Today: Data Compression

Some limits to compressibility.

First: A review of the last two lectures.

First few lectures: Studied properties of:
- entropy, conditional entropy, mutual information, relative entropy
- One consequence: if variable $X$ has large conditional entropy given $Y$ then any attempt to determine $X$ from $Y$ has large error (Fano's lemma)

Lecture 4: restrict to variables $X, Y$

An sequence of i.i.d. variables
distrbuted according to \( X \)

when \( \overline{X} = (X_1, \ldots , X_n) \) then

w.p. \( \geq 1 - \delta \)

\[ \Pr_{\overline{X}} \left[ p(\overline{X}) \in \left[ 2^{-(H(x) + \varepsilon) n}, 2^{-(H(x) - \varepsilon) n} \right] \right] \geq 1 - \delta \]

(Typical set) \( A_{e}^{(n)} = \left\{ \overline{x} \in \mathbb{Z}^n \mid p(\overline{x}) \in \left[ \right] \right\} \)

\( A_{e}^{(n)} \) has size \( \approx 2^{H(x) \cdot n} \)

with each \( A_l \) having prob. \( \approx 2^{H(x) \cdot n} \).

(flat dist. on small set)

\[ \text{[didn't say but } \delta \approx \exp(-\varepsilon^2 n) \] \]

leads to formal proof that

- \( X \) can be compressed into \( \sim H(x) \cdot n \) bits in expectation

- \( X \) can't be compressed to much less than \( H(x) \cdot n \) bits
LECTURE 5: $X = \text{stochastic process}$

(Leap, time-invariant, irreducible, aperiodic, Markov chain)

Stochastic \$\Rightarrow$ Stationary \$\Rightarrow$ Time Inv. Markov Chain in stationary distribution

Entropy rate of $(\cdot): \mathcal{H} (X_t | X_{t-1})

= \sum_{i,j} \pi_{ij} \log \frac{1}{\pi_{ij}}$

where $\pi$ = transition probability matrix

$\pi = \text{Stationary dist.}$

$(\Delta \rho)$: \[P_r \left[ \rho (\bar{x}) \in \left[ 2^{-(\mathcal{H} (\bar{x}) + \varepsilon)n}, 2^{-(\mathcal{H} (\bar{x}) + \varepsilon)n} \right] \right] \geq 1 - \delta\]

$\delta \approx \exp (-\varepsilon^2 n)$
Finally

Hidden Markov chains

\[ X_1 \rightarrow X_2 \rightarrow \cdots \rightarrow X_n \]

\[ \downarrow \quad \quad \quad \quad \quad \downarrow \]

\[ Y_1 \quad Y_2 \quad \cdots \quad Y_n \]

Entropy rate

\[ H(Y) = \lim_{n \to \infty} H(Y_n | Y_{n-1}, \ldots, Y_1) \]

Exists, but not easy to compute

Now to today's lecture.
Goal: to "compress" $X$ into a string of letters over $\mathcal{D}$. 

but ... .

1. What is compression?
2. What is the objective?

E.g. $X \in \mathcal{Z} \ldots N$?

$X = i$ w.p. $q_i$

Is $C(i) = 0 \neq i$ "compression"?

Trade off between "compression" & loss of information ... .

We wish to compress without losing (too much) information.
**Terminology**

- \( \mathcal{X} \) = space of \( X \)
- \( \mathcal{D} \) = alphabet for compression
- \( \mathcal{D}^* = \cup_{n \geq 0} \mathcal{D}^n \) = set of finite strings over \( \mathcal{D} \)
  
  for \( w = (w_1, \ldots, w_n) \in \mathcal{D}^n \), \( |w| = n \)

**Compression algorithm**

\[ C: \mathcal{X} \rightarrow \mathcal{D}^* \]

accompanied by \textit{Decompressor}

\[ \text{Dec}: \mathcal{D}^* \rightarrow \mathcal{X} \]

**Lossy compression**: \( \Pr_x \left[ \text{Dec}(C(x)) \neq x \right] \ll \text{small} \)

**Lossless compression**: \( \forall x \in \mathcal{X} \text{ Dec}(C(x)) = x \)
if $C$ s.t. $\exists$ Dec $\Rightarrow$

$(C, \text{Dec})$ lead to lossless coding

then $C$ is called $\text{Non-Singular}$

We will talk only about $\text{Non-Singular}$

(lossless coding) but many results extend to lossy coding as well.

