CS 188 Fall 2021 Introduction to Artificial Intelligence Challenge Q7 HW9

- **Due:** Tuesday 11/16 at 10:59pm.
- **Policy:** Can be solved in groups (acknowledge collaborators) but must be submitted individually.
- **Make sure to show all your work and justify your answers**.
- **Note:** This is a typical exam-level question. On the exam, you would be under time pressure, and have to complete this question on your own. We strongly encourage you to first try this on your own to help you understand where you currently stand. Then feel free to have some discussion about the question with other students and/or staff, before independently writing up your solution.
- Your submission on Gradescope should be a PDF that matches this template. Each page of the PDF should align with the corresponding page of the template (page 1 has name/collaborators, question begins on page 2.). **Do not reorder, split, combine, or add extra pages**. The intention is that you print out the template, write on the page in pen/pencil, and then scan or take pictures of the pages to make your submission. You may also fill out this template digitally (e.g. using a tablet.)

First name	NAMEERA
Last name	FAISAL AKHTAR
SID	3034244256
Collaborators	SARA IMAM

For staff use only:

Q7.	Naive Bayes and Perceptron	/24

Q7. [24 pts] Naive Bayes and Perceptron

Probably the most famous application of Naive Bayes classification is spam filtering. In this problem, we will assume that for each email there is a binary variable Y that takes the values *spam* or *ham* depending on whether the email is spam or not respectively. For each email, assume that each word w in the email follows a probability distribution $P(W = w|Y)$, and the variable W can take on words from some previously determined dictionary (note that the ordering of words in the email does not affect this probability). Punctuation in the email is ignored. For example: for an email with three words $w_1 = I$, $w_2 = love$ and $w_3 = CS188$, the label of the email is given by $\text{argmax}_y P(Y = y|w_1, w_2, w_3) = \text{argmax}_y P(Y = y) \prod_{i=1}^{3} P(W = w_i|Y = y)$.

7.1) (3pts) Assume that we have trained a Naive Bayes classifier on a large dataset of emails using the above model. The following table has the estimated probabilities for some of the words.

W	Berkeley	is	amazing	Oski	rules
$P(W\|Y = \text{spam})$	1/6	1/8	1/4	1/4	1/8
$P(W\|Y = \text{ham})$	1/8	1/3	1/4	1/12	1/12

Table 1: Probability table.

Now assume that we receive a new two-word email that reads:

```
Berkeley rules
```

and we want to label it as spam or ham. Choose all the options for the value of $P(Y = spam)$ for which this new email will be classified as "spam" given the table above. Spam if $\frac{1}{48} \cdot x > \frac{1}{96} \cdot (1-x)$

A. 0.0 $0 > 1/96$ ×

B. 0.2 $1/240 > 1/120$ ×

C. 0.4 $1/120 > 1/160$ ✓

D. 0.6 $1/80 > 1/240$ ✓

E. 0.8 $1/60 > 1/480$ ✓

F. 1.0 $1/48 > 0$ ✓

$P(Y = spam \mid w_1 = Berkeley, w_2 = rules) = P(Y = spam) P(w_1 = Berkeley \mid Y = spam) P(w_2 = rules \mid Y = spam) = x \cdot 1/6 \cdot 1/8 = \frac{1}{48} \cdot x$

$P(Y = ham \mid w_1 = Berkeley, w_2 = rules) = P(Y = ham) P(w_1 = Berkeley \mid Y = ham) P(w_2 = rules \mid Y = ham) = (1-x) \cdot 1/8 \cdot 1/12 = \frac{1}{96} \cdot (1-x)$

7.2) (4pts) Now let's assume that we observe the following three emails with their true label in parentheses:

(Spam): This is a warning that your social security number has been stolen.

(Ham): More cat and dog photos?

(Ham): I love exam prep, regular sections and social events.

What are the estimates of the following probabilities using the Naive Bayes model?

