COMP2610 – Information Theory

Lecture 12: The Source Coding Theorem

Robert C. Williamson

Research School of Computer Science

Australian National University

28 August 2018

Last time

Basic goal of compression

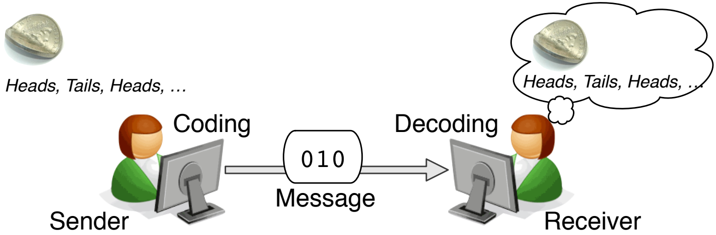
Key concepts: codes and their types, raw bit content, essential bit content

Informal statement of source coding theorem

A General Communication Game (Recap)

Data compression is the process of replacing a message with a smaller message which can be reliably converted back to the original.

- Want small messages on average when outcomes are from a fixed, known, but uncertain source (e.g., coin flips with known bias)



Definitions (Recap)

Source Code

Given an ensemble X, the function $c : \mathcal{A}_X \rightarrow \mathcal{B}$ is a **source code** for X.
The number of symbols in $c(x)$ is the **length** $l(x)$ of the codeword for x.
The **extension** of c is defined by $c(x_1 \ldots x_n) = c(x_1) \ldots c(x_n)$

Definitions (Recap)

Source Code

Given an ensemble X, the function $c : \mathcal{A}_X \to \mathcal{B}$ is a **source code** for X.
The number of symbols in $c(x)$ is the **length** $l(x)$ of the codeword for x.
The **extension** of c is defined by $c(x_1 \ldots x_n) = c(x_1) \ldots c(x_n)$

Smallest δ-sufficient subset

Let X be an ensemble and for $\delta \geq 0$ define S_δ to be the smallest subset of $\mathcal{A}_X$ such that

$$P(x \in S_\delta) \geq 1 - \delta$$

Definitions (Recap)

Source Code

Given an ensemble X, the function $c : \mathcal{A}_X \to \mathcal{B}$ is a **source code** for X.
The number of symbols in $c(x)$ is the **length** $l(x)$ of the codeword for x.
The **extension** of c is defined by $c(x_1 \ldots x_n) = c(x_1) \ldots c(x_n)$

Smallest δ-sufficient subset

Let X be an ensemble and for $\delta \geq 0$ define S_δ to be the smallest subset of $\mathcal{A}_X$ such that

$$P(x \in S_\delta) \geq 1 - \delta$$

Essential Bit Content

Let X be an ensemble then for $\delta \geq 0$ the **essential bit content** of X is

$$H_\delta(X) \stackrel{\text{def}}{=} \log_2 |S_\delta|$$

Essential Bit Content (Recap)

Intuitively, construct S_δ by removing elements of X in ascending order of probability, till we have reached the $1 - \delta$ threshold

$\mathbf{x}$	$P(\mathbf{x})$
a	1/4
b	1/4
c	1/4
d	3/16
e	1/64
f	1/64
g	1/64
h	1/64

- Outcomes ranked (high–low) by $P(x = a_i)$ removed to make set S_δ with $P(x \in S_\delta) \geq 1 - \delta$

  $\delta = 0$: $S_\delta = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}, \mathtt{f}, \mathtt{g}, \mathtt{h}\}$

Essential Bit Content (Recap)

Intuitively, construct S_δ by removing elements of X in ascending order of probability, till we have reached the $1 - \delta$ threshold

$\mathbf{x}$	$P(\mathbf{x})$
a	1/4
b	1/4
c	1/4
d	3/16
e	1/64
f	1/64
g	1/64

- Outcomes ranked (high–low) by $P(x = a_i)$ removed to make set S_δ with $P(x \in S_\delta) \geq 1 - \delta$

$$\delta = 0 \; : S_\delta = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}, \mathtt{f}, \mathtt{g}, \mathtt{h}\}$$
$$\delta = 1/64 \; : S_\delta = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}, \mathtt{f}, \mathtt{g}\}$$

Essential Bit Content (Recap)

Intuitively, construct S_δ by removing elements of X in ascending order of probability, till we have reached the $1 - \delta$ threshold

$\mathbf{x}$	$P(\mathbf{x})$
a	1/4
b	1/4
c	1/4
d	3/16

- Outcomes ranked (high–low) by $P(x = a_i)$ removed to make set S_δ with $P(x \in S_\delta) \geq 1 - \delta$

$$\begin{aligned} \delta = 0 &: S_\delta = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}, \mathtt{f}, \mathtt{g}, \mathtt{h}\} \\ \delta = 1/64 &: S_\delta = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}, \mathtt{f}, \mathtt{g}\} \\ \delta = 1/16 &: S_\delta = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\} \end{aligned}$$

Essential Bit Content (Recap)

Intuitively, construct S_δ by removing elements of X in ascending order of probability, till we have reached the $1 - \delta$ threshold

