Information Theory Primer

SOLUTIONS - do not distribute!

No solutions for coding part (a)

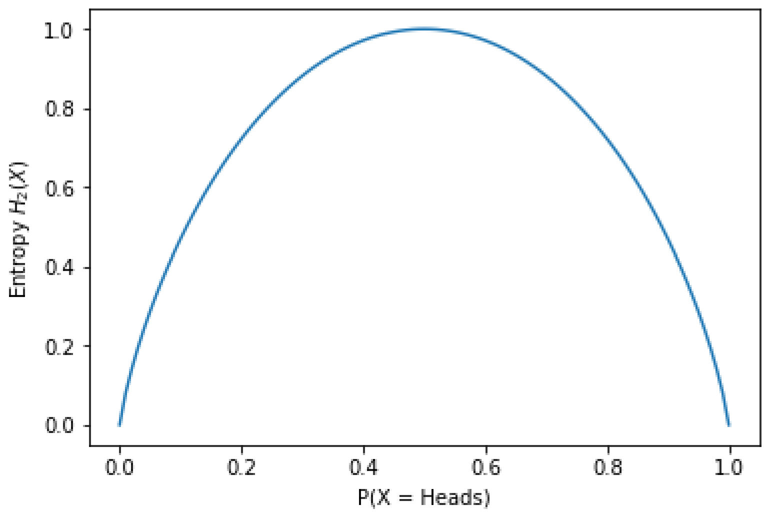
Here we'll explore a few basic concepts from information theory (https://en.wikipedia.org/wiki/Information_theory) that are particularly relevant for this course. Information theory is a fairly broad subject, founded in the 1940s by Claude Shannon (https://en.wikipedia.org/wiki/Claude_Shannon), that gives a mathematical foundation for quantifying the communication of information. Shannon's original paper included, for example, the idea of the bit (https://en.wikipedia.org/wiki/Bit), the minimal unit of information.

In [2]:

```
def XLogX(x):
    """Returns x * log2(x)."""
    return np.nan_to_num(x * np.log2(x))

def BinaryEntropy(p):
    """Compute the entropy of a coin toss with P(heads) = p."""
    #### YOUR CODE HERE ####

    #### END YOUR CODE ####

# Let's try running it for p = 0.  This means the coin always comes up "tail
s".
# We expect that the entropy of this is 0 as there is no uncertainty about the
 outcome.
assert 0.0 == BinaryEntropy(0)

# We expect p = 0.5 to be as uncertain as it gets.  There's no good prior gues
s
# as to which of heads or tails the coin is going to come down on.
# As a result, we expect this to be bigger than p=0 above, but also bigger tha
n any
# other value of p.
assert BinaryEntropy(0.5) > BinaryEntropy(0)
assert BinaryEntropy(0.5) > BinaryEntropy(0.49)
assert BinaryEntropy(0.5) > BinaryEntropy(0.51)

# As it turns out the entropy at p=0.5 is 1.0.
assert 1.0 == BinaryEntropy(0.5)
```

In [3]:
```
# Poking at a couple of individual values is interesting, but we can also simp
ly plot
# entropy for all possibly values of P(H).
# As expected, the curve is maximum at p = 0.5 when the outcome is most uncert
ain
# and decreases to 0 as either heads or tails becomes a certainty.
p_of_heads = np.arange(0, 1.01, 0.01);
plt.plot(p_of_heads, BinaryEntropy(p_of_heads))
plt.xlabel('P(X = Heads)'); plt.ylabel('Entropy $H_2(X)$')
```

Out[3]:
```
Text(0,0.5,'Entropy $H_2(X)$')
```



For a binary variable $x \in \{0, 1\}$ like our coin flip, the maximum entropy happens to be $H(X) = 1.0$. But don't be fooled by this - entropy is only bounded below (by 0), and can be arbitrarily large. We'll see this below.

KL Divergence

It is a measure of how different two probability distributions are. The more Q differs from P, the worse the penalty would be, and thus the higher the KL divergence.

That is,

$$D_{KL}(P \,||\, Q) = CE(P, Q) - H(P)$$

From a machine learning perspective, the KL divergence measures the "avoidable" error - when our model is perfect (i.e. the *distribution* $\hat{P}(y \mid x_i) = P(y \mid x_i)$, the KL divergence goes to zero. In general, the cross-entropy loss - and prediction accuracy - will not be zero, but will be equal to the entropy $H(P)$. This "unavoidable" error is the Bayes error rate (https://en.wikipedia.org/wiki/Bayes_error_rate) for the underlying task.

