Final Lab report on Banking Dataset

Main Objective: In this lab we are using the Banking data set available in Kaggle where the main objective is being able to identify customers who are willing to take long term deposits from the bank. Our target variable in the dataset is variable 'Y' which is a binary variable of 0 and 1. We will be using classification models ranging from simple Logistic Regression, Ensemble models, Random Forest to identify such customers from the dataset. Along with this we will also try to identify features from the dataset that have the most effect on a customer being able to make this decision.

Description about the data: There has been a revenue decline in the Portuguese Bank and they would like to know what actions to take. After investigation, they found that the root cause was that their customers are not investing enough for long term deposits. So the bank would like to identify existing customers that have a higher chance to subscribe for a long term deposit and focus marketing efforts on such customers.

The data is related to direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, in order to access if the product (bank term deposit) would be subscribed ('yes') or not ('no') subscribed.

- Age: numeric; age of a person
- Job: Categorical, nominal; type of job ('admin.' , 'blue collar' , 'entrepreneur' , 'housemaid' , 'management' , 'retired' , 'selfemployed' , 'services' , 'student' , 'technician' , 'unemployed' , 'unknown')
- Marital: categorical, nominal;marital status ('divorced' , 'married' , 'single' , 'unknown'; note: 'divorced' means divorced or widowed)
- Education: categorical, nominal; ('basic.4y' , 'basic.6y' , 'basic.9y' , 'high.school' , 'illiterate' , 'professional.course' , 'university.degree' , 'unknown')
- Default: categorical, nominal;has credit in default? ('no', 'yes', 'unknown')
- Housing: categorical, nominal; has housing loan? ('no', 'yes', 'unknown')
- Loan: categorical, nominal; has personal loan? ('no', 'yes', 'unknown')
- Contact: categorical, nominal; contact communication type ('cellular', 'telephone')
- Month: categorical, ordinal; last contact month of year ('jan', 'feb', 'mar', … , 'dec')
- day_of_week: categorical, ordinal; last contact dow ('mon', 'tue', 'wed', 'thu', 'fri')

- duration: numeric; last contact duration, in seconds. Important note: this attribute highly affects the output target (e.g., if duration=0 then y='no')

- campaign: numeric; number of contacts performed during this campaign and for this client (includes last contact)
- pdays: numeric; number of days that passed by after the client was last contacted from a previous campaign (999 means client was not previously contacted)
- previous: numeric; number of contacts performed before this campaign and for this client
- poutcome: categorical, nominal; outcome of the previous marketing campaign ('failure', 'nonexistent', 'success')

Summary of Data Exploration:

Train data:

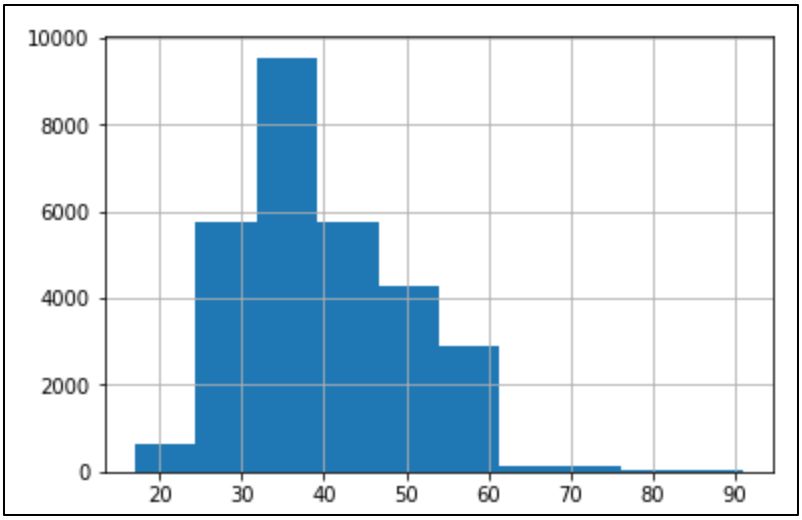
age	job	marital	education	default	housing	loan	contact	month	day_of_week	duration	campaign	pdays	previous	poutcome	y
49	blue-collar	married	basic.9y	unknown	no	no	cellular	nov	wed	227	4	999	0	nonexistent	no
37	entrepreneur	married	university.degree	no	no	no	telephone	nov	wed	202	2	999	1	failure	no
78	retired	married	basic.4y	no	no	no	cellular	jul	mon	1148	1	999	0	nonexistent	yes
36	admin.	married	university.degree	no	yes	no	telephone	may	mon	120	2	999	0	nonexistent	no
59	retired	divorced	university.degree	no	no	no	cellular	jun	tue	368	2	999	0	nonexistent	no
29	admin.	single	university.degree	no	no	no	cellular	aug	wed	256	2	999	0	nonexistent	no
26	student	single	basic.9y	no	no	no	telephone	aug	wed	449	1	999	0	nonexistent	yes
30	blue-collar	married	basic.4y	no	yes	no	cellular	nov	wed	126	2	999	0	nonexistent	no
50	blue-collar	married	basic.4y	unknown	no	no	telephone	may	fri	574	1	999	0	nonexistent	no
33	admin.	single	high.school	no	yes	no	cellular	jul	tue	498	5	999	0	nonexistent	no

There are no null values in the dataset, there are around 5 numeric variables and 11 categorical variables including the target variable.

Let's look at each of these variables and get some information about the data.

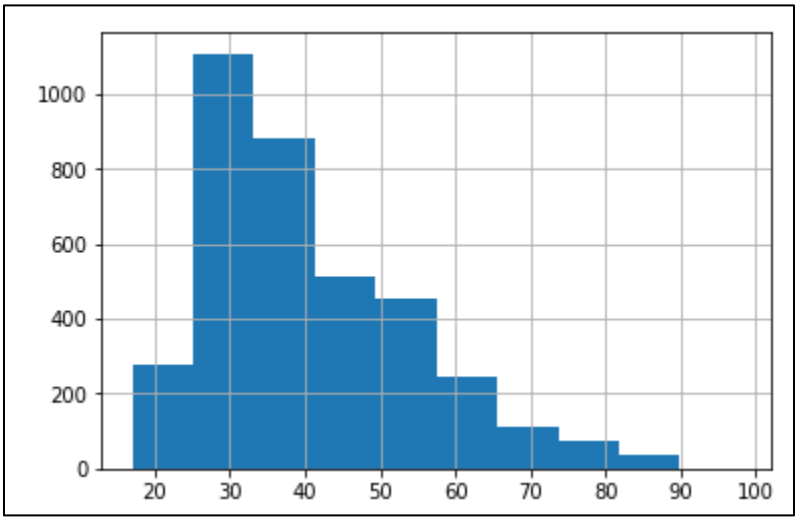
1. Y: Y variable is a categorical variable with 'Yes' and 'No', we transformed this to a numeric categorical variable with Yes: 1 and No: 0
2. Age: Age is a continuous variable comprising of ages of all the customers, we looked at age of all the people willing to get long term loan vs not willing separately

Unwilling



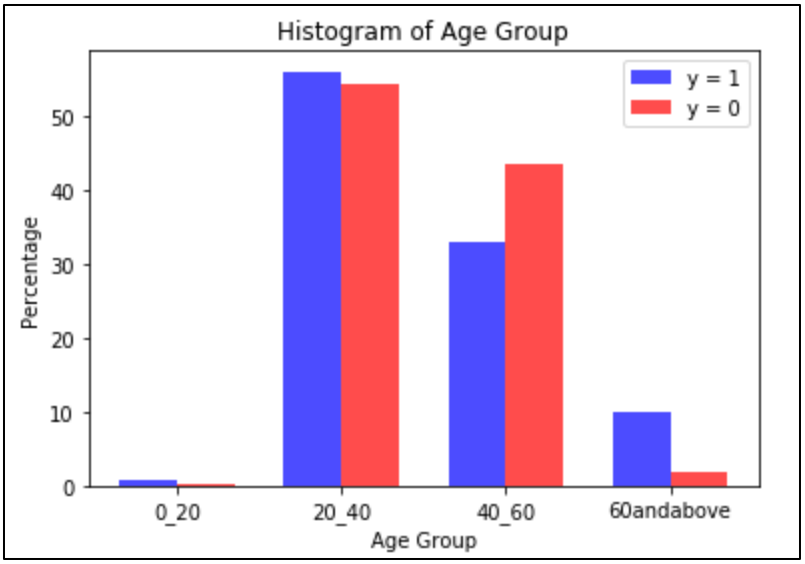
Centered around mean 39.90, and looks like a more normally distributed

Willing



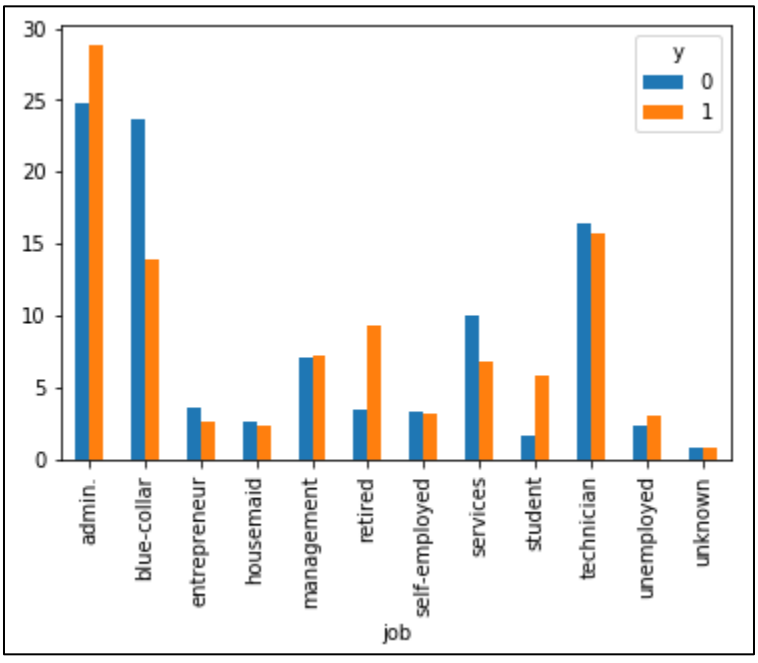
Centered around 40.85 but a bit more positively

Since overall based on just age we were not seeing much of a difference between the distribution for willing vs not willing, I decided to look at it by bucketing the age variable into age groups of 0-20, 20-40, 40-60, 60 and above.



