Predicting the result of direct marketing campaigns

1. Introduction

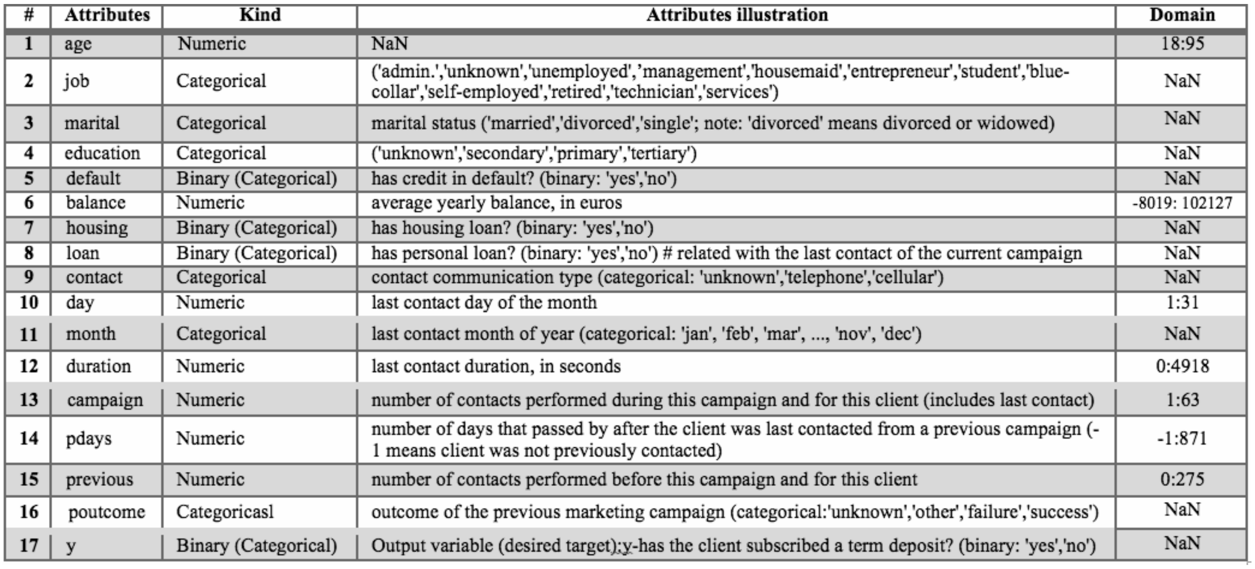
A Portuguese banking institution uses direct marketing campaigns to encourage subscription of term deposits. Direct marketing aims to target consumers who are supposedly to be keener to the product, hence receiving a lot more positive responses compared to mass campaign. However, it comes with certain drawbacks. Large quantity of contacts can be rather cost consuming and unnecessary ones may trigger negative attitude towards the company as consumers view it as an intrusion of privacy. It is thus crucial for the company to determine the optimum balance of such campaigns. Lesser contacts should be done, but an approximately number of successes should be kept. This project aims to explore various techniques in Data Mining and to sort out the best classification model that the company can use to explain the success of a contact and as a result effectively narrows down its marketing campaign only to clients who are likely to subscribe a term deposit.

We have collected the following information with regard to the bank's marketing campaigns. The telephone, with a human agent as the interlocutor, was the dominant marketing channel, although sometimes with an auxiliary use of the Internet online banking channel. Each campaign was managed in an integrated fashion and the results for all channels were outputted together. During these phone campaigns, an attractive long-term deposit application, with good interest rates, was offered.

2. Data collection & interpretation

The dataset collected is related to 17 campaigns that occurred between May 2008 and November 2010, corresponding to a total of 45211 contacts. For each contact, a large number of attributes along with the result (whether it was a success) were noted down. Within the whole database, there were a total of 5289 successes (11% success rate).

The attributes that were noted down are given in the table below.



