CS2810 Mathematics of Data Models	Gold
Spring 2019	January 10, 2020

Homework 01

Due: January 17, 9PM
Point total: 60

Instructions:

- Submit your PDF and .py file to Blackboard by the due date and time. Please do not zip your files together, as this interferes with Blackboard's preview functionality. Always show all your work, and for full credit, you must use the method that the problem instructs you to use (unless none is mentioned). Handwritten or typeset solutions are both acceptable, but unreadable submissions will be penalized. You may discuss problems with other students, but you may not write up solutions together, copy solutions from a common whiteboard, or otherwise share your written work or code. Do not use code or language that is copied from the Internet or other students; attribute the ideas *and* rephrase in your own words.

Problem 1 (15 points)

For each system of equations, write the corresponding augmented matrix. Then solve for the solution *using Gauss-Jordan elimination*, describing what you are doing with each step (for example, "row1 = row1 - 2*row2"). To avoid rewriting the same numbers over and over, you can perform multiple steps at a time as long as each step affects a different row. If there is no solution, say so; you can stop as soon as this fact is apparent from Gauss-Jordan elimination. If there are multiple solutions, give equations describing the solutions in terms of z.

Each subproblem (system) is worth 5 points. Students must show a plausible amount of work that suggests they did the work themselves instead of using a solver, else 2 points for the system. If the work is correct but an incorrect conclusion is drawn (multiple vs no solutions, for example), -2. If the answer is incorrect but the student was clearly very close, -2, else just give 1 or 2 points for evidence of correct matrix manipulations. In cases with incorrect single solutions, suggest to students that they check their answers. Not using matrices, but just sticking with variables and equations, is -3.

i. $x + y + z = 3$
$2x + y - z = 8$
$-x - y + z = -3$

<u>*Solution:*</u> $x = 5, y = -2, z = 0$, which we can derive with the following operations (others are possible):

Original augmented matrix:

$$\left[\begin{array}{ccc|c} 1 & 1 & 1 & 3 \\ 2 & 1 & -1 & 8 \\ -1 & -1 & 1 & -3 \end{array}\right]$$

Zero out row2 and row3 by adding -2*row1 to row2 and adding row1 to row3:

$$\left[\begin{array}{ccc|c} 1 & 1 & 1 & 3 \\ 0 & -1 & -3 & 2 \\ 0 & 0 & 2 & 0 \end{array}\right]$$

Scale row3 to lead with 1, then add row3*3 to row2 and row3*-1 to row1:

$$\left[\begin{array}{ccc|c} 3 & 2 & 0 & 9 \\ 0 & -1 & 0 & -6 \\ 0 & 0 & 1 & 0 \end{array}\right]$$

Multiply row2 by -1, then add -1*row2 to row1:

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & 5 \\ 0 & 1 & 0 & -2 \\ 0 & 0 & 1 & 0 \end{array}\right]$$

And thus $x = 5, y = -2, z = 0$.

ii. $x + y = -1$
$-y + z = 2$
$x + z = 1$

<u>*Solution:*</u> Multiple solutions with one degree of freedom (z), with $y = z - 2$ and $x = -z + 1$.

Original augmented matrix:

$$\left[\begin{array}{ccc|c} 1 & 1 & 0 & -1 \\ 0 & -1 & 1 & 2 \\ 1 & 0 & 1 & 1 \end{array}\right]$$

Subtract row1 from row3: $\left[\begin{array}{ccc|c} 1 & 1 & 0 & -1 \\ 0 & -1 & 1 & 2 \\ 0 & -1 & 1 & 2 \end{array}\right]$

Subtract row3 from row2: $\left[\begin{array}{ccc|c} 1 & 1 & 0 & -1 \\ 0 & 1 & -1 & -2 \\ 0 & 0 & 0 & 0 \end{array}\right]$

At this point, there's nothing further to be done (if we try to zero the 1 in the first row's second column, we'll just introduce a term in the third column), and we characterize the multiple solutions according to equations $x + y = -1$ and $y - z = -2$, which can be rearranged to give equations for x and y in terms of z: $x = -z + 1$ and $y = z - 2$.

iii. $w + y + z = 1$
$w + x + y + z = 7$
$y + z = 4$
$w + x = 4$

<u>*Solution:*</u> This system of equations does not have a solution. (The second equation disagrees with the last two about what the sum of all the elements should be, 7 or 8. But we don't have to be clever about this; Gauss-Jordan will run into the problem on its own.)

