MSCS 6040 – TAKE HOME EXAM

assigned: Wednesday, December 4
due: Thursday, December 12, **by 6:00pm local time**

Instructions:

1. You must explain all reasoning. It is not sufficient to just write the correct answer. Correct answers with no justification will receive no credit.
2. You must submit your completed exam on D2L by **6:00pm on Thursday, December 12**.
3. **In order to speed up grading, you must submit files in the following way.**

   Submit, for each question, precisely 1 ".mlx" file (not a ".m" file) and precisely 1 ".pdf" file which is the exported version of the submitted ".mlx" fie. Name these files "[lastname]-[question number].[file extension]". If you must submit an additional file for a question, name it "[lastname]-[question number]-[descriptive information about what the file is].[whatever]".

   So, for example, you submitted files might look like:
   Pantone-1.mlx
   Pantone-1.pdf
   Pantone-2.mlx
   Pantone-2.pdf
   . . .

   Note that you can write text and explanation directly inside of the ".mlx" file.
4. For any questions that ask you to write code and produce output, you **must** submit a ".mlx" file so I can run your code and verify the output.
5. **RULES FOR OUTSIDE SOURCES:**
   It is permissible to use:

   (a) Our textbook, or any other published books or articles.
   (b) *Your* course notes.
   (c) The internet for base-level questions: Matlab synatax, basic definitions of terms, etc.
   (d) Me, via email or office hours.

   It is not permissible to use:

   (a) Any classmates, other students, other professors, etc.
   (b) Anybody else's course notes (it is okay if you had previously borrowed someone else's notes because you missed a day, etc).
   (c) The internet for anything remotely approaching the actual question asked.

   For example, it is okay to search the internet for "How to compute the SVD in Matlab", but it is not okay to search the internet for the actual question.

   If you are in doubt about what is okay, email me!

   Any instances of plagiarism, working with classmates, or other cheating, will result in a score of 0 for the entire final exam.

The Marquette University honor code obliges students:

- To fully observe the rules governing exams and assignments regarding resource material, electronic aids, copying, collaborating with others, or engaging in any other behavior that subverts the purpose of the exam or assignment and the directions of the instructor.
- To turn in work done specifically for the paper or assignment, and not to borrow work either from other students, or from assignments for other courses.
- To complete individual assignments individually, and neither to accept nor give unauthorized help.
- To report any observed breaches of this honor code and academic honesty.

This exam has five questions, each worth 25 points. Choose any four to complete. Do not submit more than four answers (if you do, only the first four will be graded.

1. The goals of this question are to introduce you to an important application of linear algebra that we did not have time to cover in this course, as well as to test your ability to use the theoretical concepts we learned to understand and apply relevant techniques to your own research.

   Principal Component Analysis is a linear algebra technique used in many applications, including machine learning, community detection, and regression. It is, at the most basic level, a way to take high-dimensional data and reduce its dimensionality in a manner that captures the most possible information.

   For example, you may conduct a medical study that gathers 100 data points from each of 50 patients. Suppose 25 of the patients come from one neighborhood, while the other 25 come from another, and you want to know whether there is any appreciable difference in the 100 parameters between the two neighborhoods. In this case, PCA could be used to reduce the dimensionality to 2, and a simple x/y-plot would visually tell you whether these two communities have significantly different measurements.

   In this exercise, you will read a brief page about how PCA works, and then use Matlab's built in routines to perform the analysis.

   First, read the following two webpages. The first is a nice interactive overview without any detail, and the second explains how it works. Both should be easy reads given the material we've learned in class.

   `http://setosa.io/ev/principal-component-analysis/`
   `https://towardsdatascience.com/a-one-stop-shop-for-principal-component-analysis-5582fb7e0a9c`

   Now, you will use PCA to analyze a data set built into Matlab of car models between 1970 and 1982. You will perform PCA on this data set to determine whether certain characteristics can be predicted from others.

   To load data from the car dataset, type "`load carbig.mat`". This will create a bunch of column vectors each with 406 rows that give information about 406 different models of car that were released between 1970 and 1982. You can use the command "`who`" to list currently existing variables. The variables are:

`Model`:	model name
`Origin`:	the country the car was made in
`org`:	a less refined version of `Origin`, only 'USA', 'Europe', and 'Japan'
`Model_Year`:	the year the car was made
`MPG`:	the fuel efficiency
`Cylinders`:	number of cylinders
`Displacement`:	volume of the engine
`Acceleration`:	acceleration speed of the car
`Horsepower`:	power of the car
`Weight`:	weight of the car
`Mfg`:	manufacturer of the car

   The variables `MPG`, `Displacement`, `Acceleration`, `Horsepower`, `Weight` are the five *numeric* variables—their values are just real numbers. The variables `Origin`, `org`, `Model_Year`, `Cylinders`, and `Mfg` are *categorical* variables—their values come from some finite set.

   The goal is to use the five numeric variables to see if we can predict a categorical variable. For example, if you know a car's MPG, displacement, acceleration, horsepower, and weight, can you predict the country that built it or how many cylinders it has?

