



Data Science Intern at Data Glacier

Week 9: Report on Group Project

Topic: Bank Marketing (Campaign)

Group Name: Campaign Catalysts

Specialization: Data Science

Batch Code: LISUM19

Date: 02nd May 2023

Submitted to: Data Glacier

Group Member Details

S.No.	Name	Email	Country	College/ Company
1.	Yash Jayesh Doshi	yashjdoshi99@gmail.com	UAE	Orpheuss LLC
2.	Anuj Singh	dsanuj21@gmail.com	India	Mumbai University
3.	Yash Jadwani	yash.jadwani1998@gmail.com	United Kingdom	Kingston University
4.	Harold Wilson	haroldwilson537@gmail.com	United Kingdom	University of Buckingham

1. Problem Description

ABC Bank wants to sell its 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).

Bank wants to use ML model to shortlist customer whose chances of buying the product is more so that their marketing channel (tele marketing, SMS/email marketing etc.) can focus only to those customers whose chances of buying the product is more.

2. Data Cleaning of Numerical Features (Yash Doshi and Anuj Singh)

As discussed in the previous week's report, going forward, we will be using the bank-additional-full.csv for this project.

Basic data exploration revealed that the dataset has 10 numerical features. In the data exploration performed in the last week, it was found that some of these features have outliers and some have high correlation amongst themselves. These issues need to be dealt with at this stage because these issues can significantly affect the accuracy and performance of the machine learning model that will be created later.

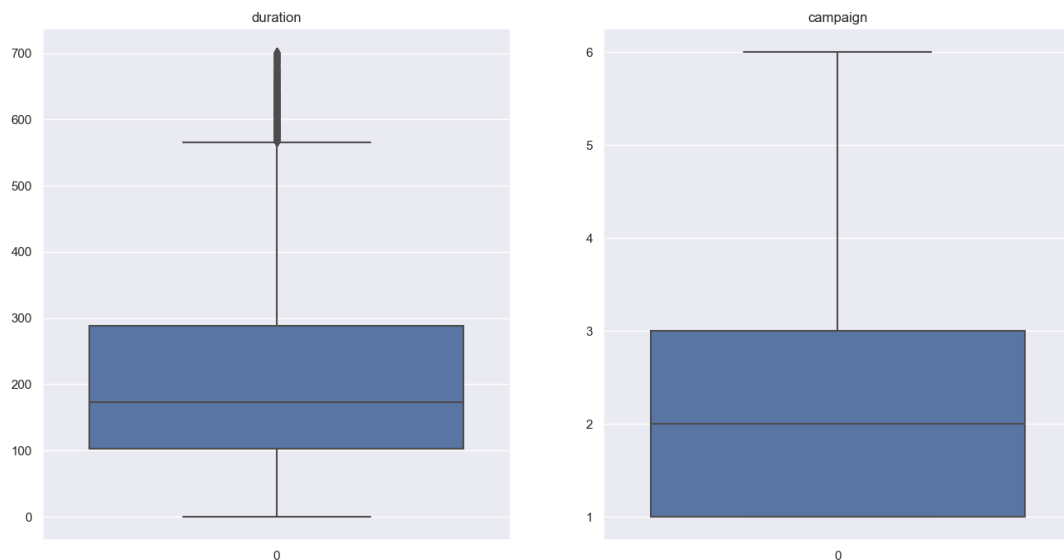
2.1 Dealing with outliers (Anuj Singh)

Checking the boxplots and the distribution plots of the numerical features revealed that three numerical features had a large number of outliers, namely, 'duration', 'pdays', and 'campaign'.