Original augmented matrix:

$$\left[\begin{array}{cccc|c} 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 7 \\ 0 & 0 & 1 & 1 & 4 \\ 1 & 1 & 0 & 0 & 4 \end{array}\right]$$

Subtract row1 from row2 and row3: $\left[\begin{array}{cccc|c} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 6 \\ 0 & 0 & 1 & 1 & 4 \\ 0 & 1 & -1 & -1 & 3 \end{array}\right]$

Subtract row2 from row3: $\left[\begin{array}{cccc|c} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 6 \\ 0 & 0 & 1 & 1 & 4 \\ 0 & 0 & -1 & -1 & -3 \end{array}\right]$

Add row3 and row4 to zero row4: $\left[\begin{array}{cccc|c} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 6 \\ 0 & 0 & 1 & 1 & 4 \\ 0 & 0 & 0 & 0 & 1 \end{array}\right]$

Now the contradiction becomes clear, because the last row states that 0=1. Thus, no solution.

Problem 2 (8 points)

For each system of equations in problem 1, determine whether the corresponding homogeneous matrix is singular or not, and explain how we know.

Solution: (i) is not singular - we know because there is a unique solution.
(ii) is singular - we know because there are multiple solutions. We could also tell because a row is zeroed out in reduced echelon form.
(iii) is singular - we know because, as with (ii), we have a row zeroed out in reduced echelon form. (If a system has no solutions, it could either be singular or not.)

Worth 2, 3, and 3. Missing at least one explanation is -2, but the explanation can apply to all three problems.

Problem 3 (10 points)

Suppose we have a small online game with a userbase consisting of three kinds of user experience levels: "hardcore," "intermediate," and "just starting." We can represent the counts of these three kinds of users with the letters h, i, and j. From month to month, we estimate that the "just starting" users are growing 20 percent, but that every "hardcore" player causes a "just starting" player to leave. Otherwise, users tend to either stop playing or graduate to the next level of user.

Our modeling department suggests the following linear equations to describe this population's change from month t to month $t+1$:

$$\begin{aligned} h_{t+1} &= 0.95h_t + 0.1i_t \\ i_{t+1} &= 0.8i_t + 0.2j_t \\ j_{t+1} &= -h_t + 1.2j_t \end{aligned}$$

i. Write this linear system as a matrix. (Columns 1, 2, and 3 should correspond to h, i, and j, respectively, and the order of the rows should match the order of the equations.)

ii. Recently, the intermediate players mostly fled after the game changed, leaving 10000 hardcore players, 10000 just starting players, and only 10 intermediate players. Use matrix multiplication to predict the populations one month later.

iii. Perform one more matrix multiplication. What could be done to the transformation from one month to the next to make this prediction more realistic? Is what you suggest a linear operation? (Multiple answers are possible here.)

<u>*Solution:*</u>

i. $\begin{bmatrix} 0.95 & 0.1 & 0 \\ 0 & 0.8 & 0.2 \\ -1 & 0 & 1.2 \end{bmatrix}$

ii. The populations will be 9501 "hardcore" players, 2008 "intermediate" players, and 2000 "just starting" players, or a vector of $\begin{bmatrix} 9501 \\ 2008 \\ 2000 \end{bmatrix}$

iii. One more multiplication produces a vector of: $\begin{bmatrix} 9226.75 \\ 2006.4 \\ -7101 \end{bmatrix}$ This isn't exactly realistic because of the negative number, so we might choose to create a minimum value of 0 as a result of each operation. But, this max(x,0) operation is not linear. Another possibility would be to have the value only asymptotically approach zero somehow, but this must result in some kind of curve, and thus be nonlinear.

2 points each for the first two parts (writing the matrix and performing the matrix multiplication.) The third part is worth 6 points: 2 points for the multiplication result, 2 points for the proposed fix being reasonable, and 2 points for identifying whether the fix is linear. Students may suggest other nonlinear changes instead of the min or clipping operation, such as asymptotically approaching zero. Anything is fine as long as the change is correctly identified and it doesn't cause the system to make no sense. If the student's fix is to conditionally swap between matrices and

call the fix linear, this is okay but make a note that the system as a whole is now not linear, since it can't be represented by a single matrix.

Problem 4 (12 points)

A common problem in the field of information retrieval is to try to identify documents that are similar to each other. If we want to use methods that are related to linear algebra, but the documents are in English, then we need to first convert those documents (composed of words) to vectors (composed of numbers). Whatever method we use should cause similar documents to turn into vectors that point in similar directions.

