CS 542: Machine Learning	Spring 2018

Problem Set 1

Lecturer: Prof. Peter Chin   *Due: February 8, 2018*

◇ Please email the written portion (either type up your answer or scan your handwritten solution) & code and report to `ken.q.zhou@gmail.com` by 23:59PM on the due date.

◇ Late policy: there will be a penalty of 10% per day, up to three days late. After that no credit will be given.

1. **Probability**

    (a) Bishop 1.6

    (b) Table 1 represents a hypothetical study of a test for HIV in a population of intravenous drug users, some carrying HIV and others not. "+" denotes a positive outcome from the test and "–" denotes a negative outcome.

Outcome	with HIV	without HIV	Total
+	72	12	84
–	3	71	74
Total	75	83	158

    Table 1.1. HIV test summary for intravenous drug users

        i. Calculate the probability of test being positive given the patient has HIV.

       ii. Calculate the probability of test being negative given the patient has HIV.

      iii. If the prevalence of HIV is 12% in intravenous drug users (12 cases per 100 patients), what is the probability of the patient having HIV after a positive test? A negative test?

2. **Bayes Theorem**

    (a) Monty Hall problem: On the game show, Let's make a Deal, you are shown four doors: A, B, C, and D, and behind exactly one of them is a big prize. Michael, the contestant, selects one of them, say door C, because he know from having watched countless number of past shows that door C has twice the probability of being the right door than door A or door B or door D. To make things more interesting, Monty Hall, game show host, opens one of the other doors, say door B, revealing that the big prize is not behind door B. He then offers Michael the opportunity to change the selection to one of the remaining doors (door A or door D). Should Michael change his selection? Justify your answer by calculating the probability that the prize is behind door A, the probability that the prize is behind door C, and the probability that the prize is behind door D, given that Monty Hall opened door B (to show that prize is not there), using Bayes Theorem.

(b) Prof. Chin knows that historically 2 out 75 students in his CS 542 class cheats (!) on his exams. Last semester, he suspected one of the students engaged in cheating during the final, but when he gently confronted the student, the student vehemently denied that he was cheating. If Prof. Chin has 90% accuracy in identifying cheaters (i.e. if a student is cheating, 90% of the time, Prof. Chin will indeed identify that the student as a cheater), but also has 20% false alarm rate (i.e. even though a student is not cheating, Prof. Chin will erroneously identify that student as a cheater). What is the probability that the student Prof. Chin confronted in the last semester's final was indeed cheating?

3. **Linear Algebra**

(a) Bishop 2.22

(b) Find the eigenvalues and eigenvectors for

$$A = \begin{bmatrix} 3 & 4 & -1 \\ -1 & -2 & 1 \\ 3 & 9 & 0 \end{bmatrix}$$

Is A positive definite?

4. **Probability Distributions**

Bishop 2.2, 2.10*, 2.12, 2.15*

5. **Programming assignment - can you beat the computer in rock-paper-scissors? - a case of counter - Artificial Intelligence**

http://www.nytimes.com/interactive/science/rock-paper-scissors.html?_r=0.

Visit the URL above. I quote from the paper:

> A truly random game of rock-paper-scissors would result in a statistical tie with each player winning, tying and losing one-third of the time. However, people are not truly random and thus can be studied and analyzed. While this computer won't win all rounds, over time it can exploit a person's tendencies and patterns to gain an advantage over its opponent.

Now play the game of the rock-paper-scissors against the computer at least 100 times while you record the outcome (your moves & the computer's move) at each time into a text file. Record the score at the bottom of the file, i.e. your number of wins, losses and ties out of N tries ($N \geq 100$). Now use this file to come up with some algorithm that takes as input the history to how the game has been played and estimate your best move for the next round. Put this algorithm into a simple program (use Java, Matlab, python, C, etc., or whatever you like) and play the computer again at least the same number of times you played before, but this time using your program. Record the score (hopefully there is an improvement in your score!!). *Hint: the more you play the better you may be at guessing*

what the computer is up to. Also, note that we haven't covered any ML techniques yet in the class yet, so this is just a warm up problem, so just do the best you can...