The outliers in the 'duration' and 'campaign' column were removed by only keeping the values less than the 95th percentile value for these features.

```
#Dropping values above 95 percentile in duration and campaign
duration_q95 = df_copy1['duration'].quantile(0.95)
campaign_q95 = df_copy1['campaign'].quantile(0.95)
# filter out values above the 95th percentile range
df_copy1 = df_copy1.loc[(df_copy1['duration'] <= duration_q95) & (df_copy1['campaign'] <= campaign_q95)]
```

The boxplots below reveal that majority of the outliers were successfully removed

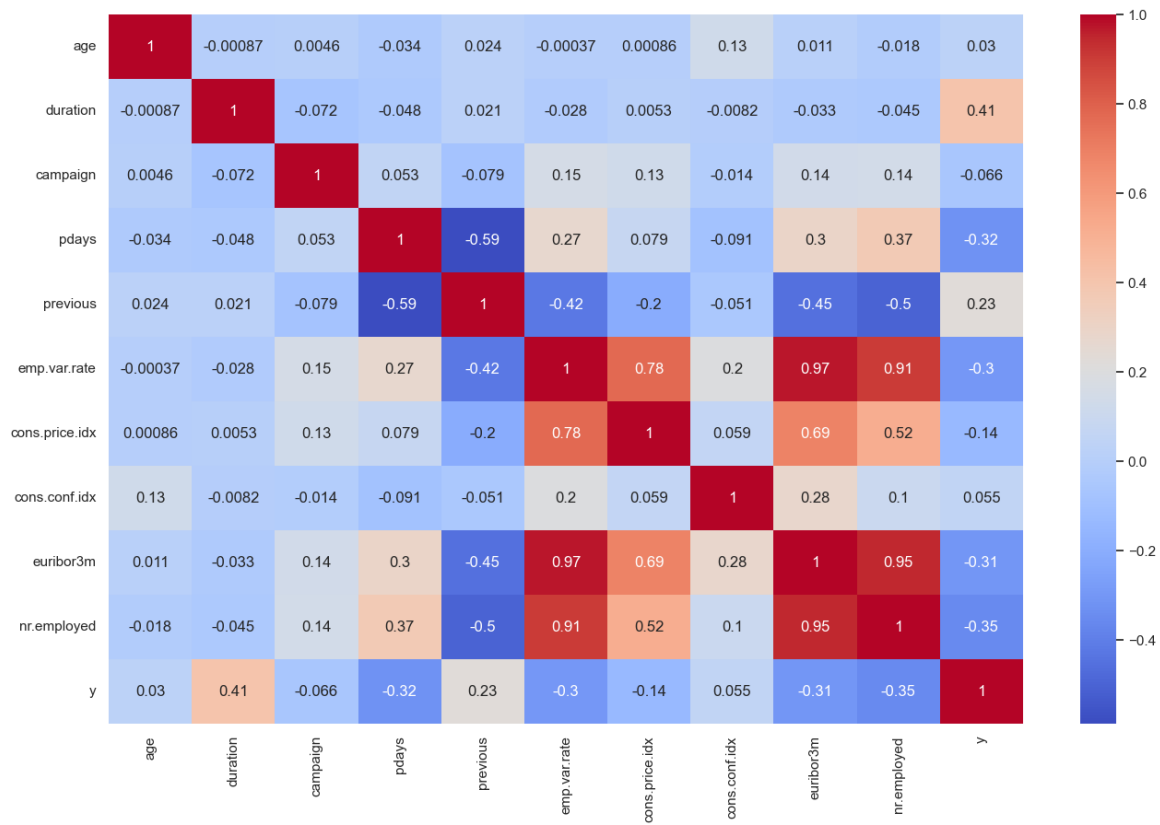


For the 'pdays' column, it was found that the outliers were due to 999 values, which were placeholders for missing data. Since more than 95% of the values are 999, we will drop the 'pdays' column entirely.

```
#Removing pdays column since too many features have missing data: 999
df_copy1.drop(['pdays'],axis=1,inplace=True)
df_copy1
```

2.2 Dealing with highly correlated features (Yash Doshi)

The heatmap of the Pearson's correlation matrix of the dataset shows that 'euribor3m', 'emp.var.rate', and 'nr.employed' are highly correlated. Note that the below figure also includes the target variable 'y', which was a categorical feature in the original dataset but was converted to a numerical feature using Label Encoding to study its correlation with the other numerical features.



We will be performing Principle Component Analysis (PCA) on these highly correlated features to reduce them into one or two transformed features.

