

---

## CSE 150A / 250A - Homework 1 (40 pts)

**Due:** *Monday, Oct 7* (by 11:59 PM, Pacific Time, via gradescope)

**Grace period:** 24 hours

---

### 1.0 General Guidelines

You should submit your homework assignments via Gradescope. Typesetting / LaTeX is preferred, but neatly handwritten solutions are also accepted. Upload a PDF of your answers to Homework 1 on Gradescope, and carefully select the pages corresponding to each question using the submission interface. If you are submitting handwritten answers, please scan them and create a PDF for upload. Here is a primer on submitting PDF homework via Gradescope:

<https://tinyurl.com/gradescope-guide>

If you have not done this before, please allow some extra time to familiarize yourself with this process.

**Late Day Policy:** There will be no penalty for turning in any of the homework assignments up to 24hrs late. However, we cannot guarantee that late assignments will be graded in a timely fashion. Beyond the 24hr grace period, we will not accept late homework.

**Collaboration:** We strongly encourage collaboration (**but NOT copying**) on the homework assignments. You may talk to anyone in the course about how to solve the problems, and you may even compare your solutions. However, you must write up your solutions yourself, and you may not copy them.

---

### 1.1 Conditioning on background evidence (3 pts)

It is often useful to consider the impact of specific events in the context of general background evidence, rather than in the absence of information.

- (a) Denoting such evidence by  $E$ , prove the conditionalized version of the product rule: (1 pts)

$$P(X, Y|E) = P(X|Y, E)P(Y|E).$$

- (b) Also, prove the conditionalized version of Bayes rule: (1 pts)

$$P(X|Y, E) = \frac{P(Y|X, E)P(X|E)}{P(Y|E)}.$$

- (c) Also, prove the conditionalized version of marginalization: (1 pts)

$$P(X|E) = \sum_y P(X, Y=y|E).$$

---

---

## 1.2 Conditional independence (3 pts)

Show that the following three statements about random variables  $X$ ,  $Y$ , and  $E$  are equivalent:

- (1)  $P(X, Y|E) = P(X|E)P(Y|E)$
- (2)  $P(X|Y, E) = P(X|E)$
- (3)  $P(Y|X, E) = P(Y|E)$

In other words, show that (1) implies (2) & (3), that (2) implies (1) & (3), and that (3) implies (1) & (2). You should become fluent with all these ways of expressing that  $X$  is conditionally independent of  $Y$  given  $E$ .

---

## 1.3 Creative writing (3 pts)

This problem does not involve any calculations: simply attach events to the binary random variables  $X$ ,  $Y$ , and  $Z$  that are consistent with the following patterns of commonsense reasoning. You may use different events for the different parts of the problem. Also, please be creative: do not use the same events (e.g., burglaries, earthquakes, alarms) that were considered in lecture.

(a) Cumulative evidence: (1 pts)

$$P(X=1) < P(X=1|Y=1) < P(X=1|Y=1, Z=1)$$

(b) Explaining away: (1 pts)

$$\begin{aligned} P(X=1|Y=1) &> P(X=1), \\ P(X=1|Y=1, Z=1) &< P(X=1|Y=1) \end{aligned}$$

(c) Conditional independence: (1 pts)

$$\begin{aligned} P(X=1, Y=1) &\neq P(X=1)P(Y=1) \\ P(X=1, Y=1|Z=1) &= P(X=1|Z=1)P(Y=1|Z=1) \end{aligned}$$

---

## 1.4 Bayes Rule (6 pts)

Suppose that 1% of competitive cyclists use performance-enhancing drugs and that a particular drug test has a 5% false positive rate and a 10% false negative rate. Let  $D \in \{0, 1\}$  indicate whether a cyclist is doping, and let  $T \in \{0, 1\}$  indicate the outcome of the drug test.

(a) Cyclist A tests negative for drug use. What is the probability that Cyclist A is not using drugs? (3 pts)

(b) Cyclist B tests positive for drug use. What is the probability that Cyclist B is using drugs? (3 pts)

---

---

## 1.5 Kullback-Leibler distance (9 pts)

Often it is useful to measure the difference between two probability distributions over the same random variable. For example, as shorthand let

$$p_i = P(X=i|E), \quad q_i = P(X=i|E')$$

denote the conditional distributions over the random variable  $X$  for different pieces of evidence  $E \neq E'$ . Note that  $\sum_i p_i = \sum_i q_i = 1$ . The Kullback-Leibler (KL) distance between these distributions (also known as the relative entropy) is defined as:

$$\text{KL}(p, q) = \sum_i p_i \log(p_i/q_i).$$

- (a) Consider the natural logarithm (in base  $e$ ). By sketching graphs of  $\log(x)$  and  $x - 1$ , verify the inequality: (2 pts)

$$\log(x) \leq x - 1,$$

with equality if and only if  $x = 1$ . Confirm this result by differentiation of  $\log(x) - (x - 1)$ .

