Model implementation, Hyperparameter Tuning & Model evaluation.
Wang Jiaheng	20%	Experiments on resampling; Insight on business intelligence; Model comparison
Zhang Yiqun	20%	Experiments on encoding; Insight on data masking.

Data Mining Project Report

Group 4:
He Shan
Lyu Yiyang
Peng Yichao
Wang Jiaheng
Zhang Yiqun

Table of Content

Research Background

Problem Statement

The bank needs to increase the number of deposits to ensure the stability of the loan business and reduce risks, enhancing its own profitability. Therefore, contacting clients by phone to promote its deposit subscription services is a major way for the bank to improve their business capabilities.The human resources of the bank in business expansion and the development of potential users are limited. Blindly selecting client groups will cause a waste of staff time and energy, thereby increasing unnecessary operating costs. On the other hand, the effect of improving business performance is not significant.

The dilemma faced by the bank xplained above illustrates the importance of effectively identifying potential client groups, the bank can use existing client data and the results of whether they participate in historical campaigns to
predict their interest in future campaigns. In the future campaigns, clients with high predicted interest can be given priority. Focusing on contacting such kinds of clients will increase deposit subscription rates significantly.

According to our problem-solving logic chain, it can be further expanded into a more specific project, such as designing a client selection system for the bank, which is based on the clients' personal characteristics information, including age, gender, bank account deposit balance and other relevant information, to evaluate the probability of subscribing to deposit campaign, as a reference when the bank selecting the contacting clients list. The system also has the function of automatically learning and improving the algorithm, which can continuously adjust the algorithm according to the clients data obtained from the newly held deposit campaign, thereby improving the accuracy of the evaluation result.

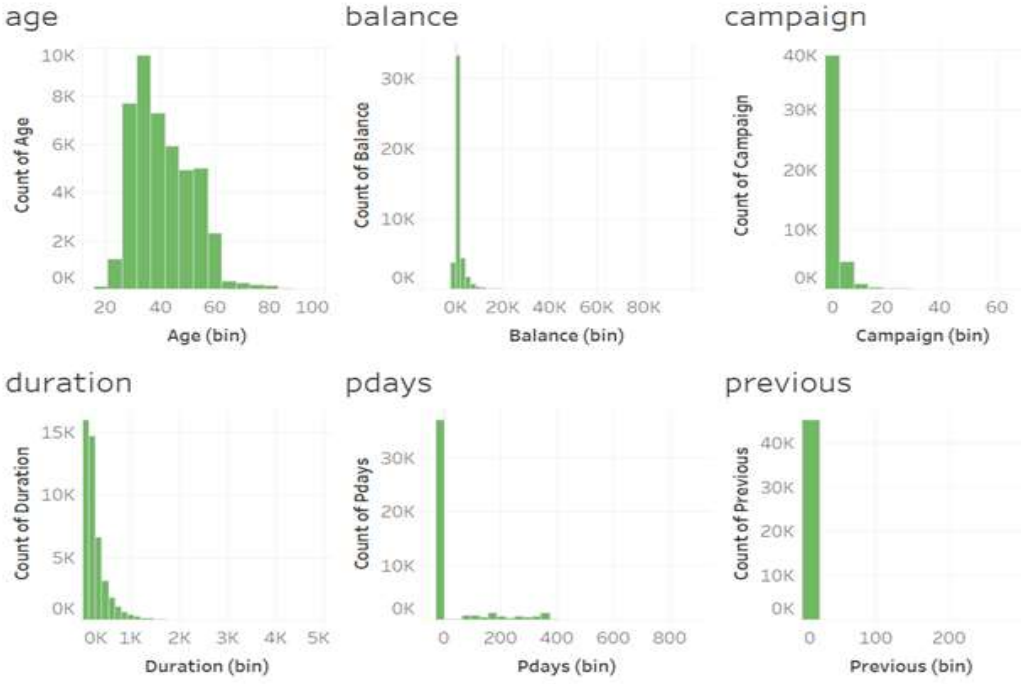
Data Set Description

The data is related to direct marketing campaigns of a Portuguese banking institution, ordered by date (from May 2008 to November 2010). The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, in order to know if the product (bank term deposit) would be ('yes') or not ('no') subscribed. The classification goal is to predict if the client will subscribe to a term deposit. Number of Instances is 45211 and number of attributes is 17(one output attribute).

Dataset Pre-processing

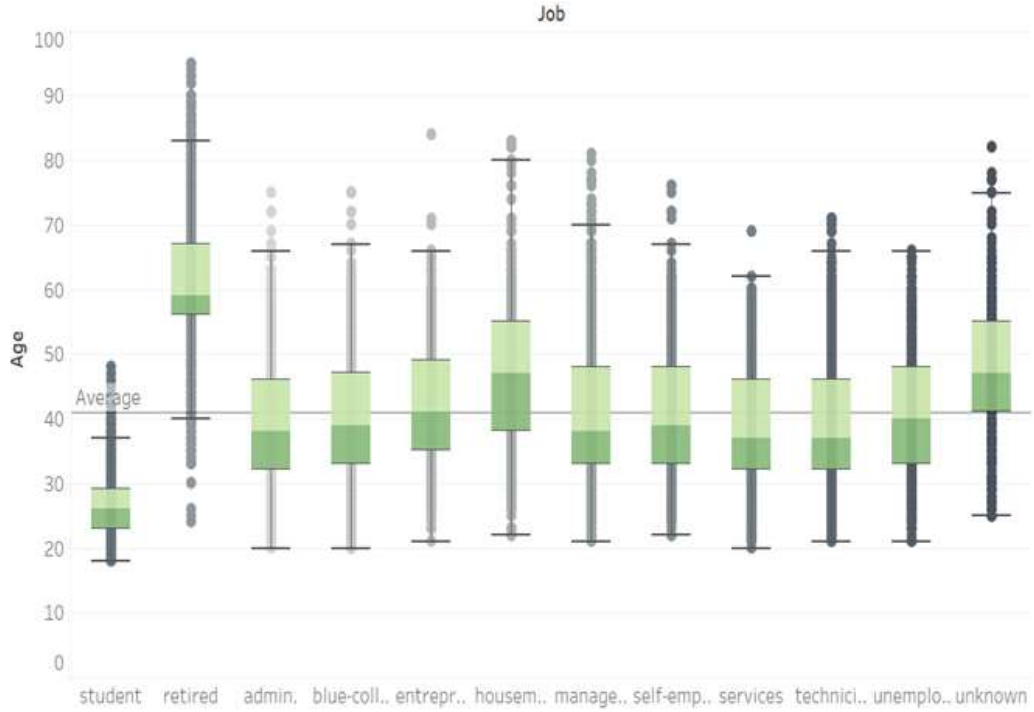
Data Exploration

Frequency Distribution Diagram of Numeric Variables



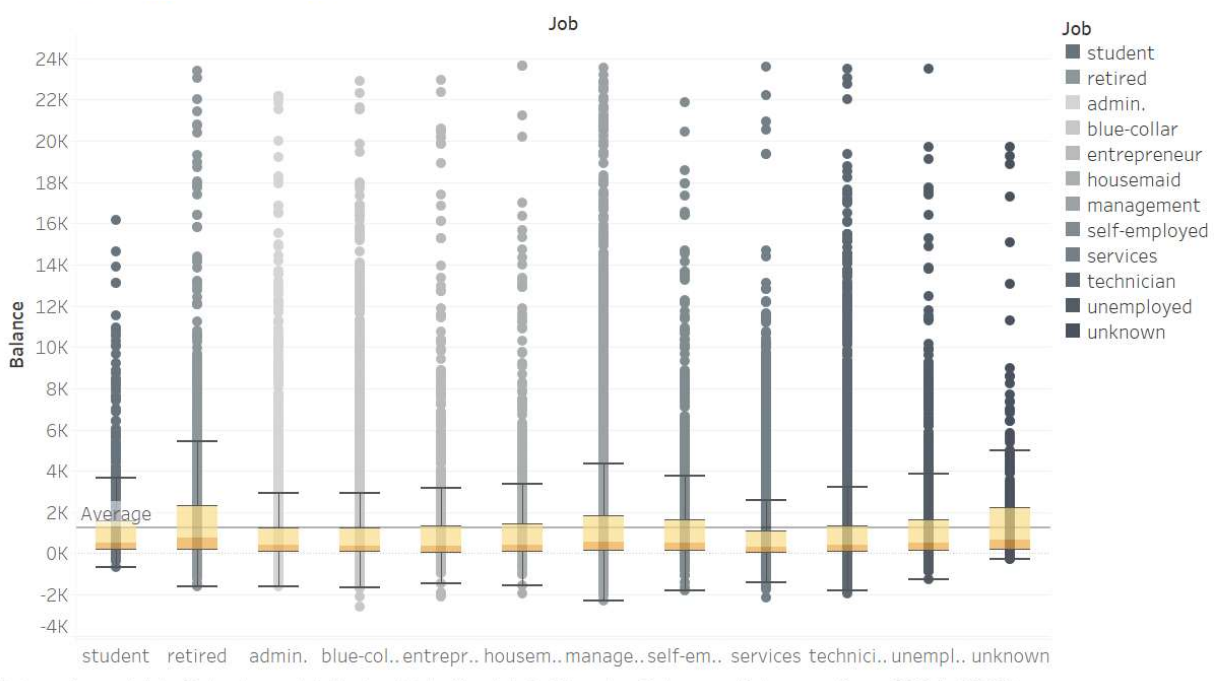
The variable frequency basically satisfies the law of the number decreasing as the value of the variable increases.The bank's target client groups are mostly young and middle-aged people aged 30 to 60, and the bank account balance level is concentrated within 5k. In the data set, the proportion of new clients (clients who have not participated in any activities and have not been contacted except this activity) is relatively large, about 80%.

Age-Occupation Distribution



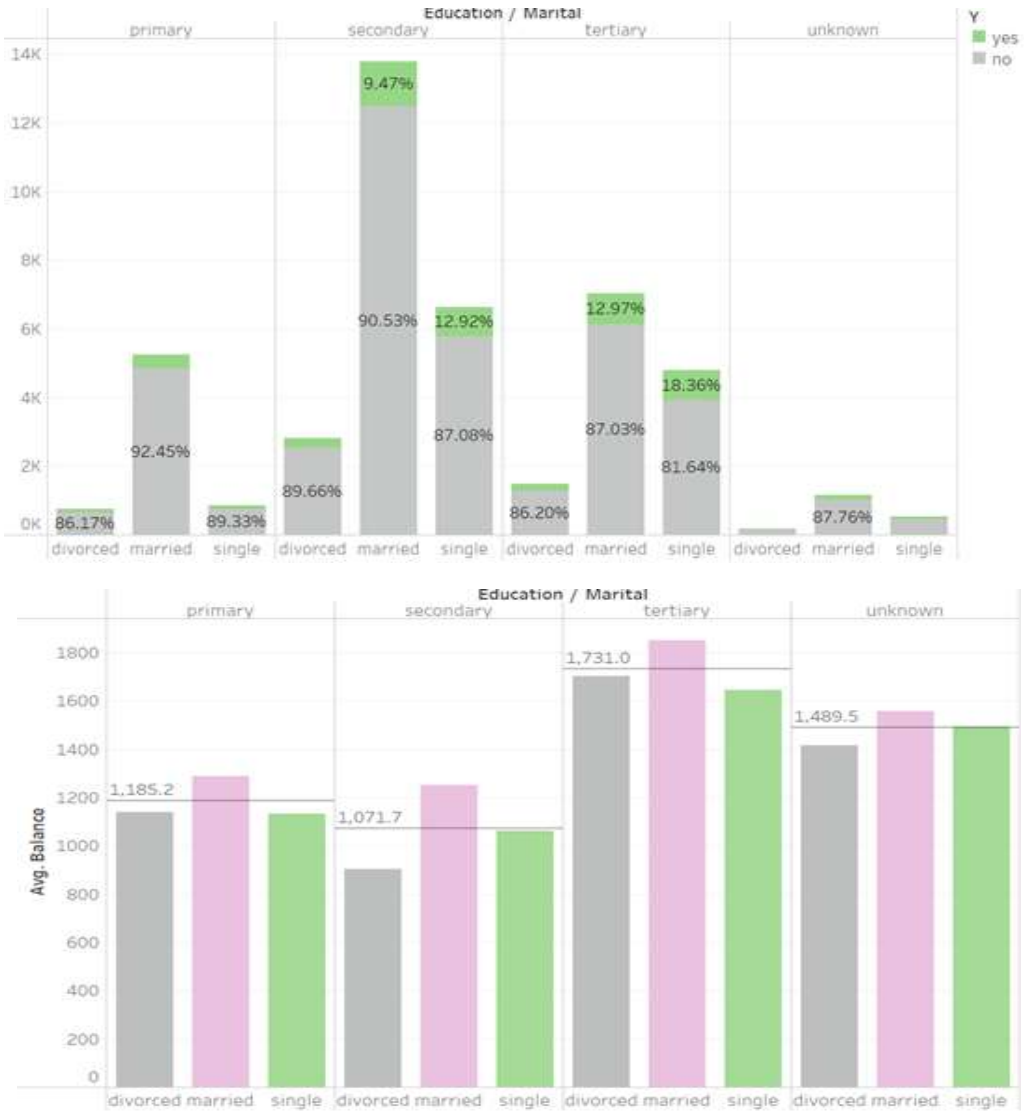
The box chart above can show the age distribution of different occupational groups, including group average age and extreme values.The overall age level of the student group is significantly lower than the overall average age. On the contrary, the overall age level of the retirement group is significantly higher than that. The age distribution levels of other occupational groups are relatively similar. Except for students and retirees， the correlation between occupation and age is not obvious. Besides, it is worth mentioning that the average age of the housemaid group is slightly higher.

