Hrishikesh Sanjay Pawar – SEC01 (NUID 002707307)

Big Data System Engineering with Scala
Spring 2023
Spark Assignment No. #2

-List of Tasks Implemented

1. Performed EDA of the Titanic survival dataset.

2. Created new attributes in the dataset.

3. Trained a random forest model, computed its accuracy, and used the model to predict survival on the test dataset.

-Findings and analysis

Explaining EDA statistics:

1. There are 577 males and 314 females in the training dataset. That is 65% males and 35% females.

2. Highest number of passengers (491) belong to the 3rd class ticket and lowest (184) belong to the 2nd class ticket.

3. Highest number of passengers (644) embarked from port Southampton.

4. Mean age is lowest for the 3rd class and highest for the 1st class for both Male and Female passengers.

5. Survival rate is highest for 1st class followed by 2nd and the 3rd classes respectively. Highest survival rate is for females in the 1st class at 96% followed by 92% in the 2nd class.

-Code

Top editor screenshot:

```scala
import org.apache.spark.sql.{DataFrame, SparkSession}
import org.apache.spark.sql.functions._
import org.apache.spark.ml.classification.RandomForestClassifier
import org.apache.spark.ml.Pipeline
import org.apache.spark.ml.feature.{StringIndexer, VectorAssembler}
import org.apache.spark.ml.evaluation.MulticlassClassificationEvaluator

object SparkAssignment2 extends App {

  // creating spark session
  implicit val spark: SparkSession = SparkSession
    .builder()
    .appName( name = "SparkAssignment2")
    .master( master = "local[*]")
    .getOrCreate()

  spark.sparkContext.setLogLevel("ERROR") // We want to ignore all of the INFO and WARN messages.

  // Read train data from CSV file
  val train = try {
```

Run console:
```
+------+-------+-------------------+
|  male|      3|0.13544668587896252|
```

Bottom editor screenshot:

```scala
  spark.sparkContext.setLogLevel("ERROR") // We want to ignore all of the INFO and WARN messages.

  // Read train data from CSV file
  val train = try {
    spark.read
      .option("header", true)
      .option("inferSchema", true)
      .csv( path = "src/main/resources/train.csv")
  } catch {
    case e: Exception =>
      println(s"An error occurred while reading the ratings CSV file: ${e.getMessage}")
      spark.emptyDataFrame
  }

  // Read test data from CSV file
  val test = try {
    spark.read
      .option("header", true)
      .option("inferSchema", true)
```

Run console:
```
+------+-------+-------------------+
|  male|      3|0.13544668587896252|
```

```scala
// Read test data from CSV file
val test = try {
  spark.read
    .option("header", true)
    .option("inferSchema", true)
    .csv( path = "src/main/resources/test.csv")
} catch {
  case e: Exception =>
    println(s"An error occurred while reading the ratings CSV file: ${e.getMessage}")
    spark.emptyDataFrame
}

/// ********** SECTION 1: EXPLORATORY DATA ANALYSIS ********** ///

// Show the first 5 rows of the dataset
train.show( numRows = 5)

// Print the schema of the dataset
train.printSchema()
```

```scala
train.printSchema()

// Compute summary statistics for numeric columns
train.describe( cols = "Age", "SibSp", "Parch", "Fare").show()

// Compute the count of each categorical value in the Sex column
train.groupBy( col1 = "Sex").count().show()

// Compute the count of each categorical value in the Pclass column
train.groupBy( col1 = "Pclass").count().show()

// Compute the count of each categorical value in the Embarked column
train.groupBy( col1 = "Embarked").count().show()

// Compute the mean age of passengers by sex and class
train.groupBy( col1 = "Sex", cols = "Pclass")
  .agg(avg( columnName = "Age").alias( alias = "mean_age"))
  .show()

// Compute the survival rate by sex and class
```

