DELIVERABLE WEEK 10

Group Name: The Powerpuff Girls

Specialization: Data Science

Team Members:

1. **Name:** Memudu Alimatou Sadia Anike

    Email: anikesadia01@gmail.com

    Country: Nigeria

    College: University of Ilorin

    Specialization: Data science

2. **Name:** Chaithanya Shivakumar Ittamadu

    Email: sichaithanya889@gmail.com

    Country: Ireland

    College: Dublin Business School

    Specialization: Data science

3. **Name:** Lakshmi Chandana Vupputuri

    Email: vupputuri.chandana@gmail.com

    Country: Ireland

    Specialization: Data Science

Problem description

ABC Bank wants to sell it's term deposit product to customers and before launching the product they want to develop a model which help them in understanding whether a particular customer will buy their product or not (based on customer's past interaction with bank or other Financial Institution).

Here we are using Exploratory Data analysis (EDA) to analyze the data, extract meaningful insights from it to make a good business decision.

GitHub Repo link

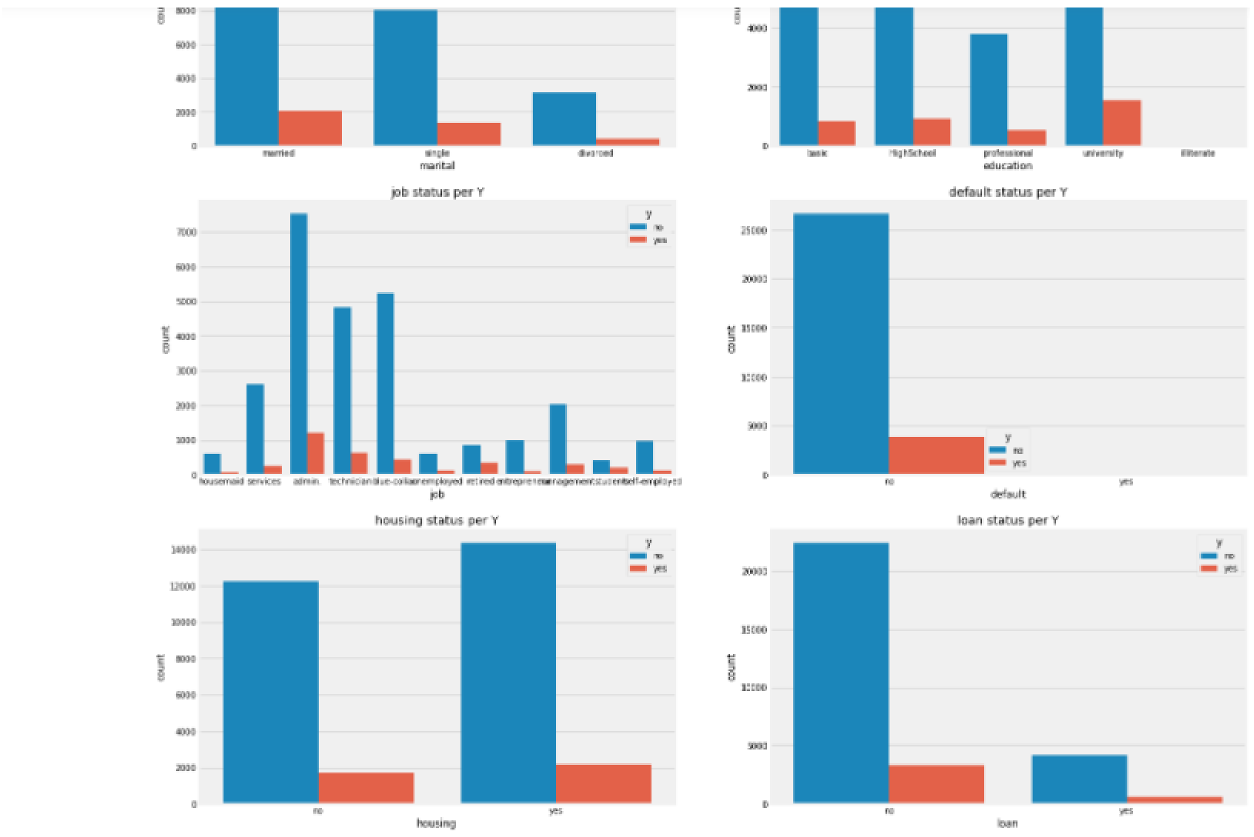
Group repos: **https://github.com/memudualimatou/Bank-DataScienceProject**

