Lionel Fillatre – Polytech Nice Sophia

Big Data Technologies

2024/2025

Lab 3

Starting procedure

- Run the Spark container (as the root)
  ```
  docker run -u 0 -it spark:latest /usr/bin/bash
  ```
- In the container terminal, download the file "les-arbres.csv" in the current directory:

  ```
  wget -O les-arbres.csv https://github.com/lionel-fillatre/BigData/raw/main/Lab3/les-arbres.csv
  ```

  The content of the CSV file is described on the web page: https://opendata.paris.fr/explore/dataset/les-arbres/
- In the Spark shell, if you want to use the up-arrow to search backward in the command history and the down-arrow to search forward in the command history, run the following commands
  ```
  echo '"\e[A":history-search-backward' >> /etc/inputrc
  echo '"\e[B":history-search-forward' >> /etc/inputrc
  ```
- In the container terminer, start the Spark Shell with the command
  ```
  /opt/spark/bin/spark-shell
  ```
- Do the following exercises
- At the end of the lab, just use ":q" to quit "Spark" and "exit" to quit the container and stop it

1 - Perform the following tasks:

- Read the file "les-arbres.csv" with SCALA. The file can be read with the command "sc.textFile("les-arbres.csv").take(1001)" where sc is the Spark context (it will be explained in the next lecture).

- Count the lines of the result to verify that the reading works correctly.

- Filter the file to remove the headline (hint: use the commands "filter" and "!line.startsWith")

- Use a map function to cut each line of the file into fields (each field corresponds to a column of the text file).

- Use a map function to retrieve the field corresponding to "height" ("HAUTEUR" in french). To convert a string variable named S representing a float into a float number, you can use "S.toFloat".

- Filter the collection of heights in order to keep only the records such as (height > 0). The final collection of values will be called "collectTreeHeights".

- Code a map-reduce function to compute the total number of elements in "collectTreeHeights".
  The result will be stored in the variable "countTrees".

- Code a reduce function to compute the total sum of tree heights.
  The result will be stored in the variable "totalHeight".

- Code a reduce function to compute the maximum height of a tree.
  It will be stored in the variable "maxHeight". What can you conclude from this result?

- Compute and print on screen the total height, the number of trees, the maximum height of a tree and the average height of a tree.

2 - Study the following MapReduce program:

```
val nb_samples = 100000
val count = sc.parallelize(1 to nb_samples).map{i =>
  val x = Math.random()
  val y = Math.random()
  if (x*x + y*y < 1) 1 else 0
}.reduce(_ + _)
println("The result is " + 4.0 * count / nb_samples)
```

1. Describe each line of the program.

2. Draw a scheme that models the whole structure of the mapReduce program.

3. How to interpret the final numerical result? Explain carefully.