Top screenshot code:

```scala
// Compute the survival rate by sex and class
train.groupBy( col1 = "Sex", cols = "Pclass")
  .agg((sum(when(col( colName = "Survived") === 1, value = 1).otherwise( value = 0)) / count( columnName = "*")).alias(
  .show()


/// ********** SECTION 2: FEATURE ENGINEERING ********** ///

// Creating attribute FamSize - total family size = SibSp + Parch + 1
val familyUDF = udf((SibSp: Int, Parch: Int) => SibSp +Parch + 1)
val train_v2 = train.withColumn( colName = "FamSize", familyUDF(col( colName = "SibSp"), col( colName = "Parch")))
val test_v2 = test.withColumn( colName = "FamSize", familyUDF(col( colName = "SibSp"), col( colName = "Parch")))

// Dropping attributes SibSp and Parch
val train_v2_1 = train_v2.drop( colNames = "SibSp", "Parch")
val test_v2_1 = test_v2.drop( colNames = "SibSp", "Parch")

// Creating attribute FarePerPerson - Fare per person = Fare / FamSize
val train_v3 = train_v2_1.withColumn( colName = "FarePerPerson", col( colName = "Fare") / col( colName = "FamSize"))
val test_v3 = test_v2_1.withColumn( colName = "FarePerPerson", col( colName = "Fare") / col( colName = "FamSize"))
```

```
+------+------+-------------------+
|  male|     3|0.13544668587896252|
```

Bottom screenshot code:

```scala
val train_v3 = train_v2_1.withColumn( colName = "FarePerPerson", col( colName = "Fare") / col( colName = "
val test_v3 = test_v2_1.withColumn( colName = "FarePerPerson", col( colName = "Fare") / col( colName = "FamSize"))

// Creating attribute Age group
// Define the age groups
val ageGroups = Map(
  0 -> "Unknown",
  1 -> "Infant",
  2 -> "Toddler",
  3 -> "Child",
  4 -> "Teenager",
  5 -> "Young Adult",
  6 -> "Adult",
  7 -> "Senior"
)

val train_final = train_v3.withColumn(
  colName = "AgeGroup",
  when(col( colName = "Age").isNull, ageGroups(0))
    .when(col( colName = "Age") <= 1, ageGroups(1))
```

```
+------+------+-------------------+
|  male|     3|0.13544668587896252|
```

Top screenshot:

```scala
val train_final = train_v3.withColumn(
    colName = "AgeGroup",
  when(col( colName = "Age").isNull, ageGroups(0))
    .when(col( colName = "Age") <= 1, ageGroups(1))
    .when(col( colName = "Age") <= 3, ageGroups(2))
    .when(col( colName = "Age") <= 12, ageGroups(3))
    .when(col( colName = "Age") <= 19, ageGroups(4))
    .when(col( colName = "Age") <= 35, ageGroups(5))
    .when(col( colName = "Age") <= 65, ageGroups(6))
    .otherwise(ageGroups(7))
)

val test_final = test_v3.withColumn(
    colName = "AgeGroup",
  when(col( colName = "Age").isNull, ageGroups(0))
    .when(col( colName = "Age") <= 1, ageGroups(1))
    .when(col( colName = "Age") <= 3, ageGroups(2))
    .when(col( colName = "Age") <= 12, ageGroups(3))
    .when(col( colName = "Age") <= 19, ageGroups(4))
```

Run console:
```
+------+------+-------------------+
|  male|     3|0.13544668587896252|
```

Bottom screenshot:

```scala
val test_final = test_v3.withColumn(
    colName = "AgeGroup",
  when(col( colName = "Age").isNull, ageGroups(0))
    .when(col( colName = "Age") <= 1, ageGroups(1))
    .when(col( colName = "Age") <= 3, ageGroups(2))
    .when(col( colName = "Age") <= 12, ageGroups(3))
    .when(col( colName = "Age") <= 19, ageGroups(4))
    .when(col( colName = "Age") <= 35, ageGroups(5))
    .when(col( colName = "Age") <= 65, ageGroups(6))
    .otherwise(ageGroups(7))
)


/// ********** SECTION 3: PREDICTION ********** ///

// Define the features to use in the model
val features = Array("Pclass", "Sex", "Fare",
    "Embarked", "FamSize", "FarePerPerson", "AgeGroup")
```

Run console:
```
+------+------+-------------------+
|  male|     3|0.13544668587896252|
```