Project EDA: **https://github.com/memudualimatou/Bank-DataScienceProject/blob/main/BankMarketingEDA.ipynb**

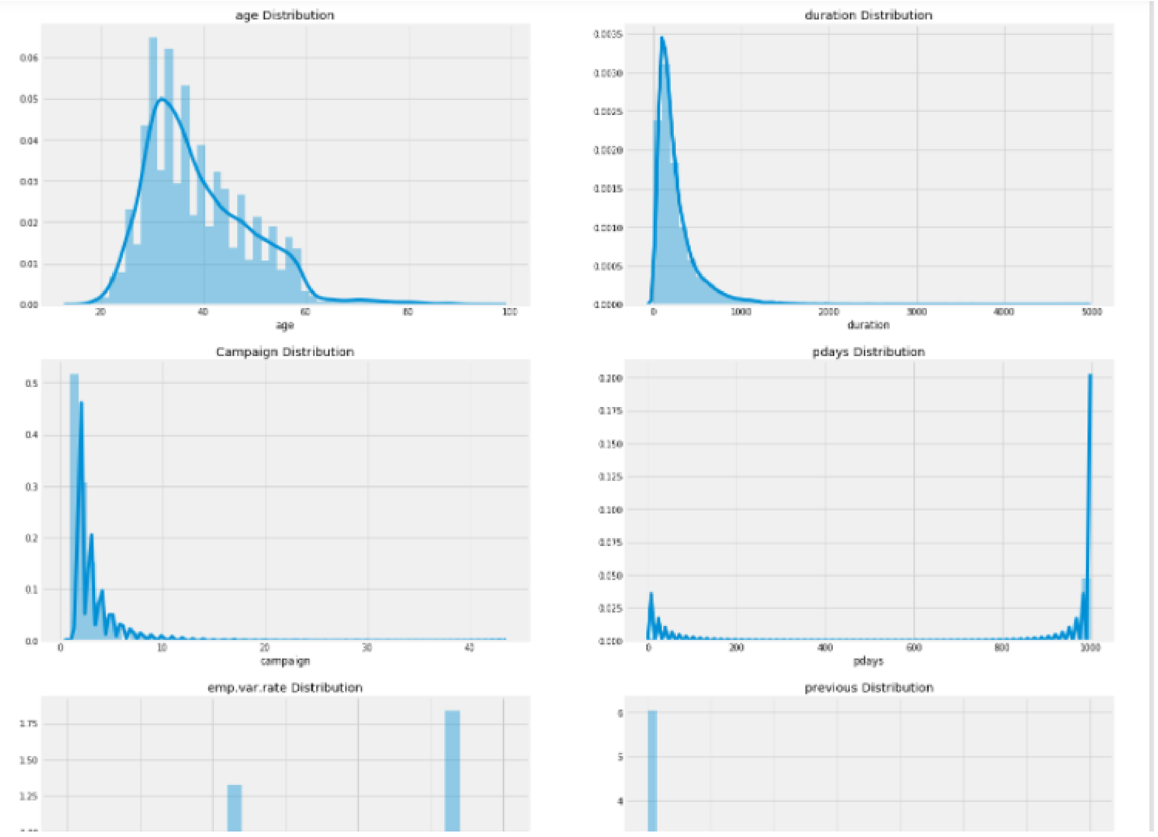
EDA Performed on the Data

Data visualization

Visualizing the categorical values distribution in terms of our target value y.

