Special Lecture

Bag Of Words, DNN,
Training and Testing ,
Decision Trees, Linear Classifiers

Lecture / Lab Schedule and Venue

Saturday 1: TI	Sunday 1	Saturday 2	Sunday 2
ML	TI	TI	TI
	SL	SL	AT
	IL	IL	
	Lunch		
	GL	CS	IIS / Hackathon
DL + IL			

Bag Of Words

Text Representation

Problem Statement

- Given a few newsgroup posts (text), tag the ***news group*** (such as ***hardware***)
- 20 classes (***hardware***, ***autos***, etc.)
- 1000 samples per class
- 950 for training and 50 for testing

Representations

Bag of Words (Classic)	Word2Vec (Modern)
Order-less documentation representation; frequencies of words from a dictionary.	Vectoral representation of words with meaningful inter-word distances

Bag of Words - Text Domain

- Orderless documentation representation; frequencies of words from a dictionary.

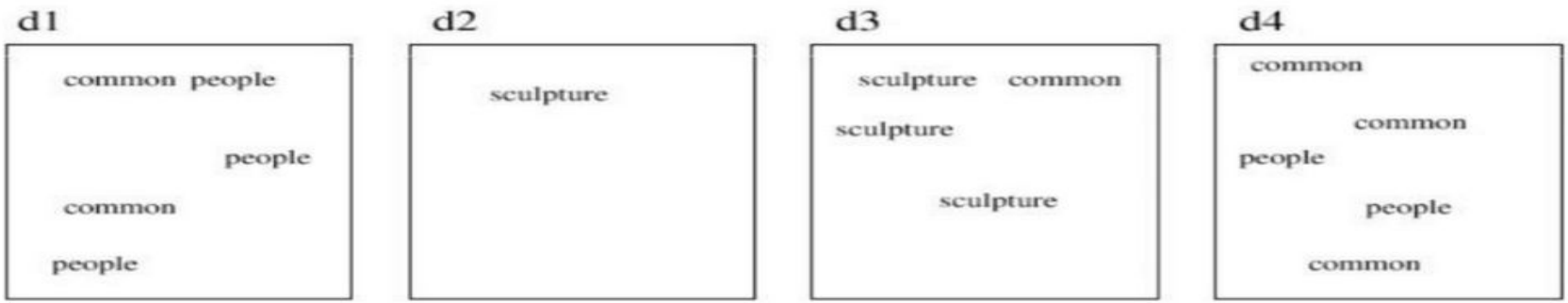
Bag of Words - Text Domain

- Orderless documentation representation, frequencies of words from a dictionary.

Bag of Words Histogram

- Orderless document representation; frequencies of words from a dictionary
- Classification to determine document categories

Bag of Words : Representation

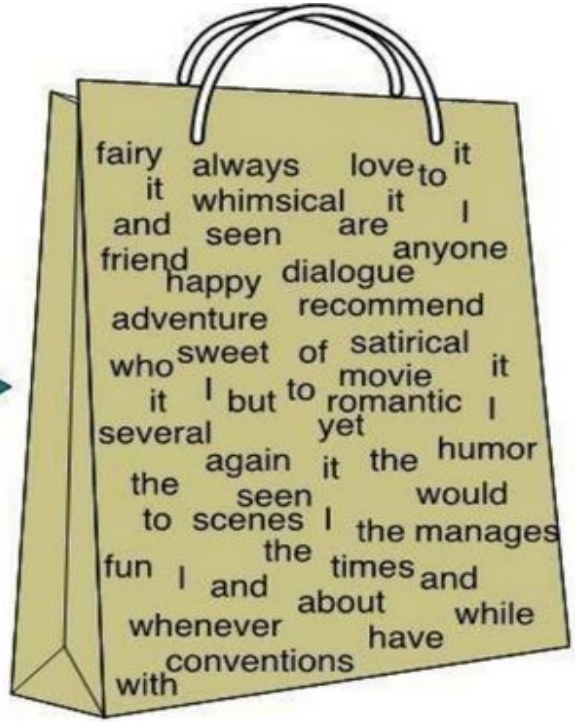


Bag-of-words

Common	2	0	1	3
People	3	0	0	2
Sculpture	0	1	3	0
...	...	...	...	...

