Stanford University

ENGR 76

Information Science and Engineering

Lecture 4: Source Coding III

Entropy and Fundamental Limits of Compression

Siddharth Chandak

Course Announcements and Reminders

Reminders

- Project 0 due tomorrow
- Project 1a will be released tomorrow

> In this project, you will:
>
> - Implement a compressor and decompressor using Huffman coding
> - Evaluate the strengths and weaknesses of Huffman coding on different data sources

Recap

Source and Huffman Coding

- Modeling a source as a random variable
- Expected Code Length
- Huffman Code
    - Optimal prefix-free code

Block Coding

- Applying Huffman codes on blocks of symbols
- Can lead to improvement in average number of bits per symbol
- Example:
  - Alphabet $\mathcal{X} = \{H, T\}$ with distribution $p(H) = 0.8$ and $p(T) = 0.2$
  - Huffman code: $H \mapsto 0$ and $T \mapsto 1$
  - Average number of bits per symbol is 1

Block Coding

- Assuming independent and identically distributed
- Let us work with blocks of size 2

X_1X_2	**Probability**	**Codeword (Huffman code)**
HH	$0.8 \times 0.8 = 0.64$	0
HT	$0.8 \times 0.2 = 0.16$	11
TH	$0.2 \times 0.8 = 0.16$	100
TT	$0.2 \times 0.2 = 0.04$	101

- Average number of bits per block $=$

$$\begin{aligned} \bar{\ell}_{block} &= 0.64 \times 1 + 0.16 \times 2 + 0.16 \times 3 + 0.04 \times 3 \\ &= 1.56 \end{aligned}$$

Average number of bits per symbol

- Average number of bits per block =

$$\begin{aligned}\bar{\ell}_{block} &= 0.64 \times 1 + 0.16 \times 2 + 0.16 \times 3 + 0.04 \times 3 \\ &= 1.56\end{aligned}$$

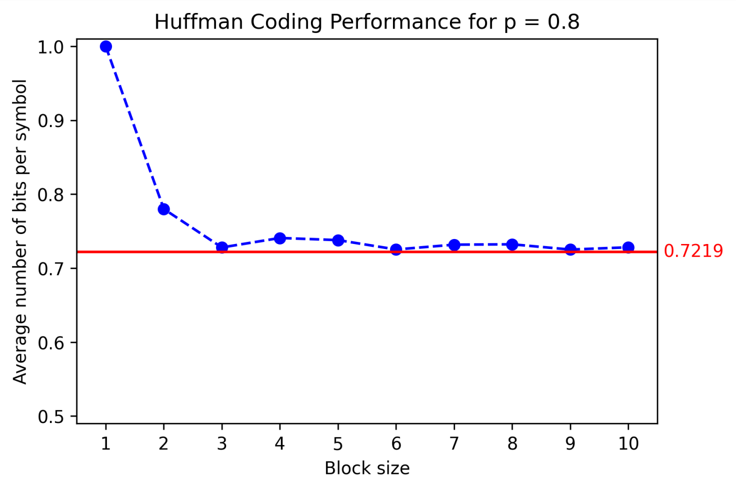
- Average number of bits per symbol = $\frac{\text{Average number of bits per block}}{\text{Block size}}$

$$\text{Average number of bits per symbol} = \frac{\bar{\ell}_{block}}{2} = 0.78$$

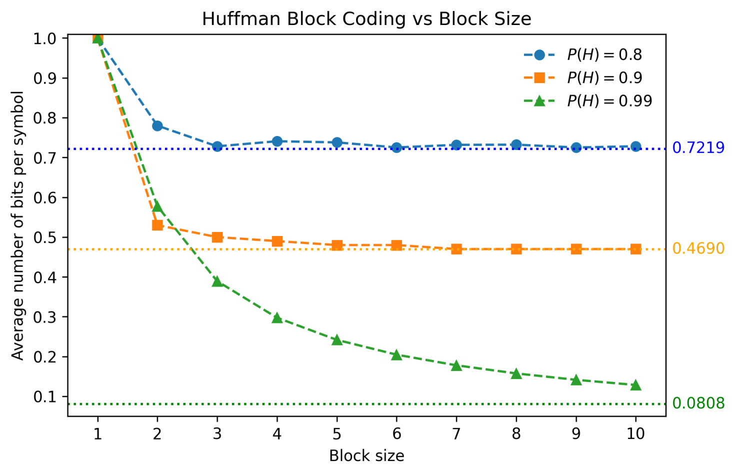
- Using blocks of size 2 we achieve 0.78 bits per symbol
    - Improvement over 1 bit per symbol achieved without block coding

Increasing block lengths

$H : 0.8$ and $T : 0.2$



Increasing block lengths



What are these lower limits of compression?

