Assignment 4: Analyzing Data with Spark
ITCS 6190/8190: Cloud Computing For Data Analysis

Due: Before midnight, Sunday, April 16, 2017

Multiple Linear Regression

In this assignment, you are to implement multiple linear regression using Spark. For this, you will use the closed form expression for the ordinary least squares estimate of the linear regression coefficients computed using summation as discussed in class.

Please use the Python interface to Spark. You can read about Spark in Chapter 19 of the Hadoop book (4th ed) by White, and in the book Learning Spark by Karau, Konwinski, Wendell, and Zaharia.

While you may use the interactive Python interface to Spark to develop your program, you should submit a standalone Spark program in Python for your assignment. We will assume the commands are being run on the dsba-hadoop cluster, and that the input data file is in HDFS. (See the notes from the Spark Lab on Moodle to get familiar with Spark.) We will provide a skeleton standalone Spark program that you can modify. The command on the cluster will be:
$ spark-submit <mysparkprogram.py> <inputdatafile>

We will provide you with a few example input data files to test your code. The input files will be in CSV (comma separated values) format, with each line having the y value followed by the corresponding x values. You are to output the linear coefficients of the linear regression model (including that for the intercept).

Extra credit: You can additionally implement the gradient descent approach for linear regression. Maximum credit for this will be 10 points.

Submission:

1. Your commented source code.
2. Output files for the provided yxlin.csv, yxlin2.csv name them yxlin.out and yxlin2.out.
3. A README with any additional information about your program including how to run it, and description of the gradient descent implementation if you implement it.
4. All code must compile. Compilation errors will result in a grade of 0.
5. All files to be submitted in a single directory named as your UNCC email login ID together with your README and tar/gzip it up before submitting.

Please follow the academic integrity guidelines for your work.

Additional information:

1. If you are unfamiliar with Python, there are several online tutorials and references. For example, see https://www.python.org/about/gettingstarted/ .

2. Python has a NumPy package which provides support for representing arrays, matrices, and matrix and vector operations. You may need to install this package, for example, using pip (type “pip help install” at the command prompt). Additional information on NumPy is available at: http://www.numpy.org/

3. When you launch pyspark the system initiates (and keeps open) a job in the scheduler. The system can only do a finite number of simultaneous jobs (there is a limit enforced within the "fair scheduler" of a maximum of 6 concurrent jobs). So if many users are sitting at a pyspark prompt, active or idle, other users may be "blocked" from initiating a pyspark prompt until some users exit theirs. Since we are limited to 6 concurrent jobs on the cluster and we do not want to allow a single user to monopolize the resources, each user is allowed to have a single job active. This prevents a user from single-handedly taking up the entire cluster. This also prevents a user from running a long job in one window and initiating a pyspark shell simultaneously in another. For this reason, running a standalone Spark Python program will be more considerate to other users of the cluster.

   On the cluster, you can get a list of all applications running and their ApplicationIds by typing:
   $ yarn application –list

   To kill your job, use:
   $ yarn application -kill <ApplicationId>

   Do NOT kill other students jobs without first consulting them.