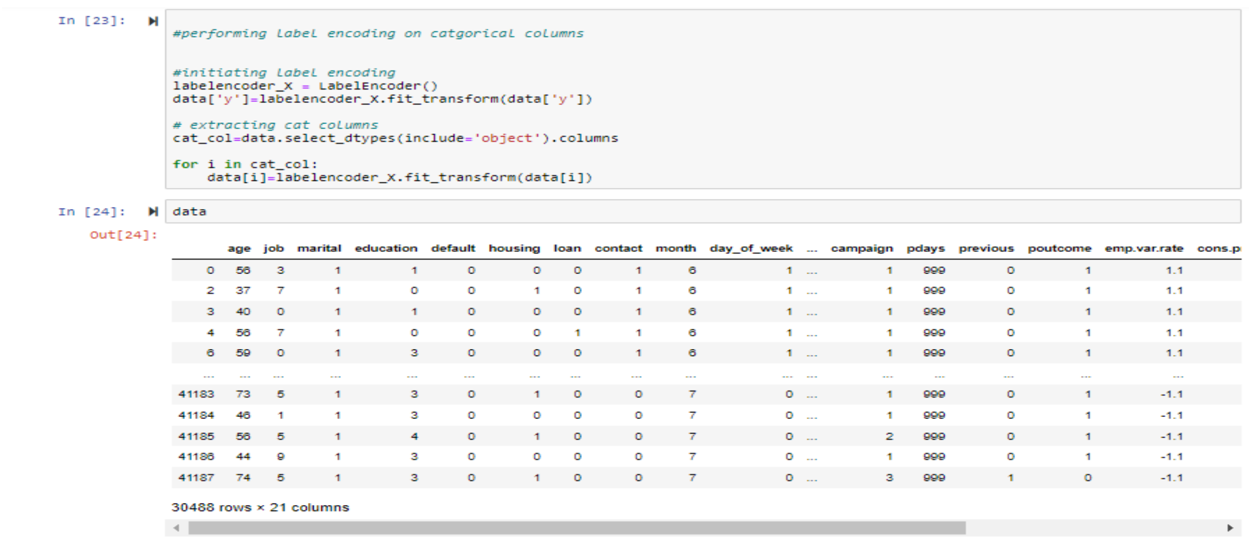


Each Numerical variables distribution



LabelEncoding:

Our data contains many categorical variables such as job, education etc, using this method to transform each column values into a range of number is the best option as shown in the image below.

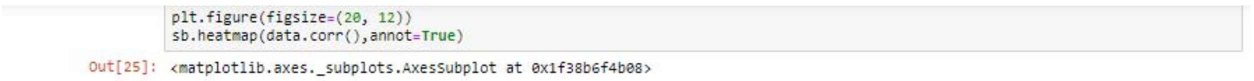


```
In [23]:  #performing label encoding on catgorical columns

          #initiating label encoding
          labelencoder_X = LabelEncoder()
          data['y']=labelencoder_X.fit_transform(data['y'])

          # extracting cat columns
          cat_col=data.select_dtypes(include='object').columns

          for i in cat_col:
              data[i]=labelencoder_X.fit_transform(data[i])
```

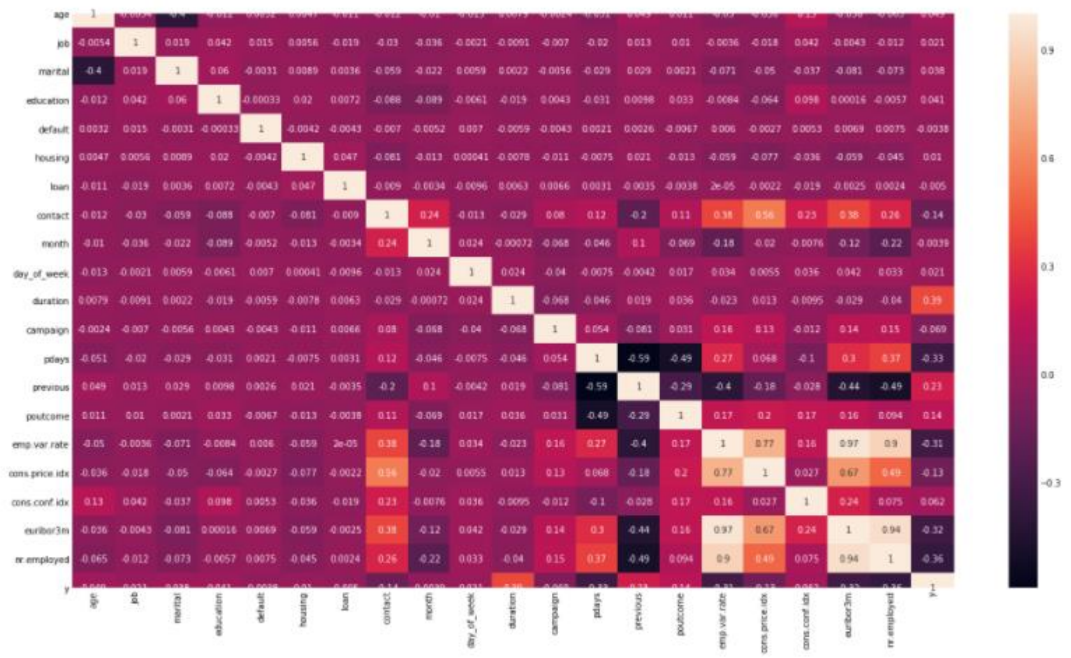
```
In [24]:  data
Out[24]:
        age  job  marital  education  default  housing  loan  contact  month  day_of_week  ...  campaign  pdays  previous  poutcome  emp.var.rate  cons.p
     0   56    3        1          1        0        0     0        1      6            1  ...         1    999         0         1           1.1
     2   37    7        1          0        0        1     0        1      6            1  ...         1    999         0         1           1.1
     3   40    0        1          1        0        0     0        1      6            1  ...         1    999         0         1           1.1
     4   56    7        1          0        0        0     1        1      6            1  ...         1    999         0         1           1.1
     6   59    0        1          3        0        0     0        1      6            1  ...         1    999         0         1           1.1
   ...  ...  ...      ...        ...      ...      ...   ...      ...    ...          ...  ...       ...    ...       ...       ...           ...
 41183   73    5        1          3        0        1     0        0      7            0  ...         1    999         0         1          -1.1
 41184   46    1        1          3        0        0     0        0      7            0  ...         1    999         0         1          -1.1
 41185   56    5        1          4        0        1     0        0      7            0  ...         2    999         0         1          -1.1
 41186   44    9        1          3        0        0     0        0      7            0  ...         1    999         0         1          -1.1
 41187   74    5        1          3        0        1     0        0      7            0  ...         3    999         1         0          -1.1

30488 rows × 21 columns
```

