

# MTECH PROJECT REPORT

## INTELLIGENT MARKETING SYSTEM



## REASONING SYSTEMS

### TEAM MEMBERS

Song Enyu , A0261704L  
Zhang Junmengyang , A0261791Y  
Tian Qingyun , A0261777R  
Yang Tiancheng, A0261626E

# **Abstract**

This report starts with the fact that in today's artificial intelligence era, business intelligence and smart marketing offer businesses or organizations the opportunity to improve the efficiency and effectiveness of their marketing. As one of the main marketing methods of banks, direct telemarketing is one of the main ways banks invest a lot of human cost and resources to market their regular products every year.

SMARTMARKETING helps companies using this product to achieve cost savings by using knowledge models to predict whether a specific user will accept the marketing or not.

Chapter 1 introduces the value and significance of the project in today's era, i.e. "to help companies reduce costs and improve operational efficiency", and researches the current digital marketing market to determine the need for it.

Chapter 2 introduces the implementation part of the system, including the key technologies and algorithm models used in the system. This part explains the technical principles used by the system.

Chapter 3 is the user guidance section, which helps new users to quickly familiarize themselves with the way the system is used by showing the interface.

Chapter 4 is the summary of the report, including a summary of the organizational process assets of this project and an outlook on the future of the project.

Chapter 1: Introduction .....	4
1.1 Project Background .....	4
1.1.1 Smart Marketing Market Size .....	4
1.1.2 Smart Marketing Market Size .....	4
1.1.3 Project Necessity Study .....	5
1.2 project Objectives & Scope .....	6
1.2.1 Project Objectives .....	6
1.2.2 Project objectives .....	7
Chapter 2: System Implementation .....	8
2.1 System Key Technology .....	8
2.1.1 Streamlit .....	8
2.1.2 Sklearn .....	8
2.2 System Knowledge Model .....	8
2.2.1 Algorithm model overview .....	8
2.2.2 XGBClassifier .....	9
2.2.3 SVC .....	10
2.2.4 RandomForestClassifier .....	11
2.2.5 GradientBoostingClassifier .....	13
2.3 Training Dataset .....	14
2.3.1 Dataset Source .....	14
2.3.2 Dataset Variables .....	15
2.3.3 Data Pre-processing .....	16
Chapter 3: Installation & User Guide .....	18
3.1 System page introduction .....	18
3.1.1 Page Structure .....	18
3.1.2 Home Page .....	18
3.1.3 Introduction Page .....	19
3.1.4 Start To Evaluate Page .....	19
3.2 System Usage Instructions .....	19
3.2.1 Installation .....	19
3.2.2 Details of Use .....	20
Chapter 4: Conclusion .....	21
4.1 Future Work .....	21
4.1.1 Algorithm Model Accuracy Optimization .....	21
4.1.2 User Experience Optimization .....	21
4.2 Individual Report .....	21
4.2.1 Yang Tiancheng .....	21
4.2.2 Zhang Junmengyang .....	21
4.2.3 Tian Qingyun .....	23
4.2.4 Song Enyu .....	24
Chapter 5: Reference .....	25

# Chapter 1: Introduction

## 1.1 Project Background

### 1.1.1 Smart Marketing Market Size

In 2021, due to the impact of the epidemic and the importance banks attach to digital marketing, online customer acquisition, customer conversion and other businesses, coupled with the impact of policies and other factors, many banks began to deepen their localized service business and began to pay attention to customer resource management, in this context, the intelligent marketing solutions market has also ushered in new development opportunities.

Intelligent marketing solutions provide intelligent, personalized, customized and accurate marketing processes for potential customer discovery, acquisition, customer activation and retention through the collection, collation, analysis and mining of online and offline customer data, and provide intelligent marketing services for the entire customer lifecycle and the entire chain.

As an integral part of building the digital journey of bank customers, intelligent marketing has become one of the hot spots in the banking IT solutions market segment. 2021 report shows a clear head effect in this market segment. From the vendor landscape, IDC believes that the senior IT vendors with deep accumulation in customer resource management solutions will continue to make efforts to marketing scenarios, and they will form a multi-competitive landscape with Internet majors that have a deep understanding of customer marketing, as well as innovative fintech vendors that focus on vertical categories in the coming years. 1.56 billion RMB, and its market size will reach 2.91 billion RMB by 2026.

### 1.1.2 Smart Marketing Market Size

In terms of data insights, most banks have weak data foundation capabilities, which in turn results in inaccurate customer insights or missing key dimensions of labels. Specifically, these banks Have not yet formed a systematic concept of big data collection and analysis. The storage of product data, transaction data, user data, operation data, and marketing activity data is scattered and not connected. Many valuable data are missed.

From the perspective of initiative design, marketing planners have difficulty using data to guide digital marketing actions. This is mainly due to their inability to discern the value of the acquired data for digital marketing actions. It is not clear which data are correlated with each other.

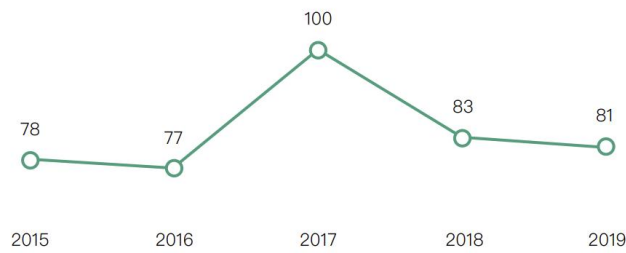


Figure 1-1 Trends in Digital Advertising Spending in China's Banking Industry

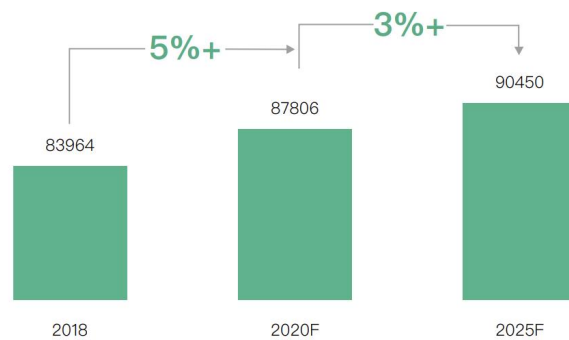


Figure 1-2 Effective audience size and forecast of digital marketing in China's banking industry

From an executive management perspective, it is difficult to optimize digital marketing actions at the executive level promptly, and it is difficult for management to make decisions. The banking industry in the digitalization process does not lack digital Theoretical knowledge of marketing, nor the ability to set indicators, but lacks efficient effect evaluation and management tools.

