**Market Analysis in Banking Domain Project**

**DESCRIPTION**

**Background and Objective:**

Your client, a Portuguese banking institution, ran a marketing campaign to convince potential customers to invest in a bank term deposit scheme.   
The marketing campaigns were based on phone calls. Often, the same customer was contacted more than once through phone, in order to assess if they would want to subscribe to the bank term deposit or not. You have to perform the marketing analysis of the data generated by this campaign.

**Domain**: Banking (Market Analysis)

**Analysis tasks to be done-:**

The data size is huge and the marketing team has asked you to perform the below analysis-

**Q1.Load data and create a Spark data frame**

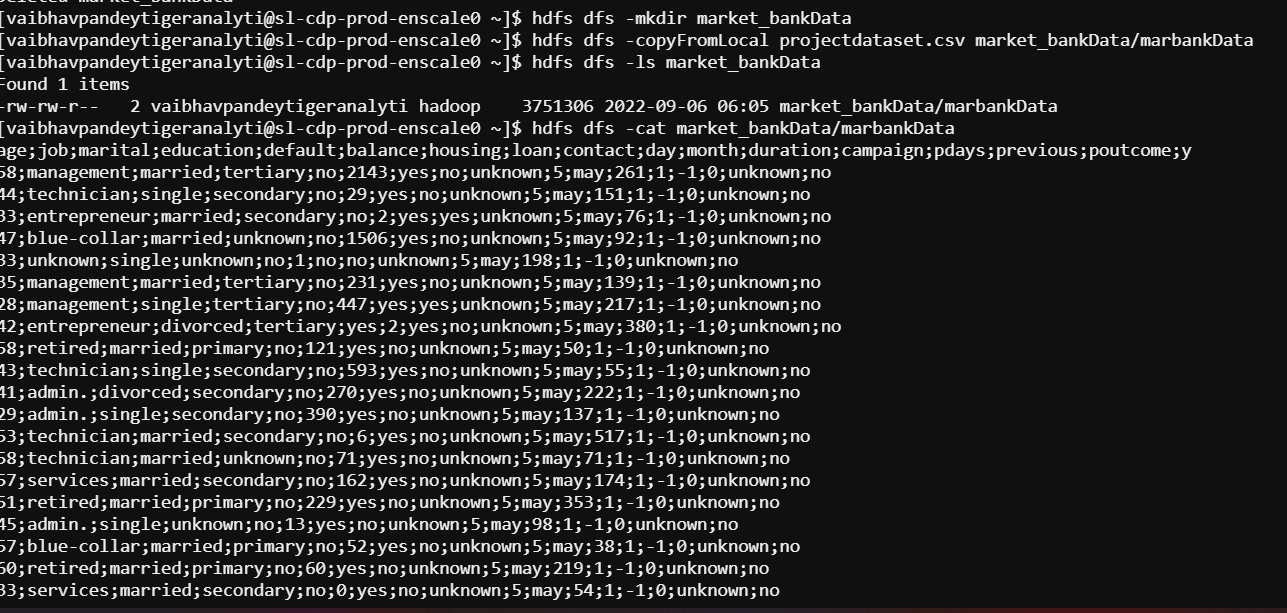
**Soln:**

*Here first import data from FTP to HDFS.*

hdfs dfs -mkdir market\_bankData

hdfs dfs -copyFromLocal projectdataset.csv market\_bankData/marbankData

hdfs dfs -cat market\_bankData/marbankData



*Then read the csv file using spark.read.csv command.*

marBankDF = spark.read.option("header",True).option("inferSchema",True).option("delimiter",";").csv('market\_bankData/marbankData')

marBankDF.show(10)

A picture containing graphical user interface

Description automatically generated

**Q 2.Give marketing success rate (No. of people subscribed / total no. of entries)**

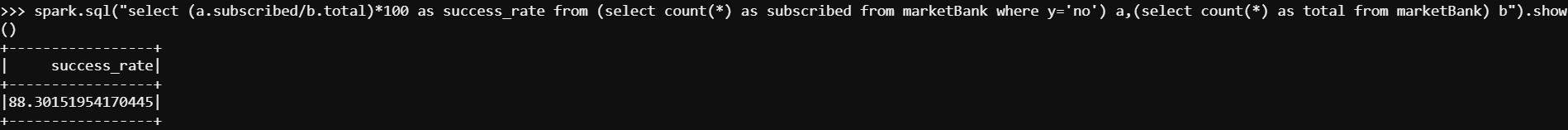
spark.sql("select (a.subscribed/b.total)\*100 as success\_rate from (select count(\*) as subscribed from marketBank where y='yes') a,(select count(\*) as total from marketBank) b").show()



*Marketing success rate = No. of subscribers/Total no. of entries \*100= (5289.0 /45211)\*100=11.69%*

**Q 3. Give marketing failure rate**

spark.sql("select (a.subscribed/b.total)\*100 as success\_rate from (select count(\*) as subscribed from marketBank where y='no') a,(select count(\*) as total from marketBank) b").show()



*Marketing failure rate = No. of Non-subscribers/Total no. of entries \*100= ( 39922.0/45211)\*100=88.30%*

