PREDICT THE SUCCESS OF TELEMARKETING – DIRECT MARKETING

TEAM NUMBER: BADM_045

SUBMISSION NUMBER: BADM_1049

BUSINESS ANALYTICS AND DATA MINING CHAMPIONSHIP 2019

BTECH 3RD YEAR DATA SCIENCE, NMIMS'S MPSTME

TEAM MEMBERS:

ABHISHEK DOPPALAPUDI

HARSH GUPTA

RIDDHI MEHTA

Table of Contents

INTRODUCTION

Respected Jury,

This following document contains the results of our analysis on the dataset of a telecom company which used voice calls from call centres to pitch data packs to old customers and fresh targets. We have curated this document to try to explain or solve the problem for the telemarketing company.

We have tried to make it more convenient for our readers to understand what we've done by writing BUSINESS INSIGHTS within this box but not following the same for DATA ANALYTICS.

We want to give you an unequivocal answer as to what the company should do in the future so in that spirit, we have paraphrased the goal and the meaning of the target variable as we have inferred it.

The goal of this report is to **EXPLAIN THE FACTORS THAT CAUSE A CHANGE IN CUSTOMERS BUYING DATA PACK and to PREDICT whether or not a customer will BUY the data pack**.

Definition of data pack - When you refer to the word data pack it can come in many forms such as a mobile data pack. A mobile data pack refers to an add-on which can **enable you to boost the amount of data which you can use on your mobile phone**. The rate at which you use your data can also be monitored, so you know how much data you have left. Mobile data is a service which provides a similar service to WIFI and allows you to connect to the internet. So, the purpose of a data pack is to increase the amount of data that your mobile has access to.

Now that we have set our premise, we humbly request you to read our report.

Thanking you,

Team Number - BADM_045

Submission Number - BADM_1049

PROPOSED METHODOLOGY

We have split the dataset in the very beginning after doing EDA because we realized the following:

****BUSINESS INSIGHT****

Although the campaign is the same, there are **two different groups of customers the telemarketing company is trying to target**. It is evident from the data that it is easy to separate out customers who have been contacted for the first time and people who have already been exposed to a previous marketing campaign.

Our motivation in doing so is because we have identified that the company has two different objectives when approaching these two different customer types –

A. **Customer retention, Customer churning and unresponsive targets**. (Already been exposed to previous campaigns)
B. **Customer acquisition**. (fresh targets)

So, the company has two distinct problems that it should focus on.

Steps taken in order to split the dataset into half: -

Based on three columns in our dataset:

- passdays: number of days that passed by after the customer was last contacted from a previous campaign - numeric, -1 means client was not previously contacted)
- previous: number of contacts performed before this campaign
- poutcome: outcome of the previous marketing campaign (categorical)

We have identified which customers are **fresh targets** and those who have already been contacted and we have split the data on the poutcome column.

Two different groups

Group 1 – Category – unknown – because they have never been contacted before – fresh targets

Group 2 – Category – success, failure and other – repeated targets

Now one might ask the question as to why we have split the dataset in the first place without using Decision Trees and forcing the first split on poutcome to achieve the same result.

MOTIVATION FOR SPLITTING THE DATASET AND NOT USING DECISION TREES DIRECTLY ON THE ENITRE DATASET WHY?

****BUSINESS INSIGHT****

It makes business sense to separate those who have already been targeted in the previous campaigns because now the telemarketing company should target both these groups of customers differently. The goal with those who have already been contacted is – Retain those who have already been converted, identify churning customers and identify which customers should not be targeted because they haven't been converted even after multiple attempts.

The goal with the **fresh** target's subset is to identify the important factors that will accurately predict if he or she will buy the data pack.

Also, if we were to simply work with the whole dataset , we do not get any insights regarding repeatedly targeted customers which is also an important factor for the company because