From the analysis of feedback, digital marketing actions ultimately can not be completed through data analysis iterative. Most banks are affected by a weak database and have to resort to experience again.

### 1.1.3 Project Necessity Study

The concept of business intelligence was first introduced in 1996 by the Gartner Group, which defines business intelligence as Business intelligence describes a set of concepts and methods that assist in business decision making through the application of fact-based support systems. Business intelligence technologies provide techniques and methods that enable organizations to rapidly analyze data, including collecting, managing, and analyzing data, transforming that data into useful information, and then distributing it throughout the enterprise.

The increasingly vast number of marketing campaigns over time has reduced its effect on the general public. Furthermore, economical pressures and competition has led marketing managers to invest in directed campaigns with a strict and rigorous selection of contacts.

Such direct campaigns can be enhanced through the use of Business Intelligence (BI) and Data Mining (DM) techniques. This paper describes an implementation of a DM project based on the CRISP-DM methodology. Real-world data were collected from a Portuguese marketing campaign related to bank deposit subscriptions. The business goal is to find a model that can explain the success of a contract, i.e. if the client subscribes to the deposit. Such a model can increase campaign efficiency by identifying the main characteristics that affect success, helping in better management of the available resources (e.g. human effort, phone calls, time), and selection of a high-quality and affordable set of potential buying customers.



Figure 1-3

In this era of information explosion, people are much less sensitive to various information pushes than before, so how to seize the attention of users and accurately understand their needs has become one of the problems faced by many enterprises.

With the advent of the intelligent era, the original marketing methods have undergone significant changes, according to user needs, personalized and accurate marketing, and gradually affect the lives of all people. Enterprises need to accurately grasp the core requirements of users, and then effective marketing so that users are not disturbed by invalid information. There is no doubt that the changes brought about by intelligent marketing are disruptive.

## 1.2 project Objectives & Scope

### 1.2.1 Project Objectives

In the traditional marketing approach, companies usually need to waste a lot of manpower and resources, and need to expand the marketing scope of the way to impress the target users, and any one of the links of error will affect the final marketing effect. Intelligent marketing can eliminate such a process, directly through the behavior of Internet user's data analysis, in hundreds of millions of Internet users to filter out the real demand for your products, and then the crowd of accurate portraits, easy to determine how the marketing approach to impress them.

For banks, these marketing campaigns are generally based on phone calls, where the bank's customer service agent contacts the customer at least once to confirm the customer's willingness to buy the bank's product (time deposit). The task is of basic type as a classification task, i.e. predicting whether the customer will buy the bank's product or not.

The product can be used to predict the user's willingness to accept the marketing using different kinds of machine learning methods, and if the user is likely to accept the marketing then it will be done manually, and if it is predicted that a certain customer will not accept the marketing then the customer will not be disturbed for some time in the future so that the marketing does not backfire and cause damage to the bank's brand image.

To be able to achieve the above, the system needs to be able to predict any personalized customer information provided by the user. The system includes modules for displaying the effects of different models, displaying predictions on data sets, and manual predictions.

### 1.2.2 Project objectives

The final scale of the project is to help the user of the system to predict whether the customer information entered by him will be accepted for marketing or not.



Figure 1-4

This project uses a predictive lifecycle for system development. The predictive lifecycle is expected to benefit from high certainty of well-defined requirements, stable teams, and low risk. As a result, project activities are typically executed in a sequential manner. To achieve this approach, the team needs a detailed plan of what to deliver and how to deliver it. The team leader's goal is to minimize changes in a predictive project. As the team creates detailed requirements and plans at the beginning of the project, they can articulate constraints. The team can then use these constraints to manage risk and cost. In turn, as the team implements the detailed plan, they monitor and control changes that may affect the scope, schedule, or budget.

# Chapter 2: System Implementation

## 2.1 System Key Technology

### 2.1.1 Streamlit

streamlit is an open-source Python library designed to make it simple and fast to create and share beautiful, customizable web applications for the machine learning and data science fields. Using Streamlit, users can create and deploy powerful data applications in just minutes.



Figure 2-1

### 2.1.2 Sklearn

Scikit-learn is a free software machine learning library for the Python programming language . It features a variety of classification, regression, and clustering algorithms, including support vector machines, random forests, gradient boosting, k-means, and DBSCAN, and is intended for use in conjunction with the Python numerical science libraries NumPy and SciPy.



Figure 2-2

## 2.2 System Knowledge Model

### 2.2.1 Algorithm model overview

In this project, we use four different models to solve the binary classification problem. They are XGBClassifier, SVC, RandomForestClassifier and GradientBoostingClassifier. The accuracy of each classifier is about 90% and the accuracy and the F1-score is shown in Table 1-1.



Table 2-1 Accuracy and F1-score of each model

model	accuracy	F1-score
XGBClassifier	0.903	0.589
SVC	0.884	0.300
RandomForestClassifier	0.897	0.565
GradientBoostingClassifier	0.899	0.489

The F1-Score is an indicator used in statistics to measure the accuracy of a binary classification model. It takes into account both the precision and recall of the classification model. The F1-score can be regarded as a harmonic average of the model's precision and recall, with a maximum value of 1 and a minimum value of 0. There are four scenarios when we do predicting and we give the definition of four case in Table 1-2.

