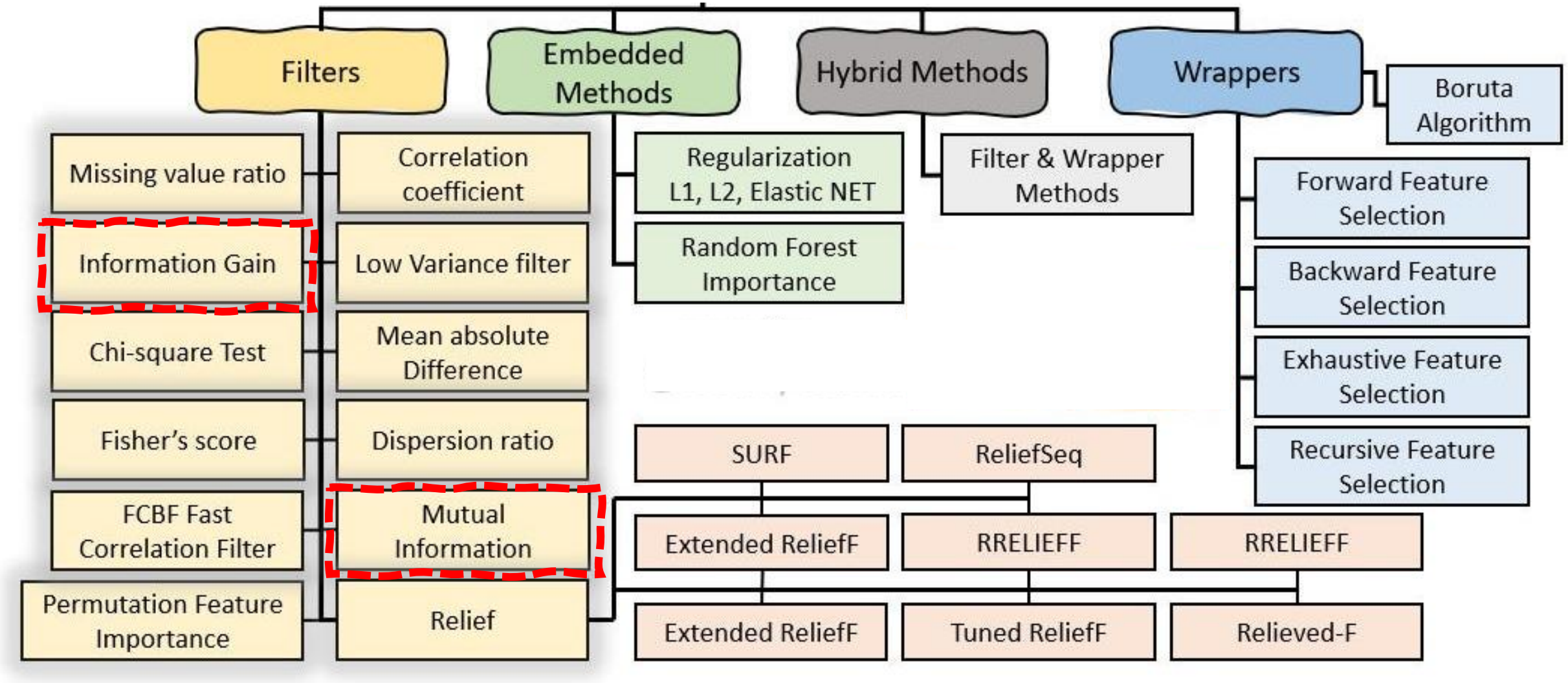
Foundation of
Information Theory

Nov 12 2025

Giảng viên: TS. Lưu Phúc Lợi
Luu.p.loi@googlemail.com
Zalo: 0901802182



https://medium.com/analytics-vidhya/feature-selection-extended-overview-b58f1d524c1c

Content

- Basic Information Theory
- What is information?
- What is entropy?
- What is cross entropy and conditional entropy?
- What is KL Divergence?
- What is Mutual Information?
- What is Information Gain?
- Decision Tree
- Random Forest
- Feature Importance from Random Forest
- Hands-on Exercises

What is information ?

If I tell you

1. Tomorrow the earth will still be going around the sun.
2. Tomorrow an asteroid will destroy the world.

Questions?

- Which statement feels like it contains more information ? Why ?
- Notice one of the question contains more words, does that automatically imply more information?

We define information in Information Theory based on uncertainty.

Looking at the 1st statement: Tomorrow the earth will still be going around the sun.

- How certain are you about earth still going around the sun?
- Since this is a very **likely** outcome, we are very certain.
- Therefore, events that are highly likely (and what we expected anyway) doesn't tells us anything.

In contrast, looking at the 2nd statement: Tomorrow an asteroid will destroy the world.

- This is a highly **unlikely** event and not what you expect.
- Therefore, this statement contains a lot of information.

From these 2 observations, we conclude that information has an inverse relationship to its likelihood of happening.

- The higher the probability of an event, the lower the information.
- The lower the probability of an event, the higher the information.
- Therefore, the concept of information is related to the inverse of the probability equation.
- In information theory, Information is defined as

$$I = \log\left(\frac{1}{p(x)}\right).$$

Let's look at the 2 statements in relation to this equation

For the statement: **Tomorrow the earth will still be going around the sun.**

- Most of us would consider the probability of this event as 100% or 1.
- From a probability sense, it therefore contains 0 information.
- From Information theory perspective, we should get a 0.

$$\underbrace{I = \log\left(\frac{1}{p(x)}\right) = \log\left(\frac{1}{1}\right) = 0.}_{\textbf{Notice that 100\% certainty implies 0 information.}}$$

For the statement: Tomorrow an asteroid will destroy the world.

- Most of us would consider the probability of this event as almost 0, or very small like p(x) = 0.0000000000001
- From a probability sense, it therefore contains an enormous amount of information.
- From Information theory perspective, we should get a larger number.

$$\underbrace{I = \log\left(\frac{1}{p(x)}\right) = \log\left(\frac{1}{0.0000000000001}\right) \approx 29.93.}_{\textbf{Notice that unlikely events contains larger information.}}$$

Let's try it with Python

We have 3 basketball players that are playing 3 way game

1. Kate is an expert player , very tall , p(win = kate) = 0.95
2. Matt is a bad player, not tall, p(win = matt) = 0.0499
3. Alice cannot walk and is blind p(win = alice) = 0.0001

How much information is contained in each statement

1. Kate wins the basketball game.
2. Matt wins the basketball game.
3. Alice wins the basketball game.

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

The amount of information we see depends on how information is stored

We have 3 basketball players that are playing 3 way game

1. Kate is an expert player , very tall , p(win = kate) = 0.95
2. Matt is a bad player, not tall, p(win = matt) = 0.0499
3. Alice cannot walk and is blind p(win = alice) = 0.0001

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

How much information is contained in each statement

1. Kate won the basketball game.

$$I = log_e\left(\frac{1}{0.95}\right) \approx 0.051, \quad I = log_2\left(\frac{1}{0.95}\right) \approx 0.074, \quad I = log_{10}\left(\frac{1}{0.95}\right) \approx 0.022$$

2. Matt won the basketball game.

$$I = log_e\left(\frac{1}{0.0499}\right) \approx 3, \quad I = log_2\left(\frac{1}{0.0499}\right) \approx 4.3, \quad I = log_{10}\left(\frac{1}{0.0499}\right) \approx 1.3$$

3. Alice won the basketball game.

$$I = log_e\left(\frac{1}{0.0001}\right) \approx 9.2, \quad I = log_2\left(\frac{1}{0.0001}\right) \approx 13.3, \quad I = log_{10}\left(\frac{1}{0.0001}\right) \approx 4$$

We are used to having log_e, but the base value tells us how the information is stored.

Without knowing the base value, information is meaningless.

Obviously, if Alice won something big must have happened

How is the base value related to storage?

Let's start by counting ,

	base 2	base 3	base 10
0	0000	0000	0000
1	0001	0001	0001
2	0010	0002	0002
3	0011	0010	0003
4	0100	0011	0004
5	0101	0012	0005
6	0110	0020	0006
7	0111	0021	0007
8	1000	0022	0008
9	1001	0100	0009
10	1010	0101	0010
11	1011	0102	0011

Base 2 means that when you reach 2, you carry over to the next digit

Some researchers believe that it is because humans have 10 fingers, our mathematics evolved to be base 10.

But in reality, we could have used any base. In fact, computers have evolved to use binary base 2.

The result is that the number 10, has vastly different meanings

Base 2 : 10 = 2
Base e : 10 = 2.718
Base 3: 10 = 3
Base 10: 10 = 10

Python code

```
>>> np.log(np.e)
1.0
>>> np.log2(2)
1.0
>>> np.log10(10)
1.0
```

Example:

Example: Flipping a coin 50/50 chance of head or tails

Someone told you that a head was flipped p(head) = 0.5, how much information was gained?

$$I = log_2\left(\frac{1}{0.5}\right) = 1$$

This make sense, in binary, its 0 or 1, so 1 bit of information is enough to hold the result of a coin flip.

Example: Flipping an unfair coin 25/75 chance of head or tails

Someone told you that a head was flipped p(head) = 0.25, how much information?

$$I = log_2\left(\frac{1}{0.25}\right) = 2$$

Notice that it now need 2 bits of information. The fact that a coin is unfair adds information into the system.

Key Takeaway

1. The more rare the event, the more information it conveys.
2. The presentation of Information depends on the numerical base that was chosen.
3. 1/p(x) is the information
4. Log is used to denote how the information is stored.
5. Giving us I = log(1/p(x))

Expectation

Example 1

A fair die is thrown. Find out the expected value of its outcomes.

Solution

If the random variable X is the top face of a tossed, fair, six sided die, then the probability mass function of X is Px (x)= 1/6, for x = 1,2,3,4,5 and 6

The average toss, that is, the expected value of X is

$$E(X)=\sum_x x\,P_X(x)$$

$$\mathrm{E(X)} = \left(1\times\frac{1}{6}\right)+\left(2\times\frac{1}{6}\right)+\left(3\times\frac{1}{6}\right)+\left(4\times\frac{1}{6}\right)+\left(5\times\frac{1}{6}\right)+\left(6\times\frac{1}{6}\right)$$

$$=\frac{1}{6}(1+2+3+4+5+6)$$

$$=\frac{7}{2}$$

$$=3.5$$

Therefore, the expected toss of a fair six sided die is 3.5.

Expectation

Example 2

Six men and five women apply for an executive position in a small company. Two of the applicants are selected for an interview. Let X denote the number of women in the interview pool. We have found the probability mass function of X.

$X = x$	0	1	2
$P(x)$	$\frac{2}{11}$	$\frac{5}{11}$	$\frac{4}{11}$

How many women do you expect in the interview pool?

Solution:

Expected number of women in the interview pool is

$$E(X) = \sum_x x P_X(x)$$

$$= \left[\left(0 \times \frac{2}{11}\right) + \left(1 \times \frac{5}{11}\right) + \left(2 \times \frac{4}{11}\right)\right]$$

$$= \frac{13}{11}$$

Understanding Information lead us into the concept of chaos/Entropy.

If an event x occurred with a probability of p(x), then the information contained in this event is

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

But what if the specific event hasn't happened yet?

At this point, there is a lot of randomness and chaos.
This chaos is measured by Entropy

For example:
Let's say we have 3 candidates running to be the mayor of Boston.
Their individual probability of winning is

Candidate 1 : 10%
Candidate 2 : 40%
Candidate 3 : 50%

Using information, we can measure the information "if an individual has won".