Exercises (12 points)

A. Pointwise Mutual Information

1. If $P(\text{rainy}, \text{cloudy}) = 0.1$, $P(\text{rainy}) = 0.2$ and $P(\text{cloudy}) = 0.8$, what is $\text{PMI}(\text{rainy}, \text{cloudy})$?
2. Imagine x is some word in a sentence, and y is the next word in the sentence. Imagine $P(\text{washington}) = 0.01$, $P(\text{post}) = 0.01$, and $P(\text{washington}, \text{post}) = 0.002$. What is $\text{PMI}(\text{washington}, \text{post})$? Speculate why this kind of metric might be useful.
3. The average PMI, otherwise known as **Mutual Information** is defined as:
$$\text{MI}(x, y) = \text{E}_{x,y}\left[\text{PMI}(x, y)\right] = \sum_{x,y} p(x, y) \cdot \text{PMI}(x, y)$$
If X and Y are independent, $MI(X, Y) = 0$.
Is the converse true? (If not, give a stronger condition based on PMI that does imply that X and Y are independent.)

A. Your Answers

SOLUTIONS - do not distribute!

1. $\text{PMI}(\text{rainy}, \text{cloudy}) = -0.678$
2. $\text{PMI}(Washington, Post) = 4.32$
3. The converse is not true. You could have:
$$\text{PMI}(x_1, y_1) = -5$$
and
$$\text{PMI}(x_2, y_2) = 5$$
so
$$\text{MI}(x, y) = \sum(PMI) = 0$$
.
However requiring PMI(x,y)=0 for all X, Y guarantees that X and Y are independent.

B. Entropy

1. What if you had 128 messages, sending each with a probability of 1/128? What's the expected number of bits? What is the entropy of this distribution? What about 1024 messages each with probability 1/1024?
2. Consider the following sentences, and a hypothetical distribution over words that could fill in the blank:
   `"How much wood could a _____ chuck if a woodchuck could chuck wood?"`
   `"Hi, my name is _____."`
   Which blank has higher entropy?
3. Consider two normal (Gaussian) distributions: $x \sim \mathcal{N}(0, 1)$ and $y \sim \mathcal{N}(7, 0.5)$. Which variable has higher entropy?

B. Your Answers

SOLUTIONS - do not distribute!

1. 128 choices: #bits = 7, 1024 choices: #bits = 10
2. The sentence "Hi, my name is ." has higher entropy. This is because there are many more equally likely choices for in this case, which means higher entropy.
3. The distribution: $x \sim \mathcal{N}(0, 1)$ has higher entropy, simply because its standard deviation is higher.

C. Cross-Entropy and KL Divergence

For the following questions, imagine you have a classification problem over four labels, $\{0, 1, 2, 3\}$. For some example x_i, the correct label is class 0. That is, our true distribution is $y_i = P(y \mid x_i) = [1, 0, 0, 0]$. Your model generates this probability distribution over the classes: $\hat{y}_i = \hat{P}(y \mid x_i) = [0.8, 0.1, 0.05, 0.05]$.

1. Compute $\mathrm{CrossEntropy}(y, \hat{y})$.
2. Find $D_{KL}(y \,||\, \hat{y})$. Compare it to your answer to (c1).
3. When the label vector is "one-hot" as it is in this case (i.e. only a single category has any probability mass), do you actually need to compute everything? Describe the simplification.
4. What would $\mathrm{CrossEntropy}(y, \hat{y})$ be if your model assigned all probability mass to the correct class (class 0)? (i.e. if $\hat{y}_i = y_i = [1, 0, 0, 0]$)
5. What if the model assigned all probability mass to class 1 instead?
6. What if the model assigned $\frac{1}{3}$ to each of classes 1, 2, and 3, and zero to class 0?

C. Your Answers

SOLUTIONS - do not distribute!

1. $\text{CrossEntropy}(y, \hat{y})$ = 0.322

2. $D_{KL}(y \,||\, \hat{y})$ = 0.322, which is the same as the $\text{CrossEntropy}(y, \hat{y})$ of (c1). This is because $\text{H}(y, \hat{y})$ = 0 in (c1).

3. No, You only need to worry about the p(x) term (where p(x) is not 0) since that's the only significant term in the cross-entropy calculation.

4. $\text{CrossEntropy} = 0$ in the case where the model assigns all probability mass to the correct class.

5. In this case where our model assigns all probability to the wrong class, our $\text{CrossEntropy} = \infty$

6. Here again, $\text{CrossEntropy} = \infty$