Lecture notes for course "Data Compresion with Deep Probabilistic Models" by Prof. Robert Bamler
Uni Tuebingen, 21 May 2021; more course materials at https://robamler.github.io/teaching/compress21/

Last Video: Theoretical bounds for lossless compression with symbol codes

↳ lower bound (for any uniq. dec. symbol code):

$$\boxed{L := \mathbb{E}_p[\ell(x)] \geq H[p]}$$

↳ upper bound for optimal symbol code:

$$\boxed{L_{opt} < H[p] + 1 \text{ bit } (\text{per symbol})}$$

"Shannon Coding"

Problem Set #2

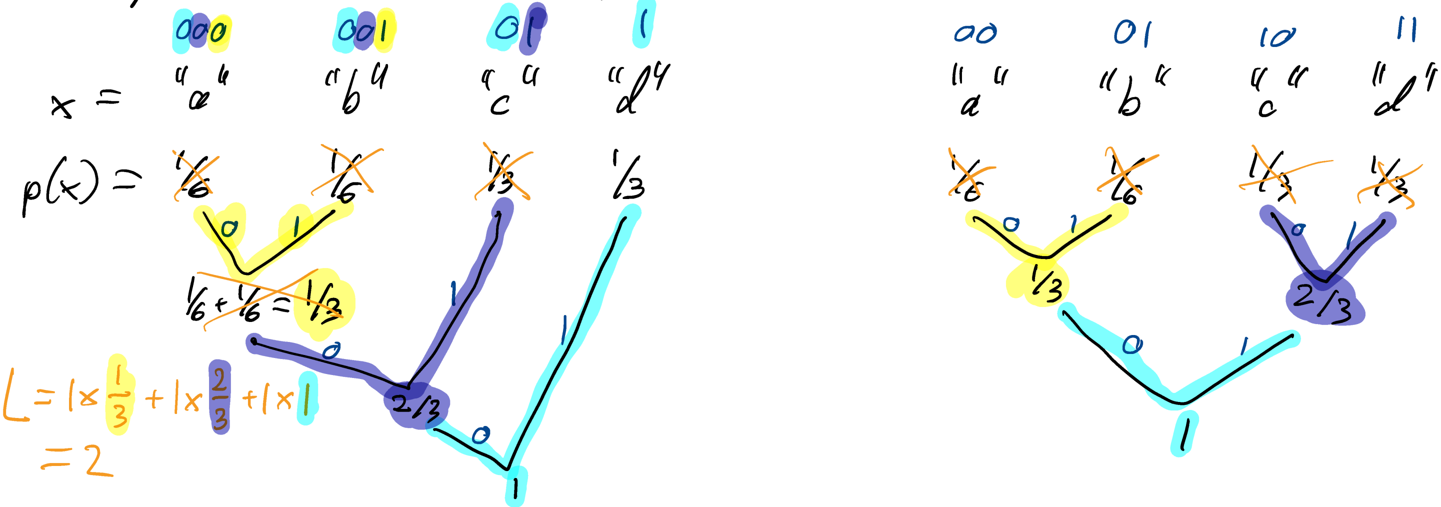
↳ theoretical bounds beyond symbol codes:

$$\boxed{H[p] \leq L_{opt} < H[p] + \varepsilon} \quad \varepsilon = \frac{1}{m} \xrightarrow{m \to \infty} 0$$

↳ implement Huffman coding in Python

Claim: H.C. is optimal

Complication: breaking ties in H.C.



$L = \sum_{x \in \mathcal{X}} p(x) \ell(x)$ $\qquad$ $L = 2$

$= \frac{1}{6} \times 3 + \frac{1}{6} \times 3 + \frac{1}{3} \times 2 + \frac{1}{3} \times 1 = 2$

<u>Theorem 1:</u> $L = \mathbb{E}_p[\ell(x)]$ does not depend on how you break ties in H.C. ✓

<u>Remark:</u> Encoder & decoder still have to break ties in the same way.

<u>Theorem 2:</u> "Huffman Coding constructs an optimal sym. code"

Assumptions: $\left\{ \begin{array}{l} \cdot \text{ alphabet } \mathcal{X} \text{ with } |\mathcal{X}| \geq 2 \\ \cdot\ p(x) > 0 \quad \forall x \in \mathcal{X} \end{array} \right\}$ ⊛

<u>Then:</u> $\forall$ uniq. decodable sym. codes C on $\mathcal{X}$ that are optimal w.r.t. p $\exists$ a Huffman Code C_H with the same code word lengths $\forall x \in \mathcal{X}$ (i.e.: $|C(x)| = |C_H(x)| \ \forall x \in \mathcal{X}$)

<u>Reminder</u> (Problem 2.1) suffices to show that Theorem 2 holds for $\forall$ optimal prefix codes.