But even before the election take place, you can only measure the uncertainty of the election.
This measurement of uncertainty or chaos of a random event is called

Entropy

Let's say that your life after graduation can turn out in 4 ways

- You haven't graduated yet, so there currently isn't any information. (Information only exists after something has happened).
- What we do have is the randomness of what **could happen**.

p(x)	X = events	Description
0.7	$x = 0$	Found a job you like.
0.2	$x = 1$	Found a job you didn't like.
0.0999999	$x = 2$	Moved back home with your parents
0.0000001	$x = 3$	Won the lottery

How do we go about measuring the randomness of what could happen? **Let's calculate the information of each possible event**

$$I(x=0) = \log_2\left(\frac{1}{0.7}\right) \approx 0.51 \quad \text{and} \quad I(x=1) = \log_2\left(\frac{1}{0.2}\right) \approx 2.32$$

$$I(x=2) = \log_2\left(\frac{1}{0.0999999}\right) \approx 3.32 \quad \text{and} \quad I(x=3) = \log_2\left(\frac{1}{0.0000001}\right) \approx 23.25$$

Entropy is simply the average information (the expected information)

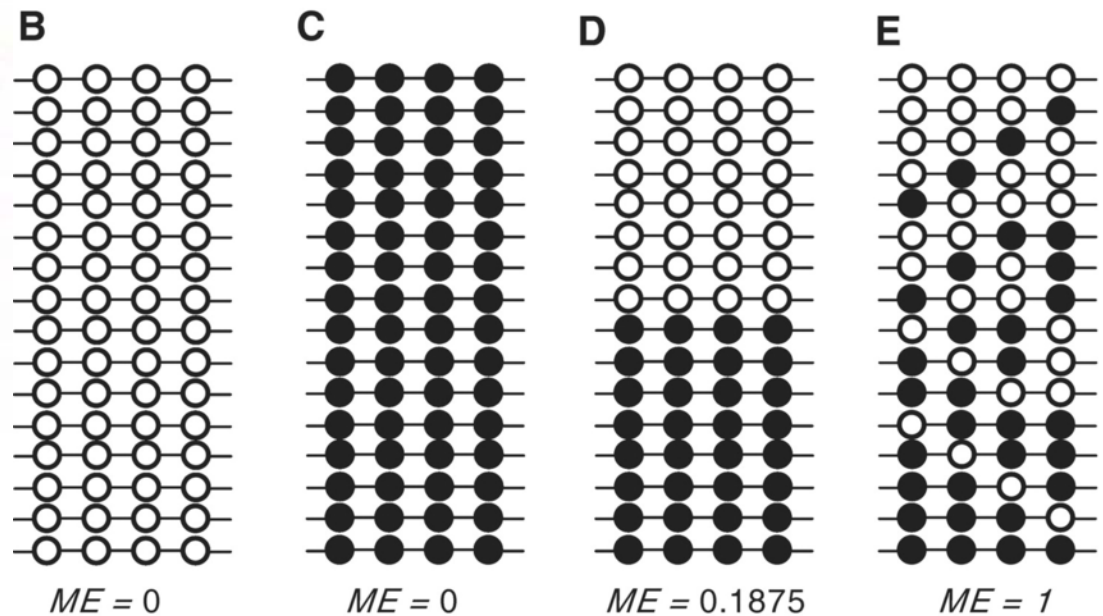
$$\mathbb{E}[I(x)] = \sum_i I(x=i)p(x) \qquad \text{We can now calculate the entropy}$$

$$\begin{aligned}
\mathbb{E}[I(x)] = &\quad I(x=0)p(x=0) &&+I(x=1)p(x=1) &&+I(x=2)p(x=2) &&+I(x=3)p(x=3) \\
\mathbb{E}[I(x)] = &\quad \underbrace{0.51 \cdot 0.7}_{x=0} &&+\underbrace{2.32 \cdot 0.2}_{x=1} &&+\underbrace{3.32 \cdot 0.0999999}_{x=2} &&+\underbrace{23.25 \cdot 0.0000001}_{x=3}
\end{aligned}$$

Continuous Vs Discrete Entropy

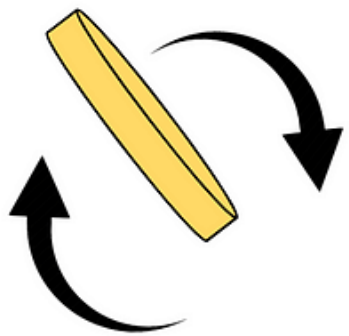
$$H(X) = \mathbb{E}\left[log\left(\frac{1}{p(X)}\right)\right] = \begin{cases} \sum_X \underbrace{log\left(\frac{1}{p(X)}\right)}_{f(x)} p(x) & X \text{ is discrete case} \\ \int_X \underbrace{log\left(\frac{1}{p(X)}\right)}_{f(x)} p(x)\, dx & X \text{ is continuous case} \end{cases}$$

The historic symbol for entropy is H. I would prefer E, but physics took it already as energy (E) :(.



Entropy of a Fair Coin Flip

$$H(X) = -\sum_{i=1}^{N} P(x_i) \log_b P(x_i)$$



$$= -\sum_{i=1}^{2} \frac{1}{2} \log_2 \frac{1}{2}$$

-1

Uniform probability yields **maximum uncertainty** and therefore **maximum entropy**.

$$= -\sum_{i=1}^{2} \frac{1}{2} \cdot (-1)$$

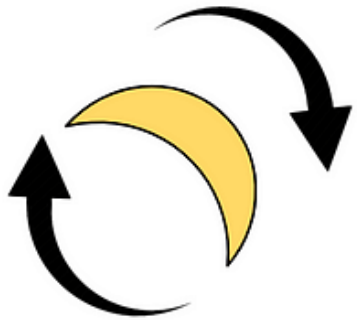
-1

$$= -\left(-\frac{1}{2} + \left(\frac{-1}{2}\right)\right) = -(-1) = 1$$

Sean McClure, 2020

https://medium.com/swlh/a-deep-conceptual-guide-to-mutual-information-a5021031fad0

Entropy of a Biased Coin Flip



$$H(X) = -\sum_{i=1}^{N} P(x_i) \log_b P(x_i)$$

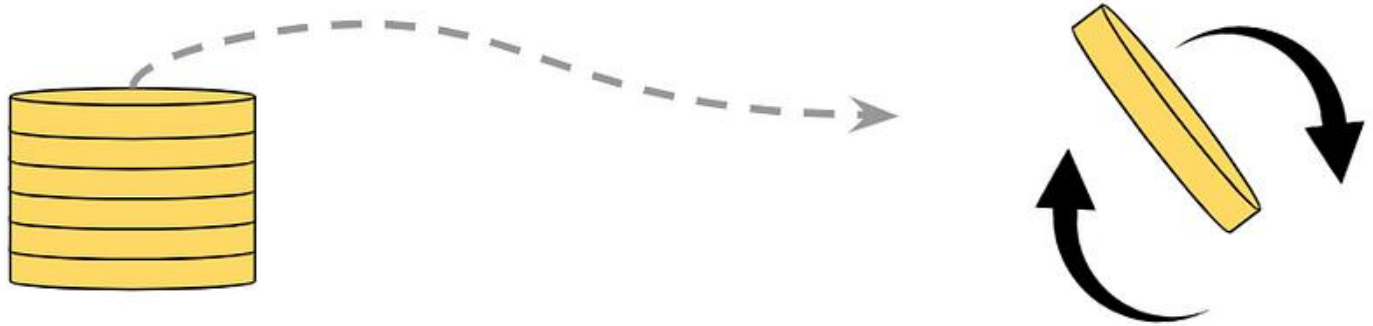
$$= -p\log_2(p) - q\log_2(q)$$

$$= -0.7\log_2(0.7) - 0.3\log_2(0.3)$$

$$= -0.7 \cdot (-0.515) - 0.3(-1.737)$$

$$= 0.88$$

Non-uniform probability yields **less uncertainty** and therefore **less entropy**.

