Data preparation and preprocessing : Discretization of continuous variables

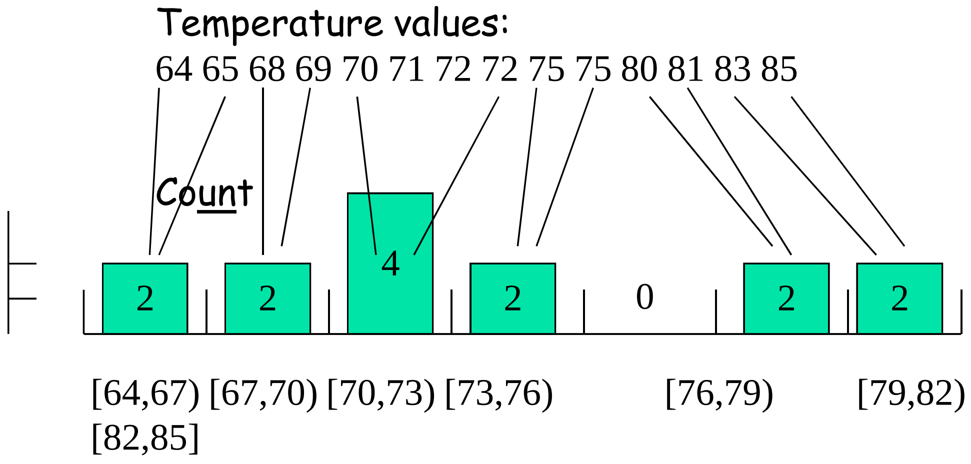
Discretization

- Divide the range of a continuous attribute into intervals
  - E.g. Annual Income, Nytrogen concentration, Speed etc

- Some methods require discrete values, e.g. most versions of Naïve Bayes, CHAID
- Reduce data size by discretization
- Prepare for further analysis
- Discretization is very useful for generating a summary of data
- Also called “binning” or “partitioning”

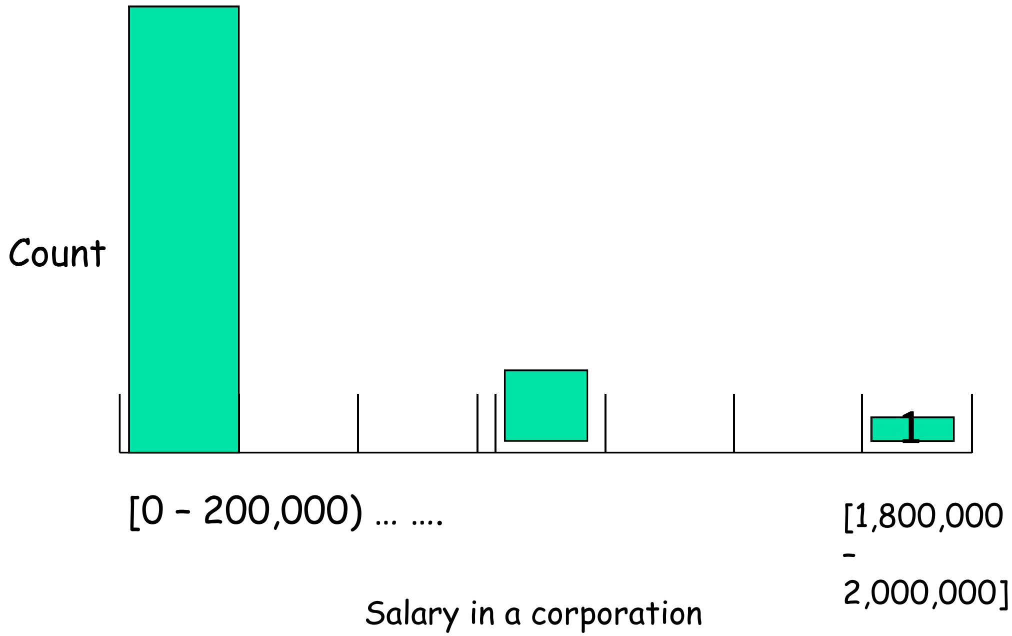
Equal-width Binning

- Also called Equal Distance Partitioning
- It divides the range into N intervals of equal size (range):
  - uniform grid
- If A and B are the lowest and highest values of the attribute
  - The width of intervals will be: W = (B -A)/N



Equal Width, bins Low <= value < High

Equal-width Binning



Disadvantage

a)Unsupervised

b)Where does N come from?

c)Sensitive to outliers

a) The most straightforward
b) But outliers may dominate presentation
c) Skewed data is not handled well.

Advantage

(a)simple and easy to implement

(b)produce a reasonable abstraction of data

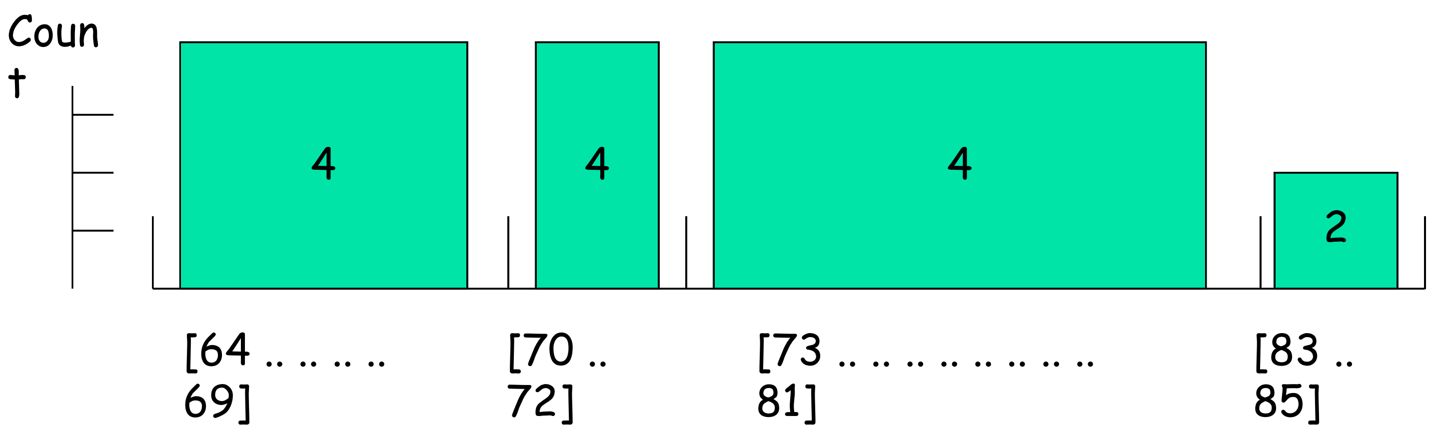
Equal-depth Binning

- Also known as equal height binning or Equal freuency binning
- It divides the range into N intervals
    - each containing approximately the same number of samples
- Generally preferred because avoids clumping
- In practice, “almost-equal” height binning is used to give more intuitive breakpoints

Equal-depth Binning

Temperature values:

64 65 68 69 70 71 72 75
75 80 81 83 85



Equal Height = 4, except for the last bin

Equal-depth Binning

- Additional considerations:
  - don’t split frequent values across bins
  - create separate bins for special values (e.g. 0)
  - readable breakpoints (e.g. round breakpoints)
- Good data scaling
- Managing categorical attributes can be tricky

Exercise

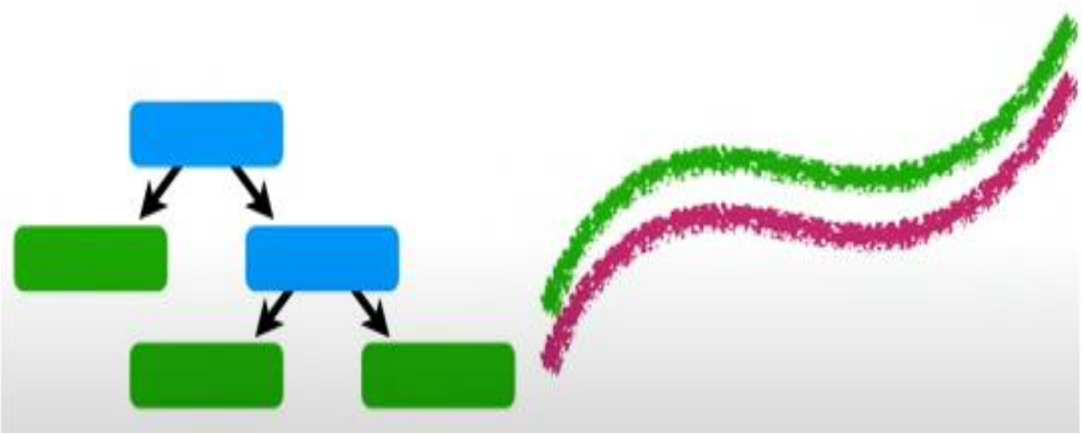
- Discretize the following values using EW and ED binning
- 13, 15, 16, 16, 19, 20, 21, 22, 22, 25, 30, 33, 35, 35, 36, 40, 45

Discretization considerations

- Class-independent methods
  - Equal Width is simpler, good for many classes
    - can fail miserably for unequal distributions
  - Equal Height gives better results
- Class-dependent methods can be better for classification
  - Decision tree methods build discretization on the fly
  - Naïve Bayes requires initial discretization
- Many other methods exist …

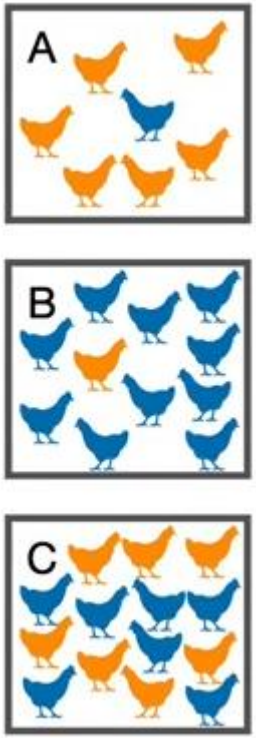
Entropy

- Entropy can be used to build classification trees which are used to classify things.
- Entropy is also the basis of something called mutual information which quantifies the relationship between two things.
- So we used entropy to something derived from it, to quantify similarities and differences.
- How entropy quantifies similarities and differences.



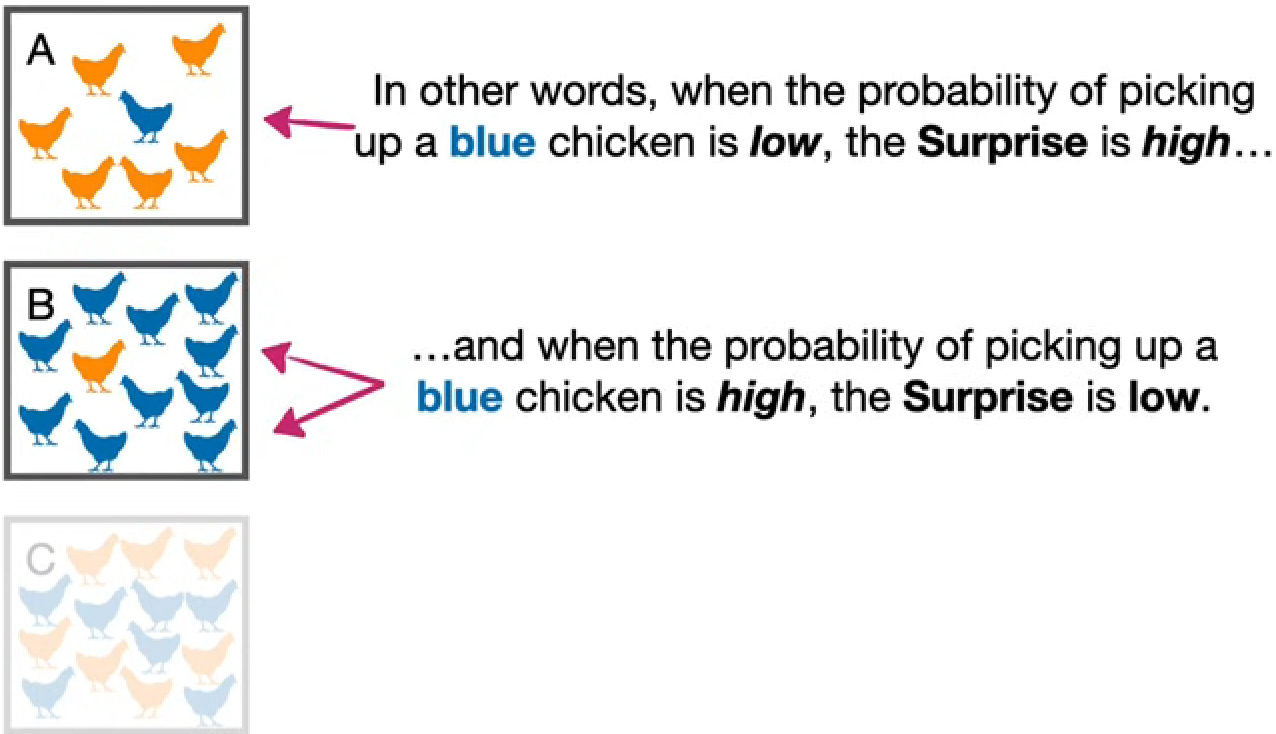
Entropy

- Before to know entropy, first we have to understand surprise.
- Imagine we had two types of chickens, orange and blue and instead of letting them randomly roam all over the screen, we organized into three separate areas: A, B and C



