Total

LOGARITHMS

Defined implicitly by the equation

$$x = b^{\log_b(x)}$$

- That is, the inverse of the exponential function.

Many useful properties:

$$\log_b(xy) = \log_b(x) + \log_b(y)$$
$$\log_b(x^n) = n \log_b(x)$$
$$\log_b(x) = \frac{\log_a(x)}{\log_a(b)}$$
$$\log_b(1) = 0$$

POTENTIAL PITFALL

$\log_b(x + y)$ is in general *not* simplifiable.

- Unless otherwise stated:
  - log means $\log_2$
  - ln means $\log_e$

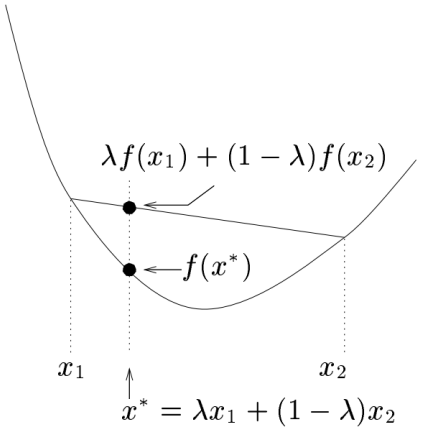
EXAMPLES

- $\log(8) = 3$, since $2^3 = 8$.
- $\log(1/8) = -3$, since $\log(1/8) = \log(8^{-1}) = -\log(8)$.
- $\ln(2) \approx 0.69$ – you will be seeing this a lot!

Concavity and convexity

A function is **convex** iff a line between any two points is above the function.

$$f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2)$$



Similarly, a **concave** function is one where the inequality is reversed.

Concave or convex?

The terms convex and concave can be a bit confusing. We will specify concave-up / convex-down ($\bigcup$) vs concave-down / convex-up ($\bigcap$).

Concavity and convexity

How do you prove a function is convex?

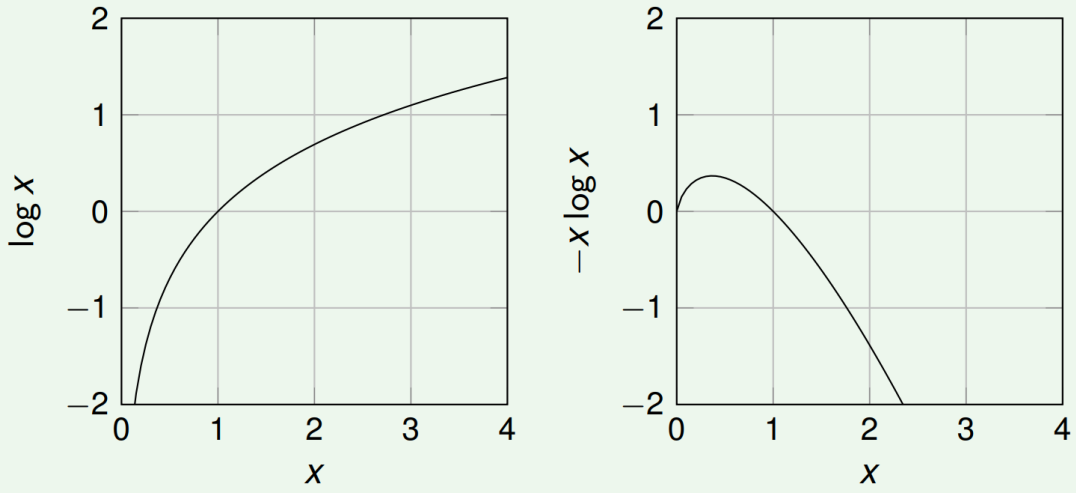
- **Option 1**: show the second derivative is always positive.
- **Option 2**: check if the function has *convexity-preserving* operations.
  - Affine mapping: if $f(x)$ is convex, then $f(Ax + b)$ is also convex.
  - Non-negative weighted sum: if $f_i(x)$ are convex and $w_i \geq 0$, then $\sum_i w_i f_i(x)$ is convex.
  - Many more.

Recommended textbook

Convex Optimization, by Boyd & Vandenberghe.

EXAMPLE: CONCAVITY OF THE LOGARITHM

The log and $-x \log x$ are concave-down ($\bigcap$) functions.



PROBABILITY DISTRIBUTIONS

THEOREM: JENSEN'S INEQUALITY

Given a convex-down ($\bigcup$) function $f(x)$ and a PDF $p(x)$.

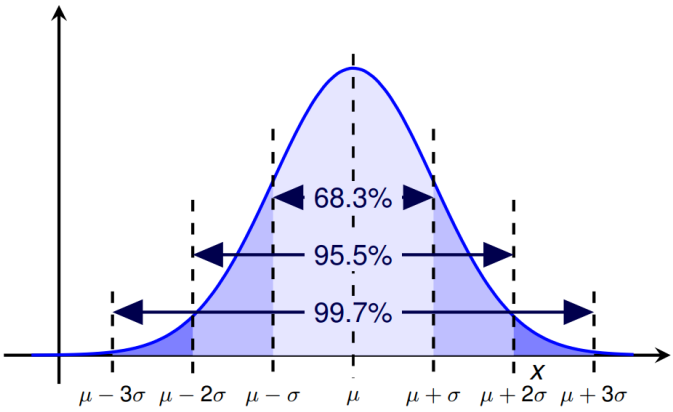
$$\mathbb{E}\left[f(x)\right] \geq f(\mathbb{E}\left[x\right])$$

The normal distribution

Definition: Normal (or Gaussian) distribution

The PDF for the d-dimensional normal distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, denoted by $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, is:

$$p(x; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-d/2} |\boldsymbol{\Sigma}|^{-1/2} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^\intercal \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right)$$



Entropy

Important definitions

Consider a RV X taking values in an alphabet $\mathcal{X}$.

Definition: Information content

The *information content* (a.k.a. surprise or surprisal) of a realisation x is

$$h(x) := \log \frac{1}{p(x)}$$

We will sometimes use the equivalent notation $h(p) := -\log p$.

Definition: Entropy

The *entropy* of a RV X is given by

$$H(X) := \sum_{x \in \mathcal{X}} p(x) \log \frac{1}{p(x)}$$

Note that entropy is the expected value of surprisal: $H(X) = \mathbb{E}[h(X)] = \sum_x p(x)h(x)$.

Entropy of a biased coin

- Mathematically, a biased coin flip X is a Bernoulli distribution with parameter p:
$$X \sim \mathcal{B}(p)$$
- The units of $h(x)$ depend on the base of the log:
  - $\log_2$ yields units of bits.
  - $\log_e$ yields units of nats.

The term 'bit'

We use the term bit to denote two things:

- The units of information content, taken with $\log_2$.
- A random variable with alphabet $\{\mathbf{0},\mathbf{1}\}$.

Details will be clear from context.

- Entropy of discrete distributions is non-negative

Entropy is additive for independent random variables.

Definition: Shannon axioms

The surprisal of an event with probability p, $i(p)$ must satisfy:

1. Certain events are unsurprising: $i(1) = 0$.
2. Less probable events are more surprising: $di/dp \leq 0$.
3. Independent events yield the sum of their surprisals: $i(p \cdot q) = i(p) + i(q)$.

The only function satisfying these axioms is the negative logarithm:

$$i(p) = h(p) = -\log_b(p)$$

Conclusions

- For uniform distributions, $h(x)$ corresponds to the length of the binary representation of x (i.e. number of yes/no questions).
- $h(x)$ is a measure of the "intrinsic" information content of x, regardless of how that information is obtained.
- Information is given by probability mass exclusions (Hartley 1928).

The goal of experiments is to reduce uncertainty about something.

• The value of a parameter, the best model for some dataset, etc

Definition: Differential entropy

The *differential entropy* of a continuous RV X with PDF $f(x)$ is given by

$$H(X) := -\int f(x) \log f(x) dx$$

Example: Differential entropy in uniform distributions

Let X be uniform RV in the interval of length a.

$$X \sim \mathcal{U}([0, a]) \qquad \text{i.e.} \qquad p(x) = \frac{1}{a}$$

$$H(X) = -\int_0^a \frac{1}{a} \log \frac{1}{a} dx = -\log \frac{1}{a} \int_0^a \frac{dx}{a} = \log a$$

- **Conclusions**: Differential entropy...
  - can be negative (e.g. for $a < 1$).
  - grows with the volume of the distribution ($2^{H(X)} = a > 0$).

Key takeaway: Entropy

- Entropy quantifies the **average information content** obtained from an observation x.
- Entropy is a **generalised variance** (e.g. how hard is it to predict X).
- In discrete distributions, entropy...
  - is **bounded**, $0 \leq H(X) \leq \log |\mathcal{X}|$.
  - is related to the **number of yes/no questions** needed to determine x.
- In continuous distributions, **differential entropy** is defined, but does not satisfy some properties of discrete entropy (e.g. can be negative).

Definition: Code and code length

Given a RV X with alphabet $\mathcal{X}$ and an alphabet $\mathcal{D}$, a **code** is a mapping

$$C : \mathcal{X} \to \mathcal{D}^* ,$$

where $\mathcal{D}^*$ is the set of all finite-length strings of symbols in $\mathcal{D}$.

The quantity $\ell(x)$ is the **code length** of $C(x)$, and $L = \mathbb{E}[\ell(x)]$ the average code length.

- We've seen that (in uniform distributions), entropy corresponds to the number of yes/no questions needed to guess x.

$35 \implies$ `100011`	$6 \implies$ `000110`	$17 \implies$ `010001`
$0 \implies$ `000000`	$42 \implies$ `101010`	...

- This forms a binary code for X with length

$$L = H(X) = \log |\mathcal{X}|$$

- More generally, $\log |\mathcal{X}|$ is referred to as the raw bit content of X.

ASYMPTOTIC EQUIPARTITION PRINCIPLE

THEOREM: AEP

Let $X_1, ..., X_n$ be i.i.d. RVs with $X_i \sim p(x)$. Then

$$\lim_{n\to\infty} -\frac{1}{n} \log p(x_1, ..., x_n) = H(X)$$

All samples are equally probable!!

Proof:

$$-\frac{1}{n} \log p(x_1, ..., x_n) = -\frac{1}{n} \log \prod p(x_i) = -\frac{1}{n} \sum_i \log p(x_i)$$

$$\overset{\text{wLLN}}{=} -\mathbb{E}\left[\log p(x)\right] = H(X)$$

TYPICAL SET

DEFINITION: TYPICAL SET

The typical set $A_\varepsilon^{(n)}$ for some $\varepsilon > 0$ is the set of strings $x_1, ..., x_n$ with the property

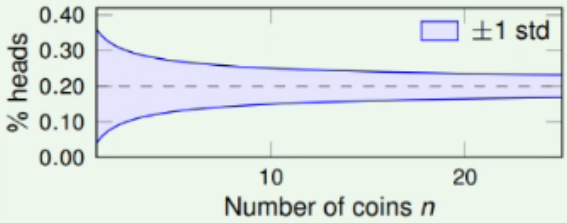
$$2^{-n(H(X)+\varepsilon)} \leq p(x_1, ..., x_n) \leq 2^{-n(H(X)-\varepsilon)}$$

- Informally: events that almost always occur when sampling $X_1, ..., X_n$.
  - In other words, events outside $A_\varepsilon^{(n)}$ are very unlikely.

