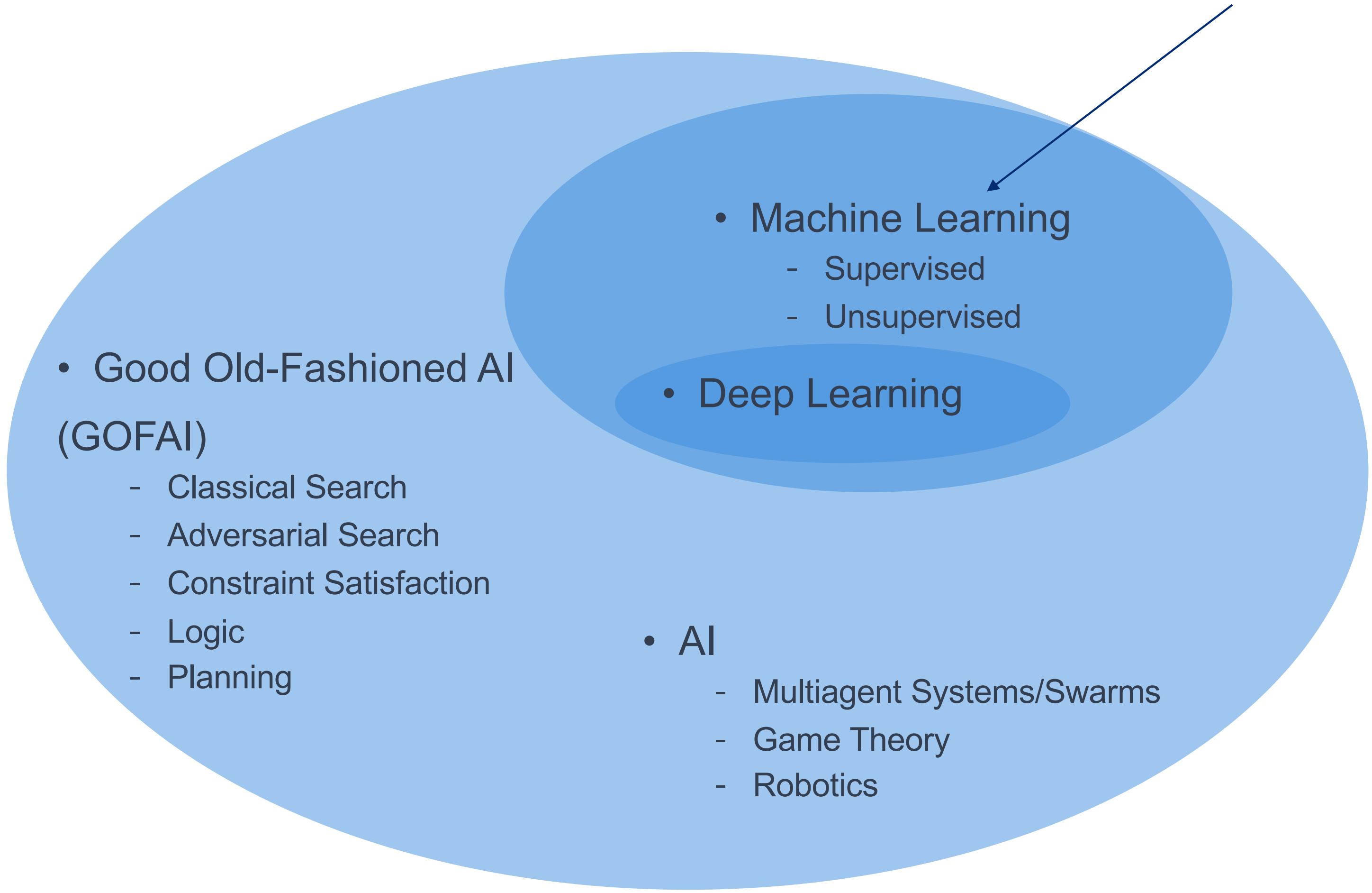
Decision Trees

Musad Haque
06-Nov-2023

Where are we in the semester?

You are here.

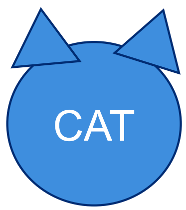


Agent days are over;
it's all about data now.

Machine Learning Problems

- Classification
  - Predict category
- Regression
  - Predict numerical value
- Clustering
  - Group similar items

- Anomaly Detection
  - Find what's "uncommon"
- Recommendation
  - Suggest based on interest



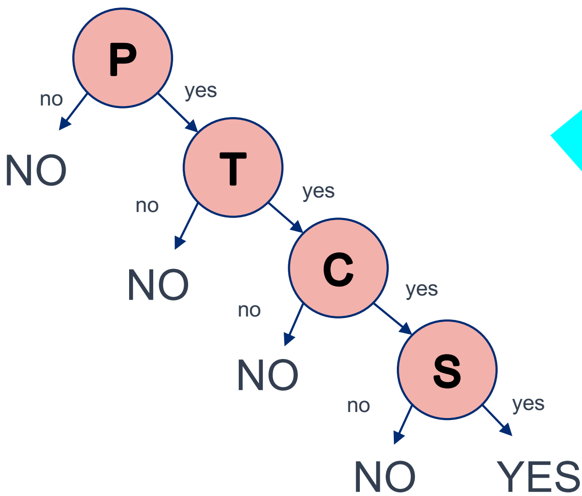
Rules

- Re-think the classification problem in terms of rules
- Rules, as in answers to a game like 20 questions (or as many questions as there are features that we care about)

https://de.wikipedia.org/wiki/20_Questions

Rules

- Answers to rules can help partition the space of possibilities
- Example: Classify `picture_on_screen` {YES or NO}
  - Is it connected to cable (C)? {no, yes}
  - Is the screen covered (S)? {no, yes}
  - Is it turned on (T)? {no, yes}
  - Is it plugged in (P)? {no, yes}