Job and Balance Distribution



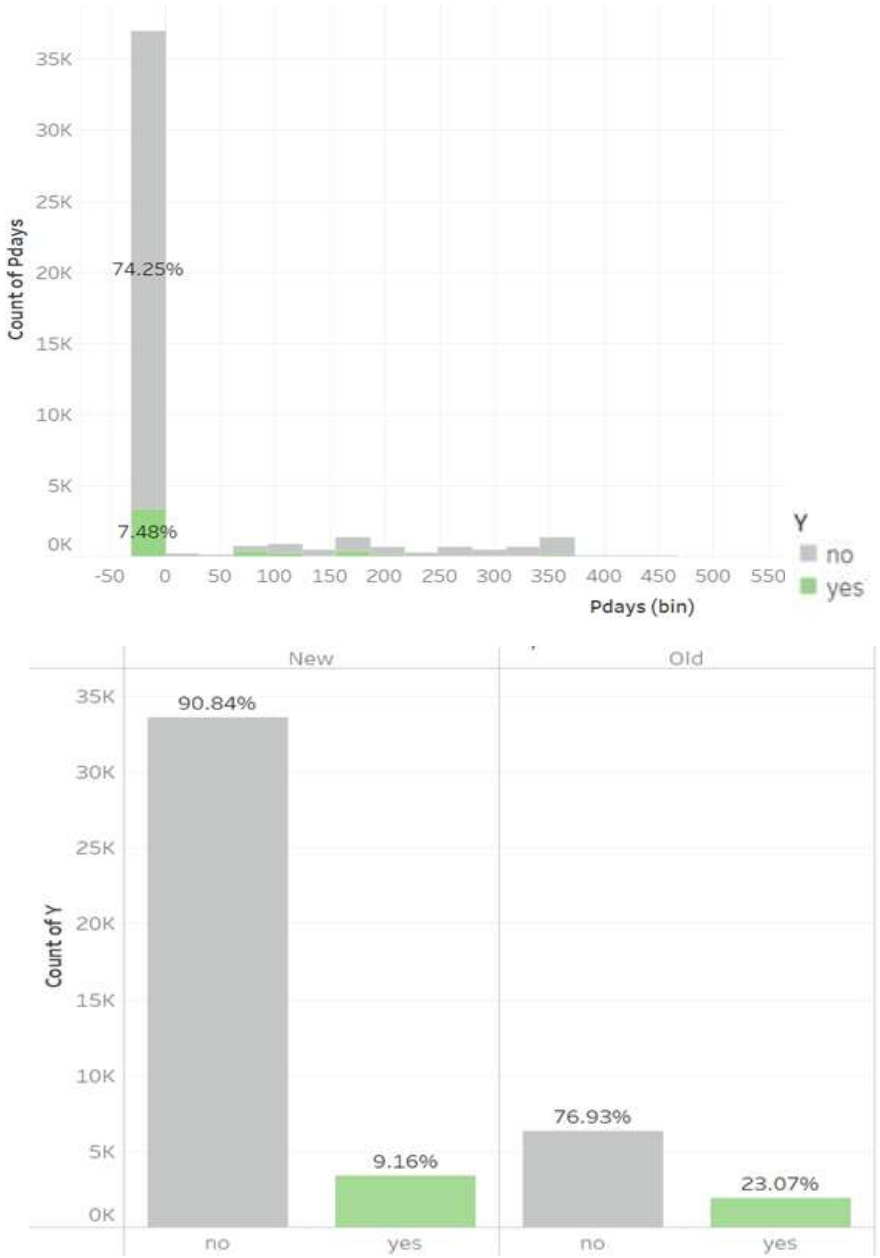
According to the classification of client groups by occupation, it can be found that the difference in the level of bank balance between people of different occupations is not obvious. Moreover, it can be seen from the distribution of the box plots that there are many extreme values and the data variance is larger. Therefore, it is not a reasonable way to judge a client's bank balance by the occupation.

Education, Marital Status and Bank Account Balance



The number of married persons is the largest, followed by the number of singles, and the least is the number of divorced persons. Generally speaking, the percentages of subscription deposits for married people and singles are relatively high, and with the improvement of education level, the percentage of subscriptions will increase slightly. Married people have the highest level of bank account balance, and their education level is positively correlated with the average balance level.

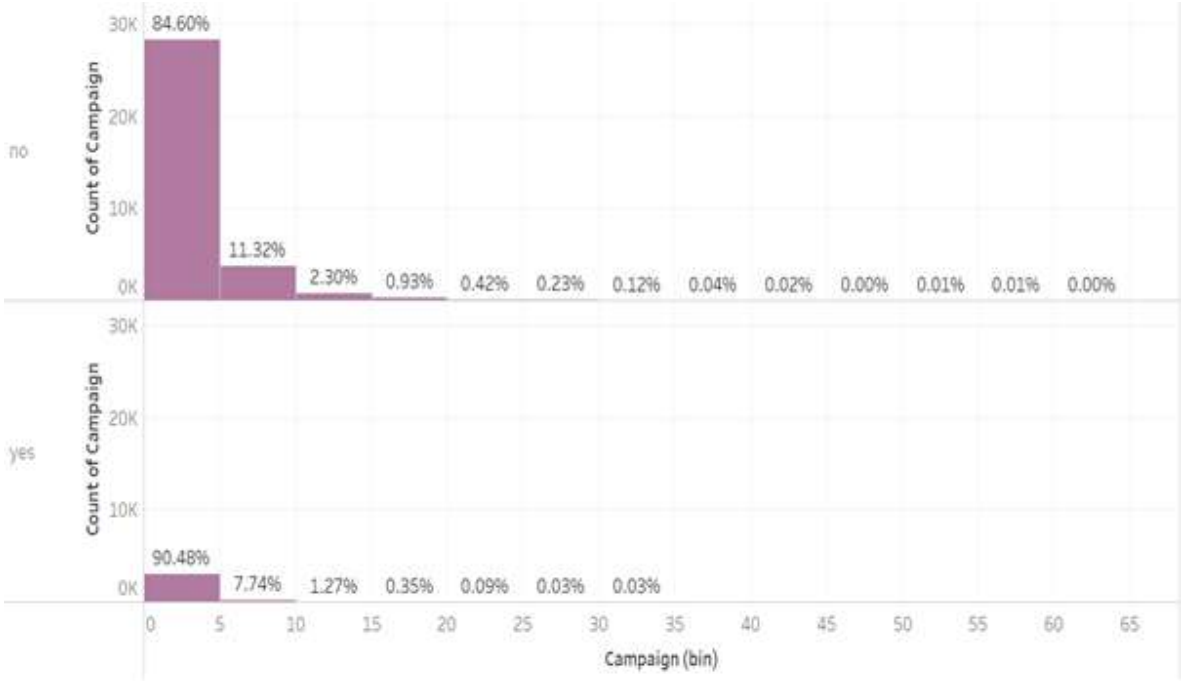
Proportion of New and Old Clients



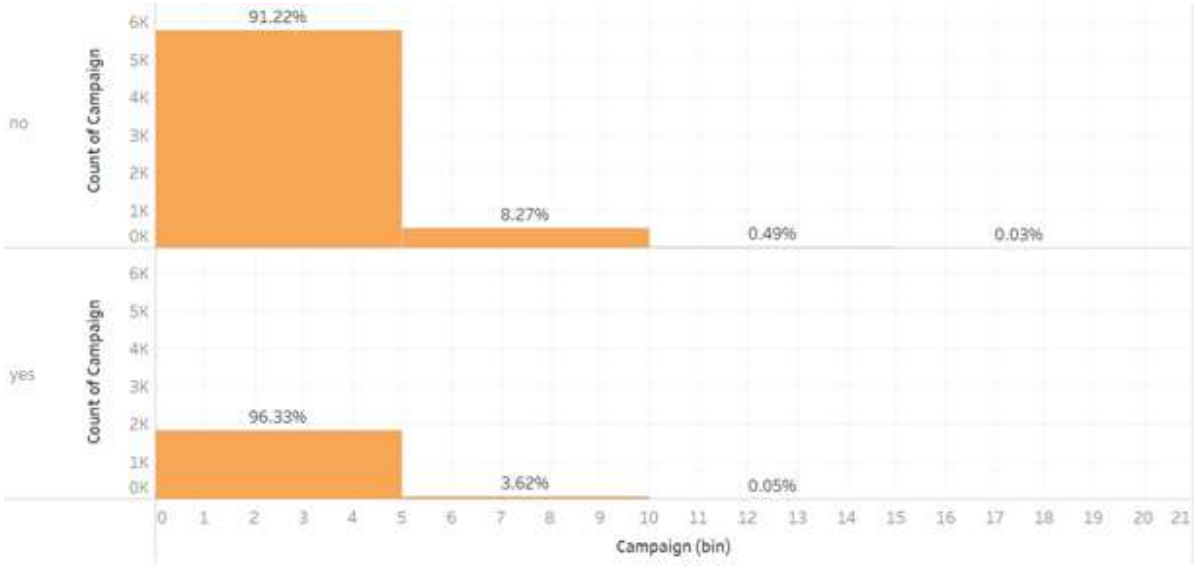
Pdays = -1 indicates that the client has never been contacted before this campaign and can be defined as a new client. The proportion of new clients is much higher than that of old clients, as high as 81.73%, but only about one in eleven of them successfully subscribed to deposit. Although the total number of old clients accounts for a small proportion of the total, the internal subscription rate is much higher than that of new clients, almost a quarter choose to subscribe.

Number of Contacts in This Campaign

New



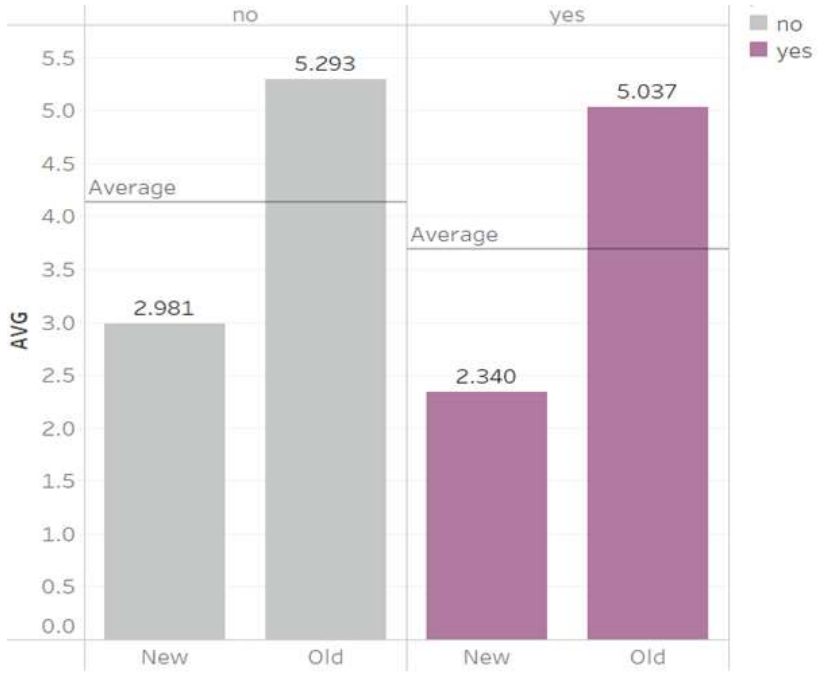
Old



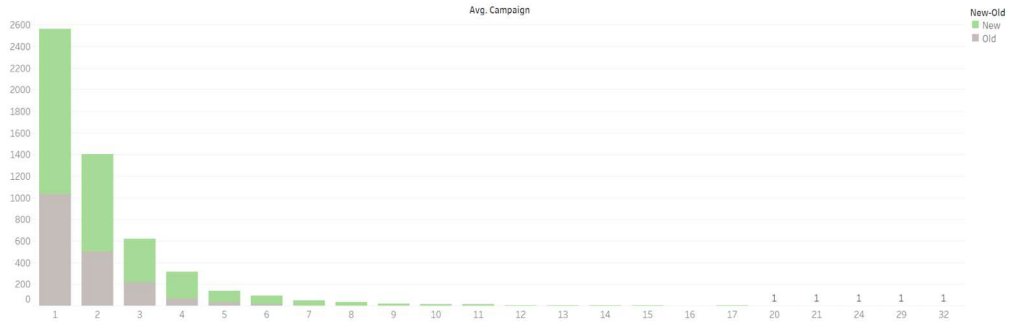
More than 90% of clients will give their final answer within five contacts whether they subscribe to this deposit campaign. For clients who are contacted for the first time in this campaign, the total number of contacts will be slightly more than that of regular clients. Overall, the number of

contacts within this campaign has no obvious influence on the final decision-making of clients. Increasing the number of contacts may not increase the likelihood of subscription.

