Sheet

```
from pyspark.sql import SparkSession
from pyspark.ml.feature import StringIndexer, VectorAssembler
from pyspark.ml.classification import LogisticRegression
from pyspark.ml.evaluation import BinaryClassificationEvaluator

spark = SparkSession.builder.getOrCreate()

suvDF = spark.read.csv("suv_data.csv",header = True, inferSchema = True)

suvDF.printSchema()

"""
root
 |-- User_ID: integer (nullable = true)
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)
 |-- Purchased: integer (nullable = true)

"""
```

```
root
 |-- User_ID: integer (nullable = true)
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)
 |-- Purchased: integer (nullable = true)


'\nroot\n |-- User_ID: integer (nullable = true)\n |-- Gender: string (nullable = true)\n |-- Age: integer (nullable = true)\n |-- E
```

```
suvDF.show(10,False)
"""
+--------+------+---+---------------+---------+
|User_ID |Gender|Age|EstimatedSalary|Purchased|
+--------+------+---+---------------+---------+
|15624510|Male  |19 |19000          |0        |
|15810944|Male  |35 |20000          |0        |
|15668575|Female|26 |43000          |0        |
|15603246|Female|27 |57000          |0        |
|15804002|Male  |19 |76000          |0        |
|15728773|Male  |27 |58000          |0        |
|15598044|Female|27 |84000          |0        |
|15694829|Female|32 |150000         |1        |
|15600575|Male  |25 |33000          |0        |
|15727311|Female|35 |65000          |0        |
+--------+------+---+---------------+---------+
"""
```

```
+--------+------+---+---------------+---------+
|User_ID |Gender|Age|EstimatedSalary|Purchased|
+--------+------+---+---------------+---------+
|15624510|Male  |19 |19000          |0        |
|15810944|Male  |35 |20000          |0        |
|15668575|Female|26 |43000          |0        |
|15603246|Female|27 |57000          |0        |
|15804002|Male  |19 |76000          |0        |
|15728773|Male  |27 |58000          |0        |
|15598044|Female|27 |84000          |0        |
|15694829|Female|32 |150000         |1        |
|15600575|Male  |25 |33000          |0        |
|15727311|Female|35 |65000          |0        |
+--------+------+---+---------------+---------+
only showing top 10 rows


'\n+--------+------+---+---------------+---------+\n|User_ID |Gender|Age|EstimatedSalary|Purchased|\n+--------+------+---+----------
```

```
suvDF.describe().show()

suvDF.createOrReplaceTempView("first_view")

"""
+-------+-----------------+------+-----------------+----------------+------------------+
|summary|          User_ID|Gender|              Age| EstimatedSalary|         Purchased|
+-------+-----------------+------+-----------------+----------------+------------------+
|  count|              400|   400|              400|             400|               400|
|   mean| 1.56915397575E7|  null|           37.655|         69742.5|            0.3575|
| stddev|71658.32158119006|  null|10.482876597307927|34096.9602824248|0.4798639635968691|
|    min|         15566689|Female|               18|           15000|                 0|
|    max|         15815236|  Male|               60|          150000|                 1|
+-------+-----------------+------+-----------------+----------------+------------------+
"""
```

```
+-------+-----------------+------+-----------------+----------------+------------------+
|summary|          User_ID|Gender|              Age| EstimatedSalary|         Purchased|
+-------+-----------------+------+-----------------+----------------+------------------+
|  count|              400|   400|              400|             400|               400|