Table 2-2 Four Definitions

Case	Definition
True Positive(TP)	Prediction: True, Fact: True
False Positive(FP)	Prediction: True, Fact: False
False Negative(FN)	Prediction: False, Fact: True
True Negative(TN)	Prediction: False, Fact: False

When we know the four definitions, we will give the formula of F1-score.

$$P(\text{Precision}) = \frac{TP}{TP+FP} \quad (1)$$

$$R(\text{Recall}) = \frac{TP}{TP+FN} \quad (2)$$

$$F1\text{-score} = \frac{2PR}{(P+R)} \quad (3)$$

The higher the F1-score, the more robust the classification model is.

Next we will focus on the four different models.

### 2.2.2 XGBClassifier

The XGBClassifier method is an implementation of the scikit-learn API for XGBoost classification. The full name of XGBoost is eXtreme Gradient Boosting. The basic idea of XGBoost is to gradually add tree by tree to the model, and each time you add a CRAT decision tree, the overall effect (objective function decreases) is improved. Multiple decision trees (multiple single weak classifiers) are used to form a combined classifier and each leaf node is assigned a certain weight.

As mentioned before, the cart algorithm was first proposed by Breiman and et al. It includes two types: classification trees and regression trees; The difference between the two is that the sample output (i.e., the response value) of the classification tree is in the form of a class; The sample output of the regression tree is in the form of numerical values, at this time it is impossible to use information gain, information gain rate, Gini coefficient to determine that

the node of the tree is split, we will use a new way: prediction error, commonly used are mean square error, logarithmic error, etc. Nodes are sometimes determined using the mean of samples within the node, and sometimes optimally, such as Xgboost.

Next we will give the model structure. The value K means the number of the trees and F means all the possible Cart Trees and f represent one tree from F.

$$\hat{y} = \sum_{k=1}^K f_k + (x_i), f_k \in F \quad (4)$$

If the structure of the K trees has been determined, then the rest of the entire model is the value of the leaf nodes of all K trees, and the regularization term of the model can also be set to the sum of squares of the values of each leaf node. At this point, the entire objective function is actually a function of the values of all leaf nodes of a K tree, and we can use gradient descent or stochastic gradient descent to optimize the objective function.

The method in Sklearn is shown below and we list some key features.

```
xgboost.XGBClassifier(max_depth=3, learning_rate=0.1, n_estimators=100,
verbosity=1, objective='binary:logistic', booster='gbtree',
tree_method='auto', n_jobs=1, gpu_id=-1, gamma=0, min_child_weight=1,
max_delta_step=0, subsample=1, colsample_bytree=1,
colsample_bylevel=1, colsample_bynode=1,
reg_alpha=0, reg_lambda=1, scale_pos_weight=1, base_score=0.5, random_state=0, miss
ing=None, **kwargs)
```

Table 2-3 XGBClassifier

Parameter	Meaning
max_depth (int)	The max depth of the tree
learning_rate (float)	The learning rate
n_estimators (int)	The number of the trees
booster (string)	Boosters specified: gbtree, gblinear or dart
random_state (int)	Number of random seeds
num_parallel_tree (int)	Used to enhance random forests
importance_type (string)	Attribute feature importance type: "gain", "weight", "cover", "total_gain" or "total_cover"

### 2.2.3 SVC

The SVC(support vector machine) is a classification algorithm, but it can also do regression, and different models can be made according to the input data (regression if the input label is a continuous value, and SVC is used to classify if the input label is a categorical value). By seeking to minimize the structural risk, the generalization ability of the learning machine is improved, and the empirical risk and confidence range are minimized, so as to achieve the purpose of obtaining good statistical laws even when the statistical sample size is small. In layman's terms, it is a second-class classification model, and its basic model is defined as a linear classifier with the largest interval in the feature space, that is, the learning strategy of the support vector machine is to maximize the interval, which can finally be transformed into a convex quadratic programming problem.

The method in Sk-learn is shown below and we list some key features. The implementation is based on libsvm, so there are many similarities in parameter setting.

```
sklearn.svm.SVC(C=1.0, kernel="rbf", degree=3, gamma="auto_deprecated",
coef0=0.0, shrinking=True, probability=False, tol=0.001,
cache_size=200, class_weight=None, verbose=False, max_iter = -1,
decision_function_shape="ovr", random_state=None)
```

Table 2-4 SVC

Parameter	Meaning
C(optional, the default value is 1)	Penalty parameter C for incorrect terms. The larger C, the greater it is equivalent to punishing the relaxation variable, and it is hoped that the relaxation variable is close to 0, that is, the punishment for misclassification increases, and it tends to be the case of full score pairs of the training set, so that the accuracy of the training set is very high, but the generalization ability is weak.
kernel(optional, the default value is "rbf")	'linear': linear kernel function 'poly': polynomial kernel function 'rbf': radial kernel function/Gaussian kernel 'sigmoid': sigmoid kernel function...
Probability(bool, the default value is False)	Whether to enable probability estimation, bool type, optional parameter, default to False, this must be enabled before calling fit() and will slow down the fit() method.
Decision_function_shape	Decision function type, optional parameters 'ovo' and 'ovr', default to 'ovr'. 'ovo' means one vs one, and 'ovr' means one vs rest.
max_iter	The maximum number of iterations, int type, defaults to -1, which means no limit.
class_weight	Category weight, dict type or str type, optional parameter, default to None. Set a different penalty parameter C for each category, and if it is not given, all categories will be given C=1, that is, the parameter C indicated in the previous parameter. If the parameter 'balance' is given, the weight inversely proportional to the class frequency in the input data is automatically adjusted using the value of y.
shrinking	Whether to use heuristic contraction mode, bool type, optional parameter, default to True.