Average Number of Contacts Per Campaign

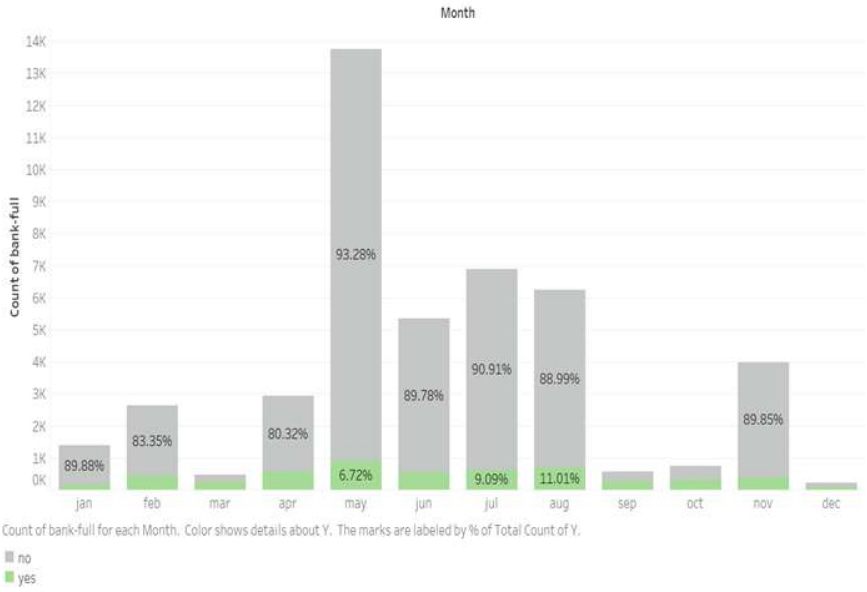


The average number of contacts between new and old clients for each campaign varies greatly. On average, old clients are contacted more than 5 times per campaign, but new clients are only contacted about twice. However, the gap between the average number of contacts per campaign among the clients who refused and accepted the subscription was not obvious, which further verified the previous view: blindly increasing the number of contacts for a certain client in a single campaign does not significantly increase this client's probability of the subscription deposit.



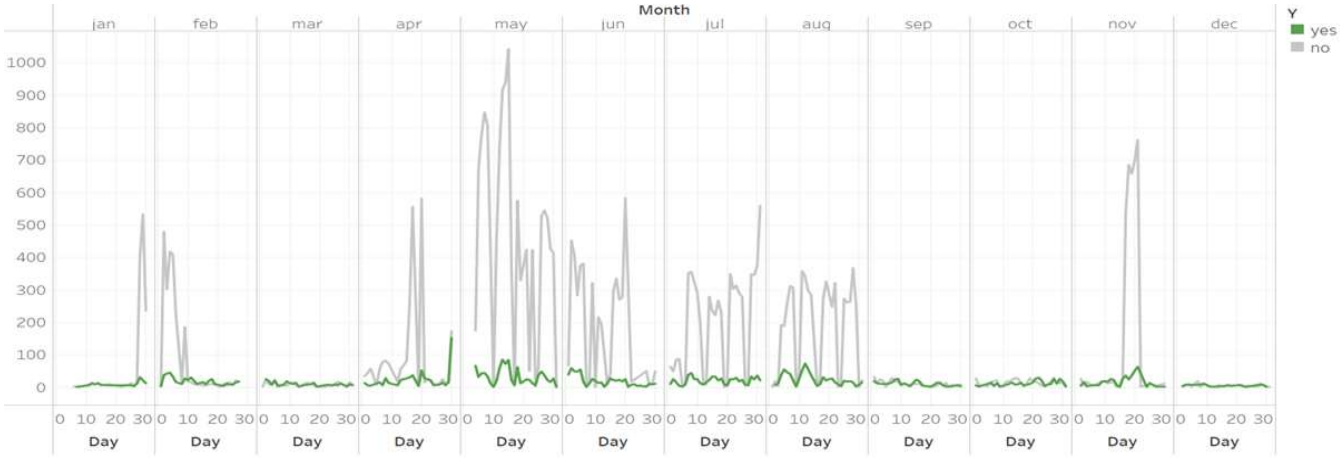
Out of extreme values, most clients are contacted within 17 times in each campaign. Overall, the number of times that old clients are contacted is less than that of new clients.

Month and Subscription Rate



After counting the proportion of the number of clients who choose to subscribe and not subscribe in different months, it can be found that the number of clients contacted in different months varies greatly. The cumulative number of clients contacted in May was the largest, but the deposit subscription rate of clients was not ideal. May to August is the peak of the cumulative number of client contacts, so we speculate that there is a seasonal factor in the frequency of contacting clients.

Fluctuation of the Cumulative Total Number of Contacts

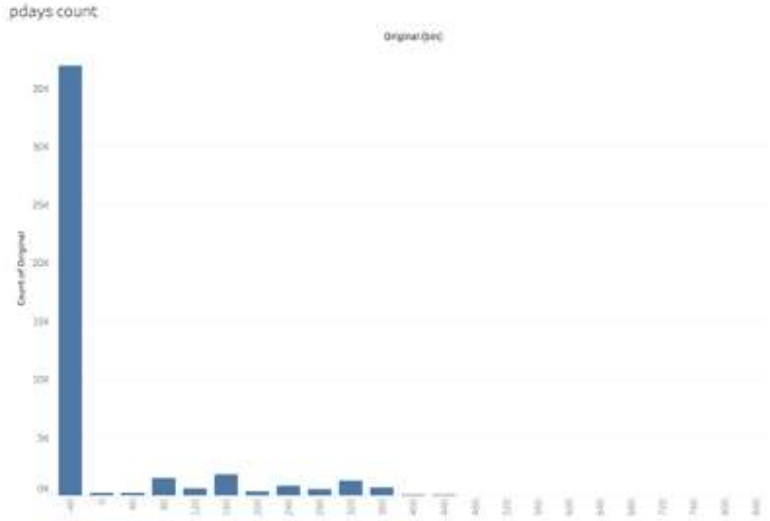


The number of contacts fluctuates cyclically within each month, and the number of peaks is about five, which is equivalent to the number of weeks in each month. In the months when the number of contacts increased intensively, the number of clients who chose to subscribe to the deposit campaign did not increase significantly, which was basically the same as other months.

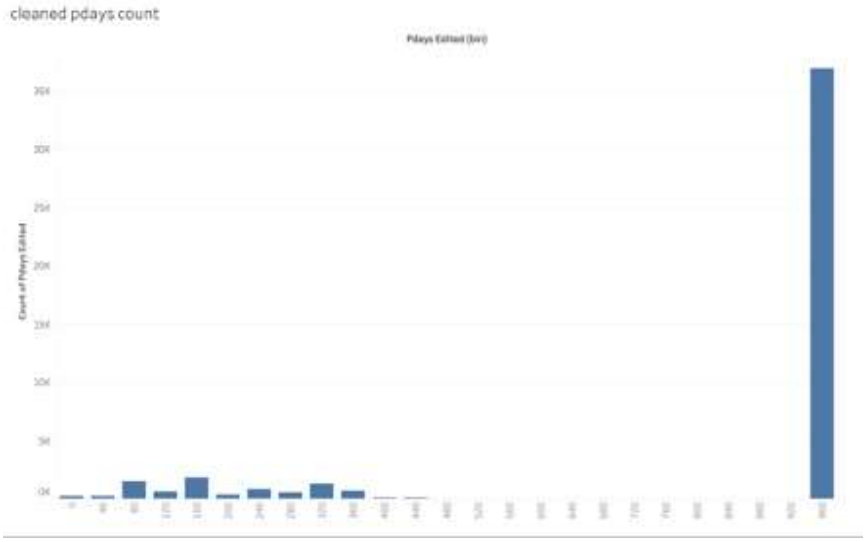
Feature Engineering

Basic Processing

For the numerical feature *pdays*, -1 is used for those clients who were never contacted before. We decided to change it to 999, to restore the order (<) in the feature. The reason for this transformation is that a client who is never contacted before (pdays = -1) is closer to one who was contacted 1000 days ago (pdays = 1000) than one who was just contacted (pday = 0) .



before and after the pdays transformation



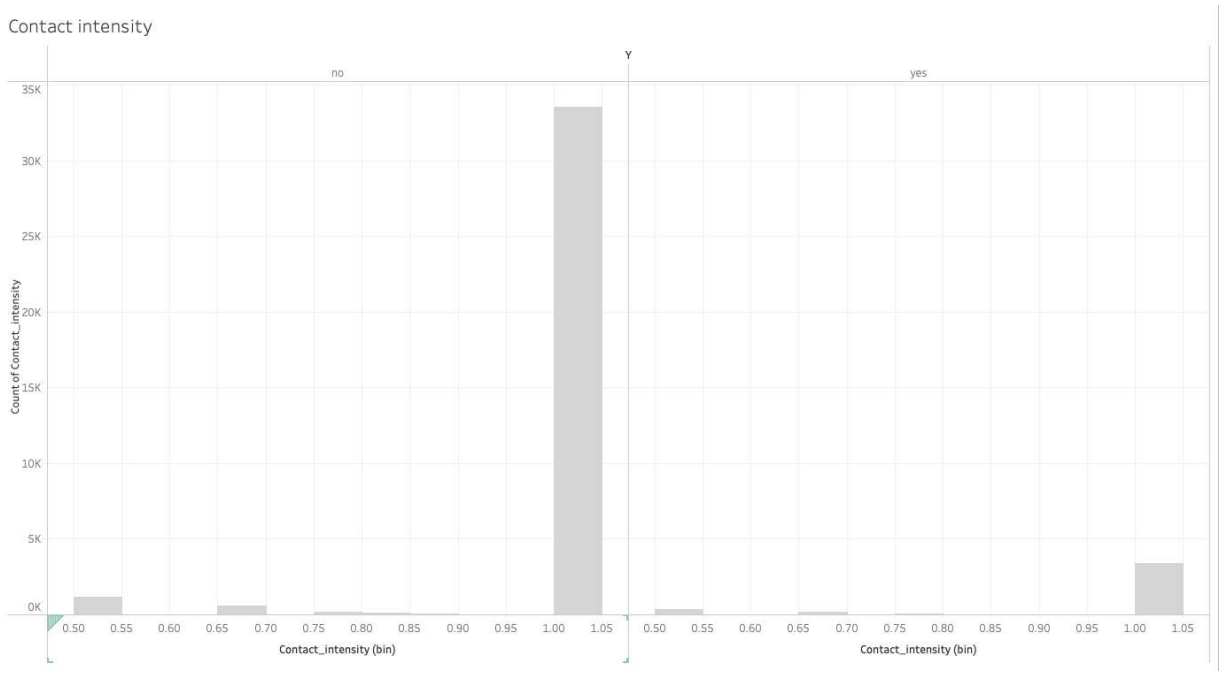
Additional features

Contact_intensity

Contact_intensity is defined as

$$\frac{\text{campaign}}{\text{campaign} + \text{previous}}$$

It is the proportion of the number of the calls received by clients in this campaign to all the cumulative number of calls in history. In other words, for an individual, this feature captures how intense this current campaign is relative to all previous campaigns. It can also tell if a client is new, if the intensity is 1.



Balance_education_index (BEI)

Education is originally a categorical variable, with values of primary, secondary, tertiary and missing values. In a way, it does have an ordered structure. Hence it would make sense to convert it as an ordered feature in certain settings.

It’s possible that education and balance together has a compounding effect, which may or may not be modeled by just a linear combination. In order to model some of the nonlinear effects, BEI is introduced. In our definition. BEI is defined as

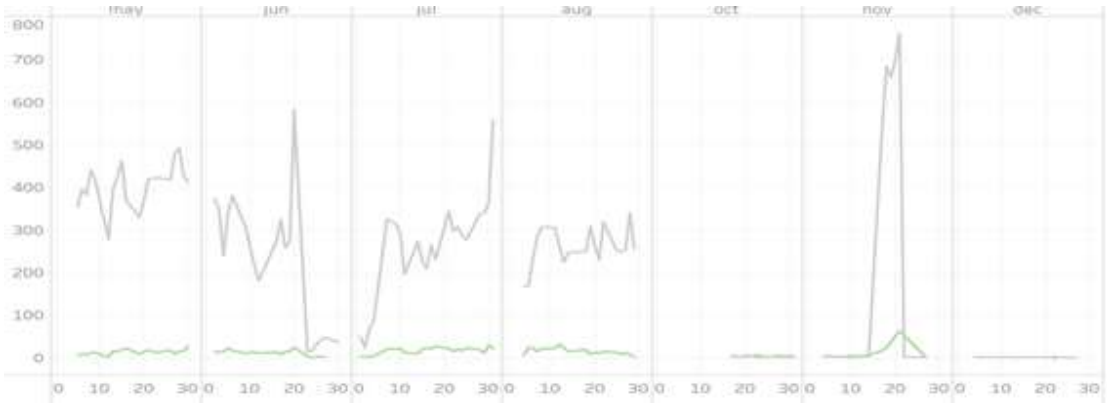
$$BEI = education * balance$$

where balance is 1,2,3,2 for primary, secondary, tertiary and missing values.