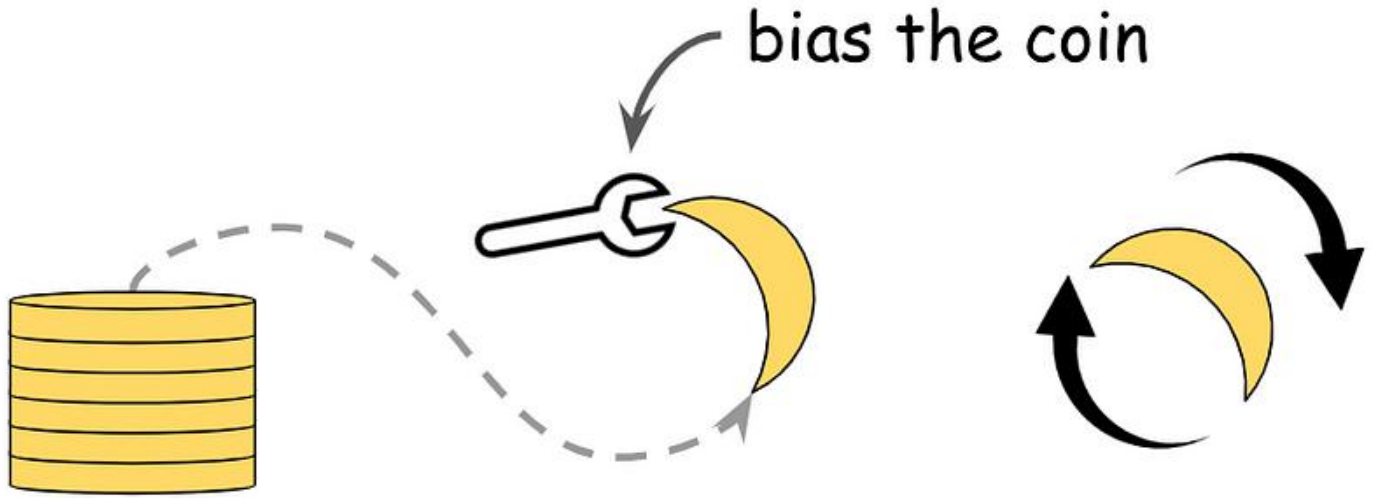


Probabilities

- heads 0.5
- tails 0.5

Entropy

$$H(X) = -\sum_{i=1}^{2} \frac{1}{2} \log_2 \frac{1}{2} = 1$$

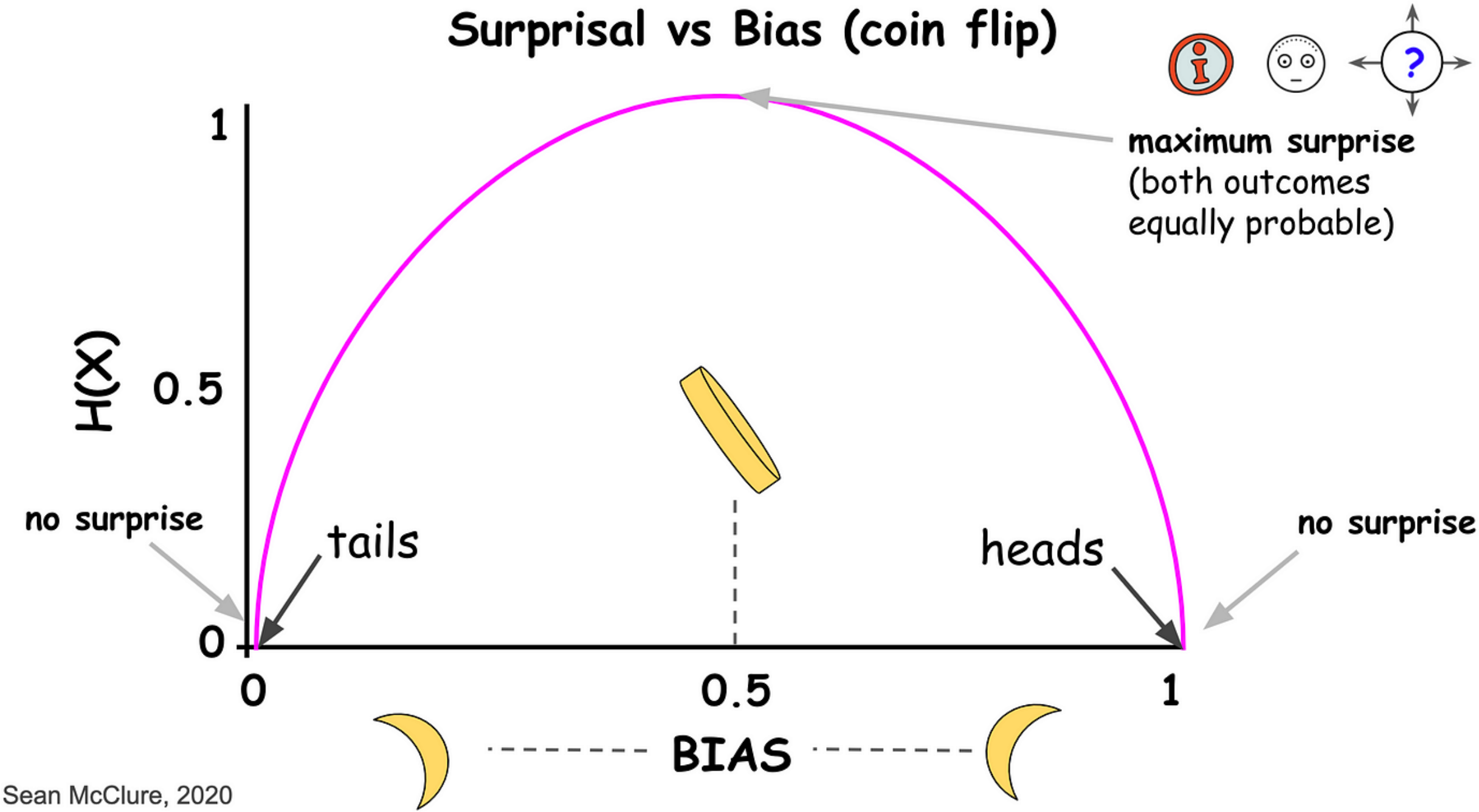


Probabilities

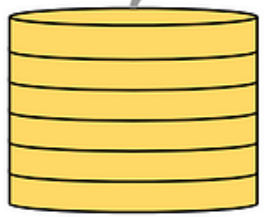
- heads 0.7
- tails 0.3

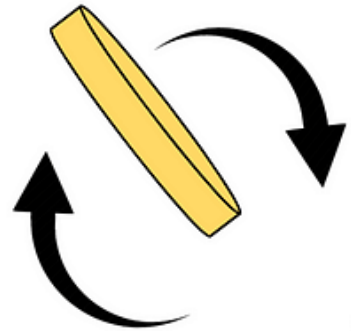
Entropy

$$-0.7\log_2(0.7) - 0.3\log_2(0.3) = 0.88$$



Sean McClure, 2020





Probabilities

- heads 0.5
- tails 0.5

Entropy

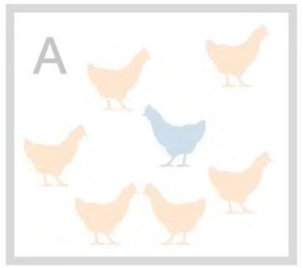
$$H(X) = -\sum_{i=1}^{2} \frac{1}{2} \log_2 \frac{1}{2} = 1$$

- 1 1/6
- 2 1/6
- 3 1/6
- 4 1/6
- 5 1/6
- 6 1/6

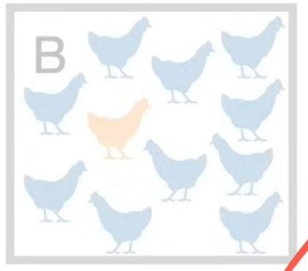
Entropy

$$H(X) = -\sum_{i=1}^{6} \frac{1}{6} \log_2 \frac{1}{6} = 2.585$$

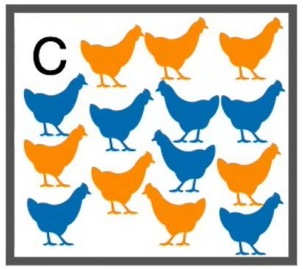
Summary of Information and Entropy



Entropy
= 0.59



Entropy
= 0.44



Entropy
= 1

Lastly, the **Entropy** for area **C** is **1**.

$$\text{Entropy} = \sum p(x)\log\left(\frac{1}{p(x)}\right)$$

$$= \frac{7}{14} \times \log_2\left(\frac{1}{\frac{7}{14}}\right) + \frac{7}{14} \times \log_2\left(\frac{1}{\frac{7}{14}}\right)$$

$$= (0.5 \times 1) + (0.5 \times 1)$$

$$= 1$$

https://www.youtube.com/watch?v=YtebGVx-Fxw

Genome-wide quantitative assessment of variation in DNA methylation patterns

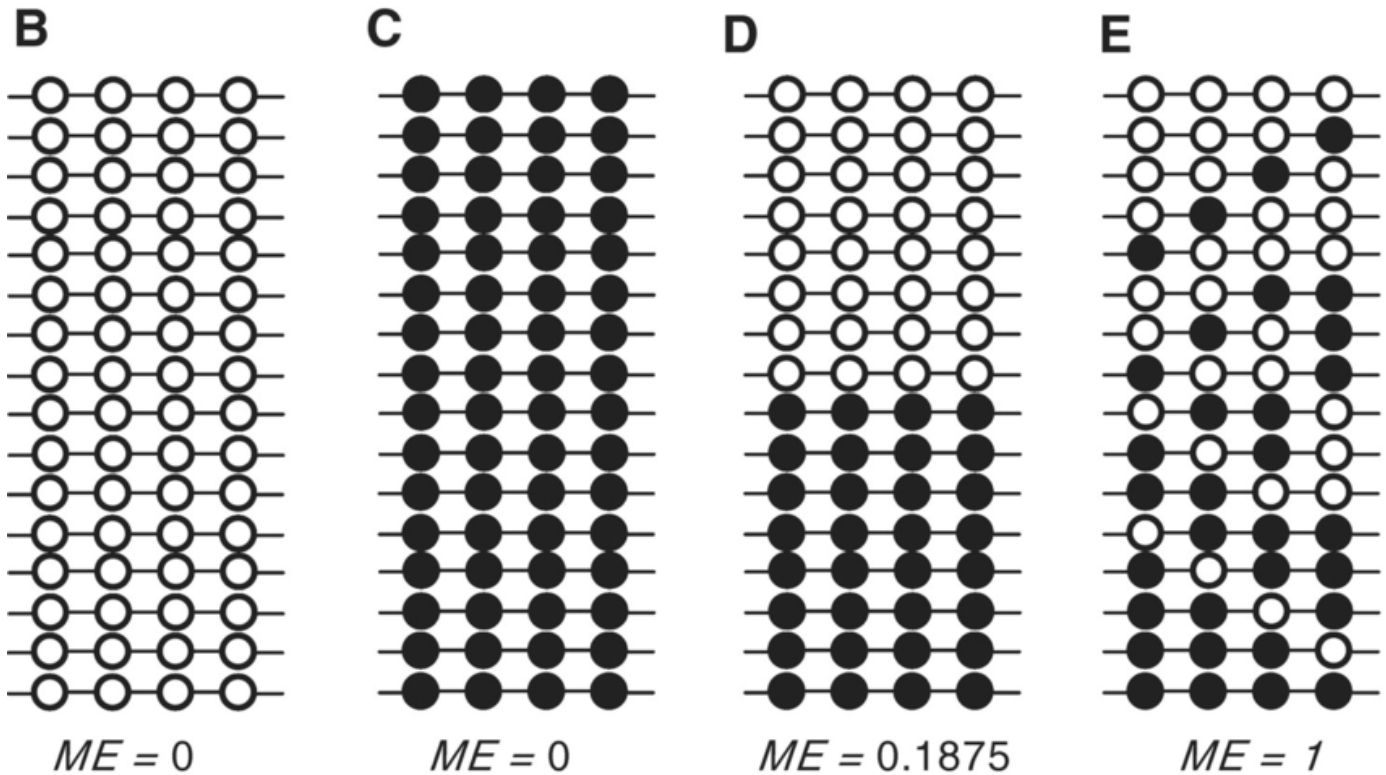
Hehuang Xie ✉, Min Wang, Alexandre de Andrade, Maria de F. Bonaldo, Vasil Galat, Kelly Arndt, Veena Rajaram, Stewart Goldman, Tadanori Tomita, Marcelo B. Soares

Nucleic Acids Research, Volume 39, Issue 10, 1 May 2011, Pages 4099–4108,
https://doi.org/10.1093/nar/gkr017
Published: 27 January 2011 **Article history** ▾

A

$$ME = \frac{e}{b}\sum\left(-\frac{n_i}{N}Log\frac{n_i}{N}\right)$$

ME: Methylation Entropy
e: Entropy for code bit
b: Number of CpG sites
n_i: Observed occurrence of methylation pattern i
N: Total number of sequence reads generated

