Week 8: Final Project

Bank Marketing Campaign

- Data Science -

Table of Contents

Team member's details:

	Group Name: *Data Science Enthusiasts*				
	Name	**Email**	**Country**	**College/Company**	**Specialization**
1	Amira Asta	amira.asta02@gmail.com	Tunisia	Afrikanda	Data Science
2	Vatsal Vinesh Mandalia	vatsalvm10@outlook.com	Oman	Graduated	Data Science

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 helps them understand whether a particular customer will buy their product or not. In order to achieve this task, they approached an Analytics company to automate this process of classification. The Analytics company has given responsibility to the **Data Science Enthusiasts** Team and has asked to come up with a ML model to shortlist customers whose chance of buying the product is higher, so that ABC's marketing channel can focus only on those customers.

Business understanding:

There has been a revenue decline for an ABC bank and they would like to know what actions to take. After investigation, they found out that the root cause is that their clients are not depositing as frequently as before. Knowing that term deposits allow banks to hold onto a deposit for a specific amount of time, banks can invest in higher gain financial products to make a profit.

In addition, banks also hold better chances to persuade term deposit clients into buying other products such as funds or insurance to further increase their revenues. As a result, the ABC bank would like to identify existing clients that have higher chances to subscribe for a term deposit and focus marketing efforts on such clients. The classification goal is to predict if the client will subscribe to a term deposit or not.

Data Understanding

The data corresponds 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 ('yes') or not ('no') subscribed.

There are four datasets provided for this classification problem. Among the four datasets, there are two pairs of train and test data available for analysis. The ‘bank-full.csv’ and ‘bank.csv’ are one of the pairs having less than 20 input features and are an older version of ‘bank-additional-full.csv’ and ‘bank-additional.csv’. The information of the datasets is given below.

	Dataset type	Description
bank-additional-full.csv	train	41118 observations and 20 inputs ordered by date (from May 2008 to November 2010)
bank-additional.csv	test	4118 observations (10% of train data) with 20 inputs
bank-full.csv	train	45211 observations and 17 inputs ordered by date (older version of bank-additional-full)
bank.csv	test	4521 observations (10% of train data) and 17 inputs

The description of the 20 input features is given below.

Table 1: Dataset input features.

N°	Feature name	Description
1	age	age
2	job	type of job
3	marital	marital status
4	education	level of education
5	default	has credit in default?
6	housing	has housing loan?
7	loan	has personal loan?
8	contact	contact communication type
9	month	last contact month of year
10	day of week	last contact day of the week
11	duration	last contact duration, in seconds
12	campaign	number of contacts performed in this campaign
13	pdays	number of days passed by after the last contact
14	previous	number of contacts performed for this client
15	poutcome	outcome of the previous marketing campaign
16	emp.var.rate	employment variation rate
17	cons.price.idx	consumer price index - monthly indicator
18	cons.conf.idx	consumer confidence index - monthly indicator
19	euribor3m	euribor 3 month rate - daily indicator
20	nr.employed	number of employees - quarterly indicator
21	y	has the client subscribed a term deposit?

The first 7 features are Bank client data.

Features 8, 9, 10 and 11 are related to the last contact of the current campaign.

Features 12, 13, 14 and 15 are other attributes.

Features 16, 17, 18, 19 and 20 are social and economic context attributes.

The last feature is the output feature (desired target).