What is Information?

How informative an event is?

- Consider an example: Two events —
  - Event A: It is snowing in Stanford
  - Event B: The weather is nice and sunny at Stanford
- Happening of which event is more *informative*?

How informative an event is?

- Less likely events are more informative because they are more surprising!

Surprise

- Let us define a surprise function $S(\cdot)$ which captures the amount of surprise in an event A
- $S(\cdot)$ is a function of the probability of the event
- Denote the function as $S(p)$ where $p = P(A)$
- Properties?
    - How does it vary with p?

Surprise

Properties of Surprise function $S(p)$:

- $S(p)$ decreases with increasing p
- $S(p)$ is a continuous function of p
- Surprise of two independent events A and B happening?
    - You win two independent lotteries...

Surprise

Properties of Surprise function $S(p)$:

- $S(p)$ decreases with increasing p
- $S(p)$ is a continuous function of p
- Surprise of independent events A and B happening is sum of individual suprises:

$$S(P(A,B)) = S(P(A)P(B)) = S(P(A)) + S(P(B)),$$

i.e., $S(pq) = S(p) + S(q)$ for probabilities p and q

Surprise Function

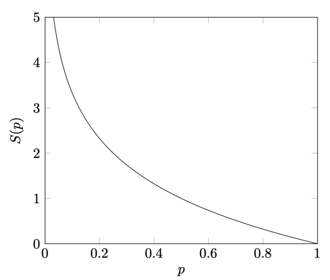
Only function which satisfies all three above[1]:

$$\boxed{S(p) = \log_2\left(\frac{1}{p}\right)}$$

[1] logarithm with any base works, we work with base 2 throughout the course

Surprise Function

$$\boxed{\boldsymbol{S(p) = \log_2\left(\frac{1}{p}\right)}}$$



- $S(p) \geq 0$ with $S(1) = 0$
- $S(p) \to \infty$ as $p \to 0$
- For probabilities p and q,

$$\begin{aligned} S(pq) &= \log_2\left(\frac{1}{pq}\right) = \log_2\left(\frac{1}{p} \times \frac{1}{q}\right) \\ &= \log_2\left(\frac{1}{p}\right) + \log_2\left(\frac{1}{q}\right) = S(p) + S(q) \end{aligned}$$

Entropy

Entropy

- Entropy or information content of a random variable
- Average surprise of the random variable X
  - For $x \in \mathcal{X}$, event $X = x$ happens with probability $p(x)$
  - Surprise associated with event $X = x$ is $S(p(x)) = \log_2(1/p(x))$
- **Entropy:**

$$\boxed{\boldsymbol{H(X) = \sum_{x \in \mathcal{X}} p(x) S(p(x)) = \sum_{x \in \mathcal{X}} p(x) \log_2 \left(\frac{1}{p(x)}\right)}}$$

- Depends only on the probability values (and not on the symbols)

Example (from last class)

Probability distribution of random variable X:

- A: $1/2$
- B: $1/4$
- C: $1/8$
- D: $1/8$

$$H(X) = \sum_{x \in \mathcal{X}} p(x) \log_2 \left(\frac{1}{p(x)} \right)$$

Example (from last class)

Probability distribution of random variable X:

- A: $1/2$
- B: $1/4$
- C: $1/8$
- D: $1/8$

$$\begin{aligned}
H(X) &= \sum_{x \in \mathcal{X}} p(x) \log_2 \left(\frac{1}{p(x)} \right) \\
&= \frac{1}{2} \log_2(2) + \frac{1}{4} \log_2(4) + \frac{1}{8} \log_2(8) + \frac{1}{8} \log_2(8) \\
&= \frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{8} \times 3 + \frac{1}{8} \times 3 \\
&= 1.75
\end{aligned}$$