(e.g., we mean limitations result of lossy)

Goal / Measure = ?

Expected Length: $\mathbb{E} \left[ |C(x)| \right]$

Minimize
Example \( S = \{ 1 \text{ w.p. } \frac{1}{2}, 2 \text{ w.p. } \frac{1}{4}, 3 \text{ w.p. } \frac{1}{8}, 4 \text{ w.p. } \frac{1}{8} \} \)

\[ D = \{0, 12\} \]

\underline{Code 1}:
\[ C(1) = 00, \quad C(2) = 01, \quad C(3) = 10, \quad C(4) = 11 \]

\[ \mathbb{E}\left[ |C(x)| \right] = 2 \]

\underline{Code 2}:
\[ C_2(1) = 00, \quad C_2(3) = 00, \quad C_2(2) = 1, \quad C_2(4) = 10 \]
\[ E \left[ 1_{C_2}(x) \right] = \frac{3}{4} \cdot 1 + \frac{1}{4} \cdot 2 = \frac{5}{4} \]

- But something unsatisfactory about second code?

Second code is not extensible.

Real goal to get a code for sequence \((X_1, \ldots, X_n)\) where \(X_i \sim X\) (say i.i.d.)

Code for \(X\) suggests extension code of \(X^n\)

\[ C(X_1, \ldots, X_n) = C(X_1), C(X_2), \ldots, C(X_n) \]

Code \(C_2\) above is non-singular

but extension is not!
(E.g. $C_2(11) = C_2(3)$)

& $C_2(113) = C_2(311)$

Motivates another definition

$C$ is uniquely decodable

if for all $n$, the $n$-extension of $C$ is non-singular.

Example of uniquely-decodable code

$C_3(1) = 0$ 1 w.p. $\frac{1}{2}$
$C_3(2) = 10$ 2 w.p. $\frac{1}{4}$
$C_3(3) = 110$ 3 w.p. $\frac{1}{8}$
$C_3(4) = 111$ 4 w.p. $\frac{1}{8}$

$E[\text{length}] = 1.75$

Another example

$C_4(1) = 0$  $C_4(2) = 01$  $C_4(3) = 011$  $C_4(4) = 111$
Proof of unique decodability of $C_2$.

Property of $C_3$

- For all $x, y \in S^2$, $C(x)$ is not a prefix of $C(y)$.

Motivates another definition:

$C$ is a prefix code if and only if:

For all $x, y \in S^2$, $C(x)$ is not a prefix of $C(y)$.

Claim: Prefix code is uniquely decodable.

Proof: Let $C(x_1, \ldots, x_n) = w_1 \ldots w_m$.

Then $x_i = y_i$ (since $C(x_i)$ must
be a prefix of $C(y)$ or vice versa).

But now by induction we have

$$C(x_1 \ldots x_n) = w_{k+1} \ldots w_m = C(y_1 \ldots y_{k-1})$$

where $C(x_1) = w_1 \ldots w_k = C(y_1)$

so $x_2 \ldots x_n = y_{k+1} \ldots y_{n-1}$ [i.e. $n=k$].

---

Proof of unique decodability of $C_4$

(well $C_4 = C_3$ in reverse)

Prefix codes ($C_3$) are nicer than non-prefix codes ($C_4$) since they can be decoded online.

Hence also called instantaneous
In the rest of the lecture we will study:

- **Limitations on Compressibility**
  - Kraft lower bound
  - McMillan lower bound
  - Entropy lower bound

Next lecture will match goals.

Main target for this lecture:

\[
\mathbb{E}_{x} \left[ |C(x)| \right] \geq \frac{H(x)}{\log D}
\]

for any prefix-free or uniquely decodable code \( C \).
Actually yields
\[ E \left[ |C(x)| \right] = K \]

then \[ K + O(\log K) \geq \frac{H(x)}{\log D} \]

for any non-singular code \( \text{[ Won't show this but will show } \)

Main tool towards Entropy bound

1. Kraft's Inequality

Theorem (Kraft): if \( C(i) \) has length \( l_i \)

for \( i = 1 \ldots N \) over \( D \)-ary alphabet

then \( \sum D^{-l_i} \leq 1 \), if \( C \) prefix code.
Proof: Consider D-ary tree with branches density various letters of D. Example with D = 2

for node at level \( i \), associate weight \( D^{-i} \). Note that the weight of the root = 1.