Histogram of Word Occurrences

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!



it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1
...	...

Comments: Weight Words (eg : TF-IDF)

- Not all words are equally useful
- Stop words
- Term Frequency (Frequent words)
- Inverse Document Frequency
- Weight Words:
  - Proportional to Term Frequency
  - Inversely proportional to Inverse Document Frequency

1-hot Representation and Histograms

- Vocabulary sized vector *(V)*
- 1 at only one place else 0
- Histogram for a document
  - Add all such vectors
  - h_i = how many times i^{th} word appear

$$\begin{array}{rl} book & [0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0] \\ library & [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0] \end{array}$$

1-hot Representation and Histograms

- Disadvantages
  - No semantics
  - Distances have no utility
  - Sparse (lots of zero)
  - High dimensional

Questions?

Data Representations

Raw data

3072 X 1 Vector

Feature Vector
32 X 32 X 3 = 3072 Dimension
Per Image (d=3072)

CONCERNS:

- Too big?
- May be redundancy?

Hand Crafting Features

MIN RED
MAX RED
MEAN RED
MIN GREEN
MAX GREEN
MEAN GREEN
MIN BLUE
MAX BLUE
MEAN BLUE

9 X 1
FEATURE VECTOR
PER IMAGE

Concerns:

- **Too naïve to capture the visual content?**
- **Too small to represent information?**

Deep Learning Features

Deep Learning = End to End Learning (Raw data to labels)
Deep Learning = Feature Learning!!

Raw Image

Initial Stages of the Deep Neural Networks
Many linear and nonlinear operations

Final Stages
Final Stages
Classifier
Classifier

1000 Labels For a 1000 class classification

An intermediate representation from a popular "DeepNet", which was designed and trained for solving a "general" 1000 class classification.

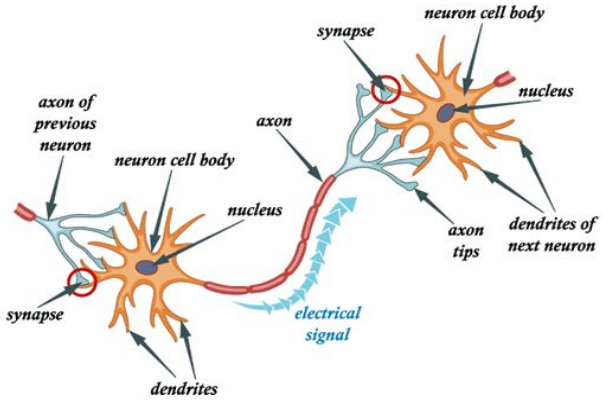
Questions?

