**Final Report for Capstone 1 project**

**Context:**

Commercial banks spend considerable time and energy in marketing various products/offerings to their clients. In terms of their marketing ROI (return on investment), it makes sense to only focus their marketing efforts on those clients who they believe will be most responsive to a particular product offering(s) vs the whole universe of banking clients.

Business stakeholders that would benefit most from this project would be marketing managers at commercial banks who want to be efficient in their use of time and energy. Using results from this project, they would be able to classify clients expected to be most responsive to a particular product/offering and hence focus their effort primarily or only on those clients vs whole universe of banking clients they could have potentially targeted.

**Specific problem being solved:**

Specific problem I will be addressing for commercial banks is that of increasing subscription to term deposit accounts among bank’s clients who have checking accounts but don’t have term deposit accounts. Increasing subscription to term deposit accounts among clients is important for commercial banks because it ensures availability of funds for a defined term that the commercial banks can then lend out to corporations/individuals for profit.

As a result of this project, marketing managers at commercial banks would be able to identify those clients (that already have current accounts but don’t have term deposit accounts) that are expected to be most responsive to an offer of term deposit account. Marketing managers would then focus their efforts only on those clients vs all the clients, resulting in high ROI on their marketing efforts.

**Methodology used:**

Following steps were performed for this project:

1. Dataset was obtained from Kaggle
2. Data was wrangled as necessary.
3. Dataset was explored to get better understanding of the data and relevant business. Dataset was further processed to prepare it for training.
4. Following binary classification models were chosen to experiment with:
   1. Logistic regression
   2. Random forest
   3. Gradient boost
   4. SVM (Support vector machine)

Each of the model above was trained on the training dataset and then its performance (i.e. its quality of prediction) evaluated on the test dataset using following metrics:

1. Accuracy
2. Confusion matrix
3. Classification report
4. Area under ROC and Precision Recall curves
5. Finally, models were evaluated against each other under above mentioned metrics and the best model(s) for the purposes of this project was recommended.

Following sections describe the steps that were performed and results obtained for this project.

**Data collection:**

Data source for this project was Kaggle:

<https://www.kaggle.com/prakharrathi25/banking-dataset-marketing-targets>

**Data wrangling**

Dataset had the following variables:

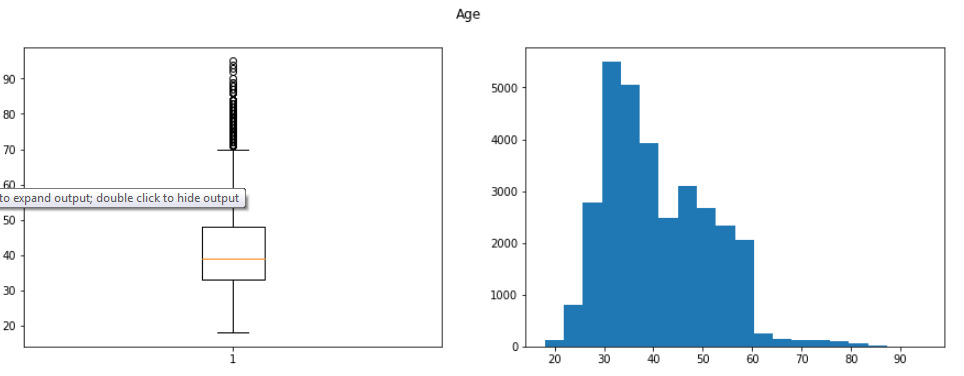
|  |  |  |
| --- | --- | --- |
| Variable name | Variable description | Variable type |
| ID | Unique client ID | Numerical |
| age | Age of the client | Numerical |
| job | Type of job | Categorical |
| marital | Marital status of client | Categorical |
| education | Education level | Categorical |
| default | Credit in default | Categorical |
| balance | Outstanding balance | Categorical |
| housing | Housing loan | Categorical |
| loan | Personal loan | Categorical |
| contact | Type of communication | Categorical |
| day | Day of month of contact | Numerical |
| month | Contact month | Numerical |
| duration | Contact duration | Numerical |
| campaign | Number of contacts performed during this campaign to the client | Numerical |
| pdays | Number of days that passed by after the client was last contacted | Numerical |
| previous | Number of contacts performed before this campaign | Numerical |
| poutcome | Has the client subscribed to term deposit in previous campaign? | Categorical |
| subscribed | Has the client subscribed to term deposit in this campaign? | Categorical |

The dataset had 18 variables/columns (8 of them numerical, 10 of them categorical) and about 31,600 rows in total. The dependent variable is “subscribed” which is of categorical type. Rest are independent variables which influence/impact the dependent variable in some way.