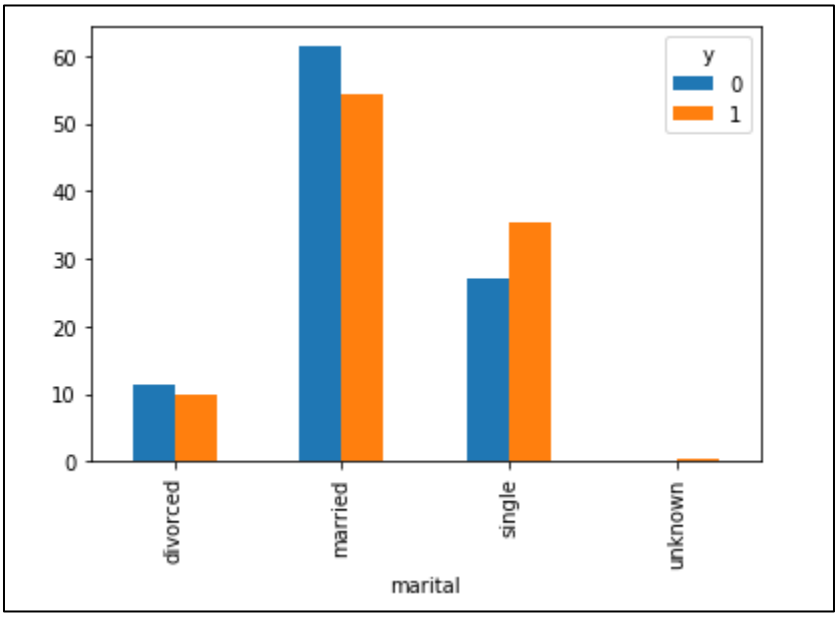
Using age buckets it can be observed that in the age group of 60 and above there are a lot more willing percentage of people than un willing, whereas age 40 – 60 has a more percentage of unwilling to willing. Later on during feature importance we found out that 60 and above age group impacts our models.

3. Job: We looked into the cross tab for this variable and we found that 'Blue-collar' job profile customers have higher unwilling percentage than willing, whereas 'retired', 'student' and 'admin' job profile customers have higher willing percentages compared to unwilling

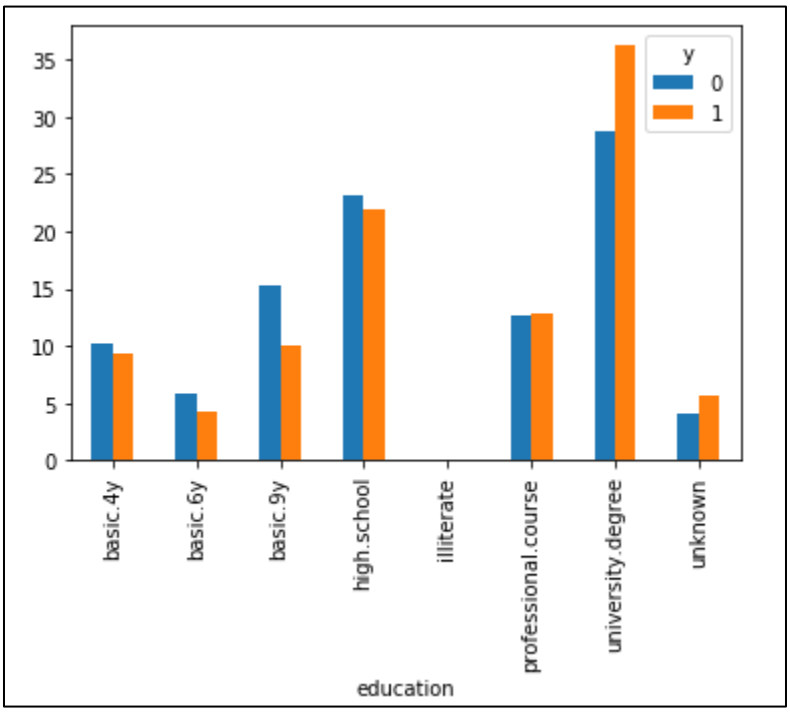


4. Marital: On looking into the cross tab for the marital column we found
    a. Divorced customers have the least tendency to make long term deposits

b. Married customers have slightly more tendency to make long term deposits

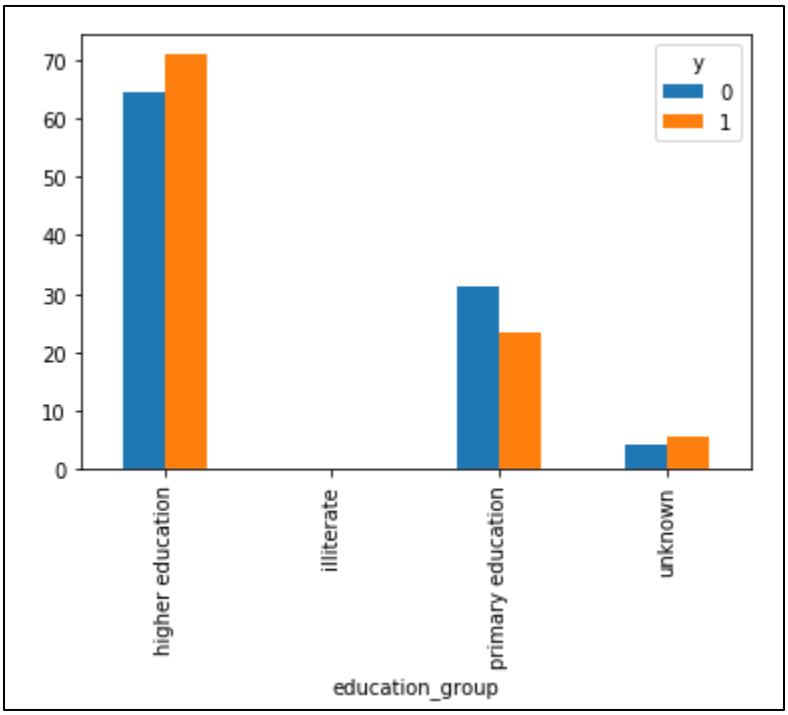


5. Education: In terms of education we first looked at the cross tab for all the different categories



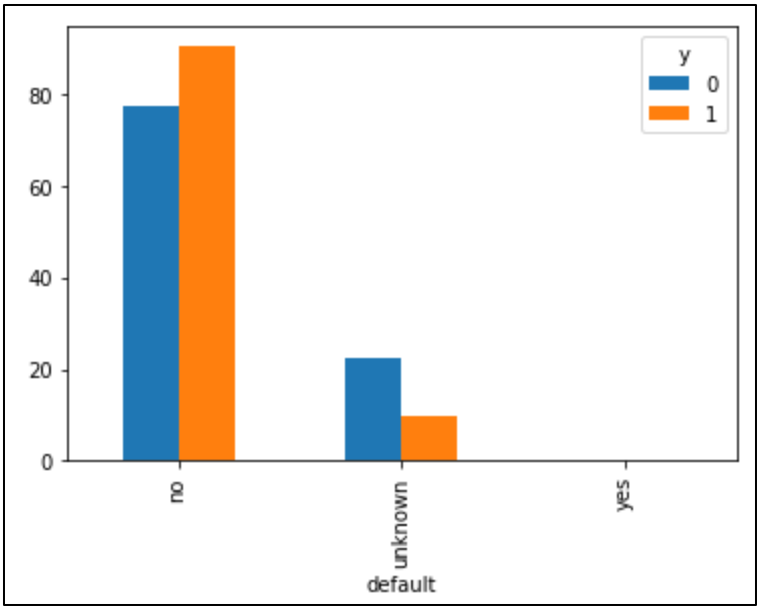
From the above plot we found that less educated customers i.e. high school or below level education tend to be less willing to make long term deposits compared to higher educated customers, using this analogy we merged some of the categories into two buckets

```
education_mapping = {
   'higher education': ['university.degree', 'professional.course', 'high.school'],
   'primary education': ['basic.4y', 'basic.6y', 'basic.9y']
}
```

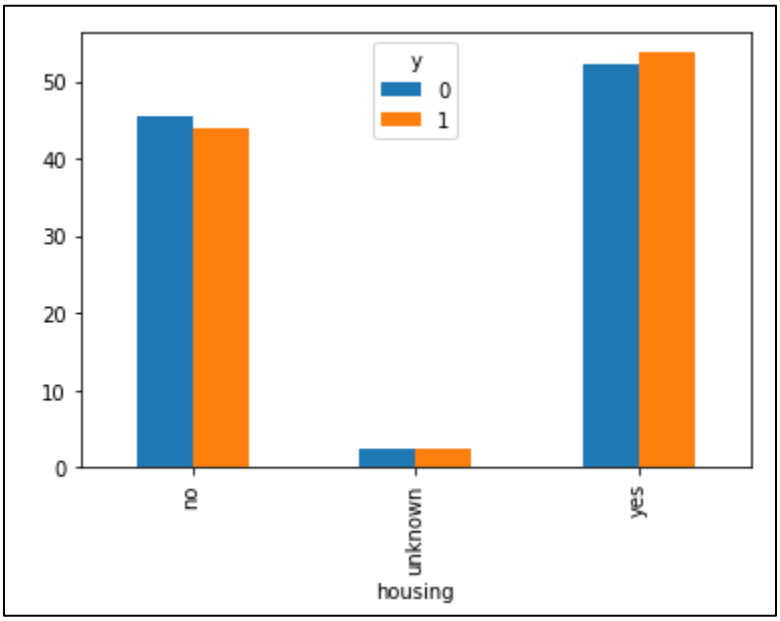


using these new categories, we are able to see a more summarized information of what we had already seen in our previous cross tab