TYPICAL SET

EXAMPLE: MORE COIN FLIPS

- A group of n people flip a biased coin ($p = 0.2$) at the same time, and write down the number of heads.
- As n grows, almost all the time we will get very close to 20% of coins being heads.



This has two important consequences:

1. Only a small fraction of all possible outcomes happens almost always.
   (There are 2^n possible strings, but only $\binom{n}{np}$ happen most of the time.)
2. These outcomes happen with the same probability.
   (Because probabilities depend only on number of heads.)

38

TYPICAL SET

- **Property 1**: If $x^n \in A_\varepsilon^{(n)}$, then $H(X) - \varepsilon \leq -\frac{1}{n} \log p(x^n) \leq H(X) + \varepsilon$.
  - In words: All samples in $A_\varepsilon^{(n)}$ are uniformly distributed (i.e. all have nearly the same probability $p(x^n) = 2^{-nH(X)}$, which does not depend on x^n).
  - Proof: follows directly from definition of $A_\varepsilon^{(n)}$.
- **Property 2**: $P\left(x^n \in A_\varepsilon^{(n)}\right) > 1 - \varepsilon$.
  - In words: Almost all samples are in the typical set.
  - Proof (sketch): Recall that

$$\frac{1}{n} \log p(x_1, \ldots, x_n) = \frac{1}{n} \sum_i \log p(x_i) \ .$$

By the wLLN, as $n \to \infty$, the sum converges to its mean *with vanishing variance*. Thus, the probability of a sample deviating from the mean more than ε vanishes.

TYPICAL SET

- **Property 3**: $|A_\varepsilon^{(n)}| \leq 2^{n(H(X)+\varepsilon)}$.
  - In words: The number of items in $A_\varepsilon^{(n)}$ is upper-bounded by $2^{n(H(X)+\varepsilon)}$
  - Proof:

$$1 = \sum_{x^n \in \mathcal{X}^n} p(x^n) \geq \sum_{x^n \in A_\varepsilon^{(n)}} p(x^n) \overset{\text{Prop 1}}{\geq} \sum_{x^n \in A_\varepsilon^{(n)}} 2^{-n(H(X)+\varepsilon)} = |A_\varepsilon^{(n)}| 2^{-n(H(X)+\varepsilon)}$$

□

- **Property 4**: $|A_\varepsilon^{(n)}| \geq (1-\varepsilon)2^{n(H(X)-\varepsilon)}$.
  - In words: The number of items in $A_\varepsilon^{(n)}$ is lower-bounded by $2^{n(H(X)-\varepsilon)}$
  - Proof:

$$1 - \varepsilon \overset{\text{Prop 2}}{<} P\left(x^n \in A_\varepsilon^{(n)}\right) \overset{\text{Prop 1}}{\leq} \sum_{x^n \in A_\varepsilon^{(n)}} 2^{-n(H(X)-\varepsilon)} = |A_\varepsilon^{(n)}| 2^{-n(H(X)-\varepsilon)}$$

□

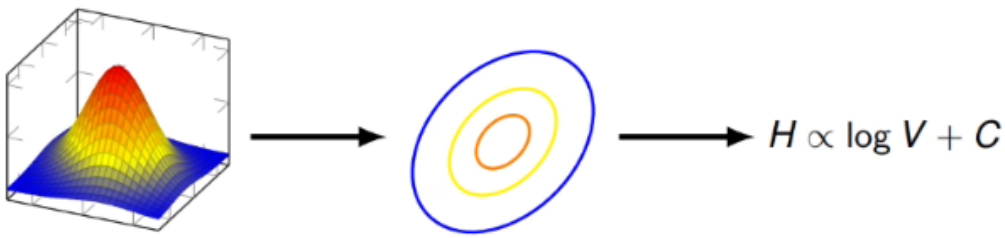
ENTROPY AND VOLUME

The AEP yields another statistical interpretation of entropy as the fraction of total volume actually occupied by X:

- In binary variables ($|\mathcal{X}| = 2$):

$$\frac{\log(\#\text{ events that actually happen})}{\log(\#\text{ events that could happen})} = \frac{\log |A_\varepsilon^{(n)}|}{\log |\mathcal{X}|^n} = \frac{nH(X)}{n} = H(X)$$

- In Gaussian distributions, entropy is equal (up to an additive constant) to the log-volume of the isoprobability contours.



Entropy is the optimal bound of the entropy

THE SOURCE CODING THEOREM

SOURCE CODING THEOREM

Let X^n be i.i.d. $\sim p(x)$. For any $\varepsilon > 0$, for sufficiently large n there exists an invertible code with

$$\mathbb{E}\left[\frac{1}{n}\ell(X^n)\right] \leq H(X) + \varepsilon$$

Conversely, there are no invertible codes with $L < H(X)$.

Symbol Coding

No free lunch theorem

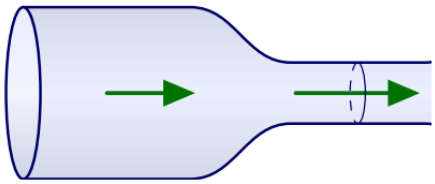
NO FREE LUNCH FOR INFORMATION THEORISTS

- Let $\mathcal{A}^L$ denote the set of all strings of length L.
- Consider a compressor $C : \mathcal{A}^L \to \mathcal{A}^*$.
- Can there be a compressor that *shortens* all strings in $\mathcal{A}^L$?

No.

Either:

- It is not invertible; or
- It lengthens some strings.



6

Lossy and lossness compression

Due to the “no free lunch” theorem, we are left with two choices:

1. **Lossy compression**: the compressor compresses most strings, but maps some strings to the same encoding – leading to a failure and loss of the original information.

2. **Lossless compression**: the compressor shortens most strings, but makes others longer.

Definition: Nonsingular code

A code is *nonsingular* if each symbol maps to a different codeword:

$$x \neq x' \implies C(x) \neq C(x')$$

A nonsingular code is invertible at the level of individual codewords.

Even if the code is invertible, it might be hard to decode

Types of code

Definition: Extended code

Given a code C, its *extension* (or extended code) $C^* : \mathcal{X}^+ \to \mathcal{D}^*$ is the code obtained by concatenating codewords from C:

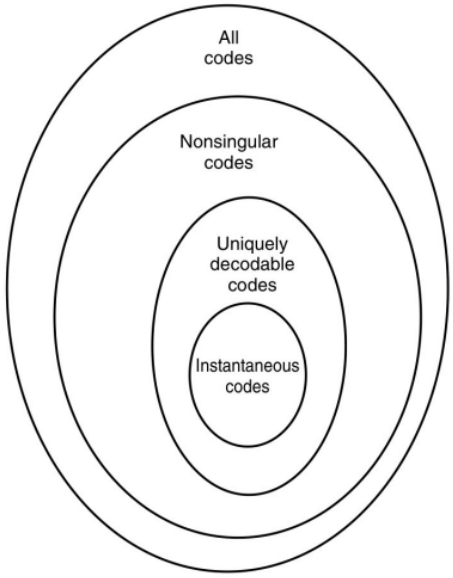
$$C^*(x_1x_2...x_n) = C(x_1)C(x_2)...C(x_n)$$

Definition: Uniquely decodable code

A code C is *uniquely decodable* if its extension is nonsingular.

Definition: Prefix code

A *prefix* (or prefix-free, or instantaneous, or self-punctuating) code is one where no codeword is the start of another codeword.



Example

x	$C(x)$
a	0
b	10
c	110

Let's focus on **prefix codes**.

It also hold for uniquely decodable code

Theorem: Kraft inequality

The codeword lengths ℓ_i of any D-ary prefix code must satisfy

$$\sum_i D^{-\ell_i} \leq 1$$

Conversely, given a set of lengths that satisfy this inequality, there exists a prefix code with those lengths.

Optimal codes

5. Take the second derivative of the Lagrangian:

$$\frac{d^2\mathcal{L}}{d\ell_i^2} = D^{-\ell_i} \ln D \geq 0 \tag{5}$$

   Since $\frac{d^2\mathcal{L}}{d\ell_i^2} \geq 0$ and there is only one solution to $\frac{d\mathcal{L}}{d\ell_i} = 0$, the solution is a global minimum.

6. Given optimal word lengths $\ell_i^* = -\log_D p_i$, average length is:

$$L^* = \sum_i p_i \ell_i^* = -\sum_i p_i \log_D p_i = H_D(X) \tag{6}$$

Entropy is the minimum length of all symbol codes.

The results above establish a duality between PDFs and prefix codes:

- Given a PDF p_i, there is an optimal code C with lengths $\ell_i = -\log_D p_i$.
- Given (Kraft-compatible) lengths ℓ_i, there exists a PDF $p_i \propto D^{-\ell_i}$ it is optimal for.
  - These are sometimes called the *implicit* probabilities of C.

Algorithm: Huffman coding

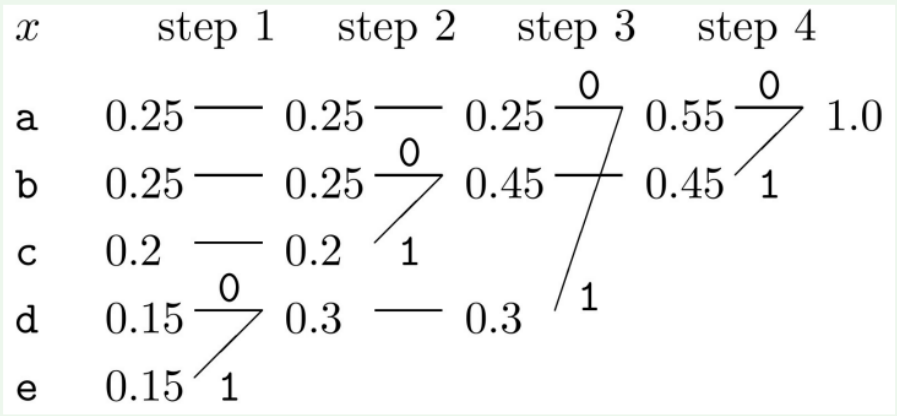
Input: Symbols x_i and associated probabilities p_i.