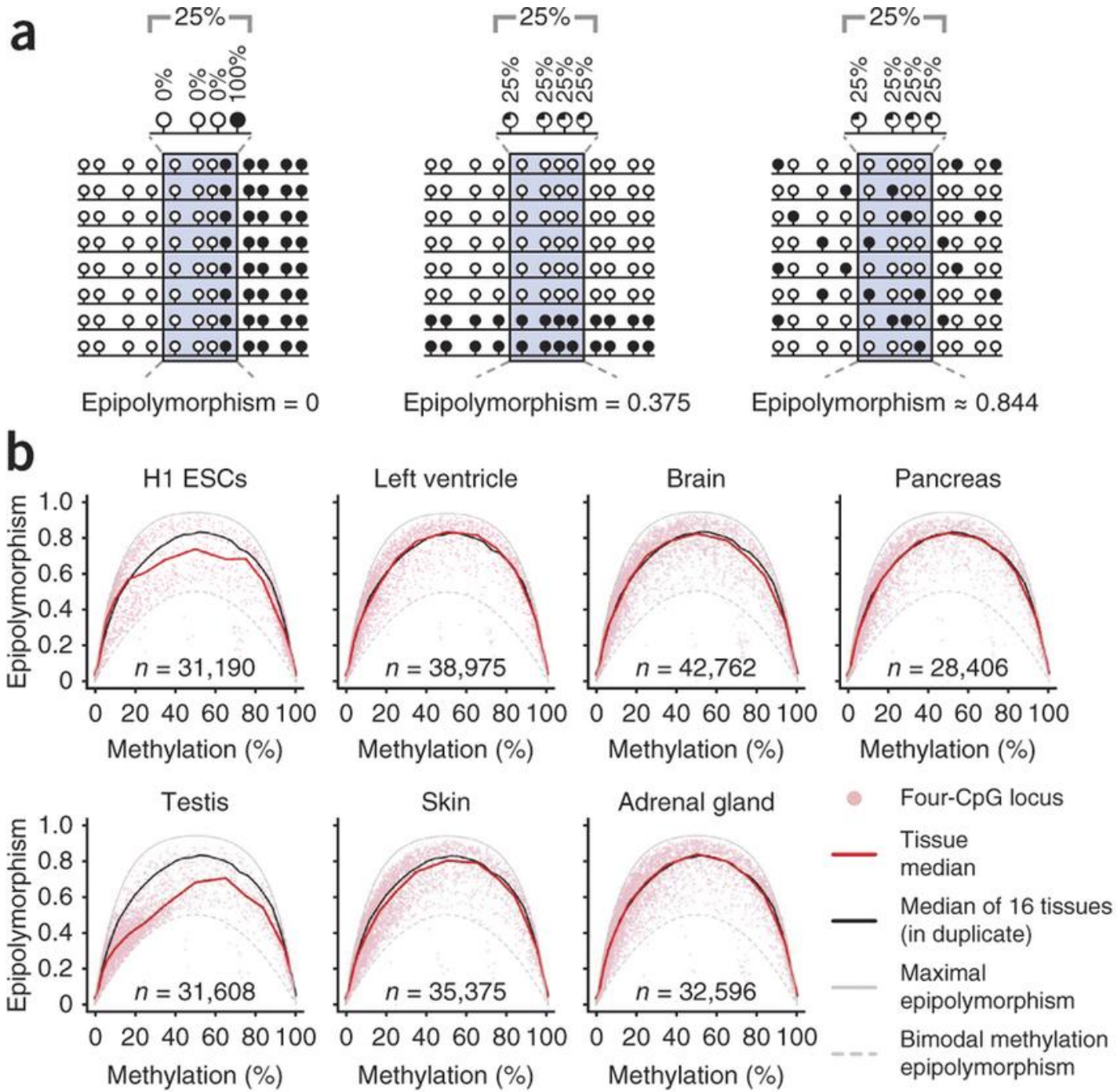


Article | Published: 14 October 2012

Epigenetic polymorphism and the stochastic formation of differentially methylated regions in normal and cancerous tissues

Gilad Landan, Netta Mendelson Cohen, Zohar Mukamel, Amir Bar, Alina Molchadsky, Ran Brosh, Shirley Horn-Saban, Daniela Amann Zalcenstein, Naomi Goldfinger, Adi Zundelevich, Einav Nili Gal-Yam, Varda Rotter & Amos Tanay

Nature Genetics **44**, 1207–1214 (2012) | Cite this article



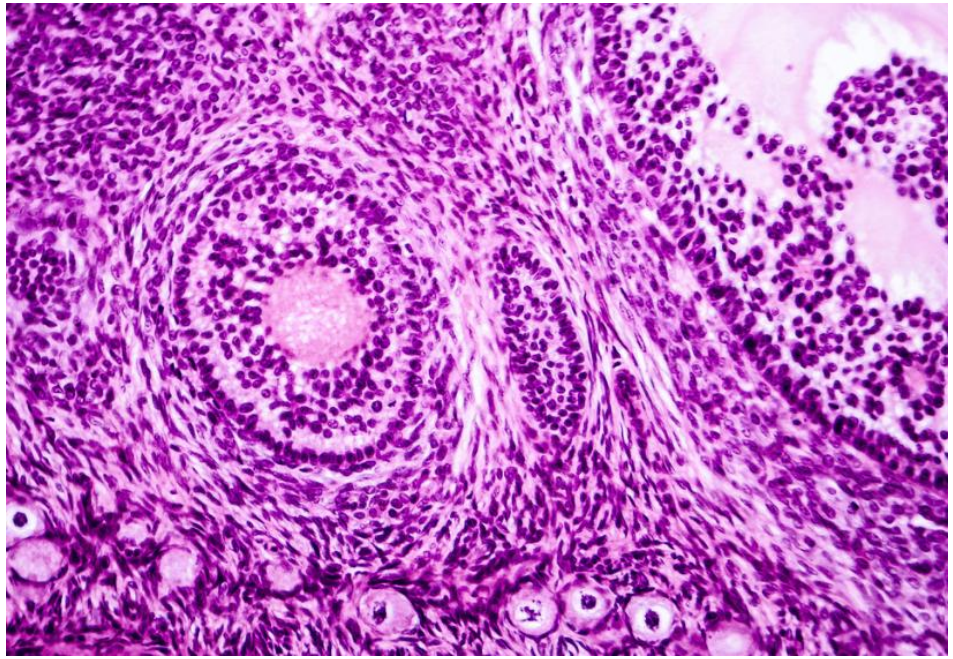
Cell Supports open access

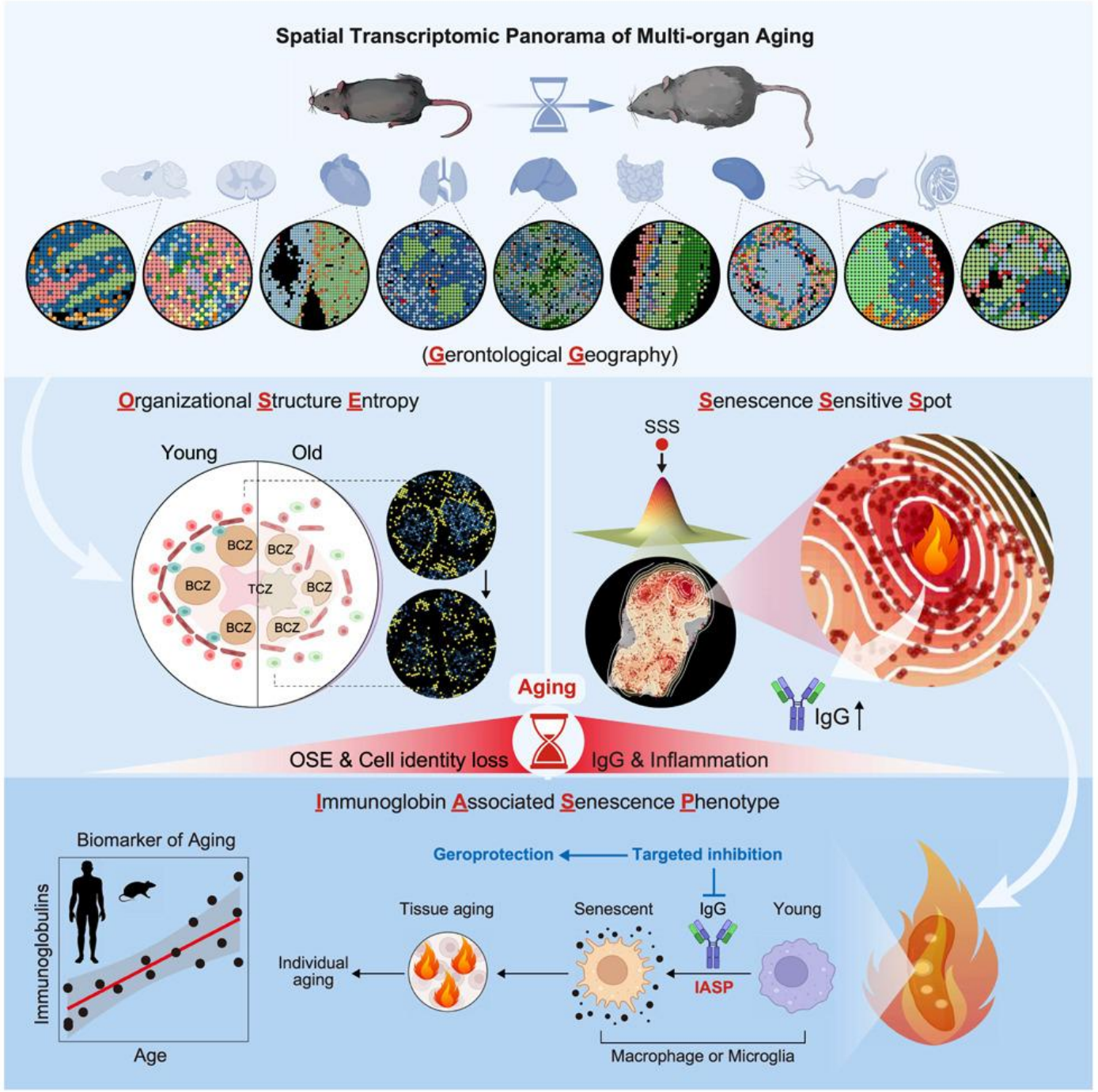
This journal Journals Publish News & events About Cell Press

RESOURCE · Volume 187, Issue 24, P7025-7044.E34, November 27, 2024 Download Full Issue

Spatial transcriptomic landscape unveils immunoglobin-associated senescence as a hallmark of aging

Shuai Ma [1,5,6,8,24,25,26] · Zhejun Ji [2,5,6,25] · Bin Zhang [1,8,25] · Lingling Geng [7,25,26] · Yusheng Cai [1,5,6,25,26] · Chao Nie [9,25,26] · Jiaming Li [3,4,8,25] · Yuesheng Zuo [3,4,8,25] · Yuzhe Sun [12,25] · Gang Xu [11,25] · Beibei Liu [3,4] · Jiaqi Ai [7] · Feifei Liu [1,5,6,10] · Liyun Zhao [7] · Jiachen Zhang [7] · Hui Zhang [1,8,10] · Shuhui Sun [1,5,6,10,26] · Haoyan Huang [7] · Yiyuan Zhang [1,5,6] · Yanxia Ye [2,5,6] · Yanling Fan [3,4] · Fangshuo Zheng [13] · Jinghao Hu [7] · Baohu Zhang [2,8] · Jingyi Li [1,5,6,8,26] · Xin Feng [14,26] · Feng Zhang [15,26] · Yuan Zhuang [16] · Tianjie Li [17] · Yang Yu [18,26] · Zhaoshi Bao [19,26] · Sipei Pan [20] · Concepcion Rodriguez Esteban [21] · Zhili Liu [9] · Haohao Deng [12] · Feng Wen [12] · Moshi Song [1,5,6,8,26] · Si Wang [7,24,26] · Guodong Zhu [22,26] · Jiayin Yang [11,26] · Tao Jiang [19,26] · Weihong Song [20,26] · Juan Carlos Izpisua Belmonte [21,26] · Jing Qu [2,5,6,8,10,24] ✉ · Weiqi Zhang [3,4,8,24] ✉ · Ying Gu [9,12,23] ✉ · Guang-Hui Liu [1,5,6,7,8,24,27] ✉ Show less





Code example that calculates discrete Entropy (the average information **before** the event)

python Copy code

```python
import math

def calculate_entropy(probabilities):
    entropy = 0
    for p in probabilities:
        entropy -= p * math.log2(p)
    return entropy

# Example probabilities
probabilities = [0.4, 0.2, 0.2, 0.2]

# Calculate entropy
entropy = calculate_entropy(probabilities)
print("Entropy:", entropy)
```

$$H(X) = \mathbb{E}\left[log\left(\frac{1}{p(X)}\right)\right] = \begin{cases} \sum_X \underbrace{log\left(\frac{1}{p(X)}\right)}_{f(x)} p(x) & X \text{ is discrete case} \\ \int_X \underbrace{log\left(\frac{1}{p(X)}\right)}_{f(x)} p(x)\, dx & X \text{ is continuous case} \end{cases}$$

Remember that

$$log\left(\frac{1}{p(x)}\right) = log(p(x)^{-1}) = -log(p(x))$$

Entropy will be use later as an objective

This time try it yourself

Suppose you have a box containing five different colored balls: two red balls, one blue ball, one green ball, and one yellow ball. What is the entropy of the ball distribution in the box?
(use base e)

$$H(X) = \mathbb{E}\left[log\left(\frac{1}{p(X)}\right)\right] = \begin{cases} \sum_X \underbrace{log\left(\frac{1}{p(X)}\right)}_{f(x)} p(x) & X \text{ is discrete case} \\ \int_X \underbrace{log\left(\frac{1}{p(X)}\right)}_{f(x)} p(x)\, dx & X \text{ is continuous case} \end{cases}$$

red	2	2/5
blue	1	1/5
green	1	1/5
yellow	1	1/5

Use python to solve this.

