Project 3 - Market Analysis in Banking Domain

```
[kuntalc1_gmail@ip-10-0-1-10 ~]$ spark2-shell
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
20/07/25 14:49:55 WARN util.Utils: Service 'SparkUI' could not bind on port 42001. Attempting port
42002.
20/07/25 14:49:55 WARN util.Utils: Service 'SparkUI' could not bind on port 42002. Attempting port
42003.
20/07/25 14:49:55 WARN util.Utils: Service 'SparkUI' could not bind on port 42003. Attempting port
42004.
20/07/25 14:49:55 WARN util.Utils: Service 'SparkUI' could not bind on port 42004. Attempting port
42005.
20/07/25 14:49:55 WARN util.Utils: Service 'SparkUI' could not bind on port 42005. Attempting port
42006.
20/07/25 14:49:55 WARN util.Utils: Service 'SparkUI' could not bind on port 42006. Attempting port
42007.
20/07/25 14:49:55 WARN util.Utils: Service 'SparkUI' could not bind on port 42007. Attempting port
42008.
20/07/25 14:49:55 WARN util.Utils: Service 'SparkUI' could not bind on port 42008. Attempting port
42009.
Spark context Web UI available at http://ip-10-0-1-
10.ec2.internal:42009
Spark context available as 'sc' (master = yarn, app id = application_1594878743366_1911).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.4.0.cloudera2
      /_/

Using Scala version 2.11.12 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_144)
Type in expressions to have them evaluated.
Type :help for more information.

scala> val sqlContext = new org.apache.spark.sql.hive.HiveContext(sc)
warning: there was one deprecation warning; re-run with -deprecation for details
sqlContext: org.apache.spark.sql.hive.HiveContext = org.apache.spark.sql.hive.HiveContext@d4962bd


scala>
```