Target Data Correlation

It is important to visualize the most correlated value to our target y.

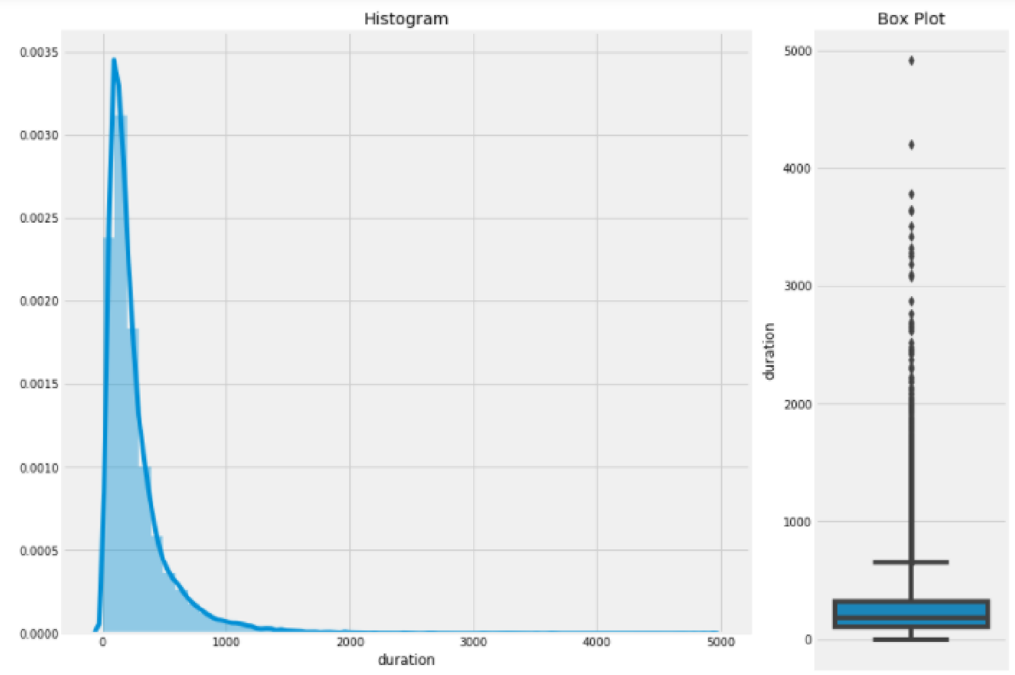


```
plt.figure(figsize=(20, 12))
sb.heatmap(data.corr(),annot=True)

Out[25]: <matplotlib.axes._subplots.AxesSubplot at 0x1f38b6f4b08>
```



As a result Duration is the most correlated column to our target. Previously we found out that this column is highly skewed. Transforming can ameliorate our model performance.