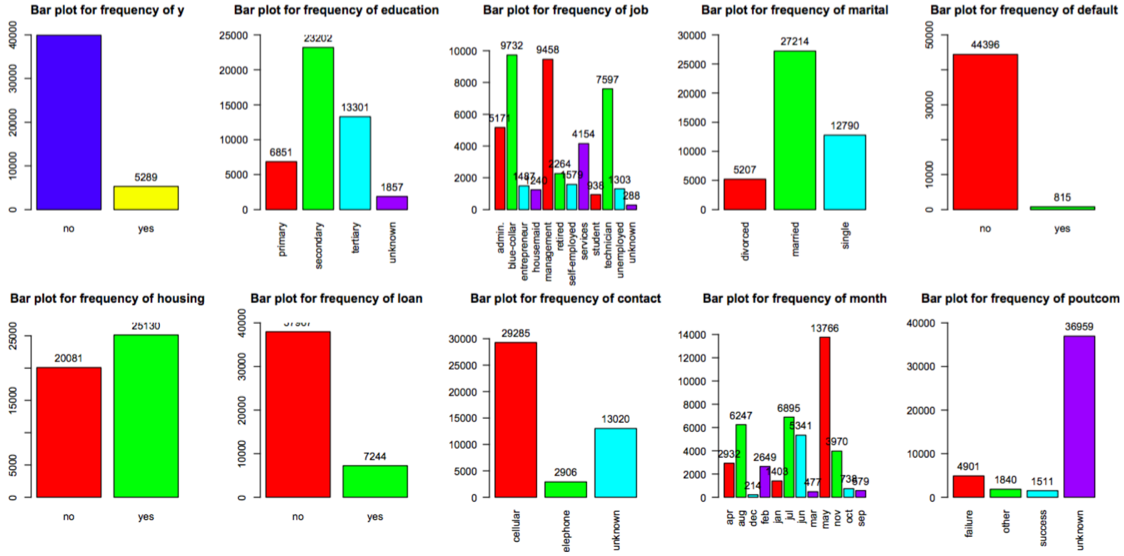
#	Attributes	Kind	Attributes illustration	Domain
1	age	Numeric	NaN	18:95
2	job	Categorical	('admin.','unknown','unemployed','management','housemaid','entrepreneur','student','blue-collar','self-employed','retired','technician','services')	NaN
3	marital	Categorical	marital status ('married','divorced','single'; note: 'divorced' means divorced or widowed)	NaN
4	education	Categorical	('unknown','secondary','primary','tertiary')	NaN
5	default	Binary (Categorical)	has credit in default? (binary: 'yes','no')	NaN
6	balance	Numeric	average yearly balance, in euros	-8019: 102127
7	housing	Binary (Categorical)	has housing loan? (binary: 'yes','no')	NaN
8	loan	Binary (Categorical)	has personal loan? (binary: 'yes','no') # related with the last contact of the current campaign	NaN
9	contact	Categorical	contact communication type (categorical: 'unknown','telephone','cellular')	NaN
10	day	Numeric	last contact day of the month	1:31
11	month	Categorical	last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')	NaN
12	duration	Numeric	last contact duration, in seconds	0:4918
13	campaign	Numeric	number of contacts performed during this campaign and for this client (includes last contact)	1:63
14	pdays	Numeric	number of days that passed by after the client was last contacted from a previous campaign (-1 means client was not previously contacted)	-1:871
15	previous	Numeric	number of contacts performed before this campaign and for this client	0:275
16	poutcome	Categoricasl	outcome of the previous marketing campaign (categorical:'unknown','other','failure','success')	NaN
17	y	Binary (Categorical)	Output variable (desired target):y-has the client subscribed a term deposit? (binary: 'yes','no')	NaN