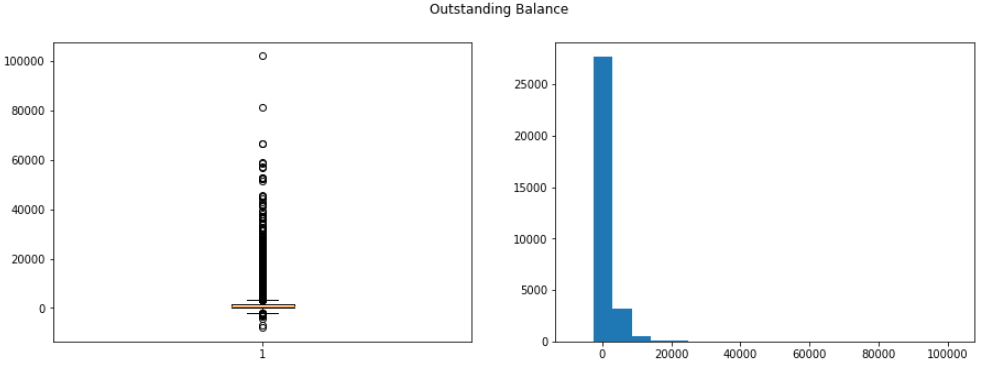
Since the dataset was obtained from Kaggle, it was relatively clean i.e. there were no missing or null values in any of the columns and hence there was not much to do in terms of treating missing values or null values.

Looking at the box plots and histograms for numerical columns gave no indication as to any variable with seemingly incorrect/wrong values which could push the mean values to one extreme or the other:

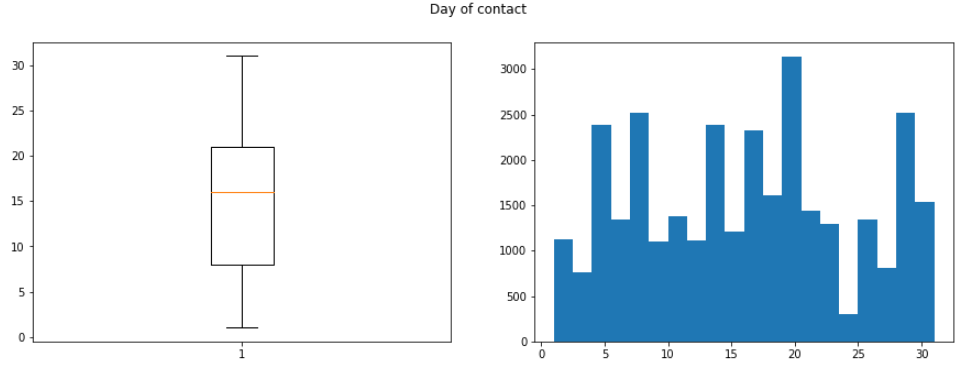
1. age (i.e. age of client) varies from 18 years to 95 years which is reasonable since clients can live upto 95years.



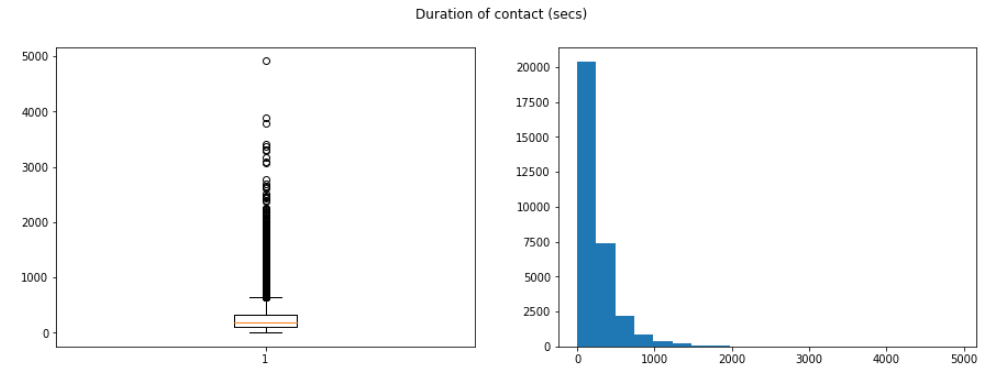
1. balance (i.e. outstanding balance) varies from -8019 to 102127 which is reasonable. It is possible for some clients to have negative balance