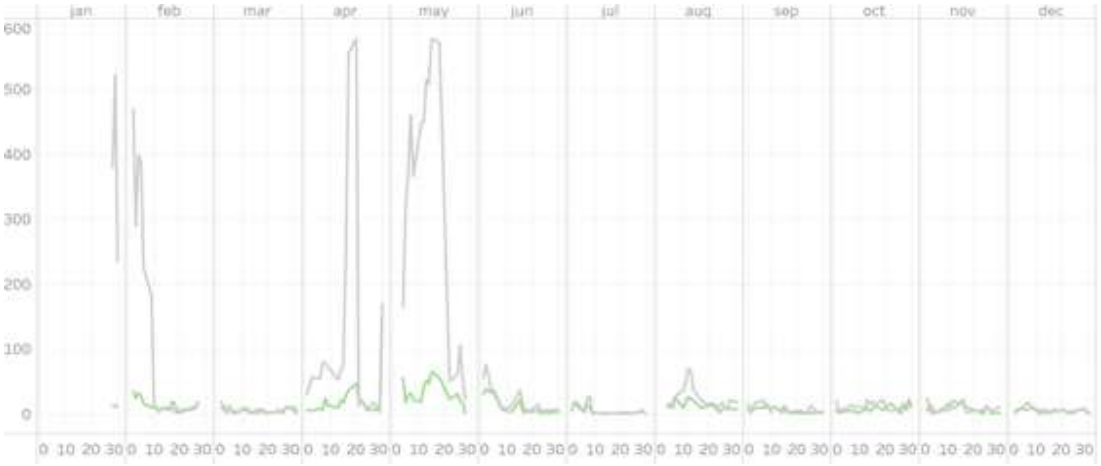
year

According to the time provided in the background information of the data set (from May 2008 to November 2010), we match the corresponding date for each contact, then mark each contact with the year 2008, 2009 and 2010. This variable is used to distinguish the contact status and final subscription results in different years.

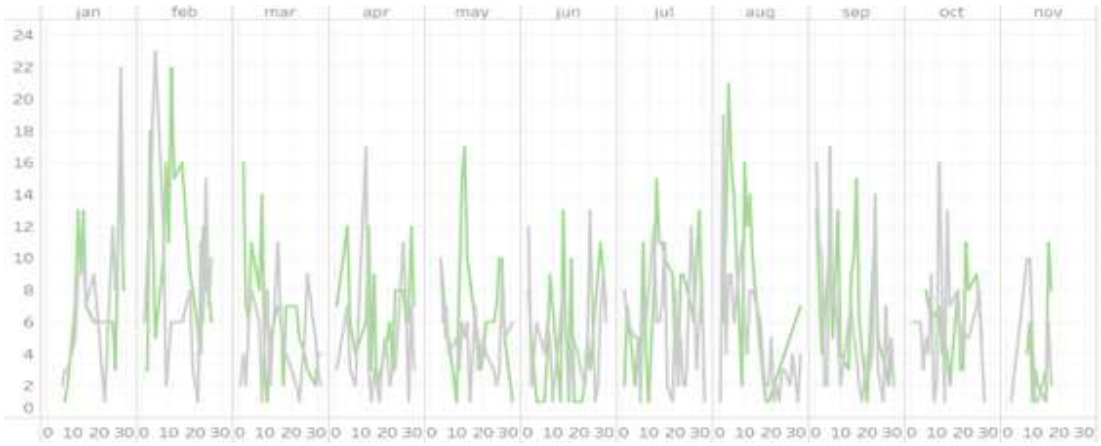
2008



2009



2010



It can be seen that the situation in 2010 is quite different from the other two years. As the total number of contact clients throughout this year was small, there is no specific pattern in the reflected clients subscription status. Based on historical facts, it is speculated that the financial crisis that year had a huge impact on the bank's business.

weekday and weeknum

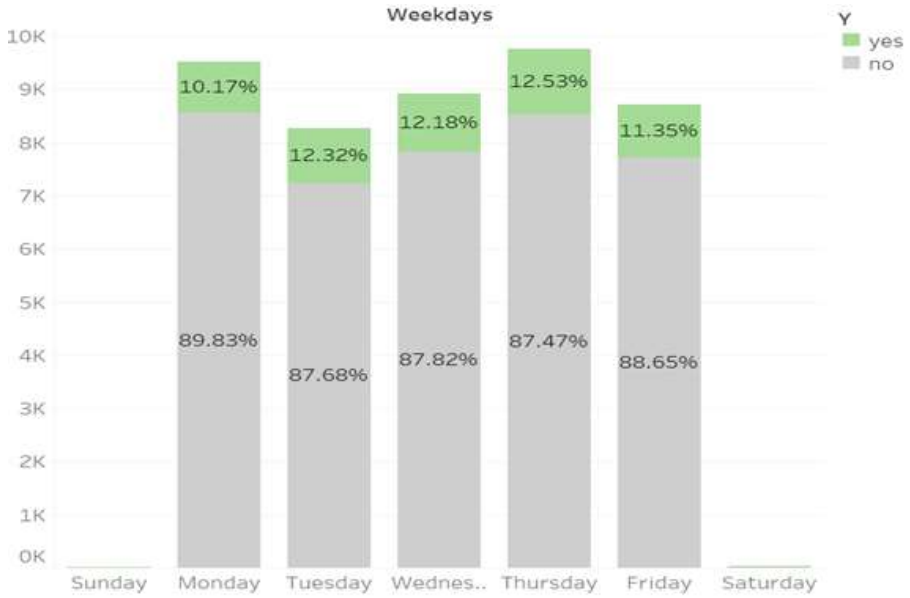
According to the time provided in the background information of the data set (from May 2008 to November 2010), we match the time for each record of contacting , and extract the week as a new variable that matches the corresponding date for each contact.

weekday: Calculate the corresponding weekdays, assigning values of 1 to 7 from Monday to Sunday to indicate.

weeknum: Count the number of weeks in which the contact occurred in the entire time period according to the date.

These variables are used to distinguish between weekdays and weekends and sort the week number.

weekday	Mon.	Tue.	Wed.	Thu.	Fri.	Sat.	Sun.
value	1	2	3	4	5	6	7



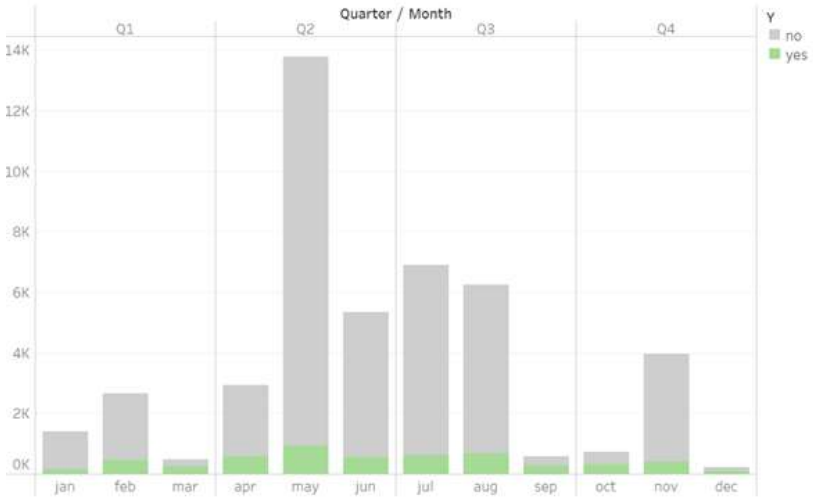
It can be found that employees of the bank never contact clients on weekends, and the contact results on each working day are basically similar. The above facts also explain why the number of contacts in each month fluctuates periodically.

quarter and rankinquarter

According to the time provided in the background information of the data set (from May 2008 to November 2010), we match the corresponding date for each contact, then mark each contact

with the year 2008, 2009 and 2010. And we divide the months according to quarters, and assign values to each month within each quarter to indicate the order.
This variable is used to distinguish the contact status and final subscription results in different quarters.

Q1	Q2	Q3	Q4	rankinquarter
Jan.	Apr.	Jul.	Oct.	1
Feb.	May.	Aug.	Nov.	2
Mar	Jun.	Sep.	Dec.	3



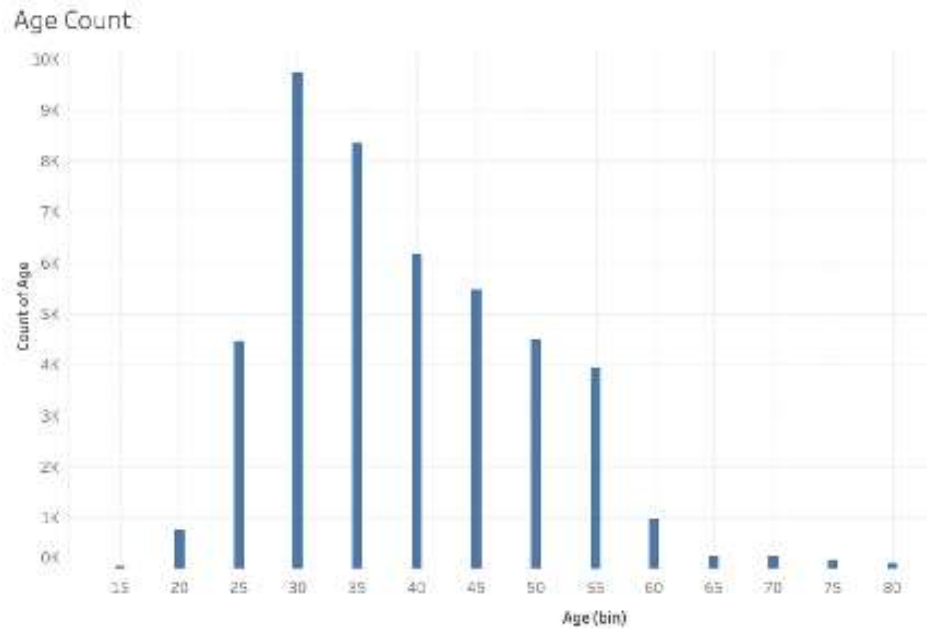
From the bar chart, the second month of each quarter always has the largest number of contacts. The total number of contacts in different months is different, which may be related to the distribution of holidays in the country.

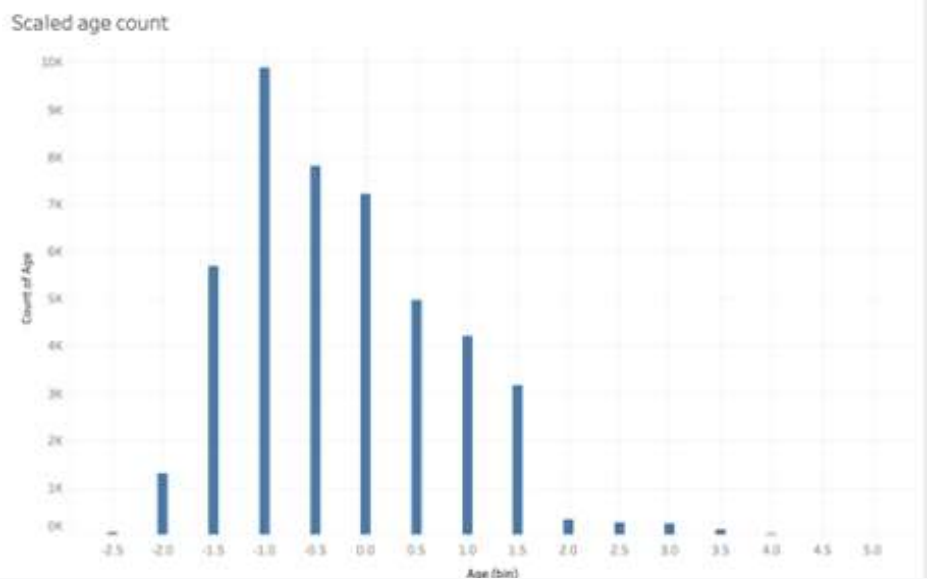
Scaling

Standardization of datasets is a common requirement for many machine learning estimators implemented in scikit-learn; they might behave badly if the individual features do not more or less look like standard normally distributed data: Gaussian with zero mean and unit variance.

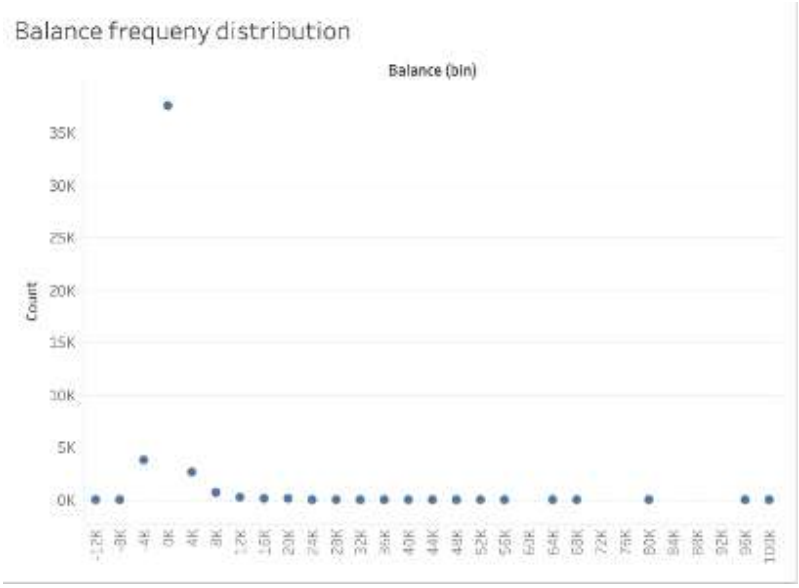
In practice we often ignore the shape of the distribution and just transform the data to center it by removing the mean value of each feature, then scale it by dividing non-constant features by their standard deviation.