3. Dataset Pre-processing

3.1. Data exploration

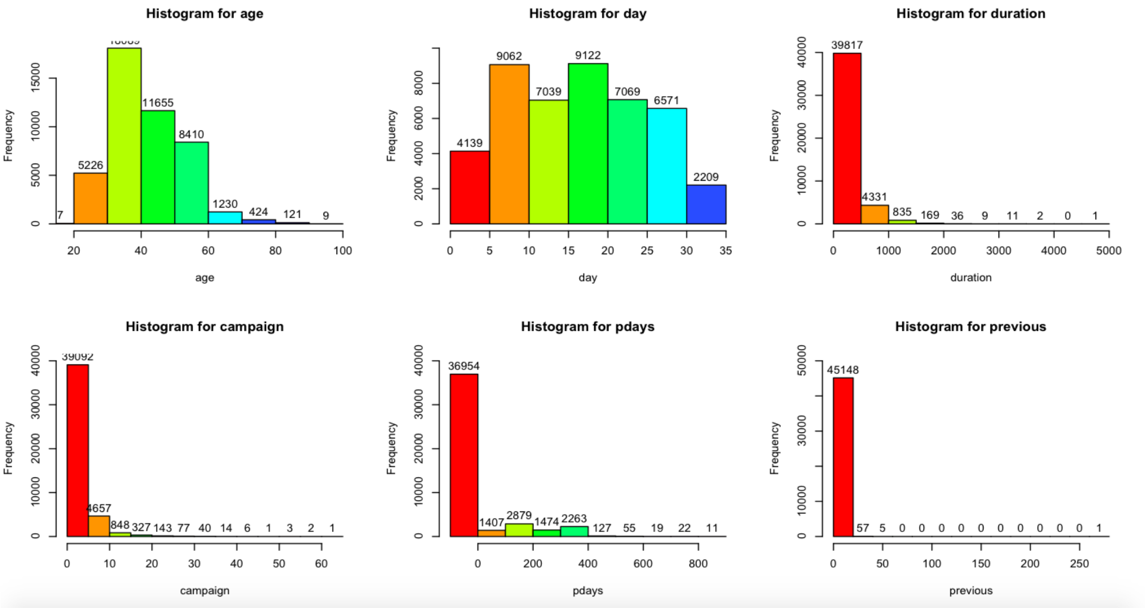
For categorical attributes:

Barplots were plotted to see the attributes distribution across different categorical values.



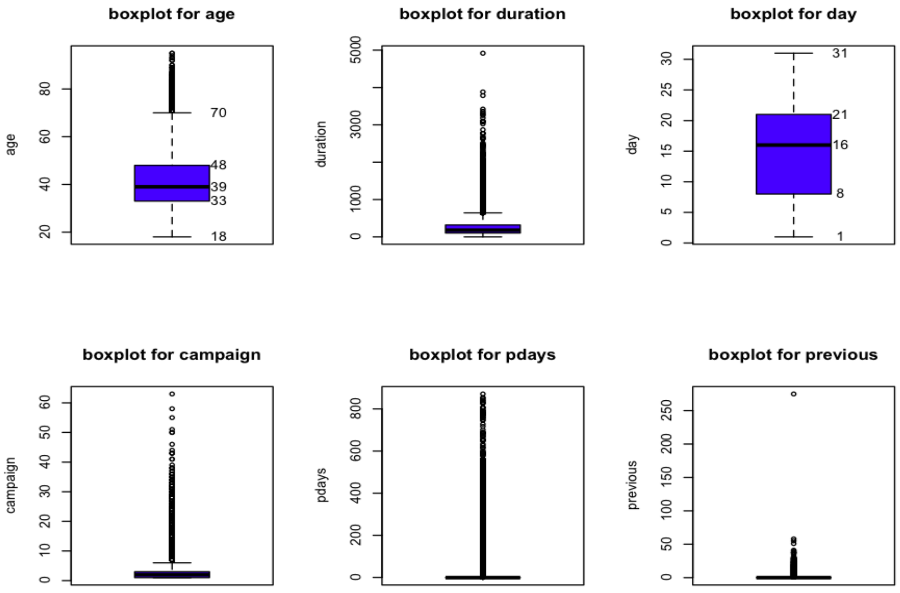
For numerical attribute:

1)Histograms were plotted to see the attributes distribution across different values.



