Assignment 6

Due: November 13, in class
No late assignments accepted

Issued: November 6, 2013

Important:

- Give complete answers: Do not only give mathematical formulae, but explain what you are doing. Conversely, do not leave out critical intermediate steps in mathematical derivations.
- Write your **name** as well as your **Sunet ID** on your assignment. **Please staple pages together.** Points will be docked otherwise.
- Questions preceded by $\star$ are harder and/or more involved.

Problem 1

PageRank algorithms revisited (65 pts) Note: For this problem, please refer back to Assignment 5. The solutions will help you in coding your PageRank algorithm.
Recall the linear system for Page Rank:

$$(I - \alpha P)\vec{x} = \vec{v}$$

where α is the fraction of a pages rank that it propagates to neighbors at each step and $\vec{v}$ the initial rank we give to each page. In our problem, we set $\alpha = 0.85$ and the entries of $\vec{v}$ to be $\vec{v}_i = \frac{1}{n}$, with n the total number of pages in our network.
The PageRank calculation need not only be used for the internet! In fact, PageRank can be calculated for nodes on any connected graph representing any abstraction. For this problem, we will consider a graph of movies connected by links if they share at least one common actor. For this purpose, we provide a matlab date file `movies.mat` that can be downloaded from the Materials $->$ Assignments2013 folder on Coursework. Place this file in a local directory accessible in matlab and type `clear all; load movies.mat`. If you look in the workspace, you should have a collection of new variables, defined as follows:

- `links` : rows are (movie, person) pairs (e.g., for `links(1,:)` equal to [779,20278] means that actor personName20278 was in movie movieName779) (James Jimmy Stewart in the movie Rope)
- `movieIMDbID` : the IMDb IDs of each movie
- `movieName` : the name of each movie
- `movieRating` : the average IMDb rating of people who have rated this movie online
- `movieVotes` : the number of people who have rated this movie on IMDb
- `movieYear` : the year in which this movie was released
- `personIMDBID` : the IMDB IDs of each actor/actress
- `personName` : the name of each actor/actress

None of these are the proper PageRank matrix P.

(a) (20 points) Let C be the $m \times n$ matrix defined by $C_{ij} = 1$ if movie i contains actor/actress j. Let B be the $m \times m$ matrix where B_{ij} is the number of actors starring in both movie i and movie j.

(i) Show how to calculate B from C.

(ii) Show how to calculate the PageRank matrix P from B: (Hint: it may help to construct a small graph of movies and actors (need not be based on real data) and to actually construct these individual matrices). Remember that movie i and movie j sharing at least one actor counts as one link from movie i to movie j.

(b) (10 points) Using matlab, construct the PageRank matrix P. DO NOT PRINT THIS MATRIX OUT. Instead, submit the sparsity plot of P using the command `spy(P)`. Use sparse matrix commands to assist you; otherwise, matlab may be temperamental.

(c) (10 points) Compute the PageRank vector $\vec{x}$ of the movies in this graph and normalize this quantity so $||\vec{x}||_1 = 1$. List the PageRank values and titles of the movies with the five highest PageRank values.

(d) (15 points) Compute the PageRank vector $\vec{x}$ of the movies via a Jacobi iteration that you write yourself and normalize this quantity so $||\vec{x}||_1 = 1$ after each iteration. Decide on an intelligent measure for convergence (assume you do not know the actual PageRank vector $\vec{x}$ because your system is too large for a simple backslash calculation.) Explain your choice of convergence criterion. Next, plot this convergence measure over the steps in your iteration. How many iterations does it take your implementation to get to a tolerance of 10^{-4}.

(e) (10 points) Plot IMDb movie rating vs. PageRank and IMDb movie votes vs. PageRank. Is PageRank a good predictor of movie rating or movie votes?

Problem 2

Least squares (35 points) We measured the temperature T below the ground on an unusually cold day a few weeks ago. The temperature was measured as a function of the distance from the ground surface. The outside temperature was a balmy 2 deg. Celsius. The data from the measurements are given in the table below:

Distance (m)	Temperature (C)
0	2.0
5	2.2
10	5.8
15	10.4
20	11.0
25	13.8
30	22.4
35	28.4
40	33.3

(a) (15 points) Write a matlab function that fits the data to a polynomial of degree n using the method of least squares. Make sure that your function allows you to specify n. (Do not use matlab built-in functions `polyfit` or `polyval` except perhaps to check that your code is correct.) Plot the data. On the same axes, plot the polynomial fit for $n = 1$ and $n = 2$. Be sure to clearly label your fit curves.

(b) (10 points) Calculate the residual error in your fitted values and the observed data for n ranging from 0 to 8.

Plot the 2-norm of this residual error against n.

Comment on what does this result says about how to choose the order of your polynomial fit.

(c) (10 points) We improve our drilling and sensing methodology and decide that we can drill to 45m and 50m below ground with minimal effort. We want to estimate the temperature at this new data point.

(i) Provide a table of n versus the predicted temperature at these new data points. It turns out that the temperatures are 48.9 deg. Celsius at 45m and 57.9 deg. Celsius at 50m below ground, respectively.

(ii) Plot the 2-norm of the prediction error only at these two points versus n.

(iii) Comment on what does this result says about how to choose the order of your polynomial fit? Be sure to use what you learned from the previous problem.