|   mean| 1.56915397575E7|  null|           37.655|         69742.5|            0.3575|
| stddev|71658.32158119006|  null|10.482876597307927|34096.9602824248|0.4798639635968691|
|    min|         15566689|Female|               18|           15000|                 0|
|    max|         15815236|  Male|               60|          150000|                 1|
+-------+-----------------+------+-----------------+----------------+------------------+



'\n+-------+-----------------+------+-----------------+----------------+------------------+\n|summary|          User_ID|Gender|
```

```
data_set_suv= spark.sql("select Gender,Age,EstimatedSalary,Purchased from first_view ")
data_set_suv.show(5,False)
"""
+------+---+---------------+---------+
|Gender|Age|EstimatedSalary|Purchased|
+------+---+---------------+---------+
|Male  |19 |19000          |0        |
|Male  |35 |20000          |0        |
|Female|26 |43000          |0        |
|Female|27 |57000          |0        |
|Male  |19 |76000          |0        |
+------+---+---------------+---------+
"""
```

```
+------+---+---------------+---------+
|Gender|Age|EstimatedSalary|Purchased|
+------+---+---------------+---------+
|Male  |19 |19000          |0        |
|Male  |35 |20000          |0        |
|Female|26 |43000          |0        |
|Female|27 |57000          |0        |
|Male  |19 |76000          |0        |
+------+---+---------------+---------+
only showing top 5 rows
```

```
suv_indexer = StringIndexer(inputCol = "Gender", outputCol = "gen_label")

gen_label_df = suv_indexer.fit(data_set_suv).transform(data_set_suv)

print("after adding label")

gen_label_df.printSchema()

gen_label_df.show(5,False)

"""
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)
 |-- Purchased: integer (nullable = true)
 |-- gen_label: double (nullable = false)
"""

"""
+------+---+---------------+---------+---------+
|Gender|Age|EstimatedSalary|Purchased|gen_label|
+------+---+---------------+---------+---------+
|Male  |19 |19000          |0        |1.0      |
|Male  |35 |20000          |0        |1.0      |
|Female|26 |43000          |0        |0.0      |
|Female|27 |57000          |0        |0.0      |
|Male  |19 |76000          |0        |1.0      |
+------+---+---------------+---------+---------+

"""
```

```
after adding label
root
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)
 |-- Purchased: integer (nullable = true)
 |-- gen_label: double (nullable = false)

+------+---+---------------+---------+---------+
|Gender|Age|EstimatedSalary|Purchased|gen_label|
+------+---+---------------+---------+---------+
|Male  |19 |19000          |0        |1.0      |
|Male  |35 |20000          |0        |1.0      |
|Female|26 |43000          |0        |0.0      |
|Female|27 |57000          |0        |0.0      |
|Male  |19 |76000          |0        |1.0      |
+------+---+---------------+---------+---------+
only showing top 5 rows
```

```
inputCols = ["Age","EstimatedSalary","gen_label"]

outputCol = "features"

suvDF_assembler = VectorAssembler(inputCols = inputCols,outputCol = outputCol)
featuresDf = suvDF_assembler.transform(gen_label_df)

print("featuresDF printSchema")

featuresDf.printSchema()

"""
root
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)
 |-- Purchased: integer (nullable = true)
 |-- gen_label: double (nullable = false)
 |-- features: vector (nullable = true)
"""
```

```
featuresDF printSchema
root
 |-- Gender: string (nullable = true)
```

```
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)
 |-- Purchased: integer (nullable = true)
