Table of Contents

Preprocess

Introduction

This project explored and preprocessed a dataset which is related with direct marketing campaigns of a Portuguese banking institution. The dataset could be divided into three parts: the bank client information, campaign data, and social and economic index.

The exploration section includes the identification of the attribute type, basic statistic of the data, clusters, and outliers. The preprocessing section covers the binning, normalisation, discretion, and binarisation. The last section summarises the significant findings in all.

The data is mainly processed by `Pandas` and `Scikit-learn` and plotted by `matplotlib` and `seaborn`. Some parts are processed by `KNIME`.

1A Initial Data Exploration

Attribute Type

Attribute	Description	Type
`age`	Age of the client	Ratio
`job`	Client's occupation	Nominal
`marital`	Marital status	Nominal
`education`	Client's education level	Nominal
`default`	Indicates whether the client has credit in default	Nominal
`housing`	Indicates whether the client has a housing loan	Nominal
`loan`	Indicates whether the client as a personal loan	Nominal
`contact`	Type of contact communication	Nominal
`month`	Month that last contact was made	Nominal

day_of_week	Day that last contact was made	Nominal
duration	Duration of last contact in seconds	Ratio
campaign	Number of contacts performed during this campaign for this client (including last contact)	Ratio
pdays	Number of days since the client was last contacted in a previous campaign	Ratio
previous	Number of contacts performed before this campaign for this client	Ratio
poutcome	Outcome of the previous marketing campaign	Nominal
emp.var.rate	Employment variation rate (quarterly indicator)	Ratio
cons.price.idx	Consumer price index (monthly indicator)	Ratio
cons.conf.idx	Consumer confidence index (monthly indicator)	Ratio
euribor3m	Euribor 3-month rate (daily indicator)	Ratio
nr.employed	Number of employees (quarterly indicator)	Ratio

Table 1 Attribute type

Data cleaning

Missing values

Attribute	Proportion	Action	Reason
job	0.45%	Drop	**Small amount**

marital	0.1%	Drop	**Small amount**
education	4.05%	Drop	**Small amount**
default	21.25%	Drop the column	**We can not infer the default status. By removing the 'unknown' entries, the 'default' column only contains 'yes' providing no helpful information to classify the client.**
housing	2.95%	Drop	**Small amount**
loan	2.95%	Drop	**Small amount**

Table 2 Missing values

One interesting finding is, in this data set, if a client did not provide `housing` status, these is no information about loan as well.

Consistency checking

As mentioned in the attribute description, a value of '999' in the `pdays` means the client is new. Thus, the corresponding `poutcome` would be 'nonexistent'. However, 193 entries are inconsistent with it counting 9.65%. They have been removed.
So far, 16.15% data has been dropped, and 1677 entries left.

Combination of values

To reduce the complexity in the later computing, some values have been combined.

Attribute	Value	Replaced with
job	blue-collar, technician	blue-collar
job	service, housemaid	service
education	basic.4y, basic.6y, basic.9y	basic

Table 3 Value replacement

Statistics

age

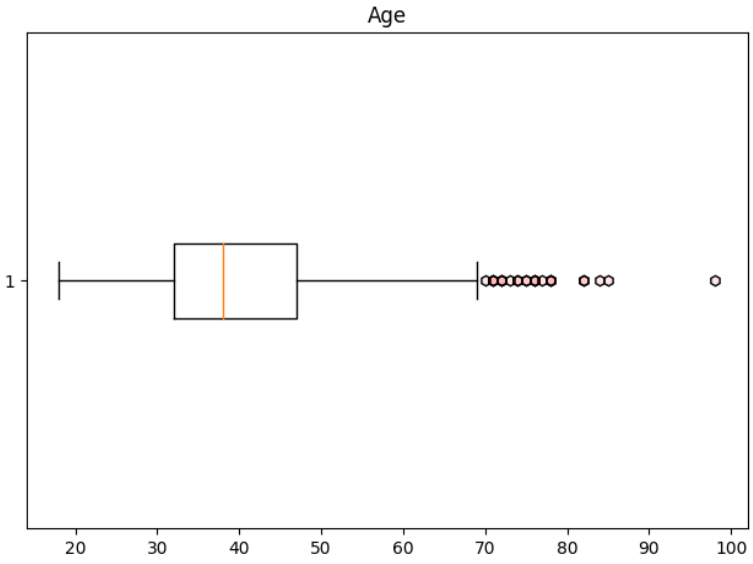

Figure 1 Box plot of age

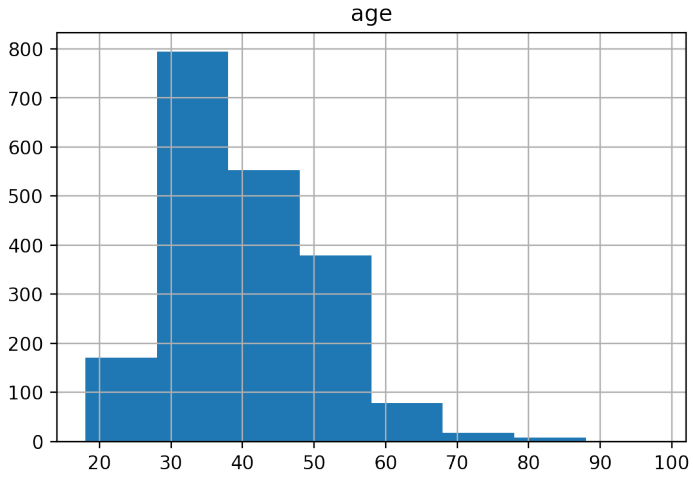

Figure 2 Histogram of age

The age of the client ranges from 18 ~ 98. The average age is 40, and the median is 38. According to the frequency, most of the client are aged from 30 - 40. The mode is 31.

job

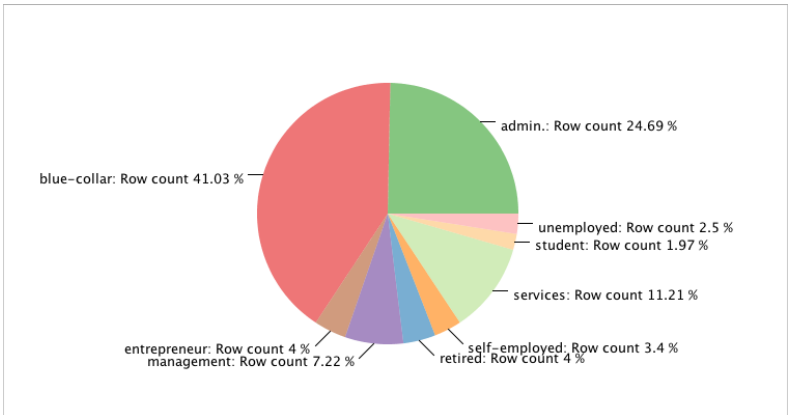

Figure 3 Jobs

The *job* has nine values (combined) including 688 (41%) blue-collar, 414 (25%) admin, 188 (11%) service, 121 (7%) management, 67 (4%) entrepreneur, 67 (4%) retired, 57 (3%) self-employed, 42 (3%) unemployed, and 33 (2%) student.

marital

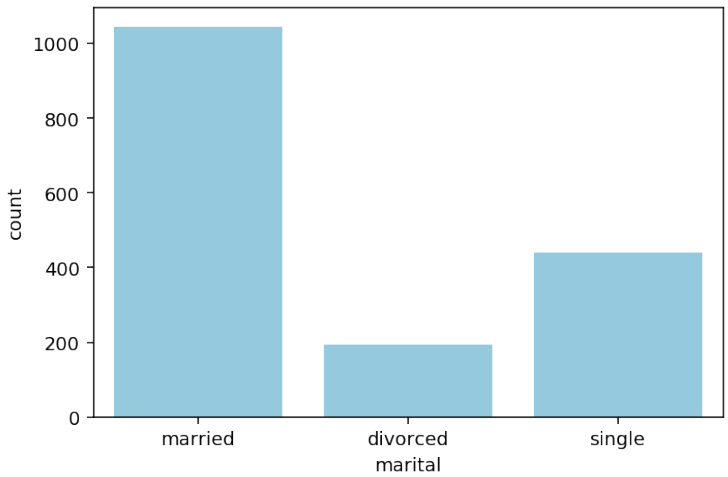

Figure 4 Marital status

The *marital* has three values including 1045 (62%) married, 439 (26%) single, and 193 (12%) divorced.

education

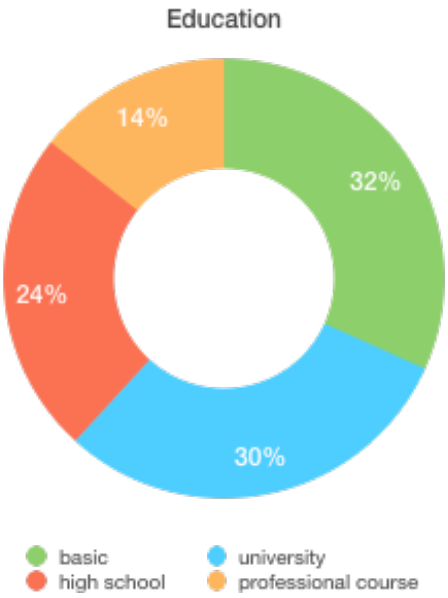

Figure 5 Education

The `education` among the clients varies. After combining, there are 532 (32%) basic, 400 (24%) with high school, 241 (14%) with professional course, and 504 (30%) with university degree.