The below Scree plot shows that one principle component explains more than 90% of the variance in the data and hence, the three highly correlated features can be transformed into one feature without losing the information of the three features.

```

# Using PCA to reduce these features into one or two features, hence reducing multi-collinearity
from sklearn.decomposition import PCA

# Standardizing the values in the features
df_subset = df_copy1[high_corr_features]
df_subset = (df_subset - df_subset.mean()) / df_subset.std()

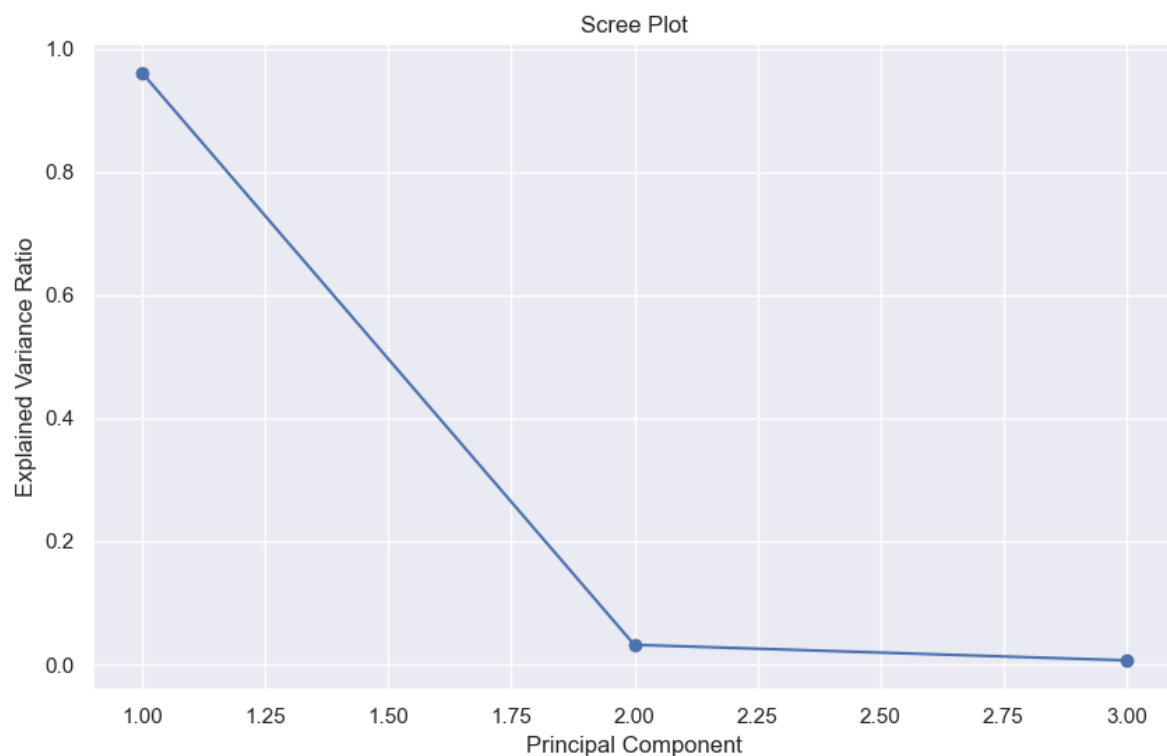
# Perform PCA with 3 components
pca = PCA(n_components=3)
pca.fit(df_subset)

# Get explained variance ratio
variance = pca.explained_variance_ratio_

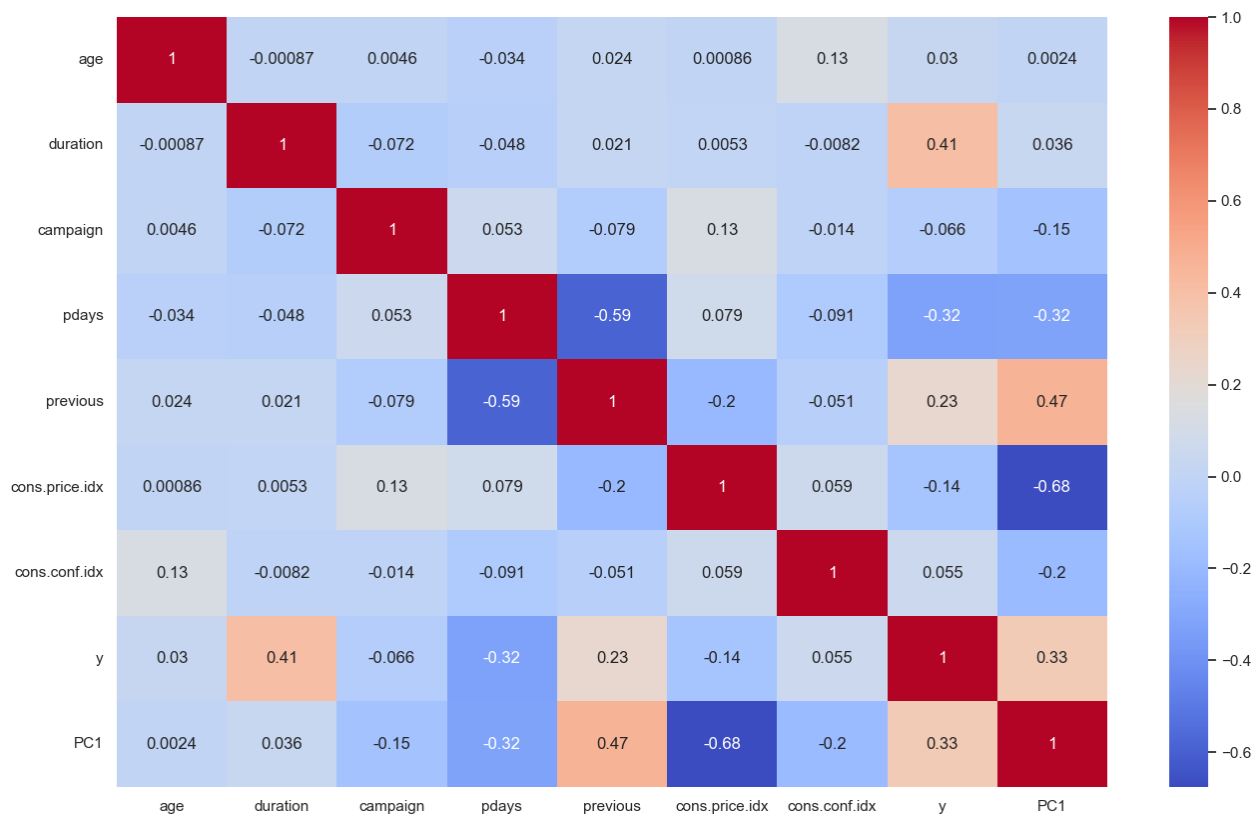
# Plot scree plot
fig = plt.figure(figsize=(10,6))
fig.add_subplot(1,1,1)
plt.plot(range(1, len(variance) + 1), variance, marker='o')
plt.xlabel('Principal Component')
plt.ylabel('Explained Variance Ratio')
plt.title('Scree Plot')

plt.savefig('screeplot_week9.png')

