How much “information” is contained in $X$?

- compress it into minimal number of $L$ bits per source symbol
- decompress reliably

$\Rightarrow$ average information content is $L$ bits per symbol

Shannon’s source-coding theorem: $L \approx H(X)$
Data Compression / Source Coding

How much “information” is contained in X?

- compress it into minimal number of L bits per source symbol
- decompress reliably

⇒ average information content is L bits per symbol

Shannon’s source-coding theorem: \( L \approx H(X) \)
Two Types of Compression

source $P_X \rightarrow X_1, \ldots, X_n \rightarrow$ compress $L^n$ bits $\rightarrow$ inflate $X_1, \ldots, X_n$

Shannon’s source-coding theorem: $L \approx H(X)$
Two Types of Compression

1. **Lossless compression**: (e.g. zip)
   - maps all source strings to different encodings
   - it shortens some, but necessarily makes others longer
   - design it such that the **average** length is shorter

Shannon’s source-coding theorem: $L \approx H(X)$
Two Types of Compression

1. **Lossless compression**: (e.g. zip)
   - maps all source strings to different encodings
   - it shortens some, but necessarily makes others longer
   - design it such that the average length is shorter

2. **Lossy compression**: (e.g. image compression)
   - map some source strings to same encoding (recovery fails sometimes)
   - If error can be made arbitrarily small, it might be useful in practice

Shannon's source-coding theorem: \( L \approx H(X) \)
### Table 4.1

<table>
<thead>
<tr>
<th>x</th>
<th>$\log_2(P(x))$</th>
</tr>
</thead>
<tbody>
<tr>
<td>........................................................................</td>
<td>−50.1</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−37.3</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−65.9</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−56.4</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−53.2</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−43.7</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−46.8</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−37.3</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−43.7</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−56.4</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−37.3</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−46.8</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−56.4</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−59.5</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−46.8</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−15.2</td>
</tr>
<tr>
<td>........................................................................</td>
<td>−332.1</td>
</tr>
</tbody>
</table>

**Figure 4.10.** The top 15 strings are samples from $X^{100}$, where $p_1 = 0.1$ and $p_0 = 0.9$. The bottom two are the most and least probable strings in this ensemble. The final column shows the log-probabilities of the random strings, which may be compared with the entropy $H(X^{100}) = 46.9$ bits.

---

**Book by David MacKay**
$n(r) = \binom{N}{r}$

$P(x) = p_1^r (1 - p_1)^{N-r}$

$\log_2 P(x)$

$n(r) P(x) = \binom{N}{r} p_1^r (1 - p_1)^{N-r}$
The ‘asymptotic equipartition’ principle is equivalent to:

**Shannon’s source coding theorem (verbal statement).** $N$ i.i.d. random variables each with entropy $H(X)$ can be compressed into more than $NH(X)$ bits with negligible risk of information loss, as $N \to \infty$; conversely if they are compressed into fewer than $NH(X)$ bits it is virtually certain that information will be lost.
The ‘asymptotic equipartition’ principle is equivalent to:

**Shannon’s source coding theorem (verbal statement).** $N$ i.i.d. random variables each with entropy $H(X)$ can be compressed into more than $NH(X)$ bits with negligible risk of information loss, as $N \to \infty$; conversely if they are compressed into fewer than $NH(X)$ bits it is virtually certain that information will be lost.
The ‘asymptotic equipartition’ principle is equivalent to:

**Shannon’s source coding theorem (verbal statement).** $N$ i.i.d. random variables each with entropy $H(X)$ can be compressed into more than $NH(X)$ bits with negligible risk of information loss, as $N \to \infty$; conversely if they are compressed into fewer than $NH(X)$ bits it is virtually certain that information will be lost.

---

**Figure 4.12.** Schematic diagram showing all strings in the ensemble $X^N$ ranked by their probability, and the typical set $T_{N\beta}$. 

---

**Book by David MacKay**
at least $H - \epsilon$ bits. These two extremes tell us that regardless of our specific allowance for error, the number of bits per symbol needed to specify $x$ is $H$ bits; no more and no less.