Basic Properties

- $H(X) \geq 0$
- For random variable X with uniform distribution over $\mathcal{X}$ with M symbols, $H(X) = ?$
  - Uniform distribution: all outcomes are equally likely
  - $p(x) = 1/M$ for all $x \in \mathcal{X}$

Basic Properties

- $H(X) \geq 0$
- For random variable X with uniform distribution over $\mathcal{X}$ with M symbols:

$$\begin{aligned} H(X) &= \sum_{x \in \mathcal{X}} p(x) \log_2 \left(\frac{1}{p(x)} \right) \\ &= M \times \frac{1}{M} \log_2(M) \\ &= \log_2(M) \end{aligned}$$

An Important Fact

Fact

For any random variable X with alphabet $\mathcal{X}$ with M symbols,

$$0 \leq H(X) \leq \log_2(M).$$

- For all random variables over alphabet $\mathcal{X}$, the uniformly distributed r.v. has the highest entropy
- *Entropy is a measure of randomness or uncertainty*

Bounds on entropy

- For any random variable X with alphabet $\mathcal{X}$ with M symbols,

$$0 \leq H(X) \leq \log_2(M).$$

  - $H(X) = 0$ if and only if there exists some symbol $x \in \mathcal{X}$ such that $P(X = x) = 1$
  - $H(X) = \log_2(M)$ for an alphabet of size M if and only if X is uniformly distributed

Entropy and Compression: Intuition

- Entropy is a measure of randomness or uncertainty
- *Intuition from second lecture:* Skewed distributions allow for more compression
  - more frequent symbols can be assigned shorter codewords
- *An informal connection*:

$$
\begin{aligned}
\text{Lower entropy} &\iff \text{Lower randomness} \\
&\iff \text{More skewed distribution} \\
&\iff \text{More compressibility} \\
&\iff \text{Requires less bits to represent on average}
\end{aligned}
$$

Entropy and Huffman Codes

Performance of Huffman Algorithm

Fact

For any source distribution, the Huffman code is the optimal prefix-free code, i.e., has the smallest $\bar{\ell}$ among all prefix-free codes. Moreover, the expected length of Huffman code for source X satisfies:

$$H(X) \leq \bar{\ell} \leq H(X) + 1.$$

- Huffman code is optimal:
  - No prefix-free code can have expected code length lower than the entropy
  - Lower bound for all prefix-free codes

Example I (from last class)

Symbol	Probability	Codeword (Huffman Code)
A	$\frac{1}{2}$	0
B	$\frac{1}{4}$	10
C	$\frac{1}{8}$	110
D	$\frac{1}{8}$	111

$$\bar{\ell} = 1.75$$

$$H(X) = 1.75$$

In this case, $H(X) = \bar{\ell}$

When is $H(X) = \bar{\ell}$?

- The average code length of Huffman code is equal to the entropy ($H(X) = \bar{\ell}$) if and only if the source X has a **dyadic distribution**
- Consider alphabet $\mathcal{X} = \{x_1, \ldots, x_M\}$. Then X has dyadic distribution if
$$P(X = x_i) = 2^{-k_i},$$
for some $k_i \in \{1, 2, 3, \ldots\}$

Fact

For a dyadic source X, the Huffman code satisfies $H(X) = \bar{\ell}$. Moreover, the length of codeword for symbol x is

$$\ell(x) = \log_2 \left(\frac{1}{p(x)} \right).$$

Example II

Symbol	Probability	Codeword (Huffman Code)
A	0.35	00
B	0.25	01
C	0.2	10
D	0.12	110
E	0.08	111

$$\bar{\ell} = 2.2$$

$$H(X) = 2.153$$

In this case, $H(X) < \bar{\ell}$

Example III

Symbol	Probability	Codeword
H	0.99	0
T	0.01	1

$$\bar{\ell} = 1$$

$$H(X) = 0.0808$$

How to reduce this gap between average # bits per symbol and entropy?

Block Coding!

Joint Entropy and Block Coding

Joint Entropy

- For two random variables X and Y with alphabets $\mathcal{X}$ and $\mathcal{Y}$
- Total information value of the two sources together
- Joint Entropy:

$$H(X,Y) = \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} P(X=x,Y=y) \log_2 \left(\frac{1}{P(X=x,Y=y)} \right)$$

- What if X and Y are independent?