Quick look at Neural Nets

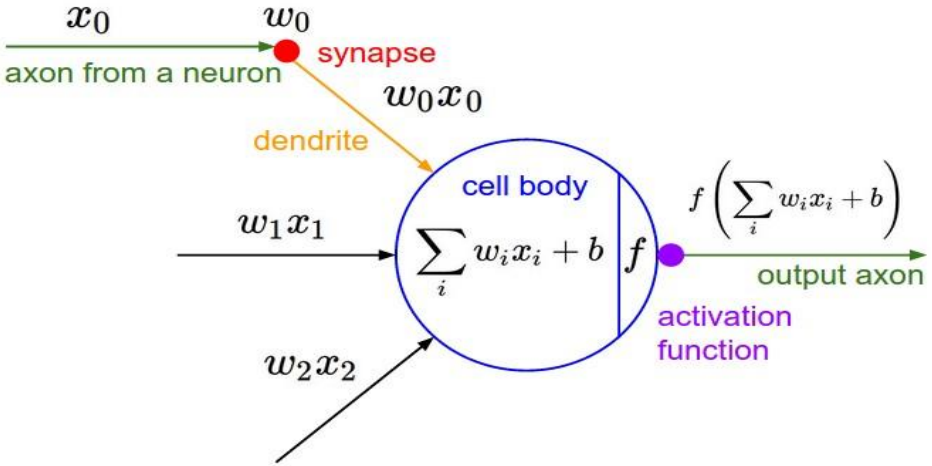
Neuron, Simple & Deep NNs

Simple Neuron

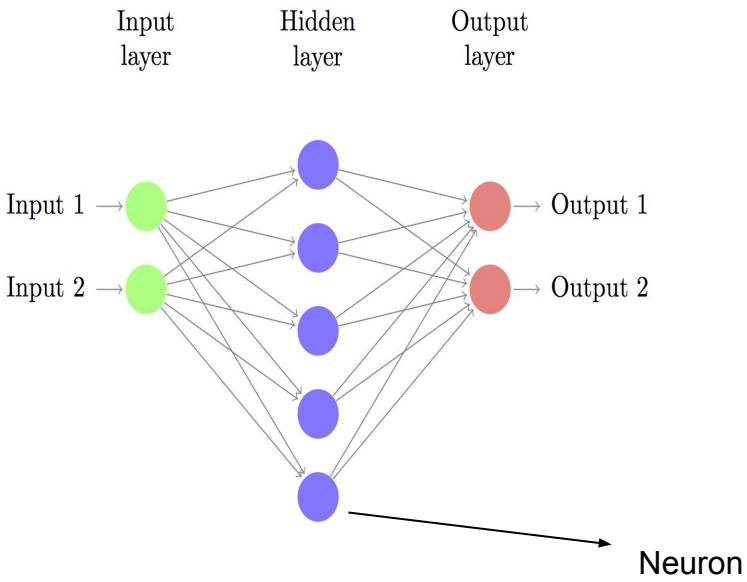
Biological Neuron



Artificial Neuron



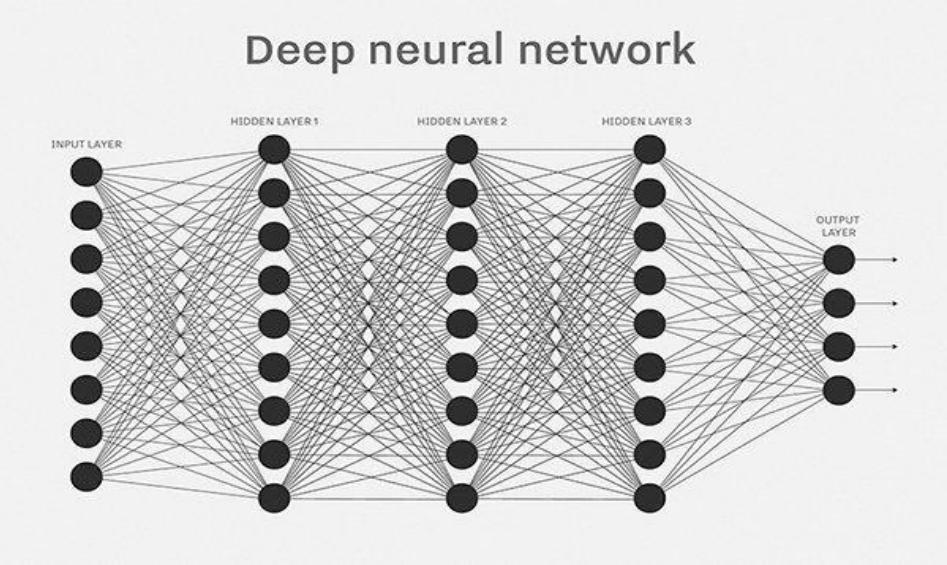
Simple Neural Network



Deep Neural Networks

Brain

DNN



Questions?