1. Load data and create Spark data frame

```
scala> val dfs =
sqlContext.read.format("com.databricks.spark.csv").option("header","true").option("inferSchema","tr
ue").option("delimiter",
",").load("/user/kuntalc1_gmail/project/Marketing_Analysis.csv")
dfs: org.apache.spark.sql.DataFrame = [age: int, job: string ... 15 more fields]

scala>
scala> dfs.show();
+---+------------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+------
--+-----+--------+--------+---+
```

```
|age|         job|
marital|education|default|balance|housing|loan|contact|day|month|duration|campaign|pdays|previous|p
outcome|  y|
+---+------------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+------
--+-----+--------+--------+---+
| 58|  management| married| tertiary|     no|   2143|    yes|  no|unknown|  5|  may|     261|
1|   -1|       0| unknown| no|
| 44|  technician|  single|secondary|     no|     29|    yes|  no|unknown|  5|  may|     151|
1|   -1|       0| unknown| no|
| 33|entrepreneur| married|secondary|     no|      2|    yes| yes|unknown|  5|  may|      76|
1|   -1|       0| unknown| no|
| 47| blue-collar| married|  unknown|     no|   1506|    yes|  no|unknown|  5|  may|      92|
1|   -1|       0| unknown| no|
| 33|     unknown|  single|  unknown|     no|      1|     no|  no|unknown|  5|  may|     198|
1|   -1|       0| unknown| no|
| 35|  management| married| tertiary|     no|    231|    yes|  no|unknown|  5|  may|     139|
1|   -1|       0| unknown| no|
| 28|  management|  single| tertiary|     no|    447|    yes| yes|unknown|  5|  may|     217|
1|   -1|       0| unknown| no|
| 42|entrepreneur|divorced| tertiary|    yes|      2|    yes|  no|unknown|  5|  may|     380|
1|   -1|       0| unknown| no|
| 58|     retired| married|  primary|     no|    121|    yes|  no|unknown|  5|  may|      50|
1|   -1|       0| unknown| no|
| 43|  technician|  single|secondary|     no|    593|    yes|  no|unknown|  5|  may|      55|
1|   -1|       0| unknown| no|
| 41|      admin.|divorced|secondary|     no|    270|    yes|  no|unknown|  5|  may|     222|
1|   -1|       0| unknown| no|
| 29|      admin.|  single|secondary|     no|    390|    yes|  no|unknown|  5|  may|     137|
1|   -1|       0| unknown| no|
| 53|  technician| married|secondary|     no|      6|    yes|  no|unknown|  5|  may|     517|
1|   -1|       0| unknown| no|
| 58|  technician| married|  unknown|     no|     71|    yes|  no|unknown|  5|  may|      71|
1|   -1|       0| unknown| no|
| 57|    services| married|secondary|     no|    162|    yes|  no|unknown|  5|  may|     174|
1|   -1|       0| unknown| no|
| 51|     retired| married|  primary|     no|    229|    yes|  no|unknown|  5|  may|     353|
1|   -1|       0| unknown| no|
| 45|      admin.|  single|  unknown|     no|     13|    yes|  no|unknown|  5|  may|      98|
1|   -1|       0| unknown| no|
| 57| blue-collar| married|  primary|     no|     52|    yes|  no|unknown|  5|  may|      38|
1|   -1|       0| unknown| no|
| 60|     retired| married|  primary|     no|     60|    yes|  no|unknown|  5|  may|     219|
1|   -1|       0| unknown| no|
| 33|    services| married|secondary|     no|      0|    yes|  no|unknown|  5|  may|      54|
1|   -1|       0| unknown| no|
+---+------------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+------
--+-----+--------+--------+---+
only showing top 20 rows


scala>

scala> dfs.createOrReplaceTempView("marketing_analysis_data")

scala> sqlContext.sql("select * from marketing_analysis_data").show;
+---+------------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+------
--+-----+--------+--------+---+
|age|         job|
marital|education|default|balance|housing|loan|contact|day|month|duration|campaign|pdays|previous|p
outcome|  y|
```

```
+---+-----------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+------
--+-----+--------+--------+---+
| 58|  management| married| tertiary|     no|   2143|    yes|  no|unknown|  5|  may|     261|
1|   -1|       0| unknown| no|
| 44|  technician|  single|secondary|     no|     29|    yes|  no|unknown|  5|  may|     151|
1|   -1|       0| unknown| no|
| 33|entrepreneur| married|secondary|     no|      2|    yes| yes|unknown|  5|  may|      76|
1|   -1|       0| unknown| no|
| 47| blue-collar| married|  unknown|     no|   1506|    yes|  no|unknown|  5|  may|      92|
1|   -1|       0| unknown| no|
| 33|     unknown|  single|  unknown|     no|      1|     no|  no|unknown|  5|  may|     198|
1|   -1|       0| unknown| no|
| 35|  management| married| tertiary|     no|    231|    yes|  no|unknown|  5|  may|     139|
1|   -1|       0| unknown| no|
| 28|  management|  single| tertiary|     no|    447|    yes| yes|unknown|  5|  may|     217|
1|   -1|       0| unknown| no|
| 42|entrepreneur|divorced| tertiary|    yes|      2|    yes|  no|unknown|  5|  may|     380|
1|   -1|       0| unknown| no|
| 58|     retired| married|  primary|     no|    121|    yes|  no|unknown|  5|  may|      50|
1|   -1|       0| unknown| no|
| 43|  technician|  single|secondary|     no|    593|    yes|  no|unknown|  5|  may|      55|
1|   -1|       0| unknown| no|
| 41|      admin.|divorced|secondary|     no|    270|    yes|  no|unknown|  5|  may|     222|
1|   -1|       0| unknown| no|
| 29|      admin.|  single|secondary|     no|    390|    yes|  no|unknown|  5|  may|     137|
1|   -1|       0| unknown| no|
| 53|  technician| married|secondary|     no|      6|    yes|  no|unknown|  5|  may|     517|
1|   -1|       0| unknown| no|
| 58|  technician| married|  unknown|     no|     71|    yes|  no|unknown|  5|  may|      71|
1|   -1|       0| unknown| no|
| 57|    services| married|secondary|     no|    162|    yes|  no|unknown|  5|  may|     174|
1|   -1|       0| unknown| no|
| 51|     retired| married|  primary|     no|    229|    yes|  no|unknown|  5|  may|     353|
1|   -1|       0| unknown| no|
| 45|      admin.|  single|  unknown|     no|     13|    yes|  no|unknown|  5|  may|      98|
1|   -1|       0| unknown| no|
| 57| blue-collar| married|  primary|     no|     52|    yes|  no|unknown|  5|  may|      38|
1|   -1|       0| unknown| no|
| 60|     retired| married|  primary|     no|     60|    yes|  no|unknown|  5|  may|     219|
1|   -1|       0| unknown| no|
| 33|    services| married|secondary|     no|      0|    yes|  no|unknown|  5|  may|      54|
1|   -1|       0| unknown| no|
+---+-----------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+------
--+-----+--------+--------+---+
only showing top 20 rows


scala>
```