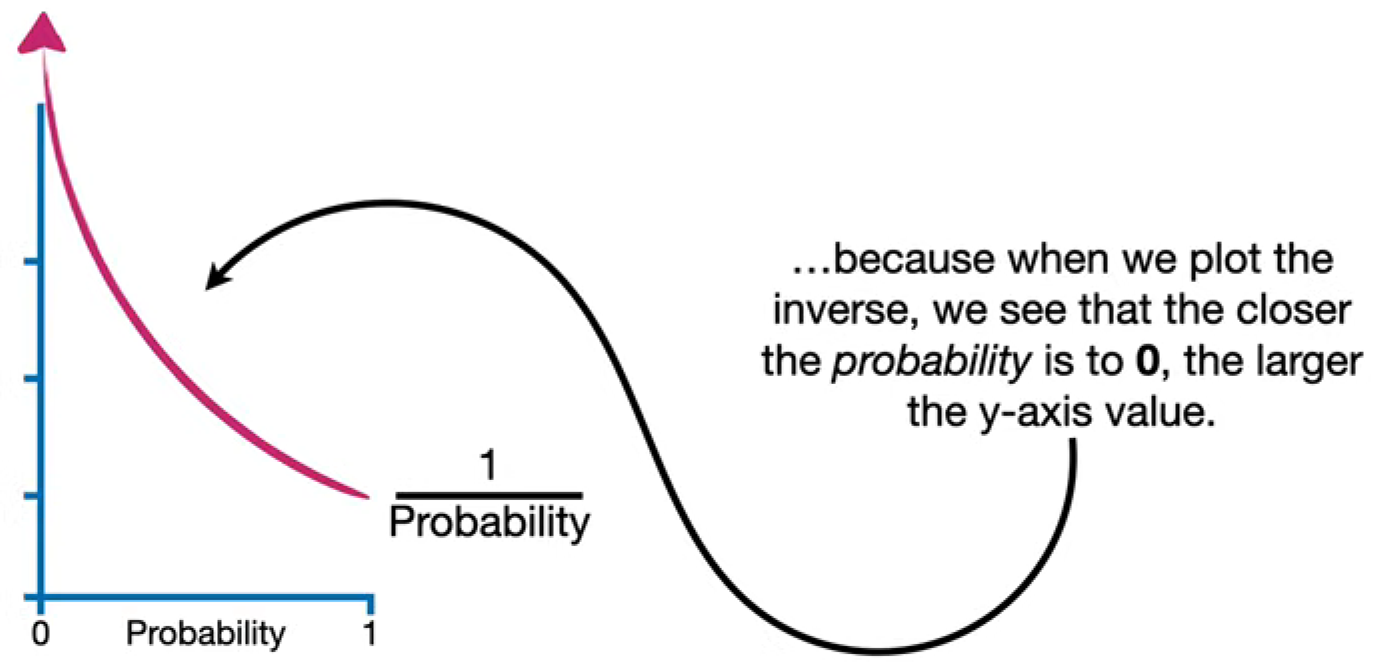
Entropy

- If we randomly picked chicken in area A then because these are 6 orange and only 1 blue chicken, there is a higher probability that they will pick up an orange chicken.
- But if we picked up the **blue** chicken from area A we would be relatively **surprised**.



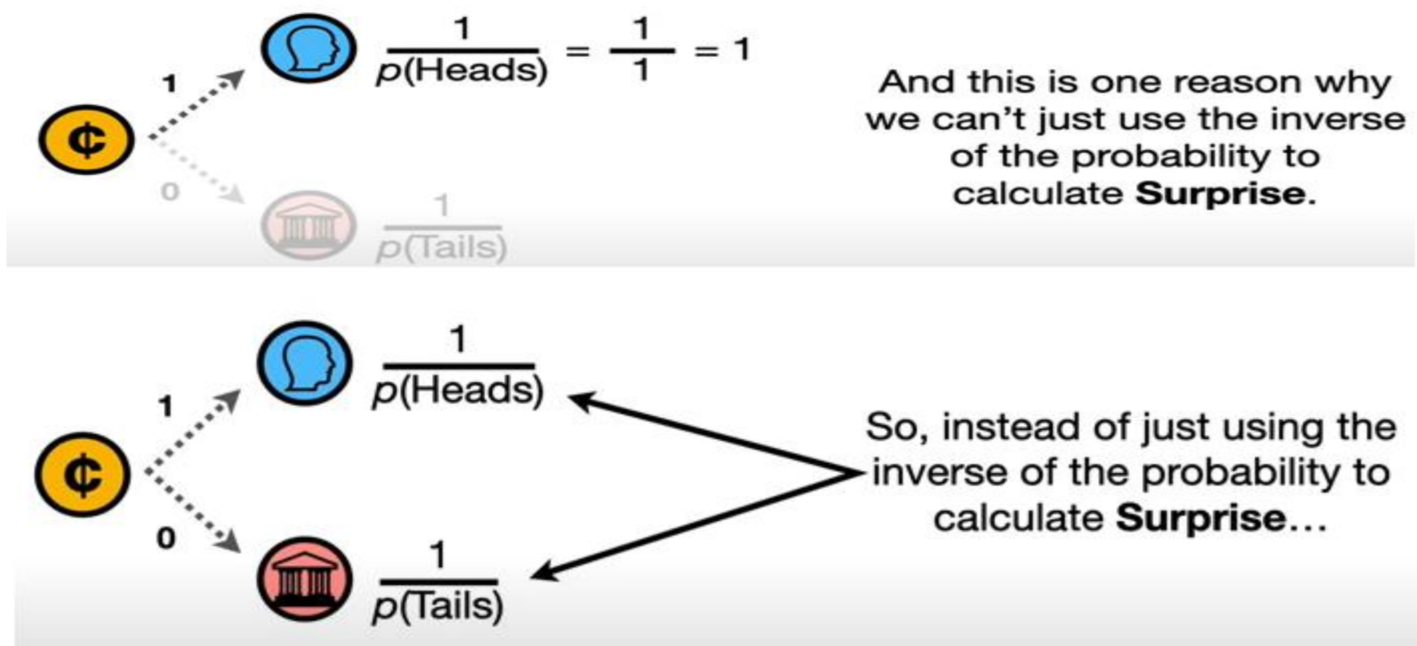
Entropy

- Now we have a general intuition of how probability is related to surprise.
- How to calculate surprise.
- We know there is a type of inverse relationship between probability and surprise.