Decision Tree

Decision Model

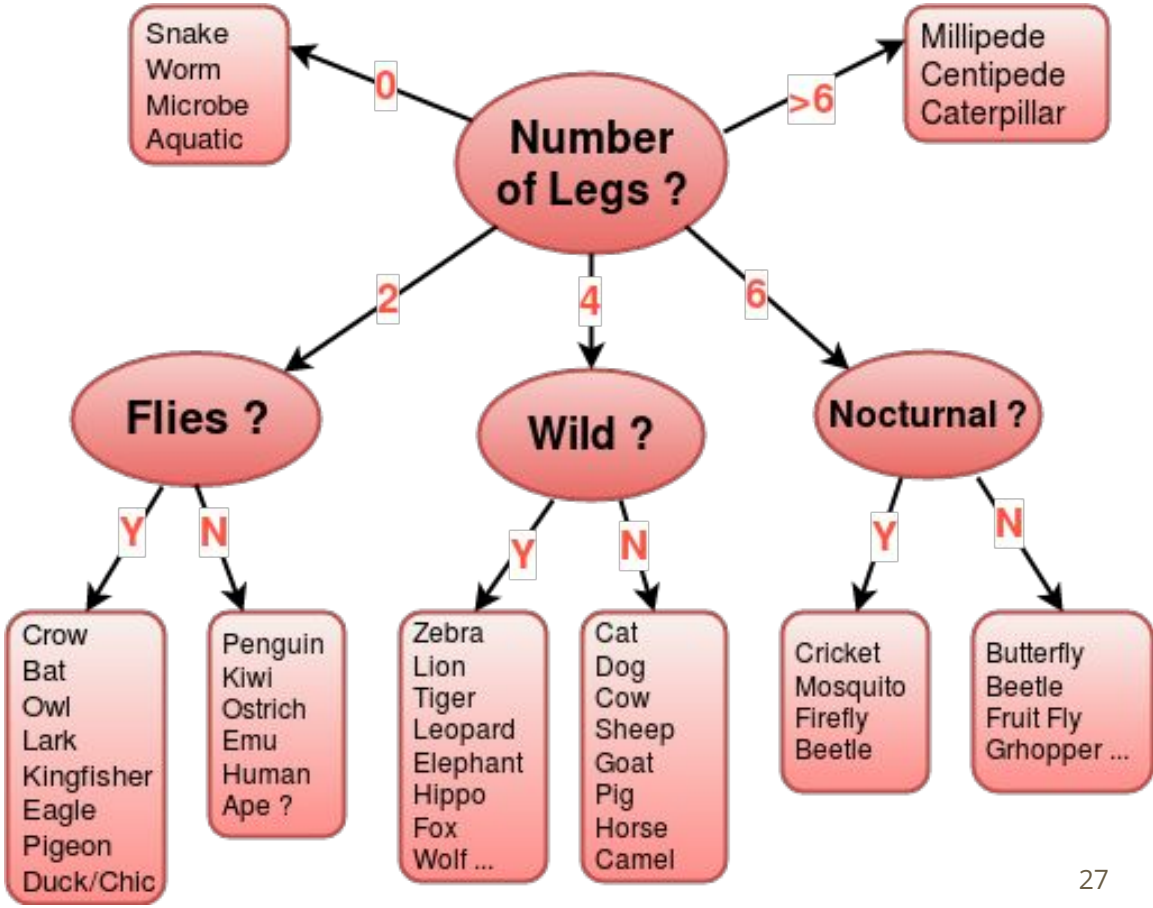
Let's play a game: Guess the Animal?

- If you think of an animal
- You can ask a set of questions
- Can you guess the animal based on my answers?
- Conditions:
  - Only single attribute questions
  - No question based on name of the animal itself

Guess the Animal

Questions

- How many legs?
- Does it fly?
- Is it a wild animal?
- Is it nocturnal?
- Fur/Feather?
- Farm Animal?