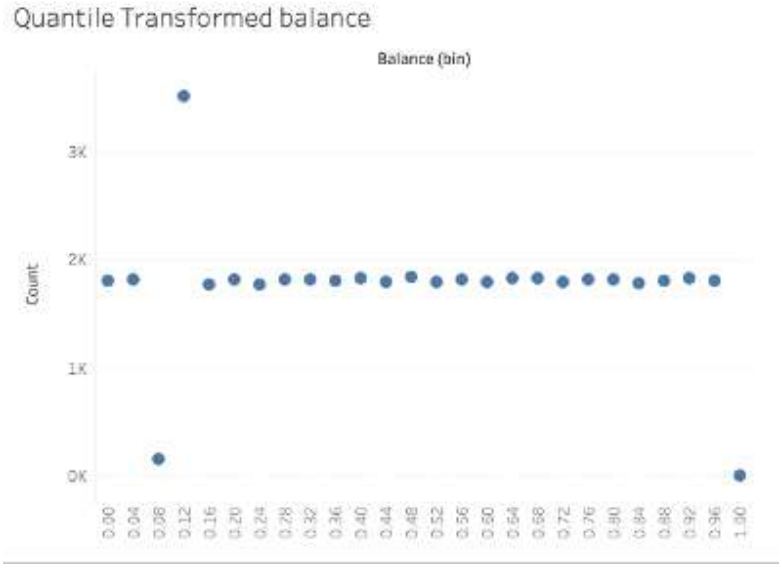
The features we standardized are age and day, since they already partially resemble a bell-shaped distribution.





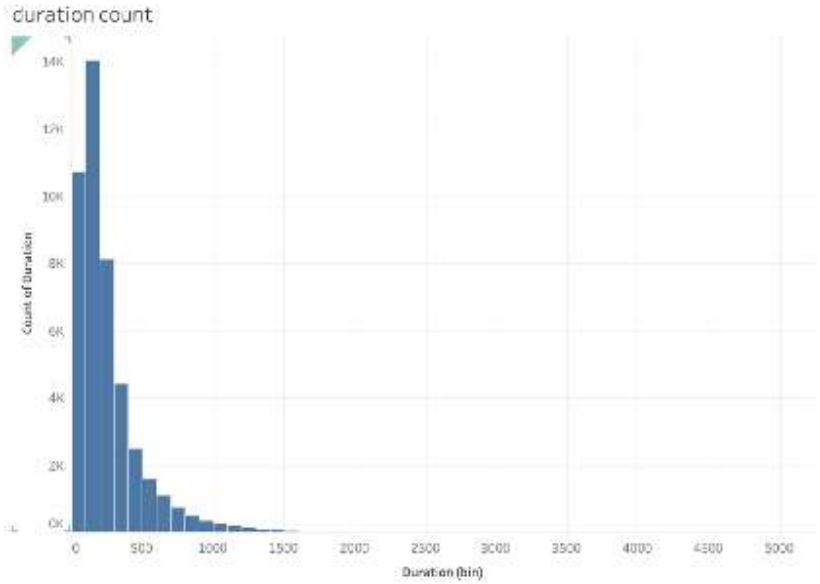
In a similar logic, QuantileTransformer provides a non-parametric transformation to map the data to a uniform distribution with values between 0 and 1:
Features we used quantile transforms include balance, pdays and contact_intensity.

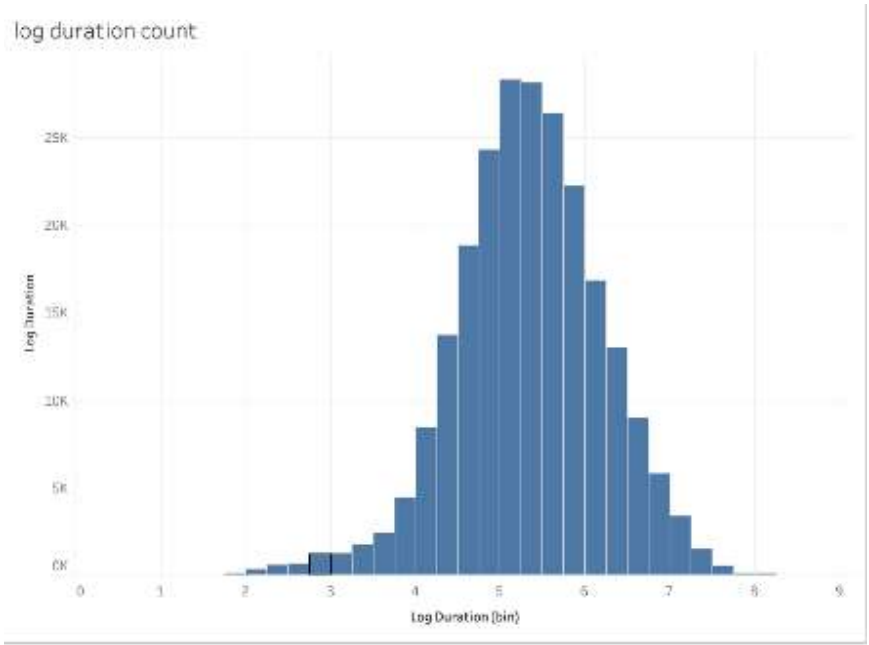




Additionally, we used log1p transform to highly skewed, nonnegative count features.

Features using log1p transformation: duration.





Encoding

After processing continuous variables, we should turn our eyes to categorical variables. Simply converting them into natural numbers would additionally give those variables orders, which is unnatural to say single < married < divorced in our case. So, we apply some encoding methods to cope with this problem. The methods we tried include one-hot encoding and WoE encoding.

marital_divorced	marital_married	marital_single
1	0	0
0	0	1
0	0	1
0	1	0
0	1	0
0	1	0

One-hot encoding

One-hot encoding: One often-used method is to use one-hot encoding, which separates every possible value into one new variable and uses 0-1 to represent whether the original value is the same as this new variable. But it may introduce a great number of new variables thus may cause the curse of dimensionality and sparse features. Fortunately, it won't in this case.

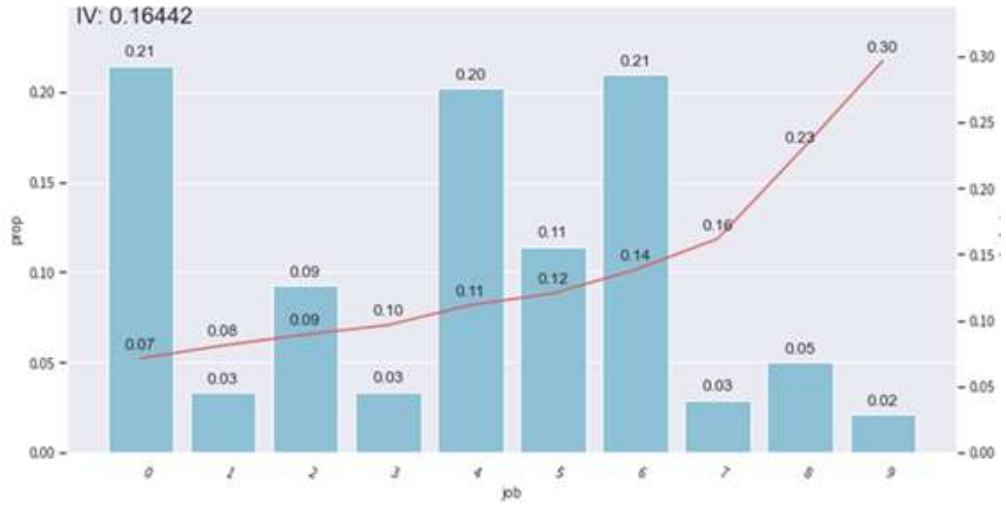
$$WoE = \left[ln\left(\frac{Distr\ Goods}{Distr\ Bads} \right) \right] * 100$$

WoE encoding

WoE encoding: Another method is to consider the distribution of 1 and 0 in each category and then merge some if needed and rearrange them by the distribution. So, we think about adopting the WoE (Weight of Evidence) encoding also. WoE can also fulfill data masking to some extent, which we will discuss more in the future improvement part.

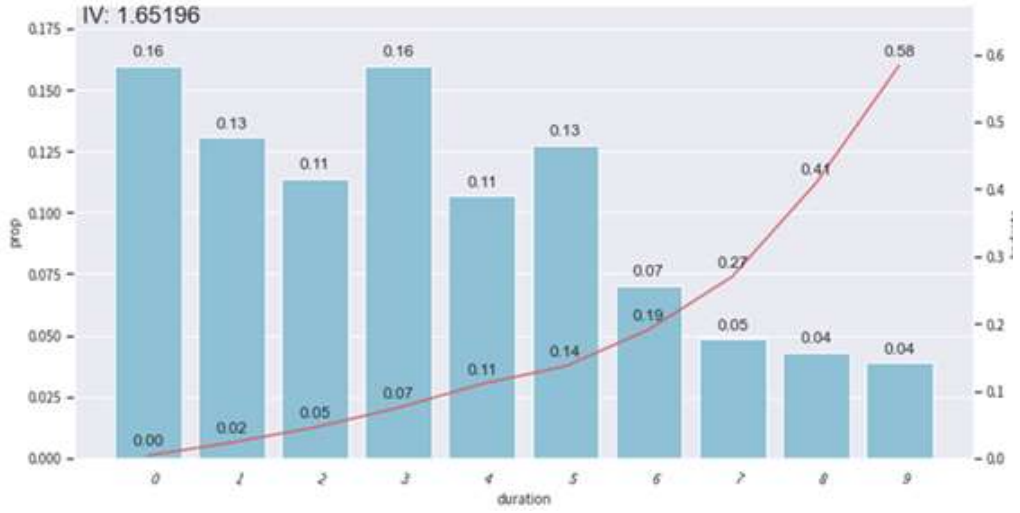
Binning

Binning is a necessary process in WoE encoding. But it is also a way of introducing orders to categorical variables. For example, after adding binning to variable *job*, we could assign labels from 0 to 9 to the bins and the yes rate in each bin would increase by the same order of labels.

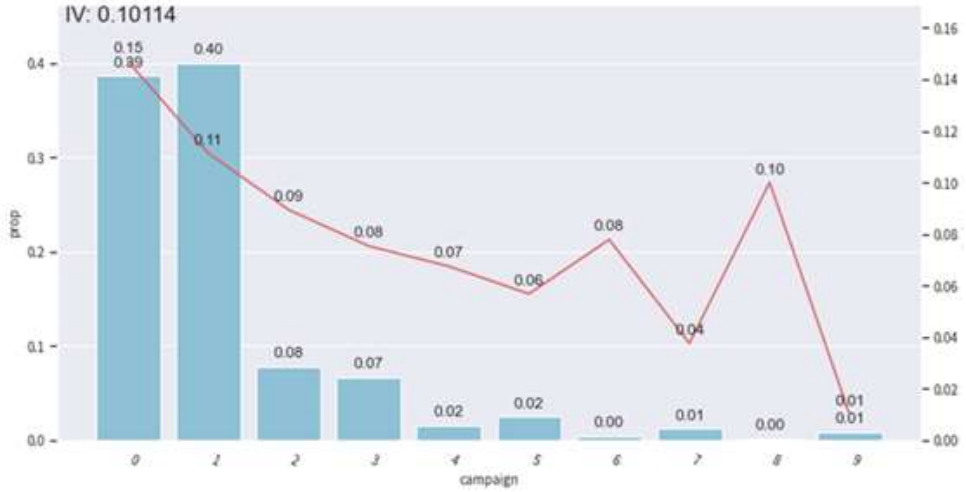


(Red line implies the yes rate and the blue bars imply the samples in this bin account for how much of all samples.)

Binning can also be applied to continuous variables to secure data masking or help us understand how the yes rate changes when one variable increases. In our case, we can conclude that as duration increases, the yes rate also increases.



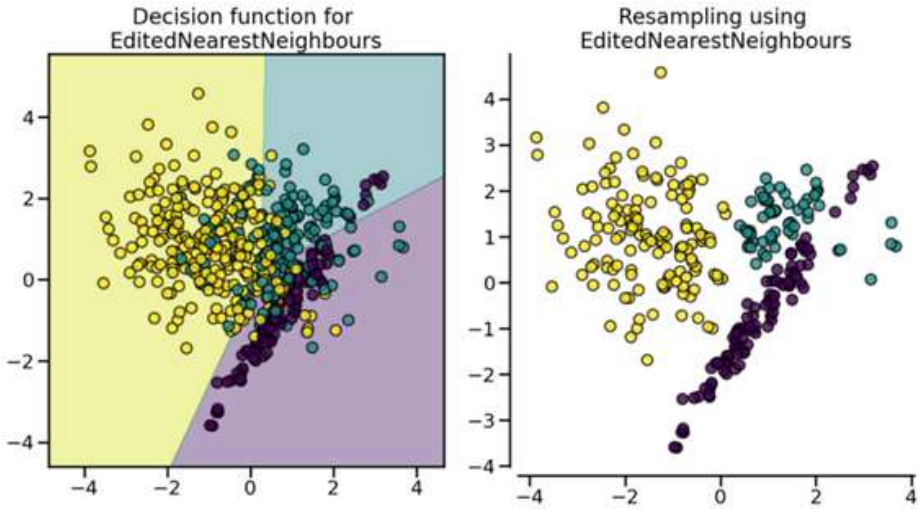
But one problem of binning for continuous variables is that the yes rate may not be monotonic according to some variables. One possible solution is to combine some bins.