One way to achieve this is to have each word correspond to a different dimension of a vector. A document that contains that word could contain a 1 for that dimension, while a document that does not contain the word would contain a zero for that dimension. In the simpler versions of these methods, word order doesn't matter; just the presence or absence of the word does.

For simplicity, let's assume our product reviews on a website just consist of the words "good" (dimension 1), "works" (dimension 2), "terrible" (dimension 3), "car" (dimension 4), and "vacuum" (dimension 5). Thus, "good car" would be the vector $\begin{bmatrix}1\\0\\0\\1\\0\end{bmatrix}$ and "terrible vacuum" would be the vector $\begin{bmatrix}0\\0\\1\\0\\1\end{bmatrix}$.

Using this system, and the formulas for the dot product, find the angle between each pair of sentences. Be exact if possible (if using radians, give any nonzero answer in terms of π) and assume angles are between 0 and π inclusive.

i. "good car works" vs "terrible vacuum"

ii. "good car" vs "car works"

iii. "terrible vacuum" vs "vacuum works"

<u>*Solution:*</u>

i. $\begin{bmatrix}1\\1\\0\\1\\0\end{bmatrix} \bullet \begin{bmatrix}0\\0\\1\\0\\1\end{bmatrix} = 0 = |A||B|\cos\theta$

$\cos\theta = 0$ so $\theta = \pi/2$ or 90 degrees.

ii. $\begin{bmatrix}0\\1\\0\\1\\0\end{bmatrix} \bullet \begin{bmatrix}0\\1\\0\\1\\0\end{bmatrix} = 1 = |A||B|\cos\theta$

Each vector is length $\sqrt{2}$, so $2\cos\theta = 1, \cos\theta = 1/2$, and $\theta = \pi/3$ or 60 degrees.

iii. $\begin{bmatrix}0\\0\\1\\0\\1\end{bmatrix} \bullet \begin{bmatrix}0\\1\\0\\0\\1\end{bmatrix} = 1 = |A||B|\cos\theta$

The math here works out the same as the previous part, and $\theta = \pi/3$ or 60 degrees. (Notice that in one case they agreed about the sentiment, and in the other, they agreed about the subject.)

4 points for each problem: 1 point for each vector successfully converted, 1 point for finding the dot product, 1 point for finding the angle.

Problem 5 (15 points)

For this part, you'll learn to use some Python. The instructions for both the tutorial and the graded part of Problem 5 are in Jupyter notebook files, which are a mix of instructions and executable code.

First, install the Anaconda distribution of Python 3, available at https://www.anaconda.com/distribution/. If you already have a Python install, you might want to do this anyway so that we're all on the same page.

Next, open up a Terminal window (Mac) or Command Prompt (Windows) and type

```
jupyter notebook
```

From there, open up the tutorial part of the assignment (`hw01_program_tutorial.ipynb`), and walk through it to explore the parts of Python that we'll be using. After that, you can open the other notebook (`hw01_program.ipynb`), which contains the actual programming assignment (about repeatedly multiplying matrices).

For this assignment, turn problems 1-4 in to Blackboard in a single PDF, and also turn in problem 5 as a separate .py file. The tutorial doesn't have anything you need to turn in, but is recommended regardless.

Solution: The results of the multiplications are:

$$A^{100}\vec{c} = A^{100}\vec{d} = \begin{bmatrix}0.24\\0.23\\0.53\end{bmatrix}$$

$$A^{100}\vec{e} = \begin{bmatrix}0.71\\0.69\\1.59\end{bmatrix}$$

$$B^{100}\vec{c} = B^{100}\vec{d} = \begin{bmatrix}0.33\\0.33\\0.33\end{bmatrix}$$

$$B^{100}\vec{e} = \begin{bmatrix}1\\1\\1\end{bmatrix}$$

The vectors for matrix A are all pointing the same direction, and the vectors for matrix B are all the same as each other as well. The lengths of the output vectors aren't the same, though – those are determined by the lengths of the input vectors.

Part A: 3 points for having the right matrices and vectors defined, -1 per typo or similar to a max of -3. Part B: 7 points for the program, using the output in part C as a correctness check.

Reversing the order of the arguments or a similar error that messes everything up while calling the right things is -3. Failure to iterate properly (or using a method that avoids iteration) is -4. Having a fencepost error, such as starting with X and multiplying by X 100 times before applying to the vector, is -1. Part C: 5 points total: 3 points for realizing the results are pointing the same direction, 1 point for realizing that they are not the same length, 1 point for realizing that the longer vectors are producing longer results. Max 1/5 for Part C if the student's conclusions fundamentally rest on flawed output.