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

2a Give marketing failure rate

```
scala> val tot_count = sqlContext.sql("SELECT * FROM marketing_analysis_data").count
tot_count: Long = 45211


scala> val reg_success = sqlContext.sql("select * from marketing_analysis_data where
y='yes'").count
reg_success: Long = 5289

scala>
```

```
scala> val success_rate = reg_success/tot_count.toFloat
success_rate: Float = 0.11698481

scala>

scala> val reg_fail = sqlContext.sql("select * from marketing_analysis_data where y='no'").count
reg_fail: Long = 39922

scala>


scala>

scala> val fail_rate = reg_fail/tot_count.toFloat
fail_rate: Float = 0.8830152

scala>
```

3. Maximum, Mean, and Minimum age of average targeted customer

```
scala> sqlContext.sql("select max(age) from marketing_analysis_data").show
+--------+
|max(age)|
+--------+
|      95|
+--------+


scala>

scala>sqlContext.sql("select min(age) from marketing_analysis_data").show
+--------+
|min(age)|
+--------+
|      18|
+--------+


scala>

scala> sqlContext.sql("select avg(age) from marketing_analysis_data").show
+-----------------+
|         avg(age)|
+-----------------+
|40.93621021432837|
+-----------------+


scala>
```

4. Check quality of customers by checking average balance, median balance of customers

```
scala> sqlContext.sql("select avg(balance) from marketing_analysis_data").show
+------------------+
|      avg(balance)|
+------------------+
|1362.2720576850766|
```

```
+------------------+


scala> sqlContext.sql("select percentile(balance, 0.5) as Median from
marketing_analysis_data").show
+------+
|Median|
+------+
| 448.0|
+------+


scala>
```

5. Check if age matters in marketing subscription for deposit

```
scala> sqlContext.sql("select age,y from marketing_analysis_data").show
+---+---+
|age|  y|
+---+---+
| 58| no|
| 44| no|
| 33| no|
| 47| no|
| 33| no|
| 35| no|
| 28| no|
| 42| no|
| 58| no|
| 43| no|
| 41| no|
| 29| no|
| 53| no|
| 58| no|
| 57| no|
| 51| no|
| 45| no|
| 57| no|
| 60| no|
| 33| no|
+---+---+
only showing top 20 rows


scala>




scala> sqlContext.sql("select avg(age),y from marketing_analysis_data group by y").show
+------------------+---+
|          avg(age)|  y|
+------------------+---+
| 40.83898602274435| no|
|41.670069956513515|yes|
+------------------+---+
```

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

```
scala> sqlContext.sql("select marital,y from marketing_analysis_data").show
+--------+---+
| marital|  y|
+--------+---+
| married| no|
|  single| no|
| married| no|
| married| no|
|  single| no|
| married| no|
|  single| no|
|divorced| no|
| married| no|
|  single| no|
|divorced| no|
|  single| no|
| married| no|
| married| no|
| married| no|
| married| no|
|  single| no|
| married| no|
| married| no|
| married| no|
+--------+---+
only showing top 20 rows


scala>
```

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

```
scala> sqlContext.sql("select age,marital,y from marketing_analysis_data").show
+---+--------+---+
|age| marital|  y|
+---+--------+---+
| 58| married| no|
| 44|  single| no|
| 33| married| no|
| 47| married| no|
| 33|  single| no|
| 35| married| no|
| 28|  single| no|
| 42|divorced| no|
| 58| married| no|
| 43|  single| no|
| 41|divorced| no|
| 29|  single| no|
| 53| married| no|
| 58| married| no|
| 57| married| no|
| 51| married| no|
| 45|  single| no|
| 57| married| no|
| 60| married| no|
| 33| married| no|
+---+--------+---+
only showing top 20 rows
```

```
scala>
```