   First, put the numeric variables into a 406 × 5 matrix with the command

```
numeric = [MPG Displacement Acceleration Horsepower Weight]
```

(a) Perform PCA on the `numeric` matrix using Matlab's built-in `pca` command. Plot a two-dimensional scatterplot of the first two principal components, coloring the points of the scatter plot according to the `Origin`. You will find the "`gscatter`" command very useful for this. What does this plot tell you? If you see big clumps of one color, that means cars from that origin have inherently different properties than those from others. Another way to think of this is that if you're given numeric data for a car that falls in that region, you can predict its origin pretty well. If all the colors are evenly mixed together, then you can't predict its origin.

(b) Do the same thing, but plotting according to `org`. Analyze the result.

(c) Do the same thing, but plotting according to `Mfg`. Analyze the result.

(d) Do the same thing, but plotting according to `Cylinders`. Analyze the result.

(e) Repeat parts (a) - (d) with a *three-dimensional* scatterplot, showing the first three principal components. You will find the "`scatter3`" command very useful.

You may use the internet as needed to learn about PCA and how to perform it in Matlab. You may not use the internet to search for anything specific about this dataset.

2. Implement Arnoldi Iteration (Algorithm 33.1) in Matlab. Use your algorithm to find an orthonormal basis for the Krylov subspace K^5 for the matrix

$$A = \begin{bmatrix} -2 & -7 & 5 & 6 & 5 & 4 & 4 & -6 \\ 10 & 3 & 1 & 3 & -4 & 7 & 6 & 1 \\ -2 & 6 & -4 & 7 & 10 & -2 & 0 & 7 \\ -10 & 5 & -1 & 7 & 9 & 7 & -8 & -8 \\ -10 & 5 & -10 & -5 & 4 & -7 & 1 & -1 \\ -5 & -6 & -2 & -1 & 8 & 2 & 9 & -7 \\ -6 & 6 & -10 & -5 & 10 & 0 & 9 & -10 \\ 7 & 9 & 0 & -1 & 9 & -8 & -7 & -3 \end{bmatrix}$$

matlab code:

```
[-2 -7 5 6 5 4 4 -6; 10 3 1 3 -4 7 6 1; -2 6 -4 7 10 -2 0 7; -10 5 -1 7 9 7 -8 -8; -10 5
  -10 -5 4 -7 1 -1; -5 -6 -2 -1 8 2 9 -7; -6 6 -10 -5 10 0 9 -10; 7 9 0 -1 9 -8 -7 -3]
```

and the vector

$$b = \begin{bmatrix} 3 \\ -7 \\ -3 \\ 3 \\ -5 \\ -6 \\ -7 \\ 7 \end{bmatrix}$$

matlab code:

```
[3; -7; -3; 3; -5; -6; -7; 7].
```

3. Read the following well-written and entertaining article about interesting applications of the SVD:

```
https://people.maths.ox.ac.uk/porterm/papers/s4.pdf
```

Then, write 200 words about one of the applications they talk about, or any other application you find online other than the ones we did in class (image compression and background removal). *You must write in your own words.* Explain what the application is, why the SVD is an appropriate tool for it, and what information can be learned from it. Answers that are insightful and demonstrate understanding of the underlying method (and its limitations) will earn full points.

4. Use the following commands in Matlab to generate a 100×100 symmetric matrix with random entries between -10 and 10:

   ```
   rng(123)
   M = randi(21,100,100) - 10;
   M = M - tril(M,-1) + triu(M,1)'
   ```

   The "rng" command sets the random seed to a particular number, so everybody should end up with the same matrix (which makes it easier for me to verify your answers). Implement the "Phase 1 / Phase 2" process for computing eigenvalues, consisting of the direct Algorithm 26.1 and the iterative Algorithm 28.2, and use your implementations to compute all of the eigenvalues and eigenvectors of M.

5. The heart of Google's methodology of indexing the web is the PageRank algorithm. It relies on principles of linear algebra to classify and rank websites by order of importance so that a Google search can yield the most relevant results.

   First, read this short article about how the PageRank algorithm works:

   `http://home.ie.cuhk.edu.hk/~wkshum/papers/pagerank.pdf`

   Another useful resource may be

   `http://pi.math.cornell.edu/~mec/Winter2009/RalucaRemus/Lecture3/lecture3.html`.

   (a) After forming the weighted matrix from all of the internet links (as described in the links above), it is necessary to compute the largest eigenvalue and eigenvector for that eigenvalue. Explain informally (without rigorous proof, but using conceptual ideas) why finding this eigenvalue / eigenvector pair tells us the solution to the question "What is the limit of the sequence $b, Ab, A^2b, A^3b, \ldots$?" as well as why that's the right question to ask?

   (b) Explain how you would calculate this eigenvalue / eigenvector pair, given that the matrix involved has dimensions well over $10^9 \times 10^9$. Which algorithms from class would you use? In what order? Does the matrix have any special properties that could be used to develop an algorithm faster than the most general one?