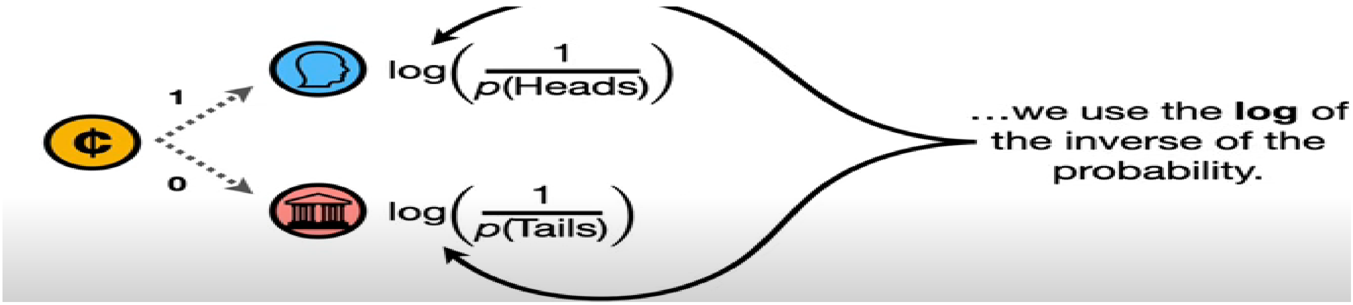


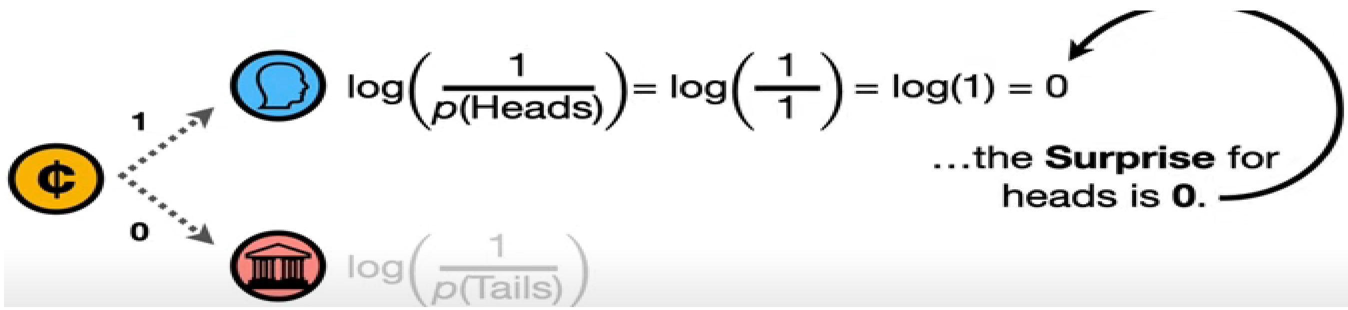
Entropy

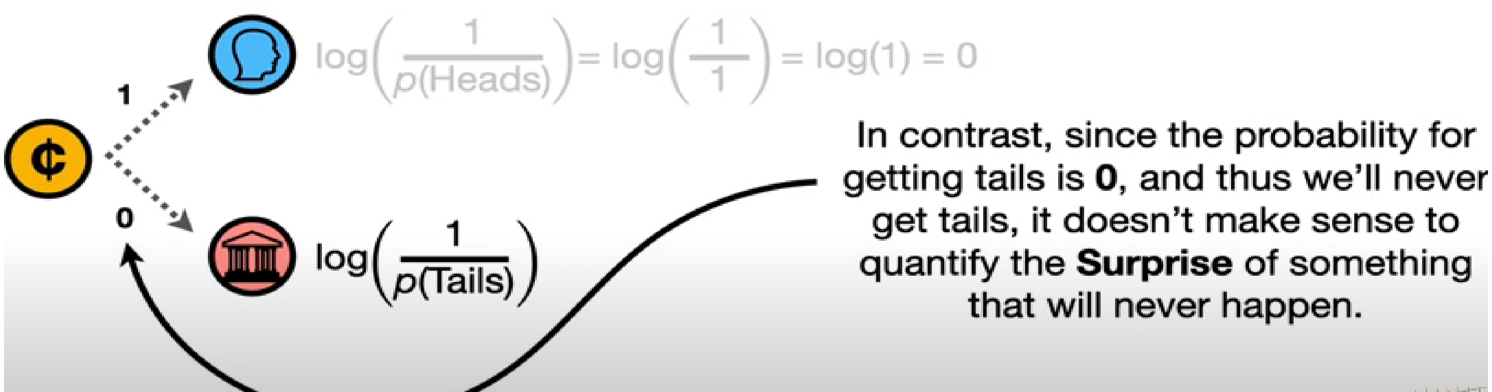
- One problem
- Surprise associated with flipping a coin.
- Imagine we had a terrible coin and every time we flipped it we got heads.
- Now how surprised we would be if the next flip gave us heads?
- So, when the probability of getting heads is 1 then we want the surprise foe getting heads to be 0.



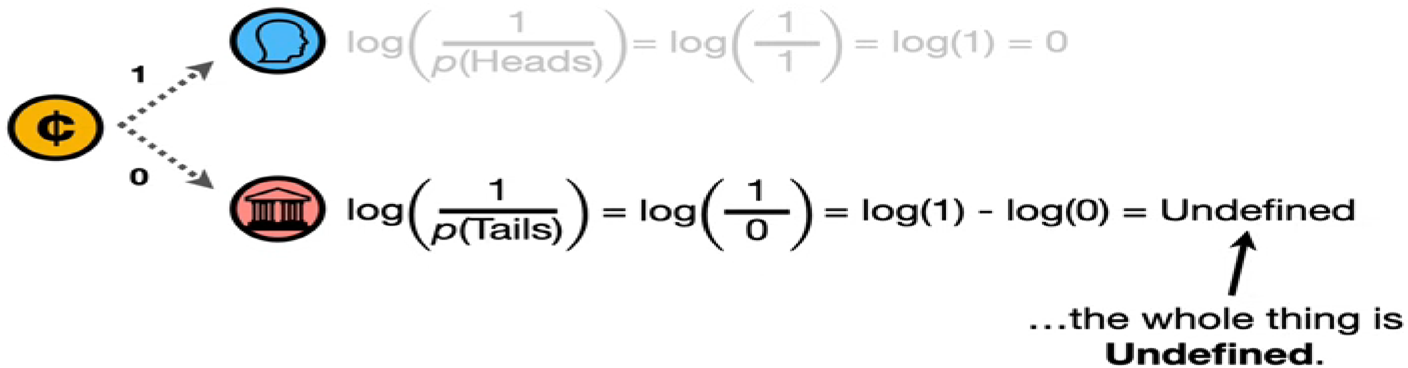
Entropy

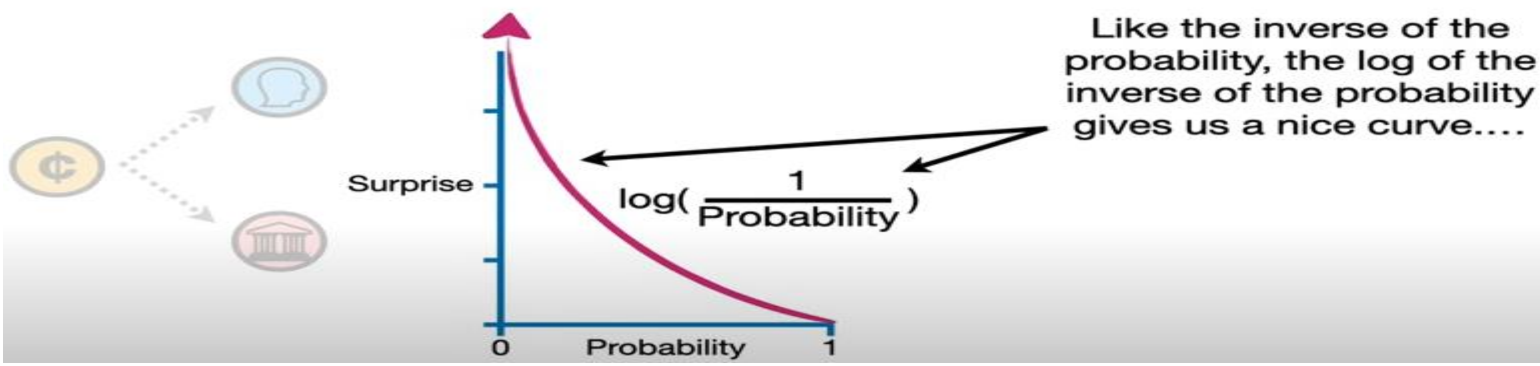

1
0
$\log\left(\frac{1}{p(\text{Heads})}\right)$
$\log\left(\frac{1}{p(\text{Tails})}\right)$
...we use the **log** of the inverse of the probability.


