732A54 Big Data Analytics

Lab 2 - Spark SQL

Submitted by:
Group G6 –
Hoda Fakharzadehjahromy (hodfa840)
Syeda Aqsa Iftekhar (syeif776)

Assignments

1. year, station with the max, maxValue ORDER BY maxValue DESC year, station with the min, minValue ORDER BY minValue DESC

Code:

```
from pyspark import SparkContext
import os
import sys
from pyspark.sql import SQLContext, Row
from pyspark.sql import functions as F

os.environ['PYSPARK_PYTHON'] = sys.executable
os.environ['PYSPARK_DRIVER_PYTHON'] = sys.executable
sc = SparkContext(appName="ex2" )
```

```
# Data
temperature = spark.read.csv("file:///home/x_syeif/input_data/temperature-readings.csv", header = False, sep = ';' )
temperature = temperature.withColumnRenamed("_c0", "stationNum")\
                                .withColumnRenamed("_c1", "date")\
                                .withColumnRenamed("_c2", "time")\
                                .withColumnRenamed("_c3", "airTemp")\
                                .withColumnRenamed("_c4", "quality")


precipitation = spark.read.csv("file:///home/x_syeif/input_data/precipitation-readings.csv", header = False, sep = ';' )
precipitation = precipitation.withColumnRenamed("_c0", "stationNum")\
                                .withColumnRenamed("_c1", "date")\
                                .withColumnRenamed("_c2", "time")\
                                .withColumnRenamed("_c3", "precip")\
                                .withColumnRenamed("_c4", "quality")

stations = sc.textFile("file:///home/x_syeif/input_data/stations-Ostergotland.csv")\
                            .map(lambda line: line.split(";"))\
                            .map(lambda line:line[0])
```

```
#Q1
filtered = temperature.select("stationNum", F.year(F.col('date')).alias("year"),\
                                    F.col("airTemp").cast("float"))\
                              .filter((F.col("year")>=1950) & ((F.col("year")<=2014)))

tempmin = filtered.groupBy("year").agg(F.min('airTemp').alias('MinTemp')).orderBy("year")

tempmax = filtered.groupBy("year").agg(F.max('airTemp').alias('MaxTemp')).orderBy("year")

#out.coalesce(1).write.csv("file:///home/x_kesma/Lab1/input_data/results/BDA_LAB2/Q1",sep=",", header=True)

#print("--------------------------------------------------")
tempmin_rdd = tempmin.rdd
tempmin_rdd.coalesce(1).saveAsTextFile("file:///home/x_syeif/Lab_2_Results/ex2_q1_min")

tempmax_rdd  = tempmax.rdd
tempmax_rdd.coalesce(1).saveAsTextFile("file:///home/x_syeif/Lab_2_Results/ex2_q1_max")
```

Output:

maxValue (first 25 rows)

```
Row(Year=1950, MaxTemp=29.399999618530273)
Row(Year=1951, MaxTemp=28.5)
Row(Year=1952, MaxTemp=30.399999618530273)
Row(Year=1953, MaxTemp=32.20000076293945)
Row(Year=1954, MaxTemp=30.5)
Row(Year=1955, MaxTemp=32.20000076293945)
Row(Year=1956, MaxTemp=33.0)
Row(Year=1957, MaxTemp=29.799999237060547)
Row(Year=1958, MaxTemp=30.799999237060547)
Row(Year=1959, MaxTemp=32.79999923706055)
Row(Year=1960, MaxTemp=29.399999618530273)
Row(Year=1961, MaxTemp=31.0)
Row(Year=1962, MaxTemp=27.399999618530273)
Row(Year=1963, MaxTemp=31.0)
Row(Year=1964, MaxTemp=31.200000762939453)
Row(Year=1965, MaxTemp=28.5)
Row(Year=1966, MaxTemp=33.5)
Row(Year=1967, MaxTemp=29.5)
Row(Year=1968, MaxTemp=33.70000076293945)
Row(Year=1969, MaxTemp=32.0)
Row(Year=1970, MaxTemp=33.20000076293945)
Row(Year=1971, MaxTemp=31.200000762939453)
Row(Year=1972, MaxTemp=31.200000762939453)
Row(Year=1973, MaxTemp=32.20000076293945)
Row(Year=1974, MaxTemp=30.600000381469727)
```

