Carnegie Mellon University
15-826 – Multimedia Databases and Data Mining
Fall 2013, C. Faloutsos
Homework 3, Due Date: Nov. 19, at classroom 1:30pm
Prepared by: Alex Beutel (Q1), Seunghak Lee (Q2, Q3), Vagelis Papalexakis (Q4, Q5, Q6)

Reminders

- All homeworks are to be done **INDIVIDUALLY**.
- All written answers should be **TYPED**.
- For code submission to blackboard, please follow the template of the `hw3.zip` file that you can find download from `http://www.cs.cmu.edu/~christos/courses/826.F13/HOMEWORKS/HW3/hw3.zip`. You should rename the final .zip file to [andrew-id].zip.
- **The above `hw3.zip` also contains all the data for every question.**
- FYI: Total number of points is 100.
- FYI: Expected effort for this homework (approximate times): 45 hours.
  - Q1: ≈ 10 hours (**Grader: Alex Beutel**)
    - 1 hour to set up Hadoop
    - 2 hours learning HDFS commands
    - 5 hours modifying WordCount to NGrams
    - 1 hour running the code to get the answers
    - 1 hour making the entire thing into a bash script that runs smoothly
  - Q2: ≈ 4 hours (**Grader: Seunghak Lee**)
  - Q3: 10 hours (**Grader: Seunghak Lee**)
  - Q4: ≈ 5 hours (**Grader: Vagelis Papalexakis**)
    - 1-2 hours to download and set up the tensor toolbox.
    - 2 hours to run the required commands.
    - 1 hour to finalize the question
  - Q5: ≈ 5 hours (**Grader: Vagelis Papalexakis**)
  - Q6: ≈ 10 hours (**Grader: Vagelis Papalexakis**)

Q1 – Hadoop [30 pts]

(On separate page)
Problem Description: In this question, we will use Hadoop to find the top n-grams used in Reddit titles[1]. We will be using Hadoop 1.2.1. Note, this question has been designed

[1] Prepared from SNAP's Reddit dataset `http://snap.stanford.edu/data/redditSubmissions.csv.gz`

for and tested on the GHC Andrew machines (ghc#.ghc.andrew.cmu.edu where # ranges from 01-79), and we highly suggest you work on this question on one of those machines[2]. **Each student has a specific machine assigned to them in the grade center on Blackboard that they should use (under the column "andrew-machine-id").**

To set up your machine download the script `http://cs.cmu.edu/~abeutel/build-hadoop-1.2.sh` and place it in the directory where you want to run Hadoop and solve this question. Then run

```
bash build-hadoop-1.2.sh; source setenv.csh
```

You will likely need to enter your Andrew password a few times to while the script is running. This script will install Hadoop and HDFS locally in the directory you placed the original script [3]. When the script is done running, Hadoop will be running on the machine, `start-hadoop.sh` and `stop-hadoop.sh` will be in your directory that will respectively start and stop Hadoop, and the necessary resources for this question will be in place, which we will go over later.
Note, if another student is currently running Hadoop on the machine you are on you will not be able to also run Hadoop on that machine. If this happens, you do not need to reinstall Hadoop on a different GHC machine (assuming you've already installed it), but rather just log into another GHC machine and try starting your Hadoop instance there. (Please try the machines from 70-79 since these will not be assigned to any students.) The script will warn you if there is an issue.

However, please shut down your Hadoop instance when you log out so that other students can use the machine!!!!!

(If it appears someone else is using the machine, you can look for long running java instances in `top` and email the user to turn off their Hadoop.) When using Hadoop on a new machine, you may get an error when running your commands that Hadoop is in "safe mode." If you get this error, go to the hadoop-1.2.1 directory and run the following command `bin/hadoop dfsadmin -safemode leave`

Ok, let's get started:

1. [4 pts] First, upload the data file `reddit_titles.csv` to HDFS. The data file [4] can be found in your directory `hadoop-1.2.1/reddit_titles/`. Upload the entire reddit_titles directory to HDFS and then verify it uploaded correctly.

[2] If you decide not to run on an Andrew machine, you will need to reset the `$JAVA_HOME` term in the script. See the comments in the script.

[3] Note that the script assumes that your default shell on the Andrew machines is csh, which it appears to be by default; if you have changed this you (to bash for example) you will need to add the correct `$JAVA_HOME` and `$HADOOP_PREFIX` to your `.bashrc` file.

[4] The data file can also be downloaded independently from `http://cs.cmu.edu/~abeutel/reddit_titles.csv`.