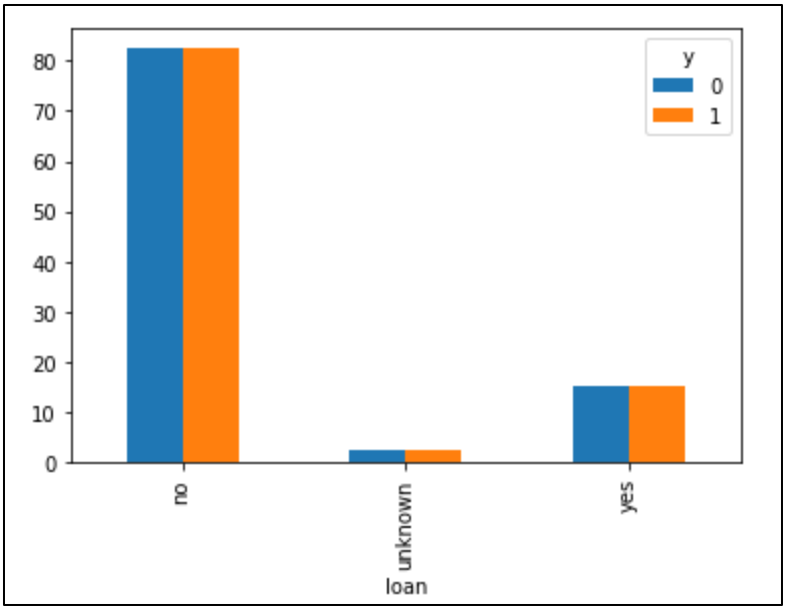
6. Default: From this variable we see that customers who tend to make no defaults on their credit payments tend to be more willing towards making long term deposits



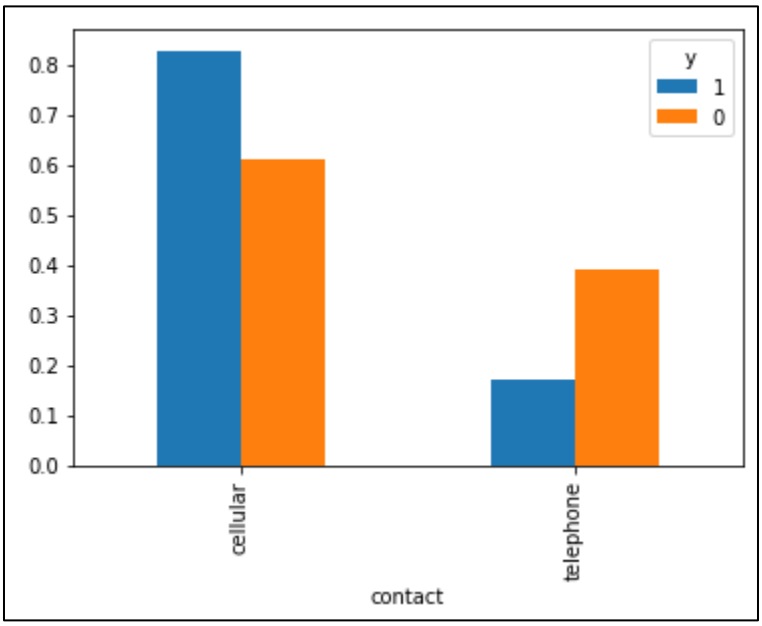
7. Housing: Not much difference between willing and unwilling can be observed from has housing loan variable, with having housing load having a slightly more willing percentage and vice versa



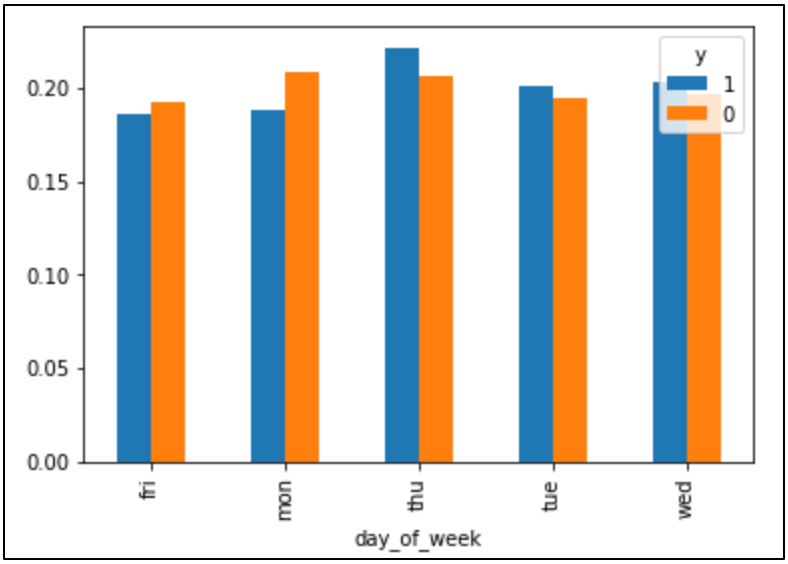
8. Loan – Similar to housing, has personal loan variable also does not give us much information



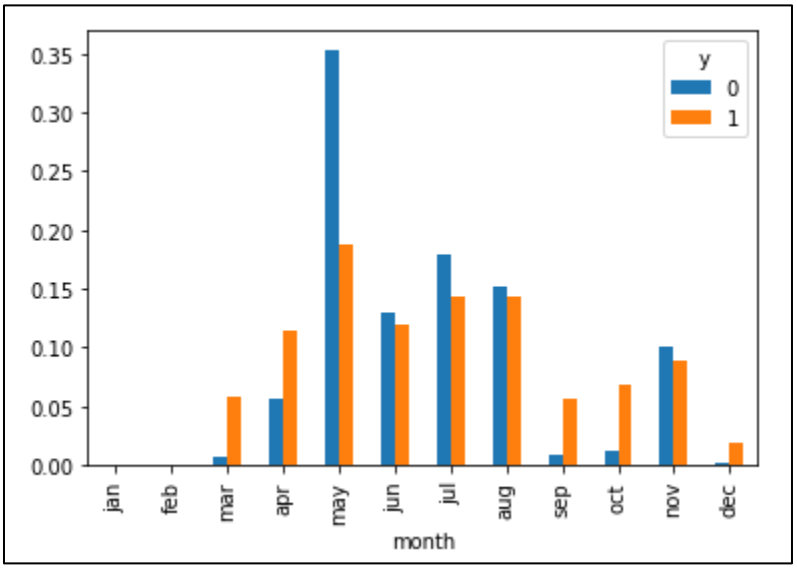
9. Contact: This variable tells us very clearly that customers contacted through telephone tend to be more willing to deposit compared to people contacted using cell phone, **this might tell us similar to age and job variables that older and retired people who tend to use telephone are more willing**.



10. Day of the week – Day of the week last contacted seems to be a very volatile variable to be used into our models and does not give us much information

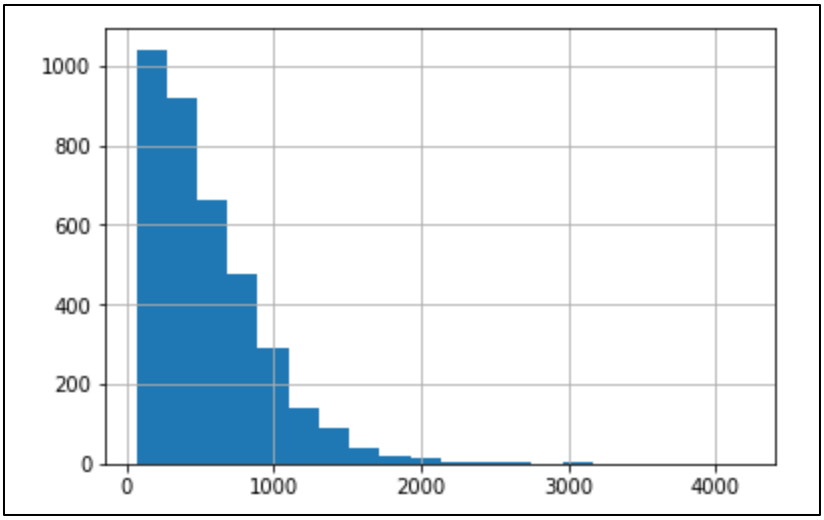


11. Month: Month of the year when the customer was last contacted as part of a campaign, tends to tell us that customers tend to be less willing when they are contacted in the middle of the year May to Aug, and respond to being more willing during the end of the year months or holiday season.



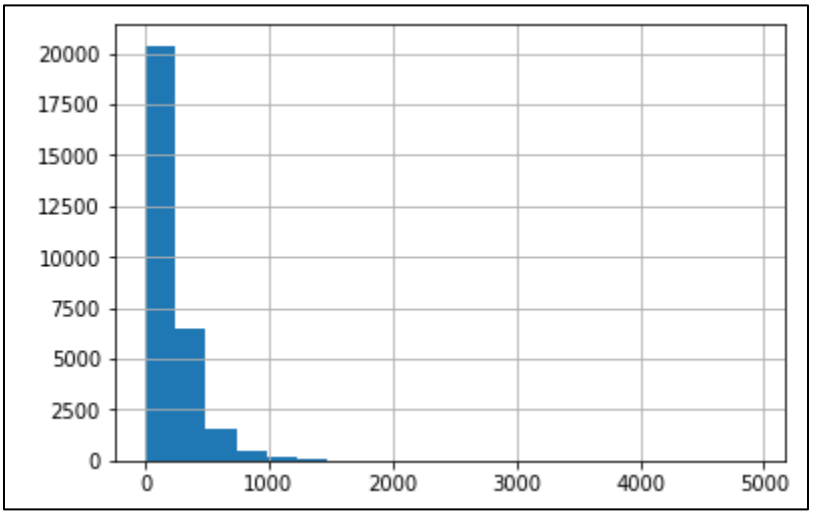
12. Duration: It is a continuous variable telling us about the time duration of the last phone call when the customer was contacted. It was already mentioned in the dataset that this has the highest correlation with the willing customers. I looked into the distribution of the duration variable for willing vs unwilling and also looked at the mean