minValue (first 25 rows)

```
Row(Year=1950, MinTemp=-42.0)
Row(Year=1951, MinTemp=-42.0)
Row(Year=1952, MinTemp=-35.5)
Row(Year=1953, MinTemp=-38.400001525878906)
Row(Year=1954, MinTemp=-36.0)
Row(Year=1955, MinTemp=-41.20000076293945)
Row(Year=1956, MinTemp=-45.0)
Row(Year=1957, MinTemp=-37.79999923706055)
Row(Year=1958, MinTemp=-43.0)
Row(Year=1959, MinTemp=-43.599998474121094)
Row(Year=1960, MinTemp=-40.0)
Row(Year=1961, MinTemp=-39.5)
Row(Year=1962, MinTemp=-42.0)
Row(Year=1963, MinTemp=-41.0)
Row(Year=1964, MinTemp=-39.5)
Row(Year=1965, MinTemp=-44.0)
Row(Year=1966, MinTemp=-49.400001525878906)
Row(Year=1967, MinTemp=-45.400001525878906)
Row(Year=1968, MinTemp=-42.0)
Row(Year=1969, MinTemp=-41.5)
Row(Year=1970, MinTemp=-39.599998474121094)
Row(Year=1971, MinTemp=-44.29999923706055)
Row(Year=1972, MinTemp=-37.5)
Row(Year=1973, MinTemp=-39.29999923706055)
Row(Year=1974, MinTemp=-35.599998474121094)
```

2. year, month, value ORDER BY value DESC
year, month, value ORDER BY value DESC

Code:

```
#Q2
filtered = temperature.select("stationNum", F.year(F.col('date')).alias("year"),\
                                    F.month(F.col("date")).alias("month"),\
                                    F.col("airTemp").cast("float"))

filtered = filtered.filter(((F.col("year")>=1950) & ((F.col("year")<=2014))) &(F.col("airTemp")>10))

tempcount = filtered.groupBy("year", "month")\
             .agg(F.count("stationNum").alias("res"))\
             .orderBy("res",ascending=False)

#print("-----------------------------------------------------")
tempcount_rdd = tempcount.rdd
tempcount_rdd.coalesce(1).saveAsTextFile("file:///home/x_syeif/Lab_2_Results/ex2_q2_count")


tempdist = filtered.groupBy("year", "month")\
             .agg(F.countDistinct("stationNum").alias("res"))\
             .orderBy("res",ascending=False)

tempdist_rdd  = tempdist.rdd
tempdist_rdd.coalesce(1).saveAsTextFile("file:///home/x_syeif/Lab_2_Results/ex2_q2_dist")
```

Output:

Count (first 25 rows)

```
Row(year=2014, month=7, res=147681)
Row(year=2011, month=7, res=146656)
Row(year=2010, month=7, res=143419)
Row(year=2012, month=7, res=137477)
Row(year=2013, month=7, res=133657)
Row(year=2009, month=7, res=133008)
Row(year=2011, month=8, res=132734)
Row(year=2009, month=8, res=128349)
Row(year=2013, month=8, res=128235)
Row(year=2003, month=7, res=128133)
Row(year=2002, month=7, res=127956)
Row(year=2006, month=8, res=127622)
Row(year=2008, month=7, res=126973)
Row(year=2002, month=8, res=126073)
Row(year=2005, month=7, res=125294)
Row(year=2011, month=6, res=125193)
Row(year=2012, month=8, res=125037)
Row(year=2006, month=7, res=124794)
Row(year=2010, month=8, res=124417)
Row(year=2014, month=8, res=124045)
Row(year=1997, month=7, res=123496)
Row(year=2007, month=7, res=123218)
Row(year=2013, month=6, res=122181)
Row(year=1997, month=8, res=121154)
Row(year=2001, month=7, res=120529)
```

Distinct (first 25 rows)

```
Row(year=1972, month=10, res=378)
Row(year=1973, month=5, res=377)
Row(year=1973, month=6, res=377)
Row(year=1973, month=9, res=376)
Row(year=1972, month=8, res=376)
Row(year=1972, month=6, res=375)
Row(year=1972, month=5, res=375)
Row(year=1971, month=8, res=375)
Row(year=1972, month=9, res=375)
Row(year=1971, month=6, res=374)
Row(year=1971, month=9, res=374)
Row(year=1972, month=7, res=374)
Row(year=1971, month=5, res=373)
Row(year=1973, month=8, res=373)
Row(year=1974, month=8, res=372)
Row(year=1974, month=6, res=372)
Row(year=1974, month=9, res=370)
Row(year=1970, month=8, res=370)
Row(year=1973, month=7, res=370)
Row(year=1974, month=5, res=370)
Row(year=1971, month=7, res=370)
Row(year=1970, month=6, res=369)
Row(year=1975, month=9, res=369)
Row(year=1976, month=5, res=369)
Row(year=1970, month=9, res=369)
```

3. year, month, station, avgMonthlyTemperature ORDER BY avgMonthlyTemperature DESC