## 2.2.4 RandomForestClassifier

The random forest classifier establishes multiple unrelated decision trees by randomly sampling samples and features, and obtains prediction results in parallel. Each decision tree can derive a prediction result from the samples and features taken, and by combining the

results of all "trees", the classification prediction result of the entire "forest" is obtained. It can be applied to binary classification or multi-classification scenarios.

As we have mentioned before, the relation between Boosting, Bagging and Random Forest is shown below.

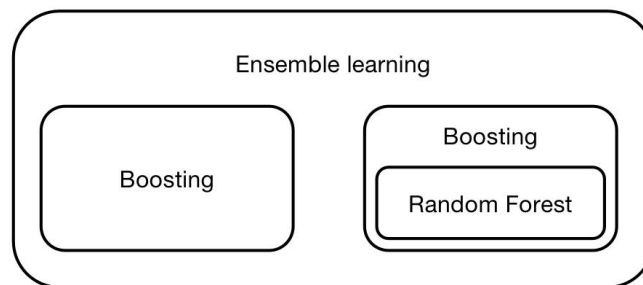


Figure 3-3

A random forest is made up of many decision trees, and there is no association between the different decision trees. When we carry out the classification task, new input samples enter, let each decision tree in the forest judge and classify separately, each decision tree will get its own classification result, which of the classification results of the decision tree is the most classified, then the random forest will take this result as the final result.

To build a Random Forest, we have four steps.

- A sample with a sample size of  $N$  is put back  $N$  times, 1 at a time, and finally  $N$  samples are formed. This selects  $N$  samples to train a decision tree as samples at the root node of the tree.
- When each sample has  $M$  attributes, when each node of the decision tree needs to be split,  $M$  attributes are randomly selected from these  $M$  attributes to meet the conditions  $m \ll M$ . Then some strategy (such as information gain) is used from these  $m$  attributes to select 1 attribute as the split attribute of the node.
- Each node in the process of decision tree formation should be split according to step 2 (it is easy to understand that if the next attribute selected by the node is the attribute that was used when its parent node split just now, the node has reached the leaf node and does not need to continue splitting). Until it can no longer split. Note that no pruning occurs throughout decision tree formation.
- Follow steps 1~3 to build a large number of decision trees, so that a random forest is formed.

The method in Sk-learn is shown below and we list some key features.

```
sklearn.ensemble.RandomForestClassifier(n_estimators=100, *, criterion='gini',
max_depth=None, min_samples_split=2, min_samples_leaf=1,
min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None,
min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None,
random_state=None, verbose=0, warm_start=False, class_weight=None,
ccp_alpha=0.0, max_samples=None)
```

Table 2-5 RandomForestClassifier

Parameter	Meaning
n_estimators	The number of trees in the forest.
criterion	The function to measure the quality of a split. Supported criteria are "gini" for the Gini impurity and "log_loss" and "entropy" both for the Shannon information gain
min_samples_split	The minimum number of samples required to split an internal node: If int, then consider min_samples_split as the minimum number. If float, then min_samples_split is a fraction and $\text{ceil}(\text{min\_samples\_split} * \text{n\_samples})$ are the minimum number of samples for each split.
min_samples_leaf	The minimum number of samples required to be at a leaf node.
max_features	The number of features to consider when looking for the best split: If int, then consider max_features features at each split. If float, then max_features is a fraction and $\text{max}(1, \text{int}(\text{max\_features} * \text{n\_features\_in\_}))$ features are considered at each split.
random_state	Controls both the randomness of the bootstrapping of the samples used when building trees (if bootstrap=True) and the sampling of the features to consider when looking for the best split at each node (if max_features < n_features).
max_samples	If bootstrap is True, the number of samples to draw from X to train each base estimator. If None (default), then draw X.shape[0] samples. If int, then draw max_samples samples. If float, then draw max_samples * X.shape[0] samples. Thus, max_samples should be in the interval (0.0, 1.0].

## 2.2.5 GradientBoostingClassifier

AdaBoost uses exponential loss, a loss function that is very sensitive to outliers and as a result, it often performs poorly on noisy datasets. Gradient Boosting improves on this area so that any loss function can be used (as long as the loss function is continuously derivable), so that some of the more robust loss functions can be applied to make the model more noise-resistant.

The method in Sk-learn is shown below and we list some key features.

```
sklearn.ensemble.GradientBoostingClassifier(*, loss='log_loss', learning_rate=0.1,
n_estimators=100, subsample=1.0, criterion='friedman_mse', min_samples_split=2,
min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_depth=3,
min_impurity_decrease=0.0, init=None, random_state=None, max_features=None,
verbose=0, max_leaf_nodes=None, warm_start=False, validation_fraction=0.1,
n_iter_no_change=None, tol=0.0001, ccp_alpha=0.0)[source]
```

Table 2-6 GradientBoostingClassifier

Parameter	Meaning
loss	The loss function to be optimized. "log_loss" refers to binomial and multinomial deviance, the same as used in logistic regression.
learning_rate	Learning rate shrinks the contribution of each tree by learning_rate. There is a trade-off between learning_rate and n_estimators. Values must be in the range (0.0, inf).
subsample	The fraction of samples to be used for fitting the individual base learners. If smaller than 1.0 this results in Stochastic Gradient Boosting.
criterion	The function to measure the quality of a split. Supported criteria are "friedman_mse" for the mean squared error with improvement score by Friedman, "squared_error" for mean squared error. The default value of "friedman_mse" is generally the best as it can provide a better approximation in some cases.
random_state	Controls the random seed given to each Tree estimator at each boosting iteration.
min_weight_fraction_leaf	The minimum weighted fraction of the sum total of weights (of all the input samples) required to be at a leaf node. Samples have equal weight when sample_weight is not provided. Values must be in the range [0.0, 0.5].
max_depth	The maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree.