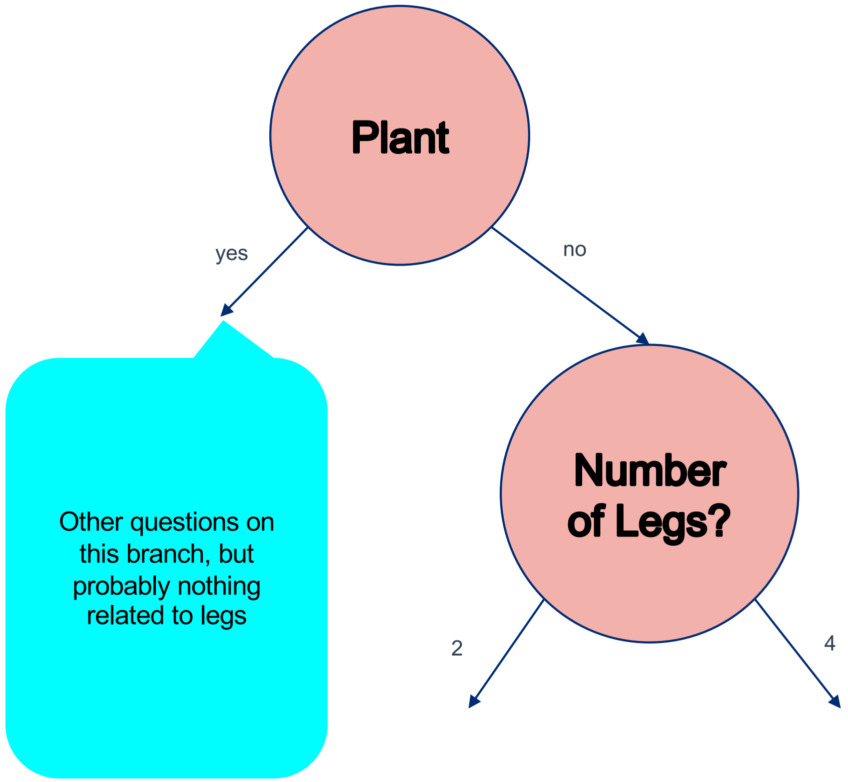


These questions lead to a tree. Design considerations: whether or not a question should be asked. For example, color of TV is not included here. If questions are asked, in what order?

Can we learn a tree?

Decision Trees

- Learn a tree from **training data**
- Each feature is a decision point
- Examine the features for the best order in which to ask our questions
  - Some features may not be needed at all
  - Some features may not be needed based on where we are on the tree



Training Data

	Tail?	Intelligent?	Lazy?	y
1	YES	YES	NO	Cat
2	YES	NO	YES	Cat
3	NO	YES	NO	Human
…	…	…	…	…
5000	YES	YES	YES	Cat

Three features, **x**, “tail?”, “intelligent?”, “lazy?”

Target Variable, **y**, is either cat or human

Supervised vs Unsupervised Learning

	Tail?	Intelligent?	Lazy?	y
1	YES	YES	NO	Cat
2	YES	NO	YES	Cat
3	NO	YES	NO	Human
…	…	…	…	…
5000	YES	YES	YES	Cat

Example of **Supervised Learning** since **y** values/labels are present in the training data

	Tail?	Intelligent?	Lazy?
1	YES	YES	NO
2	YES	NO	YES
3	NO	YES	NO
…	…	…	…
5000	YES	YES	YES

Example of training data where we would have to use techniques under the banner of **Unsupervised Learning** since **y** values/labels are missing from the training data

Supervised: Classification

	Tail?	Intelligent?	Lazy?	y
1	YES	YES	NO	Cat
2	YES	NO	YES	Cat
3	NO	YES	NO	Human
…	…	…	…	…
5000	YES	YES	YES	Cat

Three features, **x**, "tail?", "intelligent?", "lazy?"

Target Variable, **y**, is either cat or human

Given this test data, how do we classify it? Human or Cat?