1
0
$\log\left(\frac{1}{p(\text{Heads})}\right) = \log\left(\frac{1}{1}\right) = \log(1) = 0$
$\log\left(\frac{1}{p(\text{Tails})}\right)$
...the **Surprise** for heads is **0**.


1
0
$\log\left(\frac{1}{p(\text{Heads})}\right) = \log\left(\frac{1}{1}\right) = \log(1) = 0$
$\log\left(\frac{1}{p(\text{Tails})}\right)$
In contrast, since the probability for getting tails is **0**, and thus we'll never get tails, it doesn't make sense to quantify the **Surprise** of something that will never happen.

Entropy



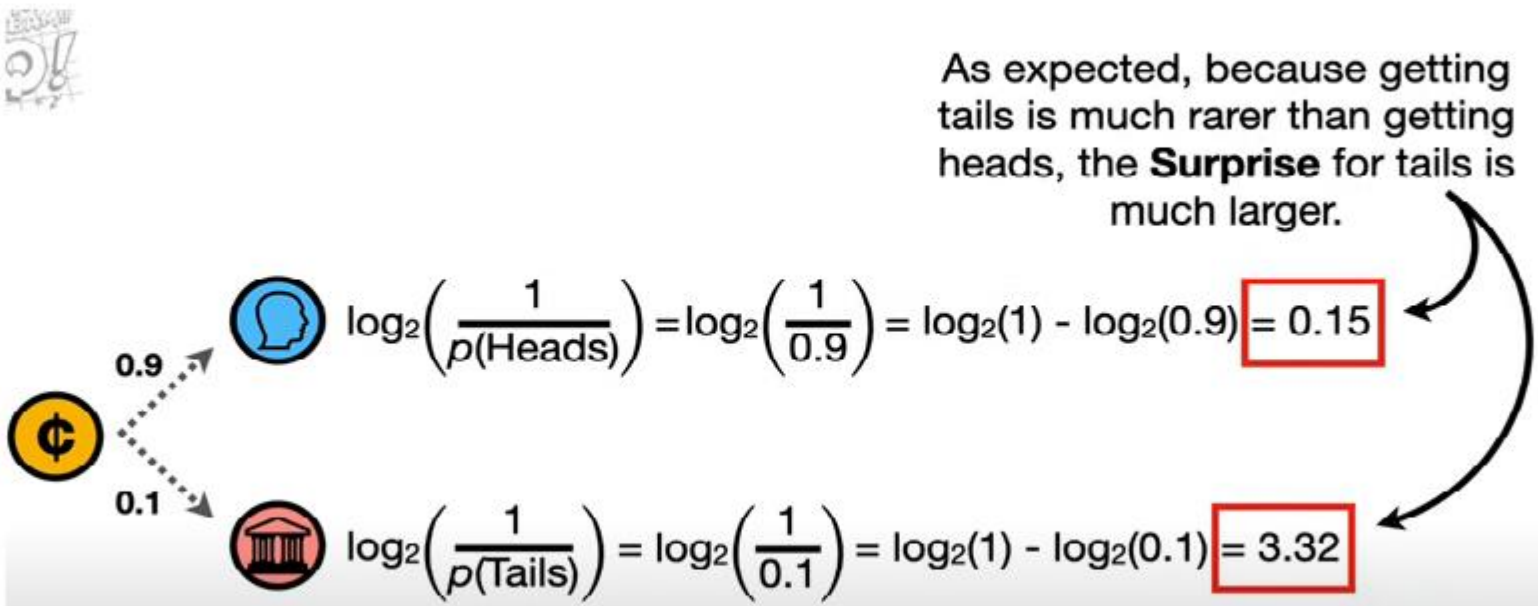


So **Surprise** is the **log** of the **inverse** of the **probability**.

$$\text{Surprise} = \log\left(\frac{1}{\text{Probability}}\right)$$

Entropy

- Calculate surprise
- Imagine that our coin that gets heads 90% of the time and it got tails 10% if the time.
- Now, calculate the surprise for getting heads and tails.



Entropy

- Now let’s flip the coin 3 times and we get heads, heads and tails..
- The probability of getting 2 heads and 1 tails is 0.9*0.9*0.1
- To know how surprising it is to get 2 heads and 1 tails then we can plug this probability into the equation of surprise.