## 2.3 Training Dataset

### 2.3.1 Dataset Source

This dataset is about a direct telemarketing campaign of a Portuguese banking institution from May 2008 to November 2010, aimed at promoting term deposits of existing customers.

It is publicly available in the UCI machine learning repository. This is real business data, with a total of over 40,000 records, each with 21 attributes. The main task goal is to classify and predict, whether the user will accept the marketing or not.

There are two datasets in this dataset, old and new, for this example bank-additional-full.csv is used

### 2.3.2 Dataset Variables

#### **Bank customer information:**

- age: age (number)
- job: type of work. administrator, blue-collar, entrepreneur, housemaid, manager, retired, self-employed, services student', technician', unemployed', unknown
- marital : Marital status, divorced ('divorced'), married ('married'), single ('single'), unknown ('unknown'). Note: Divorce also includes widowhood
- education: education status: basic 4 years ('basic.4y'), basic 6 years ('basic.6y'), basic 9 years ('basic.9y'), high school ('high.school'), illiterate ('illiterate'), professional course ('professional.course') university.degree', unknown ('unknown')
- default: Is there a credit default? ('no','yes','unknown')
- housing: Is there a mortgage ('no','yes','unknown')
- loan: whether you have a personal loan ( 'no','yes','unknown')

#### **Contact-related information:**

- contact: type of contact, mobile ('cellular'), phone: 'telephone'
- month: month of the last contact of the year (categorical: 'jan', 'feb', 'mar', ... , 'nov', 'dec')
- day\_of\_week: week of the last contact (categorical: 'mon', 'tue', 'wed', 'thu', 'fri')
- duration: duration of the last contact (in seconds). Important: This property highly affects the output target. However, the duration is not known until the call is executed. Moreover, y is obviously known after the call has ended. Therefore, this input should only be included in the benchmark test and should be discarded if one wants to have an actual prediction model. (The length of the call that will be made is not known at the time of prediction)

#### **Other attributes:**

- campaign: The number of contacts initiated for this campaign, for this customer. (number, including the last contact)
- pdays: how many days have elapsed since the last campaign. (number, if 999 means this customer has not been contacted)
- previous: how many times the customer was contacted before this marketing (number)
- poutcome: the result of the last marketing campaign ( 'failure', 'nonexistent', 'success')

#### **Social and economic related attributes**

- emp.var.rate: rate of change in employment - coefficient indicator (numeric)
- cons.price.idx: Consumer Price Index - monthly indicator (numeric)

- cons.conf.idx: Consumer confidence - monthly indicator (numeric)
- euribor3m: EURIBOR 3 months - Daily indicator (numeric)
- nr.employed: number of employees - quarterly indicator (numeric)

### 2.3.3 Data Pre-processing

#### Missing value handling

In data pre-processing, the process includes completing the processing of missing values in the dataset, completing the conversion of non-numeric variables in the dataset, completing the normalization of the dataset, and saving the pre-processed dataset.

The input variables of the dataset are 20 feature quantities, divided into numerical and categorical variables.

Missing values were observed using `df.isnull().any()` and no features were found to contain missing values (NaN).

However, in this dataset, the missing values are present in other forms. Most of the features in the categorical variables use unknown to represent missing values, while poutcome is represented by nonexistent; only pdays has missing values (in the form of number 999) in the numerical variables.

The percentage of missing values is as follows:

- job : 0.8%
- marital : 0.2%
- education : 4.2%
- default : 20.9%
- housing : 2.4%
- loan : 2.4%
- poutcome : 86.3%

After experiments, it was found that although some attributes were missing in a high proportion, they had an important impact on the prediction results, so they were not removed

#### One-hot coding

In the dataset, the variables job, marital, poutcome, month, and day\_of\_week can be considered as unordered categorical variables. It is important to note that although the variables month and day\_of\_week are ordered from a temporal perspective, they are unordered for the target variable. For unordered categorical variables, one-hot coding can be used.

The one-hot encoding uses N-bit status registers to encode N states, each of which has its own register bit and only one of which is valid at any given time.



**Standardization**

Not all algorithms require normalization of numerical variables. Some algorithms are more sensitive to whether the variables are normalized, such as logistic regression, support vector machines, neural networks, etc.; while random forests and decision trees do not require normalization of variables. In order to facilitate the subsequent selection of machine learning algorithms, standardization is performed uniformly here.

In this example, all numerical variables need to be normalized, and since EDUCATION is an ordered series, it also needs to be normalized.

# Chapter 3: Installation & User Guide

## 3.1 System page introduction

### 3.1.1 Page Structure

In smartmarketing, the core function is to classify the customer information entered by the user, and then give the classification result to predict whether the customer will accept the marketing or not.

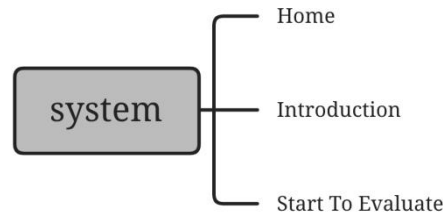


Figure 3-1

The system has three pages:

- Home page, which introduces the basic information about the project.
- Introduction page, which explains how the project uses the information in its predictions.
- Start To Evaluate page, where customer predictions can be made.