To do this, you will want to go into the `hadoop-1.2.1/` directory and use commands of the form `bin/hadoop fs`. Entering that command will give you a list of possible commands to run such as `-ls` and `-mkdir`, which of course correspond to their Unix equivalent `ls` and `mkdir`. (More information on the commands can be found at `http://hadoop.apache.org/docs/r0.18.3/hdfs_shell.html`) Your home path is just `/` When this step is complete your HDFS should have a directory `/reddit_titles/` with a file `/reddit_titles/reddit_titles.csv`.

2. [2 pts] We will now test that Hadoop is working with a classic example, WordCount. If you navigate in your directory to `hadoop-1.2.1/ngram/` you will find a script `compile.sh` and a Java program `WordCount.java`[5]. Running `bash compile.sh` will compile the WordCount example and produce a jar `WordCount.jar`.

3. [4 pts] Run the WordCount example on the reddit_titles data and report the number of unique words in the data file. (You will likely find the Unix command `wc -l` helpful here.) General instructions on how to run the code can be found at `http://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html#Usage`, although in our example there are some slight deviations.

4. [10 pts] Copy the WordCount example into a new java file `NGram.java`. Modify the WordCount example there to count the number of instances of each n-gram in the dataset. Here, an n-gram is defined as strings of n contiguous words separated by a space in each title. If a title has less than n words ignore it, and at the end of a title if a word is not followed by $n - 1$ words then it is not an n-gram. For example, the title "A B C" has two 2-grams: "A B" and "B C."

   Only output n-grams that occur at least m times in the dataset. (You can either hardcode n and m or set them as parameters to be set at runtime.)

5. [2.5 pts] How many bigrams ($n = 2$) occur at least $m = 100$ times in the data set?

6. [2.5 pts] Sort the bigrams from most common to least common using your favorite scripting language (check out Unix's `sort` command). List the top 20 most common bigrams and the number of times they occur.

7. [2.5 pts] How many trigrams ($n = 3$) occur at least $m = 20$ times in the data set?

8. [2.5 pts] List the top 20 most common trigrams and the number of times they occur.

What to turn in:

- **On Paper:** Turn in a print out of all of the code necessary to complete the tasks above as well as their results. This should include the necessary Unix commands, etc. *(On separate page)*

[5] The two can also be downloaded from `http://cs.cmu.edu/~abeutel/compile.sh` and `http://cs.cmu.edu/~abeutel/WordCount.java`. This is modified from the Apache Tutorial `http://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html`

- **Online:** Turn in all code that was necessary to complete this question. This should include a bash script that when run from your hadoop-1.2.1 directory will complete all of the questions. Also include a text file with the answers to the questions.

Q2 – Singular Value Decomposition [10 pts]

(On separate page)
Problem Description: We will investigate Singular Value Decomposition of a cloud of points.
Consider the 5-dimensional dataset on a linear manifold which is available at `http://www.cs.cmu.edu/~seunghak/15826/hw3/svd.dat`. Each point in the point cloud lies roughly on a 2-dimensional plane spanned by two vectors $\mathbf{v_1}$ and $\mathbf{v_2}$ with additional noise term $\epsilon_i \sim \mathcal{N}(0, \Sigma)$ which follows a Gaussian distribution. More specifically, $\Sigma = 6I_{5,5}$, where $I_{5,5}$ is an 5×5 identify matrix.

1. [4 pts] Use SVD to visualize the dataset in 2-dimensional space by projecting in the best 2 dimensions.
2. [4 pts] There are two outliers in the dataset. Find those two outliers by manually looking at your visualization in 2-dimensional space. Then print your Plot and mark the outliers with your pen, and submit this plot. Also, report the 2-d coordinates of the outliers in the 2-d ('concept') space.
3. [2 pts] We want to reverse engineer the two outliers from the concept space to original space. What is the formula we need to find 5-d coordinates of a point in the original space, given the 2-d coordinates in the projected ('concept') space? For example, the point (0,0) in 'concept space' corresponds to (0,0,0,0,0) in the original space.

What to turn in:

- **On Paper:** Turn in the plots for question 1 and 2, and the answers to question 2 and 3.
- **Online:** Turn in all the code that you wrote, as well as the plots in .pdf form.

Q3– Independent Component Analysis [10 pts]

(On separate page)
Problem Description:
The goal of this exercise is to make us familiar with Independent Component Analysis (ICA) which is a powerful technique to separate mixed signals. ICA is also called 'Blind source separation' (BSS), because it can often solve the 'coctail party problem': several people are speaking simultaneously, but we are able to isolate and lock-on to the discussion of interest.

Now consider this dataset, available at
`http://www.cs.cmu.edu/~seunghak/15826/hw3/ica.dat`, which contains point clould generated by three lines in 4-dimensional space with random noise following a Laplace distribution. Here you will find the membership of each data point (i.e, the line which was used to generate a data point) using ICA.