From the histogram, we notice "duration", "campaign","pdays" and "previous" have different scales and large range, which requires us to do attribute normalization.

2) Plot the boxplot for numerical attributes to display the distribution of data based on five summary("minimum", "first quartile", "median", "third quartile", and "maximum"). It can also tell us the outliers and what their values are.



From the boxplot, some numerical attributes have a lot of large outliers. We will remove these extreme outliers in our original dataset. (Remove objects where “duration”>3700, or “campaign”>48 or “previous”>100)

3.2. Data categorical conversion

The categorical attributes (education, job, marital, default, housing, loan, contact, month, poutcome) need to be converted to numerical variables or dummy variables since the logistic regression model and neural network model require numerical inputs.

job	admin	blue-collar	entrepreneur	housemaid	management	retired	self-employed	services	student	technician	unemployed	unknown
Convert to numeric	1	2	3	4	5	6	7	8	9	10	11	12

marital	divorced	married	single
Convert to numeric	1	2	3

default	no	yes
Convert to numeric	1	2

housing	no	yes
Convert to numeric	1	2

loan	no	yes
Convert to numeric	1	2

contact	cellular	telephone	unknown
Convert to numeric	1	2	3

month	apr	aug	dec	feb	jan	jul	jun	mar	may	nov	oct	sep
Convert to numeric	1	2	3	4	5	6	7	8	9	10	11	12

outcome	failure	other	success	unknown
Convert to numeric	1	2	3	4

3.3 Data Normalization

Normalization is generally required when we are dealing with attributes on a different scale, Differences in the scales across input variables may increase the difficulty of the problem being modeled. So, we performed Min-Max normalization to make all attributes varying from 0 to 1.

3.4. Data Cleaning & Feature selection

(1). Correlation coefficient

Data can contain attributes that are highly correlated with each other. Many methods perform better if highly correlated attributes are removed. We take a look at the correlation coefficient of all features (categorical features are converted to numerics).

Generally, we want to remove attributes with an absolute correlation of 0.75 or higher. Our correlation matrix shows correlation between “pdays” and “poutcome” is -0.8584.

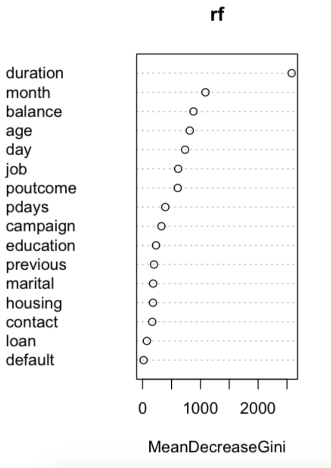
As a result, we remove the attribute “poutcome”, since “poutcome” contain 36959 “unknown” value.

(2). Unknown Values

Refer to the barplots above, we notice that four attributes contain “unknown” values, which are in fact missing values. The straightforward way of dealing with these “unknown” values is to remove all corresponding objects . We have already removed “Poutcome” completely as it is highly correlated with “P-days”. However, out of the remaining attributes, “contact” contains a large amount of “unknown” values. Removing such a huge number of objects largely reduces the amount of data (other attributes’ information) that we can work with. So we decided to only remove the “unknown” values for attributes “job” and “education”. For attribute “contact”, we treat all “unknown” values as normal values.

(3) Rank feature by importance

We create a plot of importance score by random forest to estimate feature importance. These scores which are denoted as ‘Mean Decrease Gini’ by the importance measure represents how much each feature contributes to the homogeneity in the data.



From the importance score plot, we can easily see that “duration”, “month”, “balance”, “age”, “day”, “job”, “poutcome”, “pdays”, “campaign” and “education” are the 10 most important features. In our model building later, we will use these 10 important features to build the random forest tree, and compare the performance with the model using all features.