1. Pick the two least probable symbols.
2. Combine these two symbols into a single symbol to form a tree.
3. If more than one symbol remain, go to step 1.

Theorem: Huffman coding is optimal. If C^* is the Huffman code and C' is any other uniquely decodable code, $L(C^*) \leq L(C')$.

Example

Let $\mathcal{X} = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}\}$ and $p = \{0.25, 0.25, 0.2, 0.15, 0.15\}$.



Assign codewords by traversing the tree from right to left:

$$\mathtt{a} \to \mathtt{00} \qquad \mathtt{b} \to \mathtt{10} \qquad \ldots \qquad \mathtt{e} \to \mathtt{011}$$

KEY TAKEAWAY: SYMBOL CODES

- **Symbol codes** map each individual symbol onto a codeword.
- **Prefix codes** are those where no codeword is the start of another codeword. They can be represented with trees.
- **Kraft inequality**: prefix code $\iff \sum_i D^{-\ell_i} \leq 1$.
- **Source coding theorem for symbol codes**. For any $p(x)$:
  - There exists a code that achieves length close to $H(X)$. (Achievability)
  - There are no codes that achieve length $< H(X)$. (Converse)

DEFINITION: CROSS ENTROPY

The *cross-entropy* of a PDF q relative to a PDF p is defined as:

$$H(p, q) := -\sum_{x \in \mathcal{X}} p(x) \log q(x)$$

It represents the expected length of a q-optimal code used on samples from p.

KL divergence is a convex down function, so it have a localbal unique minimum

Kullback-Leibler divergence

Definition: Kullback-Leibler divergence

The Kullback-Leibler (KL) divergence (or *relative entropy*) of PDFs p and q on $\mathcal{X}$ is defined as

$$D_{\mathrm{KL}}(p \parallel q) \coloneqq \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)}$$

- KL divergence (or KL for short) quantifies the "difference" between two distributions.
- KL can be decomposed in terms of cross-entropy:

$$D_{\mathrm{KL}}(p \parallel q) = \underbrace{H(p, q)}_{\substack{\text{length of } q\text{-optimal} \\ \text{code on } p}} - \underbrace{H(p)}_{\substack{\text{length of } p\text{-optimal} \\ \text{code on } p}}$$

- KL is the extra cost of using a wrong code.

Theorem: Gibbs' inequality

KL divergence is non-negative:

$$D_{\mathrm{KL}}(p \parallel q) \geq 0$$

With equality if and only if $p(x) = q(x) \, \forall x \in \mathcal{X}$.

Extremely important theorem.

KULLBACK-LEIBLER DIVERGENCE

Some interesting properties:

- Entropy is the negative of the KL to the uniform distribution:

$$D_{\mathrm{KL}}(p \parallel u) = \sum_i p_i \log \frac{p_i}{u_i} = -\sum_i p_i \log \frac{1}{p_i} - \sum_i p_i \log \frac{1}{|\mathcal{X}|}$$
$$= -H(P) + \log |\mathcal{X}|$$

- This can be used to prove bounds on entropy:

$$H(P) = \log |\mathcal{X}| - D_{\mathrm{KL}}(p \parallel u) \overset{\text{Gibbs}}{\leq} \log |\mathcal{X}|$$

- KL is asymmetric (and thus not a distance).

$$D_{\mathrm{KL}}(p \parallel q) \neq D_{\mathrm{KL}}(q \parallel p)$$

- KL can be infinite if there exists some x where $q(x) = 0$ and $p(x) > 0$.
  - Intuitively: p has symbols you didn't know existed.

KL DIVERGENCE IN CONTINUOUS DISTRIBUTIONS

To define entropy in continuous RVs, we had to discretise the domain into bins of width Δ. For a RV $X_f \in \mathbb{R}$ with PDF $f(x)$:

$$H(X_f^\Delta) = -\sum_i \Delta f(x_i) \log f(x_i) - \log \Delta$$

By analogy, cross-entropy becomes:

$$H(X_f^\Delta, X_g^\Delta) = -\sum_i \Delta f(x_i) \log g(x_i) - \log \Delta$$

Putting those together:

$$D_{\mathrm{KL}}(X_f^\Delta \parallel X_g^\Delta) = H(X_f^\Delta, X_g^\Delta) - H(X_f^\Delta)$$
$$= -\sum_i \Delta f(x_i) \log g(x_i) - \log \Delta - \left(-\sum_i \Delta f(x_i) \log f(x_i) - \log \Delta \right)$$
$$= \sum_i \Delta f(x_i) \log \frac{f(x_i)}{g(x_i)}$$

- KL is non-negative in continuous RVs because the $\log \Delta$ cancel out.

- Cross-entropy represents the expected code length under a wrong (i.e. suboptimal) code.

- Kullback-Leibler (KL) divergence is a non-negative measure of the difference between two PDFs.
  - Compression interpretation: the "extra cost" of using a wrong code.
  - Geometric interpretation: grows as the two distributions get far.

Definition: Maximum likelihood estimation

Given a dataset $\boldsymbol{x}$ and a family of PDFs $p(\boldsymbol{x}; \theta)$, the *maximum likelihood estimate* of θ is

$$\theta^*_{\text{MLE}} = \operatorname*{argmax}_{\theta} p(\boldsymbol{x}; \theta) \ .$$

- ✓ Symbol codes allow us to compress individual symbols (not blocks).
  - Prefix codes are particularly handy (Kraft inequality, tree representation).
  - Practical algorithm: Huffman coding.

- ✓ KL divergence quantifies the difference between two PDFs.
  - It is always non-negative (Gibbs' inequality).
  - It has both geometric and compression interpretations.

- ✓ Mathematical equivalence between maximum likelihood inference and minimum code length.
  - "A good compressor is a good predictor."

Mutual Information

Multivariate probabilities

- The marginal probability $p(X = x)$ is the probability of x happening, regardless of Y:

$$p(X = x) = \sum_{y \in \mathcal{Y}} p(X = x, Y = y)$$

- The conditional probability $p(X = x | Y = y)$ is the probability of x happening, given we know y happens.

$$p(X = x | Y = y) = \frac{p(X = x, Y = y)}{p(Y = y)}$$

Conditioning on impossible events

If $p(Y = y) = 0$, then $p(X | Y = y)$ is undefined.

Joint probability recap

Joint probabilities obey some fundamental properties:

- **Product rule**:

$$p(x, y) = p(x|y)p(y)$$

- **Sum rule**:

$$p(x) = \sum_{y \in \mathcal{Y}} p(x|y)p(y)$$

- **Bayes' theorem**:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}$$

Warning: Notation overload

Technically, the symbol p in the expressions $p(x)$ and $p(y)$ represents two different functions. The precise way to write this would be $p_X(X = x)$. When it is unambiguous from the context, we will use simply $p(x)$.

Given three RVs X, Y, Z, we distinguish between two types of independence:

- Marginal independence: $X \perp\!\!\!\perp Y \iff p(X, Y) = p(X)p(Y)$.
- Conditional independence: $X \perp\!\!\!\perp Y|Z \iff p(X, Y|Z) = p(X|Z)p(Y|Z)$.

GRAPHICAL MODELS

- Let's look at a particularly interesting Bayesian network:

DEFINITION: MARKOV CHAIN

A stochastic process $X_1, ..., X_n$ forms a Markov chain if, for all t,

$$p(X_t|X_{t-1}, X_{t-2}, ..., X_1) = p(X_t|X_{t-1})$$

- In writing, we will denote Markov chains as $X - Y - Z$.
- We can also consider Markov chains of order p:
  $p(X_t|X_{t-1}, X_{t-2}, ..., X_1) = p(X_t|X_{t-1}, ..., X_{t-p})$.

JOINT ENTROPY

DEFINITION: JOINT ENTROPY

The joint entropy of two RVs X and Y, with alphabets $\mathcal{X}$ and $\mathcal{Y}$, is given by

$$H(X, Y) = -\sum_{x,y \in \mathcal{X} \times \mathcal{Y}} p(x, y) \log p(x, y)$$

- All the properties of entropy apply to joint entropy:
  - Expected value of joint surprisal, $\mathbb{E}_p [-\log p(x, y)]$.
  - Uncertainty when predicting X and Y jointly.
  - Minimum code length when encoding X and Y simultaneously.
- Note that joint entropy is symmetric: $H(X, Y) = H(Y, X)$.

CONDITIONAL ENTROPY

DEFINITION: CONDITIONAL ENTROPY

The entropy of X conditioned on Y is given by

$$H(X|Y) = \sum_{y \in \mathcal{Y}} p(y) H(X|Y = y) \,,$$

where

$$H(X|Y = y) = -\sum_{x \in \mathcal{X}} p(x|y) \log p(x|y)$$

WARNING: EMPTY CONDITIONAL

Written this way, $H(X|Y)$ is not defined when $p(y) = 0$ for any y. To fix this, we can either take the sum over y with $p(y) > 0$, or define $\frac{0}{0} \log 0 = 0$.

Conditional entropy

Some properties of conditional entropy in discrete RVs:

- $H(X|Y) \geq 0$.
  - **Proof**: Since $0 \leq p(x|y) \leq 1$, $-\log p(x|y) \geq 0$.
- $H(X|Y) \leq \log |\mathcal{X}|$.
  - **Proof**: Similar to the proof of $H(X) \leq \log |\mathcal{X}|$.
- $H(X|Y) = 0$ if X is a deterministic function of Y.
  - **Proof**: Similar to the proof of $H(X) = 0$ for the Kronecker delta.
- (Applies to discrete and continuous.) If $X \perp\!\!\!\perp Y$, then $H(X|Y) = H(X)$.
  - **Proof**: If $X \perp\!\!\!\perp Y$, $p(x|y) = p(x)$. Then,
    $$H(X|Y=y) = -\sum_{x\in\mathcal{X}} p(x|y)\log p(x|y) = -\sum_{x\in\mathcal{X}} p(x)\log p(x) = H(X)$$

Conditional Kullback-Leibler divergence

Definition: Conditional KL divergence

The KL divergence between two conditional PDFs $p(X|Y)$ and $q(X|Y)$ is

$$D_{\mathrm{KL}}(p(X|Y) \parallel q(X|Y)) = \sum_{y\in\mathcal{Y}} p(y) D_{\mathrm{KL}}(p(X|Y=y) \parallel q(X|Y=y)) ,$$

where

$$D_{\mathrm{KL}}(p(X|Y=y) \parallel q(X|Y=y)) = \sum_{x\in\mathcal{X}} p(x|y)\log\frac{p(x|y)}{q(x|y)}$$

- Since CKL is the sum of non-negative terms, then CKL is non-negative.

Theorem: Entropy chain rule

The **entropy chain rule** states that

$$H(X, Y) = H(X|Y) + H(Y)$$

Tremendously useful property.