Try not to look at the code and do it yourself first. If you are stuck, then look.

Solution

Suppose you have a box containing five different colored balls: two red balls, one blue ball, one green ball, and one yellow ball. What is the entropy of the ball distribution in the box? (use base e)

$$H(X) = \mathbb{E}\left[log\left(\frac{1}{p(X)}\right)\right] = \begin{cases} \sum_X \underbrace{log\left(\frac{1}{p(X)}\right)}_{f(x)} p(x) & X \text{ is discrete case} \\ \int_X \underbrace{log\left(\frac{1}{p(X)}\right)}_{f(x)} p(x)\, dx & X \text{ is continuous case} \end{cases}$$

red	2	2/5
blue	1	1/5
green	1	1/5
yellow	1	1/5

$$H(X) = \underbrace{\frac{2}{5} log_e\left(\frac{1}{2/5}\right)}_{\text{red}} + \underbrace{\frac{1}{5} log_e\left(\frac{1}{1/5}\right)}_{\text{blue}} + \underbrace{\frac{1}{5} log_e\left(\frac{1}{1/5}\right)}_{\text{green}} + \underbrace{\frac{1}{5} log_e\left(\frac{1}{1/5}\right)}_{\text{yellow}} \approx 1.33$$

Fun fact, I ask Chat GPT to answer this question too, it got it wrong 😂

red	2	2/5
blue	1	1/5
green	1	1/5
yellow	1	1/5

In this case, there are five possible outcomes (the five different colored balls). To calculate the probabilities, divide the number of each colored ball by the total number of balls.

Total number of balls = 2 (red) + 1 (blue) + 1 (green) + 1 (yellow) = 5

The probabilities are as follows:

P(red) = 2/5 = 0.4
P(blue) = 1/5 = 0.2
P(green) = 1/5 = 0.2
P(yellow) = 1/5 = 0.2

Now, substitute the probabilities into the entropy formula:

Entropy = $-[(0.4 * \log_2(0.4)) + (0.2 * \log_2(0.2)) + (0.2 * \log_2(0.2)) + (0.2 * \log_2(0.2))]$

Calculating each term:

Entropy = -[(0.4 * -2) + (0.2 * -2.3219) + (0.2 * -2.3219) + (0.2 * -2.3219)]

Entropy = -[-0.8 + (-0.4644) + (-0.4644) + (-0.4644)]

Entropy = -[-2.1928]

Entropy ≈ 2.1928

My calculator result

```
octave:17> -0.4*log2(0.4) =  0.52877
```

Took me a while to find the mistake. But it calculated

log2(0.4) as 2 (wrong)

log2(4) = 2 <- I think it was confused by this

This tells you that ChatGPT is useful, but it makes a lot of mistakes.

My calculator result

```
octave:19> (2/5)*log2(5/2) + (3/5)*log2(5) =  1.9219
```

It would be even funnier if I am making fun of Chat GPT, but I was the one that's wrong

Practice Calculating Entropy

1. Load the file `time_until_phone_drop.csv` and identify $p(x)$.
2. Calculate the entropy of this data using Numpy automatically Integration with $p(x)$.

$$H(X) = \int_{\mathcal{X}} p(x) \log\left(\frac{1}{p(x)}\right) dx.$$

3. Calculate again the entropy of this data using sampling $p(x)$ approximation of integrals.

$$H(X) = \int_{\mathcal{X}} p(x) \log\left(\frac{1}{p(x)}\right) dx \approx \frac{1}{n} \sum_{i} \log\left(\frac{1}{p(x)}\right)$$

Interest fact about Entropy, Equal probability always yields the largest entropy

```python
#!/usr/bin/env python
#
import math
import numpy as np
#
def calculate_entropy(P):
    entropy = 0
    for p in P:
        entropy -= p * math.log2(p)
    return entropy
#
H0 = 0
for i in range(300000):
    v = np.random.rand(4)
    P = v/np.sum(v) # as probability
#
    H = calculate_entropy(P)
    if H0 < H:
        print(np.round(P,2), "%.3f"%H)
        H0 = H
#
```

```
[0.32 0.32 0.09 0.26] 1.873
[0.31 0.36 0.17 0.17] 1.918
[0.29 0.28 0.12 0.31] 1.924
[0.27 0.26 0.25 0.23] 1.998
[0.26 0.24 0.24 0.25] 1.999
[0.26 0.25 0.24 0.25] 2.000
[0.25 0.25 0.25 0.24] 2.000
[0.25 0.25 0.25 0.25] 2.000
[0.25 0.25 0.25 0.25] 2.000
```

In this code, we generate 300,000 random probabilities and only print out the probability if it has a larger entropy compare to the previous largest entropy.

This code tells us that

- The entropy of a system is largest when
- All events are equally likely.

From this, we learn how to calculate the maximum possible entropy value with k possible outcomes.

- Make each probability outcome to be 1/k

Normalizing Entropy

But if I tell you that the entropy of a system is 0.9

- It is not clear if we should consider 0.9 a large or a small entropy.
- Depending on the events or base we used, 0.9 could be large or small.

Therefore, it is more useful to scale the entropy value to be between 0 and 1.

- In this case 0 would be the minimum entropy
- And 1 would be maximum entropy

We call this version of the entropy the **Normalized Entropy.**

- To calculate the normalized entropy, we simply divide our entropy to the maximum entropy

$$H_n = \text{entropy} / \text{maximum entropy}$$

The concept of Entropy is really useful !!!

Understanding Entropy leads us to a way to measure relationships Between Distributions

If you have distributions, $p(x) \sim X$ and $p(y) \sim Y$, what query function can you ask about them?

1. How similar or different are the probability distributions from the 2 datasets?
   a. Cross Entropy
   b. KL Divergence
   c. MMD
   d. F-divergence
   e. Integral Probability Metric
   f. And many more.....

2. How dependent are the 2 datasets?
   a. Mutual Information
   b. Hilbert Schmidt Independence Criterion
   c. Chi-Square Test of Independence
   d. And many more.....

In this class, we are going to cover the ones in green.
We will cover the ones in red in Machine learning II.

Cross Entropy (measuring the difference between two distributions)

$$H(p,q) = \mathbb{E}_{p(x)}\left[\log\left(\frac{1}{q(x)}\right)\right] = \begin{cases} \sum_{\mathcal{X}} \log\left(\frac{1}{q(x)}\right) p(x) & X \text{ is discrete} \\ \int_{\mathcal{X}} \log\left(\frac{1}{q(x)}\right) p(x)\, dx & X \text{ is continuous} \end{cases}$$

Cross entropy is the expected information of q(x) using the probability of p(x)

Example: Different

	x=0	x=1
p(x)	0.1	0.9
q(x)	0.7	0.3

$$\begin{aligned} H(p,q) &= p(0)\log\left(\frac{1}{q(0)}\right) + p(1)\log\left(\frac{1}{q(1)}\right) \\ &= 0.1\log\left(\frac{1}{0.7}\right) + 0.9\log\left(\frac{1}{0.3}\right) \approx 1.12 \end{aligned}$$

Big difference

Example: Small difference

	x=0	x=1
p(x)	0.1	0.9
q(x)	0.2	0.8

$$\begin{aligned} H(p,q) &= p(0)\log\left(\frac{1}{q(0)}\right) + p(1)\log\left(\frac{1}{q(1)}\right) \\ &= 0.1\log\left(\frac{1}{0.2}\right) + 0.9\log\left(\frac{1}{0.8}\right) \approx 0.36 \end{aligned}$$

Small difference

Besides the mean squared error MSE

$$f(w) = \frac{1}{2}\sum_{i}^{N}(w^\top \phi(x_i) - y_i)^2$$

Cross Entropy is the **most** used objective in machine learning. This is an objective you have to memorize.

You will definitely need to know how to take the derivative of this objective later.

Take a sec and think how you can approximate the cross entropy with sampling ?

$$H(p,q) \approx \frac{1}{n}\sum \log\left(\frac{1}{q(x)}\right)$$

Cross Entropy (measuring the difference between two distributions)

Example: Different

	x=0	x=1
p(x)	0.1	0.9
q(x)	0.7	0.3

$$H(p,q) = p(0)\log\left(\frac{1}{q(0)}\right) + p(1)\log\left(\frac{1}{q(1)}\right)$$
$$= 0.1\log\left(\frac{1}{0.7}\right) + 0.9\log\left(\frac{1}{0.3}\right) \approx 1.12$$

Example: Small Difference

	x=0	x=1
p(x)	0.1	0.9
q(x)	0.2	0.8

$$H(p,q) = p(0)\log\left(\frac{1}{q(0)}\right) + p(1)\log\left(\frac{1}{q(1)}\right)$$
$$= 0.1\log\left(\frac{1}{0.2}\right) + 0.9\log\left(\frac{1}{0.8}\right) \approx 0.36$$

Example: No Difference

	x=0	x=1
p(x)	0.1	0.9
q(x)	0.1	0.9

Same as just entropy H(X)

$$H(p,q) = p(0)\log\left(\frac{1}{q(0)}\right) + p(1)\log\left(\frac{1}{q(1)}\right)$$
$$= 0.1\log\left(\frac{1}{0.1}\right) + 0.9\log\left(\frac{1}{0.9}\right) \approx 0.33$$

Very Important to notice:

- As the 2 distributions p, q become more similar, their difference gets smaller
- When p, q becomes equal, Cross Entropy reduces down to regular entropy.
- Cross Entropy is always larger than Entropy
- Cross Entropy is the total entropy after adding q(x) into the system !!!
- If you want to measure the addition entropy added by q(x), you can simply do

$$\text{Extra Entropy from } q(x) = H(p,q) - H(p)$$

The name for this excessive entropy is called

The KL Divergence

KL Divergence (The excessive entropy from Cross Entropy)

$$\text{Extra Entropy from } q(x) = H(p,q) - H(p)$$

$$\begin{aligned}
D(p||q) &= H(p,q) - H(p,p) \\
&= \sum_i p(x) log\left(\frac{1}{q(x)}\right) - \sum_i p(x) log\left(\frac{1}{p(x)}\right) \\
&= \sum_i p(x) log\left(\frac{1}{q(x)}\right) - p(x) log\left(\frac{1}{p(x)}\right) \\
&= \sum_i p(x)\left[log\left(\frac{1}{q(x)}\right) - log\left(\frac{1}{p(x)}\right)\right] \\
&= \sum_i p(x)\left[log\left(\frac{1}{q(x)}\right) + log\left(p(x)\right)\right] \\
&= \sum_i p(x)\left[log\left(\frac{p(x)}{q(x)}\right)\right]
\end{aligned}$$