Resampling

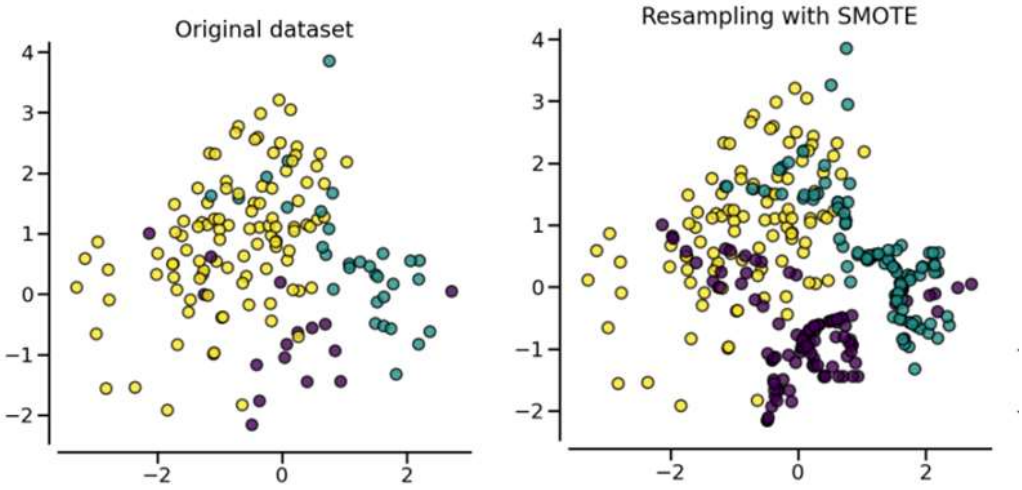
Edited dataset using nearest neighbors(ENN)

ENN applies a nearest-neighbors algorithm and “edit” the dataset by removing samples which do not agree “enough” with their neighborhood. For each sample in the class to be under-sampled, the nearest-neighbors are computed and if the selection criterion is not fulfilled, the sample is removed.



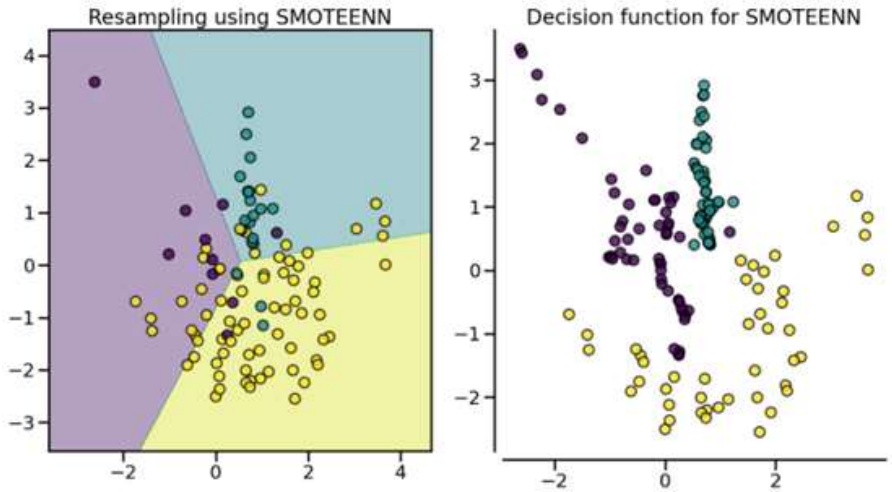
Synthetic Minority Oversampling Technique(SMOTE)

SMOTE works by selecting examples that are close in the feature space, drawing a line between the examples in the feature space and drawing a new sample at a point along that line.



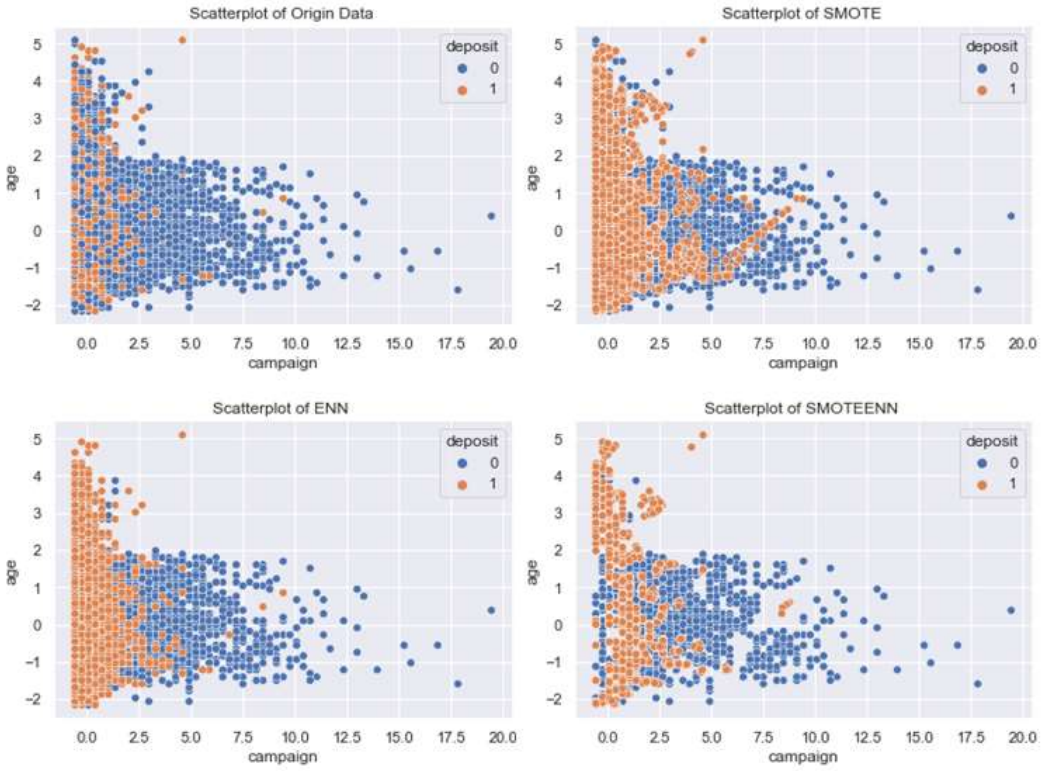
SMOTEENN

In this project, we try to use a combined algorithm to deal with the imbalanced problem. We first Over-sampling using SMOTE and then cleaned the data using ENN. We combine over- and under-sampling using SMOTE and Edited Nearest Neighbors.



The following pictures show the result of resampling. Origin data is so imbalanced that most points are bule. After oversampling we get more orange points. After undersampling, it shows

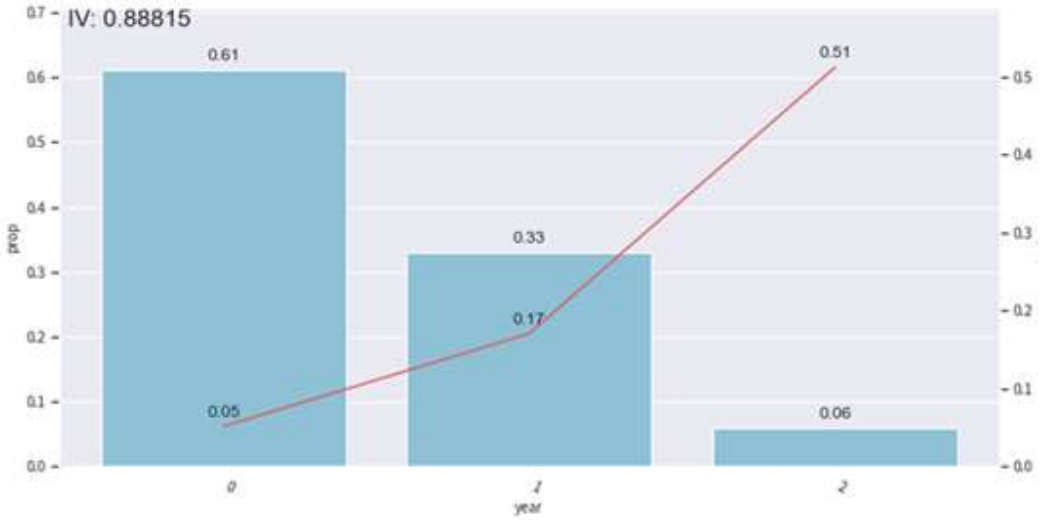
that some blue points are eliminated. Combine these two resampling, the data looks more elegant.



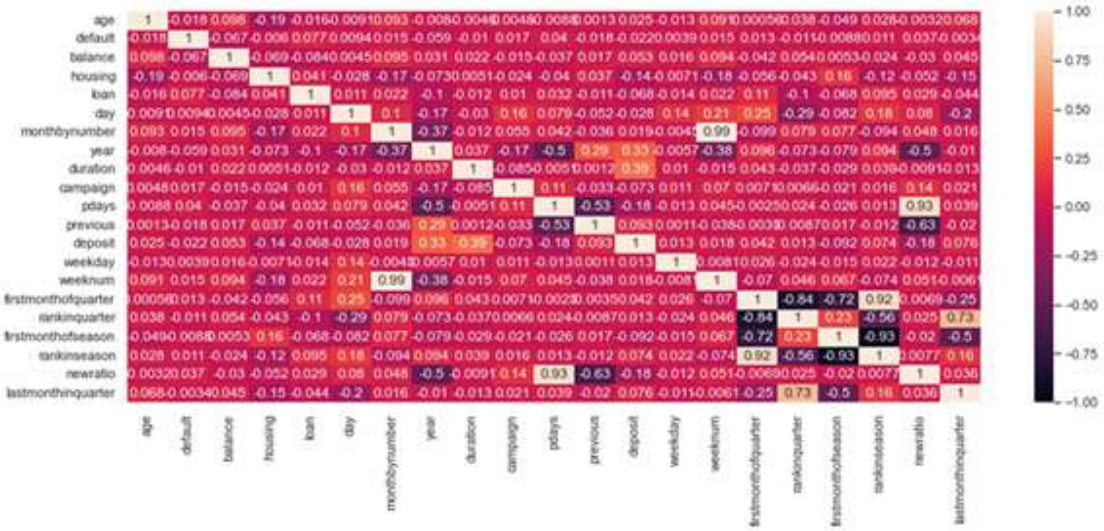
Feature Selection

A good feature selection may improve the performance of models.

The first step is to exclude variables that are not usable. In our case, variable *year* should be excluded. Though this variable may be quite different in distribution, it is not usable since we cannot recommend agents to make the phone call in 2010 (the last bar with the highest yes rate) because it was a past year. We have to exclude it.

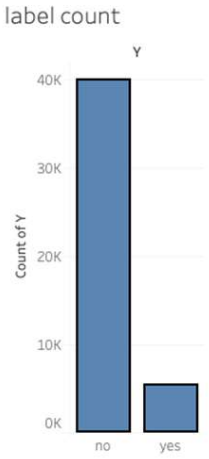


And the second step is to exclude some highly related variables. It could improve the performances of models who assume the variables should be independent of each other. In the models we tried, it should improve the logistic regression model or naïve Bayesian model. But for tree models, the help may be limited. So in tree models, we need not do the feature selection. Excluding high correlated variables means we can only leave one of them.



Spitting

Splitting is especially important in an imbalanced dataset. And the dataset is indeed very imbalanced.



The ratio of negative instances: positive instances is about 8: 1. If we randomly separate the dataset into train set and test set, there is a possibility that there may be too few positive instances in the train or test set, which may lead to worse results (Because the model only needs to predict all as negative instances if there are very few positive instances in the train set.).

So we utilize stratified splitting on response variable *deposits*. By doing this we can save the original distribution. And in the cross-validation phase, we will use stratified 5-fold cross-validation to ensure each pair of train set and test set has similar, if not the same distribution to the entire dataset.

Data modelling

Model introduction

The model based on the decision tree algorithm can solve the classification problem well,Therefore, we decided to deploy cat boost, lightGBM, and random forest models. At the same time, we also adopted some classic models such as Naive Bayes, logistic regression and neural networks.

LightGBM

The LightGBM framework supports different algorithms including GBT, GBDT, GBRT, GBM, MART and RF. LightGBM has many of XGBoost's advantages, including sparse optimization, parallel training, multiple loss functions, regularization, bagging, and early stopping. A major difference between the two lies in the construction of trees. LightGBM does not grow a tree level-wise — row by row — as most other implementations do.Instead it grows trees leaf-wise. It chooses the leaf it believes will yield the largest decrease in loss.

Besides, LightGBM does not use the widely-used sorted-based decision tree learning algorithm, which searches the best split point on sorted feature values,as XGBoost or other implementations do. Instead, LightGBM implements a highly optimized histogram-based decision tree learning algorithm, which yields great advantages on both efficiency and memory consumption. The LightGBM algorithm utilizes two novel techniques called Gradient-Based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) which allow the algorithm to run faster while maintaining a high level of accuracy

Catboost

Catboost is a GBDT framework that has fewer parameters implemented by Oblivious Trees as the base learner. The main solution is the mainly resolved pain points to handle category features efficiently. Catboost isCategorical and Boosting make up.In addition, Catboost also solves the problem of gradient bias and predict shift, thereby reducing the occurrence of prefraction, thereby increasing the accuracy and generalization of algorithms.