What are we doing?(Larger Picture)

Animal	Legs	Wild	Flies	Noct	Fur/feather	Farm
Zebra	4	Y	N	N	N	N
Horse	4	Y/N	N	N	N	Y
Cow	4	N	N	N	N	Y
Cat	4	Y/N	N	Y/N	Y	N
Penguin	2	Y	N	N	N	N
Owl	2	Y	Y	Y	Y	N
Fish	0	Y	N	Y/N	N	N
Snake	0	Y	N	Y/N	N	N
Millipede	1000	Y	N	Y	N	N
Firefly	6	Y	Y	Y	N	N
Butterfly	6	Y	Y	N	N	N

What are we doing?(Larger Picture)...

- We have possible animals
- Each has a set of attributes
- Look at 1 attribute at a time
- Narrow down the class label
- Goal:
  - Arrive at a single class label
- Can we learn the tree?
  - Which question to ask at any point?

What is a Good Question?

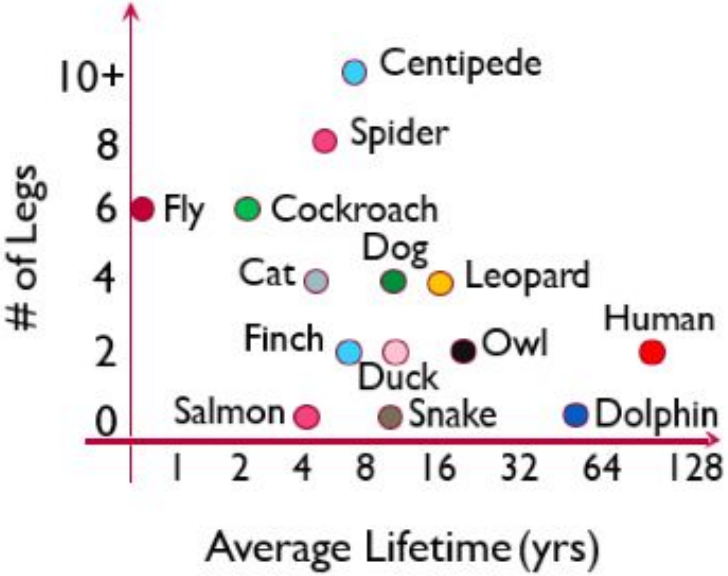
- Which question if answered will reduce the possible number of animals the most?
- More precisely, try to reduce our uncertainty the most
- Mathematically, reduce our Entropy the most

Summary on Decision Trees

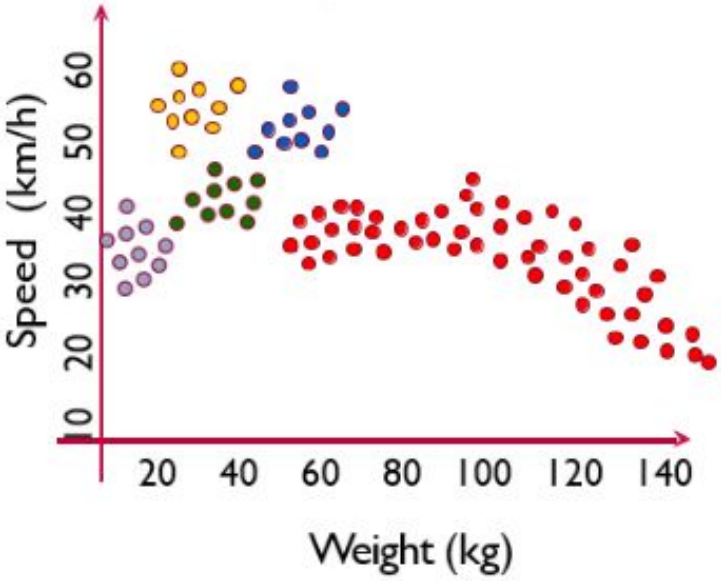
- At each step, ask the question that minimizes uncertainty
- Once the set contains only a single class, label it
- The sequence of questions represents:The Decision Tree
- Each question is on a single feature
- Tree Terms: Node, Edge, Root, Leaf, Depth, Height, Path, Parent, Child