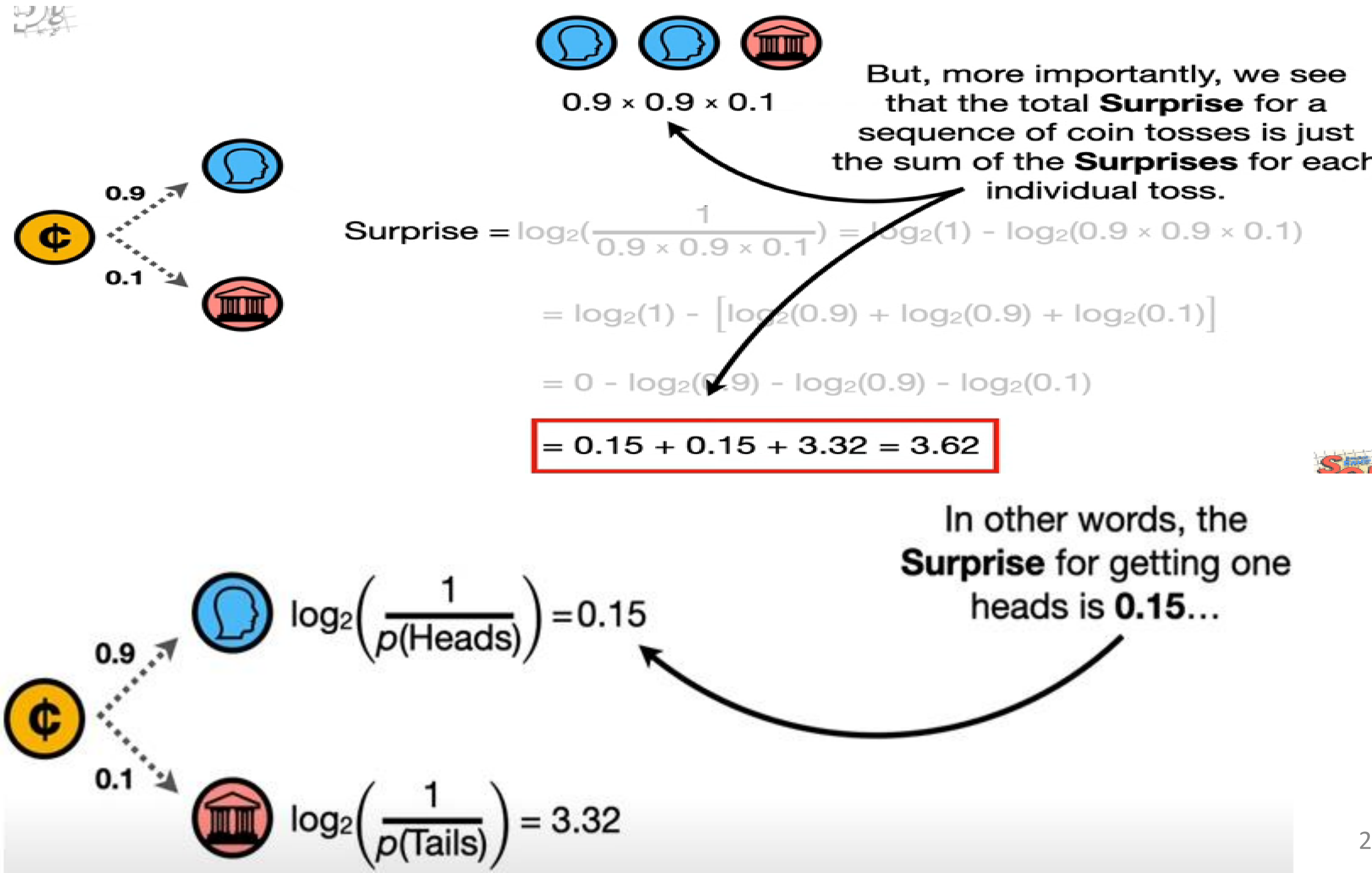
$$\text{Surprise} = \log_2\left(\frac{1}{0.9 \times 0.9 \times 0.1}\right) = \log_2(1) - \log_2(0.9 \times 0.9 \times 0.1)$$

$$= \log_2(1) - \left[\log_2(0.9) + \log_2(0.9) + \log_2(0.1)\right]$$

$$= 0 - \log_2(0.9) - \log_2(0.9) - \log_2(0.1)$$

$$= 0.15 + 0.15 + 3.32 = 3.62$$

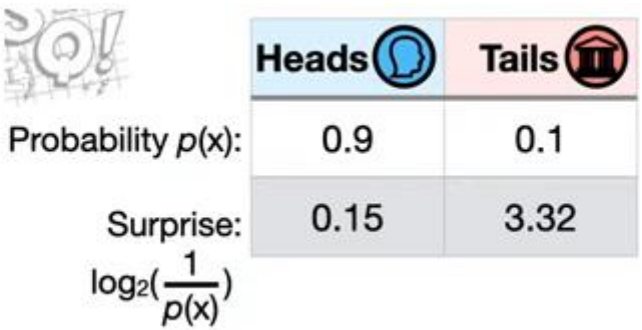
Entropy

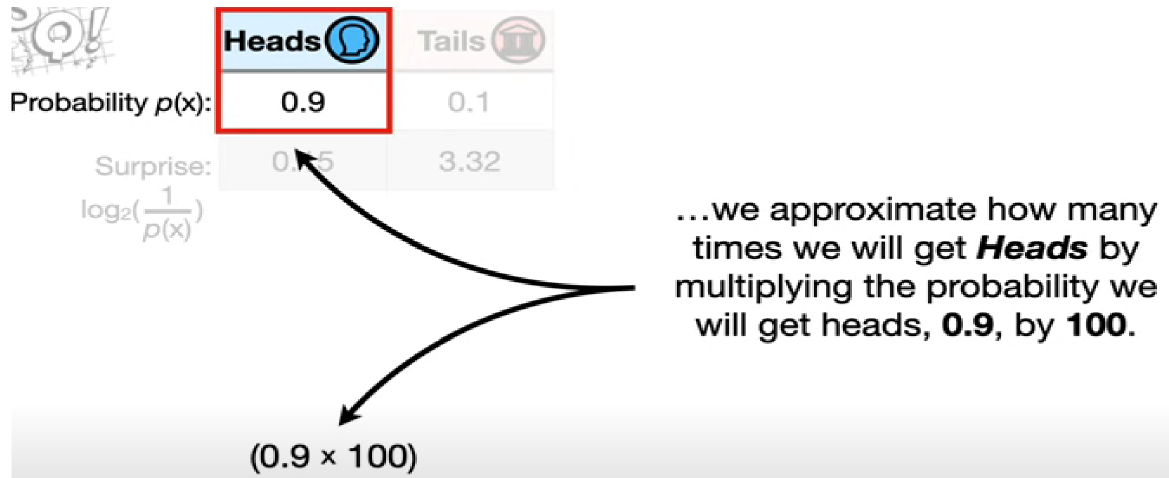


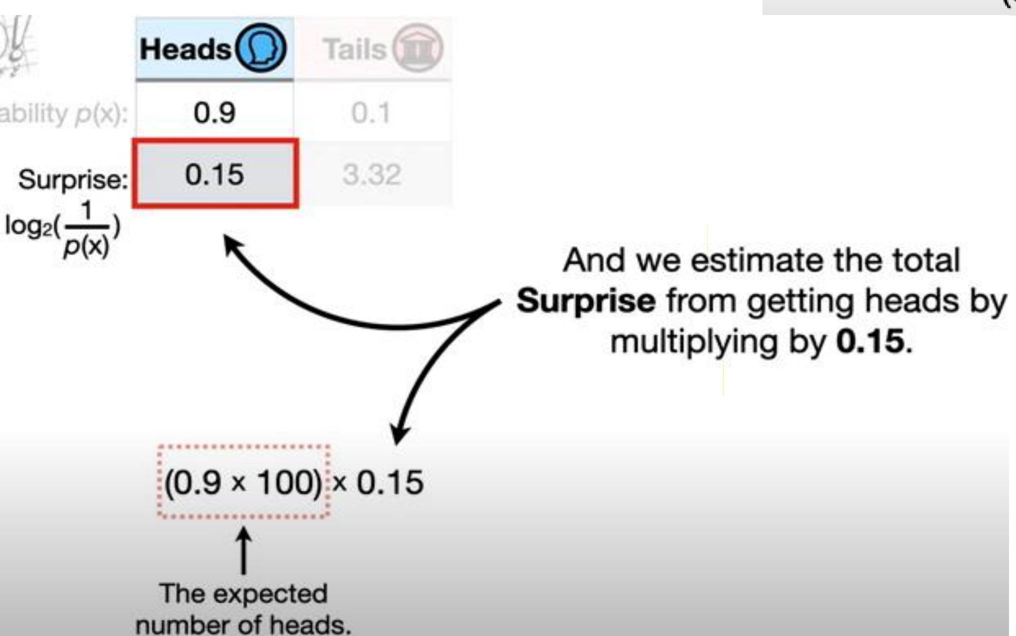
$0.9 \times 0.9 \times 0.1$
But, more importantly, we see that the total **Surprise** for a sequence of coin tosses is just the sum of the **Surprises** for each individual toss.
0.9
0.1
$\text{Surprise} = \log_2\left(\frac{1}{0.9 \times 0.9 \times 0.1}\right) = \log_2(1) - \log_2(0.9 \times 0.9 \times 0.1)$
$= \log_2(1) - [\log_2(0.9) + \log_2(0.9) + \log_2(0.1)]$
$= 0 - \log_2(0.9) - \log_2(0.9) - \log_2(0.1)$
$= 0.15 + 0.15 + 3.32 = 3.62$
In other words, the **Surprise** for getting one heads is **0.15**…
0.9
0.1
$\log_2\left(\frac{1}{p(\text{Heads})}\right) = 0.15$
$\log_2\left(\frac{1}{p(\text{Tails})}\right) = 3.32$

