ECE 3502 / CS 3501: Foundations of Data Analysis **Due Tu 2/25**

Homework 2: Hypothesis Testing & K-means Clustering

Instructions: Submit a single Jupyter notebook (.ipynb) of your work to Collab by 11:59pm on the due date. All code should be written in Python. **Be sure to show all the work involved in deriving your answers! If you just give a final answer without explanation, you may not receive credit for that question.**

You may discuss the concepts with your classmates, but write up the answers entirely on your own. Do not look at another student's answers, do not use answers from the internet, and do not show your answers to anyone.

1. (30%) Write a Python function that computes the probability function for a hypergeometric random variable, X. (See the class notes and Wikipedia page for this formula.) Your function should take inputs:

   $N =$ number of available bits to select from
   $K =$ number of available bits that are 1
   $n =$ number of bits drawn at random
   $k =$ number of bits drawn that are 1

   Your function should return $P(X = k)$. Using your function, compute the following:

   (a) Recall the "lady drinking tea" example from class. Verify that your function gives the correct values for $k = 2, 3, 4$. (See the notes for the right answers!)

   (b) You are running an internet security firm trying to catch packets sent to a server by hackers. There are 100 packets sent to the server, with 10 of them from hackers, 90 from legitimate traffic. If you sample 50 packets at random, what is the probability that you will capture all 10 packets from the hackers?

   (c) What is the chance that you will capture at least half of the hackers' packets? That is, what is $P(X \geq 5)$? **Hint:** You are going to need to sum probabilities from multiple calls to your function.

2. (20%) Here we are going to test a hypotheses about cardiac measurements from the following data: `http://www.stat.ucla.edu/projects/datasets/cardiac.dat`

   Download this data set and load it into Python. It is just a CSV file, so you can load it the same way you have in the previous homework.

   To understand what the variables mean, read the description of the data set here:
   `http://www.stat.ucla.edu/projects/datasets/cardiac-explanation.html`

   You want to test the hypothesis that women are more likely to have hypertension (high blood pressure) than men. Hypertension is the variable `hxofHT` (be careful, `hxofHT = 0` indicates they **do** have hypertension) and `gender` is male = 0, female = 1.

(a) What is the 2×2 contingency table for this data? The rows of your table should be `gender` and the columns should be `hxofHT`. The four entries of the table will be counts from the data. For example, one entry will count the number of people who are both women (`gender = 1`) and have hypertension (`hxofHT = 0`), etc.

(b) Using your hypergeometric probability function from the previous question, compute the probability of getting *exactly* this table.

(c) If you want to test if women have hypertension more frequently than men, what is the null hypothesis?

(d) Again, using your hypergeometric probability function, perform the Fisher exact test to get a p value for the hypothesis that women have hypertension more frequently than men. Can you "reject the null hypothesis" with the threshold $p \leq 0.05$?

3. (50%) We experiment with unsupervised learning: K-means. Your implementation functions are: (1) `clusterInit`, which should initialize random clusters according to the "furthest first" heuristic (these "real" centers should be as far as possible from any of the previous centers, e.g., if you've already selected four centers, the fifth center should be the point whose distance to any of the 4 centers is *maximum*.), and (2) `test_kmeans` for testing the implementation on a simple 2D data (`provided as "2D_data.txt"`) If you don't see reasonable clusters coming out, something is probably broken.

   Your results should include:

   - A plot of the data, unclustered.
   - The data clustered with $K = 2$ and random initialization.
   - The data clustered with $K = 3$ and random initialization (run 20 times... you should see a small amount of variability in the outputs). The plots also include distance scores.
   - The data clustered with $K = 3$ and "furthest" initialization (run 20 times... you should see a small amount of variability in the outputs). The plots also include distance scores. Note that this won't work until you do the furthest-first initialization below.
   - A plot of $K \in \{4, 6, 8, 10, 15, 20\}$ versus distance score.

   Note: Please use different colors for different clusters.

   In this experiment, test your K-means clustering with the furthest-first initialization on the MNIST handwritten digits database (`provided as "trainX.txt", "trainY.txt", "testX.txt", and "testY.txt"`) with different values of K.

   - Report the means found for $K = 5, 10, 15$ (you have to reshape each row of the "trainX" or "testX" as a 28^2 image).
   - For each of these, do you see clusters that look like the actual digits? How big does K have to be before you essentially see a mean for each "true" digit? Why do you suppose some digits are "over-represented?"