 |-- gen_label: double (nullable = false)
 |-- features: vector (nullable = true)


'\nroot\n |-- Gender: string (nullable = true)\n |-- Age: integer (nullable = true)\n |-- EstimatedSalary: integer (nullable = true)'
```

```
featuresDf.show(10,False)

print("featureDf show")

"""
+------+---+---------------+---------+---------+------------------+
|Gender|Age|EstimatedSalary|Purchased|gen_label|features          |
+------+---+---------------+---------+---------+------------------+
|Male  |19 |19000          |0        |1.0      |[19.0,19000.0,1.0] |
|Male  |35 |20000          |0        |1.0      |[35.0,20000.0,1.0] |
|Female|26 |43000          |0        |0.0      |[26.0,43000.0,0.0] |
|Female|27 |57000          |0        |0.0      |[27.0,57000.0,0.0] |
|Male  |19 |76000          |0        |1.0      |[19.0,76000.0,1.0] |
|Male  |27 |58000          |0        |1.0      |[27.0,58000.0,1.0] |
|Female|27 |84000          |0        |0.0      |[27.0,84000.0,0.0] |
|Female|32 |150000         |1        |0.0      |[32.0,150000.0,0.0]|
|Male  |25 |33000          |0        |1.0      |[25.0,33000.0,1.0] |
|Female|35 |65000          |0        |0.0      |[35.0,65000.0,0.0] |
+------+---+---------------+---------+---------+------------------+
"""
```

```
+------+---+---------------+---------+---------+------------------+
|Gender|Age|EstimatedSalary|Purchased|gen_label|features          |
+------+---+---------------+---------+---------+------------------+
|Male  |19 |19000          |0        |1.0      |[19.0,19000.0,1.0] |
|Male  |35 |20000          |0        |1.0      |[35.0,20000.0,1.0] |
|Female|26 |43000          |0        |0.0      |[26.0,43000.0,0.0] |
|Female|27 |57000          |0        |0.0      |[27.0,57000.0,0.0] |
|Male  |19 |76000          |0        |1.0      |[19.0,76000.0,1.0] |
|Male  |27 |58000          |0        |1.0      |[27.0,58000.0,1.0] |
|Female|27 |84000          |0        |0.0      |[27.0,84000.0,0.0] |
|Female|32 |150000         |1        |0.0      |[32.0,150000.0,0.0]|
|Male  |25 |33000          |0        |1.0      |[25.0,33000.0,1.0] |
|Female|35 |65000          |0        |0.0      |[35.0,65000.0,0.0] |
+------+---+---------------+---------+---------+------------------+
only showing top 10 rows

featureDf show


'\n+------+---+---------------+---------+---------+------------------+\n|Gender|Age|EstimatedSalary|Purchased|gen_label|features
```

```
suv_indexer = StringIndexer(inputCol = "Purchased", outputCol = "label")

label_df = suv_indexer.fit(featuresDf).transform(featuresDf)

print("after adding pur_label")

label_df.createOrReplaceTempView("main_df")

label_df.printSchema()

"""
root
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)
 |-- Purchased: integer (nullable = true)
 |-- gen_label: double (nullable = false)
 |-- features: vector (nullable = true)
 |-- label: double (nullable = false)

"""
```

```
Py4JJavaError: An error occurred while calling o945.fit.
: org.apache.spark.SparkException: Input column Purchased does not exist.
        at org.apache.spark.ml.feature.StringIndexerBase.$anonfun$validateAndTransformSchema$2(StringIndexer.scala:128)
        at scala.collection.TraversableLike.$anonfun$flatMap$1(TraversableLike.scala:293)
        at scala.collection.IndexedSeqOptimized.foreach(IndexedSeqOptimized.scala:36)
        at scala.collection.IndexedSeqOptimized.foreach$(IndexedSeqOptimized.scala:33)
        at scala.collection.mutable.ArrayOps$ofRef.foreach(ArrayOps.scala:198)
        at scala.collection.TraversableLike.flatMap(TraversableLike.scala:293)
        at scala.collection.TraversableLike.flatMap$(TraversableLike.scala:290)
