# Lempel-Ziv Compression (§6.4)

**So far:** Symbol codes achieve $H(X) \leq L(X,C) < H(X)+1$, but always $\geq 1 \frac{\text{bit}}{\text{symbol}}$.

By looking at large blocks, can achieve $H(P)$ for IID sources. ← both in the lossy and in the lossless scenario

**Today:** Lossless compression of "stream" of symbols that can emit $< 1 \frac{\text{bit}}{\text{symbol}}$ is asymptotically optimal for IID sources ($R \to H(X)$), and even is adaptive!

Variations are used in GIF, ZIP, PNG, ... (sometimes combined with Huffman)

## Lempel-Ziv Coding algo

input: stream that ends with special symbol $\perp$

* phrases ← [$\emptyset$]
* while more to compress:
  - read symbols until we obtain "phrase" $\pi \notin$ phrases
    $\Rightarrow \pi = \boxed{\tau}\boxed{x}$ where $\tau \in$ phrases, $x \in \mathcal{A}$
  - append $\pi$ to phrases
  - $k \leftarrow$ index of $\tau$ in phrases
  - write $(k, x)$ in bits

} Splits input into minimal distinct "phrases"

use $\lceil \log(j) \rceil$ bits in $j$-th step ($j = 1, 2, ..$)

can skip if $x = \perp$ (last step)

**Example:** Let's compress A|B|BA|BAA|BAAB|AB|A$\perp$:

| Step | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| phrases | $\emptyset$ | A | B | BA | BAA | BAAB | AB | A$\perp$ |
| $(k,x)$ | — | (0,A) | (0,B) | (2,A) | (3,A) | (4,B) | (1,B) | (1,$\perp$) |
| Compression | | –,0 | 0,1 | 10,0 | 11,0 | 100,1 | 001,1 | 001,– |
| #bits for k | | 0 | 1 | 2 | 2 | 3 | 3 | 3 |

$\Rightarrow$ ☹ 14 bits compressed into 20 bits... but the principle is sound ☺

Q: Intuition how it works? Clear how to decompress?

**Analysis?** How well does it compress? Consider:

$\ell$ = # bits of compression & $R = \frac{\ell}{N}$ compression rate

* **Worst case:** For any string $x^N = x_1 \cdots x_N$,

$$\boxed{R \leq \log \# \mathcal{A} + O\left(\frac{1}{\log N}\right) \longrightarrow \log \# \mathcal{A}}$$

EX

$f = O(g)$ means $\exists C > 0 : f(N) \leq C g(N) \ \forall N$

Thus: LZ does no worse than not compressing at all! (for large N)

* **Average rate:** Let $X^N = X_1 \cdots X_N \overset{\text{iid}}{\sim} P$.

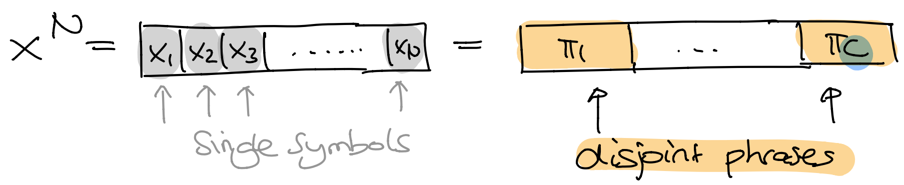
$$\boxed{E[R] \leq H(P) + O\left(\frac{1}{\log N}\right) \longrightarrow H(P)}$$

Thus: For an IID source, LZ achieves entropy $H(P)$! (for large N)

This optimality holds even more generally for an "ergodic" source.

How to prove this?

**Warmup:** Fix source string $x^N$ and assume LZ compresses it into $c$ phrases:




$$\Rightarrow \boxed{\ell} = \sum_{j=1}^{c} \left( \lceil \log(j) \rceil + \lceil \log \# \mathcal{A} \rceil \right) \qquad (A)$$

dominant term

$$\leq \boxed{c \cdot \log(c) + c(1 + \lceil \log \# \mathcal{A} \rceil)}$$

Thus: Need to understand how number of phrases $c$ grows with $N$.

* Worst-case analysis? $\longrightarrow$ Challenge exercise tomorrow. $\boxed{\text{EX CLASS}}$.

* We focus on average rate. <u>key idea</u>: Relate $c$ to $\log \frac{1}{P(x^N)}$ !

