Exercise: Mutual Information

(2/2 points)

Consider the following joint probability table for random variables X and Y. We'll compute the mutual information $I(X;Y)$ of random variables X and Y step-by-step.

X \ Y	0	1	2
0	0.10	0.09	0.11
1	0.08	0.07	0.07
2	0.18	0.13	0.17

Mutual information is about comparing the joint distribution of X and Y with what the joint distribution would be if X and Y were actually independent.

In Python (where we won't explicitly store the labels of the rows and columns):

```
import numpy as np
joint_prob_XY = np.array([[0.10, 0.09, 0.11], [0.08, 0.07, 0.07],
[0.18, 0.13, 0.17]])
```

The marginal distributions p_X and p_Y are given by:

```
prob_X = joint_prob_XY.sum(axis=1)
prob_Y = joint_prob_XY.sum(axis=0)
```

Next, we produce what the joint probability table would be if X and Y were actually independent:

```
joint_prob_XY_indep = np.outer(prob_X, prob_Y)
```

At this point, we have the joint distribution of X and Y (denoted $p_{X,Y}$) stored in code as `joint_prob_XY`, and also what the joint distribution would be if X and Y were independent (denoted $p_X p_Y$) stored in code as `joint_prob_XY_indep`. The mutual information of X and Y is precisely given by the KL divergence between $p_{X,Y}$ and $p_X p_Y$:

$$I(X;Y) = D(p_{X,Y} \parallel p_X p_Y) = \sum_x \sum_y p_{X,Y}(x,y) \log_2 \frac{p_{X,Y}(x,y)}{p_X(x) p_Y(y)}.$$

- What is $I(X;Y)$? Provide just the number and don't write “bits" at the end. We suggest that you code a Python function that computes the information divergence between any two distributions, and then you can just plug in `joint_prob_XY` and `joint_prob_XY_indep`.

  (Please be precise with at least **5** decimal places, unless of course the answer doesn't need that many decimal places. You could also put a fraction.)

  0.00226108299607 ✔ **Answer:** 0.0022610829960697087

- Are X and Y independent?

  ○ Yes

  ◉ No ✔

Solution:

- What is $I(X;Y)$? Provide just the number and don't write “bits" at the end.

  Solution: With NumPy, we can code the information divergence as a one-liner:

  ```
  info_divergence = lambda p, q: np.sum(p * np.log2(p / q))
  ```

  If you haven't seen `lambda` before, the above single line is equivalent to:

```
def info_divergence(p, q):
    return np.sum(p * np.log2(p / q))
```

Then, to compute $I(X;Y)$, we do:

```
mutual_info_XY = info_divergence(joint_prob_XY,
joint_prob_XY_indep)
```

Printing out `mutual_info_XY` yields **0.0022610829960697087** bits.

- Are X and Y independent?

  Solution: The answer is **no**. If X and Y were independent, then $p_{X,Y}$ and $p_X p_Y$ would be the same distribution, which means that the KL divergence between them would be 0, which means that the mutual information $I(X;Y)$ would be 0. But as we just computed, $I(X;Y)$ is nonzero!

You have used 2 of 5 submissions