        at scala.collection.mutable.ArrayOps$ofRef.flatMap(ArrayOps.scala:198)
        at org.apache.spark.ml.feature.StringIndexerBase.validateAndTransformSchema(StringIndexer.scala:123)
        at org.apache.spark.ml.feature.StringIndexerBase.validateAndTransformSchema$(StringIndexer.scala:115)
        at org.apache.spark.ml.feature.StringIndexer.validateAndTransformSchema(StringIndexer.scala:145)
        at org.apache.spark.ml.feature.StringIndexer.transformSchema(StringIndexer.scala:252)
        at org.apache.spark.ml.PipelineStage.transformSchema(Pipeline.scala:71)
        at org.apache.spark.ml.feature.StringIndexer.fit(StringIndexer.scala:237)
        at org.apache.spark.ml.feature.StringIndexer.fit(StringIndexer.scala:145)
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.base/java.lang.reflect.Method.invoke(Method.java:566)
        at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:244)
        at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.java:357)
        at py4j.Gateway.invoke(Gateway.java:282)
        at py4j.commands.AbstractCommand.invokeMethod(AbstractCommand.java:132)
        at py4j.commands.CallCommand.execute(CallCommand.java:79)
        at py4j.ClientServerConnection.waitForCommands(ClientServerConnection.java:182)
        at py4j.ClientServerConnection.run(ClientServerConnection.java:106)
        at java.base/java.lang.Thread.run(Thread.java:829)
```

```
label_df.show(10,False)

"""
+------+---+---------------+---------+---------+------------------+-----+
|Gender|Age|EstimatedSalary|Purchased|gen_label|features          |label|
+------+---+---------------+---------+---------+------------------+-----+
|Male  |19 |19000          |0        |1.0      |[19.0,19000.0,1.0]|0.0  |
|Male  |35 |20000          |0        |1.0      |[35.0,20000.0,1.0]|0.0  |
|Female|26 |43000          |0        |0.0      |[26.0,43000.0,0.0]|0.0  |
|Female|27 |57000          |0        |0.0      |[27.0,57000.0,0.0]|0.0  |
|Male  |19 |76000          |0        |1.0      |[19.0,76000.0,1.0]|0.0  |
|Male  |27 |58000          |0        |1.0      |[27.0,58000.0,1.0]|0.0  |
|Female|27 |84000          |0        |0.0      |[27.0,84000.0,0.0]|0.0  |
|Female|32 |150000         |1        |0.0      |[32.0,150000.0,0.0]|1.0  |
|Male  |25 |33000          |0        |1.0      |[25.0,33000.0,1.0]|0.0  |
|Female|35 |65000          |0        |0.0      |[35.0,65000.0,0.0]|0.0  |
+------+---+---------------+---------+---------+------------------+-----+
"""
```

```
+------+---+---------------+---------+---------+------------------+-----+
|Gender|Age|EstimatedSalary|Purchased|gen_label|features          |label|
+------+---+---------------+---------+---------+------------------+-----+
|Male  |19 |19000          |0        |1.0      |[19.0,19000.0,1.0]|0.0  |
|Male  |35 |20000          |0        |1.0      |[35.0,20000.0,1.0]|0.0  |
|Female|26 |43000          |0        |0.0      |[26.0,43000.0,0.0]|0.0  |
|Female|27 |57000          |0        |0.0      |[27.0,57000.0,0.0]|0.0  |
|Male  |19 |76000          |0        |1.0      |[19.0,76000.0,1.0]|0.0  |
|Male  |27 |58000          |0        |1.0      |[27.0,58000.0,1.0]|0.0  |
|Female|27 |84000          |0        |0.0      |[27.0,84000.0,0.0]|0.0  |
|Female|32 |150000         |1        |0.0      |[32.0,150000.0,0.0]|1.0  |
|Male  |25 |33000          |0        |1.0      |[25.0,33000.0,1.0]|0.0  |
|Female|35 |65000          |0        |0.0      |[35.0,65000.0,0.0]|0.0  |
+------+---+---------------+---------+---------+------------------+-----+
only showing top 10 rows