$\mathbf{x}$	$P(\mathbf{x})$
a	1/4

- Outcomes ranked (high–low) by $P(x = a_i)$ removed to make set S_δ with $P(x \in S_\delta) \geq 1 - \delta$

$$\begin{aligned}
\delta = 0 &: S_\delta = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}, \mathtt{f}, \mathtt{g}, \mathtt{h}\} \\
\delta = 1/64 &: S_\delta = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}, \mathtt{f}, \mathtt{g}\} \\
\delta = 1/16 &: S_\delta = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\} \\
\delta = 3/4 &: S_\delta = \{\mathtt{a}\}
\end{aligned}$$

Lossy Coding (Recap)

Consider a coin with $P(\mathit{Heads}) = 0.9$

If we are happy to fail on up to 2% of the sequences we can ignore any sequence of 10 outcomes with more than 3 tails

There are only $176 < 2^8$ sequences with 3 or fewer tails

So, we can just code those, and **ignore** the rest!

- Coding 10 outcomes with 2% failure doable with 8 bits, or 0.8 bits/outcome

This time

Recap: typical sets

Formal statement of source coding theorem

Proof of source coding theorem

The Source Coding Theorem

(Theorem 4.1 in MacKay)

Our aim this week is to understand this:

The Source Coding Theorem

Let X be an ensemble with entropy $H = H(X)$ bits. Given $\epsilon > 0$ and $0 < \delta < 1$, there exists a positive integer N_0 such that for all $N > N_0$

$$\left| \frac{1}{N} H_\delta \left(X^N \right) - H \right| < \epsilon.$$

The Source Coding Theorem

(Theorem 4.1 in MacKay)

Our aim this week is to understand this:

The Source Coding Theorem

Let X be an ensemble with entropy $H = H(X)$ bits. Given $\epsilon > 0$ and $0 < \delta < 1$, there exists a positive integer N_0 such that for all $N > N_0$

$$\left| \frac{1}{N} H_\delta \left(X^N \right) - H \right| < \epsilon.$$

In English:

- Given outcomes drawn from X . . .

The Source Coding Theorem

(Theorem 4.1 in MacKay)

Our aim this week is to understand this:

The Source Coding Theorem

Let X be an ensemble with entropy $H = H(X)$ bits. Given $\epsilon > 0$ and $0 < \delta < 1$, there exists a positive integer N_0 such that for all $N > N_0$

$$\left| \frac{1}{N} H_\delta \left(X^N\right) - H \right| < \epsilon.$$

In English:

- Given outcomes drawn from X ...
- ... no matter what *reliability* $1 - \delta$ and *tolerance* ϵ you choose ...

The Source Coding Theorem

(Theorem 4.1 in MacKay)

Our aim this week is to understand this:

The Source Coding Theorem

Let X be an ensemble with entropy $H = H(X)$ bits. Given $\epsilon > 0$ and $0 < \delta < 1$, there exists a positive integer N_0 such that for all $N > N_0$

$$\left| \frac{1}{N} H_\delta \left(X^N \right) - H \right| < \epsilon.$$

In English:

- Given outcomes drawn from X ...
- ... no matter what *reliability* $1 - \delta$ and *tolerance* ϵ you choose ...
- ... there is always a length N_0 so sequences X^N longer than this ...

The Source Coding Theorem

(Theorem 4.1 in MacKay)

Our aim this week is to understand this:

The Source Coding Theorem

Let X be an ensemble with entropy $H = H(X)$ bits. Given $\epsilon > 0$ and $0 < \delta < 1$, there exists a positive integer N_0 such that for all $N > N_0$

$$\left| \frac{1}{N} H_\delta \left(X^N \right) - H \right| < \epsilon.$$

In English:

- Given outcomes drawn from X ...
- ... no matter what *reliability* $1 - \delta$ and *tolerance* ϵ you choose ...
- ... there is always a length N_0 so sequences X^N longer than this ...
- ... have an average essential bit content $\frac{1}{N} H_\delta(X^N)$ within ϵ of $H(X)$

The Source Coding Theorem

(Theorem 4.1 in MacKay)

Our aim this week is to understand this:

The Source Coding Theorem

Let X be an ensemble with entropy $H = H(X)$ bits. Given $\epsilon > 0$ and $0 < \delta < 1$, there exists a positive integer N_0 such that for all $N > N_0$

$$\left| \frac{1}{N} H_\delta\left(X^N\right) - H \right| < \epsilon.$$

In English:

- Given outcomes drawn from X ...
- ... no matter what *reliability* $1 - \delta$ and *tolerance* ϵ you choose ...
- ... there is always a length N_0 so sequences X^N longer than this ...
- ... have an average essential bit content $\frac{1}{N} H_\delta(X^N)$ within ϵ of $H(X)$

$H_\delta(X^N)$ measures the *fewest* number of bits needed to uniformly code *smallest* set of N-outcome sequence S_δ with $P(x \in S_\delta) \geq 1 - \delta$.