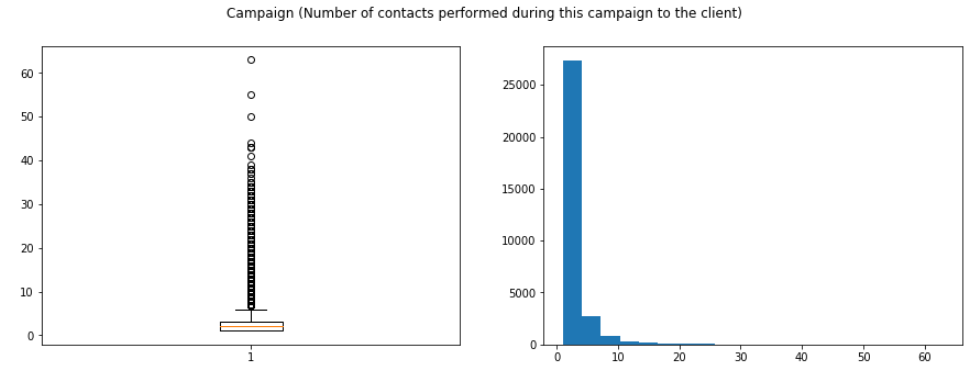
1. day (i.e. day of month of contact) varies from 1 to 31 which is reasonable since maximum number of days in a month is 31



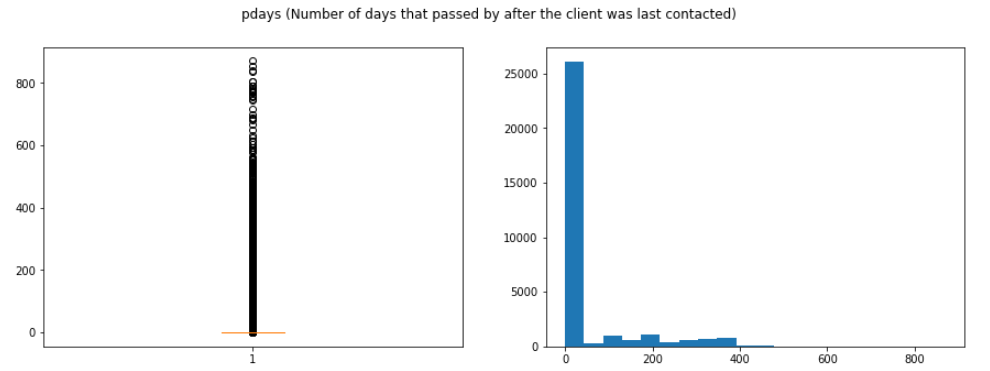
1. duration (i.e. contact duration) varies from 0 to 4918 secs which is reasonable since 4918/3600 secs is 1.36hrs. One to two hours is reasonable timeframe for contact duration.



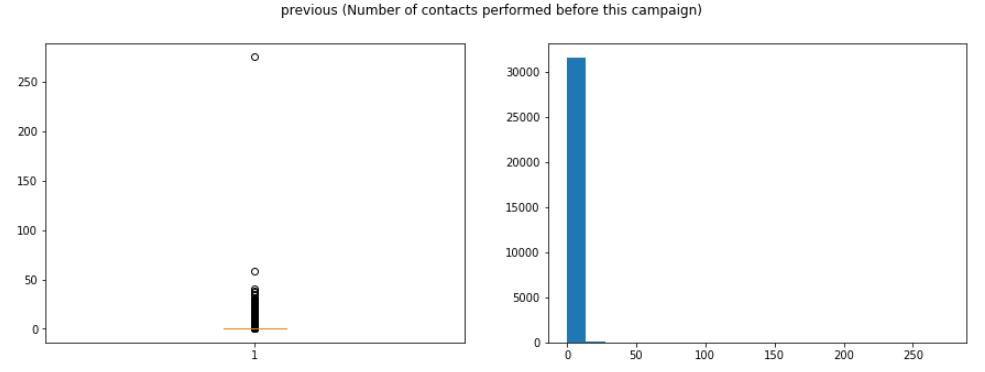
1. campaign( i.e. number of contacts performed during this campaign to the client) varies from 1 to 63. 63 seems to lie on the continum of the values. So, there was no need to treat it as an incorrect/wrong value.