Learning with Examples

Per Class Features



Per Sample Features

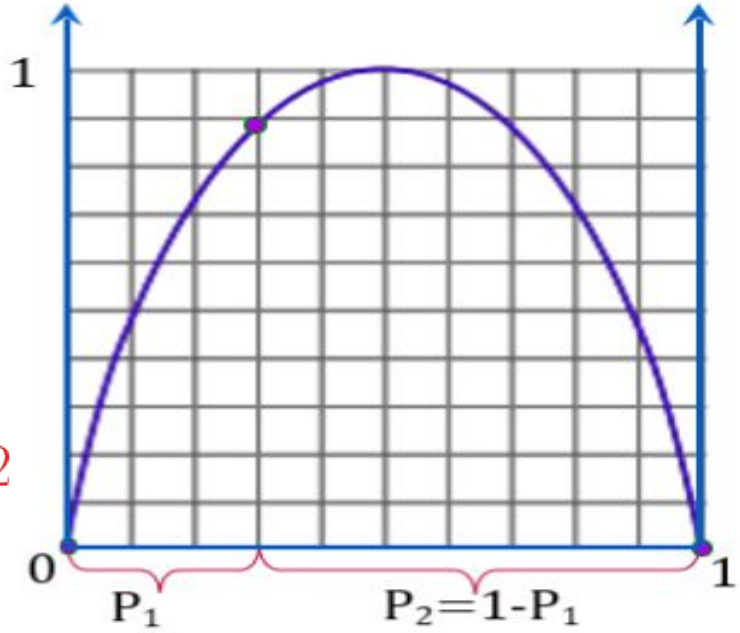


What is Entropy?

- Measure of **Uncertainty**
- Mathematically:

$$H(x) = -\sigma_i P(i) \log_2 P(i)$$

- Assume a set contains two classes:
  - $H = -P_1 \log_2 P_1 - P_2 \log_2 P_2$



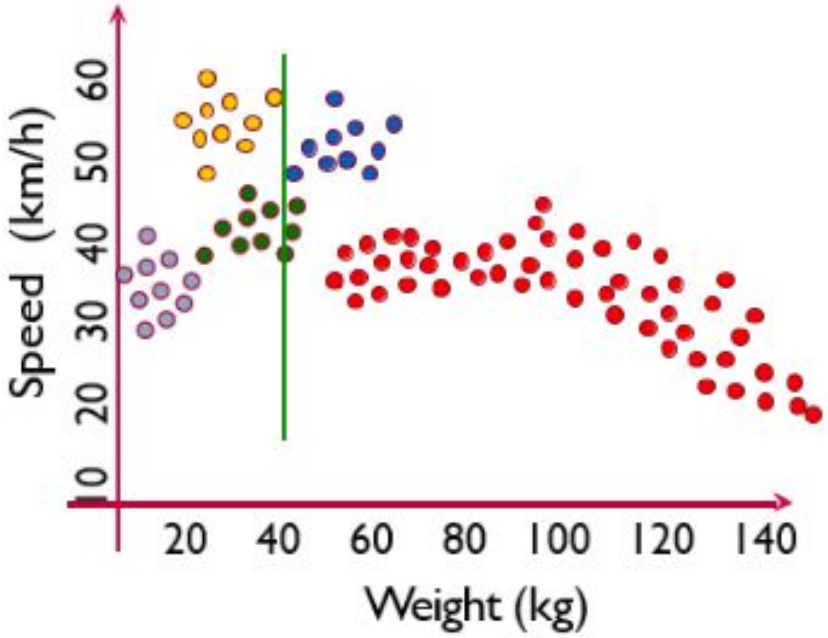
- Measure of Impurity
  - Not the only one