Extended Ensembles (Review)

Instead of coding single outcomes, we now consider coding blocks and sequences of blocks

Example (Coin Flips):

$$
\begin{aligned}
\texttt{hhhhthhththh} &\rightarrow \texttt{hh hh th ht ht hh} && \text{(6 × 2 outcome blocks)}\\
&\rightarrow \texttt{hhh hth hth thh} && \text{(4 × 3 outcome blocks)}\\
&\rightarrow \texttt{hhhh thht hthh} && \text{(3 × 4 outcome blocks)}
\end{aligned}
$$

Extended Ensembles (Review)

Instead of coding single outcomes, we now consider coding blocks and sequences of blocks

Example (Coin Flips):

$$
\begin{aligned}
\texttt{hhhhthhththh} &\rightarrow \texttt{hh hh th ht ht hh} && \text{(6} \times \text{2 outcome blocks)} \\
&\rightarrow \texttt{hhh hth hth thh} && \text{(4} \times \text{3 outcome blocks)} \\
&\rightarrow \texttt{hhhh thht hthh} && \text{(3} \times \text{4 outcome blocks)}
\end{aligned}
$$

Extended Ensemble

The **extended ensemble** of blocks of size N is denoted X^N. Outcomes from X^N are denoted $\mathbf{x} = (x_1, x_2, \dots, x_N)$. The **probability** of $\mathbf{x}$ is defined to be $P(\mathbf{x}) = P(x_1)P(x_2)\dots P(x_N)$.

Extended Ensembles (Review)

Instead of coding single outcomes, we now consider coding blocks and sequences of blocks

Example (Coin Flips):

$$
\begin{aligned}
\texttt{hhhhthhththh} &\to \texttt{hh hh th ht ht hh} && (6 \times 2 \text{ outcome blocks}) \\
&\to \texttt{hhh hth hth thh} && (4 \times 3 \text{ outcome blocks}) \\
&\to \texttt{hhhh thht hthh} && (3 \times 4 \text{ outcome blocks})
\end{aligned}
$$

Extended Ensemble

The **extended ensemble** of blocks of size N is denoted X^N. Outcomes from X^N are denoted $\mathbf{x} = (x_1, x_2, \ldots, x_N)$. The **probability** of $\mathbf{x}$ is defined to be $P(\mathbf{x}) = P(x_1)P(x_2)\ldots P(x_N)$.

What is the entropy of X^N?

Extended Ensembles (Review)

Example: Bent Coin

Let X be an ensemble with outcomes $\mathcal{A}_X = \{\mathtt{h}, \mathtt{t}\}$ with $p_{\mathtt{h}} = 0.9$ and $p_{\mathtt{t}} = 0.1$.

Consider X^4 – i.e., 4 flips of the coin.

$$\mathcal{A}_{X^4} = \{\mathtt{hhhh}, \mathtt{hhht}, \mathtt{hhth}, \ldots, \mathtt{tttt}\}$$

Extended Ensembles (Review)

Example: Bent Coin

Let X be an ensemble with outcomes $\mathcal{A}_X = \{\mathtt{h}, \mathtt{t}\}$ with $p_{\mathtt{h}} = 0.9$ and $p_{\mathtt{t}} = 0.1$.

Consider X^4 – i.e., 4 flips of the coin.

$$\mathcal{A}_{X^4} = \{\mathtt{hhhh}, \mathtt{hhht}, \mathtt{hhth}, \ldots, \mathtt{tttt}\}$$

What is the probability of

- Four heads? $P(\mathtt{hhhh}) = (0.9)^4 \approx 0.656$
- Four tails? $P(\mathtt{tttt}) = (0.1)^4 = 0.0001$

Extended Ensembles (Review)

Example: Bent Coin

Let X be an ensemble with outcomes $\mathcal{A}_X = \{\mathtt{h}, \mathtt{t}\}$ with $p_\mathtt{h} = 0.9$ and $p_\mathtt{t} = 0.1$.

Consider X^4 – i.e., 4 flips of the coin.

$$\mathcal{A}_{X^4} = \{\mathtt{hhhh}, \mathtt{hhht}, \mathtt{hhth}, \dots, \mathtt{tttt}\}$$

What is the probability of

- Four heads? $P(\mathtt{hhhh}) = (0.9)^4 \approx 0.656$
- Four tails? $P(\mathtt{tttt}) = (0.1)^4 = 0.0001$

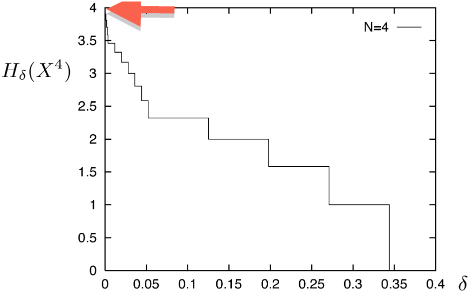
What is the entropy and raw bit content of X^4?

- The outcome set size is $|\mathcal{A}_{X^4}| = |\{0000, 0001, 0010, \dots, 1111\}| = 16$
- Raw bit content: $H_0(X^4) = \log_2 |\mathcal{A}_{X^4}| = 4$
- Entropy: $H(X^4) = 4H(X) = 4.\,(-0.9 \log_2 0.9 - 0.1 \log_2 0.1) = 1.88$

Essential Bit Content of Extended Ensembles

What if we use a lossy uniform code on the extended ensemble?