'\n+------+---+---------------+---------+---------+------------------+-----+\n|Gender|Age|EstimatedSalary|Purchased|gen_label|featu
```

```
trainingData,testdata = label_df.randomSplit([0.7,0.3],seed = 42)

print("display training data")

trainingData.show(10,False)

"""
+------+---+---------------+---------+---------+------------------+-----+
|Gender|Age|EstimatedSalary|Purchased|gen_label|features          |label|
+------+---+---------------+---------+---------+------------------+-----+
|Female|18 |44000          |0        |0.0      |[18.0,44000.0,0.0]|0.0  |
|Female|18 |68000          |0        |0.0      |[18.0,68000.0,0.0]|0.0  |
|Female|19 |21000          |0        |0.0      |[19.0,21000.0,0.0]|0.0  |
|Female|19 |26000          |0        |0.0      |[19.0,26000.0,0.0]|0.0  |
|Female|20 |23000          |0        |0.0      |[20.0,23000.0,0.0]|0.0  |
|Female|20 |82000          |0        |0.0      |[20.0,82000.0,0.0]|0.0  |
|Female|21 |68000          |0        |0.0      |[21.0,68000.0,0.0]|0.0  |
|Female|22 |27000          |0        |0.0      |[22.0,27000.0,0.0]|0.0  |
|Female|22 |55000          |0        |0.0      |[22.0,55000.0,0.0]|0.0  |
|Female|23 |66000          |0        |0.0      |[23.0,66000.0,0.0]|0.0  |
+------+---+---------------+---------+---------+------------------+-----+
"""
```

```
display training data
+------+---+---------------+---------+---------+------------------+-----+
|Gender|Age|EstimatedSalary|Purchased|gen_label|features          |label|
+------+---+---------------+---------+---------+------------------+-----+
|Female|18 |44000          |0        |0.0      |[18.0,44000.0,0.0]|0.0  |
|Female|18 |68000          |0        |0.0      |[18.0,68000.0,0.0]|0.0  |
|Female|19 |21000          |0        |0.0      |[19.0,21000.0,0.0]|0.0  |
|Female|19 |26000          |0        |0.0      |[19.0,26000.0,0.0]|0.0  |
|Female|20 |23000          |0        |0.0      |[20.0,23000.0,0.0]|0.0  |
|Female|20 |82000          |0        |0.0      |[20.0,82000.0,0.0]|0.0  |
|Female|21 |68000          |0        |0.0      |[21.0,68000.0,0.0]|0.0  |
|Female|22 |27000          |0        |0.0      |[22.0,27000.0,0.0]|0.0  |
|Female|22 |55000          |0        |0.0      |[22.0,55000.0,0.0]|0.0  |
|Female|23 |66000          |0        |0.0      |[23.0,66000.0,0.0]|0.0  |
+------+---+---------------+---------+---------+------------------+-----+
only showing top 10 rows


'\n+------+---+---------------+---------+---------+------------------+-----+\n|Gender|Age|EstimatedSalary|Purchased|gen_label|featur
```

```
logisticRegression = LogisticRegression().setMaxIter(100).setRegParam(0.02).setElasticNetParam(0.8)

logisticRegressionModel = logisticRegression.fit(trainingData)

predictionDf = logisticRegressionModel.transform(testdata)

print("logisticregession model prediction")

predictionDf.show(10,False)

"""
+------+---+---------------+---------+---------+------------------+-----+----------------------------------------+----------
|Gender|Age|EstimatedSalary|Purchased|gen_label|features          |label|rawPrediction                           |probability
+------+---+---------------+---------+---------+------------------+-----+----------------------------------------+----------
|Female|18 |86000          |0        |0.0      |[18.0,86000.0,0.0]|0.0  |[3.6705747720243806,-3.6705747720243806]|[0.9751703766806409,0
|Female|20 |36000          |0        |0.0      |[20.0,36000.0,0.0]|0.0  |[4.504284227708987,-4.504284227708987]  |[0.9890595133734785,0
|Female|20 |82000          |0        |0.0      |[20.0,82000.0,0.0]|0.0  |[3.4398933877329734,-3.4398933877329734]|[0.9689283063059155,0
