CS378 Assignment 0: Review of Basics

Due date: Friday, January 25 at 5:00pm CST

Academic Honesty: Reminder that assignments should be completed independently by each student. See the syllabus for more detailed discussion of academic honesty. Limit any discussion of assignments with other students to clarification of the requirements or definitions of the problems, or to understanding the existing code or general course material. Never discuss issues directly relevant to problem solutions. Finally, you may not publish solutions to these assignments or consult solutions that exist in the wild.

Goals The main goal of this assignment is for you to **assess whether you have adequate preparation for the course.** It's fine to not be familiar with every concept here. However, if you find yourself struggling with much of this assignment, you should ask the course staff whether this course is appropriate for you given your background. This assignment is designed to take 1-2 hours.

Grading The assignment is out of 100 points. Note that it is worth half as much as the other assignments, so your grade on Canvas will show up as your points divided by 2.

1 Linear Algebra (25 points)

Q1 (15 points) For each of the following matrices, give the answer or write "undefined" if the operation is invalid. You do not need to show work.

$$\text{a)} \begin{bmatrix} 1 & 5 & 4 \\ 3 & 4 & 5 \end{bmatrix} \begin{bmatrix} 4 \\ 2 \end{bmatrix} \qquad \text{b)} \begin{bmatrix} 1 & 5 & 4 \\ 3 & 4 & 5 \end{bmatrix} \begin{bmatrix} 4 & 2 \end{bmatrix} \qquad \text{c)} \begin{bmatrix} 1 & 5 & 4 \\ 3 & 4 & 5 \end{bmatrix} \begin{bmatrix} 4 \\ 5 \\ 2 \end{bmatrix}$$

$$\text{d)} \begin{bmatrix} 6 & 2 & 4 \end{bmatrix}^\top \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \qquad \text{e)} \begin{bmatrix} 6 \\ 2 \\ 4 \end{bmatrix}^\top \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \qquad \text{f)} \begin{bmatrix} 1 & 5 & 4 \\ 3 & 4 & 5 \end{bmatrix} \begin{bmatrix} 2 & 2 & 4 \\ 1 & 3 & 2 \\ 4 & 4 & 1 \end{bmatrix}$$

Q2 (10 points) Write a matrix operation capturing the following computation. Your answer should be a mathematical expression involving A, B, and C. Your math expression does not need to account for initialization of A, B, and C; it only needs to return the same value as sum given the same inputs.

```
A = np.rand(3)
B = np.rand(3,2)
C = np.rand(2)
sum = 0.0
for i in range(0,3):
  for j in range(0,2):
    sum += A[i] * B[i,j] * C[j]
return sum
```

2 Probability (40 points)

Q1 (10 points) Consider the following joint distribution:

$P(X, Y)$	$Y = 1$	$Y = 2$	$Y = 3$
$X = 1$	0.1	0.2	0.2
$X = 2$	0.05	0.1	0.1
$X = 3$	0.1	0.1	0.05

a) What is $P(X|Y = 2)$?

b) What is $P(Y|X = 1)$?

c) Are X and Y independent? Justify your answer.

Q2 (15 points) Suppose you have a distribution $P(X, Y)$ where $X \in \{0, 1\}$ and $Y \in \{0, 1\}$. You know that the marginal distribution $P(X) = [0.5, 0.5]$ and $P(Y) = [0.2, 0.8]$.

a) If X and Y are independent, what do we know about the value of the joint probability $P(X = 0, Y = 0)$? Give upper and lower bounds as precise as you can.

b) If X and Y are not independent, what do we know about the value of the joint probability $P(X = 0, Y = 0)$? Give upper and lower bounds as precisely as you can.

Q3 (15 points) The entropy of a random variable X with discrete domain D is defined as:

$$H(X) = -\sum_{x \in D(X)} P(x) \log P(x)$$

a) Compute the entropy of $P(X) = \text{Multinomial}([0.1, 0.3, 0.6])$. You should evaluate your answer numerically.

b) Compute the entropy of $P(X) = \text{Multinomial}([\frac{1}{n}]_{i=1}^{n})$, the uniform distribution over n variables. Your answer should be written symbolically.

c) When you have a joint distribution over X and Y, define the entropy:

$$H(X, Y) = -\sum_{x \in D(X)} \sum_{y \in D(Y)} P(x, y) \log P(x, y)$$

How does this relate to the entropy of the marginal distributions $P(X)$ and $P(Y)$ when X and Y are independent? How about when they are not independent?

3 Language Basics / Coding Warmup (35 points)

In this part of the assignment, you will read in and do some basic manipulation of a text corpus. Included with the assignment is a file `nyt.txt` containing 8860 sentences taken from New York Times articles, one sentence per line. You should implement your solutions in a file called `a0.py`.

Q1 (15 points) Here you will investigate tokenization schemes. Tokenization is the process of splitting raw text into words. In English, this involves splitting out punctuation and contractions (*shouldn't* becomes *should 'nt*) and is typically done with rules. In other languages like Chinese or Arabic, the process can be significantly more involved.

a) What are the 10 most frequent words in this dataset using whitespace tokenization? That is, split each sentence into words simply based on where the spaces are. List each word and its count and describe any patterns you see.

b) What are the 10 most frequent words in this dataset using smarter tokenization? You can either use the tokenizer in `tokenizer.py` or invoke the NLTK tokenizer with `nltk.word_tokenize(sentence)` after importing NLTK. List each word and its count and describe any patterns you see.

c) Explain in a few sentences how these differences in tokenization could affect a downstream text processing system. Discuss at least two ways.

Q2 (20 points) In this part, we are going to confirm a phenomenon known as Zipf's Law. A word has *rank* n if it is the nth most common word. Zipf's Law states that the frequency of a word in a corpus is inversely proportional to its rank. Roughly speaking, this means that the fifth most common word should be five times less frequent than the most common word, and the tenth most common word should occur half as much as the fifth most common word.

a) Make a plot of inverse rank vs. word frequency for the tokenized data. Include your plots in your submission. Matplotlib is a good tool to use, but Excel/Matlab/Gnuplot/others are okay too.

b) Where does Zipf's law appear to hold? Are there any outliers?

Submission

If you wish to complete either Parts 1 or 2 of this assignment on paper, you may. Submit it in person by 5pm Friday at Greg's office, GDC 3.420 (slip it under his door if he's not there).

You should upload two files to Canvas:

1. A PDF of your answers/output; this might contain your answers to the whole assignment, or just Part 3
2. A single python file called `a0.py`. There are no specific requirements for what the code has to do, but it should provide sufficient evidence that you have completed this part of the assignment. For example, displaying the plot from part (a) is fine, or printing out the values that you used to create it is fine too.