CS 784: Computational Linguistics
Lecture 2.2: Probability and Information Theory

Freda Shi

School of Computer Science, University of Waterloo
fhs@uwaterloo.ca

January 9, 2025

[Some slides adapted from Madhur Tulsiani and David McAllester.]

Probability Spaces

Let Ω be a finite set. Let $P : \Omega \to [0, 1]$ be a function such that

$$\sum_{\omega \in \Omega} P(\omega) = 1.$$

We often refer to Ω as a **sample space** or **outcome space** and the function P as a **probability distribution** on this space.

Probability Spaces

Let Ω be a finite set. Let $P : \Omega \to [0, 1]$ be a function such that

$$\sum_{\omega \in \Omega} P(\omega) = 1.$$

We often refer to Ω as a **sample space** or **outcome space** and the function P as a **probability distribution** on this space.

An **event** can be thought of as a subset of all possible outcomes, i.e., any $E \subseteq \Omega$ defines an event, and we define its probability as

$$\mathbb{P}[E] = \sum_{\omega \in E} P(\omega).$$

Random Variable and Expectation (Simplified)

In most cases we encounter, a **random variable** is a function $X : \Omega \to \mathbb{R}$.

Random Variable and Expectation (Simplified)

In most cases we encounter, a **random variable** is a function $X : \Omega \to \mathbb{R}$.

We may define the **expectation** of a random variable X as

$$\mathbb{E}[X] = \sum_{\omega \in \Omega} P(\omega) X(\omega).$$

Random Variable and Expectation (Simplified)

In most cases we encounter, a **random variable** is a function $X : \Omega \to \mathbb{R}$.

We may define the **expectation** of a random variable X as

$$\mathbb{E}[X] = \sum_{\omega \in \Omega} P(\omega) X(\omega).$$

A random variable X is technically **neither random nor a variable**. However, we may informally understand it as a variable whose value is randomly drawn.

Probability Space and Random Variables: Example

Rolling a fair dice gives

$$\text{Sample space } \Omega = \{\text{one}, \text{two}, \text{three}, \text{four}, \text{five}, \text{six}\}$$

$$\text{Random variable } X : \Omega \to \mathbb{R} \text{ such that } X(\text{one}) = 1, \quad X(\text{two}) = 2, \ldots, X(\text{six}) = 6$$

$$\text{Probability } P(\omega) = \frac{1}{6} \qquad \forall \omega \in \Omega$$

$$\text{Event Prob. } \mathbb{P}[\text{even number}] = P(\text{two}) + P(\text{four}) + P(\text{six}) = \frac{1}{2}$$

$$\text{Event Prob. } \mathbb{P}[\text{number} \leq \text{two}] = P(\text{one}) + P(\text{two}) = \frac{1}{3}$$

$$\text{Expectation } \mathbb{E}[X] = \sum_{\omega \in \Omega} P(\omega) X(\omega) = 3.5$$

Probability Space and Random Variables: Example

If you'd like to change $X(\text{one}) = 6$, it also works:

$$\text{Sample space } \Omega = \{\text{one}, \text{two}, \text{three}, \text{four}, \text{five}, \text{six}\}$$

$$\text{Random variable } X : \Omega \to \mathbb{R} \text{ such that } X(\text{one}) = \mathbf{6},$$

$$X(\text{two}) = 2, \ldots, X(\text{six}) = 6$$

$$\text{Probability } P(\omega) = \frac{1}{6} \qquad \forall \omega \in \Omega$$

$$\text{Event Prob. } \mathbb{P}[\text{even number}] = P(\text{two}) + P(\text{four}) + P(\text{six}) = \frac{1}{2}$$

$$\text{Event Prob. } \mathbb{P}[\text{number} \leq \text{two}] = P(\text{one}) + P(\text{two}) = \frac{1}{3}$$

$$\text{Event Prob. } \mathbb{P}[X(\text{number}) \leq 2] = P(\text{two}) = \frac{1}{6}$$

$$\text{Expectation } \mathbb{E}[X] = \sum_{\omega \in \Omega} P(\omega) X(\omega) = \mathbf{4.33}$$

