Assignment 3

Tools & Techniques for Large-Scale Data Analytics (CT5105)

NUI Galway, Academic year 2016/2017, Semester 1

- ***Submission deadline (strict): Friday, 28th October, 23:59***
- *Please put your code into a single .zip archive with name "YourName_Assignment3.zip", submit via Blackboard*
- *Include all source code files (that is, files with name ending .java) required to compile and run your code..*
- *Unless specified otherwise in the question, use only Java 8 together with Apache Spark for this assignment.*
- *Please note that all submissions will be checked for plagiarism.*
- *Use comments to explain your source code. Missing or insufficient comments can lead to mark deductions. Also, don't forget to handle error conditions.*

Question 1 [max. marks: 40]

This question is similar to Question 2 of the previous assignment, but now your code should use Spark. In detail:

Add a method `countTemperature(t)` to your Java class `WeatherStation` from Q1 of Assignment 1. It should return the number of times the temperature `t` has been approximately measured so far by any of the weather stations in *`stations`* ("approximately" means `t` +/- `1`).

Again, your approach needs to follow the MapReduce scheme, but now it should do so using Spark, by making appropriate use of RDDs (instead of streams) and – again – parallel computing.

Also provide code which invokes your method with some test data and prints the results.

Remark: Use Java 8 lambda expressions instead of anonymous classes.

Question 2

a) [max. marks: 30] A typical large-scale Data Analytics task is *sentiment analysis*, i.e., the computational determination of the attitudes of people towards a certain topic or item. This can be achieved using binary classification.

Download the following data archive:
https://archive.ics.uci.edu/ml/machine-learning-databases/00331/sentiment%20labelled%20sentences.zip
Data set imdb_labelled.txt in this archive contains a number of single-sentence movie reviews, each labeled with "0" (negative sentiment) or "1" (positive sentiment).

Create a program which builds a Linear Support Vector Machine (SVM) model using any 60% of the labeled sentences in imdb_labelled.txt as training data, using Spark MLlib. Afterwards, use the learned model to predict and print the labels (sentiments) of a few test movie reviews (taken from the remaining 40% of the data file).

b) [max. marks: 30] One way to estimate the accuracy of a classifier is by computing the *Area Under the ROC Curve* (AUROC, see https://spark.apache.org/docs/latest/mllib-evaluation-metrics.html for details). Improve the accuracy of your approach to part a) as far as possible, by preprocessing the sentences in the given data set in appropriate ways.

Also let your code print the AUROC without and with use of preprocessing.