Data Types

In this dataset the features that we described above, are divided between “object” types i.e. categorical attributes and “int64 / float64” types i.e. numerical attributes.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 41188 entries, 0 to 41187
Data columns (total 21 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   age             41188 non-null  int64
 1   job             41188 non-null  object
 2   marital         41188 non-null  object
 3   education       41188 non-null  object
 4   default         41188 non-null  object
 5   housing         41188 non-null  object
 6   loan            41188 non-null  object
 7   contact         41188 non-null  object
 8   month           41188 non-null  object
 9   day_of_week     41188 non-null  object
 10  duration        41188 non-null  int64
 11  campaign        41188 non-null  int64
 12  pdays           41188 non-null  int64
 13  previous        41188 non-null  int64
 14  poutcome        41188 non-null  object
 15  emp.var.rate    41188 non-null  float64
 16  cons.price.idx  41188 non-null  float64
 17  cons.conf.idx   41188 non-null  float64
 18  euribor3m       41188 non-null  float64
 19  nr.employed     41188 non-null  float64
 20  y               41188 non-null  object
dtypes: float64(5), int64(5), object(11)
memory usage: 6.6+ MB
```

Data Problems

Missing Attribute Values:

All 4 datasets have no missing values.

bank.csv

```
Entrée [18]: data1.isnull().sum()

Out[18]: age          0
         job          0
         marital      0
         education    0
         default      0
         balance      0
         housing      0
         loan         0
         contact      0
         day          0
         month        0
         duration     0
         campaign     0
         pdays        0
         previous     0
         poutcome     0
         y            0
         dtype: int64
```

bank-full.csv

```
Entrée [19]: data2.isnull().sum()

Out[19]: age          0
         job          0
         marital      0
         education    0
         default      0
         balance      0
         housing      0
         loan         0
         contact      0
         day          0
         month        0
         duration     0
         campaign     0
         pdays        0
         previous     0
         poutcome     0
         y            0
         dtype: int64
```

bank-additional.csv

```
Entrée [20]: data3.isnull().sum()

Out[20]: age             0
         job             0
         marital         0
         education       0
         default         0
         housing         0
         loan            0
         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
```

bank-additional-full.csv

```
Entrée [15]: data4.isnull().sum()

Out[15]: age             0
         job             0
         marital         0
         education       0
         default         0
         housing         0
         loan            0
         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
```

Note: There are a significant number of observations/rows with a value 'unknown' for the majority of the categorical features in the four datasets. So, we assume the value 'unknown' as another category for these variables in our analysis.

```
admin.           10422
blue-collar       9254
technician        6743
services          3969
management        2924
retired           1720
entrepreneur      1456
self-employed     1421
housemaid         1060
unemployed        1014
student            875
unknown            330
Name: job, dtype: int64
```

```
married     24928
single      11568
divorced     4612
unknown        80
Name: marital, dtype: int64

university.degree      12168
high.school             9515
basic.9y                6045
professional.course     5243
basic.4y                4176
basic.6y                2292
unknown                 1731
illiterate                18
Name: education, dtype: int64
```

```
no         32588
unknown     8597
yes            3
Name: default, dtype: int64
yes        21576
no         18622
unknown      990
Name: housing, dtype: int64
no         33950
yes         6248
unknown      990
Name: loan, dtype: int64
```

Outliers and skewness in the features to be mentioned.

Duplicate rows:

- There are 12 rows whose duplicates are present in bank-additional-full data. The .drop_duplicates() method is used to drop the duplicate rows.

```
In [6]: # To check for duplicate rows
        # bank-additional-full and bank-additional data

        print(bk_add_full.duplicated().sum())
        # print(bk_add_full[bk_add_full.duplicated()])
        print('There are 12 rows whose duplicates are also present in bank-additional-full data')