Conditional Probability

Conditioning on an event E is equivalent to restricting the probability space to E. The probability measure is then

$$P_E(\omega) = \begin{cases} \frac{P(\omega)}{\mathbb{P}[E]} & \forall \omega \in E, \\ 0 & \text{otherwise.} \end{cases}$$

Conditional Probability

Conditioning on an event E is equivalent to restricting the probability space to E. The probability measure is then

$$P_E(\omega) = \begin{cases} \frac{P(\omega)}{\mathbb{P}[E]} & \forall \omega \in E, \\ 0 & \text{otherwise.} \end{cases}$$

We define the **conditional probability** of an event F given E as

$$\mathbb{P}[F \mid E] = \sum_{\omega \in F} P_E(\omega) = \sum_{\omega \in E \cap F} \frac{P(\omega)}{\mathbb{P}[E]} = \frac{\mathbb{P}[E \wedge F]}{\mathbb{P}[E]}.$$

Conditional Probability

Conditioning on an event E is equivalent to restricting the probability space to E. The probability measure is then

$$P_E(\omega) = \begin{cases} \frac{P(\omega)}{\mathbb{P}[E]} & \forall \omega \in E, \\ 0 & \text{otherwise.} \end{cases}$$

We define the **conditional probability** of an event F given E as

$$\mathbb{P}[F \mid E] = \sum_{\omega \in F} P_E(\omega) = \sum_{\omega \in E \cap F} \frac{P(\omega)}{\mathbb{P}[E]} = \frac{\mathbb{P}[E \wedge F]}{\mathbb{P}[E]}.$$

We can calculate the conditional expectation similarly:

$$\mathbb{E}[X \mid E] = \sum_{\omega \in E} P_E(\omega) X(\omega).$$

Joint Probability

The **joint probability** of two events E and F is defined as

$$\mathbb{P}[E \wedge F] = \sum_{\omega \in E \cap F} P(\omega).$$

Joint Probability

The **joint probability** of two events E and F is defined as

$$\mathbb{P}[E \wedge F] = \sum_{\omega \in E \cap F} P(\omega).$$

From the previous slide, we know that

$$\begin{aligned} \mathbb{P}[E \wedge F] &= \mathbb{P}[F \mid E]\mathbb{P}[E] \\ &= \mathbb{P}[E \mid F]\mathbb{P}[F]. \end{aligned}$$

Independence

Two non-zero probability events E and F are **independent** if $\mathbb{P}[E \mid F] = \mathbb{P}[E]$ (or $\mathbb{P}[F \mid E] = \mathbb{P}[F]$).

Independence

Two non-zero probability events E and F are **independent** if $\mathbb{P}[E \mid F] = \mathbb{P}[E]$ (or $\mathbb{P}[F \mid E] = \mathbb{P}[F]$).

Two random variables X and Y defined on the same finite probability space are independent if

$$\mathbb{P}[X = x \mid Y = y] = \mathbb{P}[X = x]$$

for all non-zero probability events $\{X = x\} := \{\omega : X(\omega) = x\}$ and $\{Y = y\} := \{\omega : Y(\omega) = y\}$.

Common Notations on Random Variables

In literature, $P(X = x)$ usually denotes “the probability that the random variable X takes on the value x”—this is, actually, $\mathbb{P}(\{\omega : X(\omega) = x\})$.

$\{\omega : X(\omega) = x\}$ is an event, with event probability applicable.

Common Notations on Random Variables

In literature, $P(X = x)$ usually denotes "the probability that the random variable X takes on the value x"—this is, actually, $\mathbb{P}(\{\omega : X(\omega) = x\})$.

$\{\omega : X(\omega) = x\}$ is an event, with event probability applicable.