The standard equation for KL divergence in textbooks

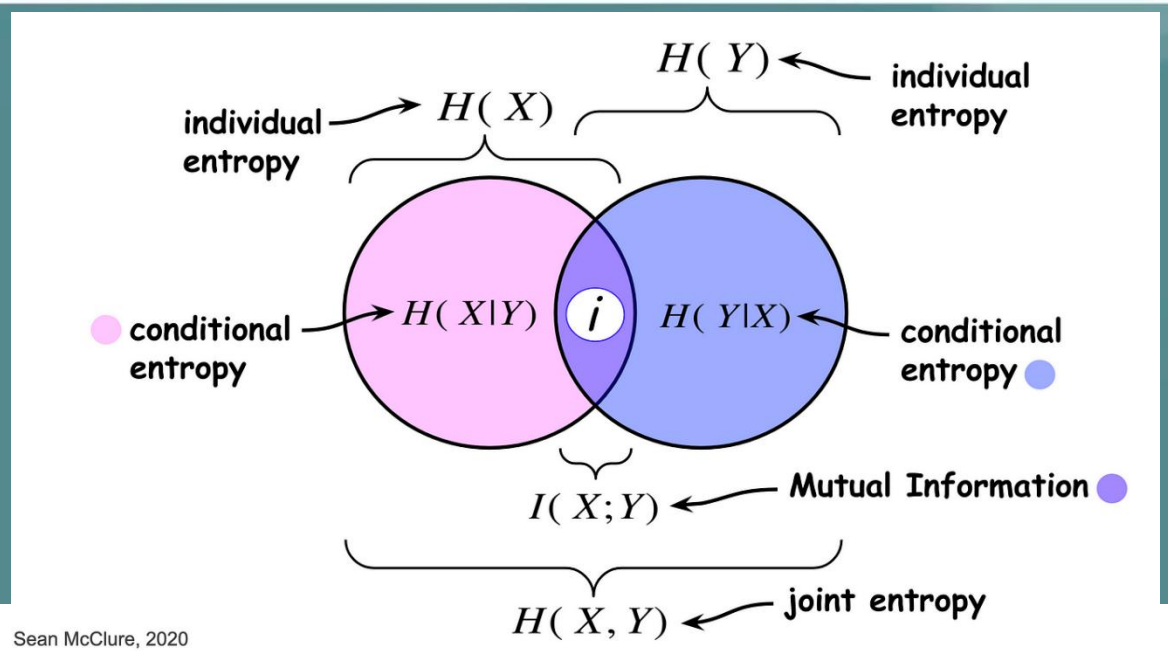
- Unlike Cross entropy, KL Divergence starts from 0 (because it removes Entropy)
- It is a common way to measure the distance between two distributions where
  - 0 means no difference between The two distributions
  - As the value increase, it implies that the difference between the 2 distributions, p and q is also larger.
- It is also a very common objective used in machine learning.

Mutual Information (MI) measure the dependence between 2 distributions.

Remember that $p(x, y) = p(x|y)p(y)$
But if x, y are independence, then it becomes $p(x, y) = p(x)p(y)$
Mutual Information is the KL divergence between $p(x, y)$ and $p(x)p(y)$

$$I(X;Y) = D(p(x,y)||p(x)p(y))$$
$$= \sum_{y}\sum_{x} p(x,y)\log\left(\frac{p(x,y)}{p(x)p(y)}\right)$$

Where $I = 0$ implies $p(x, y) = p(x)p(y)$ and complete independence.
The larger the mutual information, the larger the dependence between $p(x)$ and $p(y)$



Practice Computing the Cross Entropy, KL Divergence, and Mutual information.

	X = 0	X = 1
Y = 0	0.1	0.2
Y = 1	0.4	0.1
Y = 2	0.1	0.1

For the probability table on the left, use Python with base e to calculate the

1. Entropy of $p(x)$.
2. Entropy of $p(y)$.
3. Compute the Cross-Entropy : $H(p(y|x=0), p(y|x=1))$.
4. KL Divergence $KL(p(y|x=0)||p(y|x=1))$.
5. Entropy of $H(p(y|x=0))$.
6. Mutual Information $I(X;Y)$.
7. You cannot calculate $H(X), H(X,Y), I(X,Y)$ when there's a 0 in the probability. Can you explain why?

Solution

1. $H(X) = -0.6\log(0.6) - 0.4\log(0.4)$
2. $H(Y) = -0.3\log(0.3) - 0.5\log(0.5) - 0.3\log(0.3)$
3. $H(Y_{X=0}, Y_{X=1}) = -\frac{1}{6}\log(\frac{2}{4}) - \frac{4}{6}\log(\frac{1}{4}) - \frac{1}{6}\log(\frac{1}{4})$
4. $KL(Y_{X=0}, Y_{X=1}) = \frac{1}{6}\log(\frac{1}{6}\ \frac{4}{2}) + \frac{4}{6}\log(\frac{4}{6}\ \frac{4}{1}) + \frac{1}{6}\log(\frac{1}{6}\ \frac{4}{1})$
5. Here's all the marginal distribution.

$$p(x=0) = 0.6 \quad p(x=1) = 0.4$$
$$p(y=0) = 0.3 \quad p(y=1) = 0.5 \quad p(y=1) = 0.2$$

If $p(x)$ and $p(y)$ are independent then $p(x, y)$ would be

$p(x=0)p(y=0) = 0.6 \cdot 0.3 = 0.18$	$p(x=1)p(y=0) = 0.4 \cdot 0.3 = 0.12$
$p(x=0)p(y=1) = 0.6 \cdot 0.5 = 0.30$	$p(x=1)p(y=1) = 0.4 \cdot 0.5 = 0.20$
$p(x=0)p(y=2) = 0.6 \cdot 0.2 = 0.12$	$p(x=1)p(y=2) = 0.4 \cdot 0.2 = 0.08$

	X = 0	X = 1
Y = 0	0.1	0.2
Y = 1	0.4	0.1
Y = 2	0.1	0.1

$$H(p,q) = p(0)\log\left(\frac{1}{q(0)}\right) + p(1)\log\left(\frac{1}{q(1)}\right)$$
$$= 0.1\log\left(\frac{1}{0.7}\right) + 0.9\log\left(\frac{1}{0.3}\right) \approx 1.12$$

$$D(p||q) = \sum_i p(x)\left[log\left(\frac{p(x)}{q(x)}\right)\right]$$

Mutual Information Continued...

$$
\begin{aligned}
I(X,Y) = {} & p(x=0,y=0)\log\left(\frac{p(x=0,y=0)}{p(x=0)p(y=0)}\right) + p(x=1,y=0)\log\left(\frac{p(x=1,y=0)}{p(x=1)p(y=0)}\right) \\
& + p(x=0,y=1)\log\left(\frac{p(x=0,y=1)}{p(x=0)p(y=1)}\right) + p(x=1,y=1)\log\left(\frac{p(x=1,y=1)}{p(x=1)p(y=1)}\right) \\
& + p(x=0,y=2)\log\left(\frac{p(x=0,y=2)}{p(x=0)p(y=2)}\right) + p(x=1,y=2)\log\left(\frac{p(x=1,y=2)}{p(x=1)p(y=2)}\right)
\end{aligned}
$$

Plugging the numbers, we get

$$
\begin{aligned}
I(X,Y) = {} & 0.1\log\left(\frac{0.1}{0.18}\right) + 0.2\log\left(\frac{0.2}{0.12}\right) \\
& + 0.4\log\left(\frac{0.4}{0.30}\right) + 0.1\log\left(\frac{0.1}{0.20}\right) \\
& + 0.1\log\left(\frac{0.1}{0.12}\right) + 0.1\log\left(\frac{0.1}{0.08}\right)
\end{aligned}
$$

Example 1: Independent vs. Dependent Events (Dice Rolls)

- Scenario A: Independent Dice Rolls
  - Let X be the outcome of one fair six-sided die, and Y be the outcome of a second, separate fair six-sided die.
  - Knowing the value of X (e.g., you rolled a 4) gives you no information about Y.
  - Mutual Information: I(X, Y) = 0
- Scenario B: Dependent Events
  - Let X be the outcome of a fair six-sided die, and Y be a variable that is 0 if X is even and 1 if X is odd.
  - Knowing the value of X (e.g., you rolled a 4) tells you for certain that Y is 0.
  - Mutual Information: I(X, Y) will be a positive value (specifically, it would equal the entropy of Y, which is 1 bit), indicating a strong relationship.
- Comparison: The MI in Scenario A (I(X, Y) = 0) is lower than the MI in Scenario B (I(X, Y) = 1 bit), correctly showing that the variables are related in B but not in A.

Example 2: Feature Selection (Housing Data)

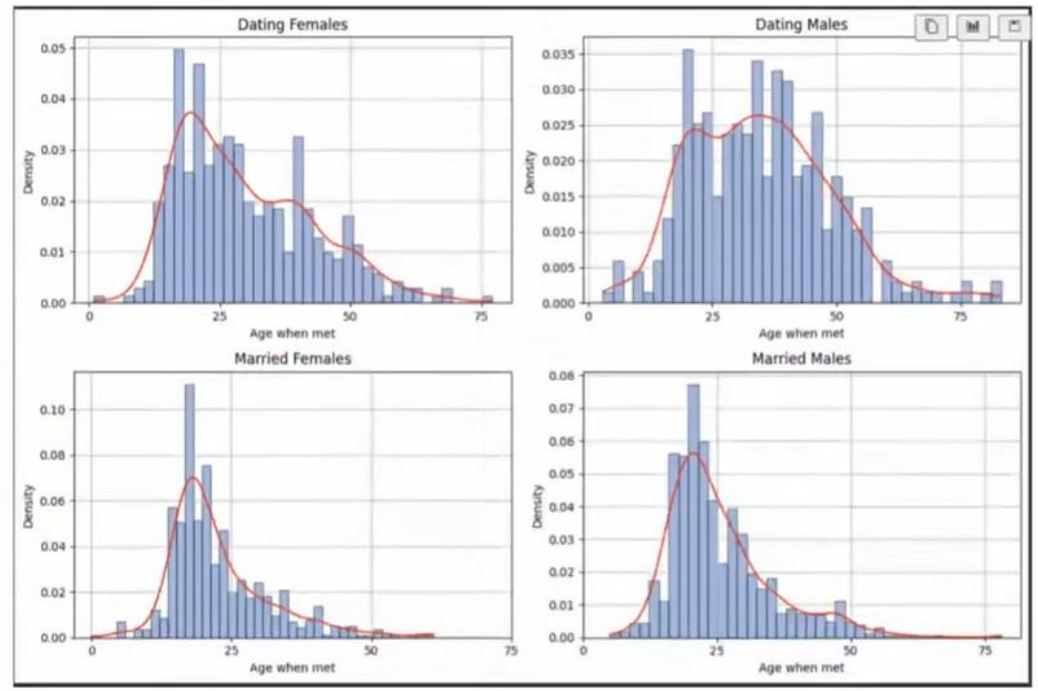
In a machine learning task to predict house prices, you might calculate the mutual information between the target variable (SalePrice) and various features:

Feature	Mutual Information with `SalePrice`	Interpretation
`Square Footage`	High (e.g., 0.85)	A large reduction in price uncertainty; strong relationship.
`Exterior Quality`	Medium (e.g., 0.60)	Knowing the quality helps predict the price, but less so than square footage.
`Fireplace`	Low (e.g., 0.10)	Some information is shared, but knowing if there is a fireplace only slightly reduces price uncertainty.
`Latitude` (for a single city)	Very Low/Zero (e.g., 0.01)	Location within the city provides almost no information about the price.

Comparison: By comparing the MI scores, you can determine that Square Footage is the most informative feature for predicting SalePrice, followed by Exterior Quality, and so on. This ranking is used to select the best features for a predictive model.