Code:

```
#Q3
filtered = temperature.select("stationNum", F.year(F.col('date')).alias("year"),\
                                F.month(F.col("date")).alias("month"),F.col("airTemp").cast("float"))

filtered = filtered.filter((F.col("year")>=1960) & ((F.col("year")<=2014)))

## Fix: Calculating Avg by Sum(AirTemp)/number of (values) instead of using the Avg Key word
tempavg = filtered.groupBy('year','month', 'stationNum').agg((F.sum('airTemp')/F.count('airTemp'))
                                    .alias('avgMonthlyTemperature'))\.orderBy("stationNum","year","month",ascending=False)

#print("-----------------------------------------------------")
tempavg_rdd = tempavg.rdd
tempavg_rdd.coalesce(1).saveAsTextFile("file:///home/x_syeif/Lab_2_Results/ex2_q3_avg")
```

Output:

```
Row(year=2014, month=12, stationNum=u'99450', avgMonthlyTemperature=1.989784944191655)
Row(year=2014, month=11, stationNum=u'99450', avgMonthlyTemperature=5.973888883491357)
Row(year=2014, month=10, stationNum=u'99450', avgMonthlyTemperature=9.300811913726456)
Row(year=2014, month=9, stationNum=u'99450', avgMonthlyTemperature=13.712222230434417)
Row(year=2014, month=8, stationNum=u'99450', avgMonthlyTemperature=16.91505377651543)
Row(year=2014, month=7, stationNum=u'99450', avgMonthlyTemperature=18.455510759866367)
Row(year=2014, month=6, stationNum=u'99450', avgMonthlyTemperature=11.006944455703099)
Row(year=2014, month=5, stationNum=u'99450', avgMonthlyTemperature=7.565456981700595)
Row(year=2014, month=4, stationNum=u'99450', avgMonthlyTemperature=4.473472221982148)
Row(year=2014, month=3, stationNum=u'99450', avgMonthlyTemperature=2.797446236154565)
Row(year=2014, month=2, stationNum=u'99450', avgMonthlyTemperature=1.833333333244636)
Row(year=2014, month=1, stationNum=u'99450', avgMonthlyTemperature=-0.9764784909985078)
Row(year=2013, month=12, stationNum=u'99450', avgMonthlyTemperature=3.6639077343360054)
Row(year=2013, month=11, stationNum=u'99450', avgMonthlyTemperature=5.528194454602069)
Row(year=2013, month=10, stationNum=u'99450', avgMonthlyTemperature=9.18629032725929)
Row(year=2013, month=9, stationNum=u'99450', avgMonthlyTemperature=13.620972209506565)
Row(year=2013, month=8, stationNum=u'99450', avgMonthlyTemperature=17.183333362302474)
Row(year=2013, month=7, stationNum=u'99450', avgMonthlyTemperature=15.397849473901974)
Row(year=2013, month=6, stationNum=u'99450', avgMonthlyTemperature=13.851805555820466)
Row(year=2013, month=5, stationNum=u'99450', avgMonthlyTemperature=8.73279569386154)
Row(year=2013, month=4, stationNum=u'99450', avgMonthlyTemperature=2.6151388929122024)
Row(year=2013, month=3, stationNum=u'99450', avgMonthlyTemperature=-2.2607526867900805)
Row(year=2013, month=2, stationNum=u'99450', avgMonthlyTemperature=-0.6566964270813125)
Row(year=2013, month=1, stationNum=u'99450', avgMonthlyTemperature=-1.9122311783333619)
Row(year=2012, month=12, stationNum=u'99450', avgMonthlyTemperature=-0.9508064506595494)
```

4. station, maxTemp, maxDailyPrecipitation ORDER BY station DESC

Code:

```
#Q4

# MAP
temperature1 = sc.textFile("file:///home/x_syeif/input_data/temperature-readings.csv")
lines = temperature1.map(lambda x: x.split(";"))

#converting lines to rows for temperature
convertedrows = lines.map(lambda x: Row(station=x[0], year=x[1].split("-")[0],
month=x[1].split("-")[1],day=x[1].split("-")[2], time=x[2], temperature=float(x[3]), quality=x[4] ))

TempReadings = sqlContext.createDataFrame(convertedrows)
TempReadings.registerTempTable("convertedrows_sql")

#spark.sql("select max(temperature) as maxTemp from convertedrows_sql")

tempmax = schemaTempReadings.groupBy('station').agg(F.max('temperature').alias('maxTemperature'))\
.orderBy(['maxTemperature'],ascending=False)

filtertemperature = tempmax.filter((F.col("maxTemperature") > 25)  & ( F.col("maxTemperature") < 30))


precipitation = sc.textFile("file:///home/x_syeif/input_data/precipitation-readings.csv")
lines = precipitation.map(lambda x: x.split(";"))

#converting lines to rows for precipitation
convertedrows1 = lines.map(lambda x: Row(station=x[0], year=x[1].split("-")[0],
month=x[1].split("-")[1],day=x[1].split("-")[2], time=x[2], precipitation=float(x[3]), quality=x[4] ))
```

```
PrecipReadings = sqlContext.createDataFrame(convertedrows1)
PrecipReadings.registerTempTable("convertedrows1")

## Fix: Added Summation of precipitation per station before taking max
prepsum = PrecipReadings.groupBy('station').agg(F.sum('precipitation').alias('precipitation'))

precpmax = prepsum.groupBy('station').agg(F.max('precipitation').alias('maxPrec')).orderBy(['maxPrec'],ascending=False)

filterprec = precpmax.filter((F.col("maxPrec") > 100 ) & (F.col("maxPrec") < 200))

jointempprec = filtertemperature.join(filterprec, on="station",how="inner").select("station", "maxTemperature", "maxPrec")
                            .orderBy(["station"], ascending=False)


#print("-----------------------------------------------------")
finaltempprec_rdd = jointempprec.rdd
finaltempprec_rdd.coalesce(1).saveAsTextFile("file:///home/x_syeif/Lab_2_Results/ex2_q4_filteredtempprec")
```