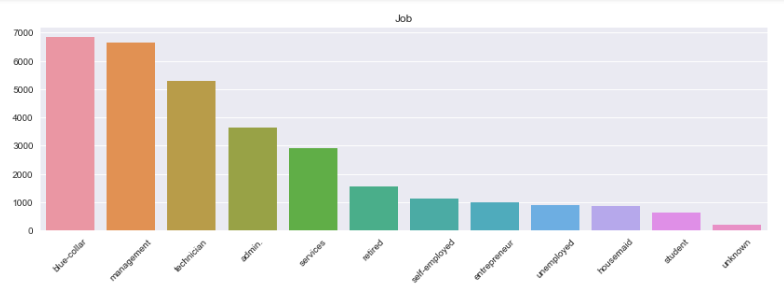
1. pdays (i.e. number of days that passed by after the client was last contacted) varies from -1 to 871 days. 871/364 is about 2.39 years which is possible since banks may not be in touch with the clients for few years.

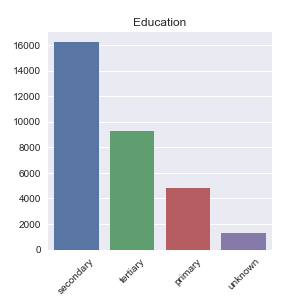
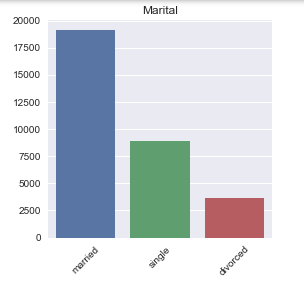


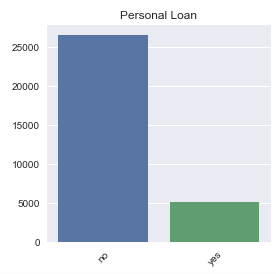
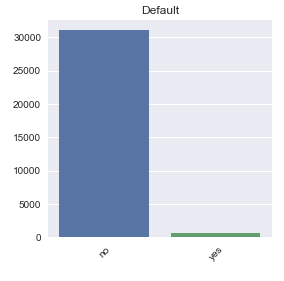
1. previous (i.e. number of contacts performed before this campaign) varies from 0 to 275. 275 lies way away from other values. So it can potentially be considered an outlier. However, for the purpose of this project, it was left as it is since it is possible for a bank client to have been contacted that many times in previous campaigns.

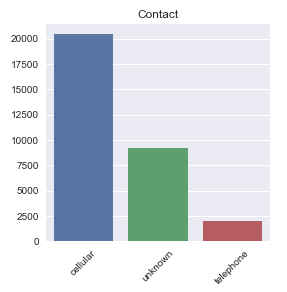
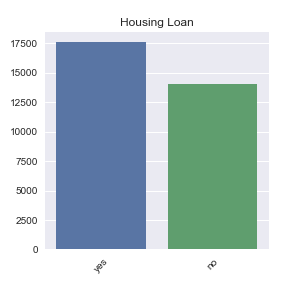


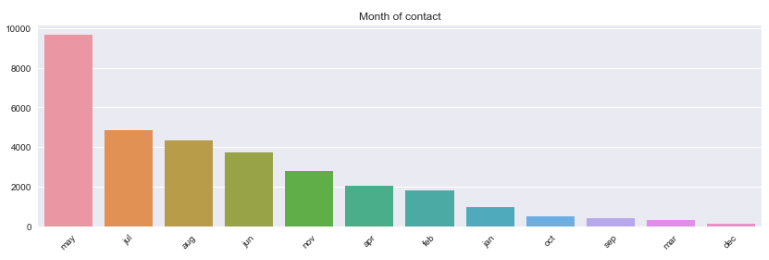
Based on results above, there was not much data wrangling done with numerical variables.