- **Conditional probability** $P(X = x \mid Y = y) = P(x \mid y)$.
- **Joint probability** $P(X = x, Y = y) = P(x, y)$.
- **Marginal probability** $P(x) = \sum_y P(x, y)$, $P(y) = \sum_x P(x, y)$.
  X and Y are independent iff. $P(x, y) = P(x)P(y)$.
- **Expectation** $\mathbb{E}[X] = \mathbb{E}_{x\sim P}[x] = \sum_x x \cdot P(X = x) = \sum_x xP(x)$.

Common Notations on Random Variables

In literature, $P(X = x)$ usually denotes "the probability that the random variable X takes on the value x"—this is, actually, $\mathbb{P}(\{\omega : X(\omega) = x\})$.

$\{\omega : X(\omega) = x\}$ is an event, with event probability applicable.

- **Conditional probability** $P(X = x \mid Y = y) = P(x \mid y)$.
- **Joint probability** $P(X = x, Y = y) = P(x, y)$.
- **Marginal probability** $P(x) = \sum_y P(x, y)$, $P(y) = \sum_x P(x, y)$.
  X and Y are independent iff. $P(x, y) = P(x)P(y)$.
- **Expectation** $\mathbb{E}[X] = \mathbb{E}_{x\sim P}[x] = \sum_x x \cdot P(X = x) = \sum_x xP(x)$.

Most cases we see in this class will be **discrete random variables**, with the possible values from a finite set.

Common Notations on Random Variables

In literature, $P(X = x)$ usually denotes "the probability that the random variable X takes on the value x"—this is, actually, $\mathbb{P}(\{\omega : X(\omega) = x\})$.

$\{\omega : X(\omega) = x\}$ is an event, with event probability applicable.

- **Conditional probability** $P(X = x \mid Y = y) = P(x \mid y)$.
- **Joint probability** $P(X = x, Y = y) = P(x, y)$.
- **Marginal probability** $P(x) = \sum_y P(x, y)$, $P(y) = \sum_x P(x, y)$.
  X and Y are independent iff. $P(x, y) = P(x)P(y)$.
- **Expectation** $\mathbb{E}[X] = \mathbb{E}_{x\sim P}[x] = \sum_x x \cdot P(X = x) = \sum_x xP(x)$.

Most cases we see in this class will be **discrete random variables**, with the possible values from a finite set.

In what follows, we will use the (intuitive) notations on this slide.

Why Information Theory?

Information theory arises in many places and many forms in computational linguistics and deep learning.

Information-theoretic concepts gives us a formal way to reason about the amount of information in data, and rationalize many linguistic phenomena.

Why Information Theory?

Information theory arises in many places and many forms in computational linguistics and deep learning.

Information-theoretic concepts gives us a formal way to reason about the amount of information in data, and rationalize many linguistic phenomena.

Caveat, important: Information-theoretic explanations make sense when they are supported by empirical evidence; it is never the case that information theory is a universal explanation for everything.

Why Information Theory?

Information theory arises in many places and many forms in computational linguistics and deep learning.

Information-theoretic concepts gives us a formal way to reason about the amount of information in data, and rationalize many linguistic phenomena.

Caveat, important: Information-theoretic explanations make sense when they are supported by empirical evidence; it is never the case that information theory is a universal explanation for everything.

Many model training objectives are derived from information theory.

Entropy

The entropy of a (discrete) random variable X with probability distribution $P(X)$ is defined as

$$H(X) = -\sum_x P(x) \log P(x).$$

Entropy

The entropy of a (discrete) random variable X with probability distribution $P(X)$ is defined as

$$H(X) = -\sum_x P(x) \log P(x).$$

It's obvious that $H(X) \geq 0$:

$$P(x) \leq 1 \Rightarrow -\log P(x) \geq 0.$$

Entropy

The entropy of a (discrete) random variable X with probability distribution $P(X)$ is defined as

$$H(X) = -\sum_x P(x) \log P(x).$$

It's obvious that $H(X) \geq 0$:

$$P(x) \leq 1 \Rightarrow -\log P(x) \geq 0.$$

$H(X)$ is measured in bits if the base of the logarithm is 2.