Practice Calculating KL Divergence with KDE

1. Load the files from the folder `Age meeting a partner`.
2. Notice that the distributions between male/female marriage distribution look similar to each other while the dating profiles look different.
3. This implies that if you find the distance between marriage distributions, it should be shorter compared to the dating distribution.
4. Use KDE to identify the 2 marriage distributions $p_1(x), p_2(x)$, as well as the male dating distribution $p_3(x)$.
5. Calculate the KL divergence and confirm that $KL(p_1(x)||p_2(x)) \leq KL(p_1(x)||p_3(x))$.



Solution

Parametric Vs Non-parametric Classification Models

We can divide classification models into two broad categories.

- Parametric Models (Where we are trying to optimize for some parameter)
  - Logistic Regression (Optimize for θ)

$$\min_{\theta} \quad \underbrace{\frac{1}{n}\sum_{i=1}^{n}\left[p_i(e_1)\log\left(\frac{1}{\frac{1}{1+e^{-x_i^\top\theta}}}\right)+p_i(e_2)\log\left(\frac{1}{1-\frac{1}{1+e^{-x_i^\top\theta}}}\right)\right]}_{\text{objective function as } \mathcal{L}} \quad \text{where} \quad \theta=\begin{bmatrix}\theta_1\\ \theta_2\\ \theta_3\end{bmatrix}$$

  - SVM (Optimize for w)

$$\min_{w} \quad ||w||_2^2 - \sum_{i=1}^{n} y_i(w^\top x_i - b).$$

- Non-parametric Models (There isn't a set of parameter to optimize)
  - K Nearest Neighbor (Finding k nearest samples and set the label to them
  - **Decision Tree** (This is what we are going to learn today)

Entropy and Conditional Entropy

To understand decision trees, you must first know 2 concepts really well.

- Entropy

$$H(X) = \mathbb{E}\left[log\left(\frac{1}{p(X)}\right)\right] = \sum_{x \in \mathcal{X}} log\left(\frac{1}{p(X)}\right) p(x) \tag{2}$$

  - Entropy is the **average** information for all the possible outcomes.

- Conditional Entropy

$$H(Y|X) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y)\ log(p(y|x)) \tag{3}$$

  - Conditional Entropy is the **average** information for all the possible outcomes **after** you were told $X = x$ happened.
  - Using Bayes Theorem where $p(x,y) = p(y|x)p(x)$, we can rewrite the conditional entropy equation as

$$H(Y|X) = -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(y|x)p(x)\ log(p(y|x)) \tag{4}$$

$$= -\sum_{x \in \mathcal{X}} p(x) \sum_{y \in \mathcal{Y}} p(y|x)\ log(p(y|x)). \tag{5}$$

Eq. (4) is going to be the version we use for conditional entropy.

Quick Review of Entropy

- Calculate the entropy for these outcomes
- Let the random variable of play golf be Y
- Use log base 2.

$$H(X) = \mathbb{E}\left[log\left(\frac{1}{p(X)}\right)\right] = \sum_{x \in \mathcal{X}} log\left(\frac{1}{p(X)}\right) p(x)$$

Play Golf(14)	
Yes	No
9	5

Quick Review of Entropy

- Calculate the entropy for these outcomes
- Use log base 2.

Play Golf(14)	
Yes	No
9	5

$$H(X) = \mathbb{E}\left[log\left(\frac{1}{p(X)}\right)\right] = \sum_{x \in \mathcal{X}} log\left(\frac{1}{p(X)}\right) p(x)$$

$$H(Y) = -\frac{9}{14}\log\left(\frac{9}{14}\right) - \frac{5}{14}\log\left(\frac{5}{14}\right) \approx 0.94.$$

Practice calculating Conditional Entropy

- Calculate the conditional entropy for these outcomes
- Let x be the Outlook
- Let y be the Play golf decision
- Use log base 2.

		PlayGolf(14)		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5

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

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

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

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

		PlayGolf(14)		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5

Example Calculating Entropy and Conditional Entropy

$$\begin{aligned}H(Y|X) = &-p(x=\text{sunny})\,[p(y=\text{yes}|x=\text{sunny})\log_2(p(y=\text{yes}|x=\text{sunny})) + p(y=\text{no}|x=\text{sunny})\log_2(p(y=\text{no}|x=\text{sunny}))]\\ &- p(x=\text{overcast})\,[p(y=\text{yes}|x=\text{overcast})\log_2(p(y=\text{yes}|x=\text{overcast})) + p(y=\text{no}|x=\text{overcast})\log_2(p(y=\text{no}|x=\text{overcast}))]\\ &- p(x=\text{Rainy})\,[p(y=\text{yes}|x=\text{Rainy})\log_2(p(y=\text{yes}|x=\text{Rainy})) + p(y=\text{no}|x=\text{Rainy})\log_2(p(y=\text{no}|x=\text{Rainy}))]\end{aligned}$$

$$\begin{aligned}H(Y|X) = &-\frac{5}{14}\left[\frac{3}{5}\log_2(\frac{3}{5}) + \frac{2}{5}\log_2(\frac{2}{5})\right]\\ &- \frac{4}{14}\,[1\log_2(1) + 0\log_2(0)]\\ &- \frac{5}{14}\left[\frac{2}{5}\log_2(\frac{2}{5}) + \frac{3}{5}\log_2(\frac{3}{5})\right] = 0.347 - 0 + 0.347 = 0.69\end{aligned}$$

Practice calculating Conditional Entropy

- Calculate the conditional entropy for these outcomes
- Let z be the Temperature
- Let y be the Play golf decision
- Use log base 2.

		PlayGolf(14)		
		Yes	No	
Temperature	Hot	2	2	4
	Cold	3	1	4
	Mild	4	2	6

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

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

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

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

		PlayGolf(14)		
		Yes	No	
Temperature	Hot	2	2	4
	Cold	3	1	4
	Mild	4	2	6

Example Calculating Entropy and Conditional Entropy

$$\begin{aligned} H(Y|Z) = &-p(z=\text{hot})\,[p(y=\text{yes}|z=\text{hot})\log_2(p(y=\text{yes}|z=\text{hot})) + p(y=\text{no}|z=\text{hot})\log_2(p(y=\text{no}|z=\text{hot}))] \\ &- p(x=\text{cold})\,[p(y=\text{yes}|z=\text{cold})\log_2(p(y=\text{yes}|z=\text{cold})) + p(y=\text{no}|z=\text{cold})\log_2(p(y=\text{no}|z=\text{cold}))] \\ &- p(x=\text{mild})\,[p(y=\text{yes}|z=\text{mild})\log_2(p(y=\text{yes}|z=\text{mild})) + p(y=\text{no}|z=\text{mild})\log_2(p(y=\text{no}|z=\text{mild}))] \end{aligned}$$

$$\begin{aligned} H(Y|Z) = &-\frac{4}{14}\left[\frac{1}{2}\log_2(\frac{1}{2}) + \frac{1}{2}\log_2(\frac{1}{2})\right] \\ &-\frac{4}{14}\left[\frac{3}{4}\log_2(\frac{3}{4}) + \frac{1}{4}\log_2(\frac{1}{4})\right] \\ &-\frac{6}{14}\left[\frac{4}{6}\log_2(\frac{4}{6}) + \frac{2}{6}\log_2(\frac{2}{6})\right] = 0.286 + 0.232 + 0.391 = 0.91 \end{aligned}$$

Using Entropy and Conditional Entropy together.

Entropy $H(Y)$

- Is the **average** amount of information for an event of multiple possible outcomes.
- Entropy is based on chaos. The more random and chaotic, the more information exists in the system.

Cross Entropy $H(p, q)$

- Given the original randomness of p, Cross Entropy is the additional chaos added when distribution q is introduced.

Conditional Entropy $H(Y|X)$

- after X is revealed, there's less chaos and less information in the system Y.
- $H(Y|X)$ is the leftover information after X is revealed.
- It is always going to be equal to or less than the original amount of information,

$$H(Y|X) \leq H(Y). \tag{1}$$

The information gained by knowing X is the difference between the original $H(Y)$ and the conditional $H(Y|X)$

$$\textbf{Information Gained From } X = \Delta H_{Y|X=x} = H(Y) - H(Y|X = x). \tag{2}$$

- The bigger the $\Delta H_{Y|X=x}$, the more information X give us about Y.
- In general, we call the difference between original and after knowing X as **Information Gain**.

Which factor gives us more information on golf playing?

We now have 3 random variables X, Y, Z.

- $H(Y)$: Is the entropy of playing golf without any additional information
- $H(Y|X)$: Is the entropy of playing golf given weather outlook.
- $H(Y|Z)$: Is the entropy of playing golf given temperature.

Play Golf(14)	
Yes	No
9	5

		PlayGolf(14)		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5

		PlayGolf(14)		
		Yes	No	
Temperature	Hot	2	2	4
	Cold	3	1	4
	Mild	4	2	6

Which factor gives us more information on golf playing?

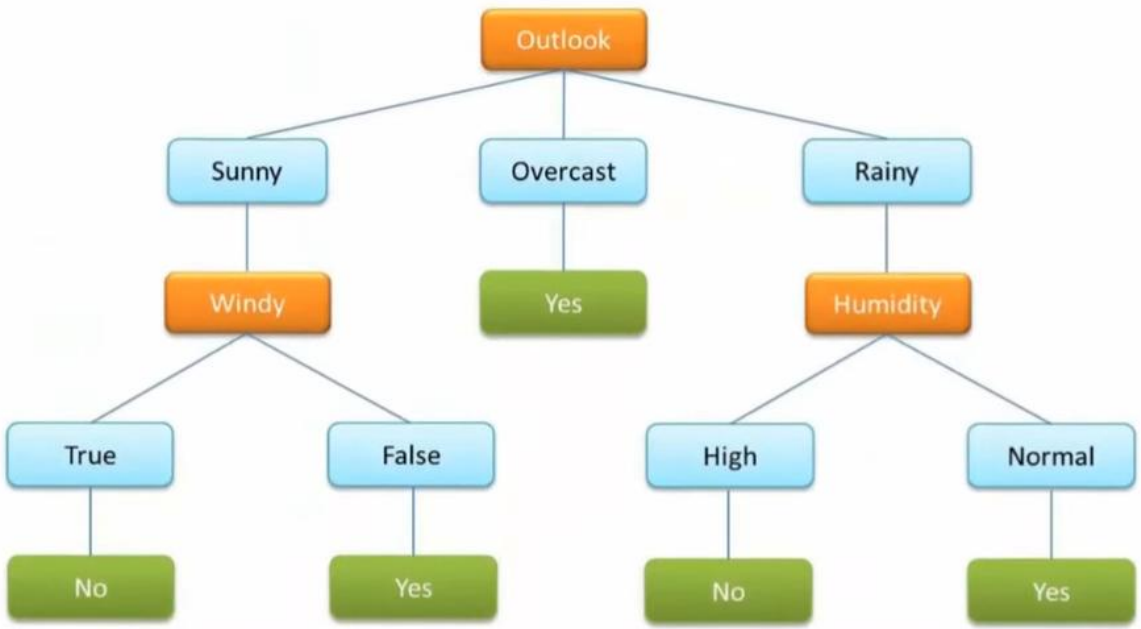
$$H(Y) = -\frac{9}{14}\log\left(\frac{9}{14}\right) - \frac{5}{14}\log\left(\frac{5}{14}\right) \approx 0.94.$$

$$\begin{aligned} H(Y|X) = &-\frac{5}{14}\left[\frac{3}{5}\log_2(\frac{3}{5}) + \frac{2}{5}\log_2(\frac{2}{5})\right] \\ &-\frac{4}{14}\left[1\log_2(1) + 0\log_2(0)\right] \\ &-\frac{5}{14}\left[\frac{2}{5}\log_2(\frac{2}{5}) + \frac{3}{5}\log_2(\frac{3}{5})\right] = 0.347 - 0 + 0.347 = 0.69 \end{aligned}$$

$$\begin{aligned} H(Y|Z) = &-\frac{4}{14}\left[\frac{1}{2}\log_2(\frac{1}{2}) + \frac{1}{2}\log_2(\frac{1}{2})\right] \\ &-\frac{4}{14}\left[\frac{3}{4}\log_2(\frac{3}{4}) + \frac{1}{4}\log_2(\frac{1}{4})\right] \\ &-\frac{6}{14}\left[\frac{4}{6}\log_2(\frac{4}{6}) + \frac{2}{6}\log_2(\frac{2}{6})\right] = 0.286 + 0.232 + 0.391 = 0.91 \end{aligned}$$

We are now ready to talk about the Decision Tree Classifier

- Given past historical data of golf playing
- Based on various weather patterns
- The decision tree algorithm generates a Tree of questions that leads to a golf decision.