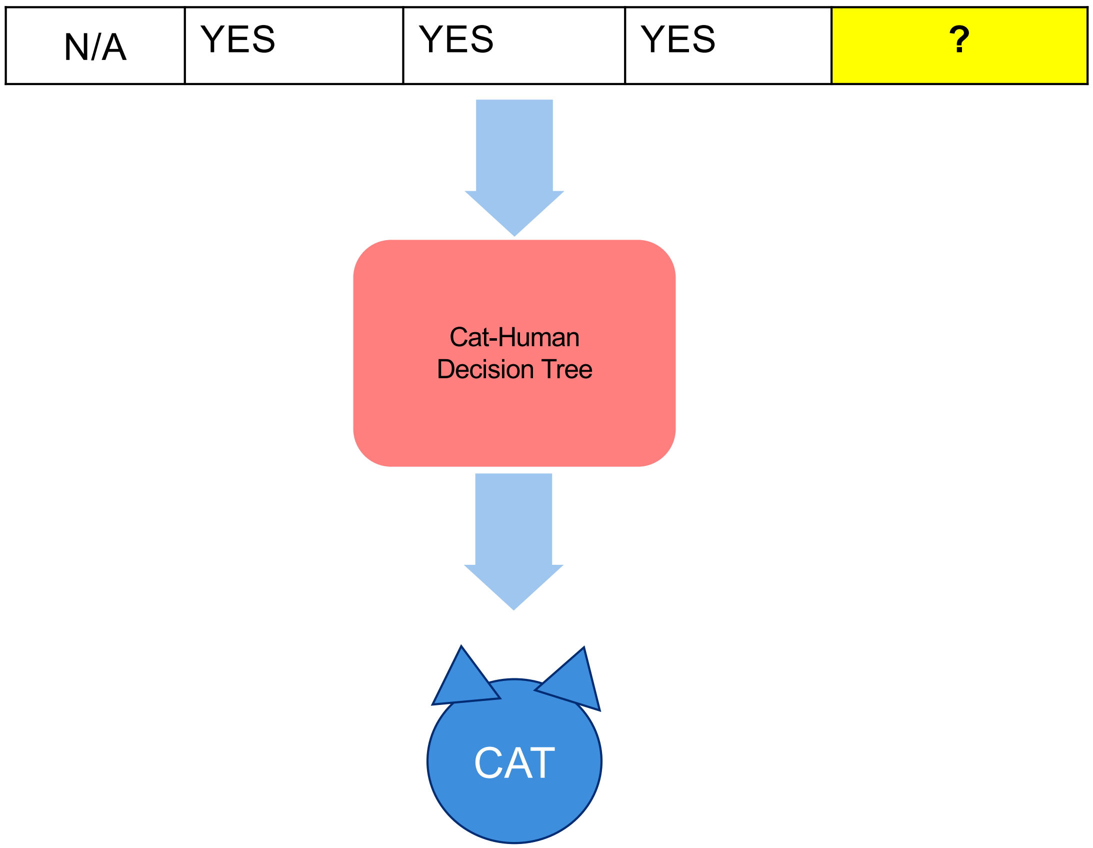
N/A	YES	YES	YES	**?**

Learn the Decision Tree

	Tail?	Intelligent?	Lazy?	y
1	YES	YES	NO	Cat
2	YES	NO	YES	Cat
3	NO	YES	NO	Human
…	…	…	…	…
5000	YES	YES	YES	Cat

Cat-Human
Decision Tree

Apply to Make Classifications



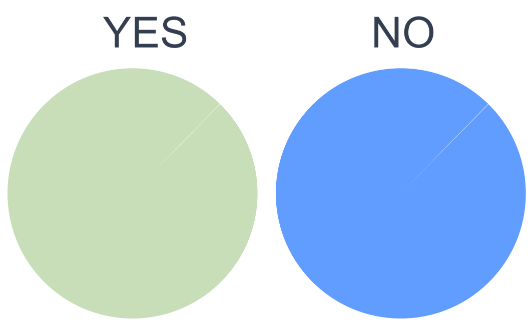
Features and Decision Trees

	Tail?	Intelligent?	Lazy?	y
1	YES	YES	NO	Cat
2	YES	NO	YES	Cat
3	NO	YES	NO	Human
…	…	…	…	…
5000	YES	YES	YES	Cat

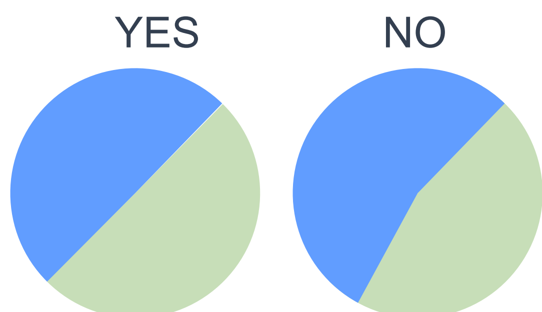
The **ideal** feature. We'll rarely come across this in practice. Knowing this feature is enough to classify the target variable.

"Learning the Decision Tree" is stacking features in a way to classify the target variable.

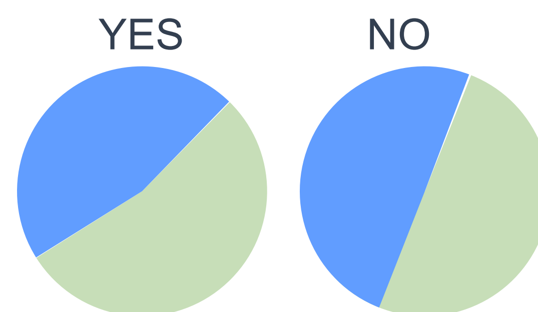
Tail?



Intelligent?



Lazy?



Generally…

- Imagine three features (or attributes), each with two possible values
  - Feature #m: yes, no
  - Feature #n: yes, no
  - Feature #k: yes, no
- The target variable can be either 1 or 0

- Feature #m (Ideal):
  - If yes, target is 1 for all training points
  - If no, target is 0 for all training points

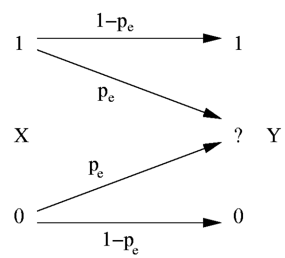
- Feature #n (pretty good):
  - If yes, target is mostly 1, but sometimes 0 across training points
  - If no, target is mostly 0, but sometimes 1 across training points

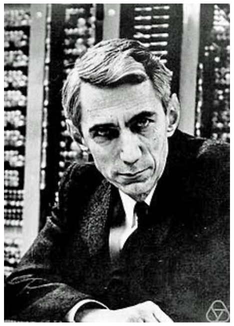
- Feature #k (yikes):
  - If yes, target is split between 0 and 1 across training points
  - If no, target is split between 0 and 1 across training points

How do we get a computer program to capture preference, i.e., **homogeneity** is preferred to heterogeneity