**Q 4. Give the maximum, mean, and minimum age of the average targeted customer**

spark.sql("select max(age), min(age),mean(age) from marketBank").show()

Text

Description automatically generated

*Maximum age= 95*

*Mean age= 40.93*

*Minimum age= 18*

**Q 5. Check the quality of customers by checking average balance, median balance of customers**

spark.sql("select balance from marketBank").summary().show()

Text

Description automatically generated

*Average balance = 1362.27*

*Median balance = 448*

*So, the quality of customer is good.*

**Q 6. Check if age matters in marketing subscription for deposit**

spark.sql("select age,count(\*) as number from marketBank where y='yes' group by age order by number desc").show()



*Customers between 30-40 ages have the highest number of subscriptions.*

**Q 7. Check if marital status mattered for a subscription to deposit**

spark.sql("select marital, count(\*) as number from marketBank where y='yes' group by marital order by number desc").show()

Text

Description automatically generated

*Married customers make more deposits than other groups.*

**Q 8. Check if age and marital status together mattered for a subscription to deposit scheme**

spark.sql("select age,marital,count(\*) as number from marketBank where y='yes' group by age, marital order by number desc").show()

Text

Description automatically generated

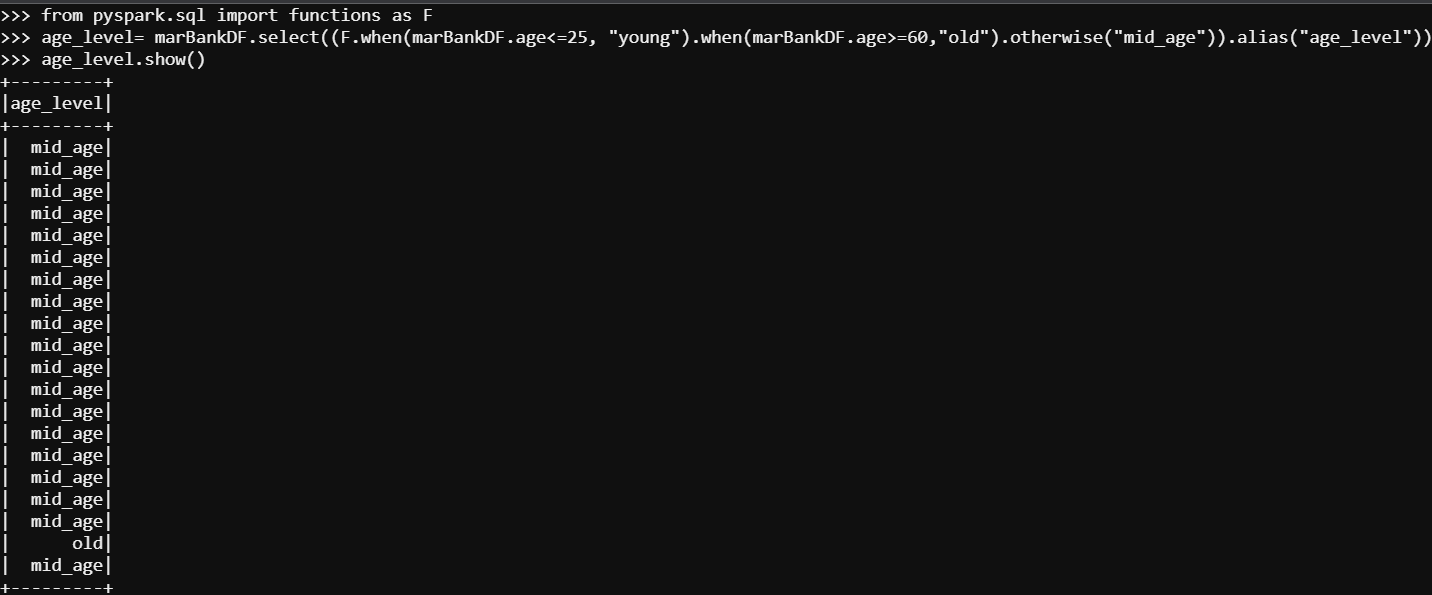
*Customers whose marital status is single and age group is between 25 to 35 are more likely to take subscriptions and have highest amount of deposits.*

**Q 9. Do feature engineering for the bank and find the right age effect on the campaign.**

from pyspark.sql import functions as F

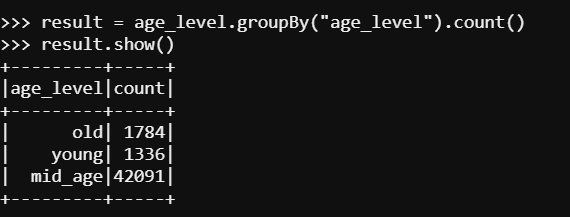
age\_level= marBankDF.select((F.when(marBankDF.age<=25,"young").when(marBankDF.age>=60,"old").otherwise("mid\_age")).alias("age\_level"))

age\_level.show()



result = age\_level.groupBy("age\_level").count()

result.show()



*As predicted before the age group with most deposits belongs to adults or mid aged group between 30 to 40 years.*