    Willing



Mean 549.39

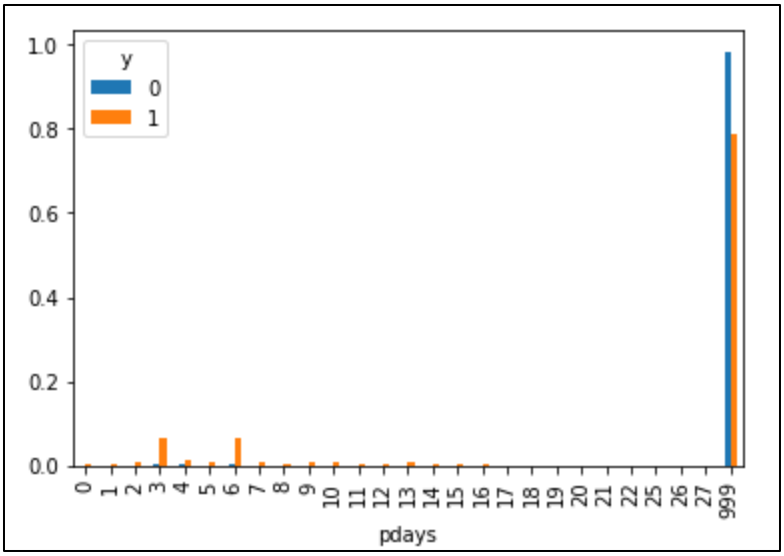
Unwilling



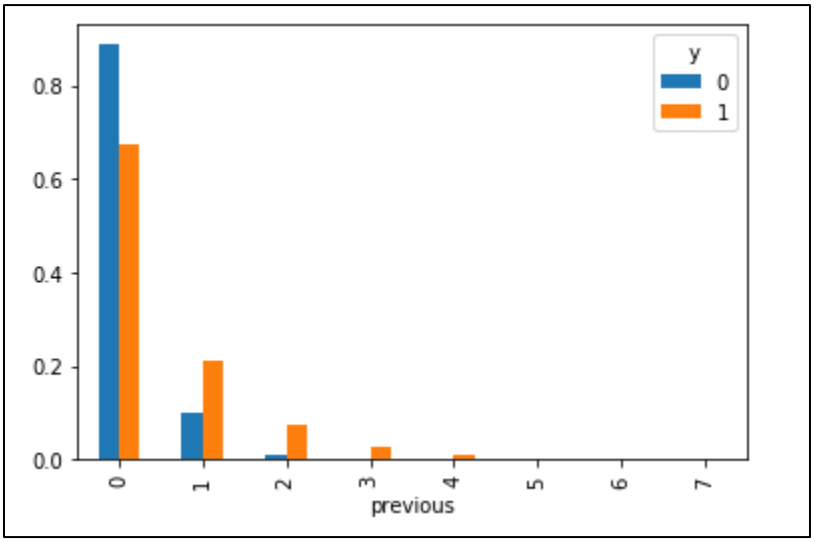
Mean 221.148

Both the distributions are right skewed and there is a huge difference in mean of the distributions, from this we can clearly see that when the customer is more engaged during the phone call and takes more time to talk they can be convinced to make a long term deposit.

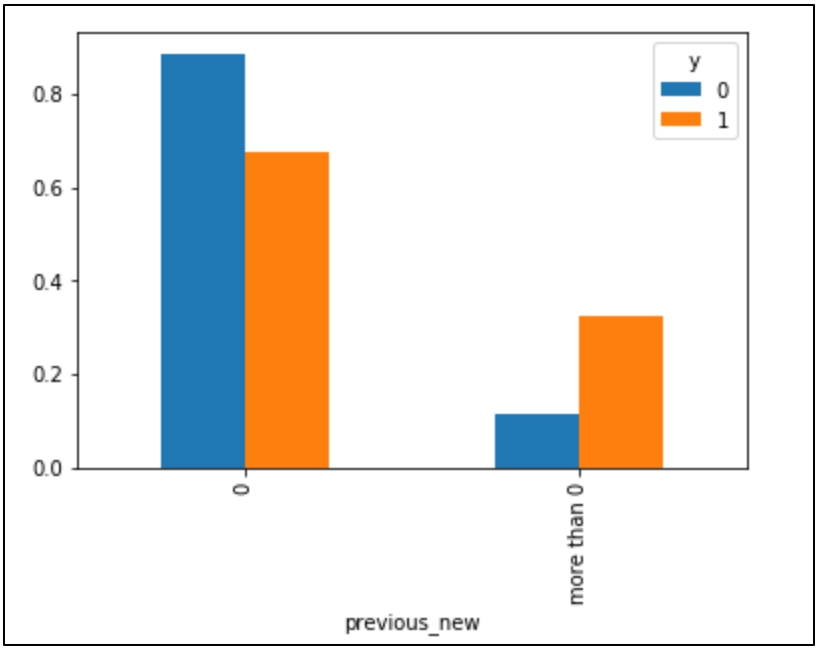
13. Pdays: From the pdays cross tab it can be said that customers who are frequently contacted for campaigns tend to be more responsive towards long-term deposit



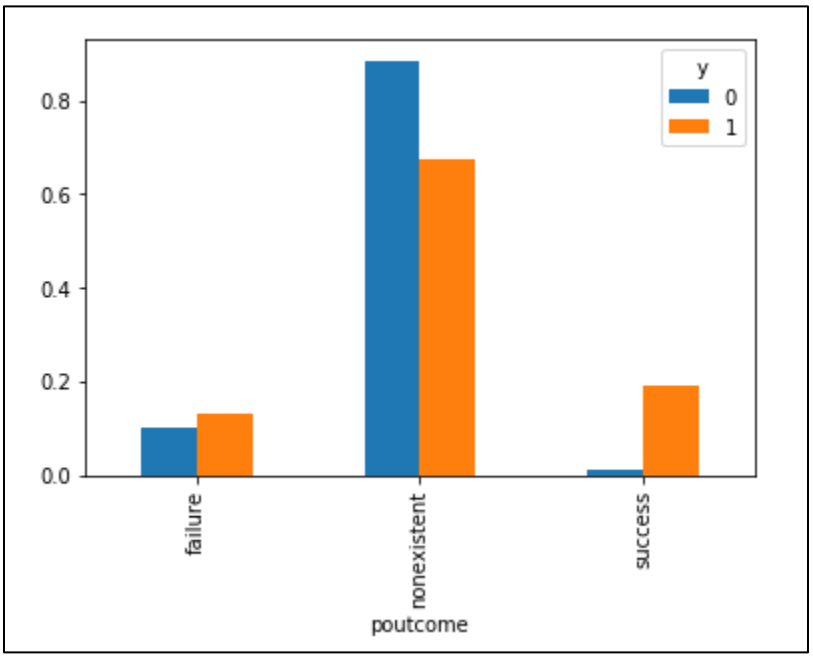
14. Previous: From Previous variable it can be observed if the customer is being contacted the first time though a campaign they are less likely to get a long term deposit



Keeping this is mind what I have done here is create a new variable, where I have kept previous = 0 as is, and transformed rest of the values as more than 0. With this we can say that customers who have been contacted at least once have more chance of making a long term deposit



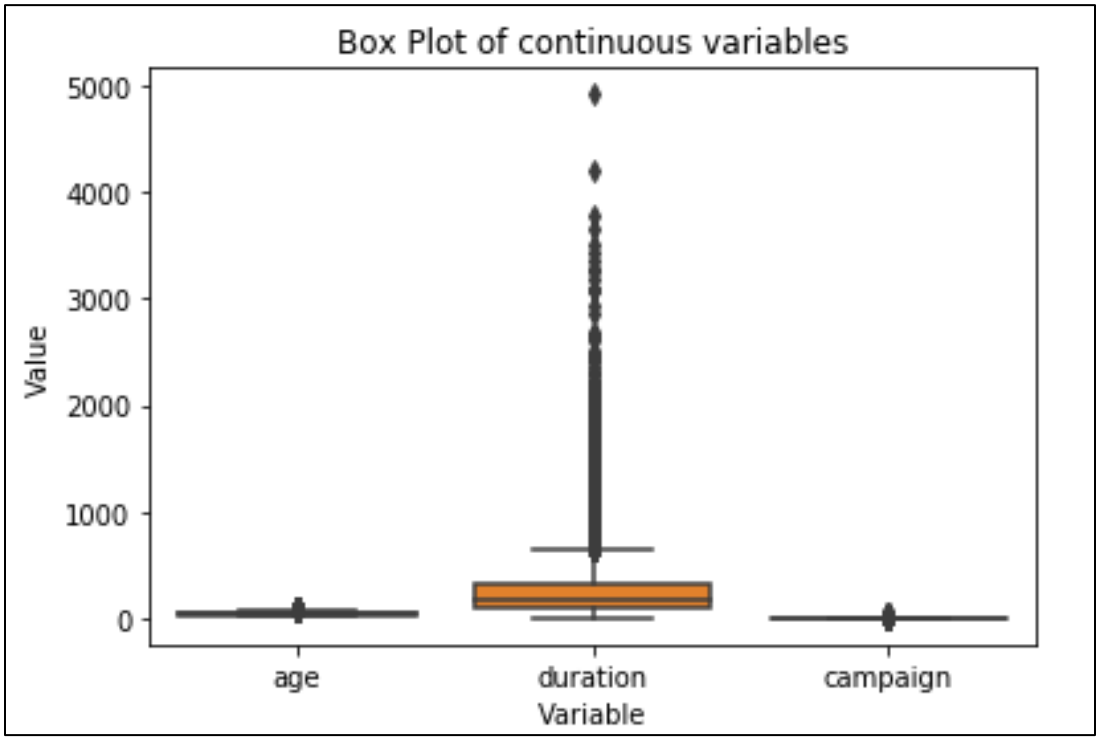
15. Poutcome – It tells us that if the outcome of the previous campaign was successful that means it is more likely to make a deposit, compared to failure of previous campaign or nonexistent. Again putting forward the same information as Previous where people who have been targeted previously in a campaign are more likely to convert.



Next, we performed one hot encoding of some of the variables we had created during EDA

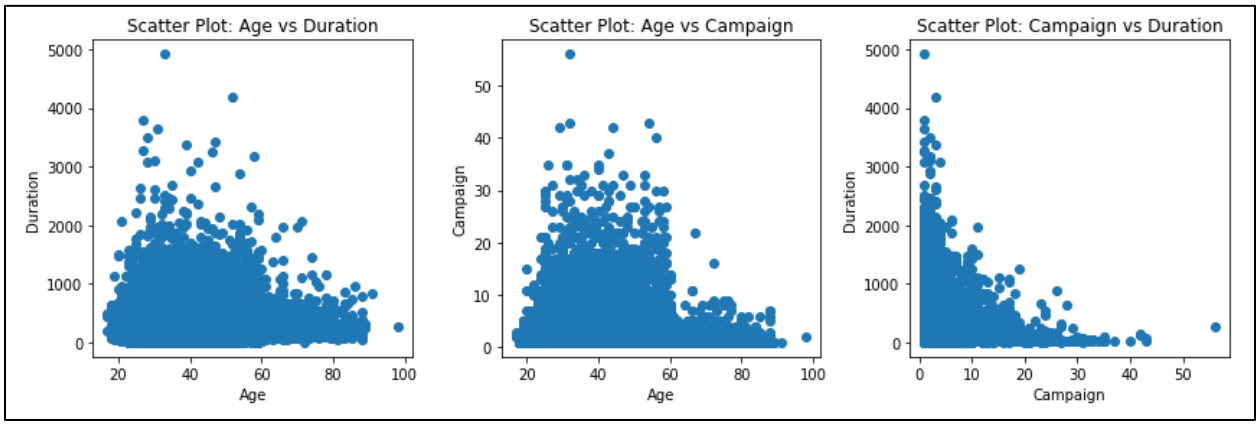
Handling of outliers: Bank data mostly comprises of categorical columns, only continuous variables that we have are age, campaign and duration.

