Programming Assignment 4
Histogram on Spark

Due on Thursday April 2 before midnight

Description

The purpose of this project is to develop a data analysis program using Apache Spark.

This project must be done individually. No copying is permitted. **Note: We will use a system for detecting software plagiarism, called Moss, which is an automatic system for determining the similarity of programs.** That is, your program will be compared with the programs of the other students in class as well as with the programs submitted in previous years. This program will find similarities even if you rename variables, move code, change code structure, etc.

Note that, if you use a Search Engine to find similar programs on the web, we will find these programs too. So don't do it because you will get caught and you will get an F in the course (this is cheating). Don't look for code to use for your project on the web or from other students (current or past). Just do your project alone using the help given in this project description and from your instructor and GTA only.

Platform

As in the previous projects, you will develop your program on SDSC Comet. Optionally, you may use your laptop or IntelliJ Idea or Eclipse to help you develop your program, but you should test your programs on Comet before you submit them.

Setting up your Project

Login into Comet and download and untar project4:

```
wget http://lambda.uta.edu/cse6331/project4.tgz
tar xfz project4.tgz
chmod -R g-wrx,o-wrx project4
```

Go to project4/examples and look at the Spark example JoinSpark.scala. You can compile JoinSpark.scala using:

```
run joinSparkScala.build
```

and you can run it in local mode using:

```
sbatch joinSpark.local.run
```

File join.local.out will contain the trace log of the Spark evaluation.

Project Description

You are asked to re-implement Project #1 (Histogram clustering) using Spark and Scala. An empty `project4/src/main/scala/Histogram.scala` is provided, as well as scripts to build and run this code on Comet. **You should modify Histogram.scala only**. Your main program should take one argument: the text file that contains the pixels (pixels-small.txt or pixels-large.txt). The resulting histograms must be written to the output. Note that you do not need to define a Color class. You can just use a pair of two integers, one for the color type and one for the color intensity.

You can compile Histogram.scala using:

```
run histogram.build
```

and you can run it in local mode over the small file using:

```
sbatch histogram.local.run
```

You should modify and run your programs in local mode until you get the correct result. Your output results must be the same as the `solution-small.txt` but doesn't have to be in the same order. After you make sure that your program runs correctly in local mode, you run it in distributed mode using:

```
sbatch histogram.distr.run
```

This will work on the moderate-sized file and will print the results to the output. Your output results must be the same as the `solution-large.txt` but doesn't have to be in the same order.

Optional: Use your laptop to develop your project

If you'd prefer, you may use your laptop to develop your program and then test it and run it on Comet. If you have Mac OS or Linux, make sure you have Java and Maven installed. If you have Windows 10, see project1 on how to install Ubuntu Shell. On Windows 10, you will need to set up your JAVA_PATH using: `export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64` (you can put this in your `.bashrc` file so you don't have to type it every time).

To install Spark and the project:

```
cd
wget https://archive.apache.org/dist/spark/spark-1.5.2/spark-1.5.2-bin-hadoop2.6.tgz
tar xfz spark-1.5.2-bin-hadoop2.6.tgz
wget http://lambda.uta.edu/cse6331/project4.tgz
tar xfz project4.tgz
```

To compile and run the `examples/src/main/scala/JoinSpark.scala` example:

```
cd project4/examples
rm -rf output
mvn install
~/spark-1.5.2-bin-hadoop2.6/bin/spark-submit --class JoinSpark target/cse6331-spark-examples-0.1.jar e.txt d.txt output
```

To compile and run project4:

```
cd project4
mvn install
~/spark-1.5.2-bin-hadoop2.6/bin/spark-submit --class Histogram target/cse6331-project4-0.1.jar pixels-small.txt
```

Your output must be the same as the solution-small.txt but doesn't have to be in the same order.

Optional: Use an IDE to develop your project

If you have a prior good experience with an IDE (IntelliJ IDEA or Eclipse), you may want to develop your program using an IDE and then test it and run it on Comet. Using an IDE is optional; you shouldn't do this if you haven't used an IDE before.

On IntelliJ IDEA, go to New→Project from Existing Sources, then choose your project4 directory, select Maven, and the Finish. To compile the project, go to Run→Edit Configurations, use + to Add New Configuration, select Maven, give it a name (eg, build), use Working directory: your project4 directory, Command line: install, then Apply. To run your project in local mode, you need to add the line conf.setMaster("local[2]") in the main program before you create SparkContex (you should remove this line before you test your project on Comet). Go to Run→Edit Configurations, use + to Add New Configuration, select Application, give it a name (eg, run), use the Main class: Histogram, Program arguments: pixels-small.txt.

On Eclipse, you first need to install m2e (Maven on Eclipse), if it's not already installed. Then, install Scala on Eclipse from scala-ide.org using Install New Software... and then cut-and-paste the update site URL. Then go to Open File...→Import Project from File System, then choose your project4 directory. To compile your project, right click on the project name at the Package Explorer, select Run As, and then Maven install. To run your project in local mode, you need to add the line conf.setMaster("local[2]") in the main program before you create SparkContex (you should remove this line before you test your project on Comet). Right-click on Histogram.java→Run As→Run Configurations, select Scala Application, press the New button to create a new configuration, give it a name (eg, run), add the main class Histogram, and go to Arguments. Add the argument: pixels-small.txt. Now you can run it in local mode by hitting Run.

Documentation

You can learn more about Scala at:

- A Scala Tutorial for Java Programmers
- A Tour of Scala