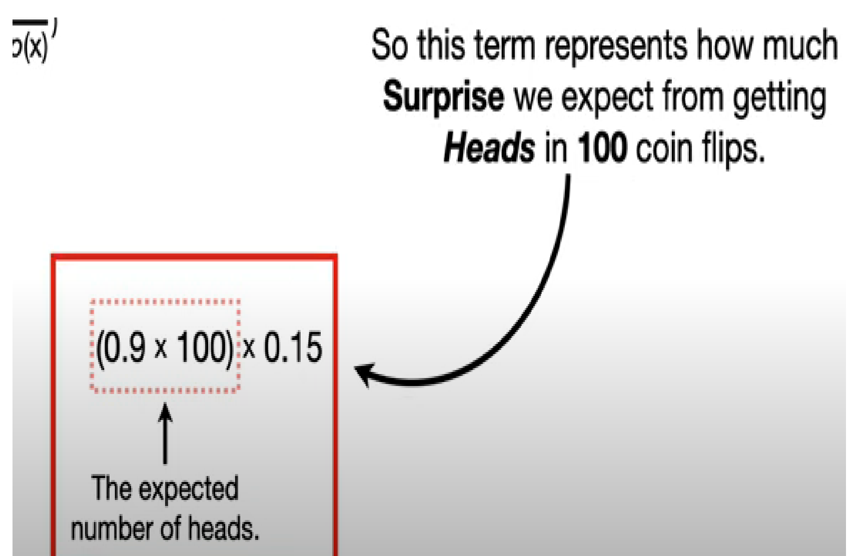
Entropy

- Estimate total surprise after flipping the coin 100 times









Entropy

- Estimate total surprise after flipping the coin 100 times
- Aren't we supposed to be talking about entropy?

Now we can add the two terms together to find out the total **Surprise**…

$$\boxed{(0.9 \times 100)} \times 0.15 + \boxed{(0.1 \times 100)} \times 3.32$$

The expected number of heads.

The expected number of tails.

…then we get the average amount of **Surprise** ***per*** coin toss, **0.47**.

$$\frac{(0.9 \times 100) \times 0.15 + (0.1 \times 100) \times 3.32}{100} = \frac{46.7}{100} \boxed{= 0.47}$$

So, on *average*, we *expect* the **Surprise** to be **0.47** every time we flip the coin…

…and that is the **Entropy** of the coin: the expected **Surprise** every time we flip the coin.

Entropy

$\log_2(\frac{1}{p(x)})$

In fancy statistics notation, we say that **Entropy** is the **Expected Value** of the **Surprise**.

$$E(\textbf{\textit{Surprise}}) = \frac{(0.9 \times 100) \times 0.15 + (0.1 \times 100) \times 3.32}{100} = \frac{46.7}{100} \boxed{= 0.47}$$

	Heads	Tails
Probability $p(x)$:	0.9	0.1
Surprise: $\log_2(\frac{1}{p(x)})$	0.15	3.32

...plus the probability that a **Surprise** for ***Tails*** will occur times its **Surprise**.

$$E(\textbf{\textit{Surprise}}) = (0.9 \times 0.15) + (0.1 \times 3.32)$$

$\log_2(\frac{1}{p(x)})$

NOTE: We can rewrite **Entropy** just like an **Expected Value**, using fancy ***Sigma*** notation.

$$E(\textbf{\textit{Surprise}}) = (0.9 \times 0.15) + (0.1 \times 3.32) = 0.47 = \boxed{\sum x\, P(X = x)}$$

Entropy

$$E(\textit{Surprise}) = (0.9 \times 0.15) + (0.1 \times 3.32) = 0.47 = \sum x \, P(X = x)$$

Specific value for **Surprise**.

The probability of observing that specific value for **Surprise**.

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

Surprise

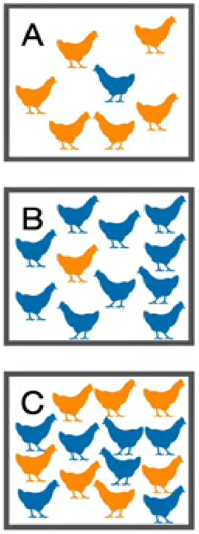
The probability of the **Surprise**.

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

$$\boxed{\text{Entropy} = -\sum p(x)\log(p(x))}$$

$$p(x)\left[\log(1) - \log(p(x))\right] \rightarrow \sum p(x)\left[0 - \log(p(x))\right] \rightarrow \sum -p(x)\log(p(x))$$

Entropy



Now, going back to the original example, we can now calculate the **Entropy** of the chickens.

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

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

$$= 6/7 \times \log_2\left(\frac{1}{6/7}\right) + 1/7 \times \log_2\left(\frac{1}{1/7}\right)$$

$$= (0.86 \times 0.22) + (0.14 \times 2.81)$$

$$= 0.59$$

…thus, the total **Entropy**, **0.59**, is much closer to the **Surprise** associated with **orange** chickens (0.22) than **blue** chickens (2.81).

