THIRD ASSIGNMENT. 2 POINTS

This assignment has two parts: *programming with Spark* and *machine learning with Spark*.

PROGRAMMING WITH SPARK (1 point)

The aim of this part is to program in Spark the Nearest Neighbor algorithm (KNN) by filling in the notebook supplied. In order to do this, you need to go through the "Programming k-means" notebook and understand the main ideas, because KNN uses similar concepts to k-means. In fact, it is much simpler, but it will require some thinking. The main part of the algorithm takes only two or three lines but please, explain clearly in your notebook what your code does. You have two alternatives, but you are not allowed to sort the RDD (i.e. you cannot use takeOrdered(), top(), sortByKey(), or similar):

- Alternative 1: Program KNN with K=1 (0.75 points maximum)
- Alternative 2: Program KNN with K=2 (1.0 points maximum)

MACHINE LEARNING WITH SPARK (1.0 points)

Here, we will use again the SEMEION dataset, but the reduced version that includes only digits '0' and '8'. You can find an example of using decision trees in Spark in the notebook I have provided to you "Notebook for training Decision Trees in Spark". What you have to do is, instead of a decision tree, use one or several ensembles of trees in Spark (Random Forest, Gradient Boosting). You can find documentation here:

http://spark.apache.org/docs/latest/mllib-guide.html.

This part is quite open, so you can decide what to do: try RF, RF and GB, try different values of hyper-parameters by hand (unfortunately there is no grid.search in Spark ML.lib), … I will value specially if you do something original, in addition to just applying an ensemble algorithm (you can ask me for ideas). And please, explain clearly what you do in the notebook. The grade will depend on the complexity of the task you carry out and how well you explain it in the notebook.

WHAT TO HAND IN: In both cases, you will have to hand in a notebook with comments.