Exercises for Seminar 2 (morning)

On appserver2, you can start `/opt/spark/bin/pyspark` and use Spark in interactive mode. Alternatively, you can make a Python program in a file and run that by using `/opt/spark/bin/spark-submit` *`filename.py`*. For the latter, you can use the file SparkTemplate.py from Moodle as a starting point.

1. In Spark, create a DataFrame df1 with a column x holding the values 0, …, 19 (hint: spark.range). Check the contents of your DataFrame with df1.show()
2. Create a DataFrame df2 with the columns y holding the square root of x and z holding x * x.
3. For df2, find the average of z for the even and odd numbers (hint: “z % 2 = 0”)
4. Try the following and explain the results (use pyspark in interactive mode)
    a. `a = col(“nosuchcolumn”)`
    b. `b = df2[“nosuchcolumn”]`
    c. `c = df2.select(a)`
    d. `d = df2.select(col(“z”))`
    e. `e = df2.select(col(“z”)).collect()`

In the directory `/home/chr/data/baseball`, there are CSV files with baseball statistics[1]. You will need the file `Batting.csv` (you can read it from my directory or copy it to your own with the command `cp` *`source target`*).

Field 0 (“playerID”) holds a player’s ID, field 1 (“yearID”) a year, and field 7 (“R”) the number of runs by the player in that year. All fields are explained in the file `readme2014.txt`.

Your task is to use Spark to:

5. Find the number of runs for each year. The results will thus be something like (1871,3101), (1872,4487), …, (2016,42276)
6. Find the number of distinct R values in an exact way and in an approximated way with up to 5% error
7. Find for each year, the maximum number of runs and the player(s) who did the runs. The results will thus be something like …, (1902,hartsto01,109), (1902,fultzda01,109), …
8. Save the result of the previous exercise to CSV files somewhere under your home directory. Look at the output (use `ls -la` from the directory and `cat` *`filename`*)
9. If the players represent their birth state, which state has then “delivered” most runs? (Hint: You need to consider data in another file as well)

Finally, consider the following

10. Compare Spark to MapReduce, Hive, and Pig. Which (dis)advantages do you see?
11. Why bother about predicate push-downs when false positives are allowed anyway? Why are false positives allowed when false negatives are not?

[1] The data comes from http://www.seanlahman.com/ and is available under the CC BY-SA 3.0 license