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

$$= 6/7 \times \log_2\left(\frac{1}{6/7}\right) + 1/7 \times \log_2\left(\frac{1}{1/7}\right)$$

$$= (0.86 \times 0.22) + (0.14 \times 2.81)$$

$$\boxed{= 0.59}$$

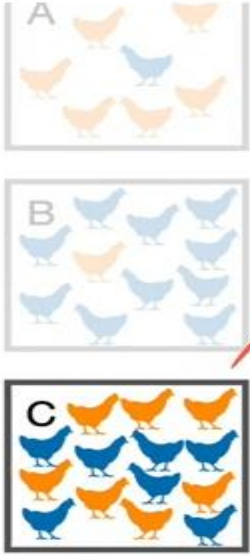
$$\text{Entropy} = \sum p(x)\log\left(\frac{1}{p(x)}\right)$$

$$= 1/11 \times \log_2\left(\frac{1}{1/11}\right) + 10/11 \times \log_2\left(\frac{1}{10/11}\right)$$

$$= (0.09 \times 3.46) + (0.91 \times 0.14)$$

$$= 0.44$$

Entropy



Entropy = 0.59

Entropy = 0.44

Entropy = 1

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

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

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

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

$$= 1$$

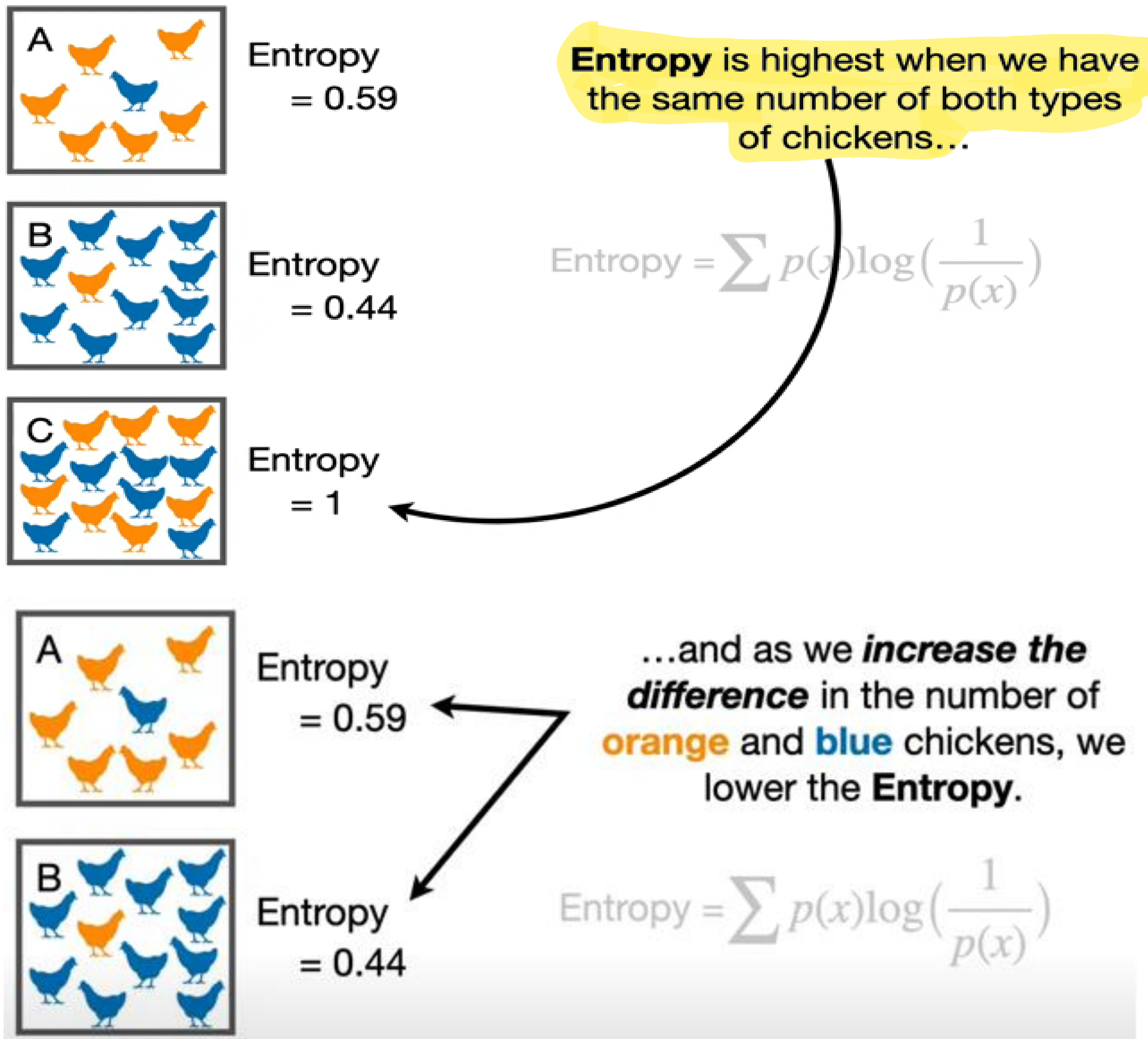
In this case, even though the **Surprise** for orange and blue chickens is relatively moderate, **1**…

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

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

$$= (0.5 \boxed{\times 1}) + (0.5 \boxed{\times 1})$$

Entropy



A
Entropy = 0.59
B
Entropy = 0.44
C
Entropy = 1
Entropy is highest when we have the same number of both types of chickens…
$\text{Entropy} = \sum p(x)\log\left(\frac{1}{p(x)}\right)$
A
Entropy = 0.59
B
Entropy = 0.44
…and as we increase the difference in the number of orange and blue chickens, we lower the Entropy.
$\text{Entropy} = \sum p(x)\log\left(\frac{1}{p(x)}\right)$

Entropy Based Discretization

Supervised technique : Class dependent (classification)

1.Sort examples in increasing order

2.Each value forms an interval ('m' intervals)

3.Calculate the entropy measure of this discretization

10000 instances

1: 200, 2:300   p(1) = 200/10000   p(2) = 300/10000   200/10000(p(1)logp(1) + 300/10000 (p(2),log(p20) + ……..

$$E(S,T) = \frac{|S_1|}{|S|} Ent(S_1) + \frac{|S_2|}{|S|} Ent(S_2)$$

1.Find the binary split boundary that minimizes the entropy function over all possible boundaries. The split is selected as a binary discretization.

5.Apply the process recursively until some stopping criterion is met, e.g.,
$Ent(S) - E(T,S) > \delta$

Entropy/Impurity

- S - training set, $C_1,\ldots,C_N$ classes
- Entropy E(S) - measure of the impurity in a group of examples
  - p_c - proportion of C_c in S

$$\text{Impurity}(S) = -\sum_{c=1}^{N} p_c \cdot \log_2 p_c$$

Binning Methods for Data Smoothing

- Sorted data (attribute values)
- for price (attribute: price in dollars): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

1. Partition into (equal-depth) bins:

- Bin 1: 4, 8, 9, 15
- Bin 2: 21, 21, 24, 25
- Bin 3: 26, 28, 29, 34

2.A . Smoothing by bin means:

- Bin 1: 9, 9, 9, 9
- Bin 2: 23, 23, 23, 23
- Bin 3: 29, 29, 29, 29

2. b. Smoothing by bin boundaries:

- Bin 1: 4, 4, 4, 15
- Bin 2: 21, 21, 25, 25
- Bin 3: 26, 26, 26, 34

- Replace all values in a BIN by ONE value (smoothing values eg. mean)
- Replace some values in a Bin by specific value (Nearest boundry)