- Two RVs X, Y are *isomorphic* if there exists a (deterministic) bijective function f such that $X = f(Y)$ and $Y = f^{-1}(X)$.

Theorem: Invariance of entropy

Entropy in discrete RVs is invariant under isomorphisms.

Definition: Stationary stochastic process

A stochastic process is an indexed set of joint random variables $\{X_i\}$. Here we will consider *semi-infinite* processes $\{X_1, ..., X_n\}$.

A stochastic process is stationary if, for all t and s,

$$p(X_1, X_2, ..., X_t) = p(X_{1+s}, X_{2+s}, ..., X_{t+s})$$

- I.e. a stochastic process is stationary if it's invariant to time shift.

Let's introduce two interesting quantities:

- **Entropy rate**: how quickly $H(X_1, ..., X_n)$ grows as n increases.

$$h_1(X) = \lim_{n\to\infty} \frac{1}{n} H(X_1, ..., X_n)$$

- **Innovation**: how much new information is introduced at time n.

$$h_2(X) = \lim_{n\to\infty} H(X_n|X_1, ..., X_{n-1})$$

THEOREM: ENTROPY RATE

For stationary stochastic processes, these two quantities are equal:

$$h(X) = \lim_{n\to\infty} \frac{1}{n} H(X_1, ..., X_n) = \lim_{n\to\infty} H(X_n|X_1, ..., X_{n-1})$$

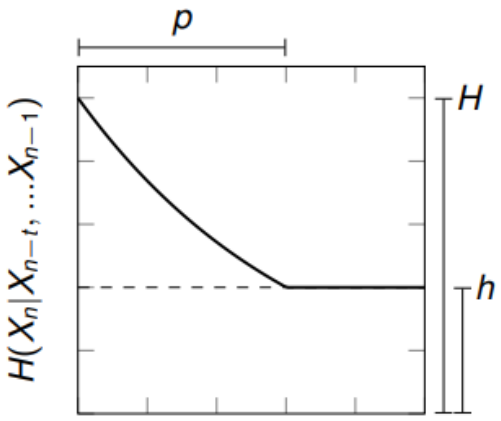
STOCHASTIC PROCESSES

- The more things you condition on, the less entropy you have left:

$$H(X_{n+1}|X_1, ..., X_n) \leq H(X_n|X_1, ..., X_{n-1})$$

- Intuitively: X is a time series, and $H(X_n|X_{n-t}, ...X_{n-1})$ is the performance of a predictor with t timesteps of "memory."
- Seen in the system's entropy rate convergence curve:

- Without memory ($t = 0$), we get the usual entropy $H(X_n)$.
- As we include more memory, conditional entropy decreases...
- ...Until it reaches a minimum at the entropy rate $h(X)$.
- If this happens at some finite t, this is the Markov order p.



Conditioning "memory" t

Mutual information

Definition: Mutual information

The **mutual information** between X and Y is given by

$$I(X;Y) := \sum_{x,y \in \mathcal{X} \times \mathcal{Y}} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

- Mutual information is symmetric: $I(X;Y) = I(Y;X)$.
- We use a semicolon (;) to separate the two arguments of MI, and a comma (,) to separate random variables.
  - E.g. $I(X,Y;Z)$ is the mutual information between (X,Y) and Z.

Mutual information

- We can rewrite MI as a difference in entropy:

$$I(X;Y) = H(X) + H(Y) - H(X,Y)$$

- Using the entropy chain rule, we can also write:

$$I(X;Y) = \underbrace{H(X)}_{\substack{\text{Uncertainty} \\ \text{about } X}} - \underbrace{H(X|Y)}_{\substack{\text{Uncertainty} \\ \text{about } X \\ \text{given } Y}}$$

- In other words: MI measures the average reduction of uncertainty in X after knowing Y (or vice versa).
- You can't get out more than you put in: $I(X;Y) \leq \min\{H(X), H(Y)\}$.
  - Follows from $H(X|Y) \geq 0$ (discrete variables only).

MI is the KL between the joint PDF and the product of the marginals:

$$I(X;Y) = D_{\mathrm{KL}}(p(X,Y) \parallel p(X)p(Y))$$

This comes with two very important consequences:

- Mutual information is non-negative.
  - By the non-negativity of KL.
- Mutual information is zero iff X and Y are independent ($X \perp\!\!\!\perp Y$).
  - Because $D_{\mathrm{KL}}(p \parallel q) = 0 \iff p = q$.

THEOREM: INFORMATION DOESN'T HURT

Conditioning reduces entropy:

$$H(X|Y) \leq H(X)$$

Proof: $H(X) - H(X|Y) = I(X;Y) \geq 0$.

- Note: this holds for continuous variables, although $H(X)$ and $H(X|Y)$ can both be negative.
- In general, $H(X|Y=y)$ can be greater than $H(X)$.

Conditional mutual information

Definition: Conditional mutual information

The **conditional mutual information** between X and Y given Z is defined as

$$I(X;Y|Z) := \sum_{x,y,z \in \mathcal{X}\times\mathcal{Y}\times\mathcal{Z}} p(x,y,z) \log \frac{p(x,y|z)}{p(x|z)p(y|z)}$$

- CMI can be rewritten in terms of conditional entropy:

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

- CMI is a KL divergence:

$$I(X;Y|Z) = D_{\mathrm{KL}}(p(X,Y|Z) \parallel p(X|Z)p(Y|Z))$$

- Thus, CMI is non-negative and is zero iff $X \perp\!\!\!\perp Y|Z$.

Mutual information chain rule

The **mutual information chain rule** states that

$$I(X,Y;Z) = I(X;Z) + I(Y;Z|X)$$

Theorem: Data Processing Inequality

If $X - Y - Z$ form a Markov chain,

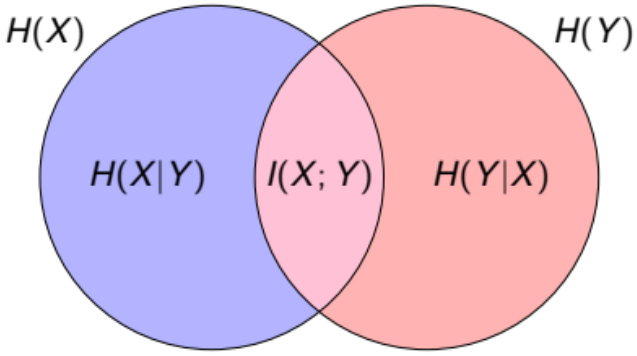
$$I(X;Z) \leq I(X;Y)$$

Theorem: Invariance of mutual information

Mutual information $I(X;Y)$ is invariant under isomorphisms in X and Y.

Connections with set theory

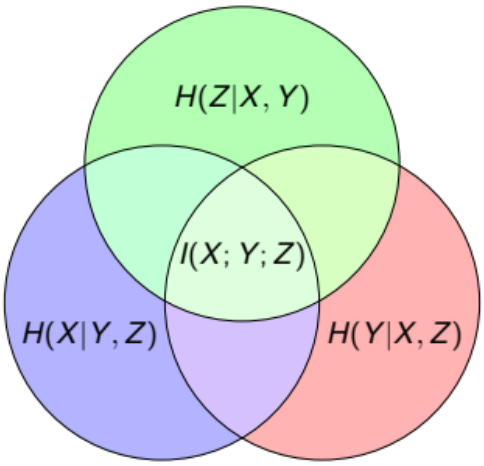
The relationship between H and I can be understood with a Venn diagram:



This motivates a **set-theoretic interpretation of entropy**:

$$\begin{aligned} H(X) &\longleftrightarrow \mu(A) \\ H(X,Y) &\longleftrightarrow \mu(A \cup B) \\ H(X|Y) &\longleftrightarrow \mu(A \setminus B) \\ I(X;Y) &\longleftrightarrow \mu(A \cap B) \end{aligned}$$

Connections with set theory



- This motivates the definition of new quantities like the co-information:

$$I(X;Y;Z) = H(X,Y,Z) - H(X,Y) - H(Y,Z) - H(X,Z) + H(X) + H(Y) + H(Z)$$

- But co-information can be negative. Information is in fact a signed measure.

(Related: Down & Mediano. A Logarithmic Decomposition for Information. ISIT 2023.)

MI IN GAUSSIAN DISTRIBUTIONS

- Consider two joint Gaussian variables X, Y with covariance

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{yx} & \sigma_{yy} \end{pmatrix} \quad \text{and} \quad |\boldsymbol{\Sigma}| = \sigma_{xx}\sigma_{yy} - \sigma_{xy}^2$$

Where $\sigma_{xx} = \text{var}(X) = \sigma_x^2$ and $\sigma_{y|x}^2 = \sigma_y^2 - \sigma_{xy}^2\sigma_x^{-2}$.

- Let's calculate:

$$\begin{aligned} I(X;Y) &= H(X) + H(Y) - H(X,Y) \\ &= \frac{1}{2}\ln 2\pi e\sigma_x^2 + \frac{1}{2}\ln 2\pi e\sigma_y^2 - \frac{1}{2}\ln|2\pi e\boldsymbol{\Sigma}| \\ &= -\frac{1}{2}\ln\frac{\sigma_x^2\sigma_y^2 - \sigma_{xy}^2}{\sigma_x^2\sigma_y^2} = -\frac{1}{2}\ln\left(1 - \frac{\sigma_{xy}^2}{\sigma_x^2\sigma_y^2}\right) \\ &= \underbrace{-\frac{1}{2}\ln\left(1-\rho_{xy}^2\right)}_{\text{Correlation}} = \underbrace{\frac{1}{2}\ln\frac{\sigma_{yy}}{\sigma_{y|x}^2}}_{\substack{\text{Explained} \\ \text{variance}}} \end{aligned}$$

MUTUAL INFORMATION AND PREDICTION

- Let's consider a supervised learning setting: we have samples $\{x_i, y_i\}$ and need to predict y_i from x_i.
- Usually we consider a parameterised family $\mathcal{Q} = \{q_\theta(y|x) : \theta \in \boldsymbol{\Theta}\}$.
  - E.g. non-linear function with additive noise: $q_\theta(y|x) = \mathcal{N}(y; f_\theta(x), \sigma^2)$.
- Recall Gibbs' inequality:

$$\mathbb{E}_p\left[\log\frac{p}{q}\right] \geq 0 \implies \mathbb{E}_p[\log p] \geq \mathbb{E}_p[\log q]$$

- Applying this to mutual information:

$$\begin{aligned} I(X;Y) &= H(Y) - H(Y|X) \\ &= H(Y) + \mathbb{E}_{p(x,y)}[\log p(y|x)] \\ &\geq H(Y) + \mathbb{E}_{p(x,y)}[\log q_\theta(y|x)] \end{aligned}$$

Log-likelihood of the predictor on the data

- If $p(y|x) \in \mathcal{Q}$, then $I(X;Y) = \max_\theta H(Y) + \mathbb{E}_p[\log q_\theta(y|x)]$.

