Tashfeen, Ahmad
CSCI 6213: Data Science Fundamentals

Homework 3

1. Probability and Conditional Probability

You roll a pair of six-sided dice towards faces a and b. If asked to calculate the probability of rolling a twelve (sum of the two rolls),

$$\text{Probabilty that } (a + b = 12)$$

in notation,

$$\mathbf{P}(a + b = 12)$$

you might answer,

$$\mathbf{P}(a + b = 12) = \mathbf{P}(a = 6) \text{ and } \mathbf{P}(b = 6) = \frac{1}{6^2}$$

What if you are asked the same question, $\mathbf{P}(a + b = 12)$ given that $b < 6$? You might reason that, $b \leq 5$ and $a + 5 = 12 \Rightarrow a = 7$. Which as we know is impossible. Therefore,

$$\mathbf{P}(a + b = 12 \mid b < 6) = 0$$

2. Bayes' Theorem

The English polymath Thomas Bayes figured out that in general the probability that event a occurs given that event b has occurred or in other words, the probability that a will be true given that b is true can be written as the following,

$$\text{Probability of } a \text{ given } b = \mathbf{P}(a \mid b) = \frac{\mathbf{P}(b \mid a)\mathbf{P}(a)}{\mathbf{P}(b)}$$

The above is referred to as the Bayes' Theorem. We can use the Bayes' Theorem as one of our *Data Science* models. Details follow.

3. Naïve Bayes' Classification

Assume that we are trying to learn a model $f(x) \in \{0, 1\}$ for *binary classification*. We might try approximating $f(x)$ as $f'(x)$ by letting $f'(x)$ equal to 0 or 1 depending upon whichever is more probable given x,

$$f'(x) = \begin{cases} 0, & \text{if } \mathbf{P}(y = 1 \mid x) \leq \mathbf{P}(y = 0 \mid x) \\ 1, & \text{if } \mathbf{P}(y = 1 \mid x) > \mathbf{P}(y = 0 \mid x) \end{cases}$$

We can simplify the above as,

$$f'(x_i) = \mathbf{P}(y = 1 \mid x) > \mathbf{P}(y = 0 \mid x)$$

This works for binary classification, but we may use Bayes' Theorem for any multi-class classification. Let $\mathcal{Y} = \{0, 1, 2, \ldots, n\}$ be the set of all possible outcomes, i. e., $f(x) \in \mathcal{Y}$, then we let $f'(x) = y_j$ such that for all $k \neq j$ we know that,

$$\mathbf{P}(y = y_j \mid x) > \mathbf{P}(y = y_k \mid x)$$

or simply,

$$\mathbf{P}(y_j \mid x) > \mathbf{P}(y_k \mid x)$$

In other words, $f'(x) = y_j$ such that $\mathbf{P}(y_j \mid x)$ is maximum (the most probable y_j given x). This is often written as (see question 1),

$$f'(x) = \underset{y \in \mathcal{Y}}{\operatorname{argmax}}\, \mathbf{P}(y \mid x)$$

The crux of the matter is now that how do we compute $\mathbf{P}(y \mid x)$. We use the Bayes' Theorem!

$$\begin{aligned} f'(x) &= \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \mathbf{P}(y \mid x) \\ &= \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \left(\frac{\mathbf{P}(x \mid y)\mathbf{P}(y)}{\mathbf{P}(x)} \right) \\ &= \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \left(\mathbf{P}(x \mid y)\mathbf{P}(y) \right) \end{aligned}$$

$\mathbf{P}(x)$ is disregarded because it's value does not change the $\operatorname{argmax}_{y \in \mathcal{Y}}$ since it stays the same over all y. See questions 2 and 3.

The above model works when x is a scaler, but what if $\mathbf{x} = [x_1, x_2, x_3, \ldots, x_n]^T$? The model becomes,

$$\begin{aligned} f'(x) &= \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \left(\mathbf{P}(\mathbf{x} \mid y)\mathbf{P}(y) \right) \\ f'(x) &= \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \left(\mathbf{P}(y)\mathbf{P}(\mathbf{x} \mid y) \right) \\ &= \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \left(\mathbf{P}(y)\mathbf{P}(x_1, x_2, x_3, \ldots, x_n \mid y) \right) \\ &= \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \left(\mathbf{P}(y)\mathbf{P}(x_1 \text{ and } x_2 \text{ and } x_3 \text{ and } \ldots \text{ and } x_n \mid y) \right) \end{aligned}$$

Calculating $\mathbf{P}(x_1 \text{ and } x_2 \text{ and } x_3 \text{ and } \ldots \text{ and } x_n \mid y)$ may take $\mathcal{O}(2^n)$ unless we cut our losses and assume (naïvely) that x_i's are *statistically* independent. In which case,

$$\begin{aligned} \mathbf{P}(\mathbf{x} \mid y) &= \mathbf{P}(x_1 \text{ and } x_2 \text{ and } x_3 \text{ and } \ldots \text{ and } x_n \mid y) \\ &= \mathbf{P}(x_1 \mid y) \times \mathbf{P}(x_2 \mid y) \times \mathbf{P}(x_3 \mid y) \times \ldots \times \mathbf{P}(x_n \mid y) \\ &= \prod_{i=1}^{n} \mathbf{P}(x_i \mid y) \end{aligned}$$

Hence, our naïve bayes model,

$$f'(x) = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \left(\mathbf{P}(y) \times \prod_{i=1}^{n} \mathbf{P}(x_i \mid y) \right)$$