Joint Entropy for Independent Random Variables

- If X and Y are independent, each have their own randomness and uncertainty
- Their joint entropy is sum of entropies

Theorem

For any pair of independent random variables X and Y,

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

- Simple proof; presented in lecture notes

General Result

Fact

For any two random variables X and Y,

$$H(X,Y) \leq H(X) + H(Y)$$

- When r.v.s are dependent, they share some information
- Total information in pair is lower than sum of information in individual r.v.s

Bounds on Block Coding

- **Assumption:** $X_1, X_2, X_3, \ldots$ are independent and identically distributed
- Suppose we apply Huffman algorithm on blocks of size $n = 2$
- Let average codeword length of Huffman code be $\bar{\ell}_{block,2}$
- Upper and lower bounds on $\bar{\ell}_{block,2}$?

Bounds on Block Coding

- Huffman code on blocks (X_1X_2)
- Bounds on Huffman code gives us:

$$H(X_1, X_2) \leq \bar{\ell}_{block,2} \leq H(X_1, X_2) + 1$$

- Independence of X_1 and X_2 gives us

$$2H(X_1) \leq \bar{\ell}_{block,2} \leq 2H(X_1) + 1$$

- What is average number of bits per symbol?

Bounds on Block Coding

- Average number of bits per symbol:

$$\text{Average number of bits per symbol} = \frac{\bar{\ell}_{block,2}}{2}$$

- Huffman code on blocks X_1X_2

$$H(X_1) \leq \frac{\bar{\ell}_{block,2}}{2} \leq H(X_1) + \frac{1}{2}$$

$$\implies H(X_1) \leq \text{Average number of bits per symbol} \leq H(X_1) + \frac{1}{2}$$

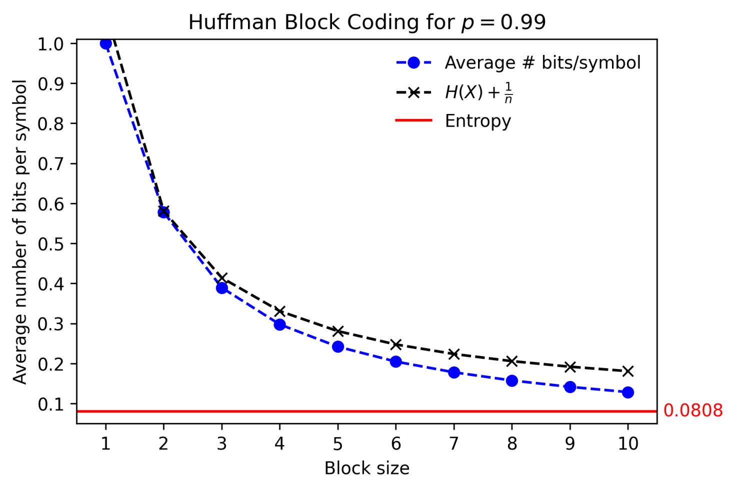
Bounds on block coding

- For blocks of size 2, we have shown that average number of bits per symbol lies between $H(X_1)$ and $H(X_1) + \frac{1}{2}$
- In general, for blocks of size n,

$$\boxed{H(X_1) \leq \text{Average number of bits per symbol} \leq H(X_1) + \frac{1}{n}}$$

- Gap between average number of bits per symbol and entropy gets smaller as blocks get larger!
- **Entropy is not just the lower limit but also achievable as we keep increasing block size**

Bounds on Block Coding



Shannon's Source Coding Theorem

Theorem

The entropy of a source equals the minimum number of bits per source symbol necessary on average to encode a sequence of **independent and identically distributed** symbols from that source. In general, this may require the use of block coding, where blocks of symbols are encoded together.

Lossless Compression: Summary of Lectures 2-4

- **Entropy:**
  - A measure of the information content of a source
  - Not just abstract: lower limit of compression
- **Huffman Coding:**
  - Efficient algorithm to obtain **optimal** prefix-free code for a source
    - Prefix-free code: allows for instantaneous decoding using a simple algorithm
  - Using block coding, achieves entropy for i.i.d. sequences
- **Exploiting Dependence:**
  - Brief discussion in next class

Lossless Compression: Beyond this course

- **Beyond this course:**
  - Different trade-offs and practical considerations
  - Other types of algorithms which can achieve entropy asymptotically

Thank You!