MI is an upper bound on the performance of the best predictor.

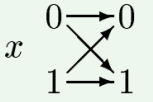
Definition: Discrete memoryless channel

A discrete memoryless channel (DMC) consists of an input alphabet $\mathcal{X}$, an output alphabet $\mathcal{Y}$, and a conditional distribution $p(y|x)$.

We will often work with an *extended channel* $p(y^n|x^n) = \prod_{i=1}^{n} p(y_i|x_i)$, i.e. sending n messages "in parallel."

Example: Binary symmetric channel

The BSC has binary inputs and outputs, and a probability f of flipping the input.



$$\begin{array}{lcllcl} P(y=0\,|\,x=0) & = & 1-f; & P(y=0\,|\,x=1) & = & f; \\ P(y=1\,|\,x=0) & = & f; & P(y=1\,|\,x=1) & = & 1-f. \end{array}$$

Example: Binary symmetric channel

Consider a BSC with the input distribution $p(X=0) = 1 - p(X=1) = 0.5$ and flip probability f. Let's calculate $I(X;Y)$.

Since $p(Y|X)$ is a Bernoulli distribution and the channel is symmetric,

$$H(Y|X=0) = H(Y|X=1) = H_2(f)$$

Thus, $H(Y|X) = H_2(f)$. By symmetry, $p(Y=0) = p(Y=1) = 0.5$. Thus,

$$I(X;Y) = H(Y) - H(Y|X) = 1 - H_2(f)$$

Definition: Channel capacity

The capacity of a channel $p(y|x)$ is given by

$$C = \max_{p(x)} I(X;Y)$$

The distribution $p^*(x)$ achieving the maximum is referred to as the *optimal input distribution* or *capacity-achieving distribution*.

KEY INTUITIONS

- We achieve error-free communication by picking a *non-confusable* subset of inputs (i.e. inputs with disjoint output sets).
- In the same way as AEP makes all PDFs close to uniform, it also makes all channels close to the noisy typewriter.

Preview:

- Number of possible channel outputs: $2^{nH(Y)}$.
- Number of confusable channel outputs for each input: $2^{nH(Y|X)}$.
- Number of non-confusable input-output pairs:

$$\frac{\text{\# total outputs for all inputs}}{\text{\# confusable outputs per input}} = \frac{2^{nH(Y)}}{2^{nH(Y|X)}} = 2^{n(H(Y)-H(Y|X))} = 2^{nI(X;Y)}$$

DEFINITION: BLOCK CODE

An (n, K) block code $\mathcal{S}$ is a set of 2^K codewords, each of length n. Mathematically:

$$\{x^n(1), x^n(2), ..., x^n(2^K)\}, \quad x^n(i) \in \mathcal{X}^n$$

Intuitively:

- $s \in \mathcal{S}$ are the source codewords, represented by a RV S with distribution $p(s) = 1/|\mathcal{S}| = 2^{-K}$. The corresponding channel input is $x^n(s)$.
- n is the number of channel uses to send one word.
- K is the information (i.e. number of bits) conveyed by one word.
- The rate $R = K/n$ is the number of bits conveyed per channel use.

each channel can only transmit one symble at one time (not the entire code word). p(s) means the probability of any speicific message.

- Decoder: A mapping $\mathcal{Y}^n \to \mathcal{S}$, denoted $\hat{s} = \mathrm{Dec}(y^n)$. Often we allow one more output symbol s_0 to represent a decoding failure.
- Mean probability of block error:

$$p_{\mathrm{B}} = \sum_s p(s) p(\hat{s} \neq s | s)$$

- Maximal probability of block error:

$$p_{\mathrm{BM}} = \max_s p(\hat{s} \neq s | s)$$

- Probability of bit error (p_b): Probability of error in one of the K bits that make up s.

Theorem: Noisy channel coding

Given a discrete memoryless channel:

- Its capacity C has the following property: For any $\varepsilon > 0$ and $R < C$, for large enough n there exists a code of length n and rate R and a decoding algorithm, such that $p_{\mathrm{BM}} < \varepsilon$.
- If a probability of bit error p_b is acceptable, then rates up to $R(p_b)$ are achievable, where

$$R(p_b) = \frac{C}{1 - H_2(p_b)} .$$

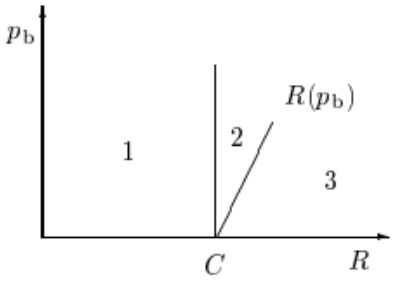
- For any p_b, rates greater than $R(p_b)$ are not achievable.

CHANNEL CODING THEOREM

The three statements of the CCT split the space of codes into three:

- ✓ (1) A region for which codes with any p_b exist (even $p_b = 0$).
- ✓ (2) A region where faster codes exist, although with some error ($p_b > 0$).
- ✗ (3) A prohibited region.

- **Let's prove (1)**, using four ingredients:
  - Joint typicality.
  - Random codes.
  - Typical-set decoding.
  - Expurgation.



"Faster code" means higher transmission rate.

DEFINITION: JOINTLY TYPICAL PAIRS

Let X, Y be joint RVs and X^n, Y^n be n i.i.d. samples from X, Y. A pair x^n, y^n is in the *jointly typical set* $J_\varepsilon^{(n)}$ if:

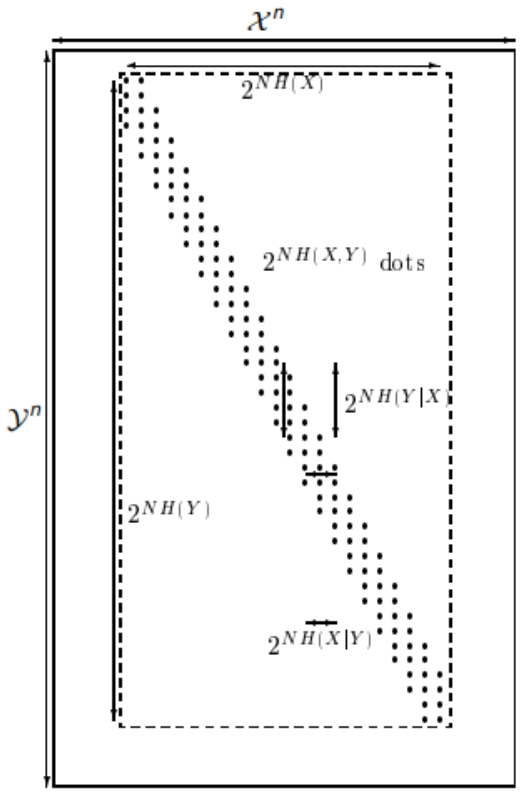
$$\left|\frac{1}{n}\log p(x^n) - H(X)\right| < \varepsilon$$
$$\left|\frac{1}{n}\log p(y^n) - H(Y)\right| < \varepsilon$$
$$\left|\frac{1}{n}\log p(x^n, y^n) - H(X, Y)\right| < \varepsilon$$

- Intuitively: x^n is typical of $p(x^n)$, y^n is typical of $p(y^n)$, and x^n, y^n is typical of $p(x^n, y^n)$.

- **Property 1**: If $x^n, y^n \in J_\varepsilon^{(n)}$, then $H(X,Y) - \varepsilon \leq -\frac{1}{n} \log p(x^n, y^n) \leq H(X,Y) + \varepsilon$.
  - In words: All jointly typical events have almost the same probability.
  - Proof: By definition of $J_\varepsilon^{(n)}$.
- **Property 2**: $\lim_{n\to\infty} \Pr\left\{x^n, y^n \in J_\varepsilon^{(n)}\right\} = 1$.
  - In words: Most samples are in the jointly typical set.
- **Property 3**: $|J_\varepsilon^{(n)}| \leq 2^{n(H(X,Y)+\varepsilon)}$.
  - In words: The number of items in $J_\varepsilon^{(n)}$ is upper-bounded by $2^{n(H(X,Y)+\varepsilon)}$.
  - Proof: Same as the proof for the typical set $A_\varepsilon^{(n)}$.
- **Property 4**: If $\tilde{x}^n, \tilde{y}^n \sim p(x^n)p(y^n)$, then $\Pr\left\{\tilde{x}^n, \tilde{y}^n \in J_\varepsilon^{(n)}\right\} \leq 2^{-n(I(X;Y)-3\varepsilon)}$.
  - In words: Independently sampled pairs are jointly typical with decreasing probability. The rate of decrease is the mutual info.

We can visualise the typical and jointly typical sets as follows:

- The horizontal span is the typical set of X.
- The area covered by the dots is the typical set of X, Y.
- Sampling from $p(x, y)$ is picking a dot at random.
- Sampling from $p(x)p(y)$ is picking a random point uniformly in the square.
- The ratio between the square area and the dots is the mutual info $I(X; Y)$.

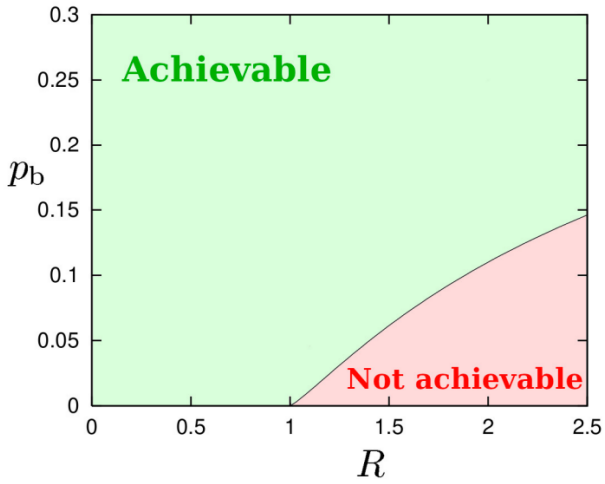


KEY TAKEAWAY: CHANNEL CODING THEOREM

We have proved the first part of the CCT, using 4 ingredients:

- We used a "channel AEP" to define **joinly typical sets**.
  - This turns any extended channel into a pseudo-typewriter: each input has a set of $(2^{nH(Y|X)})$ possible outputs and these output sets don't overlap.
- **Random codes**, which make taking averages easy.
  - By the "weighing babies" analogy, this proves existence of good codes.
- The **typical set decoder** isn't optimal, but it's good enough.
- **Expurgation** yields a bound on the maximum error based on the average error.

- The channel coding theorem is the cornerstone of communication systems.
- It puts fundamental limits on what is achievable and not achievable.
- It gives a rigorous interpretation of mutual information as the limit of data transmission.

