Information Theory

David Rosenberg

New York University

October 29, 2016

A Measure of Information?

- Consider a discrete random variable X.
- How much "information" do we gain from observing X?
- Information $\approx$ "degree of surprise" from observing $X = x$.
- If we know $\mathbb{P}(X = 0) = 1$, then observing $X = 0$ gives no information.
- If we know $\mathbb{P}(X = 0) = .999$:
  - Observing $X = 0$ gives little information.
  - Observing $X = 1$ gives a lot of surprise / "information"
- Information measure $h(x)$ should depend on $p(x)$:
  - Smaller $p(x) \implies$ More information $\implies$ Larger $h(x)$

Shannon Information Content of an Outcome

Definition

Let $X \in \mathcal{X}$ have PMF $p(x)$. The **Shannon information content of an outcome** x is

$$h(x) = \log\left(\frac{1}{p(x)}\right),$$

where the base of the log is 2. Information is measured in **bits.** (Or **nats** if the base of the log is e.)

- Less likely outcome gives more information.
- Information is **additive** for independent events:
  - If X and Y are independent,

$$\begin{aligned} h(x,y) &= -\log p(x,y) = -\log\left[p(x)p(y)\right] \\ &= -\log p(x) - \log p(y) \\ &= h(x) + h(y) \end{aligned}$$

Entropy

Definition

Let $X \in \mathcal{X}$ have PMF $p(x)$. The **entropy of X** is

$$\begin{aligned} H(X) &= \mathbb{E}_p \log\left(\frac{1}{p(X)}\right) \\ &= -\sum_{x \in \mathcal{X}} p(x) \log p(x), \end{aligned}$$

using convention that $0 \log 0 = 0$, since $\lim_{x \to 0^+} x \log x = 0$.

- Entropy of X is the expected information gain from observing X.
- Entropy only depends on distribution p, so we can write $H(p)$.

Coding

Definition

A **binary source code** C is a mapping from $\mathcal{X}$ to finite 0/1 sequences.

- Consider r.v. $X \in \mathcal{X}$ and binary source code C defined as:

x	$p(x)$	$C(x)$
1	1/2	0
2	1/4	10
3	1/8	110
4	1/8	111

Expected Code Length

- Consider r.v. $X \in \mathcal{X}$ and binary source code C defined as:

x	$p(x)$	$C(x)$	$\log \frac{1}{p(x)}$
1	1/2	0	$\log_2 2 = 1$
2	1/4	10	$\log_2 4 = 2$
3	1/8	110	$\log_2 8 = 3$
4	1/8	111	$\log_2 8 = 3$

- The **entropy** is $H(X) = \mathbb{E} \log [1/p(x)]$:

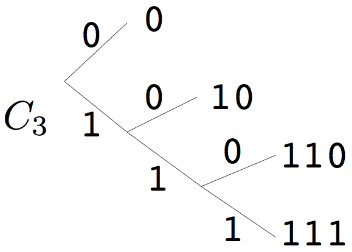
$$H(X) = \frac{1}{2}(1) + \frac{1}{4}(2) + \frac{1}{8}(3) + \frac{1}{8}(3) = 1.75 \text{ bits.}$$

- The **expected code length** is

$$L(C) = \frac{1}{2}(1) + \frac{1}{4}(2) + \frac{1}{8}(3) + \frac{1}{8}(3) = 1.75 \text{ bits.}$$

Prefix Codes

- A code is a **prefix code** if no codeword is a prefix of another.
- Prefix codes can be represented on trees:



- Each leaf node is a codeword.
- It's encoding represents the path from root to leaf.

From David MacKay's *Information Theory, Inference, and Learning Algorithms*, Section 5.1.

Data Compression: What's the Best Prefix Code?

- For $X \sim p(x)$, we get best compression with codeword lengths

$$\ell^*(x) \approx -\log p(x).$$

- **Optimal bit length of x is the Shannon Information of x.**
- Then the **expected codeword length** is

$$\begin{aligned} L^* &= \mathbb{E}\left[-\log p(X)\right] \\ &= H(X) \end{aligned}$$

- Entropy $H(X)$ gives a **lower bound** on coding performance.
- Shannon's Theorem says we can achieve $H(X)$ within 1 bit.

Shannon's Source Coding Theorem

Theorem (Shannon's Source Coding Theorem)

The expected length L of any binary prefix code for r.v. X is at least $H(X)$:

$$L \geqslant H(X).$$

There exist codes with lengths $\ell(x) = \lceil -\log_2 p(x) \rceil$ achieving

$$H(X) \leqslant L < H(X) + 1.$$

- **Notation** $\lceil x \rceil = \text{ceil}(x) = (\text{smallest integer} \geqslant x)$

Shannon's Source Coding Theorem: Summary

- For any $X \sim p(x)$, $\exists$ code with $L \approx H(X)$.
- Get arbitrarily close to $H(X)$ by grouping multiple X's and coding all at once.
- If we know the distribution of X, we can code optimally.
  - e.g. Use **Huffman codes** or **arithmetic codes.**
- What if we don't know $p(x)$, and we use $q(x)$ instead?