Information Theory

- Quantification, storage, and communication of information





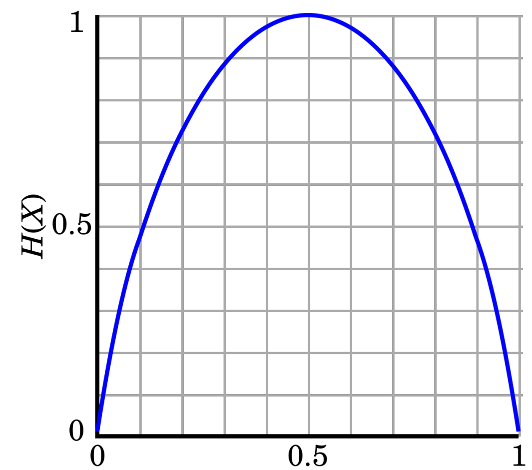
https://en.wikipedia.org/wiki/Claude_Shannon

Entropy and Mutual Information Theory, Joint **Entropy**, Conditional **Entropy**. Data Processing Theorem, Fano's Inequality, Asymptotic Equipartition Principle, Typical Sequences, Entropy, Source Coding and the AEP, Joint Typicality (Neuhoff/Forney notes), **Entropy** Rate, Conditional Independence and Markov Chains, **Entropy** Rate, Lossless Source Coding, Kraft Inequality, Shannon and Huffman Codes, Shannon, Fano, Elias Codes, Arithmetic Codes, Lempel Ziv Codes, Channel Capacity, Symmetric Channels, Discrete Memoryless Channels and Their Capacity, Arimoto-Blahut Algorithm, Proof of the Channel Coding Theorem, Converse of Channel Coding Theorem, Differential Entropy, Entropy, Mutual Information, AEP for Continuous rv's, Gaussian Channel, Capacity of AWGN, Bandlimited AWGN Channels, Capacity of Nonwhite Channels: Water Filling, Rate Distortion Theory, Quantization, Rate Distortion Functions, Vector Quantization, Vector Quantization Gains, Vector Quantization Design

Information Theory: Entropy, H(X)

- *Entropy measures uncertainty, surprise*

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$$



Information Theory: Entropy, H(X)

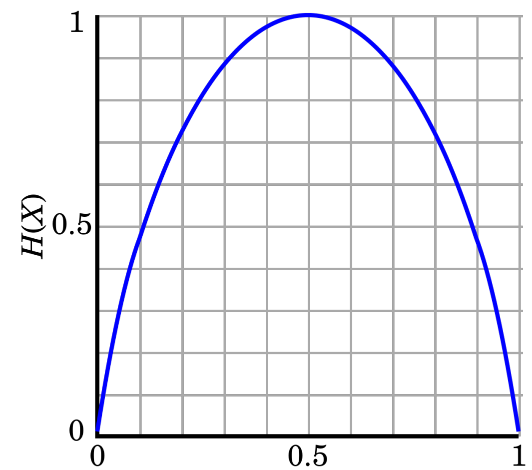
- *Entropy measures uncertainty, surprise*

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$$

Measured in bits

General case, n classes

$\log_2(0)=0$ by Information Theory convention



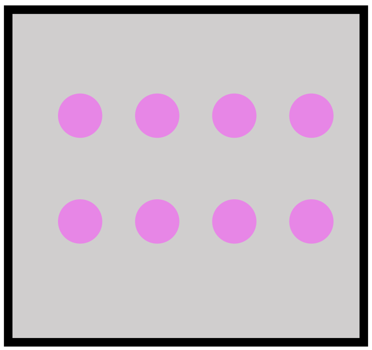
Plot of entropy when we have two *symbols* (*classes* for our purposes).

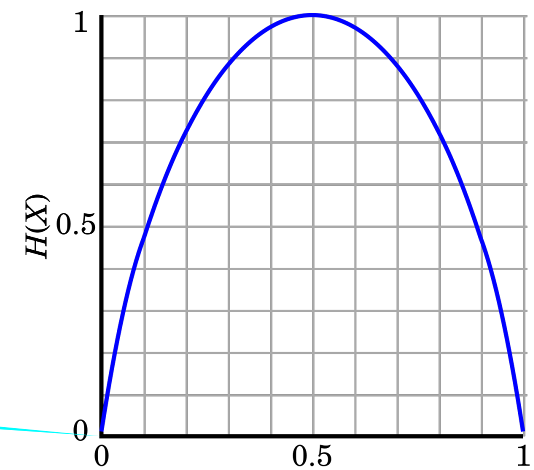
Information Theory: Entropy, H(X)

- *Entropy measures uncertainty, surprise*

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$$

Total: 8
Class 1: 0
Class 2: 8

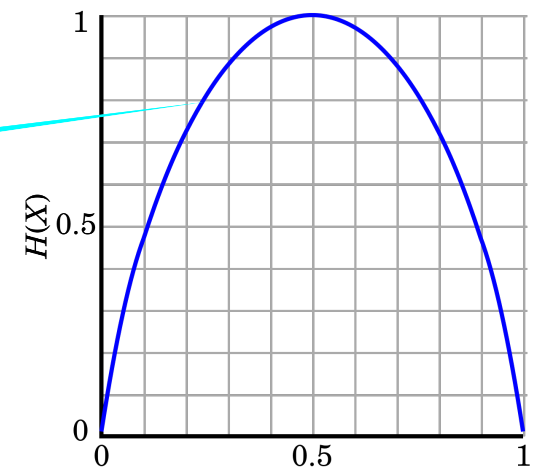




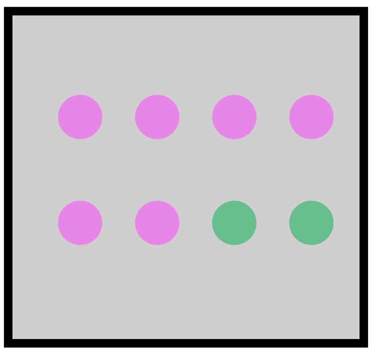
Information Theory: Entropy, H(X)

