18.085, PROBLEM SET 4, DUE 7/14 (IN CLASS)

Question 1. (30 pts.) PageRank

Consider a simplified World Wide Web with three webpages. Webpages #1 and #2 link to each other, as do webpages #2 and #3. In addition, webpage #1 links to webpage #3.

(a) Draw the graph corresponding to the situation described above.

(b) Write the modified adjacency matrix L (which we defined in class) for this graph.

(c) Show that one of the eigenvalues of L is 1, and find the corresponding eigenvector $\mathbf{R}$. This is a solution to the equation $L\mathbf{R} = \mathbf{R}$. Normalize your vector $\mathbf{R}$ so that the sum of its entries is 1. Using your answer, rank the three webpages in order of importance, according to the PageRank metric.

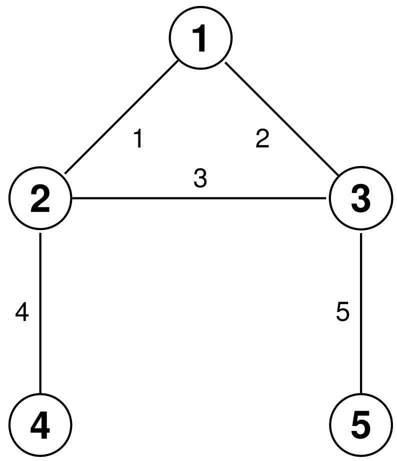
(d) There are over a billion webpages on the World Wide Web, so finding the appropriate eigenvector of L by hand is not practical. However, the method of *power iteration* can be used to quickly approximate the dominant eigenvector of a matrix L. (The dominant vector of a matrix L is the eigenvector with the largest eigenvalue by magnitude.) Beginning with a random vector $\vec{v}_0$, the method is described by the iterative formula

$$\vec{v}_{k+1} = \frac{L\vec{v}_k}{\|L\vec{v}_k\|}$$

It is known that the largest eigenvalue of L is 1. Use power iteration to solve the equation $L\mathbf{R} = \mathbf{R}$ for the graph in part (a), and verify that your answer is close to the one you obtained in part (c). Please attach your code.

Question 2. (40 pts.) Trusses

Consider the truss



Let θ_1 be the angle between edges 1 and 2, and θ_2 the angle between edges 2 and 3. Do not assume specific values for θ_1 and θ_2.

(a) Write down the matrix A for this truss.

(b) Show than any rigid motion of the whole truss satisfies the equation $A\vec{u} = \vec{0}$.

(c) Suppose that nodes 4 and 5 are supported, so they cannot move. Write the new matrix $\tilde{A}$ for this truss, and show that it is unstable (that is, it has a mechanism).

(d) For the truss in part (c), add a bar between nodes 4 and 3. Show that the truss is now stable. Rather than simply giving physical arguments, prove that this truss satisfies the stability criterion.

(e) For the truss in part (c), add a bar between nodes 4 and 5. Is this truss stable? Prove your answer.

(f) Returning to the original truss, fix node 4 and place node 5 on a roller, so that it cannot move vertically but can move horizontally. Write down the matrix $\hat{A}$ for this truss, and show that the truss is unstable. Find the vectors $\vec{u}$ corresponding to each of the instabilities, and verify that they satisfy the equation $\hat{A}\vec{u} = \vec{0}$.

Question 3. (30 pts.) Newton's method

In this question, we will use Newton's method to find the solutions to the system of equations

$$\begin{aligned} x^3 - y &= 0 \\ y^3 - x &= 0 \end{aligned}$$

(a) Find the three solutions r_1, r_2, and r_3 to the above system of equations by hand.

(b) Find the Jacobian matrix corresponding to this system of equations.

(c) Starting with the initial guess $x = 2$, $y = 1$, carry out ten iterations of Newton's method. Please attach your code, and show the output (x and y values) after each iteration.

(d) We will now plot the *basin of attraction* of Newton's method. Specifically, we will start with a number of initial guesses in the region $-5 \leq x \leq 5$ and $-5 \leq y \leq 5$, and color code each initial guess by the solution it finds. If the solution is within 10^{-5} of r_1, we color the initial point red. Similarly, we color the initial point blue for r_2, green for r_3, and black if it fails to find any of the three solutions.

   i. Enter the exact solutions from part (a) into lines 7–9 of the `Matlab` file `PSet4Q3_q.m`. For example, if $x = 2$ and $y = 1$ is a solution, you would write `r1` $= [2; 1]$.

   ii. Starting on line 23 of the `Matlab` file, write the appropriate code for the Newton iteration, using ten iterates. Your code should look very similar to the one you wrote in part (c).

   iii. Submit the plot generated by your code, and describe your findings in words. Is it true that the initial guess always converges to the nearest solution?

Bonus (10 pts.). Newton's method with a vectorized code (*challenging*)

In Question 3(d), you probably noticed that generating the plot took quite a while. To speed this up, we can vectorize the code: rather than looping over all initial guesses, we store them in matrices and execute the Newton iteration on the matrices, thereby updating *all* of the initial guesses at every step of the iteration. Modify the code in `PSet4Q3_vect_q.m` to do this, and submit both the code and the corresponding plot. This should allow you to see the basin of attraction more clearly. Note that your code should only have a **single** loop, rather than the three nested loops in Question 3.

HINT: Since you are storing the initial guesses in the matrices `X` and `Y` (see line 16 of the file), you will not be able to use Matlab's matrix operations to find the solution to $J(u^k)(u^{k+1} - u^k) = -g(u^k)$. You may need the following formula for the inverse of a 2×2 matrix:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$