3.5 Imbalanced learning

From the barplot of attribute y, we notice class “no” occurs 39922 times, which is more often than class “yes” occuring 5289 times.

One solution to the class imbalance problem is to re-sample at data level. We performed both under-sampling and oversampling on the training set. The minority class is oversampled with replacement and majority class is undersampled without replacement.

In the model building part, we will train the models with both the original training dataset and the training dataset after performing imbalanced solution. And then compare the performance on the testing dataset to see if the result using balanced dataset becomes better.

3.6 5-folder cross validation

we shuffle the original dataset and divide samples into 5 roughly equal disjoint parts. Then each time, we take one folder as testing data and take remaining 4 folders as training data. Evaluation scores, such as accuracy, F-measure, MCC and AUC are averaged.

4. Model Building & Performance Evaluation

4.1. Decision tree

Why use decision tree:

1) Requires less effort for data preparation during pre-processing. Leave us enough time to do other modeling.
2) Very intuitive and easy to explain technical teams as well as stakeholders.
3) Does not need require normalization of data.

Evaluate performance:

	Accuracy	F-measure	MCC
Original training set	0.9012	0.5340	0.4823
Balanced training set	0.8315	0.5373	0.4988

4.2. Random forest tree

Why use random forest tree

1) Can handle binary features, categorical features, and numerical features. Also, very little pre-processing needs to be done.
2) Has methods for balancing error in class population unbalanced datasets. Since we have an unbalanced data set, the larger class will get a low error rate while the smaller class will have a larger error rate.
3) Each decision tree has a high variance, but low bias. When we average all trees in random forest, we are averaging the variance as well so that we have a low bias and moderate variance model.

Evaluate performance :

1). Using all features

	Accuracy	F-measure	MCC
Original training set	0.9053	0.5246	0.4819
Balanced training set	0.8825	0.6134	0.5706

2). After selecting 9 most important features according to importance scores

	Accuracy	F-measure	MCC
Balanced training set	0.8760	0.5635	0.5053

4.3. Logistic regression

Why use logistic regression

1) Does not require too many computational resources, highly interpretable, does not require input features to be scaled.
2) Easy to implement and very efficient to train.

Evaluate performance

1) Use all features

	Accuracy	F-measure	MCC	AUC
Original training set	0.8919	0.3183	0.3148	0.8713
Balanced training set	0.8100	0.4914	0.4416	0.8753

2) Perform normalization using all features

	Accuracy	F-measure	MCC	AUC
Balanced training set	0.7939	0.4788	0.4325	0.8747

3). After checking the p-value for each attribute, we delete those whose p-value is not significantly small ('job', 'day' and 'default'):

	Accuracy	F-measure	MCC	AUC
Balanced training set	0.8094	0.4907	0.4410	0.8750

ROC curve:

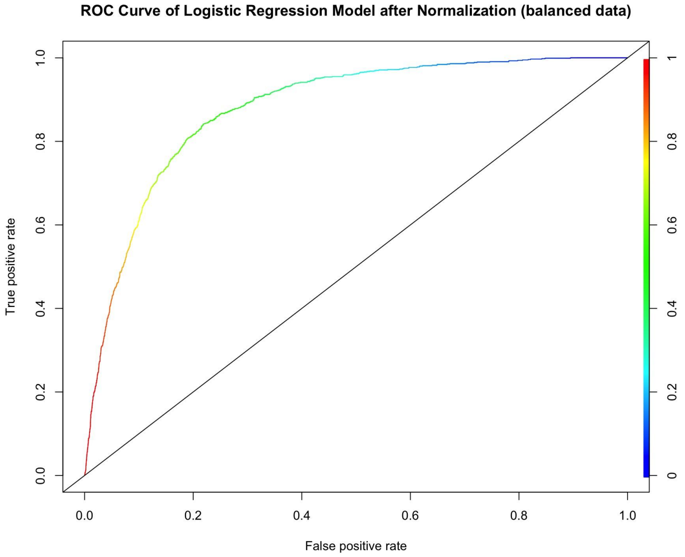


Figure:ROC curve of Logistics Regression using all features after normalization (0.8747)