- *Entropy measures uncertainty, surprise*

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$$



Total: 8
Class 1: 2
Class 2: 6



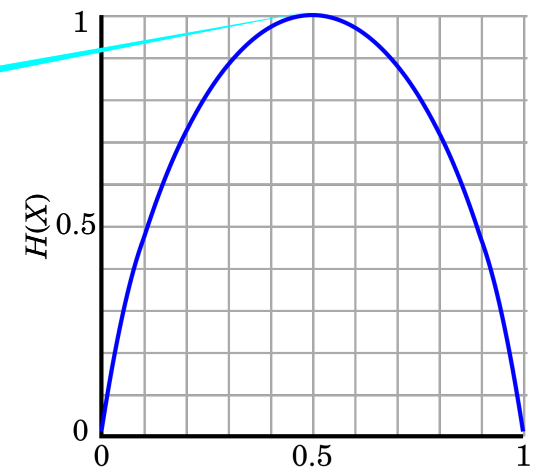
((-(2 / 8)) * log2(2 / 8)) - ((6 / 8) * log2(6 / 8)) =

0.81127812445

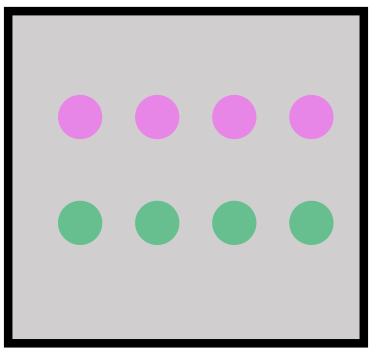
Information Theory: Entropy, H(X)

- *Entropy measures uncertainty, surprise*

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$$



Total: 8
Class 1: 4
Class 2: 4



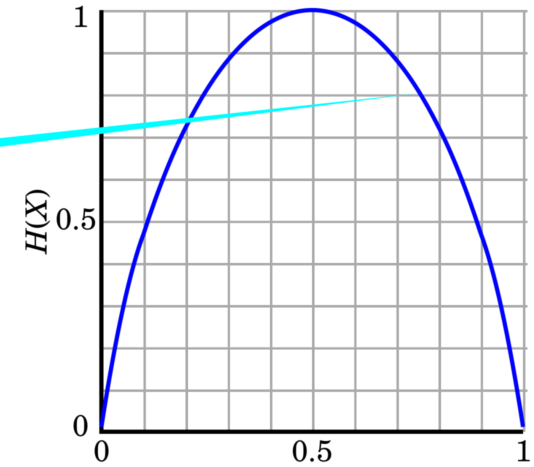
((-(4 / 8)) * log2(4 / 8)) - ((4 / 8) * log2(4 / 8)) =

1

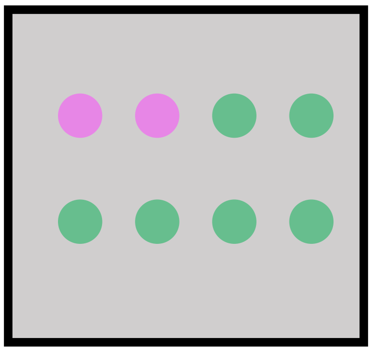
Information Theory: Entropy, H(X)

- *Entropy measures uncertainty, surprise*

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$$



Total: 8
Class 1: 6
Class 2: 2



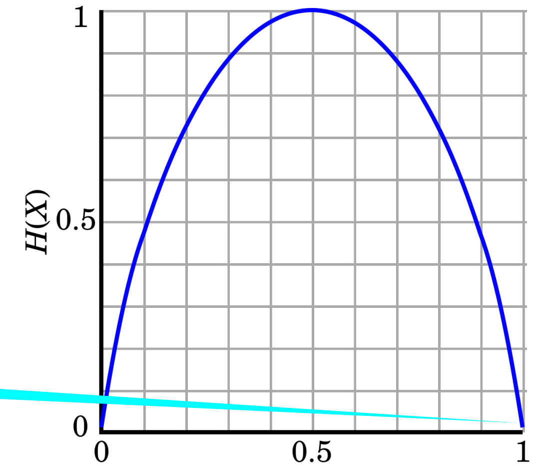
((-(6 / 8)) * log2(6 / 8)) - ((2 / 8) * log2(2 / 8)) =

0.81127812445

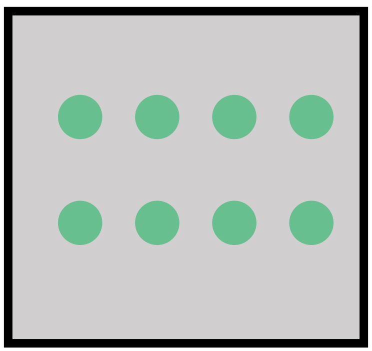
Information Theory: Entropy, H(X)

- *Entropy measures uncertainty, surprise*

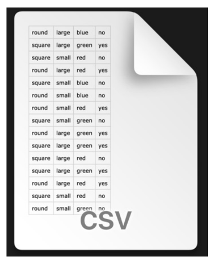
$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$$



Total: 8
Class 1: 8
Class 2: 0



Walkthrough an Example



round,large,blue,no
square,large,green,yes
square,small,red,no
round,large,red,yes
square,small,blue,no
round,small,blue,no
round,small,red,yes
square,small,green,no
round,large,green,yes
square,large,green,yes
square,large,red,no
square,large,green,yes
round,large,red,yes
square,small,red,no
round,small,green,no

