Simja649 Danko376

# Lab 2 - Spark SQL

In this lab we complete all of the tasks from laboration 1 but instead of just using Spark we are now solving them using Spark SQL.

## Assignment 1

The output for this is ordered first by maximum temperatures and then by minimum temperatures, both in descending order.

We extract the data needed, we create a dataframe using the relevant data. We use the filter function to get rid of readings outside of our desired time period. Then we use groupBy and aggregate (agg function) using built in max/min functions to get the temperatures that we are interested in. In the last row we sort the resulting table in descending order. 1a shows our results for maximum temperatures, 1b shows our results for the minimum temperatures.

1a

| year | station | value |
|---|---|---|
| 1975 | 86200 | 36.1 |
| 1992 | 63600 | 35.4 |
| 1994 | 117160 | 34.7 |
| 2010 | 75250 | 34.4 |
| 2014 | 96560 | 34.4 |
| 1989 | 63050 | 33.9 |
| 1982 | 94050 | 33.8 |
| 1968 | 137100 | 33.7 |
| 1966 | 151640 | 33.5 |
| 2002 | 78290 | 33.3 |
| 1983 | 98210 | 33.3 |
| 2002 | 78290 | 33.3 |
| 1986 | 76470 | 33.2 |
| 1970 | 103080 | 33.2 |
| 1956 | 145340 | 33.0 |
| 2000 | 62400 | 33.0 |
| 1959 | 65160 | 32.8 |
| 2006 | 75240 | 32.7 |
| 1991 | 137040 | 32.7 |
| 1988 | 102540 | 32.6 |

1b

| year | station | value |
|---|---|---|
| 1990 | 166870 | -35.0 |
| 1990 | 147270 | -35.0 |
| 1952 | 192830 | -35.5 |
| 1974 | 179950 | -35.6 |
| 1974 | 166870 | -35.6 |
| 1954 | 113410 | -36.0 |
| 1992 | 179960 | -36.1 |
| 1975 | 157860 | -37.0 |
| 1972 | 167860 | -37.5 |
| 1995 | 182910 | -37.6 |
| 2000 | 169860 | -37.6 |
| 1957 | 159970 | -37.8 |
| 1983 | 191900 | -38.2 |
| 1989 | 166870 | -38.2 |
| 1953 | 183760 | -38.4 |
| 2009 | 179960 | -38.5 |
| 1993 | 191900 | -39.0 |
| 1984 | 123480 | -39.2 |
| 1984 | 191900 | -39.2 |
| 1991 | 179960 | -39.3 |

## Assignment 2

In task 2a we create a dataframe with each 7 rows. In order, station, yearmonth (so each year with the month ‘1996-03’), month, year, time, value (temperature) and quality. We first filter so we only get year 1950-2014 and degrees over 10. Then using grouping by the year and month, we use the count function on our yearmonth row to get the count of month of each year. Then we order them by the count.

In task 2b we are doing practically the same. However, instead of using the count function we are using countDistinct with both yearmonth row and the station row to get the distinct stations once. See results below.
2a

| year | month | count |
|---|---|---|
| 2014 | 07 | 147910 |
| 2011 | 07 | 147060 |
| 2010 | 07 | 143860 |
| 2012 | 07 | 138166 |
| 2013 | 07 | 134297 |
| 2009 | 07 | 133570 |
| 2011 | 08 | 133483 |
| 2009 | 08 | 129007 |
| 2013 | 08 | 128920 |
| 2003 | 07 | 128360 |
| 2002 | 07 | 128354 |
| 2006 | 08 | 128039 |
| 2008 | 07 | 127627 |
| 2002 | 08 | 126495 |
| 2011 | 06 | 126084 |
| 2012 | 08 | 125947 |
| 2005 | 07 | 125651 |
| 2006 | 07 | 125192 |
| 2010 | 08 | 125135 |
| 2014 | 08 | 125006 |

2b

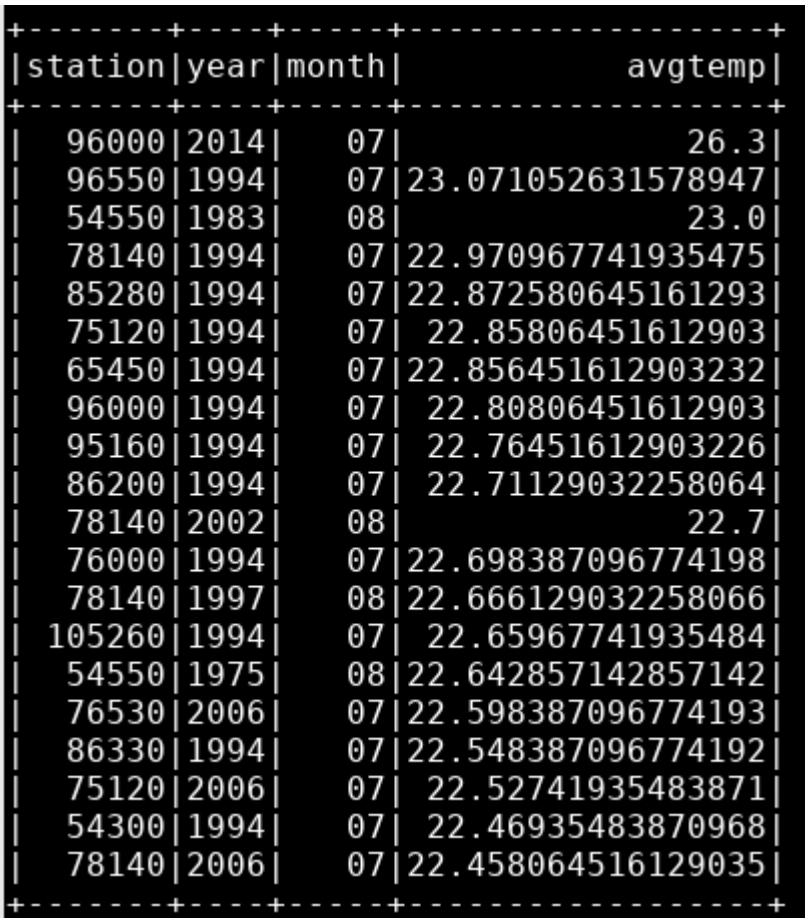
| year | month | count |
|---|---|---|
| 1972 | 10 | 378 |
| 1973 | 06 | 377 |
| 1973 | 05 | 377 |
| 1972 | 08 | 376 |
| 1973 | 09 | 376 |
| 1972 | 05 | 376 |
| 1971 | 08 | 375 |
| 1972 | 09 | 375 |
| 1972 | 06 | 375 |
| 1971 | 06 | 374 |
| 1971 | 09 | 374 |
| 1972 | 07 | 374 |
| 1973 | 08 | 373 |
| 1971 | 05 | 373 |
| 1974 | 08 | 372 |
| 1974 | 06 | 372 |
| 1974 | 09 | 370 |
| 1974 | 05 | 370 |
| 1971 | 07 | 370 |
| 1970 | 08 | 370 |

