ECE 3502 / CS 3501: Foundations of Data Analysis **Due Th 2/6**

Homework 1: Getting Started With Probability

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. Say 16% of the population is rich, 10% of the population is famous, and 8% of the population is both rich and famous. Define events R = "person is rich" and F = "person is famous" for some randomly selected person in the population. Write an expression for each of the following events using set operations involving the events R and F. **Here you can just give the answer, and do not need to show any work.**

   Example: "The person is rich and famous" would be the event $R \cap F$.

   (a) The person is not famous.

   (b) The person is rich but not famous.

   (c) The person is either rich or famous (or both).

   (d) The person is neither rich nor famous.

2. Calculate the probabilities for each of the four events in the previous problem. **Be sure to show your intermediate steps and list any probability rules that you use.**

   Example: For the "rich and famous" event, the probability is $P(R \cap F) = 0.08$

3. You have two \$1 bills, two \$5 bills, and two \$10 bills. If you pick two bills at random (without looking!), what is the probability that they are the same denomination? Note: you pick the second bill *without replacing* the first bill. **Explain how you arrived at the answer, don't just give a number!**

4. Use Python to simulate the "trick coin" example from class. Do the following steps:

   - First, simulate 10,000 coin selections, that is, generate a random binary list (0 = normal coin, 1 = trick coin).
   - Second, simulate 10,000 flips of these coins. This is another binary list (0 = tails, 1 = heads), conditioned on the type of coin from the first list.
   - Third, using your two binary lists, estimate the conditional probability of picking the trick coin given that you flipped a heads. **Hint:** this should be a ratio of two counts from your lists.

5. In this exercise we will be using data from the OASIS brain database, a publicly-available resource:

   `http://www.oasis-brains.org`

   You will be classifying dementia from the volume of the hippocampus, a brain structure that is critical to memory. The data consists of the hippocampal volume, derived from MRI, for elderly subjects, including healthy control subjects and those with mild to moderate dementia. First, download the data from the class website.

   Model the right hippocampal volume (`RightHippoVol`) as a normal random variable X_1 and the left hippocampal volume (`LeftHippoVol`) as a normal random variable X_2. Then model the diagnosis (`Dementia`) as a binary random variable Y ($Y = 0$: healthy control, $Y = 1$: dementia). Use the training subset of the data (`TrainData` $= 1$) to learn the mean and variance parameters for a naïve Bayes classifier. Finally, apply your naïve Bayes classifier to get a probabilistic diagnosis of the test subset of the data (`TrainData` $= 0$).

   Do the following:

   - Plot the data as a 2D scatterplot (right and left hippocampal volume as the two axes). Use two different colors for the two classes (healthy/dementia). Do you think there is separation between the two classes?
   - Plot two density plots for the left and right hippocampus volumes. Again, plot a different density for the two classes (with the same colors as your scatterplot).
   - Run your classifier on the testing data. For each data point, if your classifier probability is greater than 0.5, predict that it is a dementia patient. Then compare with the actual label to see if your classifier is correct. How many subjects did you correctly diagnose?

Note: There are Python machine learning libraries that include a method for naïve Bayes. You can't use these to solve this problem! You have to write your own code to implement naïve Bayes. However, it's okay to try out a library's version to check if it gives a similar answer to your code.