housing

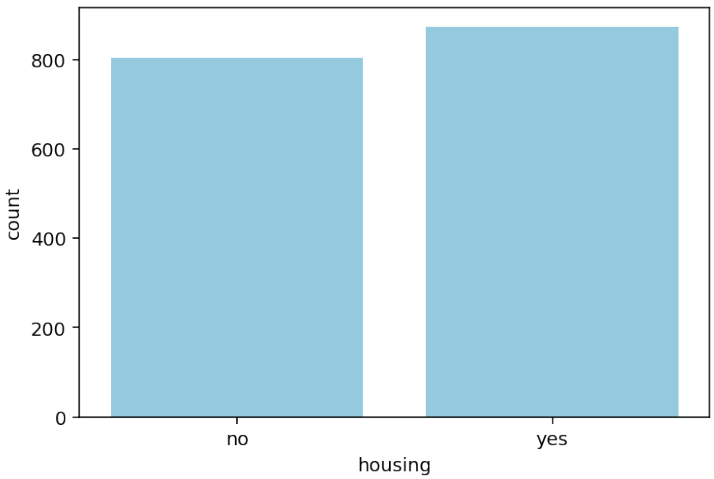

Figure 6 Housing

The housing status is half-and-half. 873 (52%) clients has one or more properties and 804 (48%) do not.

loan

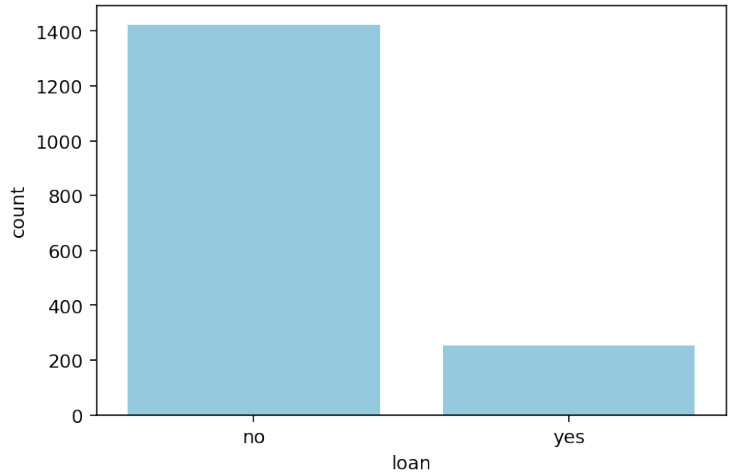

Figure 7 Loan

Most of the clients do not have loans which count 85% (1423). Meanwhile 254 (15%) clients have loans with them.

contact

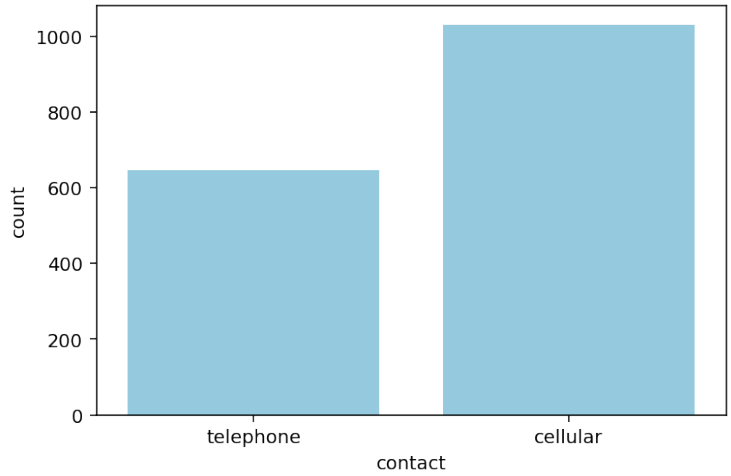

Figure 8 Contact

The `contact` records the method of communication. As showed above, most of the client (1272, 61%) are contacted by mobile, while there are still 728 (39%) contacted by landline.

month

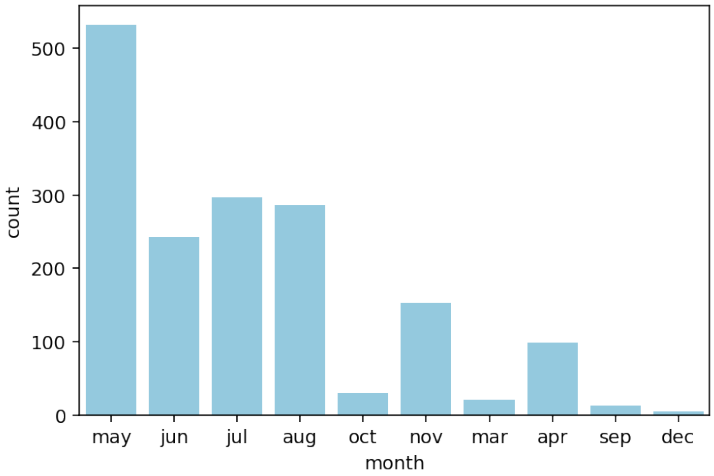

Figure 9 Month

The campaign data points do not cover the whole year. According to the histogram, the campaign had more activities in the second and third quarters. Following are the details: March 24 (1%), April 146 (6%), June 275 (14%), July 322 (18%), August 310 (17%), September 21 (0.1%), October 36 (1.8%), November 197 (9%), December 8 (0.3%).

day_of_week

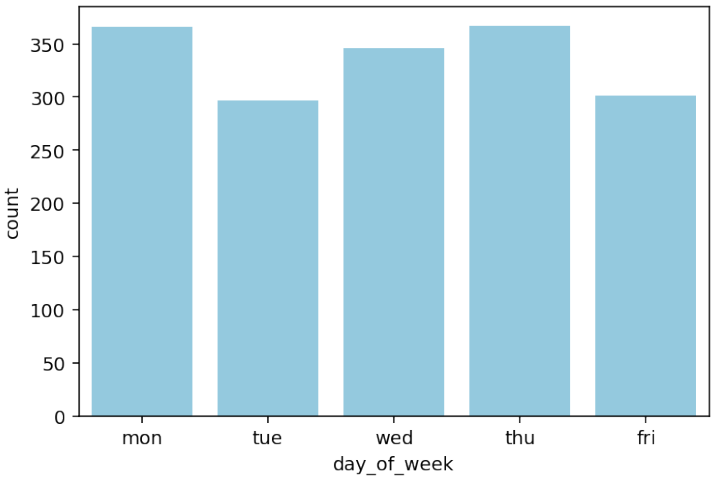

Figure 10 Day of week

The *day_of_week* distributes very equally. Following is the detail:

Monday	437	22%
Tuesday	345	18%
Wednesday	419	21%
Thursday	435	22%
Friday	364	18%

Table 4 Counts of day of week

duration

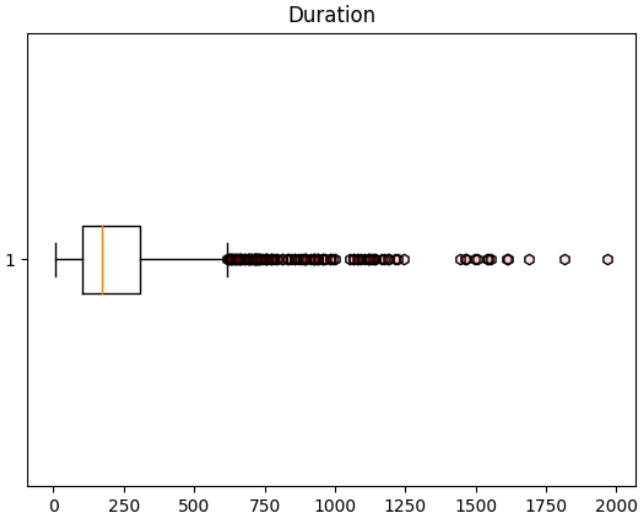

Figure 11 Duration

The *duration* indicates the duration of the calling. It varies a lot. The longest is 1970s while the shortest is 5s. However most of the data points are in the range of 9 - 491s (mean±std) with many outliers exceeding 500s. The average duration is 250. The standard deviation is 241.

