DS 222: ML with Large Datasets (2018)
Assignment 2

Weightage: 20%
Due: 11:59 PM (IST) Oct 27, 2018

In this assignment, you will implement and evaluate a L2-regularized logistic regression in the standalone and distributed setting using Parameter Server. Please use your own implementation of SGD for learning.

- Part 1: [5 points]
    - Local Logistic Regression (LR): Implement yourself a L2-regularized logistic regression and evaluate it in a local setting. Use SGD for learning. Report train & test accuracies, training time and test time. Also, plot training loss against epochs.
    - Perform the same analysis as above but with the following three learning rate strategies: constant, increasing, and decreasing. Please choose appropriate increasing and decreasing functions.
- Part 2: [15 points]
    - In this part of the assignment, you will parallelize the classifier developed above. Based on your analysis above, please freeze the learning rate and hyperparameters for the experiments below.
    - First, please select a parameter server and framework (e.g., Hadoop, Spark, etc) for this part and justify your choice.
    - Train the classifier in the BSP SGD setting and report train & test accuracies, training time and test time. Also, plot training loss against epochs.
    - Train the classifier in the Asynchronous SGD setting and report train & test accuracies, training time and test time. Also, plot training loss against epochs.
    - Plot and compare the training losses against epochs in the Asynchronous SGD setting for a few choices of the number of workers.
    - Train the classifier in Bounded Asynchronous (SSP) SGD and report train & test accuracies, training time and test time. Also, plot training loss against epochs.
    - Plot and compare the training losses against epochs in the SSP SGD setting for a few choices of the delay parameter.
    - You may merge all the above plots (best plots from Asynchronous and SSP experiments) into one for easier comparison.

Data

We are using the same dataset as in Assignment 1. This dataset was extracted from DBpedia. The labels of the article are mapping-based types of the document. There are in total 50 classes in the dataset, and they are from article labels in the dataset. For more information about this dataset, you can refer to http://wiki.dbpedia.org/Downloads2015-04.

There are multiple class labels per document. This means that there is more than one correct answer for each document. The data for this assignment is available at:

- HDFS: /user/ds222/assignment-1/DBPedia.full/
- Turing local: /scratch/ds222-2017/assignment-1/DBPedia.full

The format is one document per line with following columns separated by a tab character :

- List of labels separated by comma
- Document

The first two tokens in the document field are the names of the entity and a type field. You may ignore these. Each data set appears in full in one file, and is split into a train and test set, as indicated by the file suffix.

It might be best to debug your implementation with a small dataset before running it on the full dataset mentioned above. For this reason, a small dataset with same structure as above is available here:

- /user/ds222/assignment-1/DBPedia.verysmall (HDFS)
- /scratch/ds222-2017/assignment-1/DBPedia.verysmall/ (local)

You can refer to the document here to get familiar with the Turing Cluster to be used for the assignment.

Submission

- This is an individual assignment. Each student should submit individual work
- If you have consulted with anyone or sought help from any source, then please acknowledge that clearly in the report.
- Please use Latex to write your report. You may use the format available here
- All submissions will be done through Github. A submission form will be made available closer to the deadline.
- In all experiments, discuss your observations, and possible causes for the same. Just reporting numbers is not sufficient.

Please post on course mailing list for any issues you face or clarifications you need regarding Assignment 2.