Screenshot 1:

```scala
// Define the features to use in the model
val features = Array("Pclass", "Sex", "Fare",
  "Embarked", "FamSize", "FarePerPerson", "AgeGroup")

// Create StringIndexers for the Sex and Embarked columns
val sexIndexer = new StringIndexer()
  .setInputCol("Sex")
  .setOutputCol("SexIndex")
  .setHandleInvalid("skip")

val embarkedIndexer = new StringIndexer()
  .setInputCol("Embarked")
  .setOutputCol("EmbarkedIndex")
  .setHandleInvalid("skip")

val ageGroupIndexer = new StringIndexer()
  .setInputCol("AgeGroup")
  .setOutputCol("AgeGroupIndex")
  .setHandleInvalid("skip")
```

```
+------+------+--------------------+
| male|     3|0.13544668587896252|
```

Screenshot 2:

```scala
val ageGroupIndexer = new StringIndexer()
  .setInputCol("AgeGroup")
  .setOutputCol("AgeGroupIndex")
  .setHandleInvalid("skip")

// Create a VectorAssembler to combine the features into a single vector
val assembler = new VectorAssembler()
  .setInputCols(Array("Fare", "Pclass", "FamSize", "FarePerPerson"))
  .setOutputCol("features")

// Create a RandomForest model
val rf = new RandomForestClassifier()
  .setLabelCol("Survived")
  .setFeaturesCol("features")

// Create a Pipeline to string together the stages
val pipeline = new Pipeline()
  .setStages(Array(sexIndexer, embarkedIndexer, ageGroupIndexer, assembler, rf))
```

```
+------+------+--------------------+
| male|     3|0.13544668587896252|
```

Top screenshot code:

```scala
    .setStages(Array(sexIndexer, embarkedIndexer, ageGroupIndexer, assembler, rf))

// Split the data into training and validation sets
val Array(trainingData, validationData) = train_final.randomSplit(Array(0.8, 0.2))

// Fit the model to the training data
val model = pipeline.fit(trainingData)

// Make predictions on the validation data
val predictions = model.transform(validationData)

// Evaluate the model using the area under the ROC curve
val evaluator = new MulticlassClassificationEvaluator()
  .setLabelCol("Survived")
  .setPredictionCol("prediction")
  .setMetricName("accuracy")

val accuracy = evaluator.evaluate(predictions)

// Print the accuracy
```


Bottom screenshot code:

```scala
  .setMetricName("accuracy")

val accuracy = evaluator.evaluate(predictions)

// Print the accuracy
println(s"Accuracy: $accuracy")

// Use the trained model to make predictions on the test data
val predictions_test = model.transform(test_final)

// Show the predicted survival probabilities for each passenger
predictions_test.select( col = "PassengerId", cols = "prediction", "probability").show()
```

- Result

Screenshot 1 (top):

```
.setMetricName("accuracy")

val accuracy = evaluator.evaluate(predictions)

// Print the accuracy
```

```
2023-04-06 00:37:08,094 INFO  o.a.s.scheduler.TaskSchedulerImpl  - Killing all running tasks in stage 7: Stage finished
2023-04-06 00:37:08,094 INFO  o.a.spark.scheduler.DAGScheduler   - Job 6 finished: describe at SparkAssignment2.scala:55, took 0.095703 s
2023-04-06 00:37:08,173 INFO  o.a.s.c.e.codegen.CodeGenerator    - Code generated in 11.3474 ms
2023-04-06 00:37:08,173 INFO  o.a.s.c.e.codegen.CodeGenerator    - Code generated in 9.2682 ms
+-------+------------------+------------------+-------------------+-----------------+
|summary|               Age|             SibSp|              Parch|             Fare|
+-------+------------------+------------------+-------------------+-----------------+
|  count|               714|               891|                891|              891|
|   mean| 29.69911764705882|0.5230078563411896|0.38159371492704824|32.2042079685746|
| stddev|14.526497332334035|1.1027434322934315| 0.8060572211299488|49.69342859718089|
|    min|              0.42|                 0|                  0|              0.0|
|    max|              80.0|                 8|                  6|         512.3292|
+-------+------------------+------------------+-------------------+-----------------+

2023-04-06 00:37:08,220 INFO  o.a.s.s.e.d.FileSourceStrategy    - Pushed Filters:
2023-04-06 00:37:08,220 INFO  o.a.s.s.e.d.FileSourceStrategy    - Post-Scan Filters:
2023-04-06 00:37:08,220 INFO  o.a.s.s.e.d.FileSourceStrategy    - Output Data Schema: struct<Sex: string>
2023-04-06 00:37:08,316 INFO  o.a.s.c.e.codegen.CodeGenerator    - Code generated in 45.9435 ms
```