Coding with the Wrong Distribution: Core Calculation

- Allow fractional code lengths: $\ell_q(x) = -\log q(x)$
- Then expected length for coding $X \sim p(x)$ using $\ell_q(x)$ is

$$\begin{aligned} L &= \mathbb{E}_{X\sim p(x)}\ell_q(X) \\ &= -\sum_x p(x)\log q(x) \\ &= \sum_x p(x)\log\left[\frac{p(x)}{q(x)}\frac{1}{p(x)}\right] \\ &= \sum_x p(x)\log\frac{p(x)}{q(x)} + \sum p(x)\log\frac{1}{p(x)} \\ &= \mathrm{KL}(p\|q) + H(p), \end{aligned}$$

where $\mathrm{KL}(p\|q)$ is the Kullback-Leibler divergence between p and q.

Entropy, Cross-Entropy, and KL-Divergence

- The **Kullback-Leibler** or **"KL" Diverence** is defined by

$$\mathrm{KL}(p\|q) \quad = \quad \mathbb{E}_p \log\left(\frac{p(X)}{q(X)}\right).$$

- $\mathrm{KL}(p\|q)$: **#(extra bits)** needed if we code with $q(x)$ instead of $p(x)$.
- The **cross entropy** for $p(x)$ and $q(x)$ is defined as

$$H(p,q) = -\mathbb{E}_p \log q(X).$$

- $H(p,q)$: **#(bits)** needed to code $X \sim p(x)$ using $q(x)$.
- Summary:

$$H(p,q) = H(p) + \mathrm{KL}(p\|q).$$

Coding with the Wrong Distribution: Integer Lengths

Theorem

If we code $X \sim p(x)$ using code lengths $\ell(x) = \lceil -\log_2 q(x) \rceil$, the expected code length is bounded as

$$H(p) + KL(p\|q) \leqslant \mathbb{E}_p \ell(X) < H(p) + KL(p\|q) + 1.$$

- So with an implementable code (using integer codeword lengths), the expected code length is within 1 bit of what could be achieved with $\ell(x) = -\log_2 q(x)$.
- Proof is a slight tweak on the "core calculation".

Jensen's Inequality

Theorem (Jensen's Inequality)

If $f : \mathcal{X} \to \mathbf{R}$ is a ***convex*** *function, and $X \in \mathcal{X}$ is a random variable, then*

$$\mathbb{E}f(X) \geqslant f(\mathbb{E}X).$$

Moreover, if f is ***strictly convex****, then equality implies that $X = \mathbb{E}X$ with probability 1 (i.e. X is a constant).*

- e.g. $f(x) = x^2$ is convex. So $\mathbb{E}X^2 \geqslant (\mathbb{E}X)^2$. Thus

$$\mathrm{Var}X = \mathbb{E}X^2 - (\mathbb{E}X)^2 \geqslant 0.$$

Gibbs Inequality ($\mathrm{KL}(p\|q) \geqslant 0$)

Theorem (Gibbs Inequality)

Let $p(x)$ and $q(x)$ be PMFs on $\mathcal{X}$. Then

$$KL(p\|q) \geqslant 0,$$

with equality iff $p(x) = q(x)$ for all $x \in \mathcal{X}$.

- KL divergence measures the "distance" between distributions.
- Note:
  - KL divergence **not a metric**.
  - KL divergence is **not symmetric**.

Gibbs Inequality: Proof

$$
\begin{aligned}
\mathrm{KL}(p\|q) &= \mathbb{E}_p\left[-\log\left(\frac{q(X)}{p(X)}\right)\right] \\
&\geqslant -\log\left[\mathbb{E}_p\left(\frac{q(X)}{p(X)}\right)\right] \qquad \text{(Jensen's)} \\
&= -\log\left[\sum_{\{x \mid p(x)>0\}} p(x)\frac{q(x)}{p(x)}\right] \\
&= -\log\left[\sum_{x\in\mathcal{X}} q(x)\right] \\
&= -\log 1 = 0.
\end{aligned}
$$

- Since $-\log$ is strictly convex, we have strict equality iff $q(x)/p(x)$ is a constant, which implies $q = p$.
- Essentially the same proof for PDFs.

KL-Divergence for Model Estimation

- Suppose $\mathcal{D} = \{x_1, \ldots, x_n\}$ is a sample from **unknown** $p(x)$ on $\mathcal{X}$.
- **Hypothesis space**: $\mathcal{P}$ some set of distributions on $\mathcal{X}$.
- Idea: Find $q \in \mathcal{P}$ that minimizes $\mathrm{KL}(p\|q)$:

$$\operatorname*{arg\,min}_{q \in \mathcal{P}} \mathrm{KL}(p, q) = \operatorname*{arg\,min}_{q \in \mathcal{P}} \mathbb{E}_p \left[\log \left(\frac{p(X)}{q(X)} \right) \right]$$

- Don't know p, so **replace expectation by average over** $\mathcal{D}$:

$$\operatorname*{arg\,min}_{q \in \mathcal{P}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \log \left[\frac{p(x_i)}{q(x_i)} \right] \right\}$$

Estimated KL-Divergence

- The **estimated KL-divergence**:

$$
\begin{aligned}
& \frac{1}{n}\sum_{i=1}^{n}\log\left[\frac{p(x_i)}{q(x_i)}\right] \\
= \quad & \frac{1}{n}\sum_{i=1}^{n}\log p(x_i) - \frac{1}{n}\sum_{i=1}^{n}\log q(x_i).
\end{aligned}
$$

- The minimizer of this over $q \in \mathcal{P}$ is also

$$
\operatorname*{arg\,max}_{q\in\mathcal{P}} \sum_{i=1}^{n}\log q(x_i).
$$

- This is exactly the objective for the **MLE**.
- **Minimizing KL between model and truth leads to MLE.**