**Technique Assignment 5: K-means clustering**
Cogs 109 Spring 2020
Due: Friday June 5 11:59pm
**100 points**

**You may complete this assignment with a group** of up to 4 students**.** If you do, please clearly indicate both names at the beginning of your pdf report.
Submit your completed assignment on Gradescope as a pdf report containing all code, graphs, and answers to questions. Include comments for clarity.

Download the Jupyter notebook and the data file cluster_data.mat from Canvas. You may fill in the blank spaces of the provided notebook or create your own document.

1. (25 points) K-means step by step, k=2

   Extract `kmeandata` from `cluster_data.mat` as your dataset.

   a. Initialize the cluster centers (the means) at (-2,-2) and (4,4)
   b. Calculate the distance between each data point and each center (you should have an array containing a row for every data point and a distance value for each cluster mean).
   c. Assign each data point with the label of its nearest cluster (`cluster_ind`)
   d. Plot the clustered data points using different colors for each cluster
   e. Update the means
   f. Continue for 4 iterations, creating one plot for each iteration inside a 2x2 subplot.
   g. Repeat for a new initialization of cluster centers: (0, -1) and (-1, 4)

2. (5 points) K-means step by step, k=3

   Follow the same instructions as above, using two different cluster initializations:

   Centers = (2,-2), (-2,-2), (2,2) and

   Centers = (0,-3), (0,0), (0,3)

3. (5 points) K-means step by step, k=4

   Follow the same instructions as above (only one cluster initialization)

   Centers = (0,-3), (0,-1), (0,1), (0,2)

4. (10 points) Which value of k will produce the best result? How can you tell?

5. (30 points) Using the same k-means algorithm as above, cluster the data, but only plot the final results. The convergence criterion is when the total distance change between centers at two different iterations is less than 0.001 (or you can simply run the algorithm for a large number of iterations, like 200).
   Create a figure with 2x2 subplots. In each subplot, plot the final result of k-means clustering with different k values. Subplot 1: k=4; Subplot 2: k=8; Subplot 3: k=12; Subplot 4: k=20

6. (25 points) Using the same k-means algorithm, run k-means for k = 1 through 25. You will need to use a for loop. For each value of k, store the total distance of all data points from their cluster centers.
   a. (15) Plot the total distance for each value of k. Which k do you think explains the data best?
   b. (5) You may notice that your distance graph does not follow a smooth-line trend. If it doesn't, can you explain why there are "jumps" in the distance values?
   c. (5) Run your code for generating the total distances, but this time use a loop to repeat the process a few times (~5 or more). Store the **minimum** total distance for each value of k and use this to create a new version of your plot from part (a).

7. Extra credit (up to 10 points) Perform k-means clustering on any dataset with more than 2 dimensions (try around 500 samples). You may run until the means converge or run a large number of iterations if you have a large dataset (watch out for timing if your dataset is too large). Indicate how many clusters you think are in the dataset and justify your answer with calculations and/or a plot.