campaign

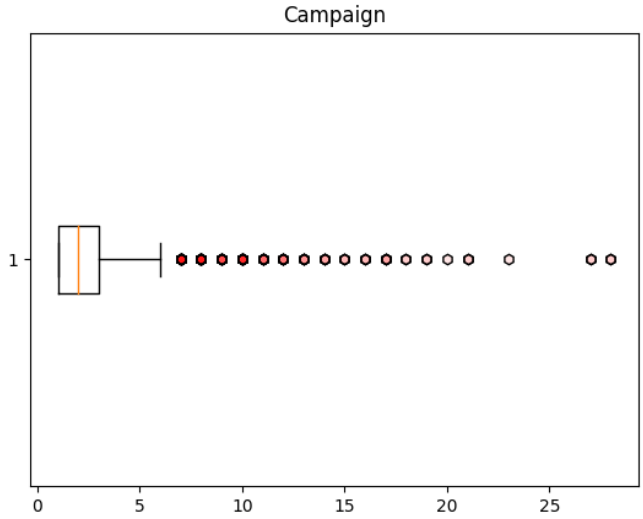

Figure 12 Box plot of campaign

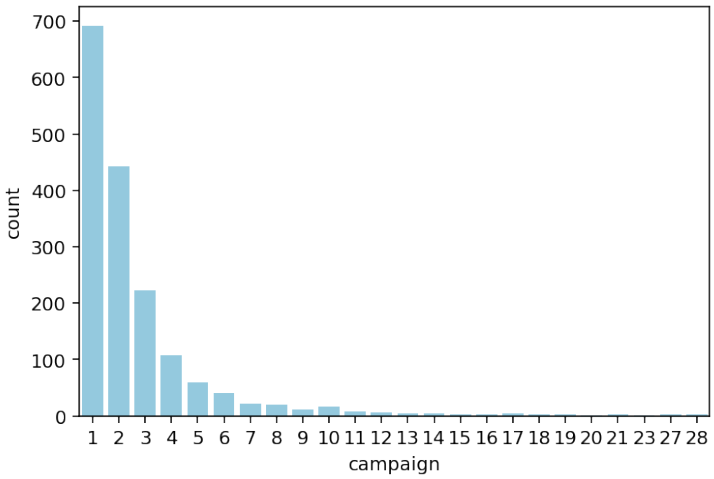

Figure 13 Histogram of campaign

The `campaign` records the round of campaign. It varies from 1 to 28. On average, clients have 2.7 campaigns. Most of the clients (692) have been contacted once counting 41%. 442 (26%) clients have two campaigns, and 222 (13%) have three. The rest counts not too much.

pdays

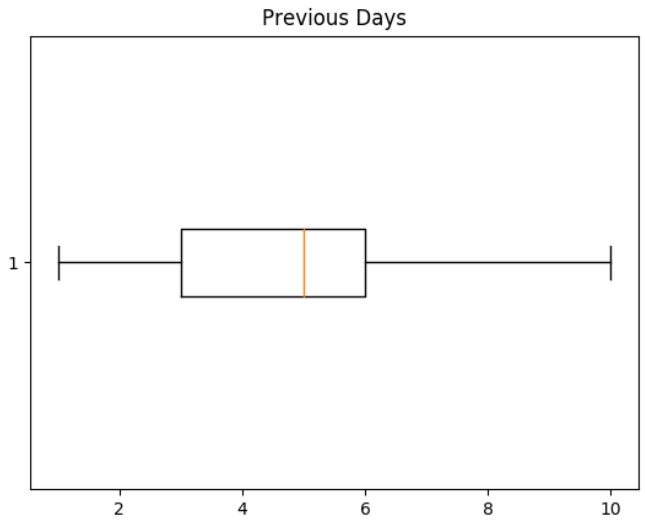

Figure 14 Box plot of pdays

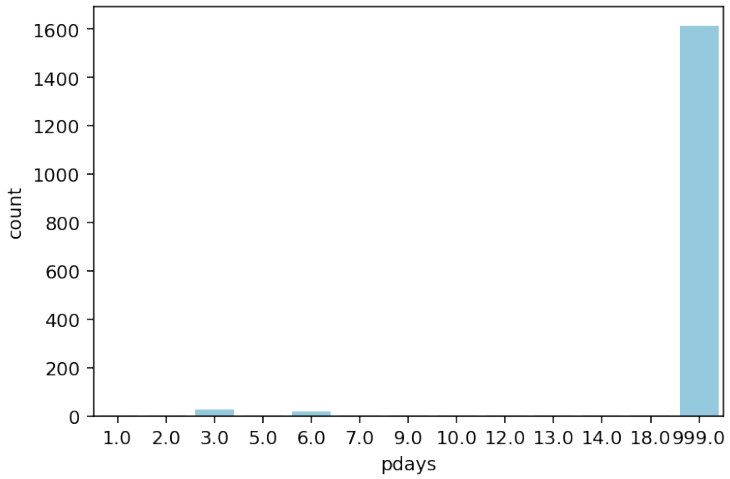

Figure 15 Histogram of pdays

As 96% of the data points in the *pdays* are '999' (i.e. new client), the whole column has been dropped for no helpful information.

previous

As `previous` and `pdays` are highly correlated, 96% of the `previous` is 0 correspondingly. The *previous* attribute is dropped.

poutcome

poutcome shows the result of last campaign. Unfortunately, in this data set, most of the data points (1614, 96%) are from new clients. Only 59 (3.5%) clients would subscribe a term deposit. 4 (0.2%) said no.

emp.var.rate

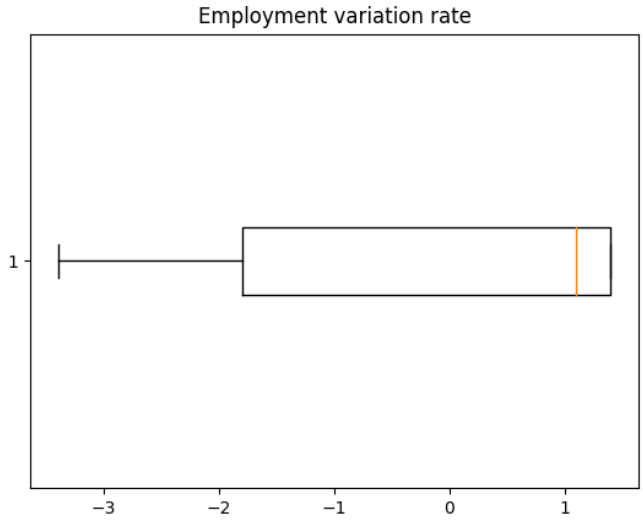

Figure 16 Box plot of employment variation rate

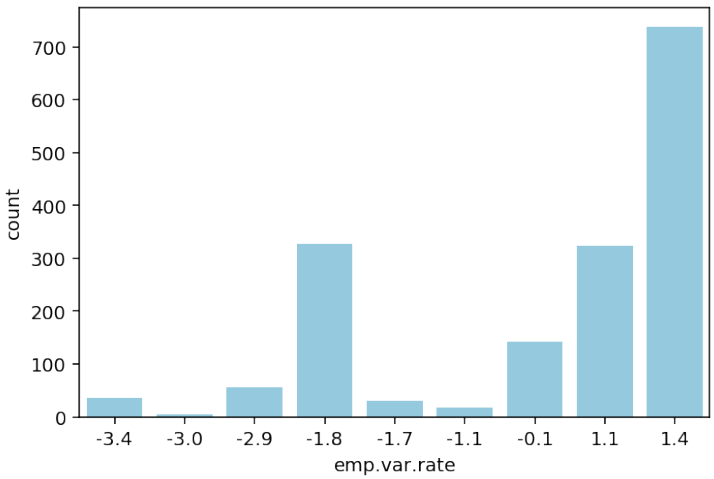

Figure 17 Histogram of employment variation rate

emp.var.rate is the quarterly employment variation rate in Portugal. It ranges from -3.4 to 1.4. The average is 0.25. Following are the details,

employment variation rate	Count	Portion
1.4	792	44%
1.1	358	19%
-0.1	180	9%
-1.1	27	1%
-1.7	38	2%
-1.8	474	21%
-2.9	76	3%
-3.0	8	0.3%
-3.4	47	2%

Table 5 Counts of employment variation rate

cons.price.idx

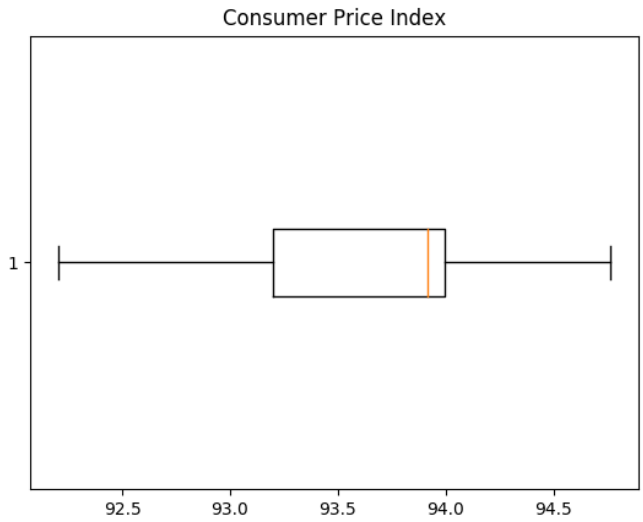

Figure 18 Box plot of CPI

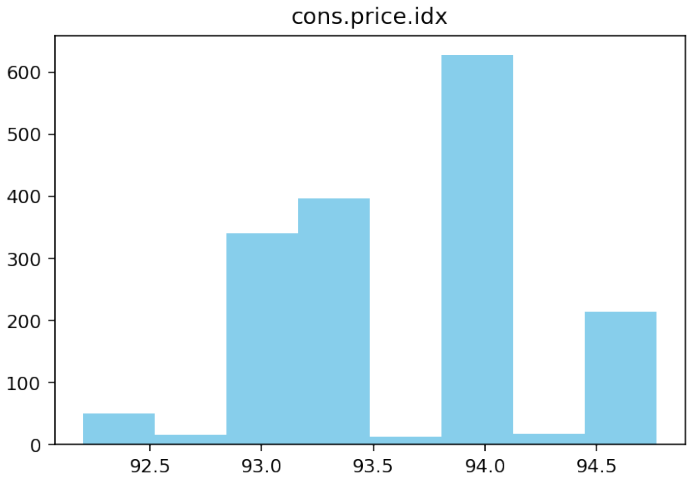

Figure 19 Histogram of CPI

`cons.price.idx` (Consumer Price Index, CPI) indicates the price of common goods in the market. It ranges from 92.201 to 94.767. On average CPI is 93.617. The top 5 are:

CPI	Count	Portion
93.994	323	19%
93.918	278	17%
93.444	257	15%
94.465	203	12%
92.893	202	12%

Table 6 CPI

cons.conf.idx

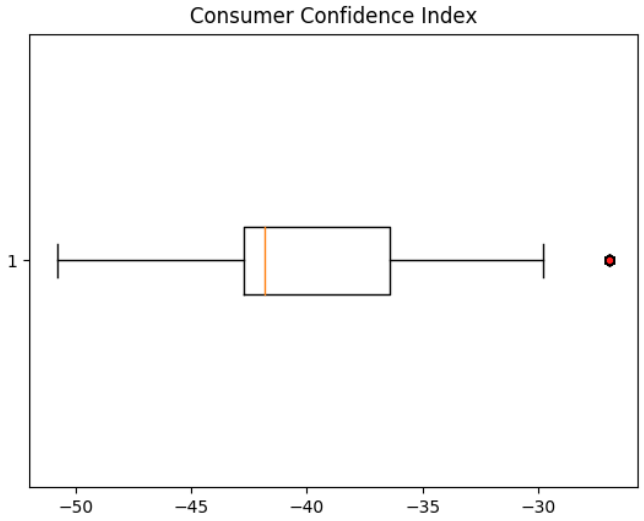

Figure 19 Box plot of CCI

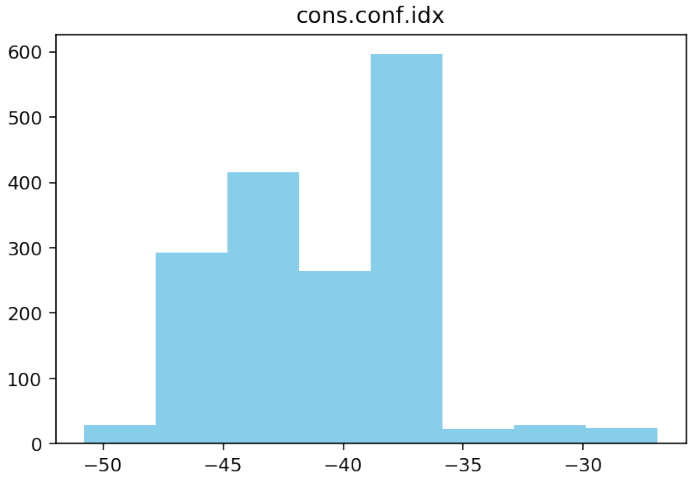

Figure 20 Histogram of CCI

cons.conf.idx (Consumer Confidence Index, CCI) reveals the confidence of consumers in the market monthly. This index ranges from -50.8 to -26.9 with average of -40.4.

euribor3m

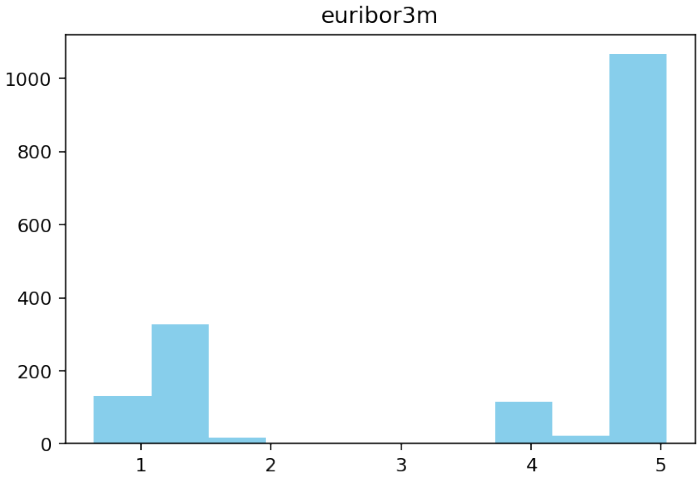

Figure 21 Histogram of Euro Interbank Offered Rate

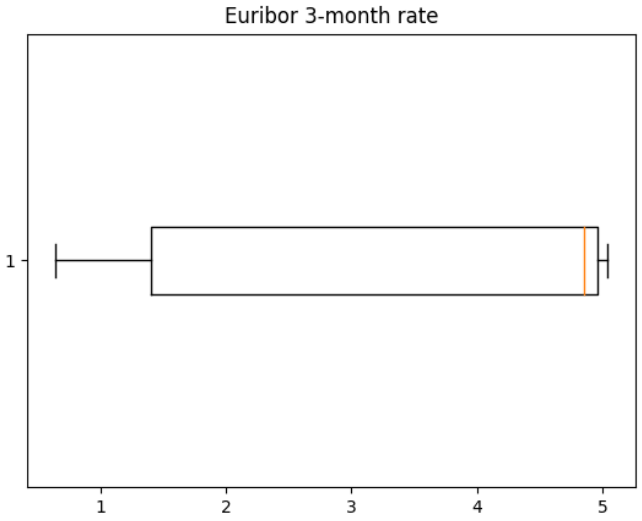

Figure 22 Box plot of Euro Interbank Offered Rate

euribor3m (Euro Interbank Offered Rate) is a daily rate of euribor 3 month. The minimum is 0.638. The max is 5.045. The average is 3.806.

nr.employed

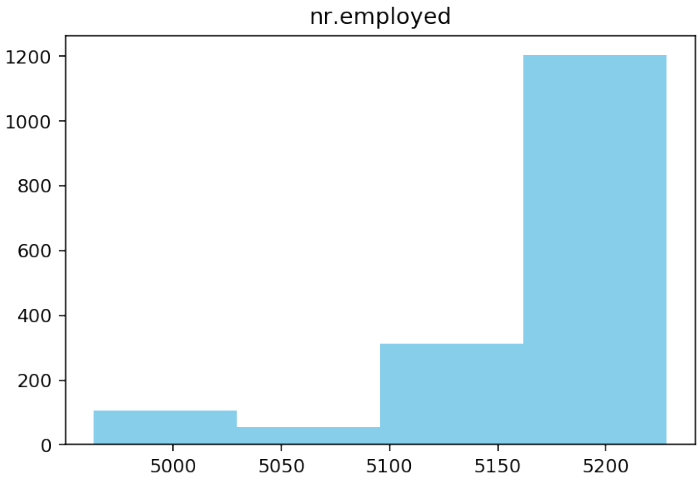

Figure 23 Histogram of number of employed

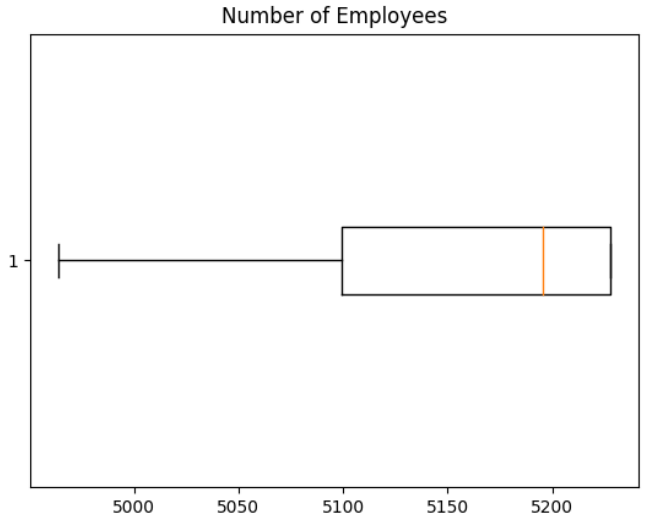

Figure 24 Box plot of number of employed

nr.employed is the indicator of number of employees quarterly. It has a range from 4963.6 to 5228.1. The average is 5174.8.

y

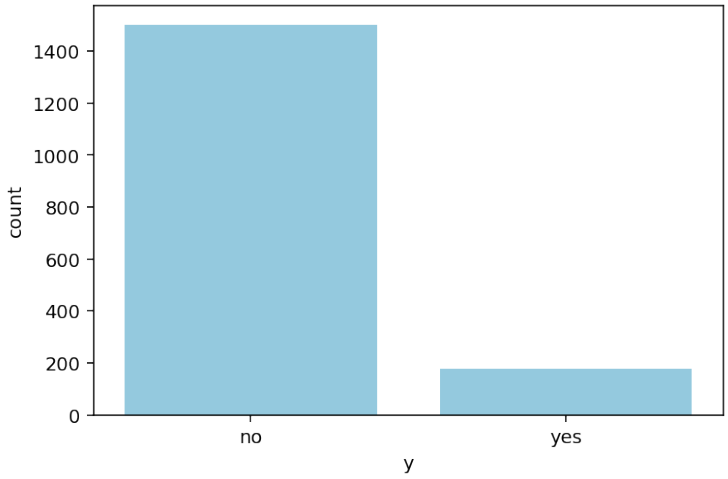

Figure 25 Description of the team deposit

y indicates whether the client subscribe a term deposit during the campaigns. 1501 (90%) out of 1677 clients did not subscribe. 176 (10%) clients did.