Baseline Entropy

no	yes
no	yes
no	yes
no	yes
no	yes
no	yes
no	yes
no	

The target variable itself, across all features

$$E = -\frac{8}{15}\log_2(\frac{8}{15}) - \frac{7}{15}\log_2(\frac{7}{15})$$

0.9968

0	round	large	blue	no
1	square	large	green	yes
2	square	small	red	no
3	round	large	red	yes
4	square	small	blue	no
5	round	small	blue	no
6	round	small	red	yes
7	square	small	green	no
8	round	large	green	yes
9	square	large	green	yes
10	square	large	red	no
11	square	large	green	yes
12	round	large	red	yes
13	square	small	red	no
14	round	small	green	no

Consider First Feature

0.02812

round

NO YES
NO YES
NO YES
YES

square

NO YES
NO YES
NO YES
NO
NO

$$E = -\frac{3}{7}\log_2(\frac{3}{7}) - \frac{4}{7}\log_2(\frac{4}{7})$$

0.985

$$E = -\frac{5}{8}\log_2(\frac{5}{8}) - \frac{3}{8}\log_2(\frac{3}{8})$$

0.9544

weighted average

(7/15)*(0.985) + (8/15)*0.9544

0.96868

Information gain = 0.9968 – 0.96868 = **0.02812**

0	round	large	blue	no
1	square	large	green	yes
2	square	small	red	no
3	round	large	red	yes
4	square	small	blue	no
5	round	small	blue	no
6	round	small	red	yes
7	square	small	green	no
8	round	large	green	yes
9	square	large	green	yes
10	square	large	red	no
11	square	large	green	yes
12	round	large	red	yes
13	square	small	red	no
14	round	small	green	no

Consider Second Feature

large

NO YES
NO YES
YES
YES
YES
YES

small

NO YES
NO
NO
NO
NO
NO

$$E = -\frac{2}{8}\log_2(\frac{2}{8}) - \frac{6}{8}\log_2(\frac{6}{8})$$

0.811

$$E = -\frac{6}{7}\log_2(\frac{6}{7}) - \frac{1}{7}\log_2(\frac{1}{7})$$

0.5916

weighted average

(8/15)*(0.811) + (7/15)*0.5916

0.70838

Information gain = 0.9968 – 0.708383 = **0.288**

		0.02812	0.288		
	0	round	large	blue	no
	1	square	large	green	yes
	2	square	small	red	no
	3	round	large	red	yes
	4	square	small	blue	no
	5	round	small	blue	no
	6	round	small	red	yes
	7	square	small	green	no
	8	round	large	green	yes
	9	square	large	green	yes
	10	square	large	red	no
	11	square	large	green	yes
	12	round	large	red	yes
	13	square	small	red	no
	14	round	small	green	no

Consider Third Feature

blue	green	red
NO NO NO	NO NO YES YES YES YES	NO NO NO YES YES YES

$$E = -\frac{2}{6}\log_2(\frac{2}{6}) - \frac{4}{6}\log_2(\frac{4}{6})$$

0.91829

weighted average

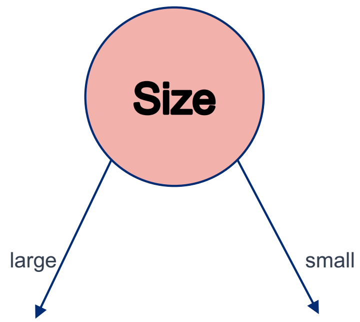
(3/15)*(0) + (6/15)*(0.91829) + (6/15)*(1)

0.7673

Information gain = 0.9968 – 0.7673 = **0.2294**

	0.02812	0.288	0.2294	
0	round	large	blue	no
1	square	large	green	yes
2	square	small	red	no
3	round	large	red	yes
4	square	small	blue	no
5	round	small	blue	no
6	round	small	red	yes
7	square	small	green	no
8	round	large	green	yes
9	square	large	green	yes
10	square	large	red	no
11	square	large	green	yes
12	round	large	red	yes
13	square	small	red	no
14	round	small	green	no

Decision Tree So Far…

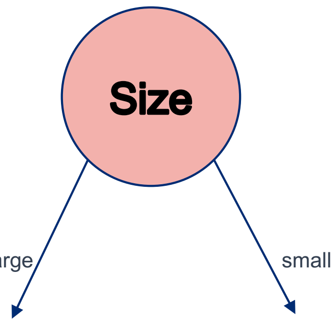


- We now know the first question to ask
- What question do we ask next? Depends on the branch we take
  - If large, look at that particular subset of data
  - Get a baseline entropy
  - Choose the feature that maximizes information gain
- When do you stop?
  - Child node is homogeneous
  - Run out of features

A path in the tree is a **rule**. Can't consider the same feature somewhere down the tree again.

0	round	large	blue	no
1	square	large	green	yes
2	square	small	red	no
3	round	large	red	yes
4	square	small	blue	no
5	round	small	blue	no
6	round	small	red	yes
7	square	small	green	no
8	round	large	green	yes
9	square	large	green	yes
10	square	large	red	no
11	square	large	green	yes
12	round	large	red	yes
13	square	small	red	no
14	round	small	green	no