1. [2 pts] Using PCA, map the provided data points onto 3-dimensional space. We want to show that PCA does not create clear groups, that is, each point belongs to every concept. To verity your results, draw plots using every pair of coordinates obtained by PCA (there are 3 coordinates, so in total, there are three plots with each coordinate pair (i.e., (1,2),(2,3),(1,3)))

2. [2 pts] Using ICA, map the provided data points onto 3-dimensional space. Now we want to show that ICA creates better grouping. To verify your results, draw plots using every pair of coordinates obtained by ICA. (there are 3 coordinates, so in total, there are three plots with each coordinate pair (i.e., (1,2),(2,3),(1,3))) (Hint: Use ICA implementation in `http://research.ics.aalto.fi/ica/fastica/`)

3. [2 pts] For PCA results, for each data point whose length is > 1 in its original point in 4-d space (i.e., $\sqrt{x_1^2 + x_2^2 + x_3^2 + x_4^2} > 1$ for a data point at (x_1, x_2, x_3, x_4)), as a measure, report "purity score," defined as $1 - \frac{\text{second largest absolute value}}{\text{largest absolute value}}$.
   (E.g. Suppose $x_j = [-0.1, 0.7, 0.4]$ (the j-th data point mapped onto 3-d space). Purity score for x_j is $1 - \frac{0.4}{0.7} = 0.4286$).
   (Report format: in each line print 'data point index [TAB] purity score'; and sort by data point index) (Note: 'data point index' is defined as the line number of that point, in the 'ica.dat' input file, starting from '1')

4. [2 pts] For ICA results, for each data point whose length is > 1 in its original point in 4-d space, as a measure, report "purity score." (Report format: in each line print 'data point index [TAB] purity score'; and sort by data point index)

5. [2 pts] Compare the two results obtained by PCA and ICA. Which one is better with respect to total purity (sum over reported purity scores above) and why?

What to turn in:

- **On Paper:** Turn in the plots for question 1 and 2, and the answers to question 3, 4, 5.

- **Online:** Turn in all the code that you wrote, as well as the plots in .pdf form.

Q4– Finding Fraudsters using Tensor Decompositions [15 pts]

(On separate page)
Problem Description: In this question, we your familiarize yourselves with the CP/PARAFAC tensor decomposition, and how to use to spot interesting structures in data. Suppose that you have a social network, like Facebook, where users like pages, on specific dates. You are given a user×page×date tensor, which contains such data.
Fraudster behaviour usually tends to create bipartite cores, when, e.g. almost all users in a set (the fraudster set) like almost all of the pages in a specific set. The dataset we give you contains such a set of an unknown number of fraudsters and pages, which form a nearly complete bipartite core, which was active on *4 different* dates. [HINT: Those dates need not be consecutive] You will use the CP/PARAFAC decomposition of this tensor in order to discover this hidden bipartite core. The dataset is in *tab separated* format and each row contains the non-zero values of the tensor, i.e. each row is formatted as

```
i j k value
```

where the indices start from 1.

1. [0 pts] Download and install the Tensor Toolbox for Matlab, from `http://www.sandia.gov/~tgkolda/TensorToolbox/index-2.5.html`
2. [2 pts] Read the tab separated file that we give you into a sparse tensor in Tensor Toolbox. [HINT: type `help dlmread` and `help sptensor`].
3. [3 pts]Run the CP/PARFAC decomposition on the tensor you created, for rank $R = 2$. Provide plots for each of the components. [HINT: Plot the factor vectors for each component]. Can you tell which component contains the bipartite core? [HUNT: You can look at the documentation of Tensor Toolbox by typing `help tensor_toolbox`. The function you will need is `cp_als`.]
4. [10 pts] Ignoring near-zero values of the rank-one component(s) that you have extracted, give:
   (a) the count of users participating in the bipartite core
   (b) the user IDs of those users
   (c) the count of pages participating in the bipartite core
   (d) the IDs of those pages
   (e) the dates for which the bipartite core can be observed

What to turn in:

- **On Paper:** Turn in a printout of all the code that you wrote in order to solve this question. Also, turn in the plots for question 3, and the answers to question 4.
- **Online:** Turn in all the code that you wrote, as well as the plots in .pdf form.

Q5– Fourier Transformation [15 pts]

(On separate page)
Problem Description: In this question, you will use the Fourier Transformation as a tool for discovering periodicities in time series.

1. [1 pt] You are given a time series dataset that records the activity of the sun spots for every month, ranging between 1749 and 2013, in file `spot_num.txt`.[HINT: You are interested in columns 1 and 3 of this file]. Plot the time series.