```



After transforming the three highly correlated features into one feature using PCA, the below image shows the heatmap of the Pearson's Correlation matrix of the updated dataframe. It is evident from the similar correlation value between the new feature PC1 and y that the correlation problem has been solved without losing the information contained in the features.



3. Data Cleaning of Categorical Features (Harold Wilson and Yash Jadwani)

Handling missing data for Bank marketing dataset

Data cleaning is an important step in data pre-processing due to its ability to help improve the quality of the dataset for a more reliable output. The presence of impurities in real-world data application has brought about the development of several methods to eradicate this problem to help improve the accuracy and usability of existing data (Müller and Freytag, 2005). [3] The data cleaning process involves the detection or removal of outliers, smoothing noisy data, filling in missing values and resolving inconsistency within a dataset (Han, Pei and Kamber, 2011). [4]

There is exactly no one way of dealing with missing data. There are different solutions for data imputation depending on the kind of problem and it always difficult to provide a general solution, and care should be taken when it comes to removing missing values in any given data set since doing so will introduce biasness in the model.

Imputing or Deleting missing values of the Data:

Before we decide to remove, replace or impute the data, we have to understand and establish the reason why data is missing.

- **Missing at Random:** This means that the tendency for a data point to be missing is not related to the missing data, but it is related to some of the observed data.
- **Missing Completely at Random:** The fact that a certain value is missing has nothing to do with its hypothetical value and with the values of other variables.

- Missing not at Random: Possible reasons are that the missing value depends on the hypothetical value or missing value is dependent on some other variable's value.

Data cleaning of categorical features in the data set:

We have the following unknown values for some of the features in the data set:

Features	Unknown Values	Minimum value	Maximum value
Job	990	unknown	admin
Marital	80	unknown	married
Education	1731	illiterate	university degree
Default	8597	yes	no
Housing	990	unknown	yes
Loan	990	unknown	no
contact	0	nil	nil
month	0	nil	nil
day_of_week	0	nil	nil
poutcome	35563	success	nonexistence
y	0	yes	no

Mode Imputation for Unknown/Missing values

a) Deleting Duplicate values

```
## Deleteing duplicate entries
print('Duplicate entries in the dataset :',bank_additional_data.duplicated().sum())
bank_additional_data.drop_duplicates(inplace=True)
print('Duplicate entries in the dataset After Deletion :',bank_additional_data.duplicated().sum())
```

✓ 0.2s

Duplicate entries in the dataset : 12
Duplicate entries in the dataset After Deletion : 0

b) Replacing Unknown/missing values with N/A and Checking for null values


```

bank_additional_data.isnull().sum()
✓ 0.0s
age          0
job         330
marital      80
education   1730
default     8596
housing      990
loan        990
contact      0
month        0
day_of_week  0
duration     0
campaign     0
pdays       0
previous     0
poutcome     0
emp.var.rate 0
cons.price.idx 0
cons.conf.idx 0
euribor3m    0
nr.employed  0
y            0
dtype: int64

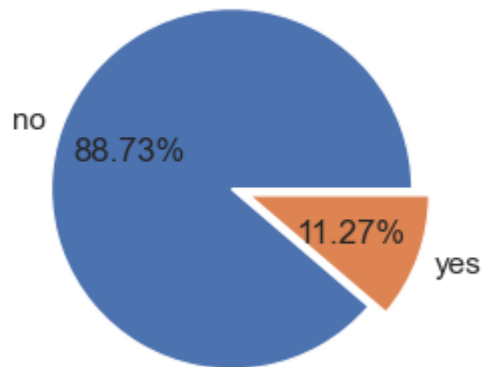
```

c) Mode Imputation Steps

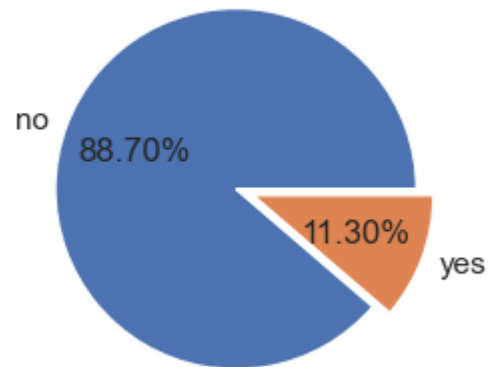
- We removed all the rows where job, default and housing are null.
- We removed all the rows where job and education are null.
- We encountered 75 missing values in the marital status variable and addressed this issue by creating an age marital map to impute these values.
- In the job variable, we found 197 instances of missing data, which we resolved by generating an age education job map and using it to impute the missing values.
- We found approximately 1600 instances of missing education records, which we addressed by generating a job education map and using it to impute the missing values.
- We removed all the rows where loan, default and housing are null.
- There was highly imbalance in default feature, only 3 individuals had defaulted. This suggests that the vast majority of individuals in the dataset have not defaulted on their payments. Therefore, we removed the default feature from the analysis as it may not be useful in making any meaningful conclusions
- We identified 763 missing values in both the loan and housing variables, which we imputed using information from the marital status, job, and education variables.

d) Target vector share before and after Imputation

Target vector before imputation



Target vector after imputation



GitHub Repo Link:

<https://github.com/singhanuj695/Data-glacier-Group-Project>