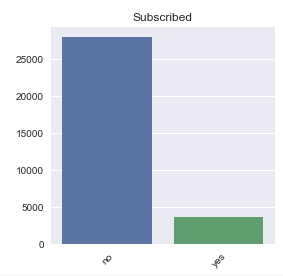
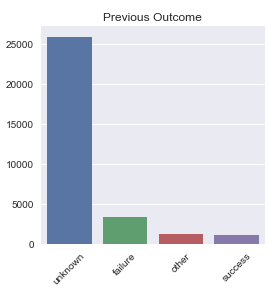
Looking at barplots for categorical columns:











The categories for all the categorical columns seemed well defined. Only thing that stood out was that the “subscribed” dependent variable has more no’s than yes’s i.e. an imbalanced variable.

Overall, since the dataset was relatively clean, there wasn’t much data wrangling to perform either on numerical columns or categorical columns.

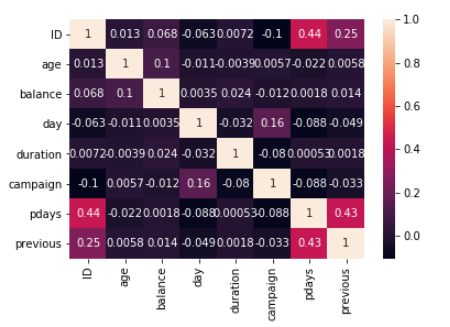
**Data exploration, Data processing and Statistical Data analysis:**

Distribution of numerical variables

As can be seen from histograms of numerical columns above, most of them are not evenly distributed. Except for "age" and "day", they are skewed i.e. distributed at one end of the x-axis or the other. Typically for skewed distributions, log transformations tend to be done to make them more normal-like before feeding them to linear models. In this case, I leave them as they are since I will be feeding them primarily into non-linear models where there is no necessity for normal looking/like distributions for numerical variables.

Correlation of numerical varibles

Pairwise correlation between numerical variables was explored via pairwise correlation plot and heatmap (shown below) to see if there was any meaningful correlation amongst them. Since there was no meaningful correlation, all the numerical columns were fed to the models.

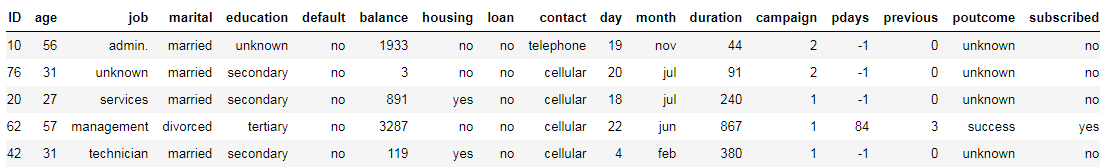


Data processing and independent variables impacting dependent variable the most

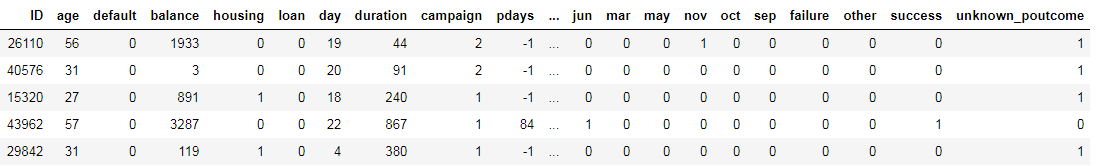
In order to identify independent variables impacting the depending variable “subscribed” the most, logistic regression can be performed to identify independent variables with the largest coefficients. However, before doing logistic regression, data would need to processed in order to make it ready for logistic regression.

Following data processing steps were performed:

1. Leave numerical variables as they are
2. Do onehot encoding for categorical variables with more than 2 labels/classes – this step increases the total number of columns from 18 to 50 for the dataset since for example if a categorical variable has 4 labels/classes, then this step produces 4 derived independent variables for it.
3. Initial dataset would look something like this:



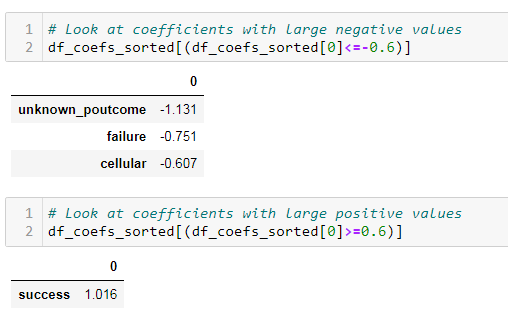
1. Dataset after onehot encoding would look something like this:



1. Create a user defined replace map for categorical variables with exactly 2 labels/classes
2. Divide the dataset into independent variables i.e. x and dependent variables i.e. y and then each of them into training set and test set.