2. [1 pts] Use the Fourier transformation, as we saw in class, in order to plot the spectrum of the above time-series (on the frequency domain). [HINT: The spectrum can be complex, so use the absolute value, as we saw in class]

3. [5 pts] Using the spectrum, what is the periodicity of the signal in years? [HINT: Non-integer answer is fine]

4. [1.5 pt] You are now given a new signal, in file `y.txt`, which is randomly generated.

   (a) Plot the signal.

   (b) Plot its spectrum.

   (c) Are there any frequencies (excluding the DC frequency) standing out? Enumerate those frequencies, if any. [HINT: By standing out, we expect the amplitude of the spectrum for a given frequency to be at least 5 times larger than the average.]

5. [1.5 pts] You are now given signal `y_mix.txt`, which is generated the same way as the previous one, but is also mixed (by addition) with a secret sinusoid.

   (a) Plot the given signal `y_mix.txt`.

   (b) Plot its spectrum.

   (c) Are there any frequencies (excluding the DC frequency) standing out? Enumerate those frequencies, if any.

6. [5 pts] Using *only* the information provided to you by the spectrum, try to recover the secret sinusoid that we have injected

   (a) Plot the injected signal in the time domain (it's fine if you don't recover the exact amplitude - assume that the phase is zero.)

What to turn in:

- **On Paper:** Turn in a printout of all the code that you wrote in order to solve this question. Also, turn in printouts of all the plots, and typed answers to the itemized sub-questions of every question.

- **Online:** Turn in all the code that you wrote, as well as all the plots required, in .pdf format.

Q6– Wavelet Transformation [20 pts]

(On separate page)
Problem Description:
In this problem, we will use wavelets for two purposes:

- Detecting hidden signals, that Fourier transform is unable to detect.
- *Sketching*: In very large databases, usually the user might require a quick and approximate answer, as opposed to an exact answer that might be very slow. One way to do that, for aggregate queries, such as the sum of the values of a certain table, is to use *sketching*, i.e. storing a small, compressed version of the table, which will enable us to answer approximate queries very fast. The Haar Wavelet transformation can be used in this scenario.

A: Hidden signal Detection [10 pts]

1. [2 pts]You are given a signal in file `y.txt`. Each row corresponds to, say, the value of a stock, at one time-tick (indices start from 1). Within this signal, we have added a sinusoid for a specific time window, starting at t_1 and finishing at t_2.
    (a) [0 pts] Plot the signal in the time domain.
    (b) [1 pt] Plot its spectrum, using the Fourier transform.
    (c) [1 pt] Are you able to detect any major periodicities (excluding the DC frequency)? List all such frequencies, if any. [HINT: same rules for the frequencies that stand out applies here, as in Q5.]
2. [8 pts] Now, let's try to examine this signal using the Haar Wavelet transformation.
    (a) [1 pts] Do the discrete Haar wavelet transformation for this signal, for 12 levels.
    (b) [1 pt] Give a plot of the Haar wavelet coefficients that you found in the previous question. This is also called a *time-frequency plot* or *wavelet scaleogram*.
    (c) [6 pts] Given the time-frequency plot / scaleogram,
        i. find a rough estimate for t_1, t_2
        ii. find a rough estimate for the frequency of the sinusoid.

    [HINT: For this purpose, you can use the Matlab function `wavelet_scaleogram.m` that we give you. If you want to implement it yourself, check the documentation of the `wavedec` function, as well as type `wavemenu`, for an interactive Wavelet toolbox. Alternatively, you may also use the code found at `http://www.cs.cmu.edu/~christos/SRC/DWT-Haar-all.tar`, that implements the Haar wavelet transformation, in Perl.]

B: Sketching [10 pts]

1. [1 pt] Consider again the file `y.txt`. Calculate the sum of its values for $t = [512, 1024]$. What is the value?

2. [4 pts] Do the Haar wavelet transformation on the above data, for the first $L = 7$ levels. [HINT: You may use the Matlab functions `wavedec` and `waverec`]. Find the N = 100 highest wavelet coefficients (in absolute value) and zero out all the rest. Then reconstruct the original data using only those coefficients [HINT: See `waverec` in Matlab]. After you reconstruct the signal:

   (a) Compute and print the approximate sum for $t = [512, 1024]$.

   (b) How much is the error, with respect to the exact sum?

3. [5 pts] Given those $N = 100$ of the wavelet coefficients that you have kept, do you need them all, in order to obtain the best reconstruction of the sum between 512 and 1024?

   (a) Describe how you can find the subset of the necessary coefficients. Write code that finds it.

   (b) After you find this subset, re-calculate the sum and verify that it is the same as before (i.e. in question 3).

What to turn in:

- **On Paper:** Turn in a printout of all the code that you wrote in order to solve this question. Also turn in your typed answers to all the questions, as well as printouts of all the figures required.

- **Online:** Turn in all the code that you wrote. Also, turn in the figures that you generated, in.pdf form.