### 3.1.2 Home Page

The page is shown below:

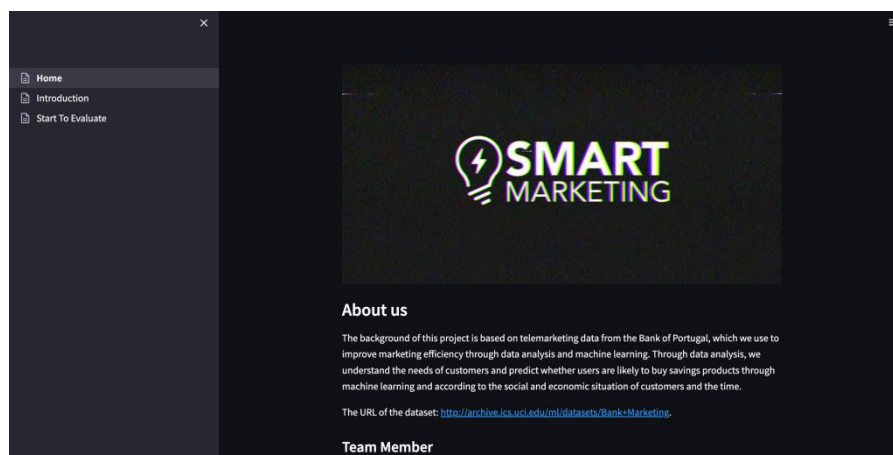


Figure 3-2

### 3.1.3 Introduction Page

The page is shown below:

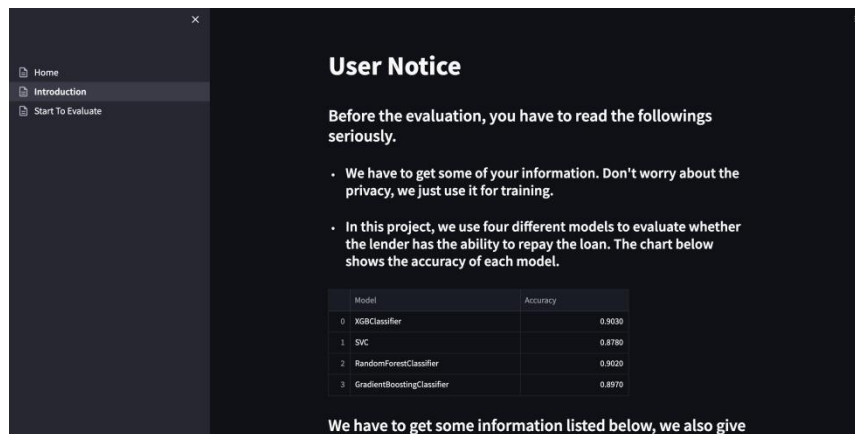


Figure 3-3

### 3.1.4 Start To Evaluate Page

The page is shown below:

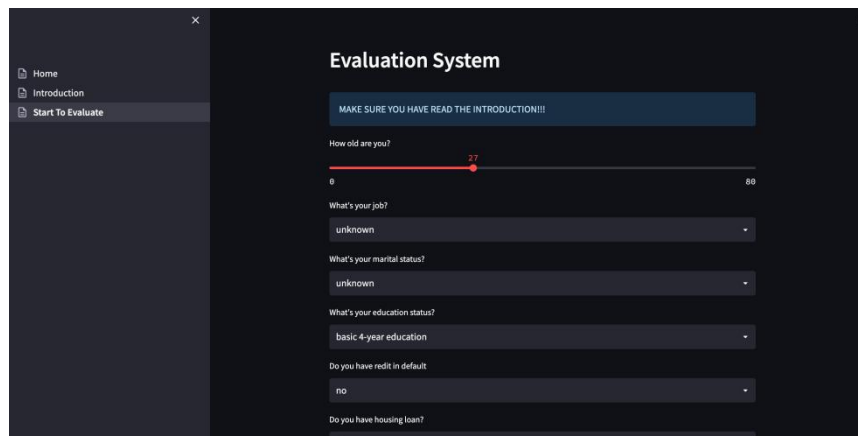


Figure 3-4

## 3.2 System Usage Instructions

### 3.2.1 Installation

Using a python IDE (e.g. pycharm), install the following dependency packages.

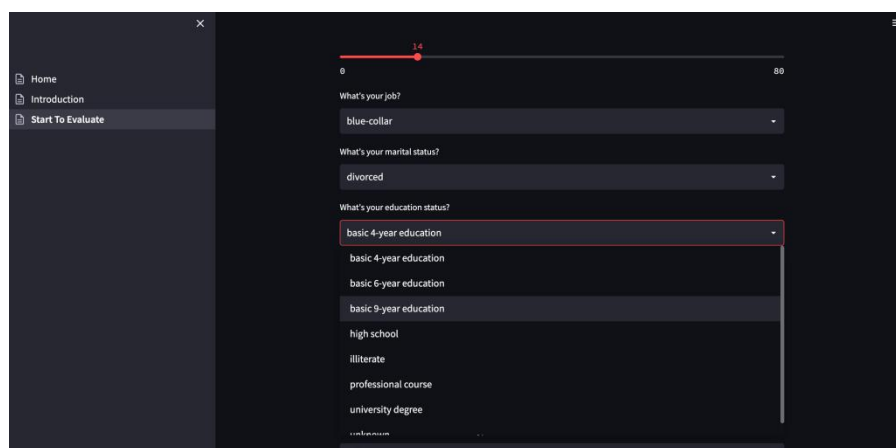
- aif360==0.5.0
- imbalanced\_learn==0.9.1
- imblearn==0.0
- joblib==1.2.0

- matplotlib==3.6.1
- numpy==1.22.4
- pandas==1.4.4
- scikit\_learn==1.1.2
- streamlit==1.13.0
- xgboost==1.6.2

Run the file using streamlit run Home.py

### 3.2.2 Details of Use

Input customer information:



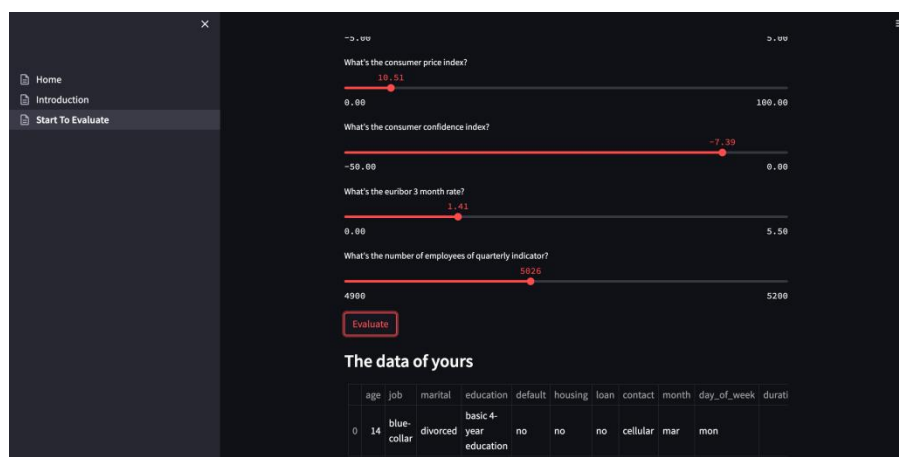
The screenshot shows a web application interface with a sidebar on the left containing links for 'Home', 'Introduction', and 'Start To Evaluate'. The main area contains three dropdown menus for inputting customer information:

- 'What's your job?' with 'blue-collar' selected.
- 'What's your marital status?' with 'divorced' selected.
- 'What's your education status?' with 'basic 4-year education' selected (highlighted with a red box).

Below the education dropdown, a list of other education options is visible: 'basic 4-year education', 'basic 6-year education', 'basic 9-year education', 'high school', 'illiterate', 'professional course', and 'university degree'.

Figure 3-5

And then start:



The screenshot shows the same web application interface, but now with economic indicators and a data table. The sidebar remains the same. The main area contains four sliders for inputting economic indicators:

- 'What's the consumer price index?' with a value of 10.51.
- 'What's the consumer confidence index?' with a value of -7.39.
- 'What's the euribor 3 month rate?' with a value of 1.41.
- 'What's the number of employees of quarterly indicator?' with a value of 5926.

Below the sliders is an 'Evaluate' button (highlighted with a red box). Underneath is a section titled 'The data of yours' which displays a table of the input data:

	age	job	marital	education	default	housing	loan	contact	month	day_of_week	duration
0	14	blue-collar	divorced	basic 4-year education	no	no	no	cellular	mar	mon	

Figure 3-6

# Chapter 4: Conclusion

## 4.1 Future Work

### 4.1.1 Algorithm Model Accuracy Optimization

The accuracy of each model should be improved. Also each dataset has one or more sensitive features which may bring bias to the model. So we have to make the most similar people be treated similarly. We can use clustering algorithms to do some post-processing on the dataset, like KNN, DBSCAN, etc.

### 4.1.2 User Experience Optimization

The current system is based on a single customer judgment, which requires users to manually select different customer attributes. In the future, the system can be optimized to allow users to make batch judgment of customers and filter out target customers who wish to receive marketing, including direct batch judgment by uploading csv files, or by reading the output data of other system modules as a sub-module of the whole business intelligence system.

## 4.2 Individual Report

### 4.2.1 Yang Tiancheng

#### Q1: Your personal contribution to the project.

- Do the data cleaning and training the model based on the existing dataset.
- Use streamlit to make the User interaction interface
- When receive the user's input data, do the data pre-processing and use different model to predict.

#### Q2: What you have learnt from the project.

- We can use machine learning to find some rules from the dataset although it's black-box.
- When using machine learning, we need to pay more attention on the dataset-cleaning.
- When we evaluate some issues in reality by using machine learning, we cannot rely on only one model, we have to use multi-model and aside by majority's prediction.

#### Q3: How you can apply this in future work-related projects.

- Before training models, we have to do proper data-cleaning and we have to drop out some irrelevant features.
- Some dataset has some sensitive feature(like age, sex and etc.), we have to avoid the influence brought by this feature.
- We have to adjust the model structure to improve the accuracy

### 4.2.2 Zhang Junmengyang

#### Q1: Your personal contribution to the project.

Project Management:

- Project organization asset management: accumulate the gains from the project and the application of relevant technologies. Ensure team members can work together as a team with greater efficiency the next time they work together.
- Project Schedule Planning: Responsible for the management of the project schedule in the early stages and develop the project advancement schedule

#### Requirement Analysis:

- Responsible for the preliminary market research work of the project, through the review of industry reports and other information, to understand the market size and determine the need for
- Determine the main product features and presentation, including part of the UI design
- Discuss and optimize user experience with team members through product prototype designing

#### Team Support

- Responsible for writing project reports
- Responsible for the production of business background video for the project

#### Algorithm modeling

- Participate in the selection of data sets and preliminary data analysis
- Participate in some data pre-processing work and some algorithm module writing
- Participate in the evaluation and comparison of algorithm models

### **Q2: What you have learnt from the project.**

#### Skills Learning

- Be able to use streamlit to build a basic web application quickly
- Proficiency in using XGBClassifier, SVC, RandomForestClassifier and GradientBoostingClassifier.
- Have a deeper understanding of project management theory and agile development principles
- Become familiar with the business process of an unfamiliar domain

#### Insights

- The profit from current product production is not only related to the manufacturing itself, but also related to the market environment and product positioning, and the future industrial artificial intelligence technology will involve more factors related to product manufacturing.
- Big data is not only a large amount of data, but also a large variety of data. The requirement for real-time is strong. The value contained in the data is large. Big data exists in all industries, but the many information and consultation are diverse and complex, requiring searching, processing, analyzing, summarizing, and concluding its deep laws to obtain valuable data.