Screenshot 2 (bottom):

```
.setMetricName("accuracy")

val accuracy = evaluator.evaluate(predictions)

// Print the accuracy
```

```
2023-04-06 00:37:08,707 INFO  o.a.spark.scheduler.DAGScheduler   - Job 8 is finished. Cancelling potential speculative or zombie tasks for this job
2023-04-06 00:37:08,707 INFO  o.a.s.scheduler.TaskSchedulerImpl  - Killing all running tasks in stage 10: Stage finished
2023-04-06 00:37:08,707 INFO  o.a.spark.scheduler.DAGScheduler   - Job 8 finished: show at SparkAssignment2.scala:58, took 0.036479 s
2023-04-06 00:37:08,716 INFO  o.a.s.c.e.codegen.CodeGenerator    - Code generated in 7.0194 ms
+------+-----+
|   Sex|count|
+------+-----+
|female|  314|
|  male|  577|
+------+-----+

2023-04-06 00:37:08,739 INFO  o.a.s.s.e.d.FileSourceStrategy    - Pushed Filters:
2023-04-06 00:37:08,739 INFO  o.a.s.s.e.d.FileSourceStrategy    - Post-Scan Filters:
2023-04-06 00:37:08,739 INFO  o.a.s.s.e.d.FileSourceStrategy    - Output Data Schema: struct<Pclass: int>
2023-04-06 00:37:08,786 INFO  o.a.s.c.e.codegen.CodeGenerator    - Code generated in 28.595 ms
2023-04-06 00:37:08,801 INFO  o.a.spark.storage.memory.MemoryStore - Block broadcast_16 stored as values in memory (estimated size 189.5 KiB, fre
2023-04-06 00:37:08,801 INFO  o.a.spark.storage.memory.MemoryStore - Block broadcast_16_piece0 stored as bytes in memory (estimated size 32.8 KiB
2023-04-06 00:37:08,801 INFO  o.a.spark.storage.BlockManagerInfo - Added broadcast_16_piece0 in memory on DESKTOP-S60FP07.ht.home:59230 (size: 32
```

Top screenshot:

```
File  Edit  View  Navigate  Code  Refactor  Build  Run  Tools  Git  Window  Help          CSYE7200-Spark-Assignment-2-Hrishikesh

CSYE7200-Spark-Assignment-2-Hrishikesh ) src ) main ) scala ) SparkAssignment2.scala                    Current File ▼

Project
  CSYE7200-Spark-Assignment-2-Hr                         .setMetricName("accuracy")
    .bsp                                         176
    .idea                                        177        val accuracy = evaluator.evaluate(predictions)
    project [CSYE7200-Spark-Assig                178
      target                                     179        // Print the accuracy
    build.properties                                  SparkAssignment2

Run:    SparkAssignment2

2023-04-06 00:37:09,085 INFO  o.a.s.scheduler.TaskSchedulerImpl  - Removed TaskSet 13.0, whose tasks have all completed, from pool
2023-04-06 00:37:09,085 INFO  o.a.spark.scheduler.DAGScheduler  - ResultStage 13 (show at SparkAssignment2.scala:61) finished in 0.031 s
2023-04-06 00:37:09,085 INFO  o.a.spark.scheduler.DAGScheduler  - Job 10 is finished. Cancelling potential speculative or zombie tasks for this jo
2023-04-06 00:37:09,085 INFO  o.a.spark.scheduler.TaskSchedulerImpl  - Killing all running tasks in stage 13: Stage finished
2023-04-06 00:37:09,085 INFO  o.a.spark.scheduler.DAGScheduler  - Job 10 finished: show at SparkAssignment2.scala:61, took 0.032413 s
+------+-----+
|Pclass|count|
+------+-----+
|     1|  216|
|     3|  491|
|     2|  184|
+------+-----+

2023-04-06 00:37:09,117 INFO  o.a.s.s.e.d.FileSourceStrategy  - Pushed Filters:
2023-04-06 00:37:09,117 INFO  o.a.s.s.e.d.FileSourceStrategy  - Post-Scan Filters:
2023-04-06 00:37:09,117 INFO  o.a.s.s.e.d.FileSourceStrategy  - Output Data Schema: struct<Embarked: string>
2023-04-06 00:37:09,148 INFO  o.a.spark.storage.memory.MemoryStore  - Block broadcast_19 stored as values in memory (estimated size 189.5 KiB, fre
2023-04-06 00:37:09,163 INFO  o.a.spark.storage.memory.MemoryStore  - Block broadcast_19_piece0 stored as bytes in memory (estimated size 32.8 KiB
```

```
Git   Run   TODO   Problems   Terminal   Services   sbt shell   Build
Successfully shared project on GitHub: CSYE7200-Spark-Assignment-2-Hrishikesh (23 minutes ago)          1:1  CRLF  UTF-8  2 spaces  master
```