Attributes				Classes
Outlook	Temperature	Humidity	Windy	Play Golf
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	No
Overcast	Hot	High	FALSE	Yes
Sunny	Mild	High	FALSE	Yes
Sunny	Cool	Normal	FALSE	Yes
Sunny	Cool	Normal	TRUE	No
Overcast	Cool	Normal	TRUE	Yes
Rainy	Mild	High	FALSE	No
Rainy	Cool	Normal	FALSE	Yes
Sunny	Mild	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Sunny	Mild	High	TRUE	No

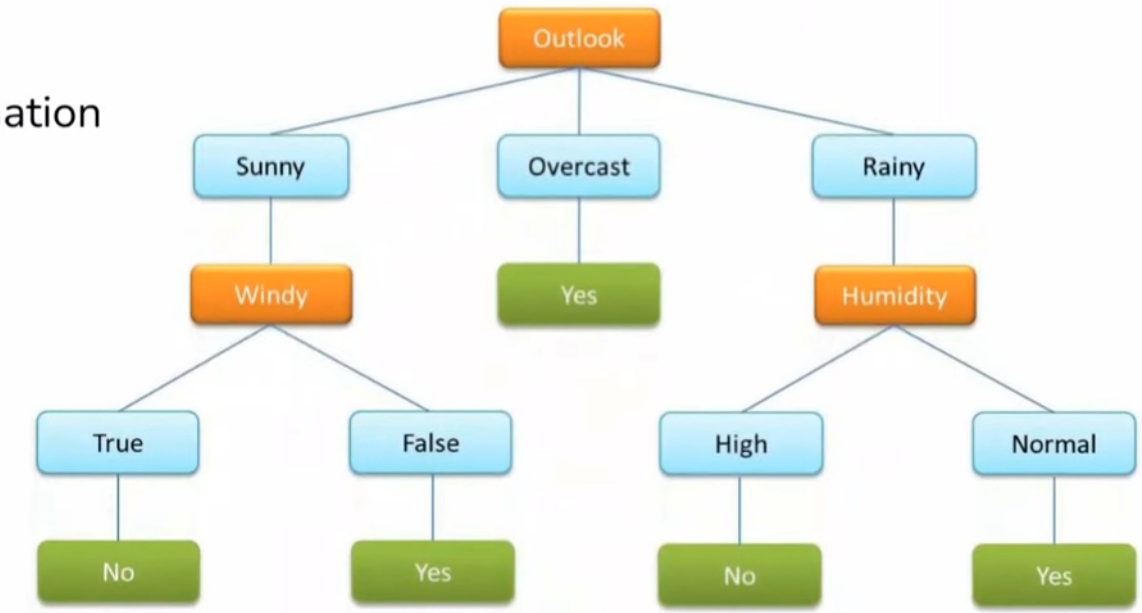
At each layer of the tree

- The decision tree picks an attribute/feature to move onto the next layer.
- The natural question arises
- How do we know which feature should be picked first?
- Answer:

The feature that gives us the most information

We already know how to calculate the information gained from each feature.

At each layer, we simply pick the feature with the highest information Gain.



Which features should we pick first?

Given the data table, we can calculate the conditional entropy for each feature. The original entropy, as we previously calculated, is

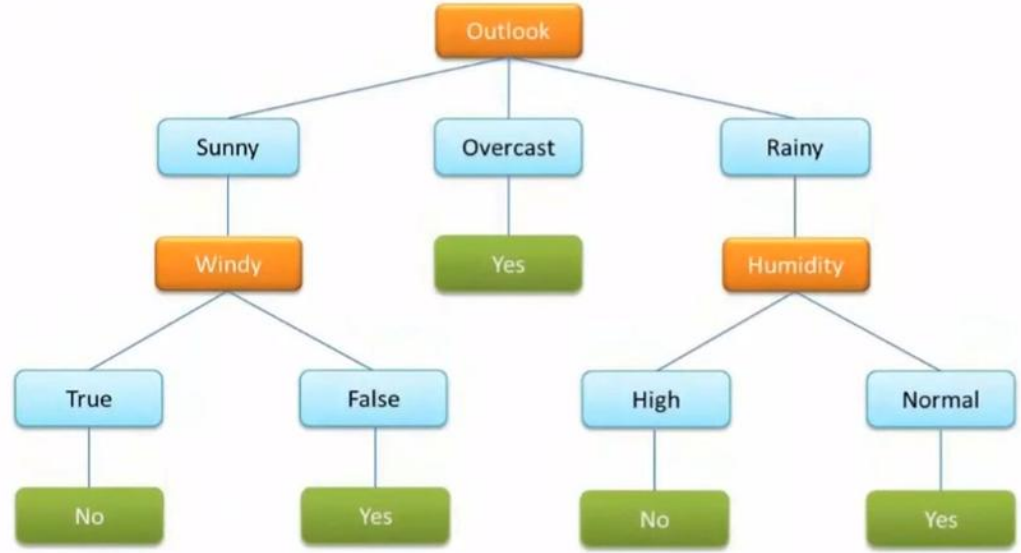
$$H(Y) = 0.94. \tag{1}$$

The conditional entropy are

$$H(Y|\text{outlook}) = 0.693 \tag{2}$$
$$H(Y|\text{Temp}) = 0.911 \tag{3}$$
$$H(Y|\text{Humidity}) = 0.788 \tag{4}$$
$$H(Y|\text{Windy}) = 0.892. \tag{5}$$

The information gained then becomes

$$\Delta H_{Y|\text{outlook}} = 0.94 - 0.693 = 0.247 \tag{6}$$
$$\Delta H_{Y|\text{Temp}} = 0.94 - 0.911 = 0.029 \tag{7}$$
$$\Delta H_{Y|\text{Humidity}} = 0.94 - 0.788 = 0.152 \tag{8}$$
$$\Delta H_{Y|\text{Windy}} = 0.94 - 0.892 = 0.048. \tag{9}$$



After splitting the branch.

Notice that after picking Outlook as the 1st feature to split the tree.

- Having the outlook=overcast, only has 1 possible golf outcome, **Yes**.
- As the tree splits, if there is only 1 possible outcome, you can safely end the branch of the tree with the label.
- As the tree splits, you have to split the branches further with other features if there are multiple possible outcomes.
- For each new branch. You will assume that the outlook is already known and shrink the table to calculate the next branch.
- For example, let's assume we are working on the Rainy branch. Then to identify the next feature to use, we will use a smaller table where the outlook is only Rainy.

Outlook	Temperature	Humidity	Windy	Play Golf
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	No
Rainy	Mild	High	FALSE	No

Rainy	Cool	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes

Attributes | Classes

Outlook	Temperature	Humidity	Windy	Play Golf
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	No
Overcast	Hot	High	FALSE	Yes
Sunny	Mild	High	FALSE	Yes
Sunny	Cool	Normal	FALSE	Yes
Sunny	Cool	Normal	TRUE	No
Overcast	Cool	Normal	TRUE	Yes
Rainy	Mild	High	FALSE	No
Rainy	Cool	Normal	FALSE	Yes
Sunny	Mild	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Sunny	Mild	High	TRUE	No

Practice, Practice

After identifying the **outlook** as the first question. We need to identify the next branch. Let's **Playing Golf** be Y and the other features be X. (use $\log_e$)

- Look at the Rain Status table.
- Identify the next feature that provides the largest information gain.

Outlook	Temperature	Humidity	Windy	Play Golf
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	No
Rainy	Mild	High	FALSE	No

Rainy	Cool	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes

Solution to the next Layer

Golf	Yes	No
	2	3

Temp/Golf	Yes	No	Prob
Hot	0	2	2/5
Mild	1	1	2/5
Cool	1	0	1/5

Humidity/Golf	Yes	No
High	0	3
Normal	2	0

windy/Golf	Yes	No
True	1	1
False	1	2

$$H(Y) = -2/5\log(2/5) - 3/5\log(3/5) = 0.67$$

$$\begin{aligned} H(Y|T) = &-p(t=\text{hot})\left[p(y=\text{yes}|t=\text{hot})\log_2(p(y=\text{yes}|t=\text{hot})) + p(y=\text{no}|t=\text{hot})\log_2(p(y=\text{no}|t=\text{hot}))\right] \\ &- p(x=\text{cold})\left[p(y=\text{yes}|t=\text{cold})\log_2(p(y=\text{yes}|t=\text{cold})) + p(y=\text{no}|t=\text{cold})\log_2(p(y=\text{no}|t=\text{cold}))\right] \\ &- p(x=\text{mild})\left[p(y=\text{yes}|t=\text{mild})\log_2(p(y=\text{yes}|t=\text{mild})) + p(y=\text{no}|t=\text{mild})\log_2(p(y=\text{no}|t=\text{mild}))\right] \end{aligned}$$

$$H(Y|T) = -\frac{2}{5}\left[0+0\right] - \frac{2}{5}\left[\frac{1}{2}\log(\frac{1}{2}) + \frac{1}{2}\log(\frac{1}{2})\right] - \frac{1}{5}\left[0+0\right] = 0.277$$

$$H(Y|H) = 3/5(0) + 2/5(0) = 0$$

$$H(Y|W) = -2/5\left[\frac{1}{2}\log(\frac{1}{2}) + \frac{1}{2}\log(\frac{1}{2})\right] - 3/5\left[\frac{1}{3}\log(\frac{1}{3}) + \frac{2}{3}\log(\frac{2}{3})\right] = 0.659$$

This implies that Humidity gives us the most information Gain

$$\begin{aligned} \Delta H_{Y|T} &= 0.67 - 0.277 = 0.393 \\ \Delta H_{Y|H} &= 0.67 - 0 = 0.67 \\ \Delta H_{Y|W} &= 0.67 - 0.659 = 0.011 \end{aligned}$$

Decision Tree is a very popular algorithm

For a long time, decision tree has been one of the most popular classification algorithms.

- It is easy to implement in Python.
- It generally does a good job on simpler datasets.
- It tells you which features are most important.
- In fact, it tells you the importance of the features in order of importance based on the information gained.
- It is commonly used with Ensemble methods like bagging to future improve its performance.

Next up

- what are Ensemble methods?
- what is bagging?
- how does bagging future a tree into a forest?

Hands-on Exercises

- https://www.geeksforgeeks.org/decision-tree-introduction-example/
- Apply the decision on the prostate cancer methylation data

Thanks for your attention!