It can be measured in nats if the base of the logarithm is e.

Shannon's Source Coding Theorem

Why is $-\log_2 P(x)$ a number of bits?

Shannon's Source Coding Theorem

Why is $-\log_2 P(x)$ a number of bits?

A prefix-free code for $\mathcal{X}$ assigns a bit (0/1) string $c(x)$ to each $x \in \mathcal{X}$ such that no $c(x)$ is a prefix of another $c(x')$.

- This ensures that the code, i.e., concatenation of arbitrarily many $c(x)$, is uniquely decodable.

Shannon's Source Coding Theorem

Why is $-\log_2 P(x)$ a number of bits?

A prefix-free code for $\mathcal{X}$ assigns a bit (0/1) string $c(x)$ to each $x \in \mathcal{X}$ such that no $c(x)$ is a prefix of another $c(x')$.

- This ensures that the code, i.e., concatenation of arbitrarily many $c(x)$, is uniquely decodable.

We consider the expected per-element code length under the distribution P, i.e., $\mathbb{E}_{x \sim P}[|c(x)|]$.

Shannon's Source Coding Theorem

Why is $-\log_2 P(x)$ a number of bits?

A prefix-free code for $\mathcal{X}$ assigns a bit (0/1) string $c(x)$ to each $x \in \mathcal{X}$ such that no $c(x)$ is a prefix of another $c(x')$.

- This ensures that the code, i.e., concatenation of arbitrarily many $c(x)$, is uniquely decodable.

We consider the expected per-element code length under the distribution P, i.e., $\mathbb{E}_{x \sim P}[|c(x)|]$.

Theorem 1: For any c, we have $\mathbb{E}_{x \sim P}[|c(x)|] \geq H_2(X)$.

- See [this url] for a proof by Michael Langer.

Shannon's Source Coding Theorem

Why is $-\log_2 P(x)$ a number of bits?

A prefix-free code for $\mathcal{X}$ assigns a bit (0/1) string $c(x)$ to each $x \in \mathcal{X}$ such that no $c(x)$ is a prefix of another $c(x')$.

- This ensures that the code, i.e., concatenation of arbitrarily many $c(x)$, is uniquely decodable.

We consider the expected per-element code length under the distribution P, i.e., $\mathbb{E}_{x\sim P}[|c(x)|]$.

Theorem 1: For any c, we have $\mathbb{E}_{x\sim P}[|c(x)|] \geq H_2(X)$.

- See [this url] for a proof by Michael Langer.

Theorem 2: There exists a prefix-free code c such that $\mathbb{E}_{x\sim P}[|c(x)|] \leq H_2(X) + 1$, by assigning each x a bit string of length $\lceil -\log_2 P(x) \rceil$.

Intuitive Example for the Source Coding Theorem

We have a random variable X that takes values from $\{a, b, c, d\}$ with probabilities $\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\}$.

There are two prefix-free codes c_1 and c_2:

Intuitive Example for the Source Coding Theorem

We have a random variable X that takes values from $\{a, b, c, d\}$ with probabilities $\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\}$.

There are two prefix-free codes c_1 and c_2:

- $c_1(a) = 00$
- $c_1(b) = 01$
- $c_1(c) = 10$
- $c_1(d) = 11$

Intuitive Example for the Source Coding Theorem

We have a random variable X that takes values from $\{a, b, c, d\}$ with probabilities $\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\}$.

There are two prefix-free codes c_1 and c_2:

- $c_1(a) = 00$
- $c_1(b) = 01$
- $c_1(c) = 10$
- $c_1(d) = 11$

- $c_2(a) = 0$
- $c_2(b) = 10$
- $c_2(c) = 110$
- $c_2(d) = 111$

Intuitive Example for the Source Coding Theorem

We have a random variable X that takes values from $\{a, b, c, d\}$ with probabilities $\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\}$.