        bk_add_full.drop_duplicates(keep = 'first', inplace = True)
```

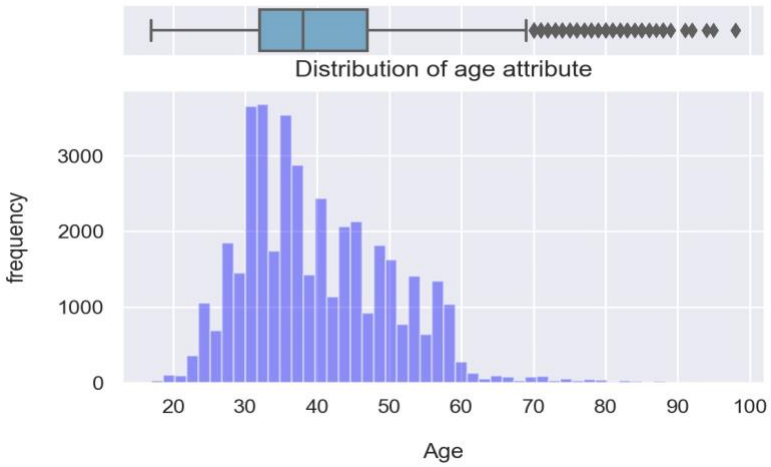
```
12
There are 12 rows whose duplicates are also present in bank-additional-full data
```

- In bank-additional, bank-full and bank.csv datasets, there are no duplicate rows.

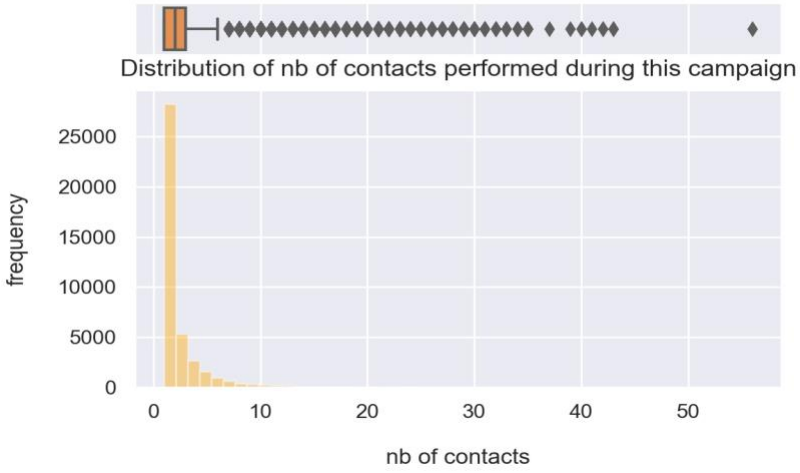
Outliers:

An outlier is a value/observation which lies at an abnormal distance from other values in the normal distribution. It can occur due to an error in measurement or data collection. The following features of bank-additional-full have significant outliers. Outliers can affect the mean of the distribution.

- **'age'** attribute:



- **'campaign'** attribute:



In our case, we don't need to remove outliers from the data since the max('age')=98 and max('campaign')=56 are not unrealistic values. This will help with the generalization of the model later since it should reflect the real world.

Skewness & Kurtosis:

The skew result shows a positive (right) or negative (left) skew. Values closer to zero show less skew. Skewness is a measure of asymmetry of the distribution relative to the normal distribution. Positive skewness implies the tail is in the right of the mean of the distribution. Negative skewness implies the tail is in the left of the mean of the distribution.

Kurtosis is the measure of whether or not a distribution has a heavy tail or not relative to the normal distribution. A value >3 means the distribution has a heavy tail. Kurtosis <3 implies the distribution has a lighter tail than the normal distribution.

```
# skewness along the index axis
bank_additional_full.skew(axis = 0, skipna = True)
```

```
age              0.784697
duration         3.263141
campaign         4.762507
pdays           -4.922190
previous         3.832042
emp.var.rate    -0.724096
cons.price.idx  -0.230888
cons.conf.idx    0.303180
euribor3m       -0.709188
nr.employed     -1.044262
dtype: float64
```

```
# Kurtosis along the index axis
bank_additional_full.kurt(axis=0)
```

```
age               0.791312
duration         20.247938
campaign         36.979795
pdays            22.229463
previous         20.108816
emp.var.rate     -1.062632
cons.price.idx   -0.829809
cons.conf.idx    -0.358558
euribor3m        -1.406803
nr.employed      -0.003760
dtype: float64
```

Github Repo link:

https://github.com/AsAmira02/Bank-Marketing-Campaign-DSEnthusiasts2021

This repository includes the four datasets, model code and necessary files used in this project.