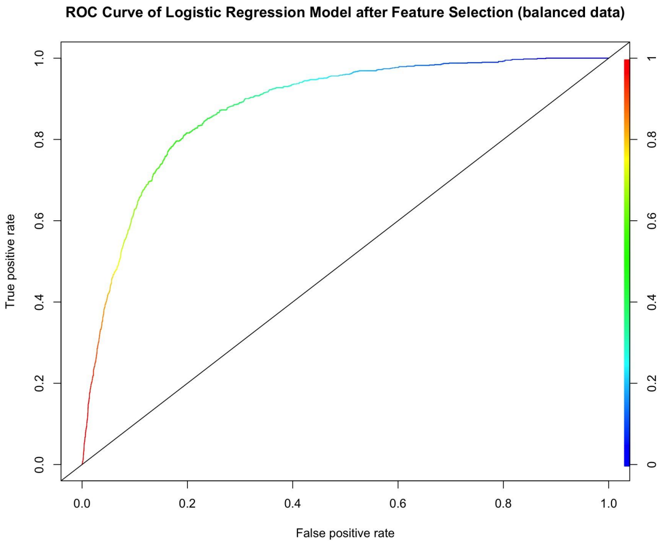


Figure: ROC curve of Logistics Regression after feature selection (0.8750)

4.4. Naive Bayes classifier

Why use Naive Bayes classifier

1) Can be used for both binary and multi-class classification problems.
2) Handles continuous and discrete data
3) Not sensitive to irrelevant features
4) Dealing with missing data easier than other algorithms.

Evaluate performance

	Accuracy	F-measure	MCC	AUC
Original training set	0.8719	0.4553	0.3829	0.8511547
Balanced training set	0.7997	0.4652	0.407	0.8479258

ROC curve

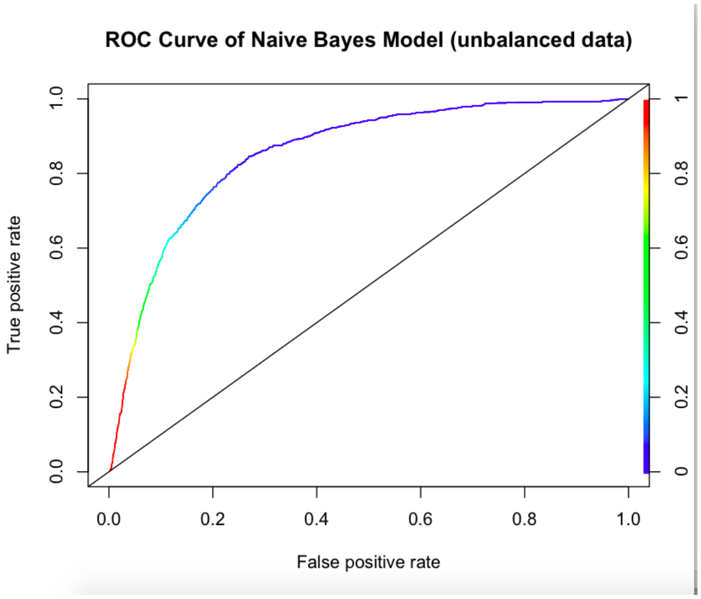


Figure: ROC curve of Naive Bayes for unbalanced data (AUC=0.8511547)