There are two prefix-free codes c_1 and c_2:

- $c_1(a) = 00$
- $c_1(b) = 01$
- $c_1(c) = 10$
- $c_1(d) = 11$

- $c_2(a) = 0$
- $c_2(b) = 10$
- $c_2(c) = 110$
- $c_2(d) = 111$

Expected encoding length:

$$2 \times \frac{1}{2} + 2 \times \frac{1}{4} + \\ 2 \times \frac{1}{8} + 2 \times \frac{1}{8} = 2$$

Intuitive Example for the Source Coding Theorem

We have a random variable X that takes values from $\{a, b, c, d\}$ with probabilities $\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\}$.

There are two prefix-free codes c_1 and c_2:

- $c_1(a) = 00$
- $c_1(b) = 01$
- $c_1(c) = 10$
- $c_1(d) = 11$

Expected encoding length:

$$2 \times \frac{1}{2} + 2 \times \frac{1}{4} + 2 \times \frac{1}{8} + 2 \times \frac{1}{8} = 2$$

- $c_2(a) = 0$
- $c_2(b) = 10$
- $c_2(c) = 110$
- $c_2(d) = 111$

Expected encoding length:

$$1 \times \frac{1}{2} + 2 \times \frac{1}{4} + 3 \times \frac{1}{8} + 3 \times \frac{1}{8} = 1.75$$

Joint Entropy

The joint entropy of two random variables X and Y is defined as

$$H(X, Y) = -\sum_{x,y} P(x, y) \log P(x, y).$$

Joint Entropy

The joint entropy of two random variables X and Y is defined as

$$H(X, Y) = -\sum_{x,y} P(x, y) \log P(x, y).$$

The joint entropy is a measure of the uncertainty in the joint distribution of X and Y.

Conditional Entropy

The **conditional entropy** of Y given X is defined as

$$
\begin{aligned}
H(Y \mid X) &= \sum_x P(x) H(Y \mid X = x) \\
&= -\sum_{x,y} P(x) P(y \mid x) \log P(y \mid x) \\
&= -\sum_{x,y} P(x, y) \log P(y \mid x).
\end{aligned}
$$

Conditional Entropy

The **conditional entropy** of Y given X is defined as

$$\begin{aligned} H(Y \mid X) &= \sum_x P(x) H(Y \mid X = x) \\ &= -\sum_{x,y} P(x) P(y \mid x) \log P(y \mid x) \\ &= -\sum_{x,y} P(x, y) \log P(y \mid x). \end{aligned}$$

The conditional entropy measures the uncertainty in Y when X is known.

Conditional Entropy

The **conditional entropy** of Y given X is defined as

$$\begin{aligned} H(Y \mid X) &= \sum_x P(x) H(Y \mid X = x) \\ &= -\sum_{x,y} P(x) P(y \mid x) \log P(y \mid x) \\ &= -\sum_{x,y} P(x, y) \log P(y \mid x). \end{aligned}$$

The conditional entropy measures the uncertainty in Y when X is known.

Exercise: show that

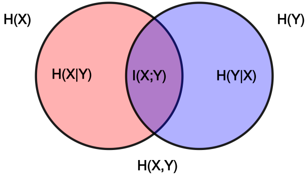
$$H(X, Y) = H(X) + H(Y \mid X) = H(Y) + H(X \mid Y).$$

Hint: expand everything.

Mutual Information

The **mutual information** between two random variables X and Y is defined as

$$\begin{aligned} I(X;Y) &= H(X) + H(Y) - H(X,Y) \\ &= H(X) - H(X \mid Y) \\ &= H(Y) - H(Y \mid X). \end{aligned}$$

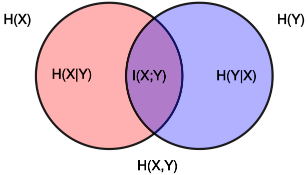


[Figure from Wikipedia]

Mutual Information

The **mutual information** between two random variables X and Y is defined as

$$
\begin{aligned}
I(X; Y) &= H(X) + H(Y) - H(X, Y) \\
&= H(X) - H(X \mid Y) \\
&= H(Y) - H(Y \mid X).
\end{aligned}
$$



[Figure from Wikipedia]

It measures the amount of information that X and Y share: how much knowing one variable reduces uncertainty about the other.

