Lab 4

August 20, 2024

```
In [1]: from pyspark.sql import SparkSession
        from pyspark.ml.recommendation import ALS
        from pyspark.ml.evaluation import RegressionEvaluator
        from pyspark.ml.feature import StringIndexer
        from pyspark.sql.functions import col
```

```
In [2]: spark = SparkSession.builder \
            .appName("Lab 4") \
            .getOrCreate()
```

```
/home/lplab/anaconda3/lib/python3.7/site-packages/pyspark/context.py:317: FutureWarning: Python
  warnings.warn("Python 3.7 support is deprecated in Spark 3.4.", FutureWarning)
```

0.1 Q1

```
In [3]: df = spark.read.json('./movies 1.json')
        df.printSchema()
        df.show()
```

```
root
 |-- helpfulness: string (nullable = true)
 |-- product_id: string (nullable = true)
 |-- profile_name: string (nullable = true)
 |-- review: string (nullable = true)
 |-- score: double (nullable = true)
 |-- summary: string (nullable = true)
 |-- time: long (nullable = true)
 |-- user_id: string (nullable = true)

+-----------+----------+--------------------+--------------------+-----+--------------------+--
|helpfulness|product_id|        profile_name|              review|score|             summary|
+-----------+----------+--------------------+--------------------+-----+--------------------+--
|        7/7|B003AI2VGA|Brian E. Erland "...|Synopsis: On the ...|  3.0|"There Is So Much...|1
|        4/4|B003AI2VGA|          Grady Harp|THE VIRGIN OF JUA...|  3.0|Worthwhile and Im...|1
|       8/10|B003AI2VGA|Chrissy K. McVay ...|The scenes in thi...|  5.0|This movie needed...|1
|        1/1|B003AI2VGA|        golgotha.gov|THE VIRGIN OF JUA...|  3.0|distantly based o...|1
|        1/1|B003AI2VGA|KerrLines "&#34;M...|Informationally, ...|  3.0|"What's going on ...|1
|        0/0|B003AI2VGA|abra "a devoted r...|The murders in Ju...|  2.0|Pretty pointless ...|12
|       3/11|B003AI2VGA| Charles R. Williams|Mexican men are m...|  1.0|This is junk, sta...|11
|      64/65|B00006HAXW|   Anthony Accordino|Over the past few...|  5.0|A  Rock N Roll Hi...|10
|      26/26|B00006HAXW|    Joseph P. Aiello|I recvd this vide...|  5.0|A  MUST-HAVE  vid...|10
|      24/24|B00006HAXW|     "bruce_from_la"|Wow! When I saw t...|  5.0|If You Like DooWo...|10
|      22/23|B00006HAXW|    Henrique Peirano|I have the Doo Wo...|  4.0|    I expected more.|10
|      14/14|B00006HAXW|      Richard Albero|Having worked in ...|  5.0|Professional Exce...|10
|        9/9|B00006HAXW|                 Les|The people who ha...|  5.0|Marvelous, just M...|10
|        9/9|B00006HAXW|     Joseph M. Kotow|I have all of the...|  5.0|Pittsburgh - Home...|10
|        7/7|B00006HAXW|      "fellafromnyc"|The performance o...|  4.0|They sang in the ...|10
|        7/7|B00006HAXW|           S. Dorman|Get it, also get ...|  5.0|DOO WOP RECORDED ...|10
|        7/7|B00006HAXW|                 RFP|Excellent, excell...|  5.0|ROCK RYTHM AND DO...|10
|        4/4|B00006HAXW|           C. Thomas|This video is awe...|  5.0|Unbelievable Best...|11
|        3/3|B00006HAXW|   Michael A. Martin|As I stated in my...|  5.0|Another outstandi...|12
|        5/6|B00006HAXW|C. W. Emblom "Bil...|I own both the VH...|  5.0|Outstanding Wheth...|10
+-----------+----------+--------------------+--------------------+-----+--------------------+--
only showing top 20 rows
```

0.2 Q3

```
In [4]: user_indexer = StringIndexer(inputCol="user_id", outputCol="userIndex",handleInvalid="
        df = user_indexer.fit(df).transform(df)


        product_indexer = StringIndexer(inputCol="product_id", outputCol="productIndex", )
        df = product_indexer.fit(df).transform(df)
```

```
In [5]: df = df.select(
            col("userIndex").alias("userId"),
            col("productIndex").alias("itemId"),
            col("score").alias("rating")
        )
```

0.3 Q2

```
In [6]: (training_data, test_data) = df.randomSplit([0.6, 0.4])
```

0.4 Q3

```
In [12]: als = ALS(
             maxIter=25,
             regParam=0.01,
             userCol="userId",
             itemCol="itemId",
             ratingCol="rating",
             coldStartStrategy="drop"
```

```
        )
        model = als.fit(training_data)
        predictions = model.transform(test_data)
```

0.5 Q4

```
In [13]: rmse_evaluator = RegressionEvaluator(
             metricName="rmse",
             labelCol="rating",
             predictionCol="prediction"
         )

         # Calculate RMSE
         rmse = rmse_evaluator.evaluate(predictions)
         print(f"Root Mean Squared Error (RMSE) = {rmse}")
```

```
Root Mean Squared Error (RMSE) = 3.1854866132459456
```

```
In [14]: mae_evaluator = RegressionEvaluator(
             metricName="mae",
             labelCol="rating",
             predictionCol="prediction"
         )

         # Calculate MAE
         mae = mae_evaluator.evaluate(predictions)
         print(f"Mean Absolute Error (MAE) = {mae}")
```

```
Mean Absolute Error (MAE) = 2.3648656755003263
```

```
In [15]: predictions_with_actuals = predictions.select(
             col("userId"),
             col("itemId"),
             col("rating"),
             col("prediction")
         )

         # Show the first few rows of predictions with actual ratings
         predictions_with_actuals.show(10)
```

```
+------+------+------+----------+
|userId|itemId|rating|prediction|
+------+------+------+----------+
|5803.0|  21.0|   2.0| 1.6464703|
| 243.0|  21.0|   2.0| 1.9394901|
| 540.0|   7.0|   5.0|0.76804197|
|4161.0|  63.0|   5.0| 0.5079631|
|1143.0|  85.0|   4.0| 1.2303276|
|1339.0| 680.0|   4.0| 3.9416995|
|2393.0|  21.0|   5.0| 4.1161757|
|3352.0|  21.0|   4.0| 3.2929406|
|5670.0|   7.0|   5.0| 1.5820978|
|1005.0|   7.0|   1.0| 0.9977566|
+------+------+------+----------+
only showing top 10 rows
```

```
In [ ]:
```