# Assignment 3

Show average monthly temperature for each available station, ordered by temperature in descending order.

Extract data. Make a map using variables that we need to complete the assignment, these are station, year, month, day, time (unnecessary we realize now), value (temperature), quality (also unnecessary). Create a dataframe, turn it into a table, filter to only have the desired time period 1960 to 2014. Then we calculate the min and max temperature for each day with agg(), F.min() and F.max() functions. When this is done we group these together by

using join(). Then we calculate an average for each day, and in the next step we add all days together and calculate the month's average. Image below shows our output.



| station | year | month | avgtemp |
|---|---|---|---|
| 96000 | 2014 | 07 | 26.3 |
| 96550 | 1994 | 07 | 23.071052631578947 |
| 54550 | 1983 | 08 | 23.0 |
| 78140 | 1994 | 07 | 22.970967741935475 |
| 85280 | 1994 | 07 | 22.872580645161293 |
| 75120 | 1994 | 07 | 22.85806451612903 |
| 65450 | 1994 | 07 | 22.856451612903232 |
| 96000 | 1994 | 07 | 22.80806451612903 |
| 95160 | 1994 | 07 | 22.76451612903226 |
| 86200 | 1994 | 07 | 22.71129032258064 |
| 78140 | 2002 | 08 | 22.7 |
| 76000 | 1994 | 07 | 22.698387096774198 |
| 78140 | 1997 | 08 | 22.666129032258066 |
| 105260 | 1994 | 07 | 22.65967741935484 |
| 54550 | 1975 | 08 | 22.642857142857142 |
| 76530 | 2006 | 07 | 22.598387096774193 |
| 86330 | 1994 | 07 | 22.548387096774192 |
| 75120 | 2006 | 07 | 22.52741935483871 |
| 54300 | 1994 | 07 | 22.46935483870968 |
| 78140 | 2006 | 07 | 22.458064516129035 |

## Assignment 4

In task 4 we created two different dataframes, one for precipitation readings and one for temperature readings.

In the temperature dataframe we first got the max temperature by station by grouping by station and using the max function on temperature. Thereafter, we filtered the maximum temperature to only get the temperature between 25 and 30.

In the precipitation dataframe we first got the daily precipitation by using the sum function on the precipitation row when grouping on station and full date (year, month, day). After getting the stations daily sum, we retrieve the maximum daily precipitation from each station. Then we filter these on the ones between 100mm and 200mm to get the correct values.

In the end we join both dataframes by taking the section of the both stations (‘inner’) and creating a “new” frame with station, maximum temperature and maximum precipitation and ordering in by the station. See results below.

| station | temp_max | max_precipitation |
|---|---|---|

## Assignment 5

Calculate the average monthly precipitation and sort it by time, i.e., year descending, month descending.

We start off by extracting the necessary data from our input files and creating two tables, one for getting all of the stations in Östergötland and one for getting the precipitation values. Using these two tables combine them by using a join function where our matching variable is station, this means that it combines the two into a new table that has all of our data that we want, i.e., data only for the stations in the Östergötland region.

Using this new table we filter for the desired time period and then sum the precipitation readings for year and month. Now we calculate the average monthly precipitation by using the built in function avg, we then orderBy year and month in descending order. Our result is shown below.

| year | month | avg(month_rain) |
|---|---|---|
| 2016 | 07 | 0.0 |
| 2016 | 06 | 47.662499999999994 |
| 2016 | 05 | 29.250000000000004 |
| 2016 | 04 | 26.900000000000006 |
| 2016 | 03 | 19.962500000000006 |
| 2016 | 02 | 21.562500000000004 |
| 2016 | 01 | 22.325 |
| 2015 | 12 | 28.925 |
| 2015 | 11 | 63.88750000000002 |
| 2015 | 10 | 2.2624999999999997 |
| 2015 | 09 | 101.29999999999998 |
| 2015 | 08 | 26.9875 |
| 2015 | 07 | 119.09999999999997 |
| 2015 | 06 | 78.66250000000002 |
| 2015 | 05 | 93.22499999999998 |
| 2015 | 04 | 15.337499999999999 |
| 2015 | 03 | 42.61250000000001 |
| 2015 | 02 | 24.825 |
| 2015 | 01 | 59.11250000000003 |
| 2014 | 12 | 35.46250000000001 |