<u>Lemma 1:</u> Assume again ⊛, and let C be an optimal (w.r.t. p) prefix code; let's sort the ~~symbols~~ s.t.

$$p(x_1) \leq p(x_2) \leq p(x_3) \leq \dots$$

break ties by codeword lengths (descendingly), i.e., if $p(x_i) = p(x_{i+1})$ then $\ell(x_i) \geq \ell(x_{i+1})$
(then break ties arbitrarily) $\qquad$ $|C(x_i)|$

Then: (i) $\ell(x_1) \geq \ell(x_2) \geq \ell(x_3) \geq \dots$

(ii) $\ell(x_1) = \ell(x_2)$

Proof of Lemma 1:

(i) assume $\exists\, i,j$ with $i<j$ and $\boxed{\ell(x_i) < \ell(x_j)}$

$\Rightarrow p(x_i) \cancel{\geq} p(x_j)$

$\Rightarrow \boxed{p(x_i) < p(x_j)}$

Claim: thus, C is not optimal because we could swap $C(x_i)$ & $C(x_j)$

$\Rightarrow$ would reduce L

(ii) assume $\ell(x_1) > \ell(x_2)$

(know from (i) that $\ell(x_2) \geq \ell(x') \quad \forall x' \neq x_1$)

$\Rightarrow \ell(x_1) > \ell(x') \quad \forall x' \neq x_1$

Claim: thus, C is not an optimal prefix code, because we could drop the last bit of $C(x_1)$; can't clash if C is a prefix code.

$C(x_1)$ 0110 $\nexists$

$C(x')$ 011

$\Rightarrow$ reduces L by $p(x_1) > 0$

Lemma 2: Assume ⊛ & C is optimal prefix code (w.r.t. p) △

Then: $\exists\, x, x' \in \mathcal{X}$ $\boxed{\text{with } x \neq x' \text{ and } \ell(x) = \ell(x') \geq \ell(\tilde{x})\ \forall \tilde{x} \in \mathcal{X}}$

s.t. $\boxed{C(x) \text{ \& } C(x') \text{ only differ on last bit.}}$ □

Proof of Lemma 2: Assume that such a pair does not exist. But, from Lemma 1, we know $\exists x \neq x'$ that satisfies $\triangle$

Claim: thus, C is not optimal because we can drop the last bit of $C(x)$ without violating prefix code

$\downarrow$

Proof: Let's call $C(x)$ with last bit dropped γ.

Then $\forall \tilde{x} \neq x$:

- $C(\tilde{x})$ is not prefix of $C(x)$
  $\Rightarrow C(\tilde{x})$ is not prefix of γ

  $C(x)\ \underbrace{01101}_{\gamma}\cancel{0}$

  $C(\tilde{x})\ 0110$

- if γ is prefix of $C(\tilde{x}) \Rightarrow |C(\tilde{x})| \geq |\gamma|$
  $\Rightarrow C(\tilde{x})$ is a longest code word
  $\Rightarrow C(\tilde{x})$ & $C(x)$ are two longest code words that differ only on last bit (i.e., they satisfy $\square$)

  $C(x)\ \underbrace{01101}_{\gamma}\cancel{0}$

  $C(\tilde{x})\ \underbrace{01101}1$

Recap:

Theorem 2: "Huffman Coding constructs an optimal sym. code"

Assumptions: $\left\{ \begin{array}{l} \cdot \text{ alphabet } \mathcal{X} \text{ with } |\mathcal{X}| \geq 2 \\ \cdot\ p(x) > 0 \quad \forall x \in \mathcal{X} \end{array} \right\}$ ⊛

Then: $\forall$ (uniq. decodable sym.) codes C on $\mathcal{X}$ that are optimal w.r.t. p $\exists$ a Huffman Code C_H with the same code word lengths $\forall x \in \mathcal{X}$

prefix code

Lemma 1: sort $p(x_1) \leq p(x_2) \leq p(x_3) \leq \dots$

break ties by codeword lengths (descendingly)

Then: (i) $\ell(x_1) \geq \ell(x_2) \geq \ell(x_3) \geq \dots$

(ii) $\ell(x_1) = \ell(x_2)$

Lemma 2: Assume ⊛ & C is optimal prefix code (w.r.t. p) △

Then: $\exists x, x' \in \mathcal{X}$ $\boxed{\text{with } x \neq x' \text{ and } \ell(x) = \ell(x') \geq \ell(\tilde{x}) \, \forall \tilde{x} \in \mathcal{X}}$

s.t. $\boxed{C(x) \,\&\, C(x') \text{ only differ on last bit.}}$ □

Proof of Theorem 2 ("optimality of H.C.")

by induction $|\mathcal{X}|$

- base case: $|\mathcal{X}| = 2$

↳ only optimal prefix codes $\begin{array}{l} C(\text{"a"}) = 0 \\ C(\text{"b"}) = 1 \end{array}$ and $\begin{array}{l} C(\text{"a"}) = 1 \\ C(\text{"b"}) = 0 \end{array}$

these are H.C.