For simplicity: $\boxed{\text{Assume all } P(x) \leq \frac{1}{2}}$ $\leftarrow$ but arbitrary $\#\mathcal{A}$ :)

① Classify phrases according to their probability:

IID distribution

$$\Pi_k = \{ \pi_i \mid 2^{-k-1} < P(\pi_i) \leq 2^{-k} \}$$

* for any phrase: $P(\pi) = \Pr(X^N \text{ has prefix } \pi)$

* any string $y^n$ has at most one prefix in any fixed $\Pi_k$

[ if $y^n = \boxed{\pi_i \mid \cdots} = \boxed{\pi_j \mid \cdots}$ then $\pi_i = \boxed{\pi_j \mid \cdots}$ (or vice versa)

$\Rightarrow P(\pi_i) \leq P(\pi_j) \frac{1}{2}$ ↯ ]

$$\Rightarrow 1 \geq \Pr(X^N \text{ has prefix in } \Pi_k) = \sum_{\pi \in \Pi_k} \Pr(X^N \text{ has prefix } \pi) \geq \#\Pi_k \cdot 2^{-(k+1)}$$

$$\Rightarrow \boxed{\#\Pi_k \leq 2^{k+1}}$$

② How large can $P(x^N)$ be if we know it has $c$ phrases?

$$P(x^N) = \prod_i P(\pi_i) = \prod_k \prod_{\pi \in \Pi_k} P(\pi)$$

maximal if $\Pi_0, \Pi_1, \ldots$ as large as possible, i.e. $\#\Pi_k = 2^{k+1}$

$$\leq (2^{-0})^{2^{0+1}} (2^{-1})^{2^{1+1}} \cdots (2^{-(L-1)})^{2^L} (2^{-L})^{c - \sum_{k=1}^{L} 2^k}$$

where $L$ is maximal with $\sum_{k=1}^{L} 2^k = 2^{L+1} - 2 \leq c$

$$\Rightarrow \boxed{L \approx \log(c)}$$

More precisely: $\log(c) - 2 < L \leq \log(c+2) - 1 \leq \log(c)$.

if $c \geq 2$, i.e. $N > 1$

$$\Longrightarrow \log\frac{1}{P(x^N)} \geq \underbrace{\sum_{k=1}^{L}(k-1)2^k} + L\big(c - \underbrace{\sum_{k=1}^{L} 2^k}\big)$$

Check by induction

$$\stackrel{\text{induction}}{=} \overbrace{(L-2)\,2^{L+1}+4} + L\big(c - \overbrace{2^{L+1}+2}\big)$$

cancel!

$$= -4\cdot 2^L + 4 + L(c+2)$$

after some manipulations

$$\geq \underbrace{c\cdot\log c}_{\text{dominant term}} - 6c$$

③ Take expectation value and use $E\left[\log\frac{1}{P(X^N)}\right] = N\cdot H(P)$:

$$\boxed{N\cdot H(P) \geq \underbrace{E[c\cdot\log c]}_{\text{dominant term}} - 6E[c]} \qquad Ⓑ$$

Suppose we could only look at the "dominant" terms in Ⓐ, Ⓑ. Then:

$$E[R] = \frac{E[\ell]}{N} \stackrel{Ⓐ}{\lesssim} \frac{E[c\cdot\log c]}{N} \stackrel{Ⓑ}{\lesssim} H(P)$$

and we would be done! (☺)

④ In reality, things are a bit more complicated:

$$E[R] = \frac{1}{N}E[\ell] \stackrel{Ⓐ}{\leq} \frac{1}{N}E[c\cdot\log c] + \frac{1}{N}\left(\lceil\log \#\Delta\rceil + 1\right)E[c]$$

$$\leq H(P) + \underbrace{O\left(\tfrac{1}{N}\right)\cdot E[c]}_{\text{want that } \to 0\ldots}$$

How to deal with $E[c]$?

$$E[c]\log E[c] \stackrel{Jensen}{\leq} E[c\cdot\log c] \stackrel{Ⓑ}{\leq} (H(P)+6)\,N \quad \text{since certainly } c \leq N$$

Jensen: $f(x) = x\cdot\log x$ is convex

...so $E[c]$ has to grow slower than linear! In fact:

$E[C] = O\left(\frac{N}{\log N}\right)$ and so we arrive at

$$\Rightarrow E[R] \leq H(P) + O\left(\frac{1}{\log N}\right)$$

Nice!

Why is this true? Assume that $f(N) \cdot \log f(N) \leq \gamma \cdot N$ for large $N$.
We claim that $f(N) < (\gamma+1)\frac{N}{\log N}$. Indeed, otherwise we have
$f(N) \geq (\gamma+1)\frac{N}{\log N}$ for a subsequence of $N \to \infty$. Then:

$$f(N) \cdot \log f(N) \geq (\gamma+1)\frac{N}{\log N} \log\left(\overbrace{(\gamma+1)}^{\geq 1}\frac{N}{\log N}\right)$$

$$\geq (\gamma+1) N \left(1 - \underbrace{\frac{\log\log N}{\log N}}_{\to 0}\right) \qquad \lightning \qquad \square$$