Let's visualize his distribution and check for outliers in the columns

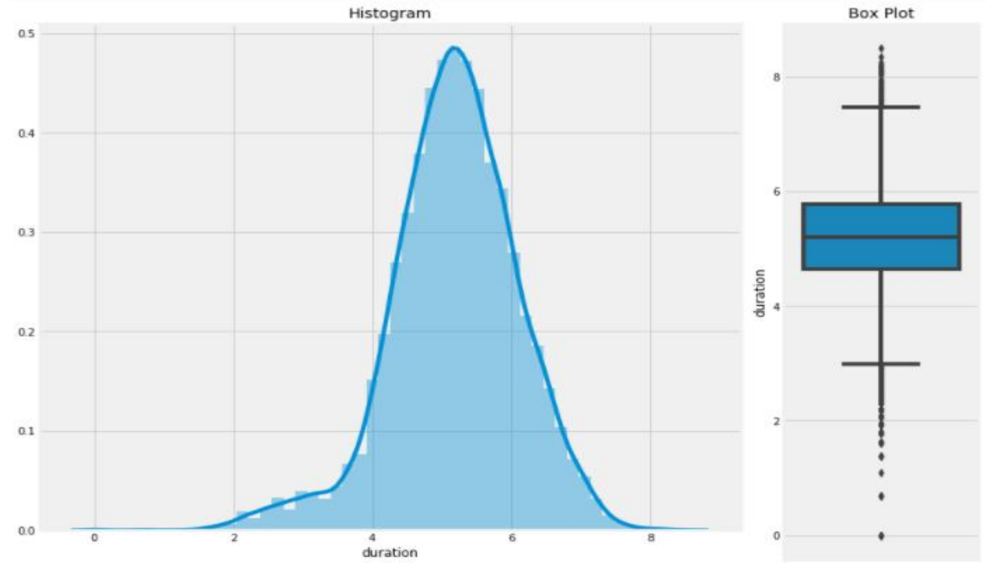


the duration distribution is righ skewed and has a lot of outliers

Duration is right-skewed, doesn't have a normal distribution and has a lot of outliers as shown in the boxplot.

One of the most effective way of dealing with a skewed column is to use numpy.log1p it will help the column have a normal distribution, reduce the number of outliers and the skewed value.

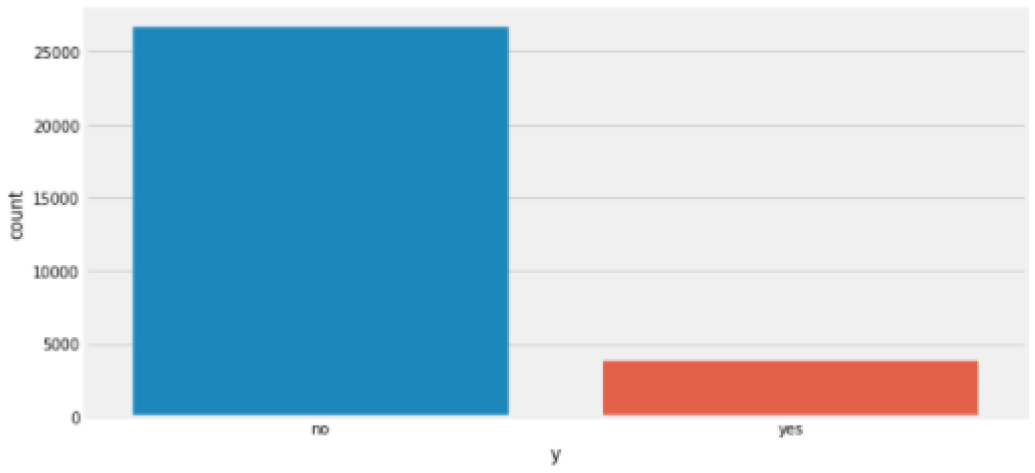
```
In [29]:  ## transforming duration variable using numpy.log1p,
          data["duration"] = np.log1p(data["duration"])

          ## Plotting the newly transformed response variable
          plotting_2_chart(data,"duration")
```