- induction step: $|\mathcal{X}| > 2$

↳ from Lemma 2: $\exists x \neq x'$ with longest code words that differ only on last bit.

↳ if $p(x)$ & $p(x')$ aren't among the two lowest probs. then apply Lemma 1: symbols x_1, x_2 with lowest probs and <u>also</u> longest code word length

→ construct prefix cod C' by swapping

only differ on last bit

$(C(x), C(x'))$ with $(C(x_1), C(x_2))$ $p(x_1), p(x_2)$ are lowest probs

all have same (longest) length
→ swapping them doesn't chang $\ell(x)$ for any $x \in \mathcal{X}$

→ in C': x_1, x_2 with
- lowest probs
- $C'(x_1), C'(x_2)$ are longest & only differ on last bit

<u>Def:</u>
- $\tilde{\mathcal{X}} := (\mathcal{X} \setminus \{x_1, x_2\}) \cup \{☆\}$ ← some new symbol

$\Rightarrow |\tilde{\mathcal{X}}| = |\mathcal{X}| - 1 \geq 2$

- $\underset{\tilde{\mathcal{X}}}{\tilde{p}(\underset{\cap}{\tilde{x}})} = \begin{cases} p(\tilde{x}) & \text{if } \tilde{x} \in \mathcal{X} \\ p(x_1) + p(x_2) & \text{if } \tilde{x} = ☆ \end{cases}$

- $\underset{\tilde{\mathcal{X}}}{\tilde{C}(\underset{\cap}{\tilde{x}})} = \begin{cases} C'(\tilde{x}) & \text{if } \tilde{x} \in \mathcal{X} \\ C'(x_1) \text{ with last bit dropped} & \text{if } \tilde{x} = ☆ \end{cases}$ } reduces L by $p(x_1) + p(x_2)$

<u>Claim:</u> $\tilde{C}$ is an optimal prefix code (w.r.t $\tilde{p}$)

↓

<u>Proof:</u> if it weren't optimal then ∃ better prefix code $\tilde{\tilde{C}}$ on $\tilde{\mathcal{X}}$

→ can construct symbol code on $\mathcal{X}$ by inverting above step (i.e., remove ☆, introduce x_1 & x_2 with $C''(x_1) = \tilde{\tilde{C}}(☆) \| 0$, $C''(x_2) = \tilde{\tilde{C}}(☆) \| 1$)

→ increases L by $p(x_1) + p(x_2)$

on $\mathcal{X}$ $\qquad\qquad\qquad\qquad\qquad$ on $\tilde{\mathcal{X}}$

$$C' \xrightarrow{\text{dropped 1 bit from } C'(x_1) \,\&\, C'(x_2)} \tilde{C}$$

$L' = L \qquad\qquad \tilde{L} = L' - (p(x_1) + p(x_2))$

on $\mathcal{X}$ $\qquad\qquad\qquad\qquad\qquad$ on $\tilde{\mathcal{X}}$

$$C'' \xleftarrow{\text{appending 1 bit to } C''(x_1) \,\&\, C''(x_2)} \tilde{\tilde{C}}$$

assumption: $\tilde{\tilde{L}} < \tilde{L}$

$$L'' = \tilde{\tilde{L}} + p(x_1) + p(x_2)$$

$$< \tilde{L} + p(x_1) + p(x_2) = L'$$

$\Rightarrow \underset{\overset{\|}{L}}{L''} < L'$ $\qquad \Rightarrow C$ is not optimal (contradiction)

$\Rightarrow \tilde{C}$ is optimal prefix code on alphabet $\tilde{\mathcal{X}}$ of size $|\mathcal{X}|-1$

$\Rightarrow$ Theorem 2 applies $\Rightarrow \exists$ H.C. on $\tilde{\mathcal{X}}$ with same $\ell(x)\ \forall x \in \tilde{\mathcal{X}}$ as $\tilde{C}$

$\Rightarrow C'$ has same codeword lengths as a H.C. on $\mathcal{X}$

$\Rightarrow C$ has same ———— " ———— ✓

Next video: · begin thinking about better probabilistic models of the data source

→ correlations

→ play back into source coding algorithms