Training dataset would be used to train the logistic regression model. During training phase, coefficients for model that best fit the training dataset would be learned.

Running the logistic regression resulted in following 4 derived independent variables having the most impact on the dependent variable "subscribed":



Unknown\_poutcome, success and failure are derived from independent variable “poutcome” and are essentially its classes/labels. Similarly, cellular is derived independent variable “contact” and is essentially its class/label. Based on coefficient values:

1. if “poutcome” was a "success", then there is a strong likelihood that clients will subscribe to term deposit in this campaign.
2. If poutcome was "unknown\_poutcome" or "failure", then there is a strong likelihood that clients will NOT subscribe to term deposit in this campaign.
3. Similarly, if “contact” was "cellular", then there is a moderate likelihood that clients will NOT subscribe to term deposit in this campaign.

Statistical data analysis for significant independent variables

Running logistic regression with statsmodels provides p-values for the derived independent variables:

Derived Independent variable p-value i.e P>|z|:

1. unknown\_poutcome 0.00
2. success 0.00
3. failure 0.00
4. cellular 0.00

Since P>|z| < 0.05 at 95% confidence level for above 4 derived independent variables, this implies that the null hypothesis (i.e. there is no statistically significant association between them and the dependent variable) can be rejected which implies they do have statistically significant association with the dependent variable "subscribed".

**Model training and Model evaluation metrics**

Once the dataset has been divided into training dataset and test dataset, training dataset is used to train the model from which the model learns the parameters that best fit the training dataset. The model that has learned these parameters is then fed the test dataset (which has all the independent variables but not the dependent variable “subscribed”) from which the trained model makes predictions. This prediction is compared against the actual value for “subscribed” dependent variable on the test dataset to see how good the trained model was in making predictions.

Four different models were explored:

1. Logistic regression model
2. Random forest model
3. Gradient boosting model
4. SVM (support vector machine)

For each of the model:

1. Training dataset was used to train the model (default hyper parameters were chosen for the model)
2. The trained model was used to make predictions regarding whether a given bank client would subscribe to term deposit or not using test dataset
3. The prediction quality/performance of the model was evaluated using 4 metrics
   1. Accuracy score
   2. Confusion matrix
   3. Classification report
   4. Area under curve for ROC (receiver operating characteristic) and Precision Recall curves

Because of severe performance issues with SVM model (training step was taking extremely long on personal laptop), this model was dropped for the purpose of this project.

**Model comparison under different metrics**

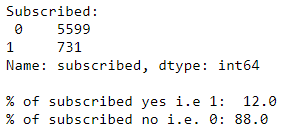
1. Accuracy score:
   1. All 3 models seem to have good accuracy scores :

Logistic regression = 0.897

Random Forest = 0.897

Gradient Boost = 0.909

* 1. However, this is misleading since the dependent variable “subscribed” is an imbalanced variable as can be seen below in the test data set:



High accuracy score in this case would mean that the model is good at predicting 0’s i.e. no’s since the dataset has 88% 0’s for “subscribed” dependent variable. High accuracy score here would not necessarily mean that the model is good at predicting 1’s i.e. yes’s since there is only about 12% 1’s. So, accuracy score in this case would not be a good metric to use since the dataset is imbalanced.

1. Confusion matrix:

The confusion matrix can be interpreted using the following table:

|  |  |  |
| --- | --- | --- |
| N = 6330 | Predicted for “subscribed”: “no”/0 | Predicted for “subscribed” = “yes”/1 |
| Actual for “subscribed” = “no”/0 | TN (True negative) | FP (False positive) |
| Actual for “subscribed” = “yes”/1 | FN (False negative) | TP (True positive) |

Comparing confusion matrix for 3 models:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N=6330 | TN (True negative) | TP (True positive) | FP (False positive) | FN (False negative) |
| Logistic regression | 5467 | 212 | 132 | 519 |
| Random forest | 5415 | 337 | 184 | 394 |
| Gradient Boost | 5381 | 378 | 218 | 353 |

We prefer models with small FP’s and FN’s and large TP’s and TN’s where possible since this means higher prediction quality of the models. From table above, we can see that, for Random Forest and Gradient Boost, FN’s (False negative) are reduced from 519 to 394 and 353 respectively at the cost of slightly increased FP’s (False positive) and TP’s (True positive) are increased from 212 to 337 and 378 respectively at the cost of slightly decreased TN’s (True negative) compared to logistic regression. What this essentially means is that both Random Forest and Gradient Boost are better binary classifiers than Logistic regression for this project when evaluated from perspective of confusion matrix.