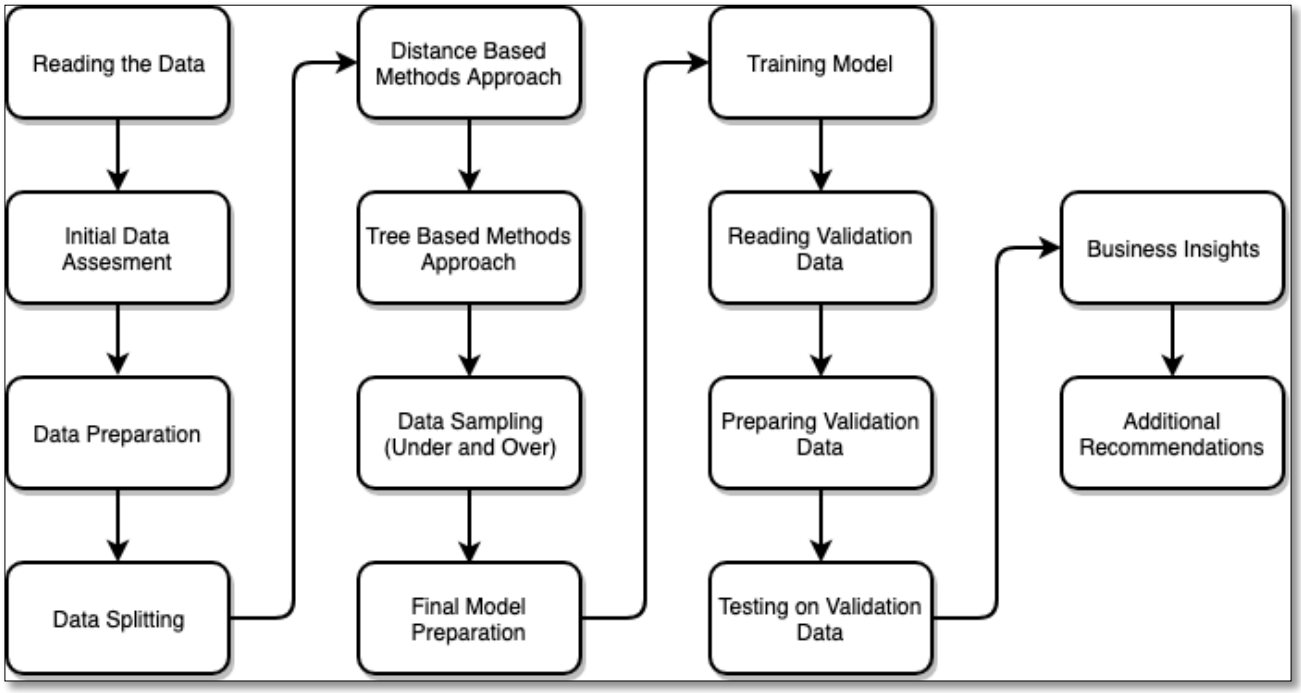
IN A PRICE SENSITIVE INDUSTRY WITH A HIGH CHURN RATE IT IS IMPORTANT TO OBTAIN ANALYSIS AND INSIGHTS ON HOW TO RETAIN AND KEEP your customers or understand why they churn.

****ANALYTICAL MOTIVATION****

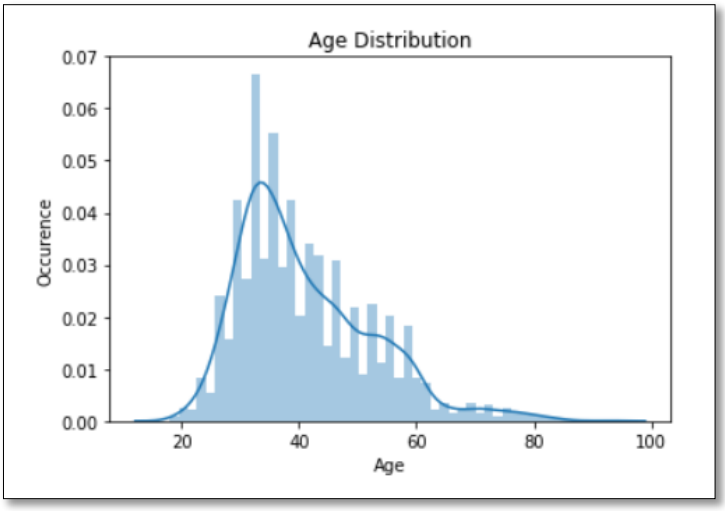
1. **Huge Class Imbalance -** There was a huge class imbalance amongst the different categories of poutcome and hence any algorithm (including decision trees that would be run on the entire dataset would find it difficult to pick up patterns regarding already targeted customers).

2. **Lack of interpretability** - Even if we did overcome the class imbalance problem, interpreting the model when both types of customers are clubbed together in the same dataset is difficult.