Clustering and outliers

duration

In the *duration*, there is one outlier. Its (row id 23355) duration is 2926 i.e. almost 50 minutes. While the average of duration is just 250s.

campaign

In the *campaign*, seven entries are abnormal. They all exceed 20 campaigns which can be seen in the box plot above.

age and duration

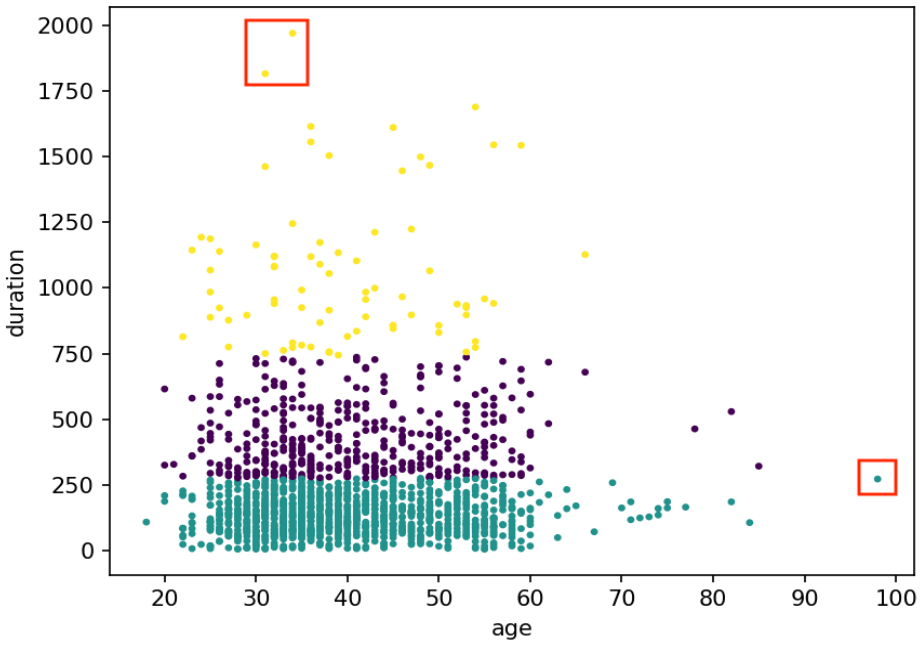

Figure 26 Age and duration

The graph shows three clusters between `duration` and `age`. Some clients' age are far older than else in the common duration range, thus they are tagged as outliers. Besides, some mid-age clients' duration exceed the average duration (250s) more than 5 standard deviation (241s).

age and euribor3m

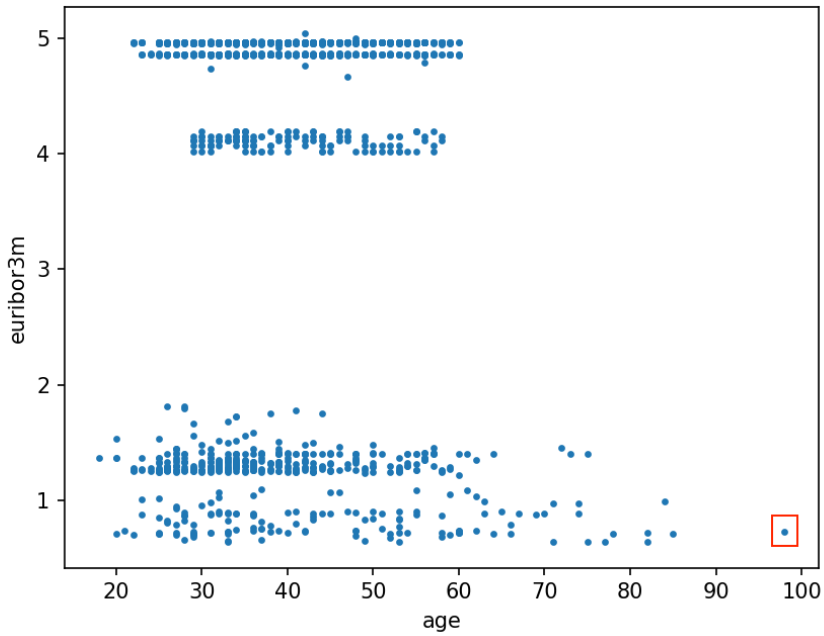

Figure 27 Age and euribor3m

In most of the cases, each euribor3m rate has all ages of clients, except for some clients are way older than others. The euribor3m splits the data points into 4 clusters.

age and number of employed

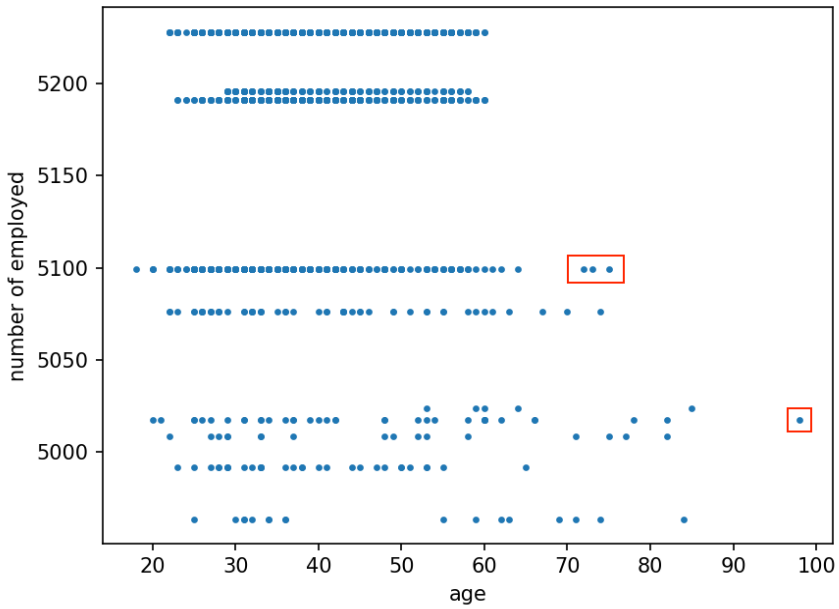

Figure 28 Age and number of employed

The number of employed is another indicator of the financial status. When the number of employed goes high, the average age of the clients is younger. Thus some elderly clients in the scatter plot turns to the outliers. It is also clear that the data points can be clustered by the number of employed.

number of employed and employment rate

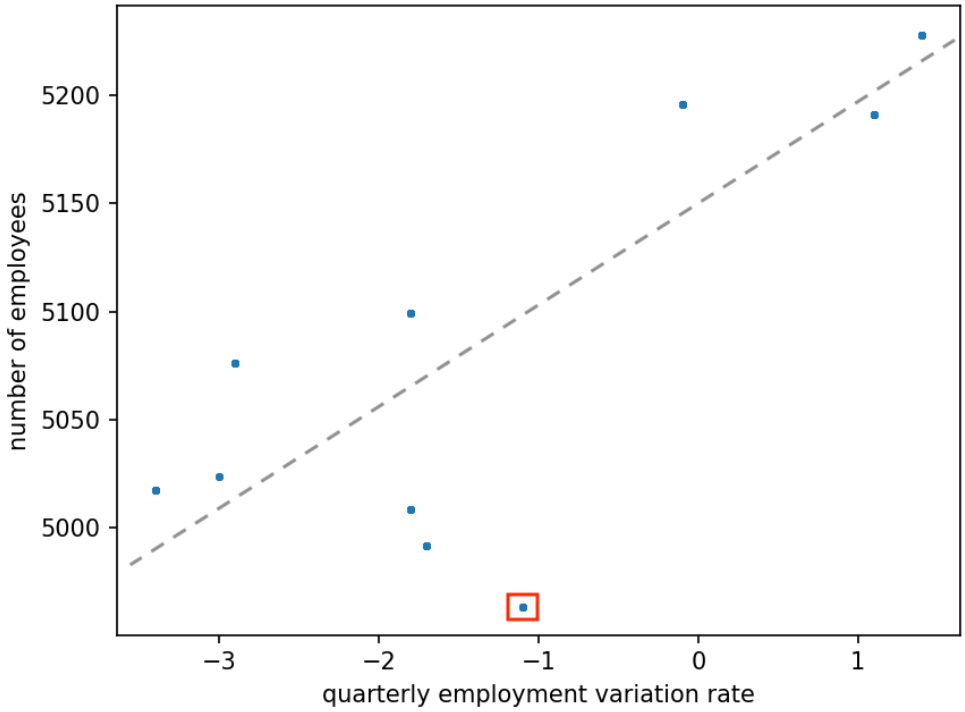

Figure 29 Number of employed and employment rate

By regression, most of the data points distribute aside line meaning the employment variation rate increases with the growth of employed. While the point at (-1.1, 4963.6) is the outlier.