Entropy and Cross Entropy

The entropy $H(X)$ of a random variable is the optimal (minimal) expected number of bits needed to encode a sample $x \sim P(x)$.

Can be also viewed as the **entropy of the distribution** P, $H(P)$.

Entropy and Cross Entropy

The entropy $H(X)$ of a random variable is the optimal (minimal) expected number of bits needed to encode a sample $x \sim P(x)$.

Can be also viewed as the **entropy of the distribution** P, $H(P)$.

Let P and Q be two probability distributions over the same set. The cross entropy of P and Q is defined as

$$H(P, Q) = \mathbb{E}_{x \sim P}[-\log Q(x)]$$

Entropy and Cross Entropy

The entropy $H(X)$ of a random variable is the optimal (minimal) expected number of bits needed to encode a sample $x \sim P(x)$.

Can be also viewed as the **entropy of the distribution** P, $H(P)$.

Let P and Q be two probability distributions over the same set. The cross entropy of P and Q is defined as

$$H(P, Q) = \mathbb{E}_{x \sim P}[-\log Q(x)]$$

Plain language: the expected number of bits per sample is $H(P, Q)$, if we use the optimal code for Q to encode samples from P.

We will show $H(P, Q) \geq H(P)$.

Entropy and Cross Entropy

The entropy $H(X)$ of a random variable is the optimal (minimal) expected number of bits needed to encode a sample $x \sim P(x)$.

Can be also viewed as the **entropy of the distribution** P, $H(P)$.

Let P and Q be two probability distributions over the same set. The cross entropy of P and Q is defined as

$$H(P, Q) = \mathbb{E}_{x \sim P}[-\log Q(x)]$$

Plain language: the expected number of bits per sample is $H(P, Q)$, if we use the optimal code for Q to encode samples from P.

We will show $H(P, Q) \geq H(P)$.

Not to be confused with the (one-distribution) joint entropy $H(X, Y)$!

The Kullback–Leibler Divergence

The Kullback–Leibler (KL) divergence of P from Q is defined as

$$D_{KL}(P \parallel Q) = \mathbb{E}_{x \sim P}\left[\log \frac{P(x)}{Q(x)}\right].$$

The Kullback–Leibler Divergence

The Kullback–Leibler (KL) divergence of P from Q is defined as

$$D_{KL}(P \parallel Q) = \mathbb{E}_{x \sim P}\left[\log \frac{P(x)}{Q(x)}\right].$$

Now we will prove that $D_{KL}(P \parallel Q) \geq 0$ for any P and Q.

Proof of $D_{KL}(P \parallel Q) \geq 0$: Jensen's Inequality

Definition (Convex Function)

A function $f: \mathbb{R} \to \mathbb{R}$ is convex if for all $x, y \in \mathbb{R}$ and $\lambda \in [0, 1]$,

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y).$$

Informally, a convex function is a function that upcurves everywhere.

Proof of $D_{KL}(P \parallel Q) \geq 0$: Jensen's Inequality

Definition (Convex Function)

A function $f: \mathbb{R} \to \mathbb{R}$ is convex if for all $x, y \in \mathbb{R}$ and $\lambda \in [0, 1]$,

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y).$$

Informally, a convex function is a function that upcurves everywhere.

A convex function f satisfies Jensen's inequality:

$$f(\mathbb{E}[X]) \leq \mathbb{E}[f(X)]$$

Can be proved by induction on the number of samples.

Proof of $D_{KL}(P \parallel Q) \geq 0$: Jensen's Inequality

Definition (Convex Function)

A function $f : \mathbb{R} \to \mathbb{R}$ is convex if for all $x, y \in \mathbb{R}$ and $\lambda \in [0, 1]$,

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y).$$

Informally, a convex function is a function that upcurves everywhere.