- (b) Use the previous result to prove that  $\text{KL}(p, q) \geq 0$ , with equality if and only if the two distributions  $p_i$  and  $q_i$  are equal. (4 pts)
- (c) Using the inequality in (a), as well as the simple equality  $\log x = 2 \log \sqrt{x}$ , derive the tighter lower bound: (2 pts)

$$\text{KL}(p, q) \geq \sum_i (\sqrt{p_i} - \sqrt{q_i})^2.$$

- (d) Provide a counterexample to show that the KL distance is not a symmetric function of its arguments: (1 pts)

$$\text{KL}(p, q) \neq \text{KL}(q, p).$$

Despite this asymmetry, it is still common to refer to  $\text{KL}(p, q)$  as a measure of distance. Many algorithms for machine learning are based on minimizing KL distances between probability distributions.

---

## 1.6 Mutual information (3 pts)

The mutual information  $I(X, Y)$  between two discrete random variables  $X$  and  $Y$  is defined as

$$I(X, Y) = \sum_x \sum_y P(x, y) \log \left[ \frac{P(x, y)}{P(x)P(y)} \right],$$

where the sum is over all possible values of the random variables  $X$  and  $Y$ . Note how the mutual information is related to the definitions in the previous two problems.

- (a) Show that the mutual information is nonnegative. (*Hint*: use the result of the previous problem.) (2 pts)
- (b) Show that the mutual information  $I(X, Y)$  vanishes if and only if  $X$  and  $Y$  are independent random variables. (Thus,  $I(X, Y)$  provides one quantitative measure of dependence between  $X$  and  $Y$ .) (1 pts)

## 1.7 Hangman (13 pts)

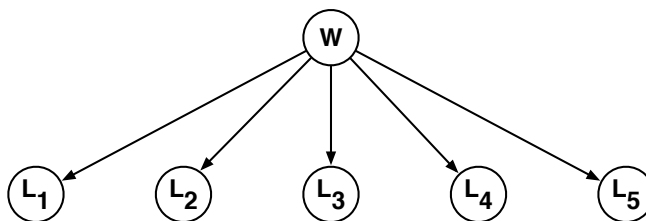
Consider the belief network shown below, where the random variable  $W$  stores a five-letter word and the random variable  $L_i \in \{A, B, \dots, Z\}$  reveals only the word's  $i$ th letter. Also, suppose that these five-letter words are chosen at random from a large corpus of text according to their frequency:

$$P(W=w) = \frac{\text{COUNT}(w)}{\sum_{w'} \text{COUNT}(w')},$$

where  $\text{COUNT}(w)$  denotes the number of times that  $w$  appears in the corpus and where the denominator is a sum over all five-letter words. Note that in this model the conditional probability tables for the random variables  $L_i$  are particularly simple:

$$P(L_i=\ell|W=w) = \begin{cases} 1 & \text{if } \ell \text{ is the } i\text{th letter of } w, \\ 0 & \text{otherwise.} \end{cases}$$

Now imagine a game in which you are asked to guess the word  $w$  one letter at a time. The rules of this game are as follows: after each letter (A through Z) that you guess, you'll be told whether the letter appears in the word and also where it appears. Given the *evidence* that you have at any stage in this game, the critical question is what letter to guess next.



Let's work an example. Suppose that after three guesses—the letters D, I, M—you've learned that the letter I does *not* appear, and that the letters D and M appear as follows:

  M                 D                 M  

Now consider your next guess: call it  $\ell$ . In this game the best guess is the letter  $\ell$  that maximizes

$$P(L_2=\ell \text{ or } L_4=\ell \mid L_1=M, L_3=D, L_5=M, L_2 \notin \{D, I, M\}, L_4 \notin \{D, I, M\}).$$

In other words, pick the letter  $\ell$  that is most likely to appear in the blank (unguessed) spaces of the word. For any letter  $\ell$  we can compute this probability as follows:

$$\begin{aligned} & P(L_2=\ell \text{ or } L_4=\ell \mid L_1=M, L_3=D, L_5=M, L_2 \notin \{D, I, M\}, L_4 \notin \{D, I, M\}) \\ &= \sum_w P(W=w, L_2=\ell \text{ or } L_4=\ell \mid L_1=M, L_3=D, L_5=M, L_2 \notin \{D, I, M\}, L_4 \notin \{D, I, M\}), \quad \boxed{\text{marginalization}} \\ &= \sum_w P(W=w \mid L_1=M, L_3=D, L_5=M, L_2 \notin \{D, I, M\}, L_4 \notin \{D, I, M\}) P(L_2=\ell \text{ or } L_4=\ell \mid W=w) \quad \boxed{\text{product rule \& CI}} \end{aligned}$$

where in the third line we have exploited the conditional independence (**CI**) of the letters  $L_i$  given the word  $W$ . Inside this sum there are two terms, and they are both easy to compute. In particular, the second term is more or less trivial:

$$P(L_2 = \ell \text{ or } L_4 = \ell | W = w) = \begin{cases} 1 & \text{if } \ell \text{ is the second or fourth letter of } w \\ 0 & \text{otherwise.} \end{cases}$$