Decision Tree



NO YES
NO YES
YES
YES
YES
YES

$$E = -\frac{2}{8}\log_2(\frac{2}{8}) - \frac{6}{8}\log_2(\frac{6}{8})$$

0.811

New baseline, as we consider children

0	round	large	blue	no
1	square	large	green	yes
3	round	large	red	yes
8	round	large	green	yes
9	square	large	green	yes
10	square	large	red	no
11	square	large	green	yes
12	round	large	red	yes

round

NO YES
YES
YES

$$E = -\frac{1}{4}\log_2(\frac{1}{4}) - \frac{3}{4}\log_2(\frac{3}{4})$$

0.811

square

NO YES
YES
YES

0.811

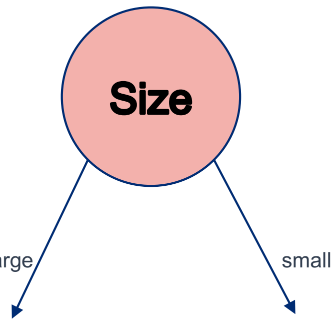
weighted average

0.811

Information gain

0

Decision Tree



NO YES
NO YES
YES
YES
YES
YES

$$E = -\frac{2}{8}\log_2(\frac{2}{8}) - \frac{6}{8}\log_2(\frac{6}{8})$$

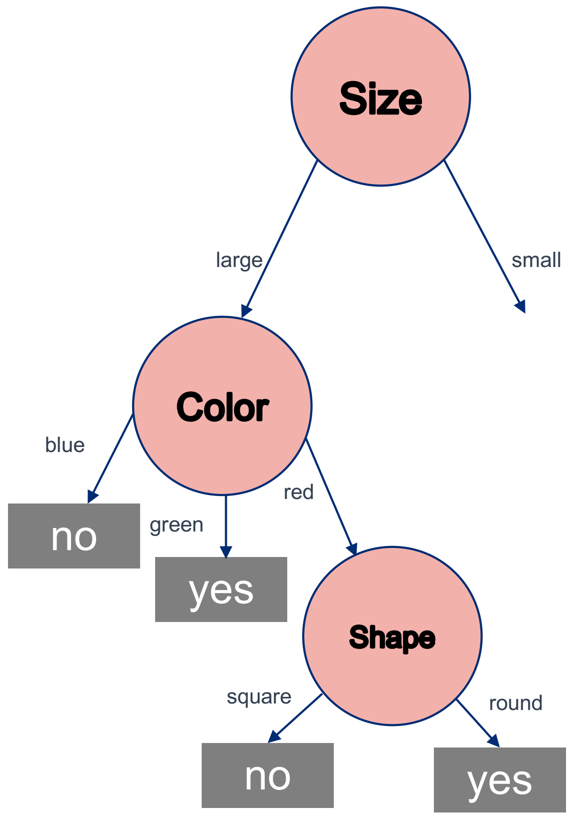
0.811

0	round	large	blue	no
1	square	large	green	yes
3	round	large	red	yes
8	round	large	green	yes
9	square	large	green	yes
10	square	large	red	no
11	square	large	green	yes
12	round	large	red	yes

blue	green	red	weighted average	Information gain
NO	YES YES YES YES	NO YES YES		
0	0	0.9182	0.344	0.467

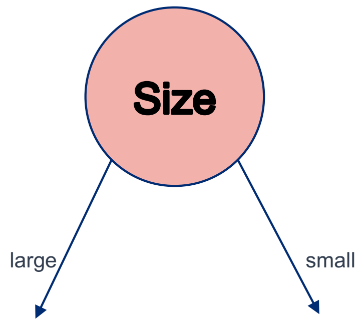
$$E = -\frac{1}{3}\log_2(\frac{1}{3}) - \frac{2}{3}\log_2(\frac{2}{3})$$

Decision Tree So Far…



0	round	large	blue	no
1	square	large	green	yes
2	square	small	red	no
3	round	large	red	yes
4	square	small	blue	no
5	round	small	blue	no
6	round	small	red	yes
7	square	small	green	no
8	round	large	green	yes
9	square	large	green	yes
10	square	large	red	no
11	square	large	green	yes
12	round	large	red	yes
13	square	small	red	no
14	round	small	green	no

The other branch + Shape



2	square	small	red	no
4	square	small	blue	no
5	round	small	blue	no
6	round	small	red	yes
7	square	small	green	no
13	square	small	red	no
14	round	small	green	no

NO YES
NO
NO
NO
NO
NO

$$E = -\frac{1}{7}\log_2(\frac{1}{7}) - \frac{6}{7}\log_2(\frac{6}{7})$$

0.5916

round

NO YES
NO

$$E = -\frac{2}{3}\log_2(\frac{2}{3}) - \frac{1}{3}\log_2(\frac{1}{3})$$

0.9182

square

NO
NO
NO
NO

0

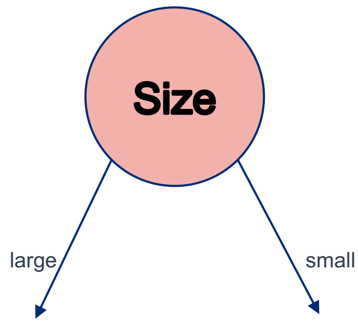
weighted average

0.393

Information gain

0.1986

The other branch + Color



2	square	small	red	no
4	square	small	blue	no
5	round	small	blue	no
6	round	small	red	yes
7	square	small	green	no
13	square	small	red	no
14	round	small	green	no