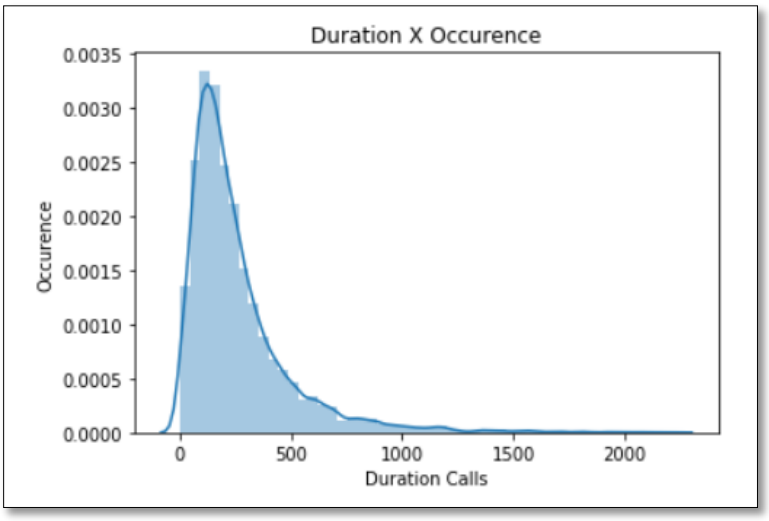
FLOWCHART OF APPROACH



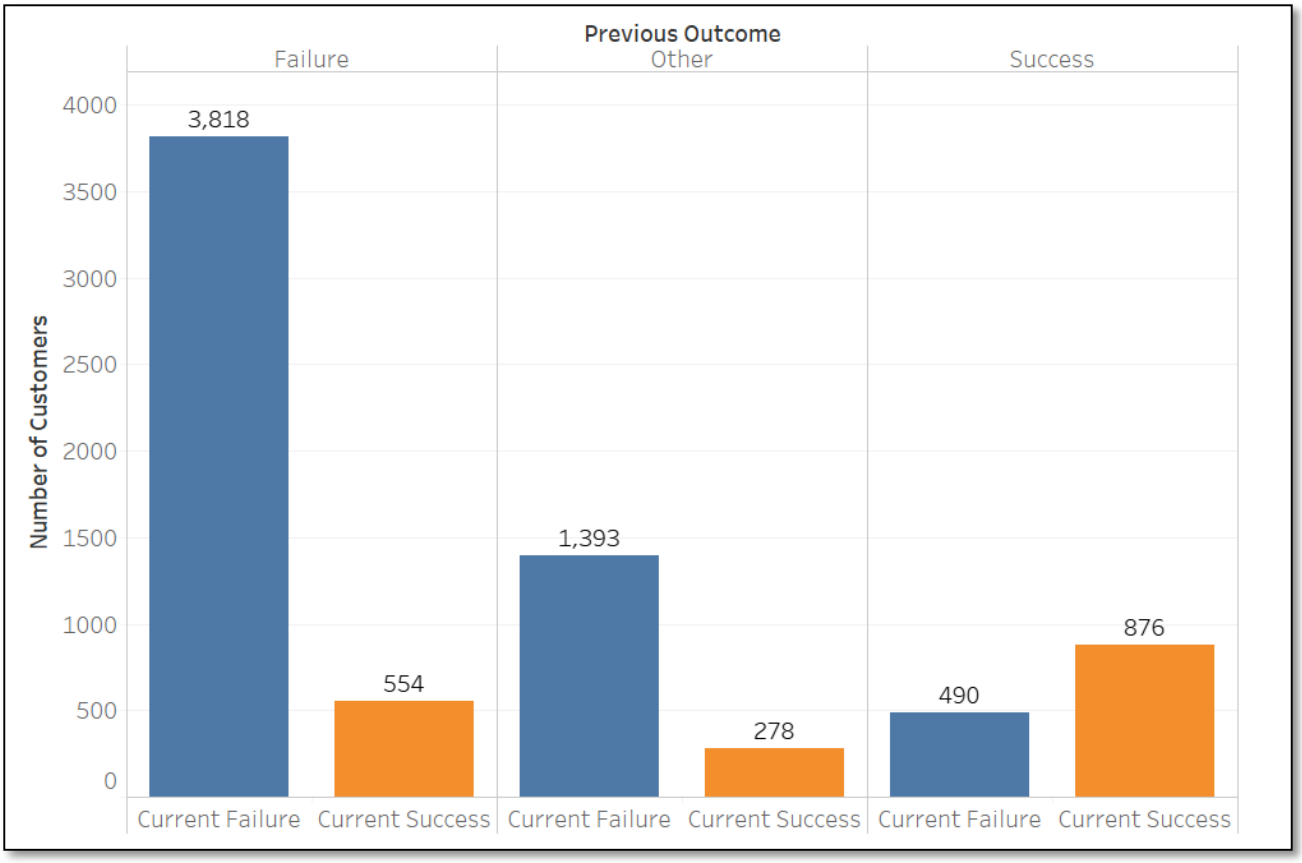
EXPLORATORY DATA ANALYSIS



This graph helped us to decide the bins of the different clusters of age. We have focused more on the middle density of the distribution, whereas the data points after 4th quartile have been altogether condensed into one cluster. Although, it was a trial and error method because we wanted to be sure about the different combinations of the bins that could be made and used.



