جامعة أم القرى
UMM AL-QURA UNIVERSITY

Information Theory and Inference for Learning: Measuring the Information Part1

Dr. Hana Alghamdi
hamghamdi@uqu.edu.sa

Computer Science and Artificial Intelligence Department
Umm Al-Qura University

Self-information

- The basic intuition behind information theory is that learning that an unlikely event has occurred is more informative than learning that a likely event has occurred.
- Which statement has more information?
  - "The sun rose this morning", are you surprised?
  - "There was a solar eclipse this morning", are you surprised?
- There is inverse relationship between probability and surprise/information:
  - probability high, surprise low
  - probability low, surprise high

Self-information

- We would like to quantify information in a way that formalizes this intuition, so we define self-information(or information content) that measures how much information is gained when an event that was initially uncertain becomes known:

$$I(x) = f\{P(x)\}$$

- The self-information $I(x)$ is a function $f(.)$ that depends on the probability that event will happen.

Self-information

- $f(.)$ must satisfy the following properties :
  1. Likely events should have low information content, and in the extreme case, events that are guaranteed to happen should have no information content whatsoever. Less likely events should have higher information content.

$$P(x) = 1 \quad \rightarrow \quad I(x) = 0$$
$$P(x) < 1 \quad \rightarrow \quad I(x) > 0$$

  2. Independent events should have additive information. For example, finding out that a tossed coin has come up as heads twice should convey twice as much information as finding out that a tossed coin has come up as heads once.

$$I(x, y) = I(x) + I(y) \qquad \text{iff} \quad P(x, y) = P(x)P(y)$$

Self-information

- Shannon found that the log function can satisfy the previous properties:

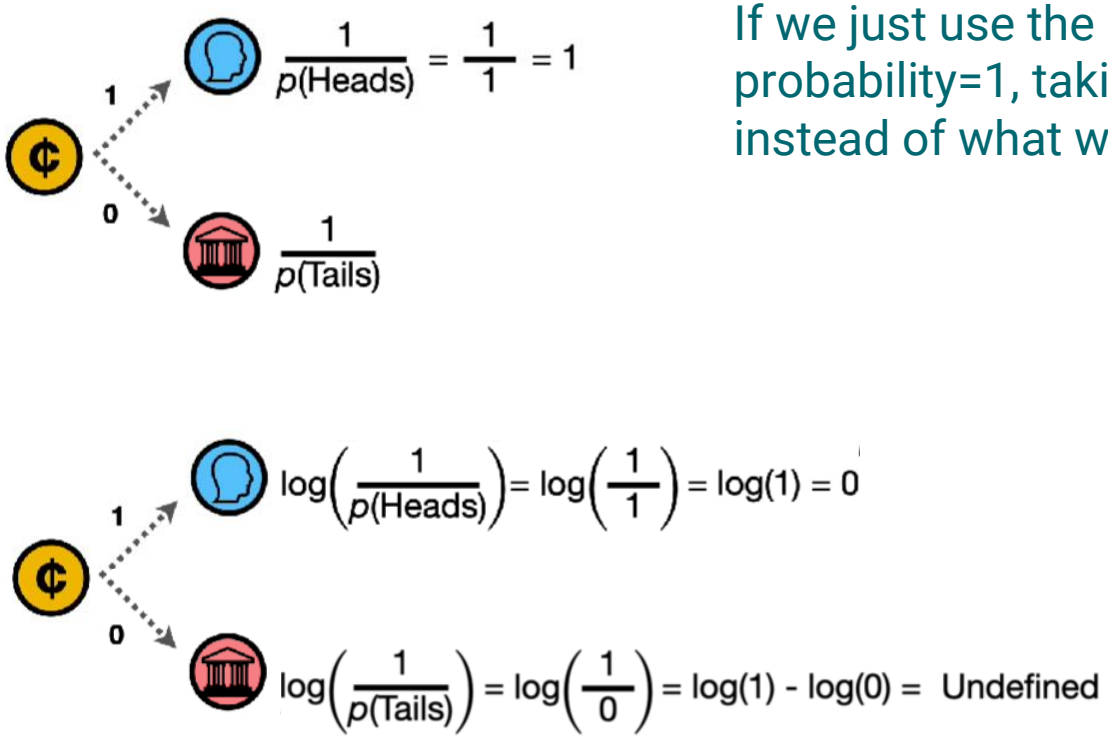
$$I(x) = \log \frac{1}{P\{x\}} = -\log P\{x\}$$