Random Forest

Random forests are actually a special Bagging method that will make decision trees as models in Bagging.First, use the bootstrap method to generate m training sets, and then construct a decision tree for each training set, when the node is characterized by split, it is not the maximum of the indicators (such as information gain),Instead, a part of the features are randomly extracted in the characteristics, and the optimal solution is found in the middle of the draw, applied to the node, and splits.The random forest method is because there is Bagging, that is, the idea of integration is actually sampled for samples and features (if it is a matrix as a matrix, it is like actually common, then one row and the procedure of the samples is carried out), so it is possible to avoid the overfitting problem.

Logistic Regression

In statistics, the logistic model (or logit model) is used to model the probability of a certain class or event existing such as pass/fail, win/lose, alive/dead or healthy/sick. This can be extended to model several classes of events such as determining whether an image contains a cat, dog, lion, etc. Each object being detected in the image would be assigned a probability between 0 and 1, with a sum of one.

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression). Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail which is represented by an indicator variable, where the two values are labeled "0" and "1". In the logistic model, the log-odds (the logarithm of the odds) for the value labeled "1" is a linear combination of one or more independent variables.

Naive Bayes

naive Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naïve) independence assumptions between the features (see Bayes classifier). They are among the simplest Bayesian network models, but coupled with kernel density estimation, they can achieve higher accuracy levels.

Naïve Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Maximum-likelihood training can

be done by evaluating a closed-form expression, which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers.

Neural Network

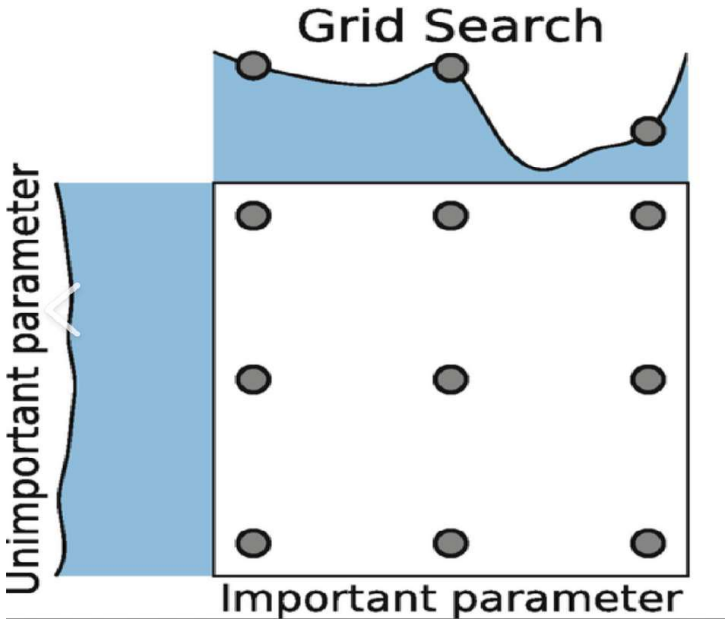
a neural network is simply a list of mathematical operations to be applied to an input. The input and output of each operation is a tensor (or more specifically a vector or matrix). Each pair of layers is connected by a list of weights. Each layer has several tensors stored in it. An individual tensor in a layer is called a node. Each node is connected to some or all of the nodes in the next layer by a weight. Each node also has a list of values called biases. The value of each layer is then the out of the activation function of the values of the current layer multiplied by the weights

Hyperparameter Tuning

GridsearchCV

Hyper-parameters are parameters that are not directly learnt within estimators. In scikit-learn they are passed as arguments to the constructor of the estimator classes.It is possible and recommended to search the hyper-parameter space for the best cross validation score.Thus ,we use GridsearchCV in sklearn to find the best matching parameters in models.

This function helps to loop through predefined hyperparameters based on a scoring metric of your choice (accuracy, f1, etc). and fit your estimator (model) on your training set. So, in the end, we can select the best parameters from the listed hyperparameters.



Tuning Example

It is often the case that we don't know how to start, or what parameter should we choose.So we will take an example of lightGBM to show how we find the best parameters.

First, we have to select the number of estimators.We set learning_rate = 0.1, in order to determine the number of estimators, the number of boosting iterations, or the number of residual trees.We can first set this parameter into a large number, then view the optimal iteration in the CV result. Secondly, set Max_Depth and Num_leaves ,which are the most important parameters that improve accuracy. What is worth mentioning is that since LightGBM uses a leaf-wise strategy, when we adjust the complexity of the tree, we should focus on num_leaves rather than max_depth. Then, min_data_in_leaf and min_sum_hessian_in_leaf are ones that decrease overfitting.The former can be set to larger number to avoid generating an excessive tree, but it is possible to lead to an underfit, while the latter means minimum sum of hessians in one leaf to allow a split. Higher values potentially decrease overfitting.The higher learning rate before is because it can make the convergence faster, but accuracy is certainly no better .Finally, we use a lower learning rate, as well as using more n_estimators to train data to see if we can further optimize the score.

Model evaluation

MCC

MCC is used in machine learning as a measure of the quality of binary classfications.it produces a more informative and truthful score in evaluating binary classifications than accuracy and F1 score, since these can dangerously show overoptimistic inflated results,especially on imbalanced datasets,while MCC takes into account the balance ratios of the four confusion matrix categories.

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

In this equation, *TP* is the number of true positives, *TN* the number of true negatives, *FP* the number of false positives and *FN* the number of false negatives. If any of the four sums in the denominator is zero, the denominator can be arbitrarily set to one.

Recall

In information retrieval, recall is the fraction of the relevant documents that are successfully retrieved.In binary classification, recall is called sensitivity. It can be viewed as the probability that a relevant document is retrieved by the query.

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{P}} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = 1 - \mathrm{FNR}$$

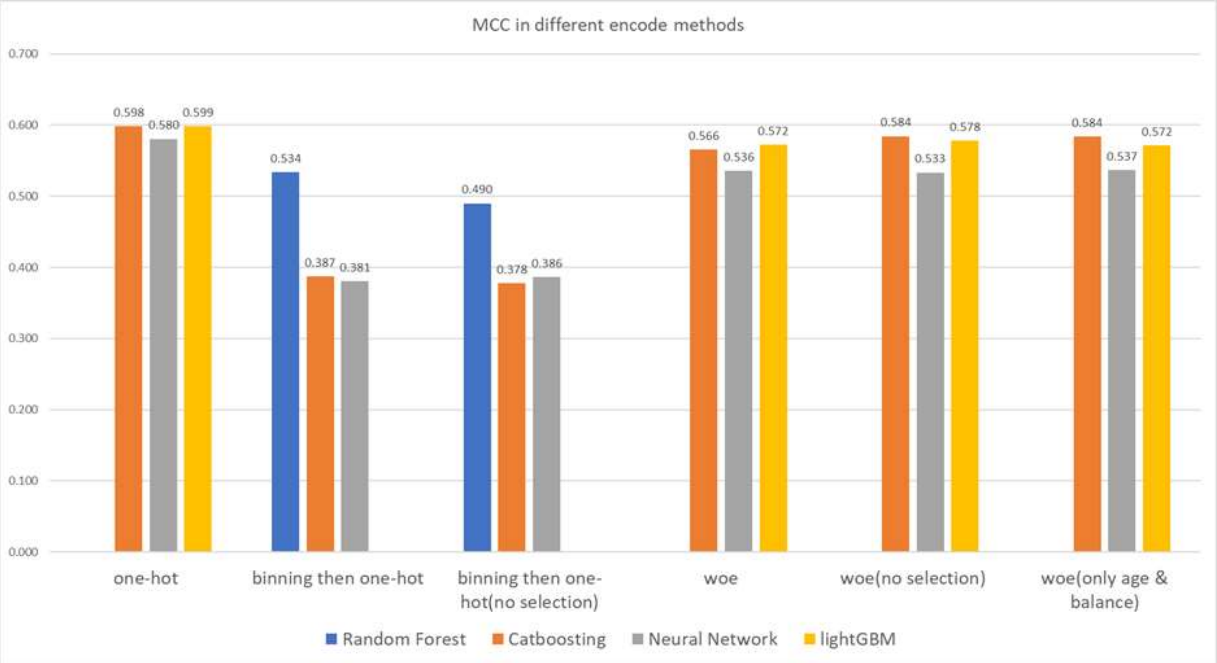
Recall, also known as True Positive Rate, indicates how many positive examples in the sample are predicted correctly.In real business scenarios, it can intuitively predict whether our customers will buy products.

We choose these two metrics as our evaluation as MCC can be taken as a technical metric and the other can be treated as a business metric ,which could be more helpful when we are exposed to real business.

Experimental Analysis

Encoding Experiment

We have experimented with 6 different encoding methods, and we can see that one-hot encoding works best, which can be seen from the picture. Although WoE will lose some information, it is not much worse than one-hot encoding, what's more WoE can prevent data leakage.



Resamplie Experiment

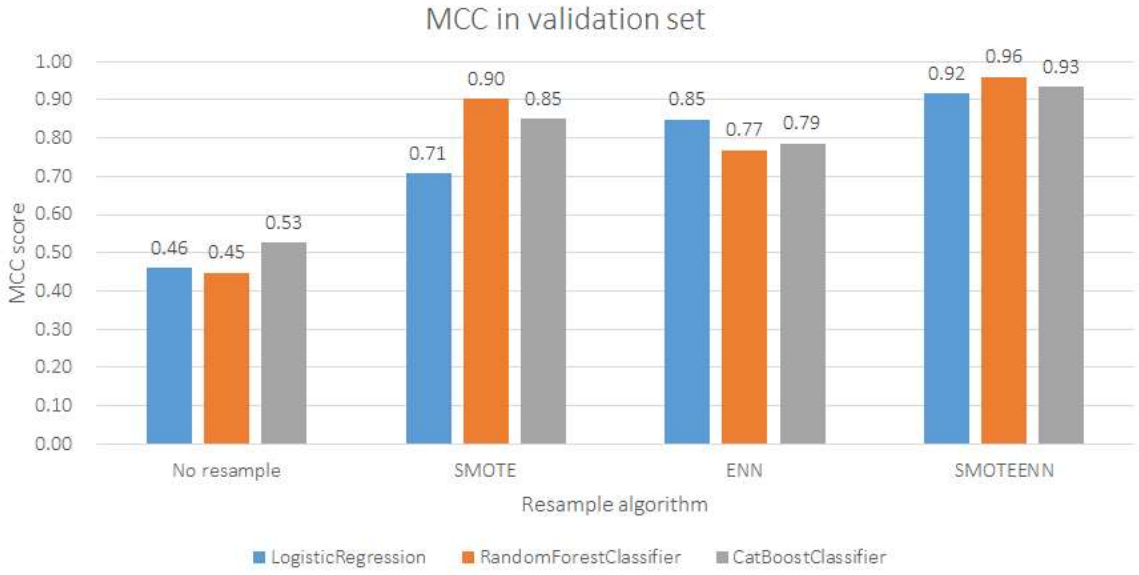
We have experimented with 3 different resampling methods. From the analysis of sampling methods, the results of downsampling or upsampling alone are not as high as the results of combining these two.

SMOTE can generate samples. However, this method of over-sampling does not have any knowledge regarding the underlying distribution. Therefore, some noisy samples can be generated. Moreover, oversampling generates some nonexistent data in real life which would lead to the overfitting problem.

ENN edits some incorrect points so that it can improve the MCC a lot compared to non-resampling dataset. However, this method doesn't change the distribution of the

original dataset much, so the result of this method is not very high. Also, there are several problems in undersampling, for example when undersampling, it might wrongly eliminate the important information which decreases the result of the machine learning model.

SMOTEENN is a method which combines both oversampling and undersampling, so it's no surprise that it got the highest score. So we used SMOTEENN as our resampling method to handle the machine learning problem.



Model Experiment

We can see Naïve Bayes is the worst model. This is because Naïve Bayes assumes that dataset attributes are independent of each other, but it is very difficult to gain data with all attributes being independent in real life data. Therefore it is hard to further improve.

Random forest and logistic classifiers are classical machine learning models, and they score well on this problem. Neural Net performs great as well, but it would take so much time to train the model when facing big data.

However, we find that gradient boosting models, such as Catboosting and LightGBM, are the models with the highest MCC score among all classifiers. This is not strange

because they are ensemble learning systems which often provide predictive accuracy that cannot be trumped.

When comparing two boosting models, with the same good MCC score but different training time, we can easily conclude that LightGBM is the best model.

Back to our problem, our goal is to repeatedly update and train our model while positioning potential customers, so as to build an automatic approval system to deal with future promotional activities. With the increase of time, the amount of data also expands, so a model with short training time and good results is well appreciated.