Seeing this graph, it makes sense to eliminate the column of duration. It is highly skewed and there is one more important thing to note that the call duration for the most part is always going to be in a particular range. Moreover, this column is dependent on the last call, and it can get very subjective depending on the situation of the caller and the client in the last call. Hence we are planning to drop it.



**** BUSINESS INSIGHTS ****

The above graph shows that a high percentage of users who have given positive responses in the past are more likely to give positive responses in the current campaign as shown in the last segment of the graph (Success Section). Similarly, a very high number of users tend to reject the offer of the current campaign if they have not subscribed to the deal in the previous campaigns, as shown as in the first segment of the graph (Failure Section).

A substantial number of users in the third segment of the graph tend to churn that is, failure after success. We see that 490 users out of a total of 1366 that had given successful responses in the last campaign have not shown positive results in the current campaign. The churn percentage is 35.8%. Hence, we infer that the company should also focus on:

1. Trying to reduce the churn (retain customers)

Interpreting the “OTHER” category, as a non-decisive choice, we can interpret that a majority of the users tend to not give positive response in future campaigns. This can be resolved by trying to offer better schemes to indecisive customers.

DATA CLEANING AND CONSOLIDATION

We have performed **two** very important transformations on the dataset before splitting it.

1. Campaign column –

    - Why?

We have created a new column where we have subtracted 1 from the campaign column because it INCLUDES THE LAST CALL – the current row of data is information on that last call.

So, it is imperative that we subtract 1 to find out the number of times they have been contacted before the last call so that we know how many times they have been contacted before the call (the current data rows were recorded)

2. Duration column –

    - Why?

Our intention here is to compare two realistic models in predicting if a customer will subscribe to the data pack. The duration is not known before a call is performed. After the last call, the result is obviously known. Thus, "duration" could not be used as an independent variable and was excluded after the data was read in.

Also converting continuous variable age into a categorical variable?

Why have we done this?

****BUSINESS INSIGHT****

Business insight – Our reasoning for converting age into a categorical variable is that it is easier to interpret the results from a business perspective. For example, there might not be a significant difference between ages 15 and 17 and it is hard to differentiate them on that basis. But if you place them into bins they will belong to the same category. We've made 4 bins on the basis of young, middle aged, old and very old.

We also did this expecting people belonging to the same age category usually display similar characteristics with regards to data usage.

SCALING

We have **NOT** used any scaling on the dataset because of the following reason – One of the primary reasons why scaling is done is to bring variables of different magnitudes into the same range so that the data becomes unit independent where the algorithm does not assign a high weightage simply because its magnitude is high. (E.g. Algorithm will treat 7kg as less than 1000gm because it does not understand the unit).

It also helps make computation of distance based metric algorithms faster.

Our motivation for NOT using any scaling is twofold: -

1. Only two of the columns in our dataset is numerical. The rest are categorical which are unaffected by scaling.

2. We are building tree-based model whose performance is scaling independent because it is not distance based algorithm (absence of Euclidean space).

Label encoding – Label Encoder converts each class under specified feature to a numerical value.

#Python fact – Let us now understand in brief why this is required for python and not in R.

Python's SKLEARN has been programmed in a way to only accept inputs in numerical formats.

Hence, any categorical variables need to be converted in terms of either ordinal, nominal or encoded format for the model to work. This is not the same case in R.

Now, let's move onto Data Analysis.

DATA ANALYSIS

Now that we have completed data consolation, cleaning and EDA we finally come to the all-important part of model building.

The crux of our problem statement in order to predict the target is NOT JUST ACCURACY.

Our primary objective is two- fold:

1. To identify important features in our model so that we can gain insights which can be implemented in business.

2. TO TACKLE THE VERY IMPORTANT PROBLEM OF CLASS IMBALANCE. The fundamental problem with the dataset is that 90% of the data is with target 0 and only 10% with target 1.