$\mathbf{x}$	$P(\mathbf{x})$	$\mathbf{x}$	$P(\mathbf{x})$
hhhh	0.656	thht	0.008
hhht	0.073	thth	0.008
hhth	0.073	tthh	0.008
hthh	0.073	httt	0.001
thhh	0.073	thtt	0.001
htht	0.008	ttht	0.001
htth	0.008	ttth	0.001
hhtt	0.008	tttt	0.000

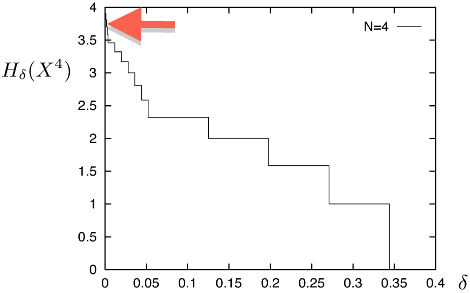


$$\delta = 0 \text{ gives } H_\delta\left(X^4\right) = \log_2 16 = 4$$

Essential Bit Content of Extended Ensembles

What if we use a lossy uniform code on the extended ensemble?

$\mathbf{x}$	$P(\mathbf{x})$	$\mathbf{x}$	$P(\mathbf{x})$
hhhh	0.656	thht	0.008
hhht	0.073	thth	0.008
hhth	0.073	tthh	0.008
hthh	0.073	httt	0.001
thhh	0.073	thtt	0.001
htht	0.008	ttht	0.001
htth	0.008	ttth	0.001
hhtt	0.008		

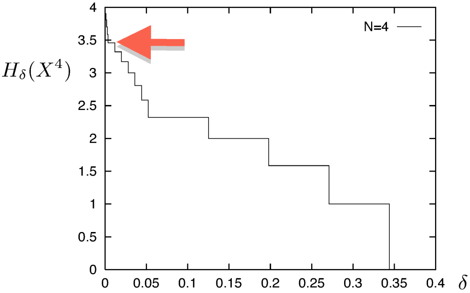


$$\delta = 0.0001 \text{ gives } H_\delta\left(X^4\right) = \log_2 15 = 3.91$$

Essential Bit Content of Extended Ensembles

What if we use a lossy uniform code on the extended ensemble?

x	$P(\mathbf{x})$	**x**	$P(\mathbf{x})$
`hhhh`	0.656	`thht`	0.008
`hhht`	0.073	`thth`	0.008
`hhth`	0.073	`tthh`	0.008
`hthh`	0.073		
`thhh`	0.073		
`htht`	0.008		
`htth`	0.008		
`hhtt`	0.008		

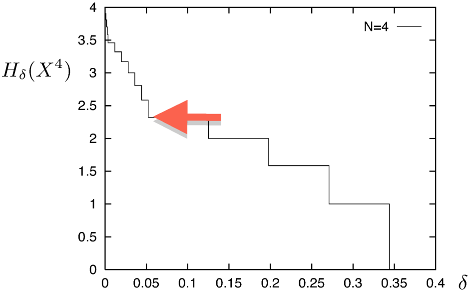


$$\delta = 0.005 \text{ gives } H_\delta\left(X^4\right) = \log_2 11 = 3.46$$

Essential Bit Content of Extended Ensembles

What if we use a lossy uniform code on the extended ensemble?

x	$P(\mathbf{x})$	**x**	$P(\mathbf{x})$
`hhhh`	0.656		
`hhht`	0.073		
`hhth`	0.073		
`hthh`	0.073		
`thhh`	0.073		

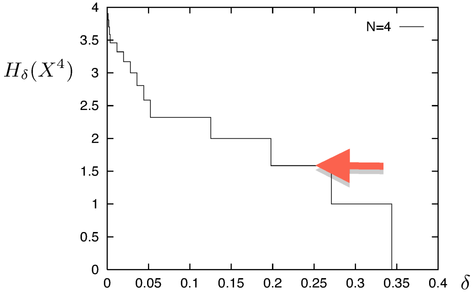


$$\delta = 0.05 \text{ gives } H_\delta\left(X^4\right) = \log_2 5 = 2.32$$

Essential Bit Content of Extended Ensembles

What if we use a lossy uniform code on the extended ensemble?

$\mathbf{x}$	$P(\mathbf{x})$	$\mathbf{x}$	$P(\mathbf{x})$
hhhh	0.656		
hhht	0.073		
hhth	0.073		

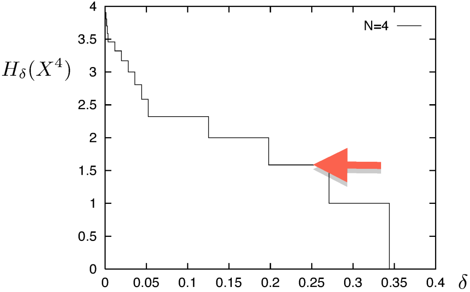


$$\delta = 0.25 \text{ gives } H_\delta\left(X^4\right) = \log_2 3 = 1.6$$

Essential Bit Content of Extended Ensembles

What if we use a lossy uniform code on the extended ensemble?