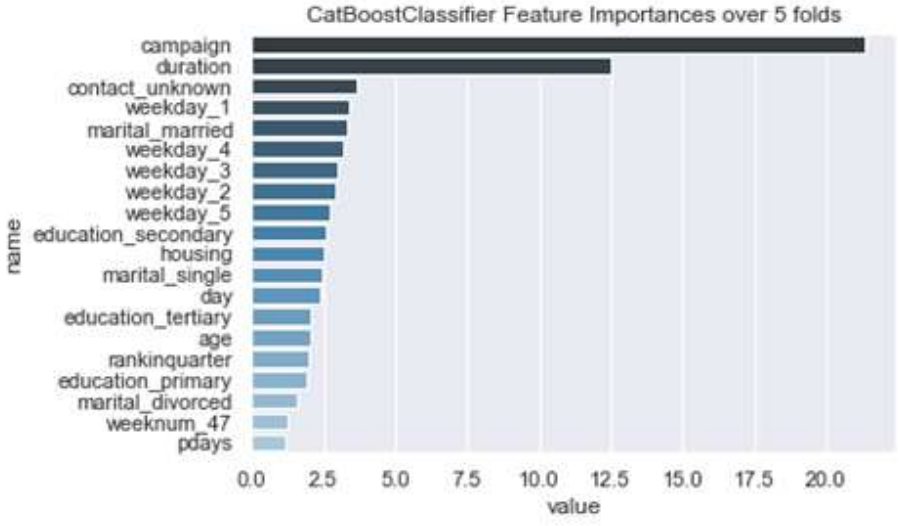
Feature importance

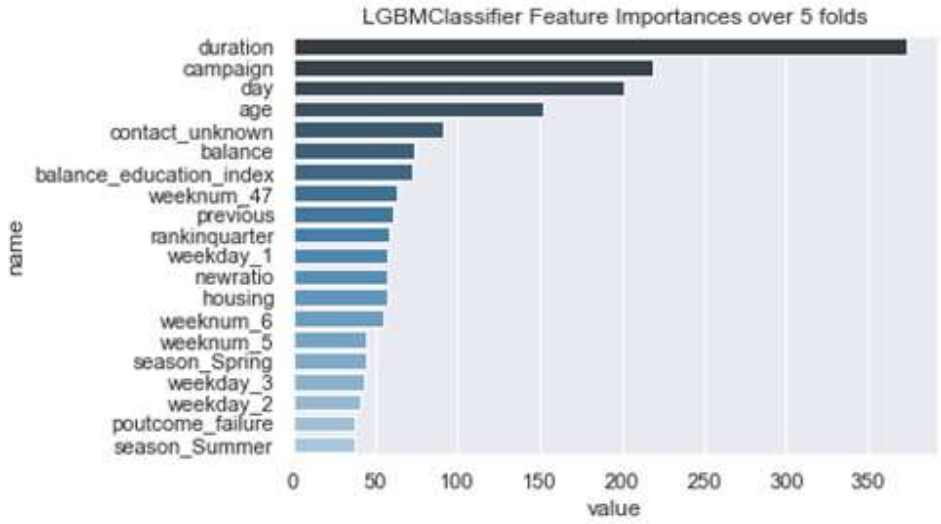
Pictures show the top 20 important features and their importance.

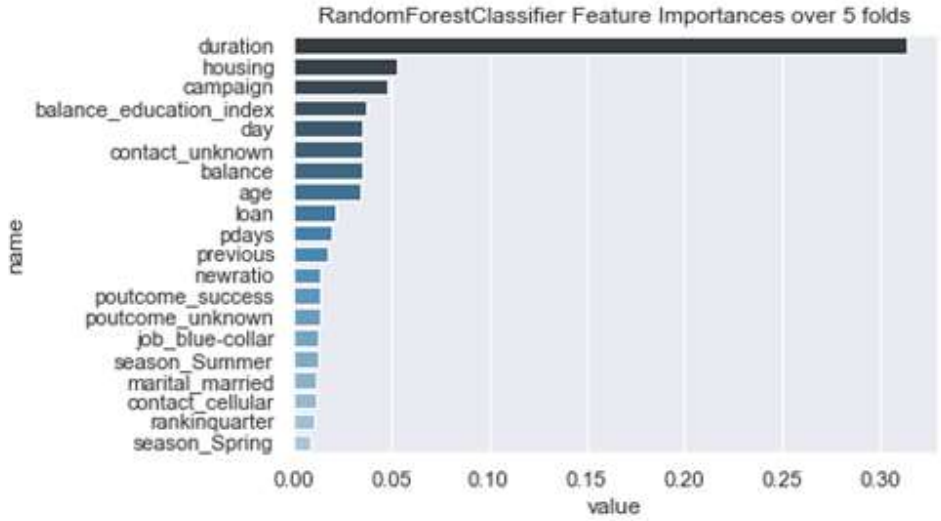
Campaign and duration are the most important features in three models.

In the random forest and LGBM model, our feature education_balance_index is important.

Date is very important to judge the potential customer.




LGBMClassifier Feature Importances over 5 folds
duration
campaign
day
age
contact_unknown
balance
balance_education_index
weeknum_47
previous
rankinquarter
weekday_1
newratio
housing
weeknum_6
weeknum_5
season_Spring
weekday_3
weekday_2
poutcome_failure
season_Summer
name
0
50
100
150
200
250
300
350
value


RandomForestClassifier Feature Importances over 5 folds
duration
housing
campaign
balance_education_index
day
contact_unknown
balance
age
loan
pdays
previous
newratio
poutcome_success
poutcome_unknown
job_blue-collar
season_Summer
marital_married
contact_cellular
rankinquarter
season_Spring
name
0.00
0.05
0.10
0.15
0.20
0.25
0.30
value

Insight

Months: May is of highest level of marketing activity(20% among a year) with lowest subscription rate(6.72%). And we should focus on March, September, October and December, which are with low level of marketing but high subscription rate(over 30%).

Season: Potential clients tend to subscribe deposits during the seasons of fall and winter(from Sep. to Dec.).

Campaign Calls: Focus more on old clients rather than new clients(subscription rate 23% rather than 9%). No more than 5 calls should be applied to the same potential client. The more we call the same potential client, the less likely he or she will subscribe a term deposit.

Age Category: The next marketing campaign of the bank should target potential clients in their 20s or younger and 60s or older.

Occupation: Corresponding to age, potential clients that are students or retired are the most likely to subscribe to a term deposit.

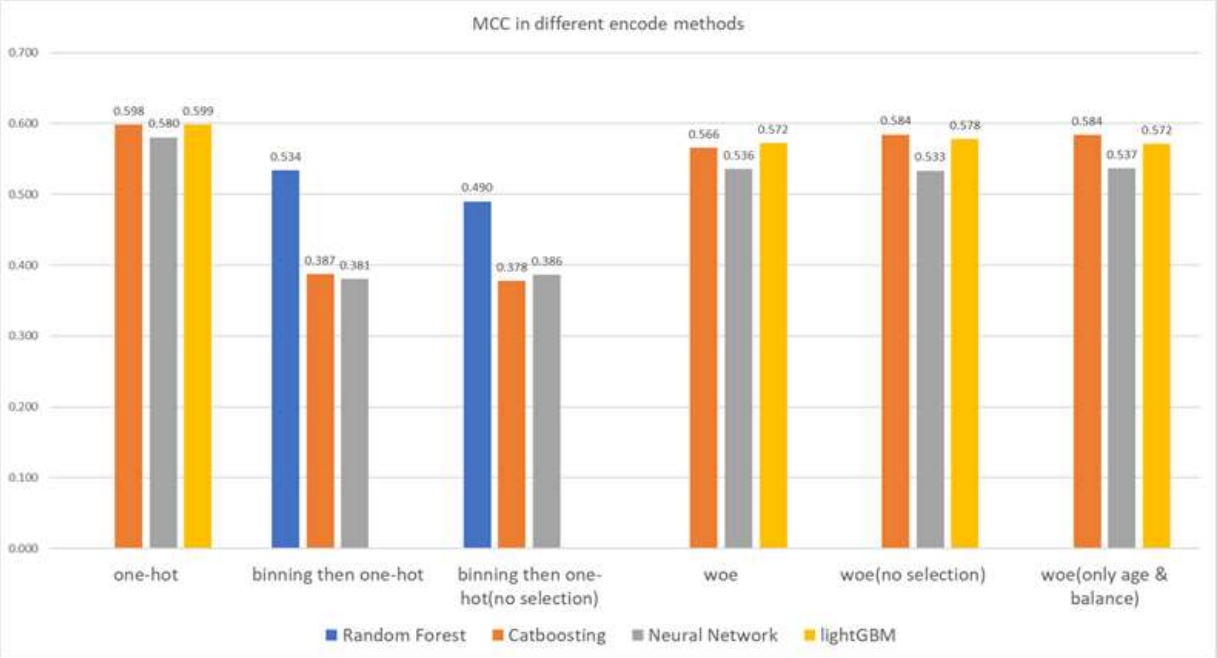
Balances and loans: the next marketing campaign should focus on individuals of high balances with low loans.

Duration: Target those who is above average in duration(above 375), there is a highly likelihood that this target group would open a term deposit account

Future

Data masking

Since the regulations are more and more strict in many countries, we would expect there will be more and more difficult for some fintech companies to get the exact data from financial institutes if fintech companies are to help use the data to build models to do a favor to businesses. So just like what we said, we stimulated that situation by applying WoE encoding which only provides ranges or labels that will not leak some key information of customers. And we are so excited to see that the best result among models whose datasets are WoE-processed can still be higher than 0.58. That result made us believe that it is possible to protect data privacy to meet regulations and fit good models simultaneously.



Further Improvement

1. We can include variables of the macroeconomic situation of this year, such as the financial crisis, national debt bankruptcy, and other information. Because it is reasonable to believe that they are also customers' concerns when they decide whether to deposit.

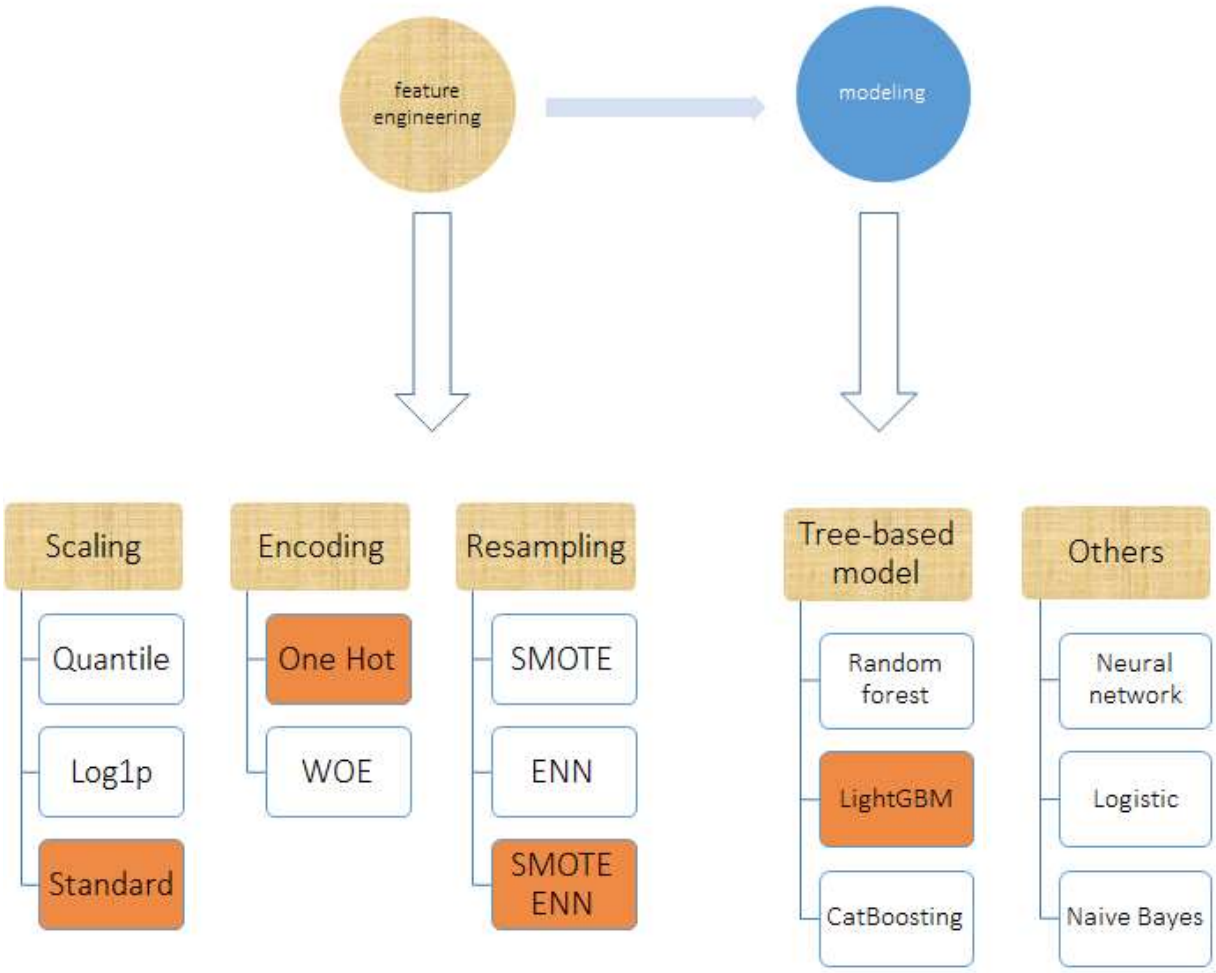
2. We can add the employee ID for each call to contact the customer. Different employees can have different yes rates since they vary in talking skills, patience, and other aspects that can impact customers.

3. Record the content of the call and perform semantic analysis on the recording to determine whether different conversations have an impact on the success rate of the promotion.

4. Add information about holidays and special dates, such as National Day or other promotional days of competing companies.

5. Use semi-supervised learning or unsupervised learning to continuously update the automatic approval system iteratively.

Summary



Based on our experimental analysis, we help banks solve the problem of targeting potential clients so that banks can maximize their subscription rates with limited resources. What's more, we point out some guides to future campaigns.


Thank You