1B Data preprocessing

Binning

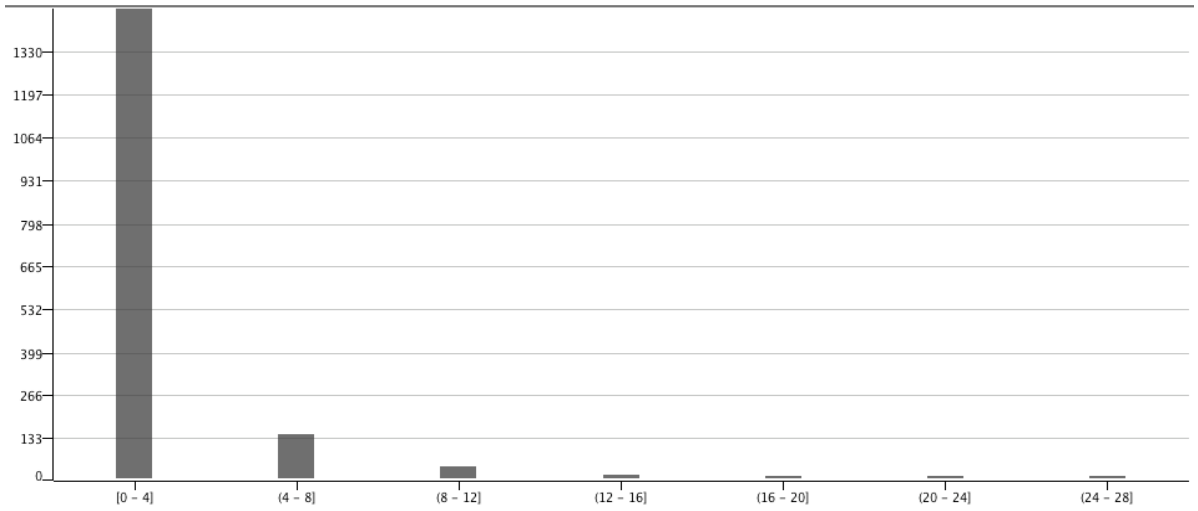

Figure 30 Binning test

Before binning, there are 24 different values in `campaign`. As shown in the histogram, small amount entries holds most of the unique values (12 - 18). It is reasonable to smooth this part out.

Equal-width binning

Since the unbalance of campaign entries, it is hard to implement equal width binning. For example, a 4-width bin would hold 87% data points causing over-smoothing, meanwhile leaves the rest 13% into 6 bins. Therefore, to keep the detail in the 1-4 campaign, the width has to be as small as possible (i.e., 2). By doing this, campaign 1-4 allocate into 2 kinds of bins. For better interaction while working out the proper number. The binning was done in KNIME.

Following is the steps:

1) Add `Auto-Binner` node to workflow.
2) Connect `Auto-Binner` to CSV Reader.
3) Configure `Auto-Binner` to equal-width.
4) Choose the maximum bin number which is 14.
5) Choose `Borders` for the bin naming and force integer bounds.
6) Execute `Auto-Binner` node.

7) Use `CSV Writer` to export the binned data.
8) Use `Histogram` to visualize the data.

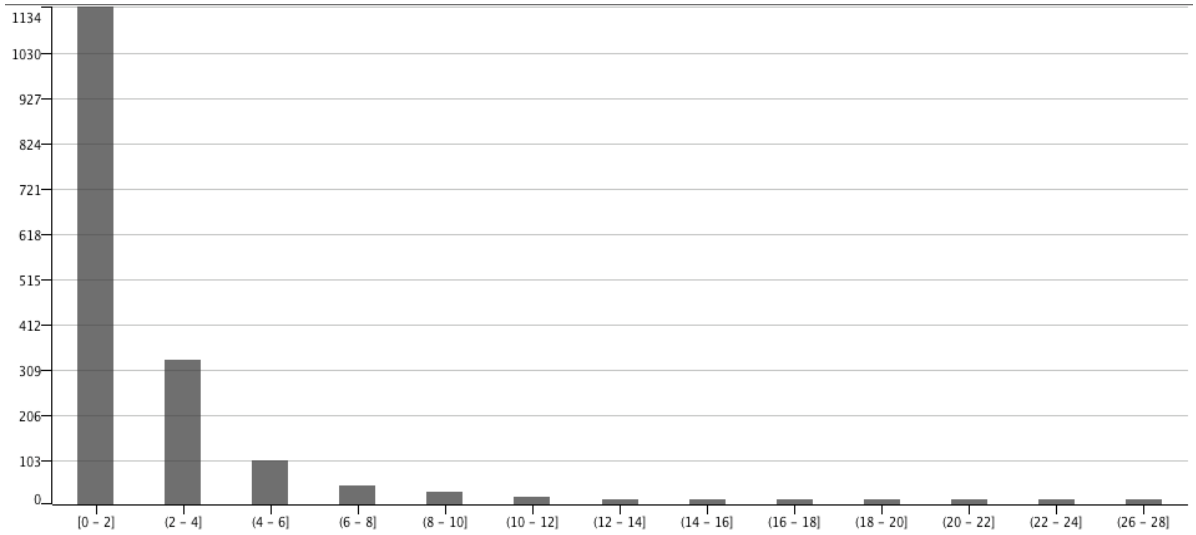

Figure 31 Equal-width binning

Equal-depth binning

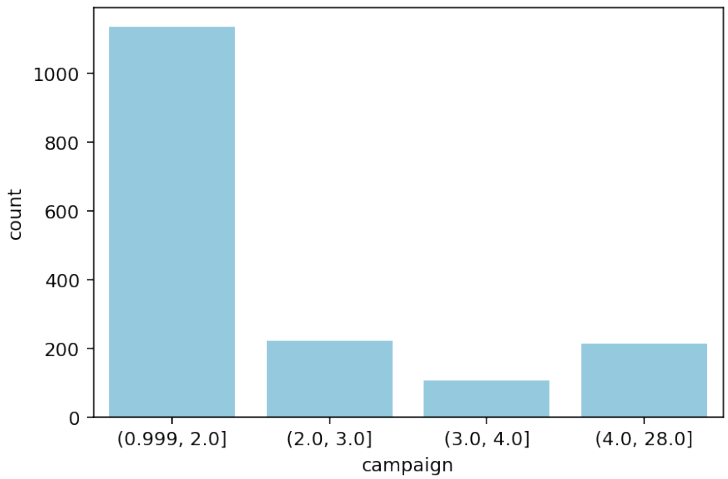

Figure 32 Equal-depth binning

In equal-depth binning, we set 6 bins for them. Each bin has around 200 data points. If using less than 6 bins, campaign 1-4 would allocate into 3 or fewer bins losing the detail of the data. If the bin number is more than 6, campaign times 4-28, which only count 13%, would hold more than one bins. It is unnecessary. So, 6 is the best bin number holding around 200 in each bin.

Following is the steps:

1) Add `Auto-Binner` node to workflow.
2) Connect `Auto-Binner` to CSV Reader.
3) Configure `Auto-Binner` to equal-frequency.
4) Choose a staring bin number, such as 4.
5) Adjust the bin number according to the smooth and empty bins.
6) Choose `Borders` for the bin naming and force integer bounds.
7) Execute `Auto-Binner` node.
8) Use `CSV Writer` to export the binned data.
9) Use `Histogram` to visualize the data.

Normalisation

Min-max normalisation

It is simple to perform normalisation by *MinMaxScaler* in Scikit-learn. Simply input the `Series` into scaler's *fit_transform* functionality is enough. By default, the range is 0 to 1 which is required. Also, the values are round to two digits. Following is the code:

```
scaler = MinMaxScaler()
duration_scaled =
scaler.fit_transform(df.duration.astype(float).values.reshape(-1, 1))
df_processed['duration_0-1_normalisation'] = duration_scaled.round(2)
```

Z-score normalisation

The z-score normalization is used to calculate the distance of spherical data in clustering. In this project, it is calculated by following expression / code:

```
df.duration - df.duration.mean())/df.duration.std(ddof=0)
```

***ddof :** Delta Degrees of Freedom. The divisor used in calculations is N – ddof, where N represents the number of elements.

Part of the result:

row id	duration_*0-1*_normalisation	duration_*z-score*_normalisation
5	0.1	-0.21
61	0.12	-0.07
78	0.74	5.04
84	0.14	0.15
85	0.1	-0.22
86	0.13	0.03
100	0.12	-0.01

Table 7 Normalisation

Discretion

The discretion is done by the *cut* method in `Pandas`.

```
pd.cut(x=df.age, bins=[0, 35, 60, 100], labels=['Adult', 'Mid-age', 'Old-age'])
```

It divides the data points according to the `bins` parameter and assign each section with the name in the `labels` array.