x	$P(\mathbf{x})$	**x**	$P(\mathbf{x})$
hhhh	0.656		
hhht	0.073		
hhth	0.073		

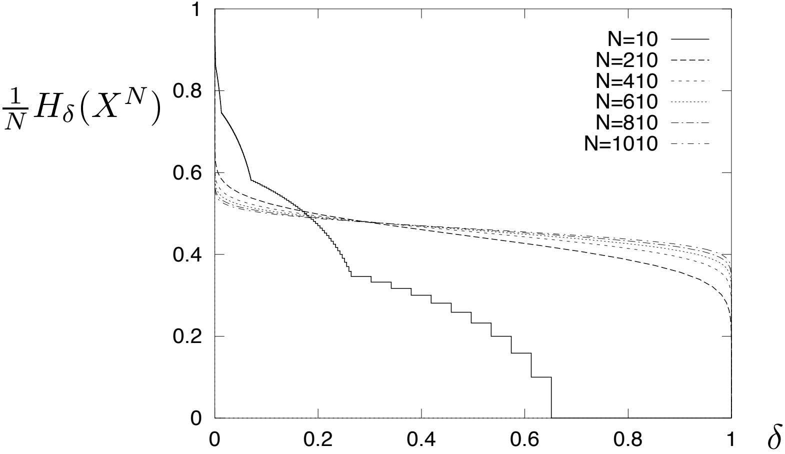


$$\delta = 0.25 \text{ gives } H_\delta\left(X^4\right) = \log_2 3 = 1.6$$

Unlike entropy, $H_\delta(X^4) \neq 4H_\delta(X) = 0$

Essential Bit Content of Extended Ensembles

What happens as *N* increases?



Recall that the entropy of a single coin flip with $p_{\mathtt{h}} = 0.9$ is $H(X) \approx 0.47$

Essential Bit Content of Extended Ensembles

Some Intuition

Why does the curve flatten for large N?

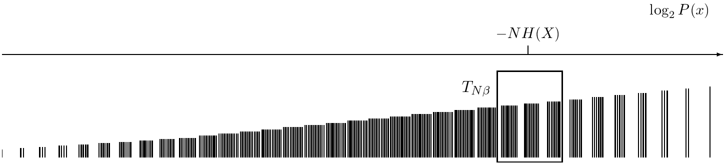
Recall that for $N = 1000$ e.g., sequences with 900 heads are considered typical

Such sequences occupy most of the probability mass, and are roughly equally likely

As we increase δ, we will quickly encounter these sequences, and make small, roughly equal sized changes to $|S_\delta|$

Typical Sets and the AEP (Review)

x	$\log_2(P(\mathbf{x}))$
...1...................1.....1....1.1.......1........1...........1......................1......11...	−50.1
......................1.....1.....1.......1....1.........1.....................................1....	−37.3
........1....1..1...1....11..1.1.........11.........................1...1.1..1...1................1.	−65.9
1.1...1................1........................11.1..1...........................1.....1..1.11.....	−56.4
...11...........1...1.....1.1......1..........1....1...1.....1............1.........................	−53.2
..............1......1.........1.1.......1..........1............1...1......................1.......	−43.7
.....1........1.......1...1............1............1...........1......1..11........................	−46.8
.....1..1..1...............111...................1...............1.........1.1...1...1.............1	−56.4
.........1..........1.....1......1..........1....1..1...	−37.3
......1........................1..............1.....1..1.1.1..1...................................1.	−43.7
1.......................1..........1...1...................1....1....1........1..11..1.1...1........	−56.4
...........11.1.........1................1......1.....................1.............................	−37.3
.1..........1...1.1.............1.......11...........1.1...1..............1.............11..........	−56.4
......1...1..1.....1..11.1.1.1...1.....................1............1.............1..1..............	−59.5
............11.1......1....1..1............................1.......1..............1.......1.........	−46.8
....	−15.2
1111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111	−332.1



Typical Sets and the AEP (Review)

Typical Set

For "closeness" $\beta > 0$ the typical set $T_{N\beta}$ for X^N is

$$T_{N\beta} \stackrel{\text{def}}{=} \left\{ \mathbf{x} : \left| -\frac{1}{N} \log_2 P(\mathbf{x}) - H(X) \right| < \beta \right\}$$

The name "typical" is used since $\mathbf{x} \in T_{N\beta}$ will have roughly $p_1 N$ occurences of symbol a_1, $p_2 N$ of a_2, ..., $p_K N$ of a_K.

Typical Sets and the AEP (Review)

Typical Set

For "closeness" $\beta > 0$ the typical set $T_{N\beta}$ for X^N is

$$T_{N\beta} \stackrel{\text{def}}{=} \left\{ \mathbf{x} : \left| -\frac{1}{N} \log_2 P(\mathbf{x}) - H(X) \right| < \beta \right\}$$

The name "typical" is used since $\mathbf{x} \in T_{N\beta}$ will have roughly $p_1 N$ occurences of symbol a_1, $p_2 N$ of a_2, ..., $p_K N$ of a_K.

Asymptotic Equipartition Property (Informal)

As $N \to \infty$, $\log_2 P(x_1, \ldots, x_N)$ is close to $-NH(X)$ with high probability.