We want to ensure when the machine learning model is tested against a data point about a customer that has been converted, it should be able to recognize and classify that person correctly.

**** IMPORTANT ****

So now we have to decide which one is the best model, and we have *two types* of wrong values:

1. **False Positive** – means the client HAS NOT SUBSCRIBED to the data pack, but the model thinks he did. This is more harmful, because we think that we already have the customer but in real, we don't and maybe we lost him in other future campaigning's.

2. **False Negative** – means the client SUBSCRIBED to the data pack, but the model said he hasn't. It is not good but still manageable, as we have that client and, in the future, we'll discover the truth and has not lost as much as we lose in the mistake made above.

SINCE WE HAVE SPLIT OUR DATASET INTO TWO DIFFERENT COMPONENTs, we are now proceeding to build the model on the dataset **which has people who have never been contacted in the past.**

Now that our OBJECTIVE is very clear let us now proceed and build the model.

1. We now begin with LOGISTIC REGRESSION – The standard algorithm to start with in order to build a supervised classification model.

   WHY?

   Motivation to use logistic regression:-

- Very easy to interpret.
- Gives us significant variables. – which others don't

But before proceeding, let's understand:

ASSUMPTIONS OF LOGISTIC REGRESSION

1. Binary logistic regression requires the dependent variable to be binary and ordinal logistic regression requires the dependent variable to be ordinal.

2. Logistic regression requires the observations to be independent of each other. In other words, the observations should not come from repeated measurements or matched data.

3. Logistic regression requires there to be little or no multi collinearity among the independent variables. This means that the independent variables should not be too highly correlated with each other.

4. Logistic regression assumes linearity of independent variables and log odds. Although this analysis does not require the dependent and independent variables to be related linearly, it requires that the independent variables are linearly related to the log odds.

5. Logistic regression typically requires a large sample size. A general guideline is that you need at minimum of 10 cases with the least frequent outcome for each independent variable in your model. For example, if you have 5 independent variables and the expected probability of your least frequent outcome is 0.10, then you would need a minimum sample size of 500 (10*5 / 0.10).

All these assumptions were not satisfied.

The reason why the 5th point is highlighted because the first time we ran logistic regression the recall score was really low which is why we had to find a way to deal with class IMBALANCE.

Hence, we dealt with the class imbalance problem using SMOTE.

WHY SMOTE?

SMOTE stands for Synthetic Minority Oversampling Technique. This is a statistical technique for increasing the number of cases in your dataset in a balanced way. The module works by generating new instances from existing minority cases that you supply as input. This implementation of SMOTE does not change the number of majority cases.

The new instances are not just copying of existing minority cases; instead, the algorithm takes samples of the feature space for each target class and its nearest neighbours, and generates new examples that combine features of the target case with features of its neighbours. This approach increases the features available to each class and makes the samples more general.

SMOTE takes the entire dataset as an input, but it increases the percentage of only the minority cases.

**** Python Code for Fresh Targets ****

```
# Logistic Regression - FRESH TARGETS
estimator = LogisticRegression(penalty='l1', class_weight='balanced', solver = 'liblinear')
selector = RFE(estimator, 5, step=1)
selector = selector.fit(fXtr, fYtr)
logpred = selector.predict(fXte)
print(classification_report(fYte,logpred))
print(confusion_matrix(fYte,logpred))
selector.estimator_.coef_
```

```
              precision    recall  f1-score   support

           0       0.73      0.77      0.75      7528
           1       0.76      0.72      0.74      7528

    accuracy                           0.75     15056
   macro avg       0.75      0.75      0.75     15056
weighted avg       0.75      0.75      0.75     15056

[[5834 1694]
 [2137 5391]]
```

**** Python Code for Repeated Targets ****

```
# Logistic Regression - REPEATED TARGETS
estimator = LogisticRegression(penalty='l1', class_weight='balanced', solver = 'liblinear')
selector = RFE(estimator, 5, step=1)
selector = selector.fit(rXtr, rYtr)
logpred = selector.predict(rXte)
print(classification_report(rYte,logpred))
print(confusion_matrix(rYte,logpred))
selector.estimator_.coef_
```

```
              precision    recall  f1-score   support

           0       0.79      0.72      0.76      1426
           1       0.74      0.81      0.78      1425

    accuracy                           0.77      2851
   macro avg       0.77      0.77      0.77      2851
weighted avg       0.77      0.77      0.77      2851
```

Above image is the code for Feature selection using Recursive Feature Selection.