Weight of every node = Sum of uts. of children.
Now let's understand "C(i)'s, for every i mark node C(i).

e.g.

\[
C(0) \implies 1^0 = 1 \\
C(1) \implies 11^1 = 11 \\
C(2) \implies 110^2 = 110 \\
C(3) \implies 111^3 = 111
\]
Retain only the part of the tree that lies on the path from some \( C(i) \) to the root.

Reassign weights, leaving weights of \( C(i) \) as they were & putting in for every node weight = sum of weight of its children.

- On the one hand weights of nodes don’t go up

- On the other hand weight of root

\[
\sum_{i=1}^{n} t_i \leq 0
\]

Thus

\[
\sum_{i=1}^{N} t_i \leq 0 \leq 1
\]
McMillan's Bound

Theorem: if \( C \) is uniquely decodable
\[ |C(i)| = l_i \] then
\[ \sum_l \cdot l_i \leq 1 \]

Proof: (Optional. See cover 4 Thomas)

Entropy lower bound

Note: Kraft says nothing about probability.
To relate to Expected Decoding Length...
\[ \mathbb{E} \left[ |C(x)| \right] = \sum_{x \in X} p(x) \cdot l_i \]
But now let's write $C_i = -\frac{\log d_i}{\log D}$

$$E \left[ \log C(x) \right] = -\sum_{i=1}^{N} p_i \log \frac{d_i}{\log D}$$

But $\sum d_i \leq 1$ [Kraft]

Let $q_i = d_i^{-\epsilon_i}$

And let $\sum d_i^{-\epsilon_i} = 1 - q_0$

$$E \left[ \log C(x) \right] = -\sum_{i=1}^{N} p_i \log q_i$$

$$= H(X) + \frac{D(P \| Q)}{\log D} \geq H(X)$$
Conclude

**Theory**: if $C$ is prefix free or uniquely decodable then for $x \sim p$

$$
\mathbb{E}_{x} \left[ |C(x)| \right] \geq H(x)
$$

---

**Answer 1**: Not a fair question. Really need unique decodability.

**Answer 2**: Doesn't make much difference.
Lemma: if C is non-singular code of length $K$, then $\exists C'$ prefix free of 
expected length $K + 2 \lceil \sqrt{K} \rceil$.

(Can probably do better....)

Proof:

Given C, first produce $C'$,

such that $\forall i \ |C'_i(i)|$ is a multiple of $\lceil \sqrt{K} \rceil$. We have

$\mathbb{E} \left[ |C'(x)| \right] \leq \mathbb{E} \left[ |C(x)| \right] + \lceil \sqrt{K} \rceil$

(since no string extends by more than $\lceil \sqrt{K} \rceil$.

Now produce $C_2$ where

$|C_2(i)|$ is a multiple of $\lceil \sqrt{K} \rceil + 1$
as follows.

Suppose \( C_1(i) = \begin{bmatrix} w_1 & w_2 & \cdots & w_c \end{bmatrix} \)

when \( |w_j| = \sqrt{K} \)

then \( C_2(i) = \begin{bmatrix} w_1 & 0 & w_2 & 0 & \cdots & w_c \end{bmatrix} \)

\[ \uparrow \quad \gamma \quad \gamma \quad \gamma \quad \gamma \quad \gamma \quad \gamma \]

all zeroes ONE

Claim: \( C_2 \) is prefix free (why?)

\[ E \left[ |C_2(X)| \right] \]

\[ \leq \frac{\lceil \sqrt{K} \rceil + 1}{\sqrt{K}} \cdot \frac{E \left[ |C_1(X)| \right]}{\sqrt{K}} \]

\[ = (\sqrt{K} + 1)(\sqrt{K} + 1) \leq K + O(\sqrt{K}) \]
Conclude

Essentially for any reasonable
encoding

\[ E \left[ 1 \cdot C(X) \right] \geq \frac{H(X)}{\log D} \]

But is this tight?

Will see in next lecture.