Marketing Analysis in Banking Domain

# Source Code

## 1. Load data and create a Spark data frame

import org.apache.spark.sql.SQLContext

val sqlContext = new SQLContext(sc)

val df = sqlContext.read.format("com.databricks.spark.csv").option("header", "true").option("inferSchema","true").load("/user/vibhashekarns/Bank/Bank\_Campaign\_Data.csv")

df.printSchema

df.show

df.select("age","job","marital","education","default","balance","housing","loan","contact","day","month","duration","campaign","pdays","previous","poutcome","y").show

df.registerTempTable("spark\_tbl\_Marketing\_Analysis")

sqlContext.sql("""select \* from spark\_tbl\_Marketing\_Analysis limit 10""").show();

## 2. Give marketing success rate (No. of people subscribed / total no. of entries) and marketing failure rate

val tot\_count = df.count()

val reg\_success = df.filter("y='yes'").count()

val success\_rate = tot\_count.toFloat/reg\_success

val reg\_fail = df.filter("y = 'no'").count()

val fail\_rate = tot\_count.toFloat/reg\_fail

## 3. Give the maximum, mean, and minimum age of the average targeted customer

val age\_stat = sqlContext.sql("select max(age), min(age), avg(age) from spark\_tbl\_Marketing\_Analysis")

age\_stat.show()

## 4. Check the quality of customers by checking average balance and median balance of customers

val cust\_qual = sqlContext.sql("select percentile\_approx(balance, .5), avg(balance) from spark\_tbl\_Marketing\_Analysis")

cust\_qual.show()

## 5. Check if age matters in marketing subscription for deposit

val age\_mktg = sqlContext.sql("select y, avg(age) from spark\_tbl\_Marketing\_Analysis group by y")

age\_mktg.show()

## 6. Check if marital status mattered for a subscription to deposit

val marital\_mktg=sqlContext.sql("select marital, y, count(marital) from spark\_tbl\_Marketing\_Analysis group by marital, y order by y")

marital\_mktg.show()

## 7. Check if age and marital status together mattered for subscription to deposit scheme

val age\_mktg\_subs=sqlContext.sql("select marital, y, count(marital), avg(age) from spark\_tbl\_Marketing\_Analysis group by marital, y order by y")

age\_mktg\_subs.show()

## 8. Do feature engineering for the bank and find the right age effect on the campaign

val age\_camp = sqlContext.sql("select y, age, count(y) from spark\_tbl\_Marketing\_Analysis group by y, age order by y, age")

age\_camp.show()