Bottom screenshot:

```
File  Edit  View  Navigate  Code  Refactor  Build  Run  Tools  Git  Window  Help          CSYE7200-Spark-Assignment-2-Hrishikesh

CSYE7200-Spark-Assignment-2-Hrishikesh ) src ) main ) scala ) SparkAssignment2.scala                    Current File ▼

Project
  CSYE7200-Spark-Assignment-2-Hr                         .setMetricName("accuracy")
    .bsp                                         176
    .idea                                        177        val accuracy = evaluator.evaluate(predictions)
    project [CSYE7200-Spark-Assig                178
      target                                     179        // Print the accuracy
    build.properties                                  SparkAssignment2

Run:    SparkAssignment2

2023-04-06 00:37:09,383 INFO  o.a.spark.scheduler.DAGScheduler  - Job 12 is finished. Cancelling potential speculative or zombie tasks for this jo
2023-04-06 00:37:09,383 INFO  o.a.spark.scheduler.TaskSchedulerImpl  - Killing all running tasks in stage 16: Stage finished
2023-04-06 00:37:09,383 INFO  o.a.spark.scheduler.DAGScheduler  - Job 12 finished: show at SparkAssignment2.scala:64, took 0.028416 s
+-------+-----+
|Embarked|count|
+-------+-----+
|      Q|   77|
|   null|    2|
|      C|  168|
|      S|  644|
+-------+-----+

2023-04-06 00:37:09,414 INFO  o.a.s.s.e.d.FileSourceStrategy  - Pushed Filters:
2023-04-06 00:37:09,414 INFO  o.a.s.s.e.d.FileSourceStrategy  - Post-Scan Filters:
2023-04-06 00:37:09,414 INFO  o.a.s.s.e.d.FileSourceStrategy  - Output Data Schema: struct<Pclass: int, Sex: string, Age: double ... 1 more fields
2023-04-06 00:37:09,458 INFO  o.a.s.s.c.e.codegen.CodeGenerator  - Code generated in 28.1833 ms
2023-04-06 00:37:09,458 INFO  o.a.spark.storage.memory.MemoryStore  - Block broadcast_22 stored as values in memory (estimated size 189.5 KiB, fre
2023-04-06 00:37:09,474 INFO  o.a.spark.storage.memory.MemoryStore  - Block broadcast_22_piece0 stored as bytes in memory (estimated size 32.8 KiB
```

```
Git   Run   TODO   Problems   Terminal   Services   sbt shell   Build
Successfully shared project on GitHub: CSYE7200-Spark-Assignment-2-Hrishikesh (23 minutes ago)          1:1  CRLF  UTF-8  2 spaces  master
```

Window 1

File Edit View Navigate Code Refactor Build Run Tools Git Window Help   CSYE7200-Spark-Assignment-2-Hrishikesh

CSYE7200-Spark-Assignment-2-Hrishikesh › src › main › scala › SparkAssignment2.scala

Current File

Project
- CSYE7200-Spark-Assignment-2-H
  - .bsp
  - .idea
  - project [CSYE7200-Spark-Assig
    - target
    - build.properties