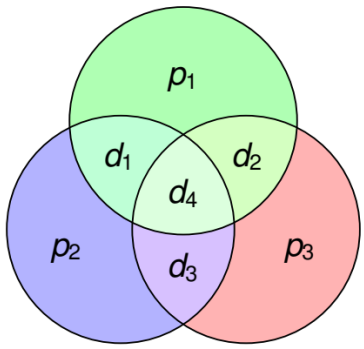


- The parity bits are placed in the positions of powers of 2:
  - For Hamming $(7, 4)$, they're 1, 2, and 4.

$$p_1 p_2 d_1 p_3 d_2 d_3 d_4$$

- The relationship between data and parity bits can be seen with a Venn diagram:

$$\begin{aligned} p_1 &= d_1 + d_2 + d_4 \mod 2 \\ p_2 &= d_1 + d_3 + d_4 \mod 2 \\ p_3 &= d_2 + d_3 + d_4 \mod 2 \end{aligned}$$



- We denote this as a parity bit covering certain data bits.

- The source string $\boldsymbol{s} \in B^k$ are the k data bits.
- The transmitted string $\boldsymbol{t} \in B^n$ are the n bits sent over the channel.
- It is given by the generator matrix:

$$\boldsymbol{t} = G^T \boldsymbol{s} \mod 2$$

- The noise string $\boldsymbol{n} \in B^n$ is the noise introduced by the channel.
- The received string $\boldsymbol{r} \in B^n$ are the n received bits.

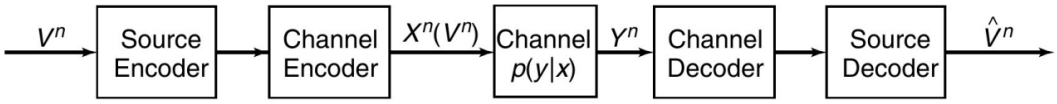
$$\boldsymbol{r} = \boldsymbol{t} + \boldsymbol{n} \mod 2$$

- The syndrome $\boldsymbol{z} \in B^m$ is the result of the parity check:

$$\boldsymbol{z} = H\boldsymbol{r} \mod 2$$

Source-channel separation theorem

- Say we want to send a RV V with arbitrary PDF through a channel.
- Naive option: first compress V into $H(V)$ uniform bits, then send those.



? Is this setup optimal?

Theorem: Source-channel separation

Given a stochastic process $V_1, ..., V_n$ we wish to send over a channel $p(Y|X)$, the optimum error-free communication rate can be achieved by a separate source code for V and a channel code for $Y|X$.

Probability Inference

Hypothesis testing

- The usual set-up involves two hypotheses:

$$\mathcal{H}_0 : \text{The object is a coin}$$
$$\mathcal{H}_1 : \text{The object is a die}$$

- Usually referred to as null and alternative hypotheses, respectively – thus the name null hypothesis significance testing (NHST).
- Typically, the null hypothesis represents "business as usual".
  - In some fields, $\mathcal{H}_1$ is referred to as a "discovery".
- Define a rejection region R such that $x \in R$ implies rejecting $\mathcal{H}_0$.

- We will deal only with parametric distributions $p_\theta(x)$, with $\theta \in \Theta$.
- A simple hypothesis is a point in parameter space: $\mathcal{H}_0 : \theta = \theta_0$.
- A composite hypothesis is a region in parameter space: $\mathcal{H}_0 : \theta \in \Theta_0$.
- $\mathcal{H}_0$ is nested if it corresponds to "fixing" one of the parameters in $\mathcal{H}_1$.

Example: Nested testing in linear regression

- You wake up to find out you are French.
- You want to make your croissant business succeed by adding x additional grams of butter to each croissant.
- You try different levels of butter x and record the sales y. You model it with a linear relationship, with unknown parameters $\theta = \{a, b\}$:

$$y = a \cdot x + b$$

- The null hypothesis "butter does not increase sales" is nested:

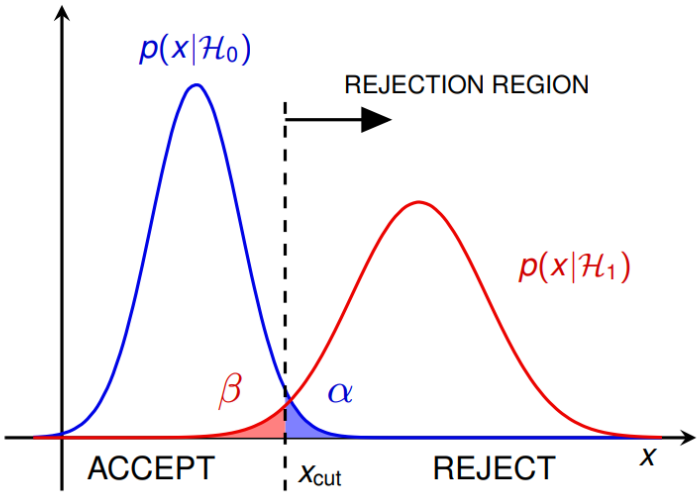
$$\mathcal{H}_0 : a = 0 \qquad \mathcal{H}_1 : a \neq 0$$

- **Type I error** is the false positive rate, i.e. probability of claiming a false discovery:

$$\alpha := P(x \in R | \mathcal{H}_0)$$

- **Type II error** is the false negative rate, i.e. probability of missing out on a real discovery:

$$\beta := P(x \notin R | \mathcal{H}_1)$$



Mnemonic: The boy who cried wolf

With type I and II errors, remembering which is which can be confusing. I use the tale of The boy who cried wolf:

- First, the boy claimed there was a wolf when there wasn't (mistaken rejection of $\mathcal{H}_0$, type I, α).
- Second, the villagers thought there was no wolf when there was one (failure to reject a false $\mathcal{H}_0$, type II, β).

- The power of a test is given by $1 - \beta$ (i.e. probability of finding an effect if it is real).
- Typically, we put an upper bound on α (to avoid too many false discoveries) and try to design a powerful test.

power is like ability to not miss a discovery

Theorem: Neyman-Pearson lemma

Consider a test between two point hypotheses $\mathcal{H}_0 : \theta = \theta_0$ vs $\mathcal{H}_1 : \theta = \theta_1$. The most powerful test with given type I error α is given by the region

$$R = \left\{ x : \frac{p_{\theta_1}(x)}{p_{\theta_0}(x)} > k \right\} ,$$

where k is chosen such that $p_{\theta_0}(x \in R) = \alpha$.

- The quantity above is known as the likelihood ratio.
- The Neyman-Pearson lemma establishes the likelihood ratio as the optimal test statistic.
- Intuitively: We reject $\mathcal{H}_0$ if the data is much more likely to come from $\mathcal{H}_1$.

- k is a threshold chosen to satisfy the significance level α:

$$P_{\theta_0}(x \in R) = \alpha$$

- This ensures that the probability of rejecting H_0 when H_0 is true is exactly α.

Definition: Likelihood ratio

The likelihood ratio statistic Λ of a hypothesis test $\mathcal{H}_0 : \theta \in \Theta_0$ vs $\mathcal{H}_1 : \theta \in \Theta_1$ is given by

$$\Lambda = \frac{\sup_{\theta \in \Theta_1} p_\theta(x)}{\sup_{\theta \in \Theta_0} p_\theta(x)}$$

Warning: Optimality of likelihood ratios

- The Neyman-Pearson lemma states that Λ is optimal for point hypotheses.
- The Karlin-Rubin theorem (not shown) states that Λ is optimal for composite hypotheses with monotonic likelihood functions.
- But Λ may not be optimal in general – although it's often really good.

- Intuitively, Λ measures how much more likely the data is under H_1 compared to H_0. Larger values of Λ provide more evidence in favor of H_1.

- Let $\theta_0^* = \text{argmax}_{\theta \in \Theta_0} p_\theta(x)$ and $\theta_1^* = \text{argmax}_{\theta \in \Theta_1} p_\theta(x)$.
- It's common to work with the log-likelihood

$$\log \Lambda = \log p_{\theta_1^*}(x) - \log p_{\theta_0^*}(x)$$

- Often, $\log \Lambda$ is calculated on n i.i.d. datapoints from $X \sim p(x)$:

$$\sum_{i=1}^{n} \log p_{\theta_1^*}(x_i) - \log p_{\theta_0^*}(x_i) \xrightarrow{n \to \infty} n \mathbb{E}_p \left[\log p_{\theta_1^*}(x) - \log p_{\theta_0^*}(x) \right]$$

- If $\mathcal{H}_1$ is true:
  - $\log p_{\theta_1^*}(x)$ grows linearly with n; and
  - $\log p_{\theta_0^*}(x)$ grows logarithmically with n.
- In frequentist statistics, $\log \Lambda$ is called the evidence for $\mathcal{H}_1$ over $\mathcal{H}_0$.
- A model is supported by an event to the extent that the event is unsurprising given the model.

- From an inference point of view, mutual information represents the **evidence for a joint model over a factorised model**:

$$I(X;Y) = \frac{1}{n}\mathbb{E}\left[\log \mathbf{\Lambda}\right] = \mathbb{E}_{p(x,y)}\left[\log \frac{p(x,y)}{p(x)p(y)}\right]$$

KEY TAKEAWAY

- Most information quantities can be seen as a **log-likelihood ratio**.
- This is (in many cases) the **optimal hypothesis test**.

- Given a likelihood and a prior, Bayes' rule provides the posterior as

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)} \ .$$

- In words, these quantities correspond to:

$$\text{Posterior} = \frac{\text{Likelihood} \ \times \ \text{Prior}}{\text{Evidence}} \ ,$$

- Bayes' rule is the direct consequence of the sum and product rules.
- The results of applying Bayes' rule are the posterior and the evidence.

- The posterior captures what values of θ are compatible with the data.
- Finding the posterior is known as the inverse probability problem.
  - We typically think of $p(D|\theta)$ as the "forward" probabilities that generate the data, and finding the posterior is "inverting" $p(D|\theta)$.
- Once the posterior is known, new predictions D' can be made by averaging across all θ values:

$$p(D'|D) = \int d\theta p(D'|\theta)p(\theta|D)$$

- (Note that conditioning on θ makes D and D' independent. This is the key feature of parametric models: $D - \theta - D'$ form a Markov chain.)
- The "minimally Bayesian" analogue of maximum likelihood is maximum a posteriori estimation:

$$\theta_{\text{MAP}} = \underset{\theta}{\text{argmax}}\, p(\theta|D) = \underset{\theta}{\text{argmax}}\, p(D|\theta)p(\theta)$$

Example: Bayesian inference in Gaussians

Let's do Bayesian inference on the mean of a Gaussian distribution, μ.