NO YES
NO
NO
NO
NO
NO

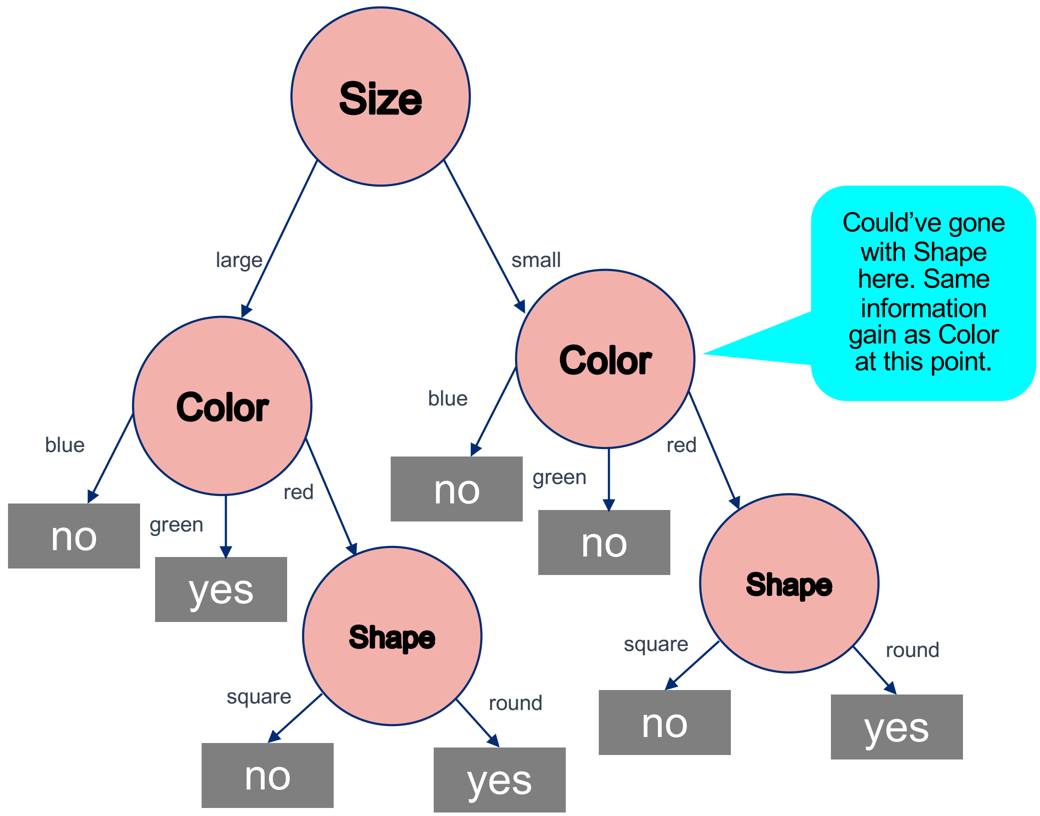
$$E = -\frac{1}{7}\log_2(\frac{1}{7}) - \frac{6}{7}\log_2(\frac{6}{7})$$

0.5916

blue	green	red	weighted average	Information gain
NO NO	NO NO	NO YES NO		
0	0	0.9182	0.393	0.1986

$$E = -\frac{2}{3}\log_2(\frac{2}{3}) - \frac{1}{3}\log_2(\frac{1}{3})$$

Decision Tree



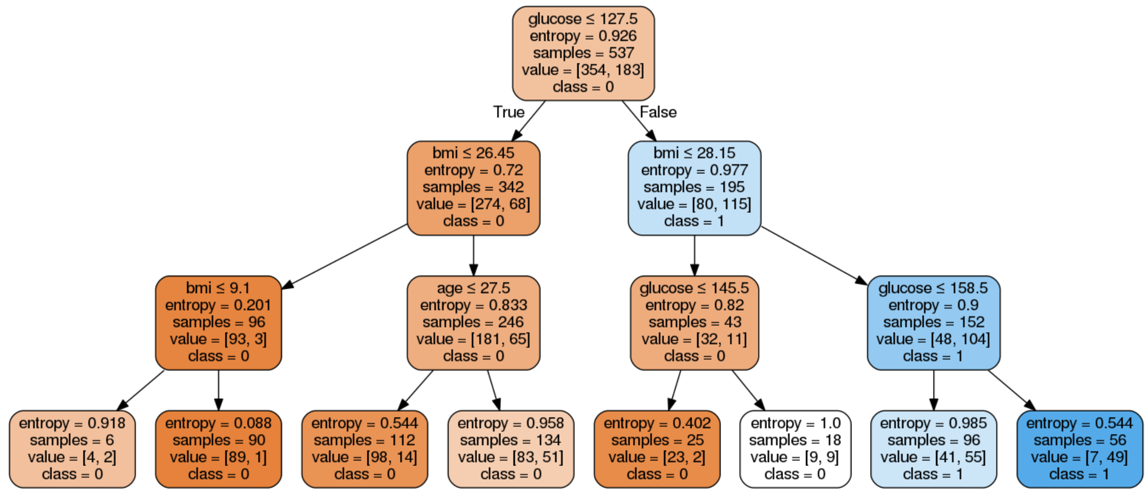
0	round	large	blue	no
1	square	large	green	yes
2	square	small	red	no
3	round	large	red	yes
4	square	small	blue	no
5	round	small	blue	no
6	round	small	red	yes
7	square	small	green	no
8	round	large	green	yes
9	square	large	green	yes
10	square	large	red	no
11	square	large	green	yes
12	round	large	red	yes
13	square	small	red	no
14	round	small	green	no

ID.3 Algorithm

```
def id3 (data, attributes, default)
      if data is empty, return default
      if data is homogeneous, return class label
      if attributes is empty, return majority_label(data)
      best_attr = pick_best_attribute(data, attributes)
      node = new Node (best_attribute)
      default_label = majority_label(data)
      for value in the domain of best_attr
            subset = examples in data where best_attr==value
            child = id3(subset, attributes – best_attr, default_label)
            add child to node
      end
      return node
end
```

Notes

- Whitebox method (as opposed to Neural Networks, which are considered as blackbox methods)
- Tailors to any inaccuracy in training data
- Notorious for overfitting
- Ways to overcome overfitting:
  - Pruning
  - C4.5 Algorithm (variation of ID.3, but replaces sections of tree at random with the majority class)
  - **Random Forests** (extremely popular five years ago; train multiple trees on subsets of the overall training data and majority rules)



https://scikit-learn.org/stable/modules/tree.html

JOHNS HOPKINS
APPLIED PHYSICS LABORATORY