SparkAssignment2.scala

```
        .setMetricName("accuracy")

        val accuracy = evaluator.evaluate(predictions)

        // Print the accuracy
```

SparkAssignment2

Run: SparkAssignment2

```
2023-04-06 00:37:09,759 INFO  o.a.s.s.c.e.codegen.CodeGenerator  - Code generated in 6.5381 ms
+------+------+------------------+
|  Sex|Pclass|          mean_age|
+------+------+------------------+
|  male|     3|26.507588932806325|
|female|     3|             21.75|
|female|     1| 34.61176470588235|
|female|     2|28.722972972972972|
|  male|     2| 30.74070707070707|
|  male|     1| 41.28138613861386|
+------+------+------------------+


2023-04-06 00:37:09,838 INFO  o.a.s.s.e.d.FileSourceStrategy  - Pushed Filters:
2023-04-06 00:37:09,838 INFO  o.a.s.s.e.d.FileSourceStrategy  - Post-Scan Filters:
2023-04-06 00:37:09,838 INFO  o.a.s.s.e.d.FileSourceStrategy  - Output Data Schema: struct<Survived: int, Pclass: int, Sex: string ... 1 more fiel
2023-04-06 00:37:09,917 INFO  o.a.s.s.c.e.codegen.CodeGenerator  - Code generated in 43.7192 ms
2023-04-06 00:37:09,917 INFO  o.a.spark.storage.memory.MemoryStore  - Block broadcast_25 stored as values in memory (estimated size 189.5 KiB, fre
2023-04-06 00:37:09,947 INFO  o.a.spark.storage.memory.MemoryStore  - Block broadcast_25_piece0 stored as bytes in memory (estimated size 32.8 KiB
```

Git   Run   TODO   Problems   Terminal   Services   sbt shell   Build

Successfully shared project on GitHub: CSYE7200-Spark-Assignment-2-Hrishikesh (23 minutes ago)   1:1   CRLF   UTF-8   2 spaces   master

5°C Cloudy   ENG IN   01:12 06-04-2023

Window 2

File Edit View Navigate Code Refactor Build Run Tools Git Window Help   CSYE7200-Spark-Assignment-2-Hrishikesh

CSYE7200-Spark-Assignment-2-Hrishikesh › src › main › scala › SparkAssignment2.scala

Current File

Project
- CSYE7200-Spark-Assignment-2-H
  - .bsp
  - .idea
  - project [CSYE7200-Spark-Assig
    - target
    - build.properties

SparkAssignment2.scala

```
        .setMetricName("accuracy")

        val accuracy = evaluator.evaluate(predictions)

        // Print the accuracy
```

SparkAssignment2

Run: SparkAssignment2

```
2023-04-06 00:37:10,220 INFO  o.a.spark.scheduler.DAGScheduler  - ResultStage 22 (show at SparkAssignment2.scala:73) finished in 0.031 s
2023-04-06 00:37:10,220 INFO  o.a.spark.scheduler.DAGScheduler  - Job 16 is finished. Cancelling potential speculative or zombie tasks for this jo
2023-04-06 00:37:10,220 INFO  o.a.s.scheduler.TaskSchedulerImpl  - Killing all running tasks in stage 22: Stage finished
2023-04-06 00:37:10,220 INFO  o.a.spark.scheduler.DAGScheduler  - Job 16 finished: show at SparkAssignment2.scala:73, took 0.029975 s
+------+------+-------------------+
|  Sex|Pclass|      survival_rate|
+------+------+-------------------+
|  male|     3|0.13544668587896252|
|female|     3|                0.5|
|female|     1| 0.9680851063829787|
|female|     2| 0.9210526315789473|
|  male|     2| 0.1574074074074074|
|  male|     1|0.36885245901639346|
+------+------+-------------------+


2023-04-06 00:37:10,636 INFO  o.a.s.s.e.d.FileSourceStrategy  - Pushed Filters:
2023-04-06 00:37:10,636 INFO  o.a.s.s.e.d.FileSourceStrategy  - Post-Scan Filters:
2023-04-06 00:37:10,636 INFO  o.a.s.s.e.d.FileSourceStrategy  - Output Data Schema: struct<PassengerId: int, Survived: int, Pclass: int, Name: str
```