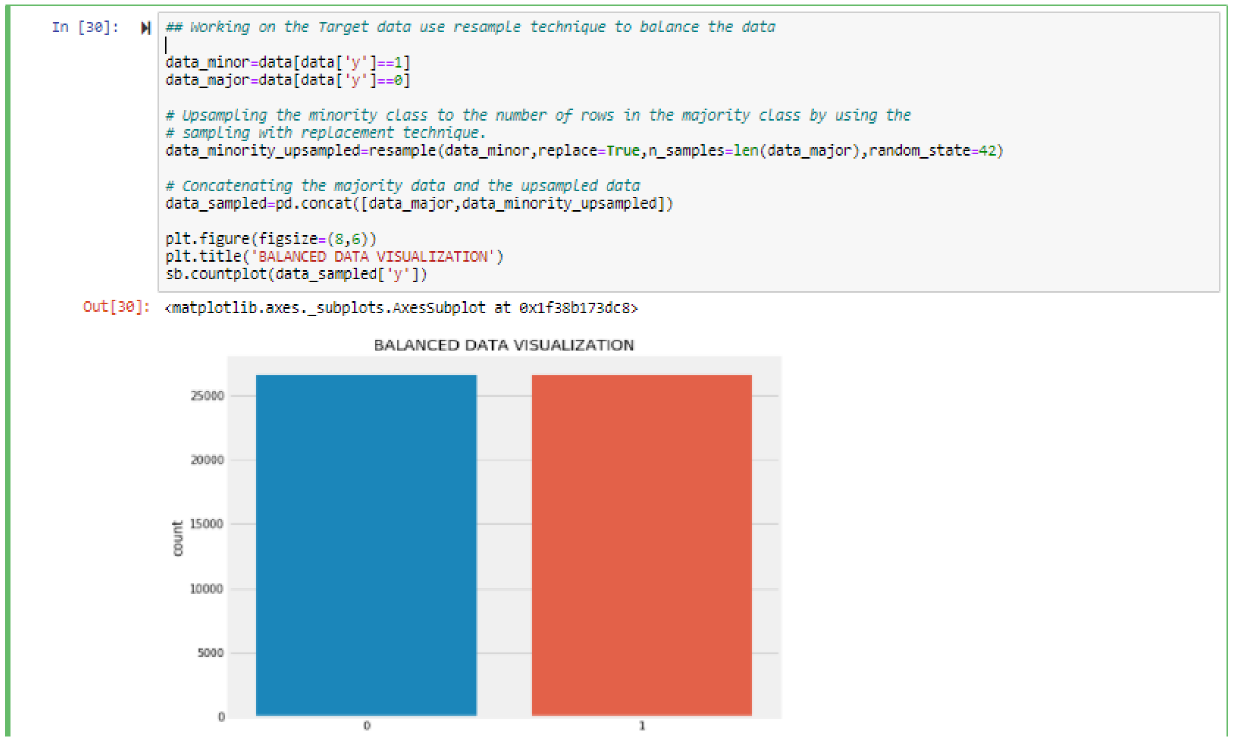
Now the most correlated value to our target data is normal.

Balancing the Data

As known our data is very imbalanced as illustrated below. This imbalance data may affect the performance of our model. Using techniques such as Oversampling may help.



After oversampling the data this is the result.



```
In [30]:  ## Working on the Target data use resample technique to balance the data

          data_minor=data[data['y']==1]
          data_major=data[data['y']==0]

          # Upsampling the minority class to the number of rows in the majority class by using the
          # sampling with replacement technique.
          data_minority_upsampled=resample(data_minor,replace=True,n_samples=len(data_major),random_state=42)

          # Concatenating the majority data and the upsampled data
          data_sampled=pd.concat([data_major,data_minority_upsampled])

          plt.figure(figsize=(8,6))
          plt.title('BALANCED DATA VISUALIZATION')
          sb.countplot(data_sampled['y'])

Out[30]:  <matplotlib.axes._subplots.AxesSubplot at 0x1f38b173dc8>
```

As the target data is balanced, it will ameliorate the model due to the equal number of value in y.

Final Recommendations

The bank should focus on the following to gain more customer that make deposit.

- Increase the amount of contact with customers
- Focus on students, retired and unemployed audience
- Pay attention to only educated individuals
- Use cellular campaigns more than telephonic ones.
- Limit the number of contacts per customer to 6.