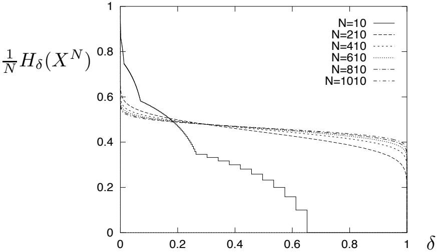
For large block sizes "almost all sequences are typical" (i.e., in $T_{N\beta}$).

The Source Coding Theorem

The Source Coding Theorem

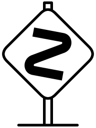
Let X be an ensemble with entropy $H = H(X)$ bits. Given $\epsilon > 0$ and $0 < \delta < 1$, there exists a positive integer N_0 such that for all $N > N_0$

$$\left| \frac{1}{N} H_\delta \left(X^N \right) - H \right| < \epsilon.$$



- Given a tiny probability of error δ, the average bits per outcome can be made as close to H as required.
- Even if we allow a large probability of error, we **cannot** compress more than H bits per outcome for large sequences.

Warning: proof ahead



I don't expect you to reproduce the following proof

- I present it as it sheds some light on why the result is true
- And it is a remarkable and fundamental result
- You are expected to **understand** and **be able to apply** the theorem

Proof of the SCT

The absolute value of a difference being bounded (e.g., $|x - y| \leq \epsilon$) says two things:

1. When $x - y$ is positive, it says $x - y < \epsilon$ which means $x < y + \epsilon$
2. When $x - y$ is negative, it says $-(x - y) < \epsilon$ which means $x < y - \epsilon$

$$|x - y| < \epsilon \quad \text{is equivalent to} \quad y - \epsilon < x < y + \epsilon$$

Proof of the SCT

The absolute value of a difference being bounded (e.g., $|x - y| \leq \epsilon$) says two things:

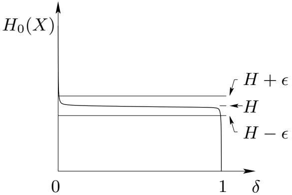
1. When $x - y$ is positive, it says $x - y < \epsilon$ which means $x < y + \epsilon$
2. When $x - y$ is negative, it says $-(x - y) < \epsilon$ which means $x < y - \epsilon$

$$|x - y| < \epsilon \quad \text{is equivalent to} \quad y - \epsilon < x < y + \epsilon$$

Using this, we break down the claim of the SCT into two parts: showing that for any ϵ and δ we can find N large enough so that:

Part 1: $\frac{1}{N} H_\delta(X^N) < H + \epsilon$

Part 2: $\frac{1}{N} H_\delta(X^N) > H - \epsilon$



Proof the SCT

Idea

Proof Idea: As N increases

- $T_{N\beta}$ has $\sim 2^{NH(X)}$ elements
- almost all $\mathbf{x}$ are in $T_{N\beta}$
- S_δ and $T_{N\beta}$ increasingly overlap
- so $\log_2 |S_\delta| \sim NH$

Basically, we look to encode all typical sequences uniformly, and relate that to the essential bit content

Proof of the SCT (Part 1)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N} H_\delta(X^N) < H(X) + \epsilon$.

Proof of the SCT (Part 1)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N} H_\delta(X^N) < H(X) + \epsilon$.

Recall (see Lecture 10) for the *typical set* $T_{N\beta}$ we have for any N, β that

$$|T_{N\beta}| \leq 2^{N(H(X)+\beta)} \tag{1}$$

and, by the AEP, for any β as $N \to \infty$ we have $P(x \in T_{N\beta}) \to 1$.
So for any $\delta > 0$ we can always find an N such that $P(x \in T_{N\beta}) \geq 1 - \delta$.

Proof of the SCT (Part 1)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N} H_\delta(X^N) < H(X) + \epsilon$.

Recall (see Lecture 10) for the *typical set* $T_{N\beta}$ we have for any N, β that

$$|T_{N\beta}| \leq 2^{N(H(X)+\beta)} \tag{1}$$

and, by the AEP, for any β as $N \to \infty$ we have $P(x \in T_{N\beta}) \to 1$.
So for any $\delta > 0$ we can always find an N such that $P(x \in T_{N\beta}) \geq 1 - \delta$.

Now recall the definition of the *smallest δ-sufficient subset S_δ*: it is the smallest subset of outcomes such that $P(x \in S_\delta) \geq 1 - \delta$ so $|S_\delta| \leq |T_{N\beta}|$.

Proof of the SCT (Part 1)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N} H_\delta(X^N) < H(X) + \epsilon$.

Recall (see Lecture 10) for the *typical set* $T_{N\beta}$ we have for any N, β that

$$|T_{N\beta}| \leq 2^{N(H(X)+\beta)} \tag{1}$$

and, by the AEP, for any β as $N \to \infty$ we have $P(x \in T_{N\beta}) \to 1$.
So for any $\delta > 0$ we can always find an N such that $P(x \in T_{N\beta}) \geq 1 - \delta$.