The goal of recursive feature elimination (RFE) is to select features by recursively considering smaller and smaller sets of features.

As you can see, after using smote, the recall has increased.

What this means is that it has BECOME better AT PREDICTING the case where the customer is actually converted.

INFERENCE BASED ON LOGISTIC REGRESSION

If we are using logistic regression, a negative coefficient simply implies that the probability that the event identified by the DV happens decreases as the value of the IV increases.

We are explaining the important feature of logistic regression and not for random forest classifier although it does give feature importance it does not give the coefficient or change in output per unit change in the independent column. It also gives us columns which are statistically significant that is $p<0.05$.

So for business insights, we are interpreting the output of the logistic regression model.

**** BUSINESS INSIGHTS for FRESH TARGETS ****

AGE
Since it has a negative coefficient this basically means that an older person would be less likely to buy a data pack. The telecom companies should emphasize on telemarketing to the younger generation.

CONNECT
This makes logical sense as someone with multiple phones might not opt for a data pack. Someone with multiple phones might also be really rich or very important which is why this might not attract him or her necessarily. The telecom company should try to avoid people with multiple phones.

MARITAL
The coefficient shows that people who are not married are more likely to buy the data pack which also corroborates our conclusion that younger people should be targeted, as people get married only after a certain age.

LANDLINE
The coefficient has the highest negative coefficient which can be easily explained. In most cases people who have LANDLINE CONNECTION also get their WIFI set up by the same facility and

IN CONCLUSION

Our solution with regards to customer acquisition calls is a PROFILE that:

- Is young
- Does not have multiple phones
- Is not married
- Does not have a landline

In order to try to increase its conversion rate, we believe that the company should target this type of profile.

SIMILARLY, FOR CUSTOMER RETENTION DATASET (REPEATED TARGETS):

All the aforementioned features are important expect for marriage and the new factor that comes in is POUTCOME – (the column that mentions the success of the previous marketing campaign).

We can conclude that there is a high chance that a customer can be retained if he has already subscribed once to the data pack. It is all the more important for the telecom company to try and retain its previous customer base as the success rate is much higher than the success rate of a first time call for customer acquisition.

Now that we have used logistic regression, with the aim of still improving recall and accuracy we move on to tree-based methods.

DECISION TREES

MOTIVATION

They provide a highly effective structure within which you can lay out options and investigate the possible outcomes of choosing those options. They also help you to form a balanced picture of the risks and rewards associated with each possible course of action.

ASSUMPTIONS OF DECISION TREES

- No assumptions about underlying data
- It is a non-parametric model

OUTPUT OF DECISION TREES WITHOUT SMOTE

- Low recall again.

**** Python Code ****

```
dtree = DecisionTreeClassifier(criterion='gini', class_weight={0:1,1:9})
dtree.fit(rXtr, rYtr)
dpred = dtree.predict(rXte)
print(confusion_matrix(rYte, dpred))
print(classification_report(rYte,dpred))
```

```
[[1216  210]
 [ 245  182]]
              precision    recall  f1-score   support

           0       0.83      0.85      0.84      1426
           1       0.46      0.43      0.44       427

    accuracy                           0.75      1853
   macro avg       0.65      0.64      0.64      1853
weighted avg       0.75      0.75      0.75      1853
```

NOW WE COME TO THE IMPORTANT PART

Using **under-sampling** and **over-sampling** methods simultaneously.

The reason why we opted for both these methods simultaneously known as "**classes imbalanced-learn implements for combining over and under sampling methods"** is as follows:

1. Simply under sampling the overrepresented sample is leading to a massive information loss as there are only 10% of the small sample leading to loss of over 25,000 rows

2. Simply using smote to oversample the data in this case is CREATING 40% Synthetic data which is not ideal for the final model that we want to learn.

Hence, we have MANUALLY IMPLEMENTED THIS METHOD WHICH COULD HAVE BEEN DONE USING SMOTETOMEK or SMOTEENN.