8. Do feature engineering for column—age and find right age effect on campaign

```
scala> sqlContext.sql("select age,y,count(*) as total_count from marketing_analysis_data group by
age,y").show
+---+---+-----------+
|age|  y|total_count|
+---+---+-----------+
| 20| no|         35|
| 78| no|         16|
| 56|yes|         68|
| 28|yes|        162|
| 29|yes|        171|
| 71| no|         29|
| 86|yes|          4|
| 57| no|        750|
| 79|yes|         10|
| 22|yes|         40|
| 42| no|       1131|
| 31|yes|        206|
| 59|yes|         88|
| 87|yes|          3|
| 25| no|        414|
| 34|yes|        198|
| 23|yes|         44|
| 63| no|         47|
| 24| no|        234|
| 64| no|         39|
+---+---+-----------+
only showing top 20 rows


scala>


scala> val ageCatDF=sqlContext.sql("""select *,case
     |
     |      when age < 30 then 'Young'
     |
     |      when age > 60 then 'Old'
     |
     |      else 'Mid Age'
     |
     |      end as age_category from marketing_analysis_data""")
ageCatDF: org.apache.spark.sql.DataFrame = [age: int, job: string ... 16 more fields]

scala>

scala> ageCatDF.show
+---+------------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+------
--+-----+--------+--------+---+----------
--+
|age|         job|
marital|education|default|balance|housing|loan|contact|day|month|duration|campaign|pdays|previous|p
outcome|  y|age_catego
ry|
+---+------------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+------
--+-----+--------+--------+---+----------
--+
```

```
| 58|  management| married| tertiary|     no|   2143|    yes|  no|unknown|  5|  may|     261|
1|   -1|       0| unknown| no|     Mid A
ge|
| 44|  technician|  single|secondary|     no|     29|    yes|  no|unknown|  5|  may|     151|
1|   -1|       0| unknown| no|     Mid A
ge|
| 33|entrepreneur| married|secondary|     no|      2|    yes| yes|unknown|  5|  may|      76|
1|   -1|       0| unknown| no|     Mid A
ge|
| 47| blue-collar| married|  unknown|     no|   1506|    yes|  no|unknown|  5|  may|      92|
1|   -1|       0| unknown| no|     Mid A
ge|
| 33|     unknown|  single|  unknown|     no|      1|     no|  no|unknown|  5|  may|     198|
1|   -1|       0| unknown| no|     Mid A
ge|
| 35|  management| married| tertiary|     no|    231|    yes|  no|unknown|  5|  may|     139|
1|   -1|       0| unknown| no|     Mid A
ge|
| 28|  management|  single| tertiary|     no|    447|    yes| yes|unknown|  5|  may|     217|
1|   -1|       0| unknown| no|       You
ng|
| 42|entrepreneur|divorced| tertiary|    yes|      2|    yes|  no|unknown|  5|  may|     380|
1|   -1|       0| unknown| no|     Mid A
ge|
| 58|     retired| married|  primary|     no|    121|    yes|  no|unknown|  5|  may|      50|
1|   -1|       0| unknown| no|     Mid A
ge|
| 43|  technician|  single|secondary|     no|    593|    yes|  no|unknown|  5|  may|      55|
1|   -1|       0| unknown| no|     Mid A
ge|
| 41|      admin.|divorced|secondary|     no|    270|    yes|  no|unknown|  5|  may|     222|
1|   -1|       0| unknown| no|     Mid A
ge|
| 29|      admin.|  single|secondary|     no|    390|    yes|  no|unknown|  5|  may|     137|
1|   -1|       0| unknown| no|       You
ng|
| 53|  technician| married|secondary|     no|      6|    yes|  no|unknown|  5|  may|     517|
1|   -1|       0| unknown| no|     Mid A
ge|
| 58|  technician| married|  unknown|     no|     71|    yes|  no|unknown|  5|  may|      71|
1|   -1|       0| unknown| no|     Mid A
ge|
| 57|    services| married|secondary|     no|    162|    yes|  no|unknown|  5|  may|     174|
1|   -1|       0| unknown| no|     Mid A
ge|
| 51|     retired| married|  primary|     no|    229|    yes|  no|unknown|  5|  may|     353|
1|   -1|       0| unknown| no|     Mid A
ge|
| 45|      admin.|  single|  unknown|     no|     13|    yes|  no|unknown|  5|  may|      98|
1|   -1|       0| unknown| no|     Mid A
ge|
| 57| blue-collar| married|  primary|     no|     52|    yes|  no|unknown|  5|  may|      38|
1|   -1|       0| unknown| no|     Mid A
ge|
| 60|     retired| married|  primary|     no|     60|    yes|  no|unknown|  5|  may|     219|
1|   -1|       0| unknown| no|     Mid A
ge|
| 33|    services| married|secondary|     no|      0|    yes|  no|unknown|  5|  may|      54|
1|   -1|       0| unknown| no|     Mid A
ge|
+---+------------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+------
--+-----+--------+--------+---+----------
```

```
--+
only showing top 20 rows


scala>

scala> ageCatDF.groupBy("age_category","y").count().sort($"count".desc).show
+------------+---+-----+
|age_category|  y|count|
+------------+---+-----+
|     Mid Age| no|34891|
|       Young| no| 4345|
|     Mid Age|yes| 3859|
|       Young|yes|  928|
|         Old| no|  686|
|         Old|yes|  502|
+------------+---+-----+


scala>
```

Conclusion: Middle Age people are showing more interest on deposits