Part of the result:

row id	age_discrete
5	mid
61	mid
78	mid
84	aged
85	mid
86	mid
100	aged
105	mid
126	mid

Table 8 Discretion

The frequency of age is,

Mid-age: 1115
Adult: 284
Old-age: 278

Binarisation

Pandas provides a method called `get_dummies` to perform one-hot encoding which is exactly the binarization required in this part.

```
pd.get_dummies(df.marital)
```

It turns all the nominal data into separate columns and assign them 0 or 1.

Part of the result:

row id	divorced	married	single
5	0	1	0
61	0	1	0
78	0	1	0
84	1	0	0
85	0	0	1
86	0	1	0
100	1	0	0
105	1	0	0

Table 9 Binarisation

1C Summary

Inconsistent entries

As mentioned above, `pdays` and `poutcome` have 193 (9.65%) inconsistent entries. The recording of the data has to be improved.

Housing and loan

There is an interesting link between *housing* and *Loan*. If a customer does not have properties, nor does he or she have a loan.

Economic and social indicator

There are strong relationships between economic and social indicator. Obviously, when employment variation rate is high, there are more people employed. With more people earning money, the expense on the goods grows. Therefore, consumer price index has a positive correlation with employment variation rate.

Euro Interbank Offered Rate and subscription

For the economic data, when Euribor is low, meaning the financial liquidity is high, people have more money to deposit. So, when the financial status is not well, it is not a good idea to perform a marketing about the term deposit.

Campaign outcome and subscription

Not all the clients who decided to subscribe a term deposit on the phone did subscribe in the end. In this data set, 59 clients had a *success* in their *poutcome* attribute while only 38 (64%) clients have subscribed eventually. Therefore, the reliability of *poutcome* is not very high in the prediction.

Data Mining

Introduction

This project aims to predict the subscription of the deposit based on provided data set. The data set is from direct marketing campaigns of a Portuguese banking institution.

There are 21 attributes provided which can be divided into the bank client information, campaign data, and social and economic index.

The detail of the attributes is list below.

Attribute	Description	Type
age	Age of the client	Ratio
job	Client's occupation	Nominal
marital	Marital status	Nominal
education	Client's education level	Nominal
default	Indicates whether the client has credit in default	Nominal
housing	Indicates whether the client has a housing loan	Nominal
loan	Indicates whether the client as a personal loan	Nominal
contact	Type of contact communication	Nominal
month	Month that last contact was made	Nominal
day_of_week	Day that last contact was made	Nominal
duration	Duration of last contact in seconds	Ratio

campaign	Number of contacts performed during this campaign for this client (including last contact)	Ratio
pdays	Number of days since the client was last contacted in a previous campaign	Ratio
previous	Number of contacts performed before this campaign for this client	Ratio
poutcome	Outcome of the previous marketing campaign	Nominal
emp.var.rate	Employment variation rate (quarterly indicator)	Ratio
cons.price.idx	Consumer price index (monthly indicator)	Ratio
cons.conf.idx	Consumer confidence index (monthly indicator)	Ratio
euribor3m	Euribor 3-month rate (daily indicator)	Ratio
nr.employed	Number of employees (quarterly indicator)	Ratio
Final_Y	The subscription of the deposit	Norminal

Figure 3 Attribute description

The report records the process of the try and error, and finally provides the best approach to the prediction problem. Sklearn is the main framework used in this project.

Data cleaning and preprocessing

Data cleaning

The data cleaning consists of filling of the missing values, fixing of inconsistency, removing of outliers, and reduction of duplication.

Missing values

There are six attributes contain missing values which are ‘job’, ‘marital’, ‘education’, ‘default’, ‘housing’, and ‘loan’,

Since none of them are numeric type, they are filled by mode. For example, all the missing values in ‘job’ are replace by ‘blue collar’.

Inconsistency

According to the attribute description, a value of '999' in the ‘pdays’ means the client is new.

Thus, the corresponding ‘poutcome’ would be 'nonexistent'. However, some entries are

inconsistent with this rule. So, they are removed.

Outliers

Since most values in the data points are within the normal range, so it is reasonable to train the model without outliers. By doing so, the model should predict well on the normal data points. The range of outlier are identified by the boxplot.

For example, all the duration exceeds 500s are treated as outliers.

Duplication

Reduce the duplication helps improve the performance of the model.

Data preprocessing

One-hot encoding

Since nominal data cannot be used in many classifiers, all the nominal data are encoded by one-hot. This is done by the dummy() function in Pandas.

Normalisation

To reduce to the complexity of computation and make the weight of the value the same, z-score normalisation is performed. Z-score normalise the data points in the sphere zone which is better for K-nearest neighbour (KNN). KNN is the assigned classifier in this project.

Discretion

Too many kinds of values in attributes consume a lot of computing resource and may downgrade the performance of the model. For example, ‘age’ is discrete into ‘young’, ‘mid’, and ‘elder’.

Approach

At beginning, a feature ranking will be performed. With the selection of important features, the candidate classifier will be tested one by one. In this project, classifiers used are K-nearest neighbour, decision tree, random forest, and gradient boosting.

With each classifier, the feature selection will be executed first. This time, different combination and number of features are tested. After finding out the best group of features, parameters are tuned by grid search.

The performance of each classifier will be compared by F score because of the imbalanced of this dataset. Comparing with the customer who does not subscribe a deposit, there is fewer customers subscribed. Scoring by accuracy is affected by the accuracy of negative behaviour too much, while F score treated precision and recall in the same weight.

For the same reason, Precision-Recall curve is better than the ROC curve.

When the best classifier is chosen, more actions could be performed to improve the score of the model, such as

1) change the sampling method the training set to reduce the imbalance.

2) predict the missing value with different classifier.

3) adjust on binning strategy.

4) use a new classifier.

5) combine different classifiers.

Classification

Ranking of feature importance

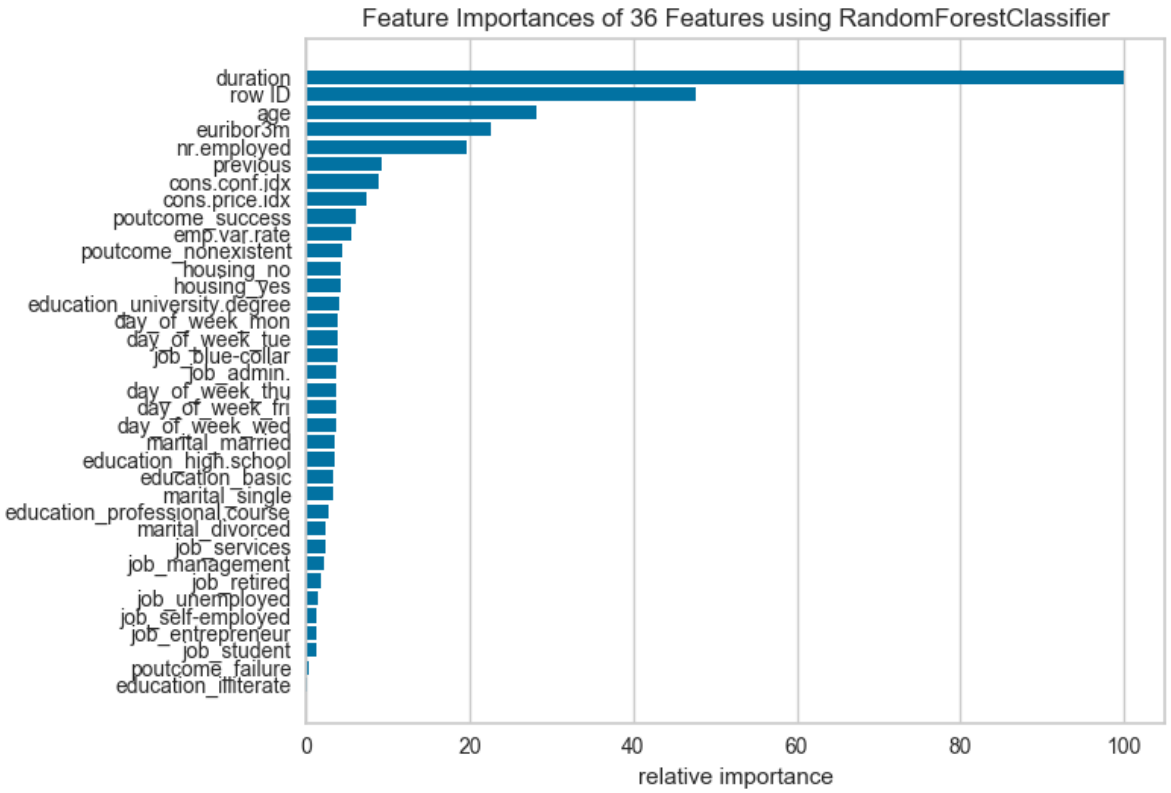

Figure 4 Ranking of feature importance

Classifier selection

K-nearest neighbour (assigned)

K-nearest neighbour (KNN) is a supervised classifier, and there is only one parameter to tune which is k. KNN does not support the feature selection in sklearn, so this step skipped. The parameter k is tested from 1 – 100, and 1 get the best F score.