OUTPUT OF DECISION TREES USING OVER AND UNDER SAMPLING TOGETHER

- Increase in Recall

**** Python Code ****

```
# downsampling of 0
downsampled = resample(fY0, replace = False, # sample without replacement,
                       n_samples = 12000, # match minority n
                                random_state = 27) # reproducible results
```

```
# upsampling of 1
upsampled = resample(fY1, replace = True, # sample without replacement,
                       n_samples = 12000, # match minority n
                                random_state = 27) # reproducible results
```

```
dtree = DecisionTreeClassifier(criterion='gini') #criterion = entopy, gini
dtree.fit(xtr, ytr)
dpred = dtree.predict(xte)
print(classification_report(yte,dpred))
print(confusion_matrix(yte, dpred))
```

```
              precision    recall  f1-score   support

           0       0.96      0.78      0.87      3000
           1       0.82      0.97      0.89      3000

    accuracy                           0.88      6000
   macro avg       0.89      0.88      0.88      6000
weighted avg       0.89      0.88      0.88      6000

[[2353  647]
 [  87 2913]]
```

RANDOM FORESTS – FINAL MODEL

MOTIVATION

- It is one of the most accurate learning algorithms available. For many data sets, it produces a highly accurate classifier.
- It runs efficiently on large databases.
- It can handle thousands of input variables without variable deletion.
- It gives estimates of what variables are important in the classification.

ASSUMPTIONS OF RANDOM FORESTS

It has no model underneath, and the **only assumption that it relies is that sampling is representative**. But this is usually a common assumption. For example, if one class consist of two components and in our dataset one component is represented by 100 samples, and another component is represented by 1 sample - probably most individual decision trees will see only the first component and Random Forest will misclassify the second one.

GRID SEARCH CROSS VALIDATION ON RANDOM FORESTS

We have improvised the model by using grid search CV since, we needed to evaluate the model using all the different and essential combinations of the hyper parameters. We also observe that, the recall as well as the precision increases when we select the best estimator given by grid search CV. The code snippet as well as the output is shown below as our final working model, which we have also used for validation set.

**** PYTHON CODE – FINAL WORKING MODEL – FRESH TARGETS ****

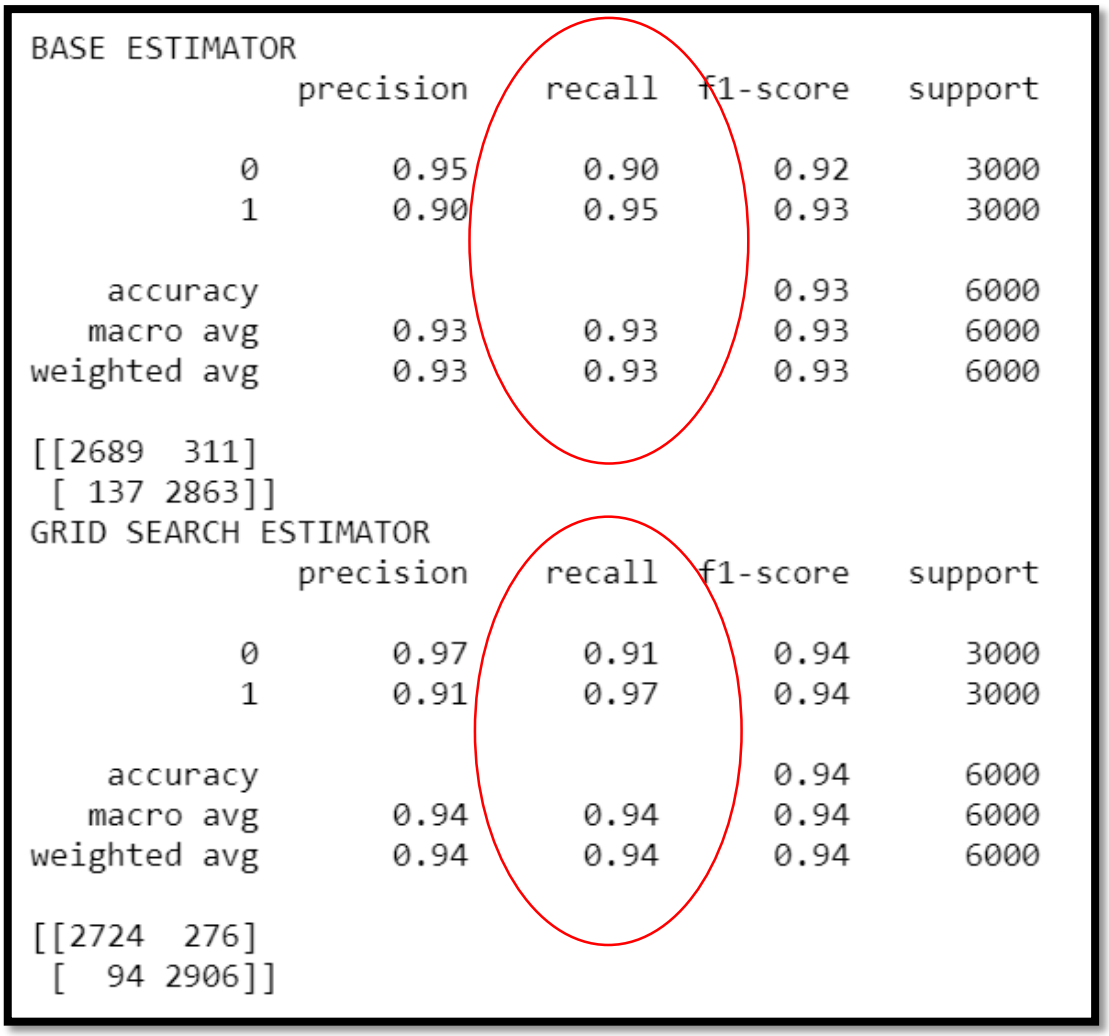
```
rfc = RandomForestClassifier()
rfc.fit(xtr,ytr)
rfcpred = rfc.predict(xte)


params = {'n_estimators':np.arange(50,200,50),
          'max_depth': [80, 90, 100, 110],
          'max_features': [3, 4],
          'criterion':['gini','entropy']}

grid_search = GridSearchCV(estimator = rfc, param_grid = params, cv = 3)
grid_search.fit(xtr, ytr)
gridpred = grid_search.predict(xte)

print("BASE ESTIMATOR")
print(classification_report(yte,rfcpred))
print(confusion_matrix(yte, rfcpred))

print("GRID SEARCH ESTIMATOR")
print(classification_report(yte,gridpred))
print(confusion_matrix(yte, gridpred))
```



```
BASE ESTIMATOR
              precision    recall  f1-score   support

           0       0.95      0.90      0.92      3000
           1       0.90      0.95      0.93      3000

    accuracy                           0.93      6000
   macro avg       0.93      0.93      0.93      6000
weighted avg       0.93      0.93      0.93      6000

[[2689  311]
 [ 137 2863]]
GRID SEARCH ESTIMATOR
              precision    recall  f1-score   support

           0       0.97      0.91      0.94      3000
           1       0.91      0.97      0.94      3000

    accuracy                           0.94      6000
   macro avg       0.94      0.94      0.94      6000
weighted avg       0.94      0.94      0.94      6000

[[2724  276]
 [  94 2906]]
```