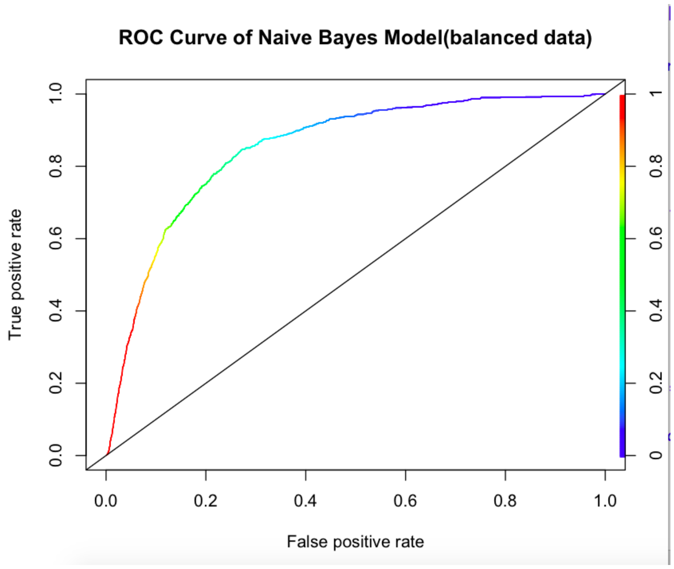


Figure: ROC curve of Naive Bayes for balanced data (0.8479258)

4.5. Artificial Neural Network

Why use Artificial Neural Network

1) Has the ability to learn and model non-linear and complex relationships.

2) Unlike many other prediction techniques, ANN does not impose any restrictions on the input variables.

Evaluate performance

We try out one layer Neural Network with 1 to 5 neurons.

Number of neurons	Accuracy	F-measure	MCC
1	0.2240409	0.04571718	-0.4242
2	0.0480722	0.006716748	-0.08608
5	0.04575663	0.007290612	-0.08746

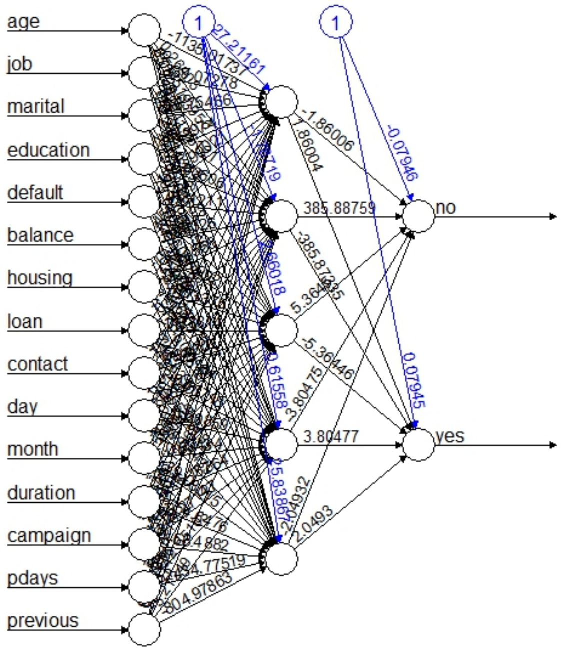


Figure: One layer Neural Network with 5 neurons

5. Conclusion

In each model, the F-measure and MCC become slightly better after balancing the training dataset.
However, while the recall for "yes" class becomes much better in the balanced dataset, the recall for "no" class becomes a little bit worse. In marketing campaign, the cost of calling a non-respondent is very small, but the cost of not calling someone who would respond is the entire profit lost. It is important to improve the recall rate for "yes" class by balancing the dataset.

We thus conclude that MCC did not improve much.

The random forest tree performs best among these five models. It is recommended for the company to adopt this model to forecast the likelihood of success in its future campaigns

6. Further improvement
    - The “pday” attribute, namely the number of days that passed by after the client was last contacted from a previous campaign is categorized as -1 when the client was never contacted before. This is not an ideal representation as the -1 does not really have the same numerical meaning as other values. This will affect the data mining result, especially when the model is sensitive to numerical values. (For example, Logistic regression). A more appropriate way of reflecting the actual situation is needed to accurately analyse the dataset.
    - The MCC score & F-measure have become worse after feature selection, which seem to be the opposite of our expectations. Further research will be needed to explore why that is the case.
    - For the artificial Neural Network Model. We have only used three neurons due to limited computing power. To further enhance the accuracy of the model, we need to add more neurons and layers.