Box plots of Continuous variables:



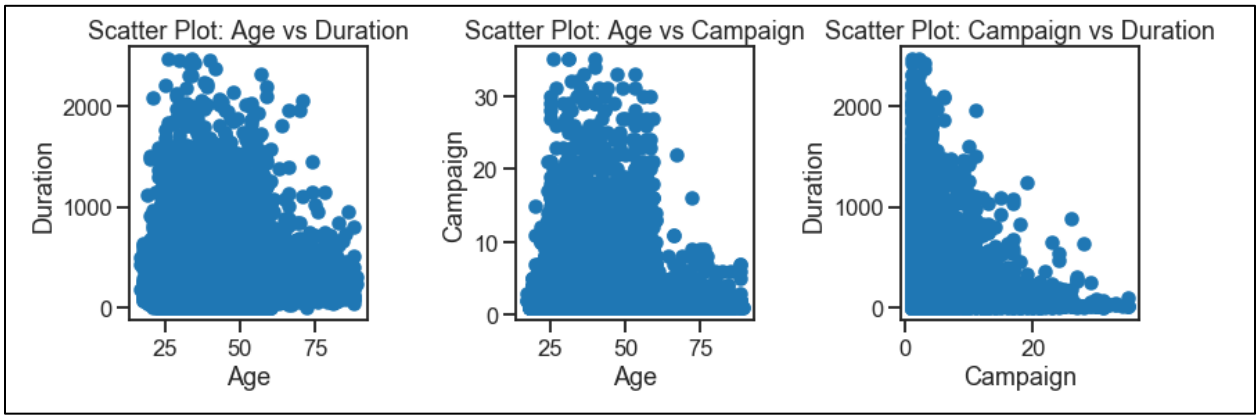
As it can be observed from the box plots that age and campaign have some outliers, but most of the outliers are in duration variable which is depicting the amount of time spent in a call with the customer during latest campaign.

Let's also look at the scatter plots of these variables

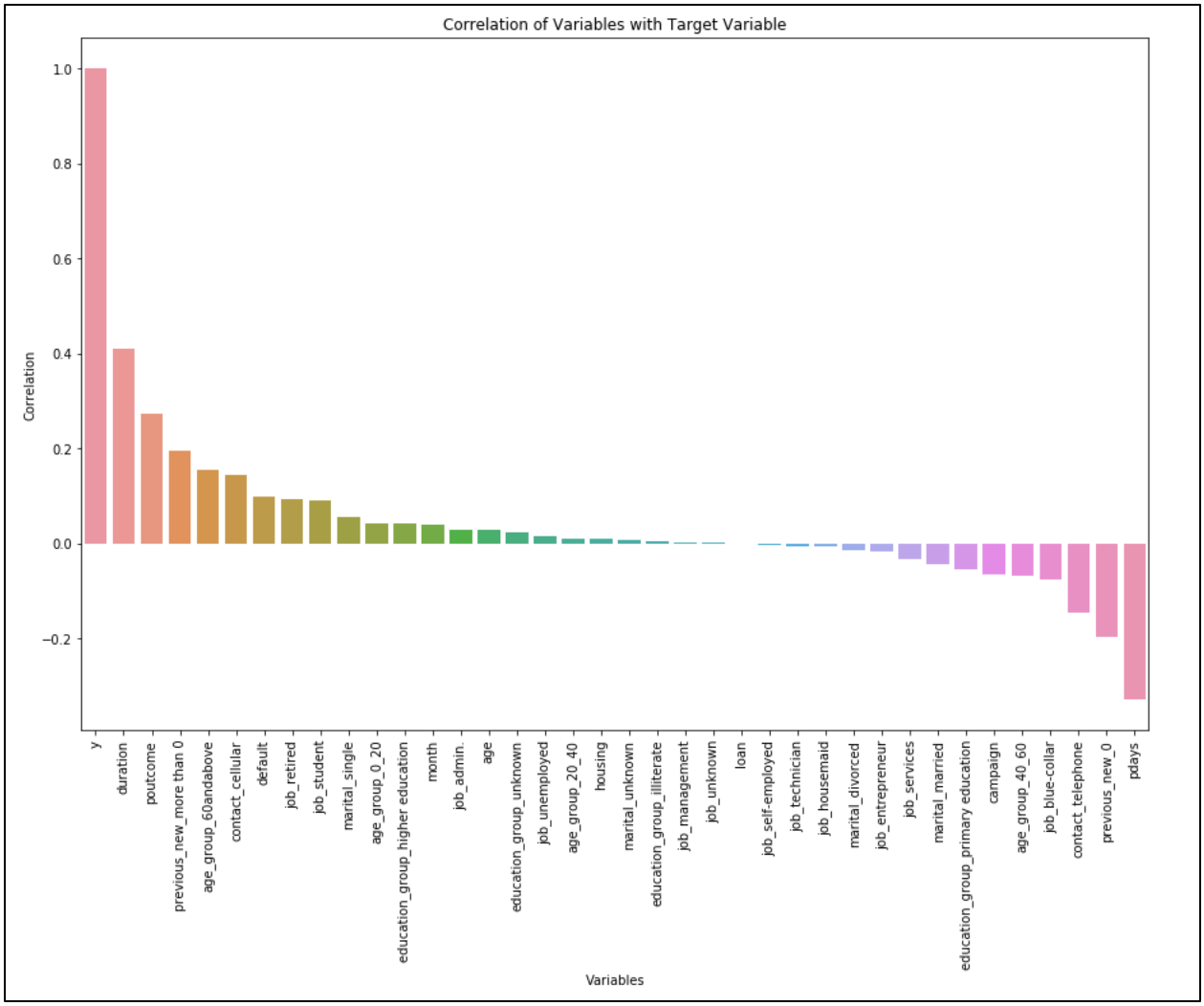


In the above plot in figure 1 and 3 and also from the box plots we can tell that anything above 2,500 in duration is tending to act like and outlier in our data

Also, for age we see that there is one outlier around 99, so we can clip that off. In Campaign again anything above 35 is looking to be an outlier. Using these filter, we cleaned our data of outliers and this is how it is looking after that.



Correlation with Target Y variable



From the correlation histogram plot, we can quickly identify that variable like duration, poutcome, previous more than 0, age group 60 and above are some of the variables that are positively correlated with the target variable as we had also seen during EDA. Similarly, pdays, previous new 0, contact telephone are some of the features that are negatively correlated with Y.

Whereas features like loan, job_self employed, job unknown, job housemaid are features that are having no significant effect on Y.

Deleting some of the original columns which were treated during EDA phase like age, job, marital, education.

Feature Engineering:

In feature engineering mainly we identified our binary, ordinal and numeric variables.

- Binary – Along with the original variables that had categories Yes and No and the new one hot encoded variable, we are having around 29 binary variables.

- Ordinal variables – Default, housing, loan, poutcome, month, pdays these are some of the ordinal variables that we had where the categories were encoded as yes – 1; no – 0; and not sure - -1.
- Numerical variables – After dropping age the only two numeric variables left are duration and campaign

Next, we performed label encoding, ordinal encoding and scaling on the data to bring all of our feature set into the same scale.

We split the data into train and test using stratified sampling and a split ration of 7:3

Modeling:

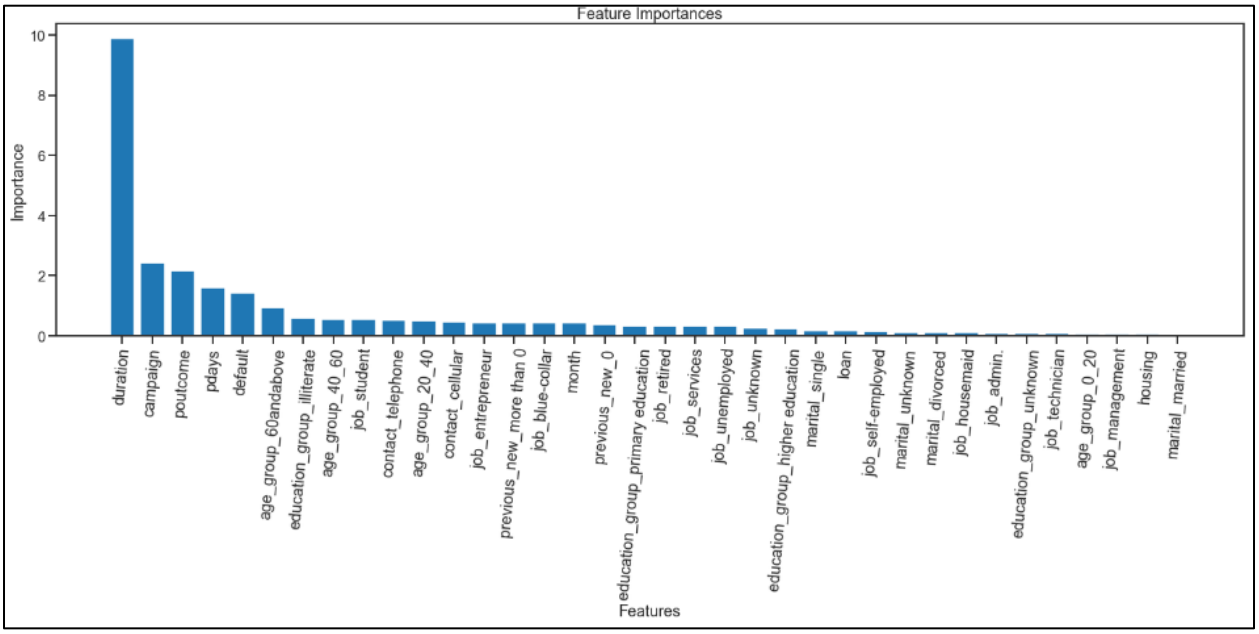
1. Logistic regression with L2 regularization: utilized saga solver and max iteration of 1000