CHALLENGES FACED

Summary of the challenges faced while solving the problem statement are as follows:

1. The main challenge we faced was the challenge of **Class imbalance**. This problem was more or less resolved by properly understanding different methods of making both the samples reach parity so that we could work on them.

2. Another problem we faced and we tackled was the problem of what to do to obtain insights regarding those who have already been targeted. We decided to split the data and treat them as **two different problems**.

3. **Lack of interpretability** for our final model. Although the precision, recall and accuracy of our final model was high, we faced hurdles in trying to convert the same in business insights. We will work on understanding that better in the future.

USEFUL DATA STRATEGIES – ADDITIONAL RECOMMENDATIONS

Additional steps the telemarketing company should take in order to predict the target variable with better accuracy in the future.

As Data Scientists, it's not just the target variable that we should try to explain and predict. We should also, as consultants on the matter of Data gathering and analysis also suggest other variables or columns the company should record in order to predict the target better.

The following suggestions to collect data on additional factors is based on research online regarding telemarketing and its impact on people buying data packs.

**** BUSINESS INSIGHTS - DATA STRATEGIES ****

1. Service provider used by other family members – This data will be very useful because if there are family members who use the same service then the company can pitch different family pack schemes to them. This data is also not difficult to obtain as credit card companies or banks know the Billers. When you pay the bill, they ask for the service provider. So, this data can be collected and then used for analysis.

2. Location of user – This is one of the most important aspects that has been missed out in this dataset. With this information we can find out if the company has a leading position or trailing position with regards to the number of towers. If that area has better connectivity then this aspect can be pitched while making the call. Also, proportion of users in an area that consume data packs from the same company can be analysed and used to predict.

3. Time when the call is made to the customer/potential customer – This is very important because we can infer from the JOB COLUMN about the rough schedule of each person and avoid calling them when he or she is busy. We can also analyse the times when customers are the most responsive.

4. **This is for customers only** – Number of calls made to complain about the service. With this data the company can then analyse how to retain its customers better.

----------XXXXX----------