AM 147: Computational Methods and Applications: Winter 2022

Homework #9

Instructor: Abhishek Halder

All rights reserved.

Due: March 11, 2022

NOTE: Please submit your Homework as a single zip file named `YourlastnameYourfirstnameHW9.zip` via CANVAS. For example, `HalderAbhishekHW9.zip`. Please strictly follow the capital and small letters in the filename of the zip file you submit. You may not receive full credit if you do not follow the file-naming conventions. Your zip file should contain all .m files (MATLAB scripts) for the questions below.

Your zip file must be uploaded to CANVAS by 11:59 PM Pacific Time on the due date. The uploads in CANVAS are time-stamped, so please don't wait till last moment. Late homework will not be accepted.

Problem 1

Power iteration to compute PageRank (50 points)

We learnt in Lec. 24-25 that for a given Word Wide Web, or equivalently given its adjacency matrix $\boldsymbol{A}$, the PageRank $\boldsymbol{p}$ is simply the dominant eigenvector of the corresponding Google matrix $\boldsymbol{G}$ (see Lec. 25, p. 3–6).

(a) [5 + 10 = 15 points]

Write a MATLAB code `YourlastnameYourfirstnameHW9p1.m` that loads the 500×500 adjacency matrix $\boldsymbol{A}$ from the file `AdjacencyMatrix.txt` in the CANVAS Files Section, folder: HW Problems. Use this $\boldsymbol{A}$ matrix to construct the column-stochastic matrix $\boldsymbol{S}$. You must write the $\boldsymbol{S}$ matrix construction code in a way that dangling nodes are accounted as per Lec. 24, p. 7. In other words, your code should be general enough to handle any $\boldsymbol{A}$ that has one or more zero columns.

(b) [10 + 15 = 25 points]

In the same .m file in part (a), define the damping vector `alpha = 0.25:0.25:1`. Each entry of `alpha` denotes a specific damping factor α in Lec. 25, p. 3. For each entry of `alpha`, perform 10 power iterations on the corresponding Google matrix to compute the associated damped PageRank $\boldsymbol{p}(\alpha)$. For all power iterations, use the initial guess $(1/n, 1/n, \ldots, 1/n)^\top$ with $n = 500$.

Write a MATLAB function `TVdist.m` that takes two probability vectors $\boldsymbol{a}, \boldsymbol{b}$ of same size as input, and outputs the total variation (TV) distance $\mathrm{TV}(\boldsymbol{a}, \boldsymbol{b}) := \frac{1}{2}\|\boldsymbol{a} - \boldsymbol{b}\|_1$, that is, the half of the 1 norm of $\boldsymbol{a} - \boldsymbol{b}$.

For each α, obtain the convergence trend as the following figure. Plot power iteration index k in the horizontal axis, and the "relative error during iteration" $\mathrm{TV}(\boldsymbol{p}_{k+1}(\alpha), \boldsymbol{p}_k(\alpha))$ in the vertical axis using the MATLAB command `semilogy`. To do so, you need to call `TVdist` in the executable `YourlastnameYourfirstnameHW9p1.m`.

This should generate a single figure with four lineplots in four different linecolors: red for $\alpha = 0.25$, green for $\alpha = 0.50$, blue for $\alpha = 0.75$, black for $\alpha = 1.00$. Please also use MATLAB legend in your figure to clarify which curve is which. Review Lec. 25, p. 9 and the lecture video to get intuition on how this plot should look like.

(c) [10 points]
Recall that $\alpha = 1$ is the true (undamped) PageRank. Use the same .m file as in part (a), to print the loss of accuracy in PageRank due to damping in the command window, as the following three % errors:

$$\mathrm{TV}(\boldsymbol{p}(\alpha = 0.25), \boldsymbol{p}(\alpha = 1)) \times 100, \qquad \mathrm{TV}(\boldsymbol{p}(\alpha = 0.50), \boldsymbol{p}(\alpha = 1)) \times 100,$$
$$\mathrm{TV}(\boldsymbol{p}(\alpha = 0.75), \boldsymbol{p}(\alpha = 1)) \times 100.$$

Please submit your two .m files: `YourlastnameYourfirstnameHW9p1.m` and `TVdist.m` within zip file `YourlastnameYourfirstnameHW9.zip` via CANVAS.