|Female|21 |16000          |0        |0.0      |[21.0,16000.0,0.0]|0.0  |[4.805444299032157,-4.805444299032157]  |[0.9918813877668449,0
|Female|22 |63000          |0        |0.0      |[22.0,63000.0,0.0]|0.0  |[3.556295972998962,-3.556295972998962]  |[0.9722478102285415,0
|Female|23 |28000          |0        |0.0      |[23.0,28000.0,0.0]|0.0  |[4.204540013879528,-4.204540013879528]  |[0.9852919062875174,0
|Female|23 |48000          |0        |0.0      |[23.0,48000.0,0.0]|0.0  |[3.7417613878030007,-3.7417613878030007]|[0.9768369494282078,0
|Female|24 |32000          |0        |0.0      |[24.0,32000.0,0.0]|0.0  |[3.950365733910866,-3.950365733910866]  |[0.9811158154168544,0
|Female|24 |89000          |0        |0.0      |[24.0,89000.0,0.0]|0.0  |[2.631446649592763,-2.631446649592763]  |[0.9328582152820762,0
|Female|26 |17000          |0        |0.0      |[26.0,17000.0,0.0]|0.0  |[3.974212593961549,-3.974212593961549]  |[0.9815526077749361,0
+------+---+---------------+---------+---------+------------------+-----+----------------------------------------+----------
"""
```

```
logisticregession model prediction
+------+---+---------------+---------+---------+------------------+-----+----------------------------------------+----------
|Gender|Age|EstimatedSalary|Purchased|gen_label|features          |label|rawPrediction                           |probability
+------+---+---------------+---------+---------+------------------+-----+----------------------------------------+----------
```

```
|Female|18 |86000          |0        |0.0      |[18.0,86000.0,0.0]|0.0  |[3.6705747720243806,-3.6705747720243806]|[0.975170376680640
|Female|20 |36000          |0        |0.0      |[20.0,36000.0,0.0]|0.0  |[4.504284227708987,-4.504284227708987]  |[0.989059513373478
|Female|20 |82000          |0        |0.0      |[20.0,82000.0,0.0]|0.0  |[3.4398933877329734,-3.4398933877329734]|[0.968928306305915
|Female|21 |16000          |0        |0.0      |[21.0,16000.0,0.0]|0.0  |[4.805444299032157,-4.805444299032157]  |[0.991881387766844
|Female|22 |63000          |0        |0.0      |[22.0,63000.0,0.0]|0.0  |[3.556295972998962,-3.556295972998962]  |[0.972247810228541
|Female|23 |28000          |0        |0.0      |[23.0,28000.0,0.0]|0.0  |[4.204540013879528,-4.204540013879528]  |[0.985291906287517
|Female|23 |48000          |0        |0.0      |[23.0,48000.0,0.0]|0.0  |[3.7417613878030007,-3.7417613878030007]|[0.976836949428207
|Female|24 |32000          |0        |0.0      |[24.0,32000.0,0.0]|0.0  |[3.950365733910866,-3.950365733910866]  |[0.981115815416854
|Female|24 |89000          |0        |0.0      |[24.0,89000.0,0.0]|0.0  |[2.631446649592763,-2.631446649592763]  |[0.932858215282076
|Female|26 |17000          |0        |0.0      |[26.0,17000.0,0.0]|0.0  |[3.974212593961549,-3.974212593961549]  |[0.981552607774936
+------+---+---------------+---------+---------+------------------+-----+----------------------------------------+------------------
only showing top 10 rows


'\n+------+---+---------------+---------+---------+------------------+-----+----------------------------------------+---------------
```

```
evaluator = BinaryClassificationEvaluator() .setLabelCol("label").setRawPredictionCol("prediction").setMetricName("areaUnderROC")

accuracy = evaluator.evaluate(predictionDf)

print("accuracy of the model")

print(accuracy * 100)

"""
accuracy of the model
83.09386973180077

"""
```

```
accuracy of the model
83.09386973180077


'\naccuracy of the model\n83.09386973180077\n\n'
```

```