Now recall the definition of the *smallest δ-sufficient subset* S_δ: it is the smallest subset of outcomes such that $P(x \in S_\delta) \geq 1 - \delta$ so $|S_\delta| \leq |T_{N\beta}|$.

So, given any δ and β we can find an N large enough so that, by (1)

$$|S_\delta| \leq |T_{N\beta}| \leq 2^{N(H(X)+\beta)}$$

Proof of the SCT (Part 1)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N}H_\delta(X^N) < H(X) + \epsilon$.

Recall (see Lecture 10) for the *typical set* $T_{N\beta}$ we have for any N, β that

$$|T_{N\beta}| \leq 2^{N(H(X)+\beta)} \tag{1}$$

and, by the AEP, for any β as $N \to \infty$ we have $P(x \in T_{N\beta}) \to 1$.
So for any $\delta > 0$ we can always find an N such that $P(x \in T_{N\beta}) \geq 1 - \delta$.

Now recall the definition of the *smallest δ-sufficient subset* S_δ: it is the smallest subset of outcomes such that $P(x \in S_\delta) \geq 1 - \delta$ so $|S_\delta| \leq |T_{N\beta}|$.

So, given any δ and β we can find an N large enough so that, by (1)

$$\log_2 |S_\delta| \leq \log_2 |T_{N\beta}| \leq N(H(X) + \beta)$$

Proof of the SCT (Part 1)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N}H_\delta(X^N) < H(X) + \epsilon$.

Recall (see Lecture 10) for the *typical set* $T_{N\beta}$ we have for any N, β that

$$|T_{N\beta}| \leq 2^{N(H(X)+\beta)} \tag{1}$$

and, by the AEP, for any β as $N \to \infty$ we have $P(x \in T_{N\beta}) \to 1$.
So for any $\delta > 0$ we can always find an N such that $P(x \in T_{N\beta}) \geq 1 - \delta$.

Now recall the definition of the *smallest δ-sufficient subset* S_δ: it is the smallest subset of outcomes such that $P(x \in S_\delta) \geq 1 - \delta$ so $|S_\delta| \leq |T_{N\beta}|$.

So, given any δ and β we can find an N large enough so that, by (1)

$$H_\delta(X^N) = \log_2 |S_\delta| \leq \log_2 |T_{N\beta}| \leq N(H(X) + \beta)$$

Setting $\beta = \epsilon$ and dividing through by N gives result.

Proof of the SCT (Part 2)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N}H_\delta(X^N) > H(X) - \epsilon$.

Suppose this was not the case – that is, for every N we have

$$\frac{1}{N}H_\delta(X^N) \leq H(X) - \epsilon \iff |S_\delta| \leq 2^{N(H(X)-\epsilon)}$$

Proof of the SCT (Part 2)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N}H_\delta(X^N) > H(X) - \epsilon$.

Suppose this was not the case – that is, for every N we have

$$\frac{1}{N}H_\delta(X^N) \leq H(X) - \epsilon \iff |S_\delta| \leq 2^{N(H(X)-\epsilon)}$$

Let's look at what this says about $P(x \in S_\delta)$ by writing

$$\begin{aligned} P(x \in S_\delta) &= P(x \in S_\delta \cap T_{N\beta}) + P(x \in S_\delta \cap \overline{T_{N\beta}}) \\ &\leq |S_\delta| 2^{-N(H-\beta)} + P(x \in \overline{T_{N\beta}}) \end{aligned}$$

since every $x \in T_{N\beta}$ has $P(x) \leq 2^{-N(H-\beta)}$ and $S_\delta \cap \overline{T_{N\beta}} \subset \overline{T_{N\beta}}$.

Proof of the SCT (Part 2)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N}H_\delta(X^N) > H(X) - \epsilon$.

Suppose this was not the case – that is, for every N we have

$$\frac{1}{N}H_\delta(X^N) \le H(X) - \epsilon \iff |S_\delta| \le 2^{N(H(X)-\epsilon)}$$

Let's look at what this says about $P(x \in S_\delta)$ by writing

$$\begin{aligned} P(x \in S_\delta) &= P(x \in S_\delta \cap T_{N\beta}) + P(x \in S_\delta \cap \overline{T_{N\beta}}) \\ &\le |S_\delta| 2^{-N(H-\beta)} + P(x \in \overline{T_{N\beta}}) \end{aligned}$$

since every $x \in T_{N\beta}$ has $P(x) \le 2^{-N(H-\beta)}$ and $S_\delta \cap \overline{T_{N\beta}} \subset \overline{T_{N\beta}}$.

So

$$P(x \in S_\delta) \le 2^{N(H-\epsilon)} 2^{-N(H-\beta)} + P(x \in \overline{T_{N\beta}})$$

Proof of the SCT (Part 2)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N}H_\delta(X^N) > H(X) - \epsilon$.