$P(W = \text{warning}\|Y = \text{spam})$	A. 0	B. 1/10	C. 1/5	D. 1/3	E. 2/3	F. None of the options
$P(W = \text{social}\|Y = \text{ham})$	A. 0	B. 1/10	C. 1/5	D. 1/3	E. 2/3	F. None of the options
$P(W = \text{office}\|Y = \text{ham})$	A. 0	B. 1/10	C. 1/5	D. 1/3	E. 2/3	F. None of the options
$P(Y = \text{ham})$	A. 0	B. 1/10	C. 1/5	D. 1/3	E. 2/3	F. None of the options

$P(W = warning \mid Y = spam) = \frac{count(W = warning, Y = spam)}{count(Y = spam)} = \frac{1}{12}$

$P(W = office \mid Y = ham) = \frac{count(W = social, Y = ham)}{count(Y = ham)} = \frac{0}{14} = 0$

$P(W = social \mid Y = ham) = \frac{count(W = social, Y = ham)}{count(Y = ham)} = \frac{1}{14}$

$P(Y = ham) = \frac{count(Y = ham)}{count(Y)} = \frac{2}{3}$

7.3) (3pts) Using the same dataset as in the previous question, we will now utilize the power of Laplace Smoothing with $k = 2$ in our classification task. Assume that the number of different words in our dictionary is V. Write expressions for the following probability estimates. Your expressions can depend on V and you should perform Laplace smoothing on the prior probabilities as well. $P(W = \text{warning} \mid Y = \text{spam}) = \frac{\text{count}(W = \text{warning}, Y = \text{spam})}{\text{count}(Y = \text{spam})} = \frac{1+2}{12+2V} = \frac{3}{12+2V}$

$P(W = \text{warning}|Y = \text{spam}) =$ $\frac{3}{12+2V}$

$P(W = \text{social}|Y = \text{ham}) =$ $\frac{3}{14+2V}$

$P(Y = \text{ham}) =$ $\frac{2}{3}$

$P(W = \text{social} \mid Y = \text{ham}) = \frac{\text{count}(W = \text{social}, Y = \text{ham})}{\text{count}(Y = \text{ham})} = \frac{1+2}{14+2V} = \frac{3}{14+2V}$ $\quad P(Y = \text{ham}) = \frac{\text{count}(Y = \text{ham})}{\text{count}(Y)} = \frac{2}{3}$

7.4) (3pts) We observe the following values for the accuracies on the test dataset for different values of k:

k	0	1	2	10
accuracy	0.65	0.68	0.74	0.6

Table 2: Accuracy and k values.

If for $k = 0$ we achieve an accuracy of 0.8 on the training set, then which of the following options are plausible accuracies for $k = 10$ also on the training set? Select all that apply.

A. 0.1

C. 0.99

B. 0.7

D. None of the above.

This is supposed to prevent overfitting, so accuracy on the training set must go down (cannot be 0.99). However, it wouldn't drop by a drastic amount (cannot be 0.1). Therefore, it must be only 0.7.

7.5) The "Naive" assumption in Naive Bayes is quite strong. In this question we will try to improve the representational capacity of the classifier. More specifically, instead of assuming that the presence of each word depends only on the label via $P(W = w_i|Y)$, we will assume that this probability depends also on the previous word. i.e. $P(W_i = w_i|Y, W_{i-1})$. The model can also be seen in the following figure:

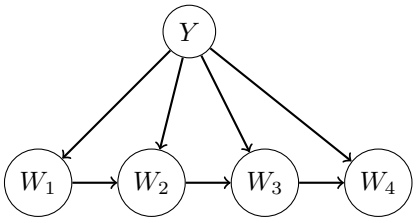

Figure 1: New Bayes Model.

i) (2pts) If our dictionary has V words, then which is the **minimal** number of conditional **word** probabilities that we need to estimate for this model?

A. V

D. $2V^2$

B. $2V$

E. 2^V

C. V^2

F. 2^{2V}

V possible words → V possible previous words ⇒ V^2. Each of V^2 has 2 possible values: spam/ham. ∴ $2V^2$

ii) (2pts) If we use a large dataset with relatively equal number of spam and ham examples to train both this model and the original Naive model, then which of the following effects are expected to be true?

A. The entropy of the posterior $P(Y|W)$ will on average be lower in the new model.

B. The accuracy on the training data will be higher with the new model.

C. The accuracy on the test data will be higher with the new model.

D. None of the above.

The new model should be closer to reality b/c there's a larger sample.

This is true for both training and test data.

We now leave Naive Bayes and move to the perceptron algorithm.

7.6) (2pts) Assume we want to train a multiclass perceptron with three classes A, B and C and with the initial values of the weights being $w_A = [1, 2]$, $w_B = [2, 0]$ and $w_C = [2, -1]$. We train the model on the following dataset:

x_0	x_1	label
1	1	A

Table 3: Training dataset.

$w_A \cdot [1, 1] = [1, 2] \cdot [1, 1] = 1 + 2 = 3$ ← max!

$w_B \cdot [1, 1] = [2, 0] \cdot [1, 1] = 2 + 0 = 2$

$w_C \cdot [1, 1] = [2, -1] \cdot [1, 1] = 2 - 1 = 1$

$\therefore$ we predict A

What are the values of the vectors w_A, w_B and w_C after training using the data sample above only once?