Log base e => unit is nats

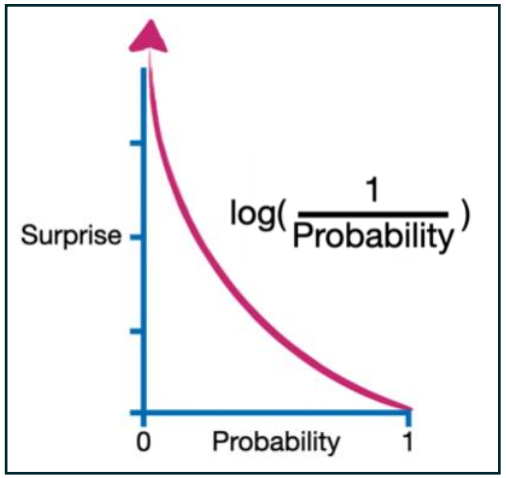
Log base 2 => unit is bits or shannons

Self-information

Why log?



If we just use the inverse directly, when the probability=1, taking the inverse we get 1 instead of what we want, 0



So we take the log of the inverse

Entropy

- **Self-information** deals only with a single outcome. It measures the amount of surprise or information gained when a specific event occurs. It focuses on individual events and how much information they provide when they happen.
- **Shannon Entropy** measures the average information content or uncertainty of a probability distribution associated with a single random variable. In other words, the entropy of a distribution is the average self-information over all possible events in the distribution:

$$H(\mathrm{x}) = \mathbb{E}_{\mathrm{x}\sim P}[I(x)] = -\mathbb{E}_{\mathrm{x}\sim P}[\log P(x)],$$

$$= -\sum_i P(x_i) \log P(x_i)$$

also denoted $H(P)$

Entropy

- Entropy is a lower bound on the number of bits (if the logarithm is base 2) needed on average to encode symbols drawn from a distribution P.
- Distributions that are nearly deterministic have low entropy.
- Distributions that are nearly uniform have high entropy.
- Entropy forms the building block of many other measures

- **Note**: Entropy measures the uncertainty of distribution **NOT** the exact number of bits we need in order to communicate the information, some references use ***Shannon*** as a unit of measure instead of ***bit***, to eliminate the confusion.

  See Figure 3.5 for a demonstration

Entropy

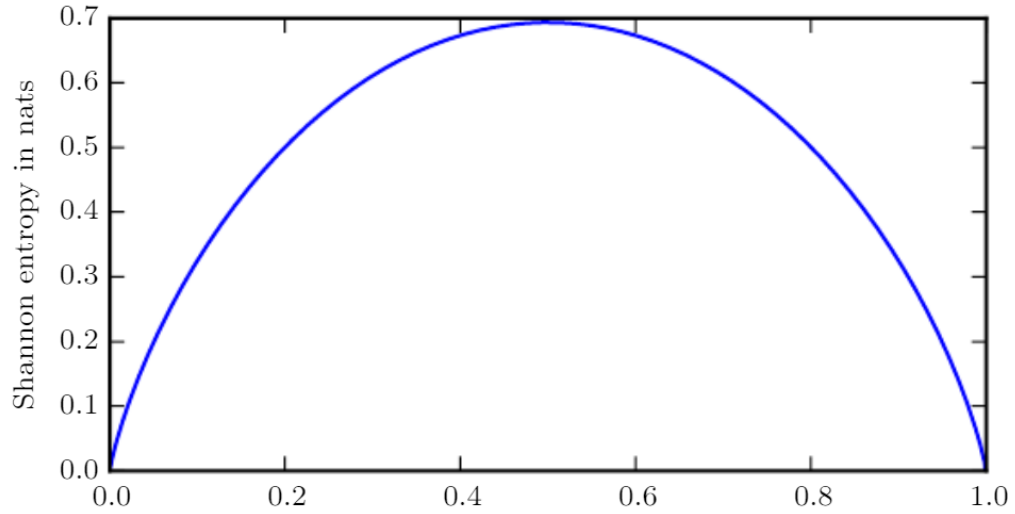


Figure 3.5: Shannon entropy of a binary random variable. This plot shows how distributions that are closer to deterministic have low Shannon entropy while distributions that are close to uniform have high Shannon entropy. On the horizontal axis, we plot p, the probability of a binary random variable being equal to 1. The entropy is given by $(p-1)\log(1-p) - p\log p$. When p is near 0, the distribution is nearly deterministic, because the random variable is nearly always 0. When p is near 1, the distribution is nearly deterministic, because the random variable is nearly always 1. When $p = 0.5$, the entropy is maximal, because the distribution is uniform over the two outcomes.