Suppose this was not the case – that is, for every N we have

$$\frac{1}{N}H_\delta(X^N) \leq H(X) - \epsilon \iff |S_\delta| \leq 2^{N(H(X)-\epsilon)}$$

Let's look at what this says about $P(x \in S_\delta)$ by writing

$$\begin{aligned} P(x \in S_\delta) &= P(x \in S_\delta \cap T_{N\beta}) + P(x \in S_\delta \cap \overline{T_{N\beta}}) \\ &\leq |S_\delta| 2^{-N(H-\beta)} + P(x \in \overline{T_{N\beta}}) \end{aligned}$$

since every $x \in T_{N\beta}$ has $P(x) \leq 2^{-N(H-\beta)}$ and $S_\delta \cap \overline{T_{N\beta}} \subset \overline{T_{N\beta}}$.

So

$$P(x \in S_\delta) \leq 2^{-N(H-H+\epsilon-\beta)} + P(x \in \overline{T_{N\beta}})$$

Proof of the SCT (Part 2)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N}H_\delta(X^N) > H(X) - \epsilon$.

Suppose this was not the case – that is, for every N we have

$$\frac{1}{N}H_\delta(X^N) \leq H(X) - \epsilon \iff |S_\delta| \leq 2^{N(H(X)-\epsilon)}$$

Let's look at what this says about $P(x \in S_\delta)$ by writing

$$\begin{aligned} P(x \in S_\delta) &= P(x \in S_\delta \cap T_{N\beta}) + P(x \in S_\delta \cap \overline{T_{N\beta}}) \\ &\leq |S_\delta| 2^{-N(H-\beta)} + P(x \in \overline{T_{N\beta}}) \end{aligned}$$

since every $x \in T_{N\beta}$ has $P(x) \leq 2^{-N(H-\beta)}$ and $S_\delta \cap \overline{T_{N\beta}} \subset \overline{T_{N\beta}}$.

So

$$P(x \in S_\delta) \leq 2^{-N(\epsilon-\beta)} + P(x \in \overline{T_{N\beta}}) \to 0 \text{ as } N \to \infty$$

since $P(x \in T_{N\beta}) \to 1$.

Proof of the SCT (Part 2)

For $\epsilon > 0$ and $\delta > 0$, want N large enough so $\frac{1}{N} H_\delta(X^N) > H(X) - \epsilon$.

Suppose this was not the case – that is, for every N we have

$$\frac{1}{N} H_\delta(X^N) \leq H(X) - \epsilon \iff |S_\delta| \leq 2^{N(H(X)-\epsilon)}$$

Let's look at what this says about $P(x \in S_\delta)$ by writing

$$\begin{aligned} P(x \in S_\delta) &= P(x \in S_\delta \cap T_{N\beta}) + P(x \in S_\delta \cap \overline{T_{N\beta}}) \\ &\leq |S_\delta| 2^{-N(H-\beta)} + P(x \in \overline{T_{N\beta}}) \end{aligned}$$

since every $x \in T_{N\beta}$ has $P(x) \leq 2^{-N(H-\beta)}$ and $S_\delta \cap \overline{T_{N\beta}} \subset \overline{T_{N\beta}}$.

So

$$P(x \in S_\delta) \leq 2^{-N(\epsilon-\beta)} + P(x \in \overline{T_{N\beta}}) \rightarrow 0 \text{ as } N \rightarrow \infty$$

since $P(x \in T_{N\beta}) \rightarrow 1$. But $P(x \in S_\delta) \geq 1 - \delta$, by defn. Contradiction

Interpretation of the SCT

The Source Coding Theorem

Let X be an ensemble with entropy $H = H(X)$ bits. Given $\epsilon > 0$ and $0 < \delta < 1$, there exists a positive integer N_0 such that for all $N > N_0$

$$\left| \frac{1}{N} H_\delta \left(X^N \right) - H \right| < \epsilon.$$

If you want to uniformly code blocks of N symbols drawn i.i.d. from X

- If you use more than $NH(X)$ bits per block you can do so without almost no loss of information as $N \to \infty$
- If you use less than $NH(X)$ bits per block you will almost certainly lose information as $N \to \infty$

Interpretation of the SCT

The Source Coding Theorem

Let X be an ensemble with entropy $H = H(X)$ bits. Given $\epsilon > 0$ and $0 < \delta < 1$, there exists a positive integer N_0 such that for all $N > N_0$

$$\left| \frac{1}{N} H_\delta \left(X^N \right) - H \right| < \epsilon.$$

Making the error probability $\delta \approx 1$ doesn't really help

- We're still "stuck with" coding the typical sequences

Assumes we deal with X^N

- If outcomes are dependent, entropy $H(X)$ need not be the limit
- We won't look at such extensions

Implications of SCT

How practical is it to perform coding inspired by the SCT?

Implications of SCT

How practical is it to perform coding inspired by the SCT?

Not very!

- Theorem might require huge block sizes N_0
- We'd need lookup tables of size $|S_\delta(X^{N_0})| \sim 2^{N_0 \cdot H(X)}$

Implications of SCT

How practical is it to perform coding inspired by the SCT?

Not very!

- Theorem might require huge block sizes N_0
- We'd need lookup tables of size $|S_\delta(X^{N_0})| \sim 2^{N_0 \cdot H(X)}$

Can we design more practical compression algorithms?

- And will the entropy still feature with the fundamental limit?

Next time

We move towards more practical compression ideas

Prefix and **Uniquely Decodeable** variable-length codes

The **Kraft Inequality**