Since the predicted label is equal to the correct label, the weights stay the same.

$w_A = [1, 2]$
$w_B = [2, 0]$
$w_C = [2, -1]$

7.7) (2pts) Now assume that we have a different multiclass perceptron model that we want to train a multiclass perceptron with three classes A, B and C and with the initial values of the weights being $w_A = [2, 4]$, $w_B = [-1, 0]$ and $w_C = [2, -2]$. We train the model on the following dataset:

x_0	x_1	label
-2	1	C

Table 4: Training dataset.

What are the values of the vectors w_A, w_B and w_C after training using the data sample above only once?

$w_A = [2, 4]$
$w_B = [1, -1]$
$w_C = [0, -1]$

$w_A \cdot [1, 1] = [2, 4] \cdot [-2, 1] = -4 + 4 = 0$

$w_B \cdot [1, 1] = [-1, 0] \cdot [-2, 1] = 2 + 0 = 2$

$w_C \cdot [1, 1] = [2, -2] \cdot [-2, 1] = -4 - 2 = -6$

$\therefore$ we predict B

However, the predicted label is A. So, update weights for B, C

$w_B = w_B - \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \end{bmatrix} - \begin{bmatrix} -2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$

$w_C = w_C + \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 2 \\ -2 \end{bmatrix} + \begin{bmatrix} -2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$

7.8) (3pts) Now assume that we have a different multiclass perceptron model that we want to train a multiclass perceptron with three classes A, B and C and with the initial values of the weights being $w_A = [1, 0]$, $w_B = [1, 1]$ and $w_C = [3, 0]$. We train the model by iterating infinitely over the following dataset:

training sample i	x_0	x_1	label
0	1	1	A
1	−1	1	B
2	1	−1	C
3	−1	−1	A

Table 5: Training dataset.

- Start with all weights = 0
- Pick up training examples one by one
- Predict with current weights

$$y = \arg\max_y \; w_y \cdot f(x)$$

- If correct, no change!
- If wrong: lower score of wrong answer, raise score of right answer

$$w_y = w_y - f(x)$$
$$w_{y^*} = w_{y^*} + f(x)$$

Convergence means that the estimated values do not change anymore by passing through the dataset. Which of the following options must be **true** without any additional information?

A. All weight vectors w_A, w_B, w_C converge.

B. Only two of the weight vectors w_A, w_B, w_C converge.

C. Only one of the weight vectors w_A, w_B, w_C converge.

D. None of the weight vectors w_A, w_B, w_C converge.

E. None of the above.

$i=0$: $w_A \cdot [1,1] = [1,0] \cdot [1,1] = 1+0 = 1$
$w_B \cdot [1,1] = [1,1] \cdot [1,1] = 1+1 = 2$
$w_C \cdot [1,1] = [3,0] \cdot [1,1] = 3+0 = 3$
$\therefore$ we predict C → incorrect.

$w_A = w_A + \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$
$w_C = w_C - \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \end{bmatrix}$

$i=1$: $w_A \cdot [-1,1] = [2,1] \cdot [-1,1] = -2+1 = -1$
$w_B \cdot [-1,1] = [1,1] \cdot [-1,1] = -1+1 = 0$
$w_C \cdot [-1,1] = [2,-1] \cdot [-1,1] = -2-1 = -3$
$\therefore$ we predict B → correct!

$i=2$: $w_A \cdot [1,-1] = [2,1] \cdot [1,-1] = 2-1 = 1$
$w_B \cdot [1,-1] = [1,1] \cdot [1,-1] = 1-1 = 0$
$w_C \cdot [1,-1] = [2,-1] \cdot [1,-1] = 2+1 = 3$
$\therefore$ we predict C → correct!

$i=3$: $w_A \cdot [-1,-1] = [2,1] \cdot [-1,-1] = -2-1 = -3$
$w_B \cdot [-1,-1] = [1,1] \cdot [-1,-1] = -1-1 = -2$
$w_C \cdot [-1,-1] = [2,-1] \cdot [-1,-1] = -2+1 = -1$
$\therefore$ we predict C → incorrect!

$w_A = w_A + \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix} + \begin{bmatrix} -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$
$w_C = w_C - \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \end{bmatrix} - \begin{bmatrix} -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \end{bmatrix}$

w_B is the only weight that does not change through these iterations.

w_A and w_C fluctuate.

So only one weight vector converges