For random variable x={0,1}
Let assume the probability of x=1 is *p*
Then the probability of x=0 is 1-*p*

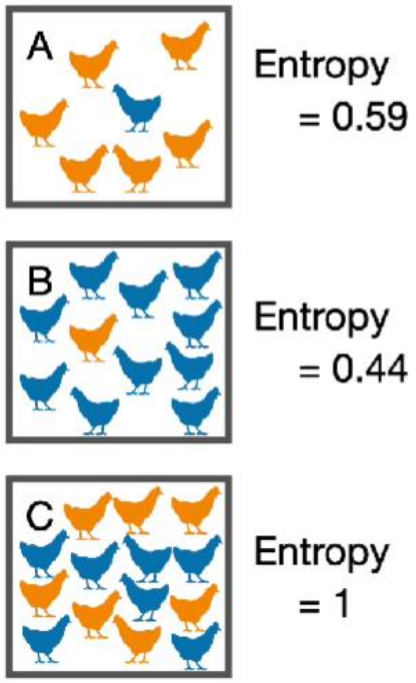
The entropy of this random variable:

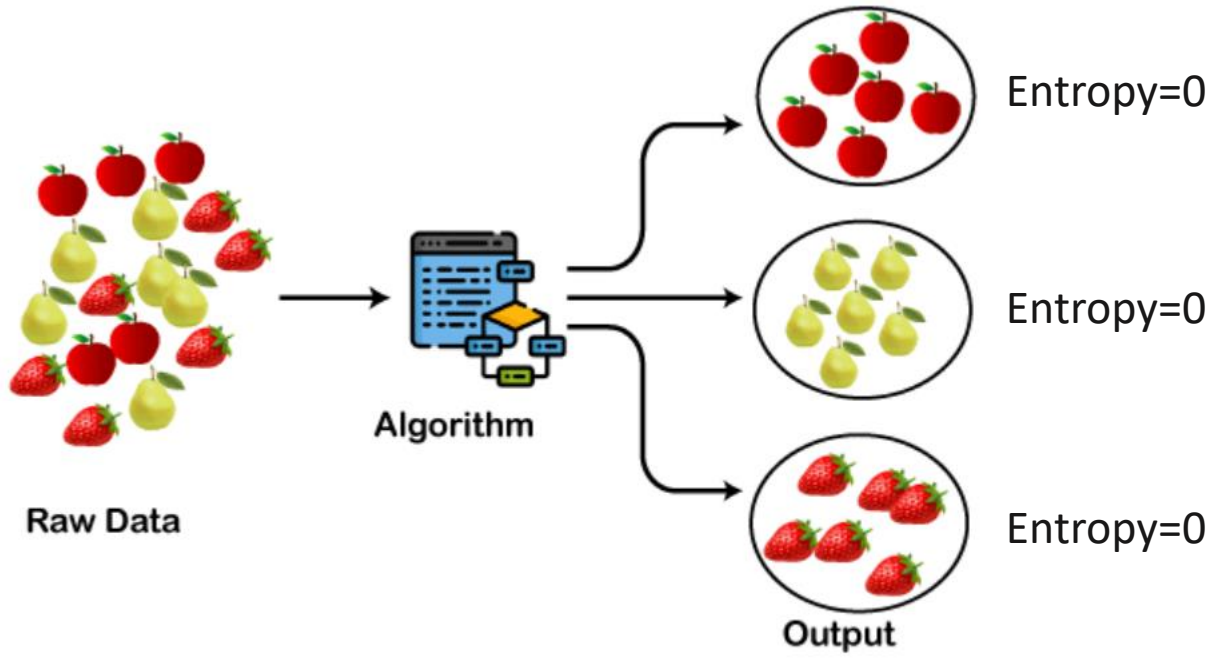
$$H(x) = -(p \log p + (1-p)\log(1-p))$$
$$= - p \log p + (-1+p)\log(1-p)$$
$$= (p-1)\log(1-p) - p \log p$$

The log base is e

Entropy

- In machine learning (Clustering Algorithms) we incorporate entropy to quantify the similarity or difference in clusters (Mutual Information)