And the first term we obtain from Bayes rule:

$$\begin{aligned} & P(W = w | L_1 = M, L_3 = D, L_5 = M, L_2 \notin \{D, I, M\}, L_4 \notin \{D, I, M\}) \\ &= \frac{P(L_1 = M, L_3 = D, L_5 = M, L_2 \notin \{D, I, M\}, L_4 \notin \{D, I, M\} | W = w) P(W = w)}{P(L_1 = M, L_3 = D, L_5 = M, L_2 \notin \{D, I, M\}, L_4 \notin \{D, I, M\})} \quad \boxed{\text{Bayes rule}} \end{aligned}$$

In the numerator of Bayes rule are two terms; the left term is equal to zero or one (depending on whether the evidence is compatible with the word  $w$ ), and the right term is the prior probability  $P(W = w)$ , as determined by the empirical word frequencies. The denominator of Bayes rule is given by:

$$\begin{aligned} & P(L_1 = M, L_3 = D, L_5 = M, L_2 \notin \{D, I, M\}, L_4 \notin \{D, I, M\}) \\ &= \sum_w P(W = w, L_1 = M, L_3 = D, L_5 = M, L_2 \notin \{D, I, M\}, L_4 \notin \{D, I, M\}), \quad \boxed{\text{marginalization}} \\ &= \sum_w P(W = w) P(L_1 = M, L_3 = D, L_5 = M, L_2 \notin \{D, I, M\}, L_4 \notin \{D, I, M\} | W = w), \quad \boxed{\text{product rule}} \end{aligned}$$

where again all the right terms inside the sum are equal to zero or one. Note that the denominator merely sums the empirical frequencies of words that are compatible with the observed evidence.

Now let's consider the general problem. Let  $E$  denote the evidence at some intermediate round of the game: in general, some letters will have been guessed correctly and their places revealed in the word, while other letters will have been guessed incorrectly and thus revealed to be absent. There are two essential computations. The first is the *posterior* probability, obtained from Bayes rule:

$$P(W = w | E) = \frac{P(E | W = w) P(W = w)}{\sum_{w'} P(E | W = w') P(W = w')}.$$

The second key computation is the *predictive* probability, based on the evidence, that the letter  $\ell$  appears somewhere in the word:

$$P(L_i = \ell \text{ for some } i \in \{1, 2, 3, 4, 5\} | E) = \sum_w P(L_i = \ell \text{ for some } i \in \{1, 2, 3, 4, 5\} | W = w) P(W = w | E).$$

Note in particular how the first computation feeds into the second. Your assignment in this problem is implement both of these calculations. **You may program in the language of your choice.**

- (a) Download the file `hw1_word_counts_05.txt` that appears with the homework assignment. The file contains a list of 5-letter words (including names and proper nouns) and their counts from a large corpus of Wall Street Journal articles (roughly three million sentences). From the counts in this file compute the prior probability  $P(w) = \text{COUNT}(w) / \sum_{w'} \text{COUNT}(w')$ . **As a sanity check, print out the fifteen most frequent 5-letter words, as well as the fourteen least frequent 5-letter words. Do your results make sense?** (1 pts)

- (b) Consider the following stages of the game. For each of the following, indicate the best next guess—namely, the letter  $\ell$  that is most likely (probable) to be among the missing letters. Also report the probability  $P(L_i = \ell \text{ for some } i \in \{1, 2, 3, 4, 5\} | E)$  for your guess  $\ell$ . Your answers should fill in the last two columns of this table. (Some answers are shown so that you can check your work.) (6 pts)

| correctly guessed | incorrectly guessed | best next guess $\ell$ | $P(L_i = \ell \text{ for some } i \in \{1, 2, 3, 4, 5\}   E)$ |
|-------------------|---------------------|------------------------|---|
| -----             | { }                 |                        |   |
| -----             | {E, A}              |                        |   |
| A----S            | { }                 |                        |   |
| A----S            | {I}                 | E                      | 0.7127  |
| --O--             | {A, E, M, N, T}     |                        |   |
| -----             | {E, O}              | I                      | 0.6366  |
| D--I-             | { }                 |                        |   |
| D--I-             | {A}                 | E                      | 0.7521  |
| -U---             | {A, E, I, O, S}     |                        |   |

- (c) Turn in your source code (including the code execution output). Create a PDF of the entire code (all necessary functions along with the driver code) and attach screenshots of the output produced by your code. You do not need to print a formatted table, but make sure the values you filled in the table (from part b) are present in the output. **Do not forget the source code**. It is worth many points on this assignment. (6 pts)

**Suggestion:** If you are using Python to write the code, you may choose to use Jupyter Notebook and export your notebook, along with the code cells and execution outputs, as a PDF.

Just to be perfectly clear, you are **not** required in this problem to implement a user interface or any general functionality for the game of hangman. You will only be graded on your word lists in (a), the completed table for (b), and your source code in (c).