A convex function f satisfies Jensen's inequality:

$$f(\mathbb{E}[X]) \leq \mathbb{E}[f(X)]$$

Can be proved by induction on the number of samples.

Example: $-\log(x)$ is a convex function.

Proving $D_{KL}(P \parallel Q) \geq 0$

$$
\begin{aligned}
D_{KL}(P \parallel Q) = \mathbb{E}_{x \sim P}\left[\log \frac{P(x)}{Q(x)}\right] &= \mathbb{E}_{x \sim P}\left[-\log \frac{Q(x)}{P(x)}\right] \\
&\geq -\log \mathbb{E}_{x \sim P}\left[\frac{Q(x)}{P(x)}\right] \\
&= -\log\left[\sum_x P(x) \frac{Q(x)}{P(x)}\right] \\
&= -\log\left[\sum_x Q(x)\right] \\
&= -\log 1 = 0
\end{aligned}
$$

Proving $D_{KL}(P \| Q) \geq 0$

$$
\begin{aligned}
D_{KL}(P \| Q) = \mathbb{E}_{x\sim P}\left[\log \frac{P(x)}{Q(x)}\right] &= \mathbb{E}_{x\sim P}\left[-\log \frac{Q(x)}{P(x)}\right] \\
&\geq -\log \mathbb{E}_{x\sim P}\left[\frac{Q(x)}{P(x)}\right] \\
&= -\log\left[\sum_x P(x)\frac{Q(x)}{P(x)}\right] \\
&= -\log\left[\sum_x Q(x)\right] \\
&= -\log 1 = 0
\end{aligned}
$$

Exercise: show that $I(X;Y) \geq 0$.

Hint: Express $I(X;Y)$ in the form of D_{KL}.

Cross Entropy and Kullback–Leibler Divergence

$$
\begin{aligned}
D_{KL}(P \parallel Q) &= \mathbb{E}_{x \sim P}\left[\log \frac{P(x)}{Q(x)}\right] \\
&= \mathbb{E}_{x \sim P}\left[\log P(x) - \log Q(x)\right] \\
&= H(P, Q) - H(P)
\end{aligned}
$$

Cross Entropy and Kullback–Leibler Divergence

$$\begin{aligned} D_{KL}(P \parallel Q) &= \mathbb{E}_{x\sim P}\left[\log \frac{P(x)}{Q(x)}\right] \\ &= \mathbb{E}_{x\sim P}\left[\log P(x) - \log Q(x)\right] \\ &= H(P, Q) - H(P) \end{aligned}$$

About $H(P, Q)$: $D_{KL}(P \parallel Q) \geq 0 \Rightarrow H(P, Q) \geq H(P)$.

Cross Entropy and Kullback–Leibler Divergence

$$
\begin{aligned}
D_{KL}(P \parallel Q) &= \mathbb{E}_{x \sim P}\left[\log \frac{P(x)}{Q(x)}\right] \\
&= \mathbb{E}_{x \sim P}\left[\log P(x) - \log Q(x)\right] \\
&= H(P, Q) - H(P)
\end{aligned}
$$

About $H(P, Q)$: $D_{KL}(P \parallel Q) \geq 0 \Rightarrow H(P, Q) \geq H(P)$.
Suboptimal code for P will result in a longer expected code length.

Cross Entropy and Kullback–Leibler Divergence

$$
\begin{aligned}
D_{KL}(P \parallel Q) &= \mathbb{E}_{x\sim P}\left[\log \frac{P(x)}{Q(x)}\right] \\
&= \mathbb{E}_{x\sim P}\left[\log P(x) - \log Q(x)\right] \\
&= H(P, Q) - H(P)
\end{aligned}
$$

About $H(P, Q)$: $D_{KL}(P \parallel Q) \geq 0 \Rightarrow H(P, Q) \geq H(P)$.
Suboptimal code for P will result in a longer expected code length.

About $D_{KL}(P \parallel Q)$: it measures the inefficiency of using code for Q to encode samples from P, i.e., how many extra bits per sample are expected to be used.

Next

Statistical Methods