1. Classification report

Precision, recall and f1-score can be interpreted using following equations:

1. Precision = TP/[Predicted “yes” or 1] = TP/(TP + FP) = Positive predictive value
2. Recall = TP/[Actual “yes” or 1] = TP/(FN + TP) = True positive rate
3. F1-score = 2 / [1/Precision + 1/Recall] = (2\*Precision\*Recall) / (Precision + Recall)

Comparing classification report for 3 models:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Label/Class of “subscribed” | Precision | Recall | F1-score |
|  |  |  |  |  |
| Logistic regression | 0 | 0.91 | 0.98 | 0.94 |
| Random forest | 0 | 0.93 | 0.97 | 0.95 |
| Gradient Boost | 0 | 0.94 | 0.96 | 0.95 |
|  |  |  |  |  |
| Logistic regression | 1 | 0.62 | 0.29 | 0.39 |
| Random forest | 1 | 0.65 | 0.46 | 0.54 |
| Gradient Boost | 1 | 0.63 | 0.52 | 0.57 |

From table above, we can see that:

1. Precision values are pretty similar for all 3 models when class of “subscribed” is 0 or 1
2. Recall values are also similar for all 3 models when class of “subscribed” is 0.
3. However, when class of “subscribed” is 1, Recall values of Random forest (0.46) and Gradient Boost (0.52) are significantly higher than that of Logistic regression (0.29). This happens because Random Forest and Gradient boost are able to detect more of the actual 1’s compared to Logistic regression, resulting in smaller value of FN which has the effect of increasing Recall value.
4. As a result of c), F1-score is higher for Random Forest (0.54) and Gradient boost (0.57) compared to that of Logistic regression (0.39) when class of “subscribed” is 1. Higher the F-1 score, higher the prediction quality of the model. So, essentially, Random forest and Gradient boost are better binary classifiers than Logistic regression when evaluated from perspective of classification report as well.
5. AUC (area under curve) for ROC and Precision Recall curves:

|  |  |  |
| --- | --- | --- |
|  | AUC for ROC curve | AUC for Precision Recall curve |
| Logistic regression | 0.888 | 0.503 |
| Random forest | 0.943 | 0.646 |
| Gradient Boost | 0.941 | 0.645 |

AUC for ROC curve provides area under TPR (true positive rate) vs FPR (false positive rate) curve for different thresholds (threshold for classifying what is a 1 vs 0). Similarly, AUC for Precision Recall curve provides area under precision vs recall curve for different thresholds. Both of them can be used for evaluating binary classifiers. Higher the score, better the predictive capability of the model. In the table above, we can see that:

1. AUC ROC for Random Forest (0.943) and Gradient Boost (0.941) are higher than that for Logistic regression (0.888).
2. AUC PR for Random Forest (0.646) and Gradient Boost (0.645) are higher than that for Logistic regression (0.503) as well.
3. So, both the metrics indicate that Random Forest and Gradient Boost are better binary classifiers than Logistic regression. However, in this case, since this is an imbalanced dataset with lot fewer positive classes than negative classes, AUC PR would be a better metric than AUC ROC. This is evident from higher magnitude change in AUC PR when going from Logistic regression to Random forest/Gradient Boost compared to magnitude change in AUC ROC.

**Recommendation:**

Based on the results above, either Random Forest or Gradient Boost model would provide the best performance on the test dataset given what was learnt from the training dataset. Among the models tried, either of the two models would help to identify most accurately the bank clients with current accounts that would be most open/likely to subscribe to an offer of term deposit from marketing team at the bank. With this information, the marketing team can focus their marketing efforts on just the bank clients identified by this model and hence derive higher ROI for their marketing efforts.

**Further work to be done:**

1. This project covered 3 models (Logistic regression, Random Forest, Gradient Boost). Because of performance issues, SVM was dropped. It would have been great to see results from SVM as well using facility like Google Colab.
2. The dataset was imbalanced (few positive classes, lots of negative classes). This was taken care of by evaluating models under AUC PR metric which is not sensitive to imbalanced classes. It would have been great to resample dataset using SMOTE technique to ensure balanced classes and then doing the model training.