### **Q3: How you can apply this in future work-related projects.**

- In addition to focusing on the underlying technology at the algorithm model level, we should also focus on specific business processes and look for industry pain points, such as the lack of data correlation in banks leading to a large amount of data not being used to its value.

- It is also necessary to consider the construction of business systems, such as automatic data collection, data processing, data analysis, use of data, and finally the generation of new data. As a business optimizer, you need to consider the automation means to improve business efficiency and make the enterprise form a closed loop of data.

### **4.2.3 Tian Qingyun**

#### **Q1: Your personal contribution to the project.**

In the entire team project, i discussed the project theme with the group members, and selected the project on the theme of smart marketing from domains such as recommendation, optimization and hybrid machine reasoning based on the knowledge learned in Intelligent Reasoning System. I participated in the collection of datasets, and selected the most matching dataset, pre-processed the data using one-hot coding and standardization, and participated in the training of the model using the dataset. In the same time, i participated in the establishment of algorithm models in the algorithm group, learned and used the XGBClassifier, SVC, RandomForestClassifier, and GradientBoostingClassifier algorithms to complete the core algorithm part of the project. Moreover, I also optimized the model, tested and adjusted various parameters, made a choice between time and accuracy, and adjusted the most suitable model for this project. And I also participated in the optimization of the front-end interface display of streamlit to make it more clear and clear. In this project, I tried to use the vue framework to complete the front-end page display, and cooperated with the django back-end framework to realize an intelligent loan judgment system that separates the front-end and the back-end, but due to some difficulties encountered, it was not completed.

#### **Q2: What you have learnt from the project.**

In this project, i learned how a software related intelligence constructed from 0 to 1, how to start, how to develop and how to finish. I learned a lot from this project. I learned how to use a dataset, from data selection , data pre-processing, to training the model with the data, and testing the model with the data. As well as understanding and learning streamlit, it is a very convenient and useful library that can quickly build a web application. Through this project, I also have a further understanding and learning of many machine learning algorithms and models. How to apply these models to suitable projects is a very important and critical point. Although many of these have been learned in the classroom, it is also the first time that I have participated in and completed the application. And this is a cooperative project. How to correctly and reasonably assign tasks with the students in the group, cooperate with each other, and communicate with each other is also an important experience I have learned in this project. This experience is very essential for us.

#### **Q3: How you can apply this in future work-related projects.**

I have learned a lot in this project and it is also a great asset, valuable experience that I will apply to my future work in the future. The first is teamwork, which requires good communication with colleagues in the department and if there is any difficulty, the team will solve it in time. which greatly improves work efficiency. And the machine learning-related knowledge I learned in this project is also very useful, making me more familiar with

machine learning, so that I can select and apply different machine learning algorithms to a specific project in my future work. I believe this is also what an AI engineer needs to consider and have. Moreover, in this project, I also learned the relevant knowledge of the front and back ends, which can be regarded as an early experience of the separation of the front and back ends in the company, and the cooperation between the front and back ends, which can also be very good for me to apply to future work.

#### **4.2.4 Song Enyu**

##### **Q1: Your personal contribution to the project.**

Our project is divided into four parts, background research, training the model, improving the accuracy of the model, and developing the system based on the model. In our project, I was mainly responsible for the system developing using streamlit. In the early stage, I discussed the issue of data dimensions with the team members and determined the models we used for training. Then, the page is designed according to the required data dimensions, and the existing models are executed according to the value obtained from the front-end page to perform the classification operation, and finally the result is displayed on the pages. In terms of data preprocessing, the processing of attribute values PDAYS, DURATION, and EURIBOR3 attributes has been completed.

##### **Q2: What you have learnt from the project.**

In this project, I first learned how to use python language to develop a small front-end and back-end system with UI. I haven't been exposed to developing systems with python before, so getting started is a bit difficult. After research on the technology stack, streamlit is suitable for beginners and more suitable for the development of projects that are strongly related to data. So I chose streamlit framework and learned it. Secondly, since the front-end interface in streamlit uses the markdown language, I also learned the related use of markdown. Also, during development, we used git. I have no experience in using git before, so this time I learned git-related commands, such as GIT ADD, GIT BATCH, etc., and the git operations ADD, COMMIT, PULL, PUSH. In addition, since our model is an stacking model that integrates GradientBoost, RandomForest, XGBoost, SVC, I also learned the related models involved and the way of integration.

##### **Q3: How you can apply this in future work-related projects.**

After getting to know git, git is a very popular devops tool in the company. Therefore, after mastering the relevant knowledge of git, the trouble will be reduced accordingly in the future work. Secondly, it is a good method to stack the model. Although our project solves a binary classification problem, the idea of stacking can be applied to the prediction problem, perhaps using linear regression, XGBoost, GradientBoost, etc. for the house price forecast. Finally, the development of the system is also very important. The end user needs a full UI to interact with the system instead of typing in CMD. For stakeholders, a system with a set of commercial value should also be complete and highly available. With a complete front-end and back-end system, the value of the model will be more prominent, and the model will be truly practical.



## Chapter 5: Reference

- [1] <https://scikit-learn.org/stable/>
- [2] [https://xgboost.readthedocs.io/en/stable/python/python\\_api.html](https://xgboost.readthedocs.io/en/stable/python/python_api.html)
- [3] [https://blog.csdn.net/qq\\_41731517/article/details/102535860](https://blog.csdn.net/qq_41731517/article/details/102535860)
- [4] <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>
- [5] <https://zhuanlan.zhihu.com/p/37702043>
- [6] <https://scikitlearn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>
- [7] <https://www.datacamp.com/tutorial/random-forests-classifier-python>
- [8] <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html>
- [9] <https://stackabuse.com/gradient-boosting-classifiers-in-python-with-scikit-learn/>
- [10] <https://www.cnblogs.com/Mangnolia/p/13124486.html>