df1 = spark.read.csv("car_suv.csv",header = True, inferSchema = True)
df1.createOrReplaceTempView("input_view")

df1.printSchema()

"""
root
 |-- User_ID: integer (nullable = true)
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)

"""
```

```
root
 |-- User_ID: integer (nullable = true)
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)


'\nroot\n |-- User_ID: integer (nullable = true)\n |-- Gender: string (nullable = true)\n |-- Age: integer (nullable = true)\n |-- E
```

```
df1= spark.sql("select Gender,Age,EstimatedSalary from input_view ")
df1.show(5,False)

"""
+------+---+---------------+
|Gender|Age|EstimatedSalary|
+------+---+---------------+
|Male  |19 |19000          |
|Male  |35 |20000          |
|Female|26 |43000          |
|Female|27 |57000          |
|Male  |19 |76000          |
+------+---+---------------+
"""
```

```
+------+---+---------------+
|Gender|Age|EstimatedSalary|
+------+---+---------------+
|Male  |19 |19000          |
|Male  |35 |20000          |
|Female|26 |43000          |
|Female|27 |57000          |
|Male  |19 |76000          |
+------+---+---------------+
only showing top 5 rows


'\n+------+---+---------------+\n|Gender|Age|EstimatedSalary|\n+------+---+---------------+\n|Male  |19 |19000          |\n|Male  |3
```

```
input_indexer = StringIndexer(inputCol = "Gender", outputCol = "gen_label")

gen_label_input_df = input_indexer.fit(data_set_suv).transform(df1)

print("after adding label")

gen_label_input_df.printSchema()

gen_label_input_df.show(5,False)
```

```
after adding label
root
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)
 |-- gen_label: double (nullable = false)

+------+---+---------------+---------+
|Gender|Age|EstimatedSalary|gen_label|
+------+---+---------------+---------+
|Male  |19 |19000          |1.0      |
|Male  |35 |20000          |1.0      |
|Female|26 |43000          |0.0      |
|Female|27 |57000          |0.0      |
|Male  |19 |76000          |1.0      |
+------+---+---------------+---------+
only showing top 5 rows
```

```
inputCols = ["Age","EstimatedSalary","gen_label"]

outputCol = "features"

input_assembler = VectorAssembler(inputCols = inputCols,outputCol = outputCol)
featuresDf = input_assembler.transform(gen_label_input_df)

print("featuresDF printSchema")

featuresDf.printSchema()

"""
root
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)
 |-- gen_label: double (nullable = false)
 |-- features: vector (nullable = true)
"""

featuresDf.show(5,False)

"""
+------+---+---------------+---------+-----------------+
|Gender|Age|EstimatedSalary|gen_label|features         |
+------+---+---------------+---------+-----------------+
|Male  |19 |19000          |1.0      |[19.0,19000.0,1.0]|
|Male  |35 |20000          |1.0      |[35.0,20000.0,1.0]|
|Female|26 |43000          |0.0      |[26.0,43000.0,0.0]|
|Female|27 |57000          |0.0      |[27.0,57000.0,0.0]|
|Male  |19 |76000          |1.0      |[19.0,76000.0,1.0]|
+------+---+---------------+---------+-----------------+

"""
```

```
featuresDF printSchema
root
 |-- Gender: string (nullable = true)
 |-- Age: integer (nullable = true)
 |-- EstimatedSalary: integer (nullable = true)
 |-- gen_label: double (nullable = false)
 |-- features: vector (nullable = true)

+------+---+---------------+---------+-----------------+
|Gender|Age|EstimatedSalary|gen_label|features         |
+------+---+---------------+---------+-----------------+
|Male  |19 |19000          |1.0      |[19.0,19000.0,1.0]|
|Male  |35 |20000          |1.0      |[35.0,20000.0,1.0]|
|Female|26 |43000          |0.0      |[26.0,43000.0,0.0]|
