EE412 Foundation of Big Data Analytics, Fall 2018
HW3

Due date: 11/21/2018 (11:59pm)

Submission instructions: Use KAIST KLMS to submit your homeworks. Your submission should be one gzipped tar file whose name is `YourStudentID_hw3.tar.gz`. For example, if your student ID is 20161234, and it is for homework #3, please name the file as `20161234_hw3.tar.gz`. You can also use these extensions: tar, gz, zip, tar.zip. Do not use other options not mentioned here.

Your zip file should contain total four files; one PDF file for writeup answers (`hw3.pdf`), two python files (`hw3_1.py` and `hw3_2.py`), and the Ethics Oath pdf file.
Before zipping your files, please make a directory named `YourStudentID_hw3` and put your files in the directory. Then, please compress the directory to make a zipped file.
Do not include Korean letters in any file name or directory name when you submit.

Submitting writeup: Prepare answers to the homework questions into a single PDF file. You can use the following template. Please write as succinctly as possible.

Submitting code: Each problem is accompanied by a programming part. Put all the code for each question into a single file. Good coding style (including comments) will be one criterion for grading. Please make sure your code is well structured and has descriptive comments.

Ethics Oath: For every homework submission, please fill out and submit the **PDF** version of this document that pledges your honor that you did not violate any ethics rules required by this course and KAIST. You can either scan a printed version into a PDF file or make the Word document into a PDF file after filling it out. Please sign on the document and submit it along with your other files.

Discussions with other people are permitted and encouraged. However, when the time comes to write your solution, such discussions (except with course staff members) are no longer appropriate: you must write down your own solutions independently. If you received any help, you must specify on the top of your written homework any individuals from whom you received help, and the nature of the help that you received. *Do not, under any circumstances, copy another person's solution.* We check all submissions for plagiarism and take any violations seriously.

1 Link Analysis (45 points)

(a) [15 pts] Solve the following problems, which are based on the exercises in the Mining of Massive Datasets 2nd edition (MMDS) textbook.

- Exercise 5.1.2
  Compute the PageRank of each page in Fig 5.7, assuming $\beta = 0.8$.
  You can use programs for simple calculations. If you use any programming, please attach your code and explain about it.

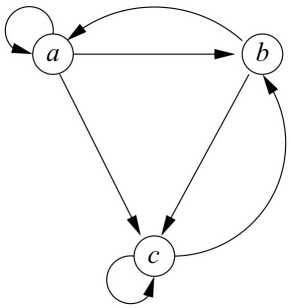

Figure 5.7: An example graph for exercises

- Exercise 5.3.1
  Compute the topic-sensitive PageRank for the graph of Fig 5.15, assuming β = 0.8 and the teleport set is:
  (a) A only
  (b) A and C

  You can use programs for simple calculations. If you use any programming, please attach your code and explain about it.

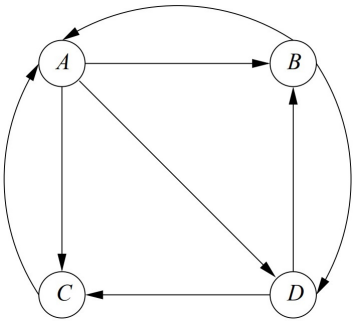

Figure 5.15: Repeat of example Web graph

- Exercise 5.4.3
  Suppose two spam farmers agree to link their spam farms. How would you link the pages in order to increase as much as possible the PageRank of each spam farm's target page? Is there an advantage to linking spam farms?

(b) [15 pts] Implement the PageRank algorithm using Spark.

Implement the PageRank algorithm described in MMDS Chapter 5.2.2 using Spark.

The dataset can be downloaded from this link:
http://www.di.kaist.ac.kr/~swhang/ee412/graph.txt[1]

The graph is randomly generated and has 1000 nodes and about 8000 edges with no dead ends. For each row, the left page id is the source and the right page id the destination. If there are duplicate edges from one page to another, treat them as the same.

You may set $\beta = 0.9$ and start from the vector $\mathbf{v}$ initialized as all 1's divided by the number of pages. You do not have to break the transition matrix M into stripes. Run your algorithm for 50 iterations to produce the final vector $\mathbf{v}$ and return the ids of the top-10 pages with the highest PageRank scores.

Please **use command-line** arguments to obtain the file path of the dataset. (Do not fix the path in your code.) For example, run:

```
bin/spark-submit hw3_1.py path/to/graph.txt
```

After run, your code (`hw3_1.py`) should `print` the top-10 page ids with the highest PageRank scores in $\mathbf{v}$. The output format is the following:

```
<PAGE_ID_0><TAB><SCORE_0>
<PAGE_ID_1><TAB><SCORE_1>
...
<PAGE_ID_9><TAB><SCORE_9>
```

The output should be 10 lines and sorted in descending order.

(c) [15 pts] Implement the HITs algorithm using Spark

Implement the HITs algorithm described in MMDS Chapter 5.5.2 starting from $\mathbf{h}$ initialized as all 1's. Use the same dataset as in (b) and perform 50 iterations like in Figure 5.20. From the final hubbiness and authority vectors $\mathbf{h}$ and $\mathbf{a}$, return the top-10 page ids with the highest scores.