Joint Entropy

- We now extend the definition to a pair of random variables. If we have joint distribution that tells us the probability of each combination of x and y, *p*(x,y). Then the joint entropy is given as follows:

$$H(X,Y) = -\sum_x \sum_y p(x,y) \log p(x,y)$$

If X and Y were independent random variables, then $H(X,Y) = H(X) + H(Y)$.

which can also be expressed as

$$H(X, Y) = -E \log p(X, Y).$$

Conditional Entropy

Definition If $(X, Y) \sim p(x, y)$, the *conditional entropy* $H(Y|X)$ is defined as

$$H(Y|X) = \sum_{x \in \mathcal{X}} p(x) H(Y|X = x) \tag{2.10}$$

$$= -\sum_{x \in \mathcal{X}} p(x) \sum_{y \in \mathcal{Y}} p(y|x) \log p(y|x) \tag{2.11}$$

$$= -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log p(y|x) \tag{2.12}$$

$$= -E \log p(Y|X). \tag{2.13}$$

Theorem 2.2.1 (*Chain rule*)

$$H(X, Y) = H(X) + H(Y|X). \tag{2.14}$$

Proof in Elements of Information Theory page 17

The relative entropy (Kullback-Leibler divergence)

- **relative entropy** is a measure of how one probability distribution is different from a second (reference distribution).
- If we have $\mathrm{x} = \{x_1, x_2, x_3, \ldots, x_n\}$, and we have two separate probability distributions over the same random variables $P(\mathrm{x})$ and $Q(\mathrm{x})$
- We can compute the difference of the two probability of one sample:

$$\log P(x_1) - \log Q(x_1) \quad \text{or} \quad \log \frac{P(x_1)}{Q(x_1)}$$

- The the above is the difference for just one sample x_1, we want to computes the average difference between the two distributions:

$$\sum_i P(x_i) \log \frac{P(x_i)}{Q(x_i)} \quad \text{or} \quad \int P(x) \log \frac{P(x)}{Q(x)}\, dx$$

The relative entropy (Kullback-Leibler divergence)

- **The general notation**:

$$D_{\mathrm{KL}}(P\|Q) = \mathbb{E}_{\mathrm{x}\sim P}\left[\log \frac{P(x)}{Q(x)}\right]$$

If *P* and *Q* are identical then Kullback-Leibler divergence =0

- **Note:** $D_{\mathrm{KL}}(P\|Q) \neq D_{\mathrm{KL}}(Q\|P)$

$$D_{\mathrm{KL}}(Q\|P) = \mathbb{E}_{\mathrm{x}\sim Q}\left[\log \frac{Q(x)}{P(x)}\right]$$

Cross Entropy

- A quantity that is closely related to the KL divergence is the cross-entropy.
- Very common used as cost function in machine learning to train classifiers.
- The cross-entropy between two probability distributions *P* and *Q* is calculated as follows:

$$H(P,Q) = H(P) + D_{\mathrm{KL}}(P\|Q)$$

$$H(P,Q) = -\mathbb{E}_{x\sim P}\log Q(x)$$

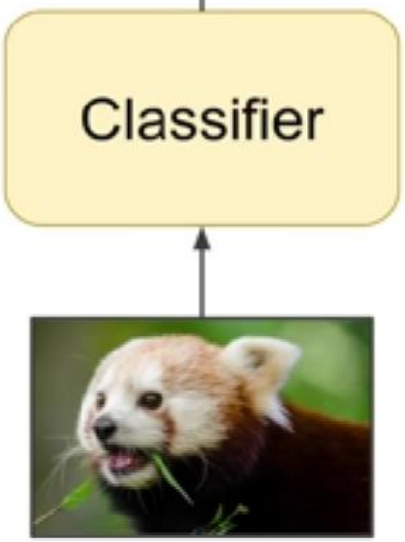
$$= -\sum_i P(x_i)\log Q(x_i)$$

- In machine learning *P*(x) is usually the true probability distribution (the ground truth).
- And *Q*(x) is the predicted probability distribution (the model's output).
- If Q is equal to P then cross-entropy is simply equal to entropy
- If the distributions differ, then cross-entropy is grater than the entropy, the extra amount is KL divergence

Cross Entropy

Example in machine learning:

	Cat	Dog	Fox	Cow	Red Panda	Bear	Dolphin
True distribution:	0%	0%	0%	0%	100%	0%	0%
Predicted distribution:	2%	30%	45%	0%	25%	5%	0%



Cross-Entropy Loss:

$H(\mathbf{p}, \mathbf{q}) = -\Sigma_i \, p_i \log(q_i)$

$= -\log(0.25) = 1.386$