- The likelihood is Gaussian: $x \sim \mathcal{N}(x; \mu, \sigma^2)$.
- The prior is also Gaussian: $\mu \sim \mathcal{N}(\mu; \mu_0; \sigma_0^2)$.

The posterior, obtained from Bayes' rule, is

$$\mu_1 = \frac{\sigma_0^2 x + \sigma^2 \mu_0}{\sigma_0^2 + \sigma^2} \qquad \frac{1}{\sigma_1^2} = \left(\frac{1}{\sigma_0^2} + \frac{1}{\sigma^2}\right)$$

- The remaining term is the evidence (or *marginal likelihood*) $p(D)$.
- It is calculated by integrating out the parameters (replacing the integral with a sum when appropriate):

$$p(D) = \int d\theta p(D|\theta)p(\theta)$$

- It is the overall probability that the model generated the observed data.

Bayesian Hypothesis Testing

- Let's rephrase the problem of hypothesis testing in Bayesian terms to learn new things about it.
- As before, we have a null and alternative hypotheses – this time with a prior on them: $p(\mathcal{H}_0)$ (and $p(\mathcal{H}_1) = 1 - p(\mathcal{H}_0)$).
- Each hypothesis induces a different prior on θ:

$$\mathcal{H}_0 : \theta \sim p(\theta|\mathcal{H}_0)$$
$$\mathcal{H}_1 : \theta \sim p(\theta|\mathcal{H}_1)$$

- We can do Bayesian inference to compute both posteriors:

$$p(\theta|D, \mathcal{H}) = \frac{p(D|\theta, \mathcal{H})p(\theta|\mathcal{H})}{p(D|\mathcal{H})} .$$

- We can also compute the posterior over $\mathcal{H}$:

$$p(\mathcal{H}|D) = \frac{p(D|\mathcal{H})\ p(\mathcal{H})}{p(D)}$$

- The new "likelihood" is the evidence from the previous slide.
- Bayesian hypothesis testing is a **two-stage inference** process:
  - First compute the evidence for each hypothesis, $p(D|\mathcal{H})$.
  - Then combine them with another Bayes' rule.
- The "second-level" evidence is often easy to calculate, since the number of hypotheses is finite and small:

$$p(D) = p(D|\mathcal{H}_0)p(\mathcal{H}_0) + p(D|\mathcal{H}_1)p(\mathcal{H}_1)$$

- To test $\mathcal{H}_0$ vs $\mathcal{H}_1$, we can make a "Bayesian likelihood ratio":

$$\frac{p(\mathcal{H}_1|D)}{p(\mathcal{H}_0|D)} = \frac{p(D|\mathcal{H}_1)}{p(D|\mathcal{H}_0)} \frac{p(\mathcal{H}_1)}{p(\mathcal{H}_0)}$$

Definition: Bayes factor

In Bayesian hypothesis testing, the Bayes factor represents the degree to which the evidence shifts towards $\mathcal{H}_1$ after observing the data:

$$\text{BF} = \frac{p(D|\mathcal{H}_1)}{p(D|\mathcal{H}_0)}$$

- Unlike frequentist tests, Bayes Factors can reject either hypothesis:
  - BF $\gg 1$: strong evidence for $\mathcal{H}_1$.
  - BF ≈ 1: inconclusive test.
  - BF $\ll 1$: strong evidence for $\mathcal{H}_0$.

Occam's razor

Accept the simplest explanation that fits the data.

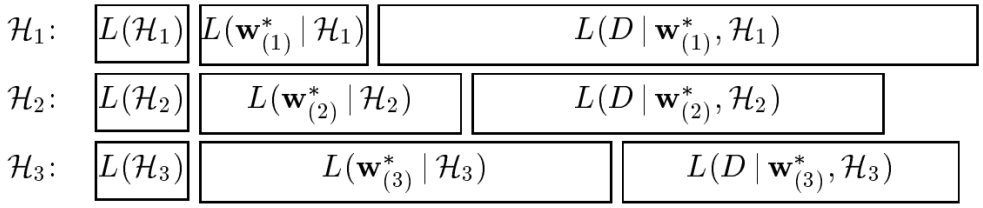
- The minimum description length (MDL) principle is a complementary view of model comparison based on information theory.

The MDL Principle (Shannon's razor)

Good models yield **short descriptions** of the data.

- Consider a dataset D we wish to compress.
- One approach is to compress the data in a three-part message:
  - We encode the hypothesis $\mathcal{H}_i$.
  - We encode the value of the optimal parameters $\boldsymbol{w}^*$.
  - We encode the value of the data D using $\boldsymbol{w}^*$.
- The resulting code length is the sum of these three:

$$L(D) = L(\mathcal{H}_i) + L(\boldsymbol{w}^*|\mathcal{H}_i) + L(D|\boldsymbol{w}^*, \mathcal{H}_i)$$



- $\mathcal{H}_1$ produces a simple model (low $L(\boldsymbol{w}^*|\mathcal{H}_1)$) but it makes the data very unlikely (thus high $L(D|\boldsymbol{w}^*, \mathcal{H}_1)$).
- $\mathcal{H}_3$ has the opposite pattern (good fit, but complicated model).
- $\mathcal{H}_2$ offers the best trade-off.

KEY TAKEAWAY

Occam's razor "falls out" from information theory and data compression.

Estimating Information

- Other methods are based on the concept of contrastive estimation.
- Contrastive methods rely on the comparison between so-called positive and negative samples.

Consider the following algorithm:

1. Start with a dataset $\{x_i, y_i\}$ (positive samples from $p(x, y)$).
2. Generate negative samples (from $p(x)p(y)$) by picking random x_i, y_j.
3. Train a classifier $f_\theta(x, y)$ to discriminate positive from negative samples.
4. The cross-entropy loss of the classifier lower-bounds mutual information:

$$I(X; Y) \geq \mathcal{L}_{\text{CE}} = -\mathbb{E}\left[\log \frac{f_\theta(x_i, y_i)}{\sum_{j=1}^{M} f_\theta(x_j, y_i)}\right]$$

Optimal classifier: $f_\theta(x, y) = \frac{p(x,y)}{p(x)p(y)}$

- Other methods are based on the concept of contrastive estimation.
- Contrastive methods rely on the comparison between so-called positive and negative samples.

Consider the following algorithm:

1. Start with a dataset $\{x_i, y_i\}$ (positive samples from $p(x, y)$).
2. Generate negative samples (from $p(x)p(y)$) by picking random x_i, y_j.
3. Train a classifier $f_\theta(x, y)$ to discriminate positive from negative samples.
4. The cross-entropy loss of the classifier lower-bounds mutual information:

$$I(X; Y) \geq \mathcal{L}_{\text{CE}} = -\mathbb{E}\left[\log \frac{f_\theta(x_i, y_i)}{\sum_{j=1}^{M} f_\theta(x_j, y_i)}\right]$$

Optimal classifier: $f_\theta(x, y) = \frac{p(x,y)}{p(x)p(y)}$

- Other methods are based on the concept of contrastive estimation.
- Contrastive methods rely on the comparison between so-called positive and negative samples.

Consider the following algorithm:

1. Start with a dataset $\{x_i, y_i\}$ (positive samples from $p(x, y)$).
2. Generate negative samples (from $p(x)p(y)$) by picking random x_i, y_j.
3. Train a classifier $f_\theta(x, y)$ to discriminate positive from negative samples.
4. The cross-entropy loss of the classifier lower-bounds mutual information:

$$I(X; Y) \geq \mathcal{L}_{\text{CE}} = -\mathbb{E}\left[\log \frac{f_\theta(x_i, y_i)}{\sum_{j=1}^{M} f_\theta(x_j, y_i)}\right]$$

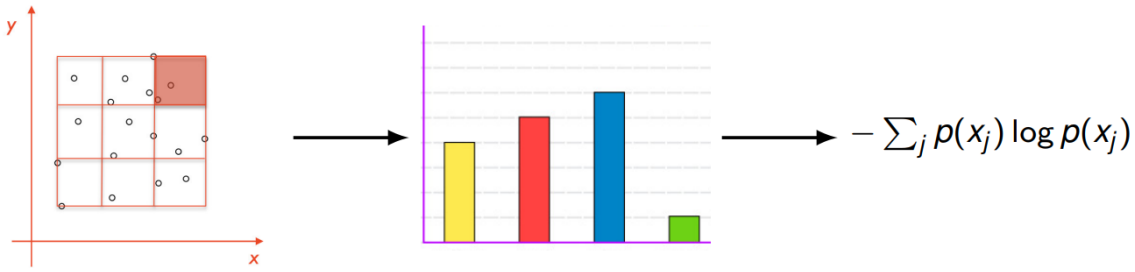
Optimal classifier: $f_\theta(x, y) = \frac{p(x,y)}{p(x)p(y)}$

Approach 0: The plug-in method

Given a set of i.i.d. samples from a discrete RV X with known alphabet $\mathcal{X}$, the *plug-in* (or *counting*) estimator of $p(x)$ is

$$p(x_j) = \frac{\#\text{ samples with } X = x_j}{\#\text{ samples in total}}$$

- These probabilities can then be "plugged in" to the entropy formula.



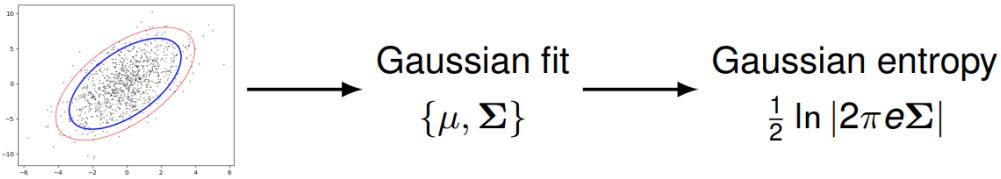
Apply to discrete data

Approach I: Parametric methods

Given a set of n i.i.d. samples x^n from a RV $X \sim p(x)$ and a parametric family of PDFs $q_\theta(x)$, a *parametric estimator* of H is given by

$$H(X) = -\mathbb{E}_{q_{\theta^*}}\left[\log q_{\theta^*}(x)\right] \qquad \text{with} \qquad \theta^* = \operatorname*{argmax}_\theta q_\theta(x^n)\ .$$

- Intuitively: Fit a distribution to the data, then compute information quantities analytically for that distribution.



- Closed-form information quantities are available for many PDFs.
  - Gaussian, gamma, exponential, beta, ...

Warning: Garbage in, garbage out

A parametric estimator is only suitable to the extent that the model describes the data well.