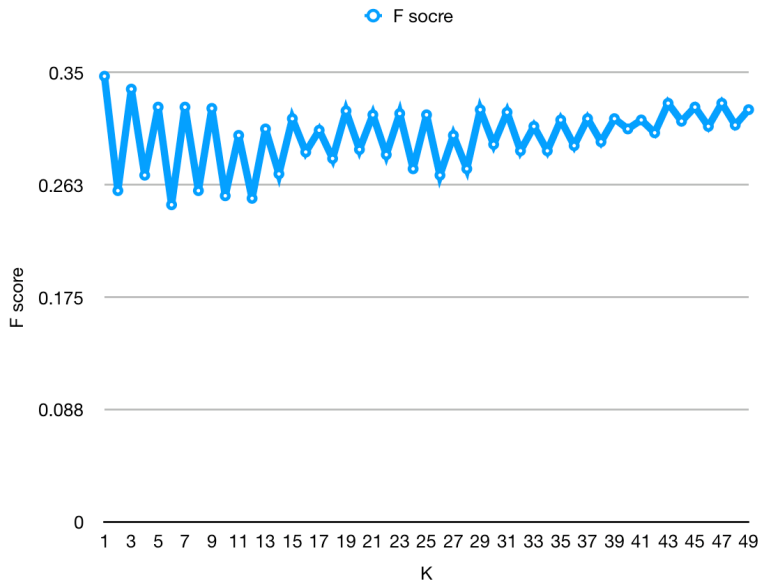

Figure 5 F score with different k

Following is the confusion matrix when k is 1.

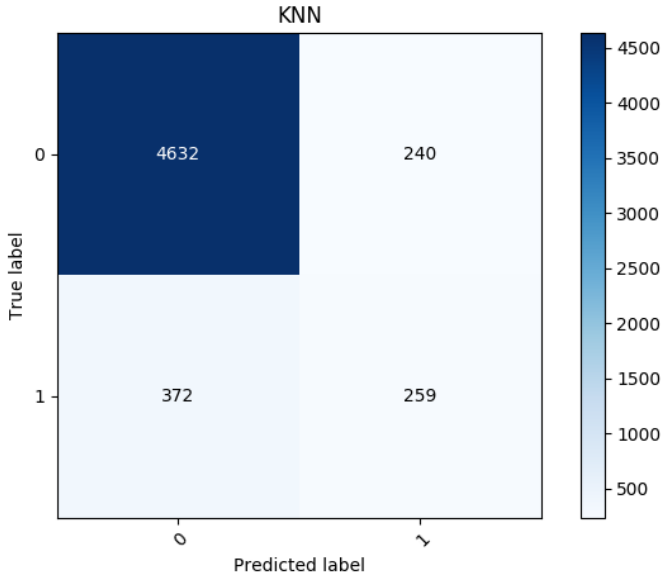

Figure 6 Confusion matrix of KNN

The F score is 0.458.

Decision tree

Feature selection is performed first. According to the graph, nine features is the best option.

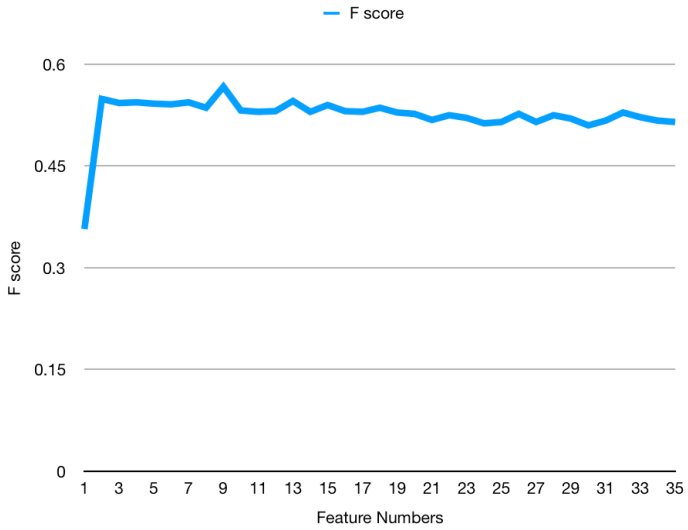

Figure 7 Different feature numbers in decision tree

The nine features are ‘age’, ‘duration’, ‘cons.conf.idx’, ‘euribor3m’, ‘marital_single’, ‘education_university.degree’, ‘housing_no’, and ‘poutcome_success’.

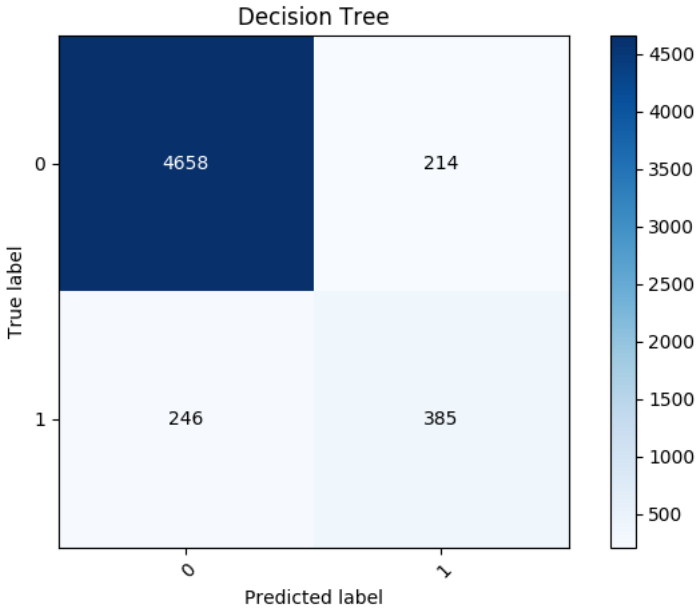

Figure 8 Confusion matrix of decision tree

The F score is 0.626.

Random forest

The final F score of random forest is 0.630. The confusion matrix is showed below.

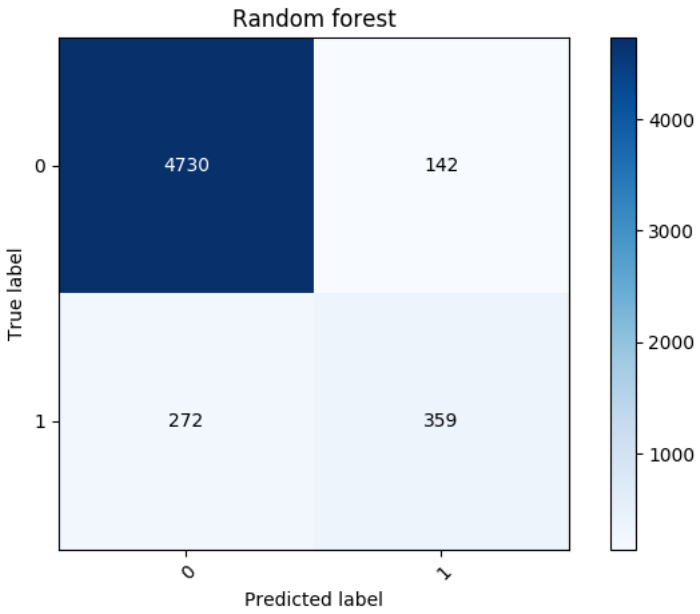

Figure 9 Confusion matrix of random forest

Gradient Boosting

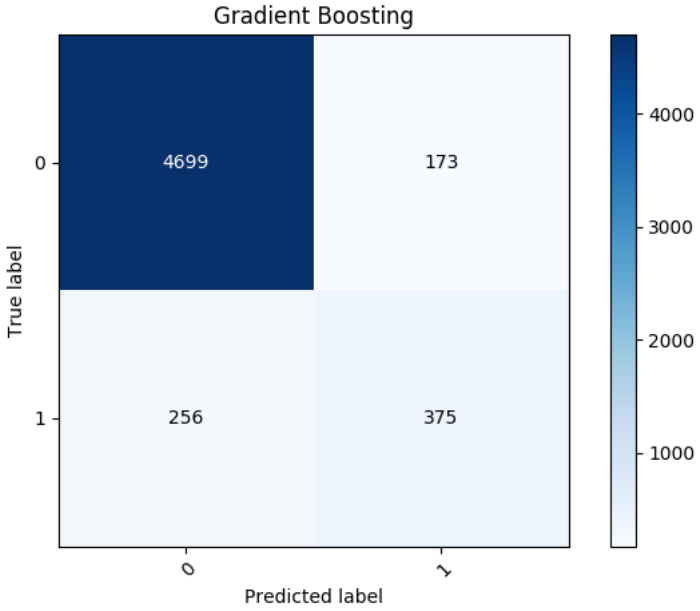

Figure 10 Confusion matrix of gradient boosting

The F score is 0.636.

Comparison

Classifier	F score
Gradient boosting	0.636
Decision tree	0.626
Random forest	0.630
KNN	0.458

Figure 11 Comparison of F score

Best classifier – Gradient Boosting

The features selected are:

- age
- duration
- cons.conf.idx
- euribor3m
- marital_single
- education_university.degree
- housing_no
- poutcome_success

The parameters settings are n_estimators=113, learning_rate=0.2, max_depth=3, subsample=1, criterion: 'friedman_mse'. The rest remains default setting.

The F score is 0.636, and corresponding accuracy is 92.20% (The accuracy on Kaggle is lower than this). The result is based on the split of the local training data set in which 80% is training data and 20% is test data.