

Shangqing Hu – SEC01 (NUID 001374342)

Big Data System Engineering with Scala
Spring 2023
Assignment No.7(Spark-CSV)



-List of Tasks Implemented

You are required to analyze a movie rating dataset. The data is stored in a CSV file (either use the one in the repository or download the latest from Kaggle). You need to read this file into spark and calculate the mean rating and standard deviation for all movies. There is no test case provided for you, so you need to write your own test cases to ensure that at least your program works well.

-Code

```
package edu.neu.coe.csye7200.csv

import org.apache.spark.sql.{DataFrame, SparkSession, functions}
case class MovieRatingAnalyzer(resource: String) {

  val spark: SparkSession = SparkSession
    .builder()
    .appName( name = "MovieRating")
    .master( master = "local[*]")
    .getOrCreate()

  spark.sparkContext.setLogLevel("ERROR")

  def apply(resource: String): MovieRatingAnalyzer = new MovieRatingAnalyzer(resource)

  private val df: DataFrame = spark.read.format( source = "csv").option("header", "true")
    .load(getClass.getResource(resource).getPath)
  df.show()

  def movieRatingMean(): Double = {
    val mean = df.select(functions.avg( columnName = "imdb_score"))
    mean.show()
    mean.first().getDouble(0)
  }

  def movieRatingSD(): Double = {
    val sd = df.select(functions.stddev( columnName = "imdb_score"))
    sd.show()
    sd.first().getDouble(0)
  }
}

object MovieRatingAnalyzer extends App {
  val movie = MovieRatingAnalyzer("/movie_metadata.csv")
  movie.movieRatingMean()
  movie.movieRatingSD()
}
```

```
package edu.neu.coe.csye7200.csv
```

```
import org.scalatest.flatspec.AnyFlatSpec
```

```
import org.scalatest.matchers.should.Matchers
```

```
|
```

```
class MovieRatingAnalyzerTest extends AnyFlatSpec with Matchers {
```

```
  behavior of "movie rating statistics"
```

```
  it should "get movie rating mean and standard deviation" in {
```

```
    val movie = MovieRatingAnalyzer("/movie_metadata.csv")
```

```
    movie.movieRatingMean() shouldBe 6.453200745804848
```

```
    movie.movieRatingSD() shouldBe 0.9988071293753289
```

```
  }
```

```
}
```

-Unit tests

The screenshot shows the IntelliJ IDEA IDE with a Scala test file named `MovieRatingAnalyzerTest.scala` in the `CSYE7200 - MovieRatingAnalyzerTest.scala [SparkCsv]` project. The test file contains the following code:

```
1 package edu.neu.coe.csye7200.csv
2
3 import org.scalatest.FlatSpec
4 import org.scalatest.matchers.should.Matchers
5
6
7 class MovieRatingAnalyzerTest extends FlatSpec with Matchers {
8
9   behavior of "movie rating statistics"
10
11   it should "get movie rating mean and standard deviation" in {
12
13     val movie = MovieRatingAnalyzer("/movie_metadata.csv")
14     movie.movieRatingMean() shouldBe 6.453200745804848
15     movie.movieRatingSD() shouldBe 0.9988071293753289
16   }
17
18 }
19
```

The test results are shown in the bottom panel, indicating that the test passed successfully. The test results are as follows:

Test Results	Duration	Message
MovieRatingAnalyzerTest	7 sec 384 ms	Testing started at 8:26 PM ...
movie rating statistics	7 sec 384 ms	
should get movie rating mean and standard deviation	7 sec 384 ms	

```
2023-03-28 20:27:46,953 INFO o.a.s.s.c.e.codegen.CodeGenerator - Code generated in 34.635317 ms
+-----+
|director_name|num_critic_for_reviews|duration|director_facebook_likes|actor_3_facebook_likes|actor_2_name|actor_1_facebook_likes|gross|genres|actor_1_name|mov|
+-----+
|James Cameron|723|178|0|855|Joel David Moore|1000|740905847|Action|Adventure|...|CCH Pounder|
|Gore Verbinski|302|169|563|1000|Orlando Bloom|4000|389404152|Action|Adventure|...|Johnny Depp|Pirates of t
|Sam Mendes|602|148|0|161|Rory Kinnear|11000|280074175|Action|Adventure|...|Christoph Waltz|
|Christopher Nolan|813|164|22000|23000|Christian Bale|27000|448130642|Action|Thriller|Tom Hardy|The Dark Kni
|Doug Walker|null|null|131|null|Rob Walker|131|null|Documentary|Doug Walker|Star Wars: E
|Andrew Stanton|442|132|475|530|Samantha Morton|640|73058679|Action|Adventure|...|Daryl Sabara|John
|Sam Raimi|392|156|0|4000|James Franco|24000|336530303|Action|Adventure|...|J.K. Simmons|Spide
|Nathan Greno|324|100|15|284|Donna Murphy|799|200807262|Adventure|Animati...|Brad Garrett|
|Joss Whedon|635|141|0|19000|Robert Downey Jr.|26000|458991599|Action|Adventure|...|Chris Hemsworth|Avengers: Ag
|David Yates|375|153|282|10000|Daniel Radcliffe|25000|301956980|Adventure|Family|...|Alan Rickman|Harry Potter
|Zack Snyder|673|183|0|2000|Lauren Cohan|15000|330249062|Action|Adventure|...|Henry Cavill|Batman v Sup
|Bryan Singer|434|169|0|903|Marlon Brando|18000|200069408|Action|Adventure|...|Kevin Spacey|Superman
|Marc Forster|403|106|395|393|Mathieu Amalric|451|168368427|Action|Adventure|Giancarlo Giannini|Quantum of
|Gore Verbinski|313|151|563|1000|Orlando Bloom|4000|423012628|Action|Adventure|...|Johnny Depp|Pirates of t
|Gore Verbinski|450|150|563|1000|Ruth Wilson|4000|89289910|Action|Adventure|...|Johnny Depp|The Lone
|Zack Snyder|733|143|0|748|Christopher Meloni|15000|291021565|Action|Adventure|...|Henry Cavill|Man o
|Andrew Adamson|258|150|80|201|Pierfrancesco Favino|22000|141614023|Action|Adventure|...|Peter Dinklage|The Chroni
|Joss Whedon|703|173|0|19000|Robert Downey Jr.|26000|623279547|Action|Adventure|...|Chris Hemsworth|The A
|Rob Marshall|448|136|252|1000|Sam Claflin|40000|241063875|Action|Adventure|...|Johnny Depp|Pirates of t
|Barry Sonnenfeld|451|106|100|718|Michael Stuhlbarg|10000|179020854|Action|Adventure|...|Will Smith|Men in
```

```
2023-03-28 20:27:47,424 INFO o.a.s.s.c.e.codegen.CodeGenerator - Code generated in 5.482931 ms
+-----+
| avg(imdb_score) |
+-----+
| 6.453200745804848 |
+-----+
```

```
2023-03-28 20:27:47,759 INFO o.a.spark.scheduler.DAGScheduler - Job 4 finished: show at MovieRatingAnalyzer.scala:28, took 0.080085 s
+-----+
| stddev_samp(imdb_score) |
+-----+
| 0.9988071293753289 |
+-----+
```