   Metrics –

   - Accuracy: 90.09%
   - Precision: 69.27%
   - Recall 34.32%
   - F1: 45.90

   Overall, it can be seen due to high class imbalance metrics such as f1 and recall are not good at all even below 50%

   Feature importance –



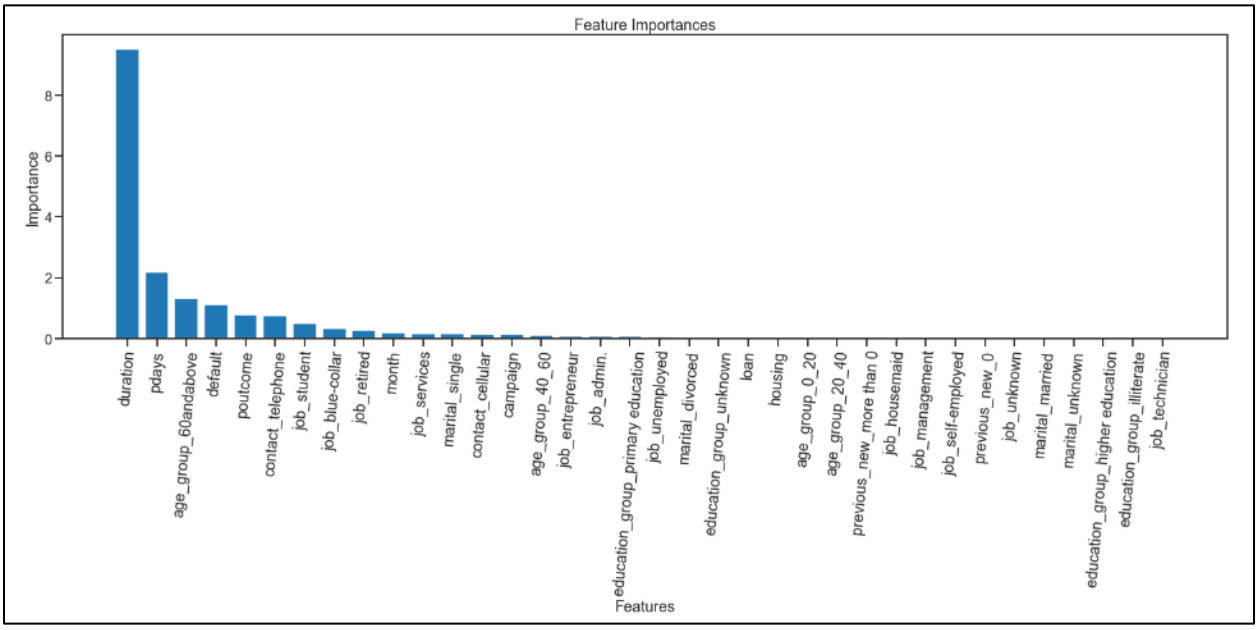
Like what was seen during the correlation plot, some of the features strongly correlated with the Y variable have high coefficicents like Duration, campaign, poutcome, pdays, default. Whereas features such as marital, housing, job age have very low coefficients.

2. Logistic regression with L1 penalty: Utilizing saga solver, max iteration = 1000 and C = 0.07

   Metrics –

   - Accuracy – 90.09%
   - Precision – 70.15%
   - Recall – 33.24%
   - F1 – 0.45%

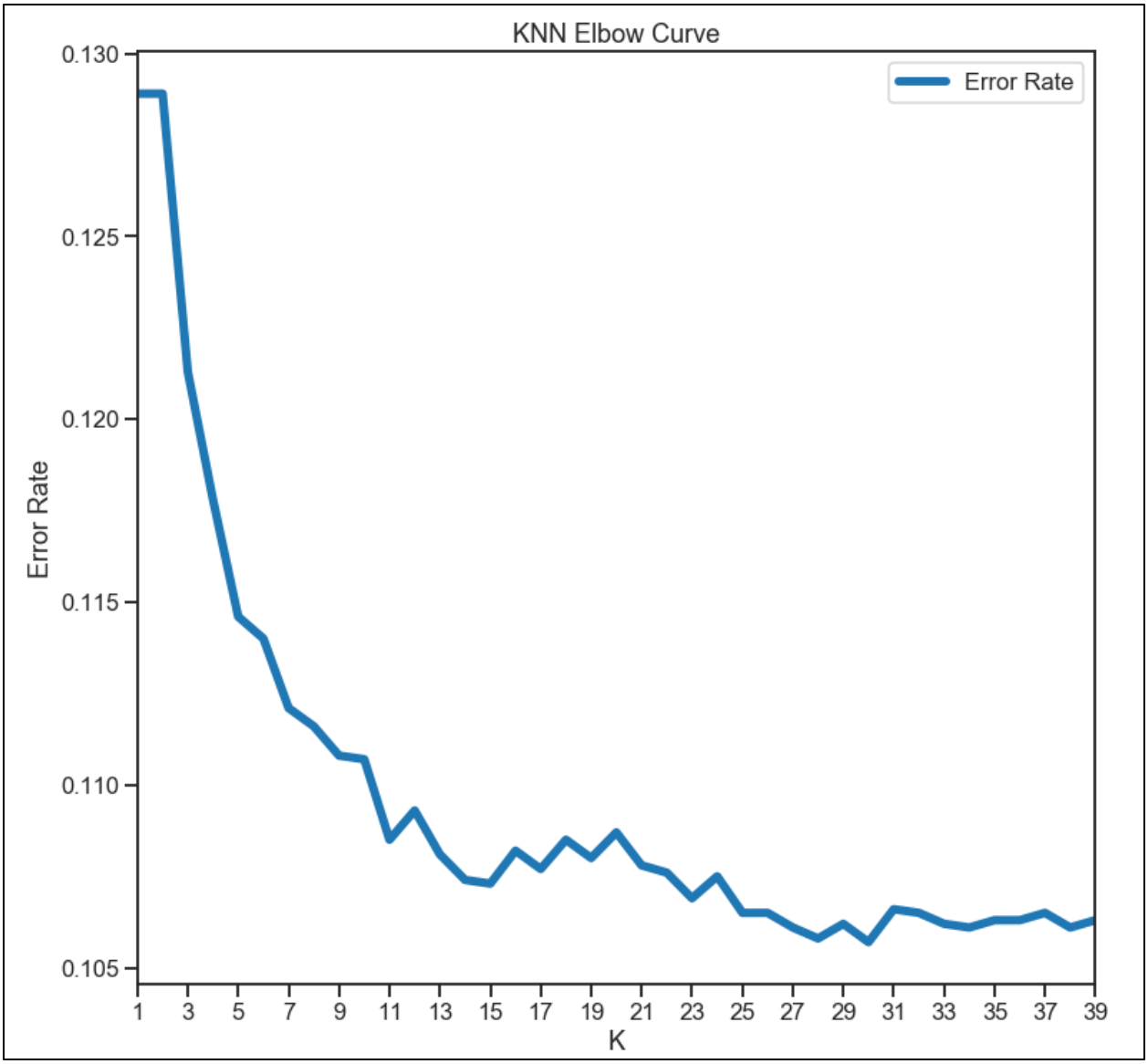
   Feature Importance –



   Overall, we do not see a huge change in terms of model performance between L1 and L2 penalty, also most of the feature's importance see with l2 penalty are like what we are seeing with l1 penalty with some minor changes.

3. KNN Models:

   Using KNN we first tried to see what the best K would be using vs error rate.



We can see that the lowest points seem to be around 15, using that we get the following metrics.

Metrics:

- Accuracy- 89.58%
- Precision- 62.53%
- Recall- 18.19%
- F1- 28.19%

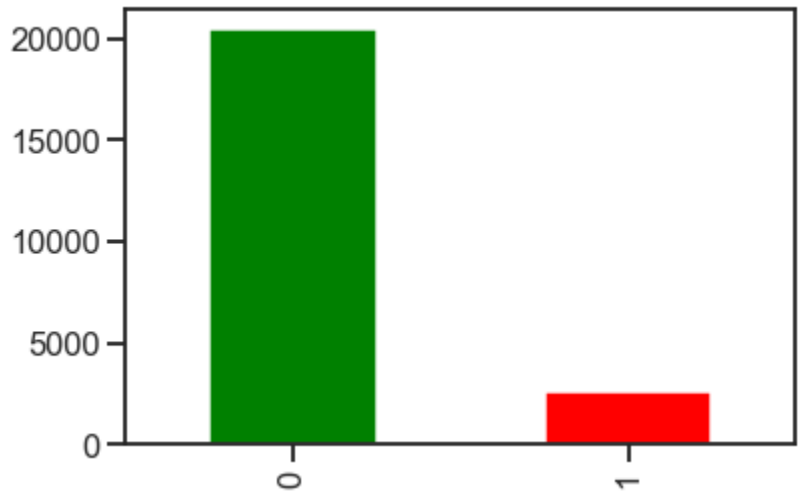
Nothing improved from our previous models.

Since our bank is already suffering from **financial losses**, therefore using our models we will be trying to **improve our Precision**. Since we want to **acquire as many customers** as possible without **making many false positives** or **losses**.