Git   Run   TODO   Problems   Terminal   Services   sbt shell   Build

Successfully shared project on GitHub: CSYE7200-Spark-Assignment-2-Hrishikesh (24 minutes ago)   6:77   CRLF   UTF-8   2 spaces   master
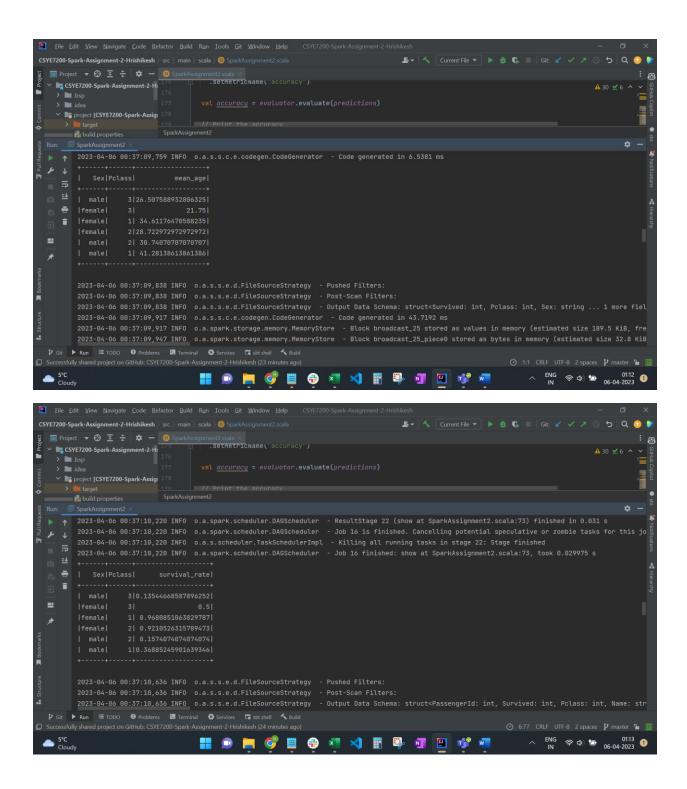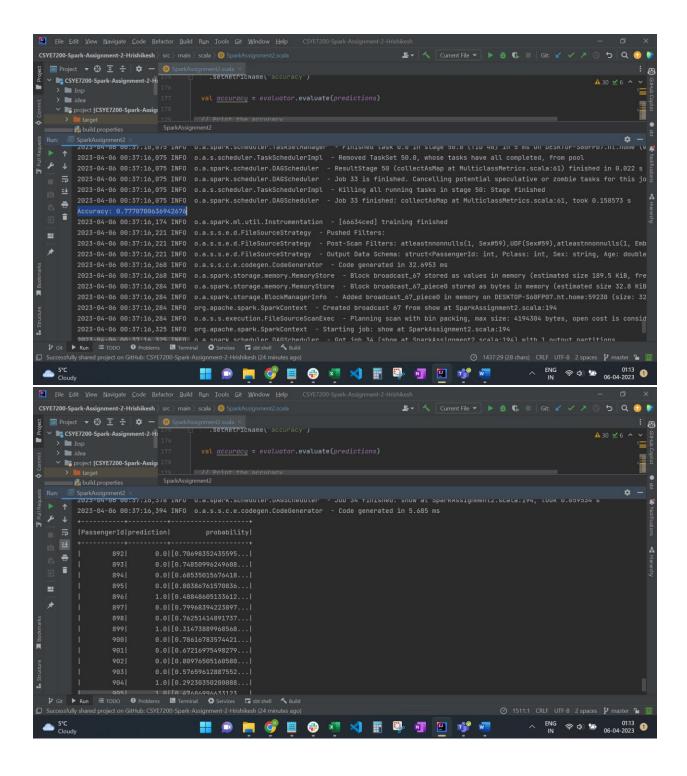
5°C Cloudy   ENG IN   01:13 06-04-2023

-GitHub link

https://github.com/hrishikesh-pawar/CSYE7200-Spark-Assignment-2-Hrishikesh