Output:

The output is empty.

5. year, month, avgMonthlyPrecipitation ORDER BY year DESC, month DESC

Code:

```
#Q5

precipitation = sc.textFile("file:///home/x_syeif/input_data/precipitation-readings.csv")
lines = precipitation.map(lambda x: x.split(";"))

#converting lines to rows for precipitation
convertedrows1 = lines.map(lambda x: Row(station=x[0], year=x[1].split("-")[0],
month=x[1].split("-")[1],day=x[1].split("-")[2], time=x[2], precipitation=float(x[3]), quality=x[4] ))

PrecipReadings = sqlContext.createDataFrame(convertedrows1)
PrecipReadings.registerTempTable("convertedrows1")


filterprec = PrecipReadings.filter(PrecipReadings["year"].between("1993", "2016"))


precpsum = filterprec.groupBy('station', 'year', 'month').agg(F.sum('precipitation')
                    .alias('sumprec')).orderBy(['year', 'month'],ascending=[0,0])

Osterstation = sc.textFile("file:///home/x_syeif/input_data/stations-Ostergotland.csv")
lines = Osterstation.map(lambda x: x.split(";"))

#converting lines to rows for stations
convertedrowstation = lines.map(lambda x: Row(stnumber=x[0], stname=x[1], stheight=float(x[2]),
                                        stlatitude=float(x[3]), stlongitude=float(x[4])))


OsterReadings = sqlContext.createDataFrame(convertedrowstation)
OsterReadings.registerTempTable("convertedrowstation")
```

```
#Join
joinprecstation = precpsum.join(OsterReadings, precpsum["station"] ==OsterReadings["stnumber"], how="inner")
                  .groupBy("year", "month").agg(F.mean("sumprec").alias("precavg"))
                  .orderBy(["year","month"], ascending=False).select("year", "month", "precavg")


#print("-----------------------------------------------------")
joinprecstation_rdd = joinprecstation.rdd
joinprecstation_rdd.coalesce(1).saveAsTextFile("file:///home/x_syeif/Lab_2_Results/ex2_q5_avg")
```

Output:

```
Row(year=u'2016', month=u'07', precavg=0.0)
Row(year=u'2016', month=u'06', precavg=47.6625)
Row(year=u'2016', month=u'05', precavg=29.250000000000007)
Row(year=u'2016', month=u'04', precavg=26.900000000000006)
Row(year=u'2016', month=u'03', precavg=19.962500000000006)
Row(year=u'2016', month=u'02', precavg=21.5625)
Row(year=u'2016', month=u'01', precavg=22.325000000000003)
Row(year=u'2015', month=u'12', precavg=28.925)
Row(year=u'2015', month=u'11', precavg=63.88750000000002)
Row(year=u'2015', month=u'10', precavg=2.2625)
Row(year=u'2015', month=u'09', precavg=101.3)
Row(year=u'2015', month=u'08', precavg=26.987499999999997)
Row(year=u'2015', month=u'07', precavg=119.09999999999994)
Row(year=u'2015', month=u'06', precavg=78.66250000000002)
Row(year=u'2015', month=u'05', precavg=93.225)
Row(year=u'2015', month=u'04', precavg=15.337499999999999)
Row(year=u'2015', month=u'03', precavg=42.61250000000001)
Row(year=u'2015', month=u'02', precavg=24.825)
Row(year=u'2015', month=u'01', precavg=59.11250000000003)
Row(year=u'2014', month=u'12', precavg=35.46250000000001)
Row(year=u'2014', month=u'11', precavg=52.425000000000054)
Row(year=u'2014', month=u'10', precavg=72.13749999999999)
Row(year=u'2014', month=u'09', precavg=48.45000000000001)
Row(year=u'2014', month=u'08', precavg=90.81249999999997)
Row(year=u'2014', month=u'07', precavg=22.987500000000004)
```