|Female|27 |57000          |0.0      |[27.0,57000.0,0.0]|
|Male  |19 |76000          |1.0      |[19.0,76000.0,1.0]|
+------+---+---------------+---------+-----------------+
only showing top 5 rows



'\n+------+---+---------------+---------+---------+-----------------+\n|Gender|Age|EstimatedSalary|Purchased|gen_label|features
```

```
print("prediction_input")
input_pre = logisticRegressionModel.transform(featuresDf)
input_pre.show(5,False)

"""
+------+---+---------------+---------+-----------------+----------------------------------------+----------------------------------------
|Gender|Age|EstimatedSalary|gen_label|features         |rawPrediction                           |probability
+------+---+---------------+---------+-----------------+----------------------------------------+----------------------------------------
|Male  |19 |19000          |1.0      |[19.0,19000.0,1.0]|[5.059264614627391,-5.059264614627391]  |[0.993689843271015,0.006310156728985
|Male  |35 |20000          |1.0      |[35.0,20000.0,1.0]|[2.4502288072698626,-2.4502288072698626]|[0.920578181439154,0.079421818560845
|Female|26 |43000          |0.0      |[26.0,43000.0,0.0]|[3.372600380062064,-3.372600380062064]  |[0.9668371685880653,0.03316283141193
|Female|27 |57000          |0.0      |[27.0,57000.0,0.0]|[2.8870367870551386,-2.8870367870551386]|[0.9472018861086681,0.05279811389133
|Male  |19 |76000          |1.0      |[19.0,76000.0,1.0]|[3.7403455303092885,-3.7403455303092885]|[0.9768048918573531,0.02319510814264
+------+---+---------------+---------+-----------------+----------------------------------------+----------------------------------------
"""
```

```
prediction_input
+------+---+---------------+---------+-----------------+----------------------------------------+----------------------------------
|Gender|Age|EstimatedSalary|gen_label|features         |rawPrediction                           |probability
+------+---+---------------+---------+-----------------+----------------------------------------+----------------------------------
|Male  |19 |19000          |1.0      |[19.0,19000.0,1.0]|[5.059264614627391,-5.059264614627391]  |[0.993689843271015,0.0063101567289
|Male  |35 |20000          |1.0      |[35.0,20000.0,1.0]|[2.4502288072698626,-2.4502288072698626]|[0.920578181439154,0.0794218185608
|Female|26 |43000          |0.0      |[26.0,43000.0,0.0]|[3.372600380062064,-3.372600380062064]  |[0.9668371685880653,0.033162831411
|Female|27 |57000          |0.0      |[27.0,57000.0,0.0]|[2.8870367870551386,-2.8870367870551386]|[0.9472018861086681,0.052798113891
|Male  |19 |76000          |1.0      |[19.0,76000.0,1.0]|[3.7403455303092885,-3.7403455303092885]|[0.9768048918573531,0.023195108142
+------+---+---------------+---------+-----------------+----------------------------------------+----------------------------------
only showing top 5 rows
```

```
final_out =spark.sql ("SELECT main_df.subject_1_gp,main_df.subject_2_gp,main_df.subject_3_gp," +
      "main_df.subject_4_gp,main_df.subject_5_gp,main_df.grade,main_df.label,input_marks_view.prediction FROM main_df  " +
      "JOIN input_marks_view  ON main_df.subject_1_gp = input_marks_view.subject_1_gp AND main_df.subject_2_gp = input_marks_view.subj
      "AND main_df.subject_3_gp = input_marks_view.subject_3_gp AND main_df.subject_4_gp = input_marks_view.subject_4_gp AND " +
      "main_df.subject_5_gp = input_marks_view.subject_5_gp  GROUP BY main_df.subject_1_gp,main_df.subject_2_gp," +
      "main_df.subject_3_gp,main_df.subject_4_gp,main_df.subject_5_gp,main_df.grade,input_marks_view.prediction,main_df.label")
```