Please **use command-line** arguments to obtain the file path of the dataset. (Do not fix the path in your code.) For example, run:

```
bin/spark-submit hw3_2.py path/to/graph.txt
```

After run, your code (`hw3_2.py`) should `print` the top-10 page ids with the highest hubbiness scores and the top-10 page ids with the highest authority scores. The output format is the following:

```
<PAGE_ID_H0><TAB><HUBBINESS_SCORE_0>
```

[1]This dataset is from Stanford University.

```
<PAGE_ID_H1><TAB><HUBBINESS_SCORE_1>
...
<PAGE_ID_H9><TAB><HUBBINESS_SCORE_9>
<PAGE_ID_A0><TAB><AUTHORITY_SCORE_0>
<PAGE_ID_A1><TAB><AUTHORITY_SCORE_1>
...
<PAGE_ID_A9><TAB><AUTHORITY_SCORE_9>
```

The output should be 20 lines and the first 10 lines (and also last 10 lines) should be sorted in descending order.

2 Mining Social-Network Graphs (30 points)

Solve the following problems, which are based on the exercises in the MMDS textbook.

- Exercise 10.1.1 (Modified)
  It is possible to think of the edges of one graph G as the nodes of another graph G'. We construct G' from G by the *dual construction*:

  1. If (X, Y) is an edge of G, then XY, representing the unordered set of X and Y is a node of G'. Note that XY and YX represent the same node of G', not two different nodes.
  2. If (X, Y) and (X, Z) are edges of G, then in G' there is an edge between XY and XZ. That is, nodes of G' have an edge between them if the edges of G that these nodes represent have a node (of G) in common.

  (a) If we apply the dual construction to a network of friends, what is the interpretation of the edges of the resulting graph G'?

  (b) Apply the dual construction to the graph of Fig. 10.1.

  (c) How is the degree of a node XY in G' related to the degrees of X and Y in G?

  (d) What we called the dual is not a true dual, because applying the construction to G' does not necessarily yield a graph isomorphic to G. Give an example graph G where the dual of G' is isomorphic to G and another example where the dual of G' is not isomorphic to G.

  Note: Two graphs G and H are isomorphic if there exists an isomorphism(or equivalently, one-to-one and onto function) between the vertex sets of G and H.

  $$f : V(G) \to V(H) \tag{1}$$

  such that any two vertices u and v of G are adjacent in G if and only if $f(u)$ and $f(v)$ are adjacent in H.)

- Exercise 10.3.2
  Suppose there is a community of $2n$ nodes. Divide the community into two groups of n members, at random, and form the bipartite graph between the two groups. Suppose that the average degree of the nodes of the bipartite graph is d. Find the set of maximal pairs (t, s), with $t \leq s$, such that an instance of $K_{s,t}$ is guaranteed to exist, for the following combinations of n and d:

(a) $n = 20$ and $d = 5$.

(b) $n = 200$ and $d = 150$.

By "maximal," we mean there is no different pair (s', t') such that both $s' \geq s$ and $t' \geq t$ hold.

- Exercise 10.5.2
  Compute the MLE for the graph in Example 10.22 for the following guesses of the memberships of the two communities.

  (a) $C = \{w, x\}$; $C = \{y, z\}$

  (b) $C = \{w, x, y, z\}$; $C = \{x, y, z\}$

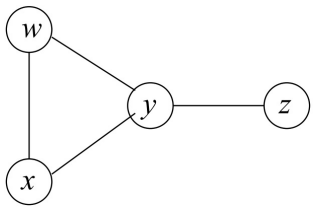

Figure 10.20: A social graph

3 Large-Scale Machine Learning (25 points)

Solve the following problems, which are based on the exercises in the MMDS textbook and the MMDS third edition beta version.

- Exercise 12.3.2
  The following training set obeys the rule that the positive examples all have vectors whose components sum to 10 or more, while the sum is less than 10 for the negative examples.

([3, 4, 5], +1)	([2, 7, 2], +1)	([5, 5, 5], +1)
([1, 2, 3], -1)	([3, 3, 2], -1)	([2, 4, 1], -1)

  (a) Which of these six vectors are the support vectors?

  (b) Suggest a vector $\mathbf{w}$ and constant b such that the hyperplane defined by $\mathbf{w.x} + b = 0$ is a good separator for the positive and negative examples. Make sure that the scale of $\mathbf{w}$ is such that all points are outside the margin; that is, for each training example (x, y), you have $y(\mathbf{w.x} + b) \geq +1$.

  (c) Starting with your answer to part (b), use gradient descent to find the optimum $\mathbf{w}$ and b. Note that if you start with a separating hyperplane, and you scale $\mathbf{w}$ properly, then the second term of Equation 12.4 will always be 0, which simplifies your work considerably. Also, if you use any programming, please attach your code and explain about it.

- Exercise 12.4.1
  Suppose we modified Example 12.11 to look at the two nearest neighbors of a query point q. Classify q with the common label if those two neighbors have the same label, and leave q unclassified if the labels of the neighbors are different.

  (a) Sketch the boundaries of the regions for the three dog breeds on Fig. 12.21.

  (b) Would the boundaries always consist of straight line segments for any training data?

- Exercise 12.5.3 (see MMDS 3rd edition beta version:
  http://i.stanford.edu/~ullman/mmds/ch12n.pdf)
  An important property of a function f is *convexity*, meaning that if $x < z < y$, then

$$f(z) > \frac{z-x}{y-x} f(x) + \frac{y-z}{y-x} f(y) \tag{2}$$

  Less formally, the curve of f between x and y lies above the straight line between the points $(x, f(x))$ and $(y, f(y))$. In the following, assume there are two classes, and f(x) is the impurity when x is the fraction of examples in the first class.

  (a) Prove that the GINI impurity is convex.

  (b) Prove that the Entropy measure of impurity is convex.

  (c) Give an example to show that the Accuracy measure of impurity is not always convex. Hint: Note that convexity requires strict inequality; a straight line is not convex.