4. Numerical Stability

As computer scientists, we know that multiplying a lot of small numbers results in underflow[1] errors which may greatly impact the argument maximum function (see question 4). Therefore, here we need to be a little creative with $\ln(x)$.

$$\begin{aligned} f'(x) &= \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \ln \left(\mathbf{P}(y) \prod_{i=1}^{n} \mathbf{P}(x_i \mid y) \right) \\ &= \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \left(\ln \mathbf{P}(y) + \sum_{i=1}^{n} \ln(\mathbf{P}(x_i \mid y)) \right) \qquad \ln(ab) = \ln(a) + \ln(b) \end{aligned}$$

Question 1. What is $\operatorname{argmax}_{y \in \mathcal{Y}} [100_0, 999_1, 24_2, 3.5_3, 4_4, 7_5]$? Simply state the index of the maximum value.

[1] A type of floating point error.

Question 2. Give,

$$\operatorname*{argmax}_{y\in\mathcal{Y}}\left[\left(\frac{100}{4}\right)_0,\left(\frac{999}{4}\right)_1,\left(\frac{24}{4}\right)_2,\left(\frac{3.5}{4}\right)_3,\left(\frac{4}{4}\right)_4,\left(\frac{7}{4}\right)_5\right]$$

Did the answer change from question 1?

Question 3. If we write,

$$\operatorname*{argmax}_{y\in\mathcal{Y}}\left[\left(\frac{100}{c}\right)_0,\left(\frac{999}{c}\right)_1,\left(\frac{24}{c}\right)_2,\left(\frac{3.5}{c}\right)_3,\left(\frac{4}{c}\right)_4,\left(\frac{7}{c}\right)_5\right]$$

for some positive c. Does that change the answer from 1?

Question 4. For this question, we'll use Numpy's function argmax, e. g., `np.argmax`. Numpy's argmax function returns the index of the first value when given two equal values. For example,

$$\texttt{np.argmax([1, 1])} = 0$$

1) What is $(1 - 0.55)/0.45$?
2) Give `np.argmax([1, 1])`.
3) Give `np.argmax([(1-0.55)/0.45, 1])`.
4) Was `np.argmax([1, 1]) = np.argmax([(1-0.55)/0.45, 1])`? Should they be equal? What happened?

Question 5. Consider the following data,

Student ID	Midterm x_0	Hours Studied x_1	Final Grade y
1	B	16	Passed
2	B	13	Passed
3	C	13	Passed
4	A	16	Passed
5	D	9	Failed
6	A	5	Passed
7	F	5	Failed
8	F	0	Failed

TABLE 1. Make-believe student performance data.

Assume that the midterm grade and the hours studied columns are statistically independent. Give exact answers a/b for $a \in \mathbb{N} \ni b$ by counting frequencies in the table 1.

1) What is $\mathbf{P}(y = \text{Passed})$ or $\mathbf{P}(\text{Passed})$.
2) What is $\mathbf{P}(\text{Failed})$.
3) What is the probability that a student got an F in the midterm $\mathbf{P}(x_0 = F)$?
4) What is the probability that a student got an F in the midterm given they failed the class, i. e., $\mathbf{P}(x_0 = \text{F} \mid y = \text{Failed})$.
5) What is $\mathbf{P}(y = \text{Failed} \mid x_0 = \text{F})$

Question 6. Download the cleaned[2] Titanic dataset from this link. The meaning of the columns are given in table 2.

1) Looking at the columns, pick four that you think will be the most important in the task of predicting whether someone survived on Titanic. Which four did you pick? Give their column header names.
2) Once we have downloaded the comma separated file containing the Titanic's passengers data, we still need to pick the relevant columns, encode qualitative data, drop the rows with missing fields and finally split the dataset in training and testing subsets. All of this is done for you in `data.py`. Download and save this python file in the same directory as your `titanic.csv`. What is the baseline accuracy of the made testing subset? Are the testing labels balanced?

[2]Commas removed from the names column so it can be read directly as a CSV.

i^{th} Column	Column Header	Header Description
0	pclass	Passenger Class
1	survival	Survival(0 = No; 1 = Yes)
2	name	Name
3	sex	Sex
4	age	Age
5	sibsp	Number of Siblings/Spouses Aboard
6	parch	Number of Parents/Children Aboard
7	ticket	Ticket Number
8	fare	Passenger Fare (British pound)
9	cabin	Cabin
10	embarked	Port of Embarkation (Cherbourg; Queenstown; Southampton)
11	boat	Lifeboat
12	body	Body Identification Number
13	home.dest	Home/Destination

TABLE 2. Titanic dataset column descriptions

3) Use the data to find out what percentage of men survived on Titanic? What percentage of women?
4) Finish the Python class (linked) implementing the *Naïve Bayes' Classifier*. Name this file `bayes.py`. What accuracy does `python3 bayes.py` print to the console?
5) Use the model you created to predict whether you would have survived Titanic. You'll need to read the `data.py` and table 2 to construct a data-point for yourself. State your findings. Put the code for this task within a file `<lastname>.py`. E. g., for the professor this is `tashfeen.py`.

SUBMISSION INSTRUCTIONS

1) Submit a PDF that answers all the questions that the assignment asks. Circle and/emphasize your final answer wherever possible.
2) Submit your `bayes.py`.
3) Submit your `<lastname>.py`.