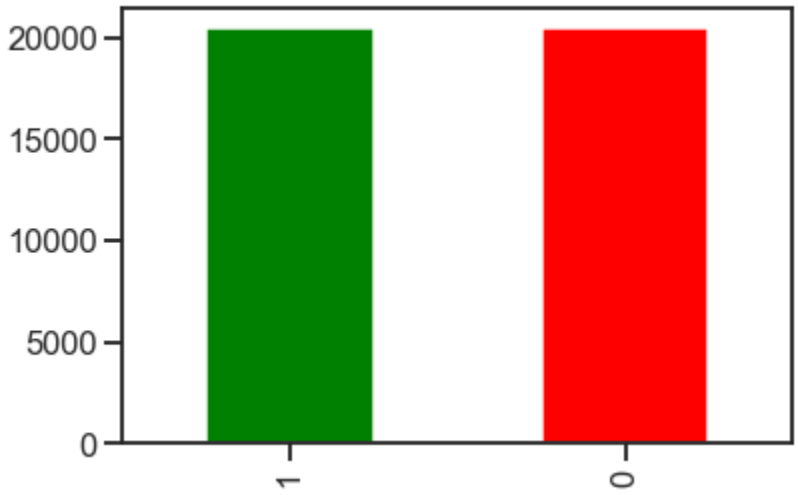
Let us try to deal with our class imbalance-

Smote:

Currently our class imbalance –



Using Smote



4. KNN using Smote –
   Metrics –
   - Accuracy – 80.01%
   - Precision – 31.37%
   - Recall – 65.58%
   - F1score – 42.44%

   Using smote as we had thought that our precision is getting hampered, but recall has improved by a lot

5. LR using class weight –
   Using class weight of 7:3 ratio of unwilling to willing, and L2 penalty with c= 0.07, we get the following results.
   Metrics –
   - Accuracy – 90.08%
   - Precision – 72.20%
   - Recall – 19.18%
   - F1score – 30.32%
6. SVM
   a. With kernel rbf and c= 100

      Metrics –

      - Accuracy – 89.83%
      - Precision – 60.19%
      - Recall – 28.19%
      - F1score – 38.40%

   b. With kenel rbf, c=100 and SMOTE

      Metrics –

      - Accuracy – 84.32%
      - Precision – 38.56%
      - Recall – 66.57%
      - F1score – 48.84%

   c. With kenel rbf, c=100 and class weights

      Metrics –

      - Accuracy – 90.04%
      - Precision – 63.42%
      - Recall – 27.02%
      - F1score – 37.90%

7. Decision trees –
    a. Using grid search cv we found out the best tree parameters

       Metrics –

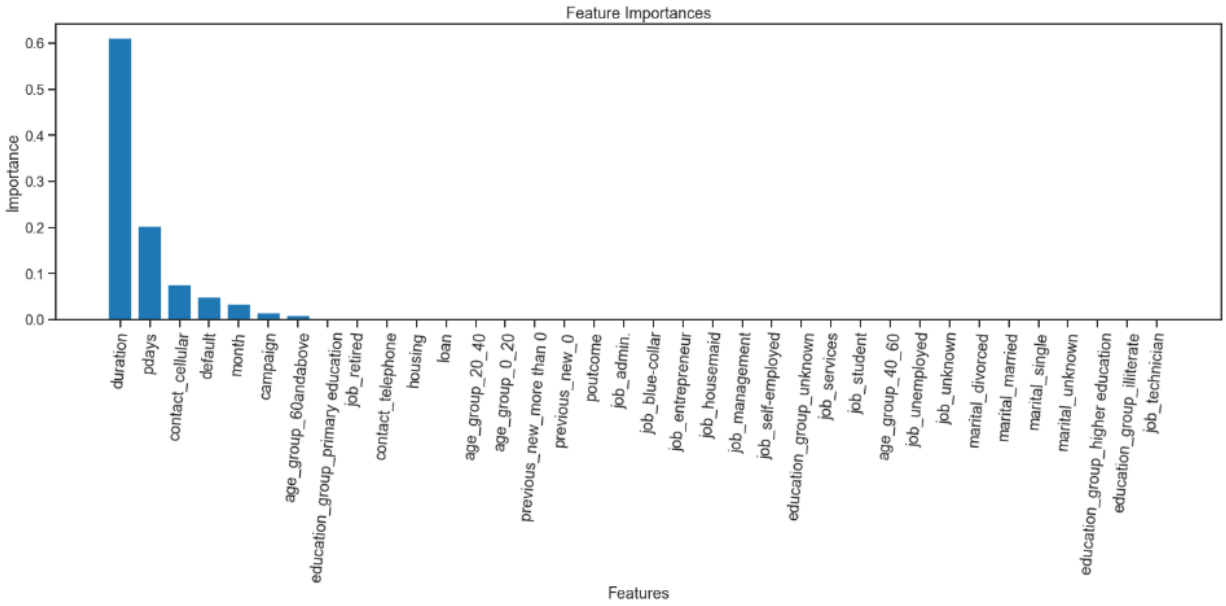
        - Accuracy – 90.87%
        - Precision – 65.30%
        - Recall – 40.18%
        - F1score – 49.74%

    b. Using best tree parameters and smote
       Metrics –
        - Accuracy – 76.26%
        - Precision – 30.04%
        - Recall – 86.39%
        - F1score – 45%

    c. Using best tree parameters and class weight
       Metrics –
        - Accuracy – 90.01%
        - Precision – 75.40%
        - Recall – 16.57%
        - F1score – 27%

With decision trees similar to before we see that without making any adjustments to the class imbalance, we get a good accuracy but average precision recall and f1 score and using smote improves the recall whereas changing the class weights increases the precision.

Feature Importance:



8. Random Forest – Using grid search cv we looked into the best parameters and found the best parameters to be max_depth = 15, max_features = sqrt and n_estimators to be 21
    a. Using grid search cv parameters

       Metrics –

        - Accuracy – 90.47%
        - Precision – 63.69%
        - Recall – 35.40%
        - F1score – 45.51%

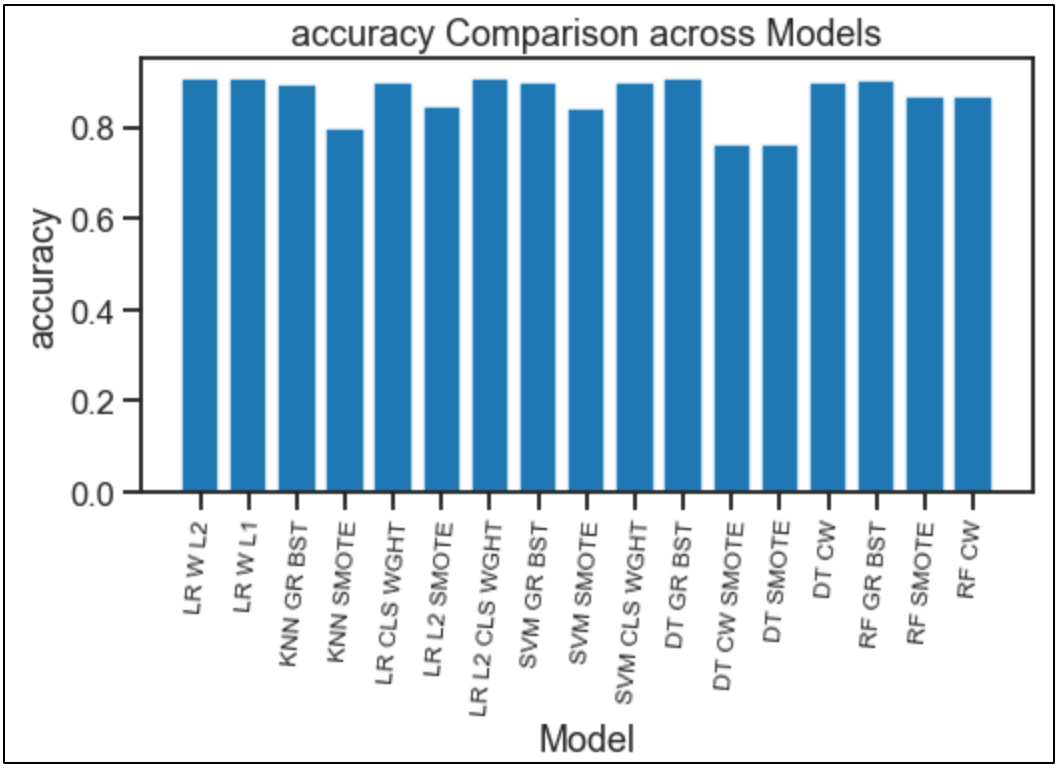
    b. Using best tree parameters and smote
       Metrics –
        - Accuracy – 87.04%
        - Precision – 45.10%
        - Recall – 70.54%
        - F1score – 55.02%

    c. Using best tree parameters and class weight
       Metrics –
        - Accuracy – 87.07%
        - Precision – 45.03%
        - Recall – 68.28%
        - F1score – 54.27%

Key Findings and information –

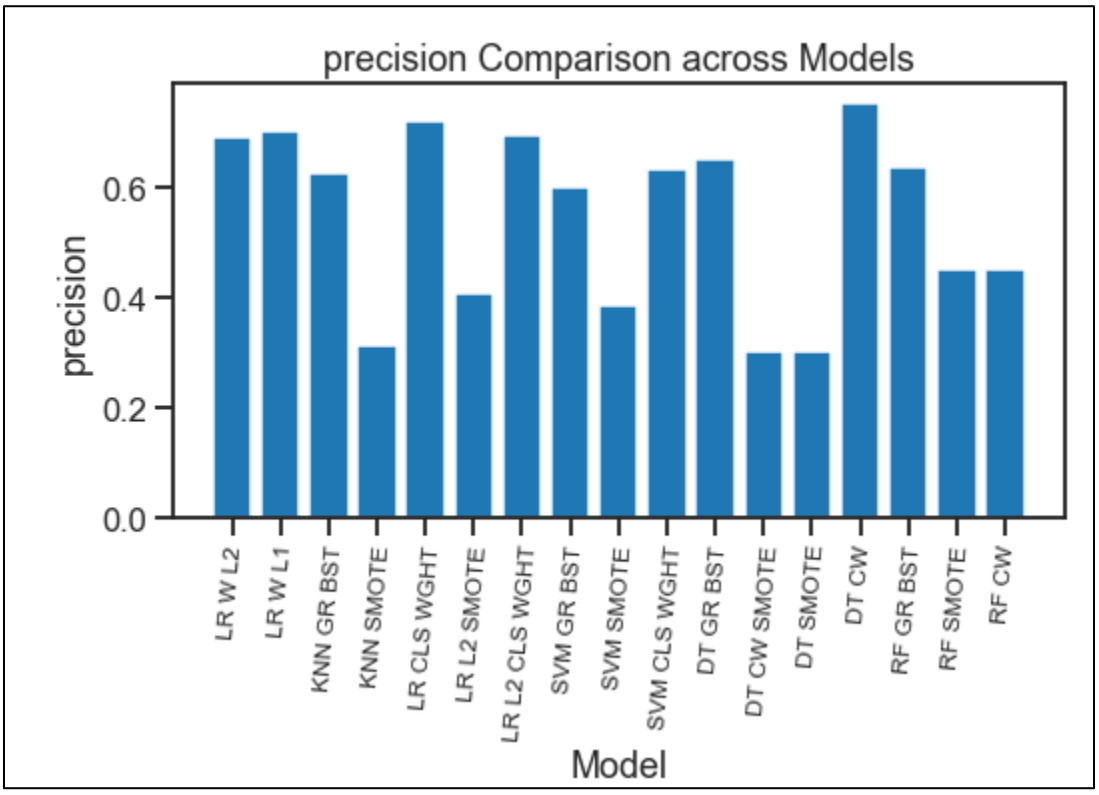
Below are the metrics comparison for the overall models:

1. Accuracy:



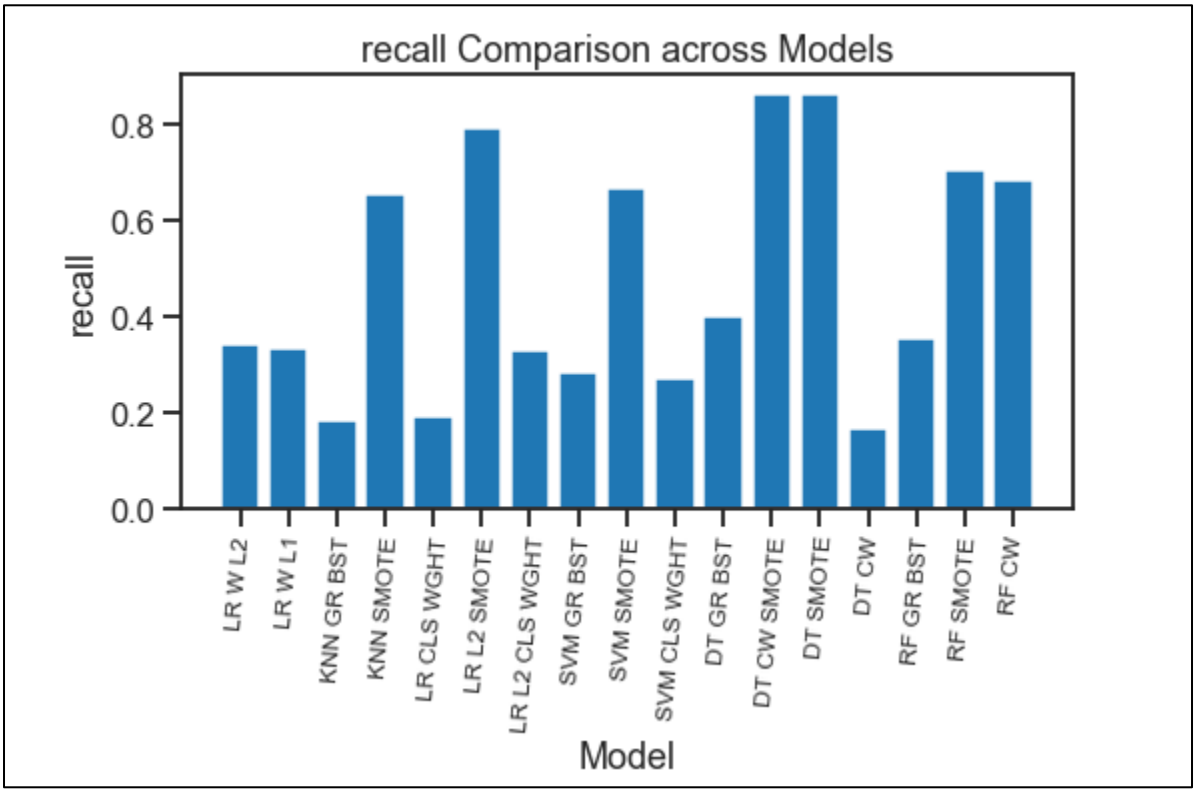
In terms of accuracy most of the models performed close to each other, best being with decision tree with 90.87% and worst model was decision tree with smote around 76%

2. Precision:



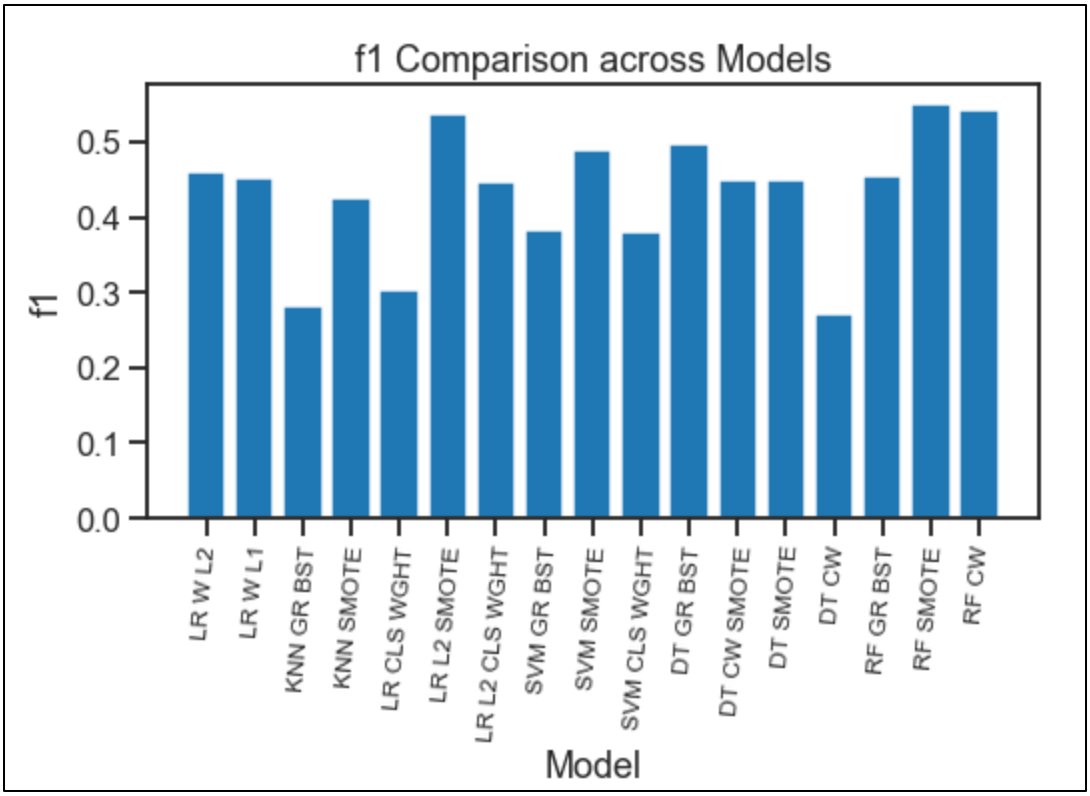
As we discussed in the modeling phase that since the data is highly imbalanced it is affecting the model precision and accuracy, We get the best precision with Class wights in Decision tree of 75% and the lowest being with smote at ~ 35%

3. Recall –



Similar to what we saw in Precision in Recall as well we see variation in model performances, with the highest being with Decision trees Smote and lowest with Decision tree using class weights.

4. F1 score –



F1 Score was the best with Random Forest Grid search.

Overall, it can be said that our models are not performing up to the mark with the model tunings we have done, but overall decision trees are doing a good job.

If the Portuguese bank wants to go for as much acquisition as possible with the **acquisition cost not being a problem** in that case, I will go with the model with the highest recall which is **decision tree with SMOTE**. Since with this model we will be able to identify most of the customers who would want to go for a long-term direct deposit, without worrying about the false positives that is targeting customers who would not want to go for long term deposits.

Whereas if the bank wants to make sure that it wants to be as effective **as possible with the budget** and **reduce** the amount of **false positive** in that case, I will be going with **Decision tree with class weight**.

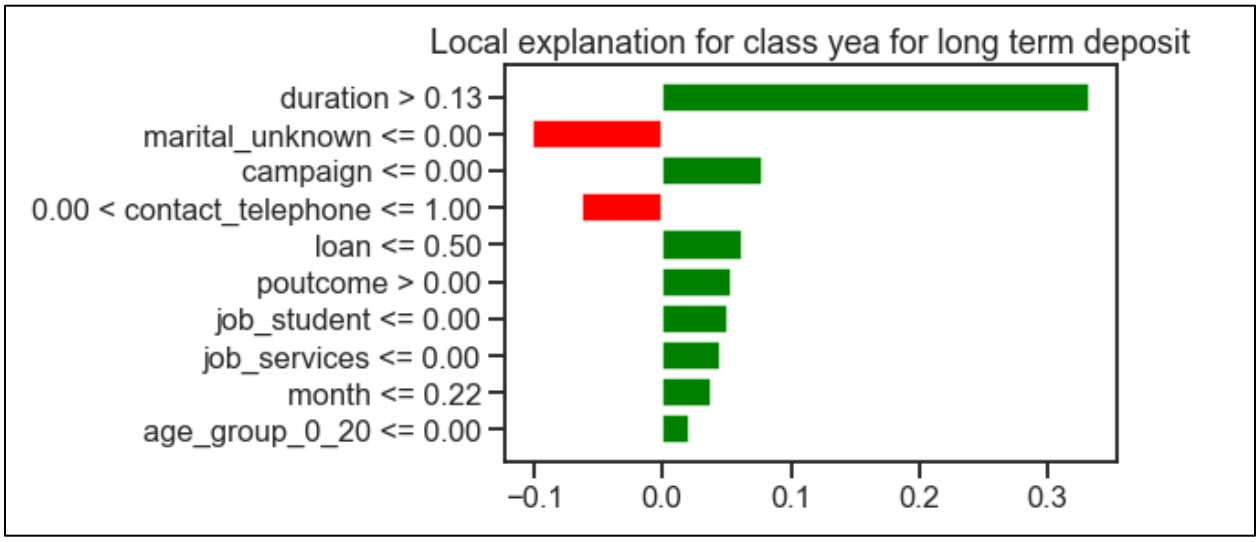
Next Steps:

As observed our models require more tuning, I would want to look more into decision trees and also look into if boosting helps our case. Since most of the features where categorical variables therefore ensemble models would be a better fit for our use case.

Also, our dataset is highly imbalanced therefore maybe looking into some more sampling methods might help us, here I have only worked with oversampling, but I would like to also look in under sampling and see if that helps.

I have not done much feature elimination before fitting my model since most of the features were showing correlation with the target variable, but I might want to look if feature elimination increases my prediction metrics.

Along with Improving the metrics I would also want to include some extra information that might help converting the customers. I.E., understand which of the metrics are the driving factors for a customer to be willing. For examples the below customer who is willing using lime we can tell what were the features that were positive for the customer



Duration, campaign, loan, are some of the features that are positively driven for this customer where as marital status, contact are some of the negative drivers.