Approach II: Non-parametric methods

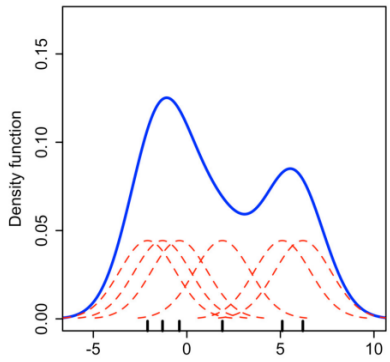
Given a set of n i.i.d. samples from a RV $X \sim p(x)$ and a PDF estimate $q(x)$, a *non-parametric estimator* of H is given by

$$H(X) = -\mathbb{E}_p \left[\log q(x)\right] \ .$$

- Intuitively: Fit a distribution to the data, then evaluate the new distribution *on the original data*.

Common approach: **kernels**.

1. Start with x_i samples.
2. Put small "bumps" on each x_i.
3. Sum them to get $p(x)$.



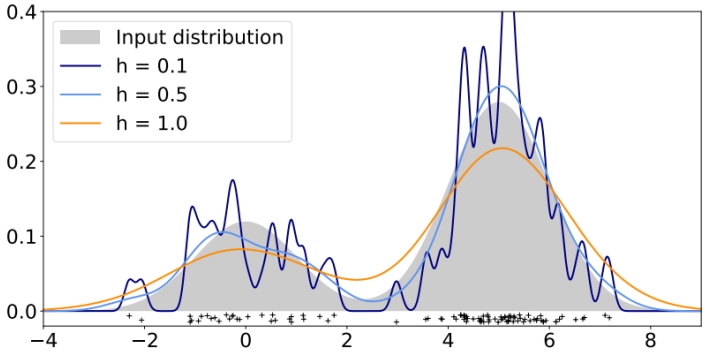
- A kernel is a fancy word for a "soft window".
- For example, the squared exponential kernel with width h is

$$K_h(x - x_i) = (2h^2\pi)^{-1/2} \exp\left(-\frac{(x-x_i)^2}{2h^2}\right)$$

- The resulting distribution is estimated as the average over datapoints:

$$q(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i)$$

- Results depend on h, but can be optimised e.g. through cross-validation.



Approach III: Variational methods

Given a set of n i.i.d. samples x^n from a RV $X \sim p(x)$ and a parameterised bound $\hat{H}_\theta(x^n)$, a *variational estimator* of H is given by

$$H(X) = \max_\theta \hat{H}_\theta(x^n)$$

- Intuitively: Variational methods turn an estimation problem into an optimisation problem.

- Other methods are based on the concept of contrastive estimation.
- Contrastive methods rely on the comparison between so-called positive and negative samples.

Consider the following algorithm:

1. Start with a dataset $\{x_i, y_i\}$ (positive samples from $p(x, y)$).
2. Generate negative samples (from $p(x)p(y)$) by picking random x_i, y_j.
3. Train a classifier $f_\theta(x, y)$ to discriminate positive from negative samples.
4. The cross-entropy loss of the classifier lower-bounds mutual information:

$$I(X; Y) \geq \mathcal{L}_{\text{CE}} = -\mathbb{E}\left[\log \frac{f_\theta(x_i, y_i)}{\sum_{j=1}^{M} f_\theta(x_j, y_i)}\right]$$

Optimal classifier: $f_\theta(x, y) = \frac{p(x,y)}{p(x)p(y)}$

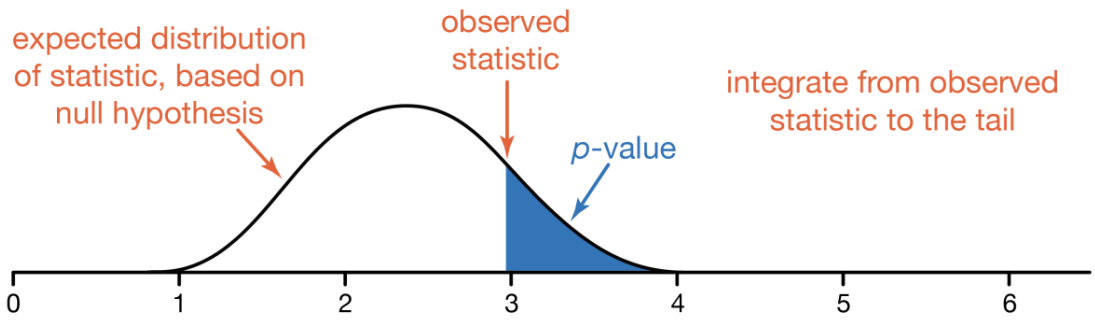
Definition: P-value

The p-value is the probability, under the null hypothesis, of obtaining a result equal to or more extreme than an observed $T(x)$.

- A null hypothesis is rejected and the test declared as significant at level α if $p_{\text{value}} < \alpha$. The most common threshold is $\alpha = 0.05$.
- The key object here is the PDF of $T(X)$ under $\mathcal{H}_0$, $p_{\theta_0}(T(X) = t)$, commonly referred to as the null distribution.
- The rejection threshold (or critical value) t_{crit} is the solution to the equation

$$\alpha = \int_{t_{\text{crit}}}^{\infty} p_{\theta_0}(T(X) = t)\, dt$$

These can be visualised together as follows:



Definition: Method of surrogate data

The method of surrogate data consists of numerically sampling from the null distribution empirically by generating 'fake' (or *surrogate*) data where $\mathcal{H}_0$ holds.

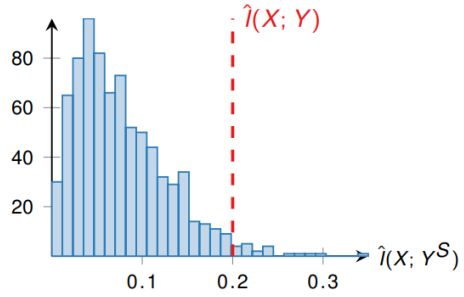
This is an **extremely useful procedure**.

- It typically works by "removing" or randomising the property of interest.
- Ideal surrogate methods remove only the information you want to test, and leave the data otherwise untouched.
- Coming up with a surrogate data method for an arbitrary $\mathcal{H}_0$ is in general difficult, but many options exist for common choices of $\mathcal{H}_0$.

Example: Surrogates for independence test

We want to test if $X \perp\!\!\!\perp Y$. As test statistic, we pick $I(X; Y)$. We can use the following algorithm:

1. Estimate $\hat{I}(X; Y)$ from observed data.
2. Randomly shuffle Y to obtain Y^S (which has no relation to X).
3. Compute $\hat{I}(X; Y^S)$ for all surrogate data.
4. Compare against the original estimate $\hat{I}(X; Y)$ to get a p-value.



$$\text{p-value} = \frac{\#\text{ surrogates with } \hat{I}(X;Y^S) \geq \hat{I}(X;Y)}{\#\text{ surrogates in total}}$$

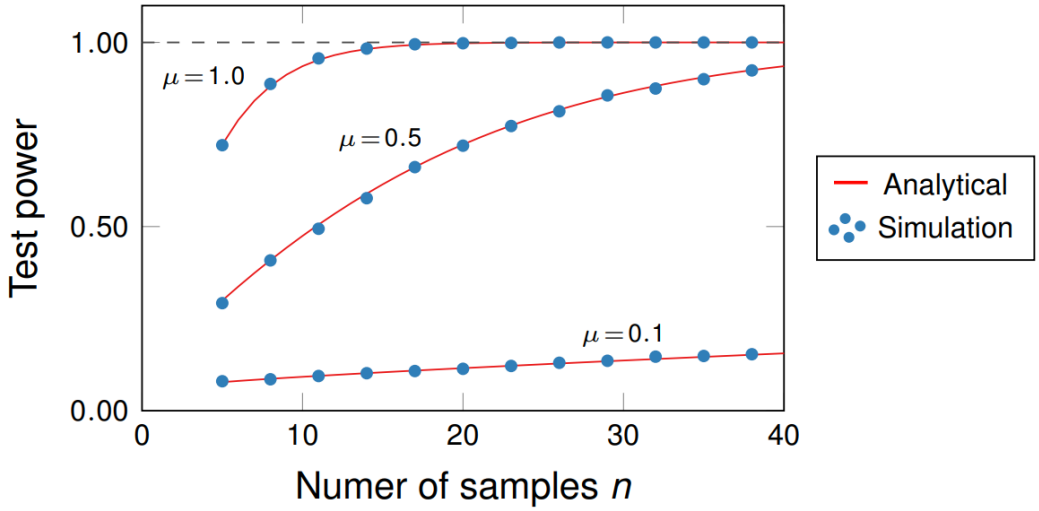
Definition: Power analysis

Set of techniques to determine the sample size needed to detect a statistical effect.

- Power analysis allows us to answer questions like:
  - How many samples do I need to reject $\mathcal{H}_0$ with a given α?
  - How many samples do I need to get an accurate estimate of $T(x)$?
  - What is the smallest effect I could reliably detect with a given sample size?
- As with significance testing, analytical results are difficult, so we often resort to numerical simulations.

Power can be estimated numerically with the following algorithm:

1. Pick a value of n and a value of $\mu > 0$.
2. Draw n samples from the alternative distribution, i.e. $\mathcal{N}(\mu, \sigma^2)$.
3. Test whether these n samples pass a NHST at level α.
4. Repeat this process M times. The power of the test is the fraction of times the null hypothesis is rejected.



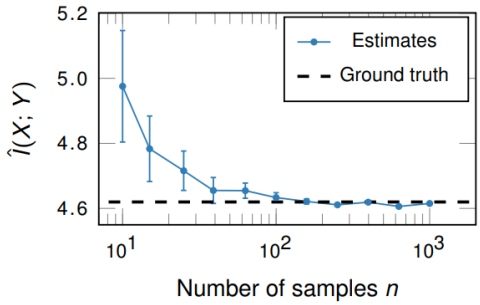
- Power increases with the true effect and with the sample size.
- In some cases this can be done analytically.
  - For common distributions (e.g. Gaussian) this can be done with their CDF and inverse CDF functions (e.g. from `scipy.stats`).
  - An extension of Wilks' theorem shows that information quantities are distributed as a non-central chi-squared $\chi^2(\lambda)$.

Algorithm to estimate bias in mutual information:

1. Estimate some $p(x, y)$ from data.
2. Use $p(x, y)$ to compute "ground truth" $I(X; Y)$.
3. Repeat M times:
   I. Draw n random samples from $p(x, y)$.
   II. Use samples to estimate $\hat{I}(X; Y)$.
4. Compare ground truth $I(X; Y)$ against estimates $\hat{I}(X; Y)$.

- **Example**: bias in MI between two 2D Gaussian variables.

Very powerful approach!



KEY TAKEAWAY

- **Surrogate data analysis** as the prime tool for significance testing in data analysis – including analyses based on information theory.
- **Power analysis** as the prime tool for analysis design and calibration.
- Common to both is the concept of **simulating** the distribution under study (null for significance testing; alternative for power analysis).