Computing Entropy

- Initial Entropy

$$H : 5(-0.2 \log_2 0.2) = 2.32$$

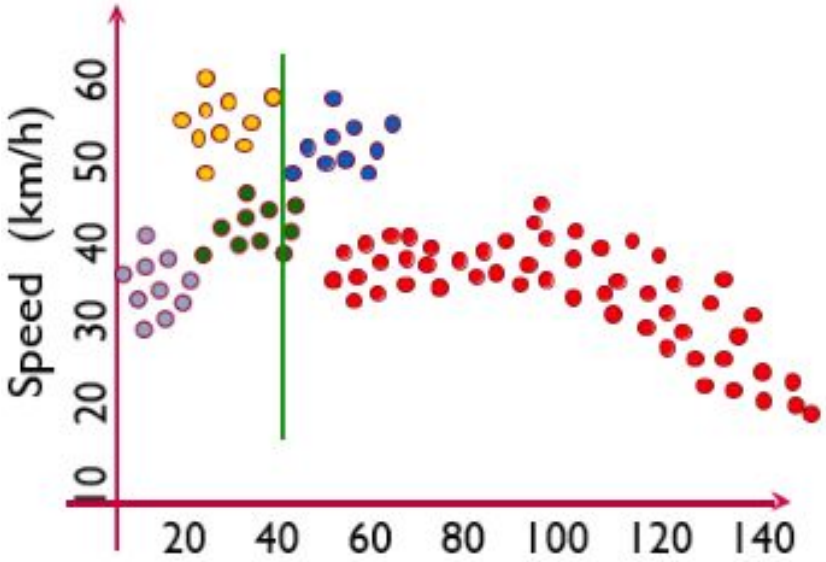
- Q1: $weight < 40$?



Computing Entropy

- Total Entropy of Children:

$H_1 = ½ (-0.4\log_2 0.4) + (-0.4 \log_2 0.4) + (-0.2\log_2 0.2)) + ½ ((-0.4\log_2 0.4) + (-0.4\log_2 0.4) + (-0.2\log_2 0.2))$



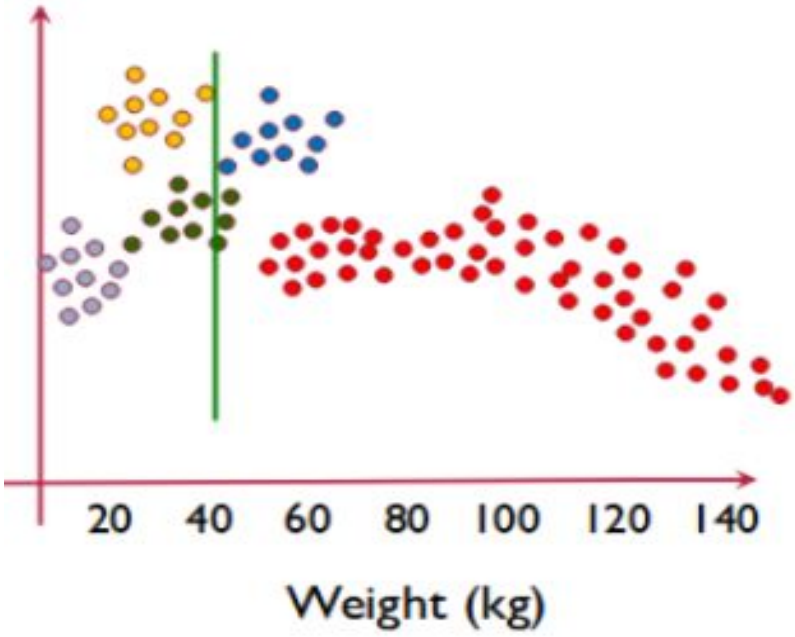
Information Gain

$$\text{Gain}(S, S_v) = \text{Entropy}(S) - \sum_v \frac{|S_v|}{|S|} Entropy(S_v) = H - H_1$$

$$H = 5 \times (-0.2\ log_2\ 0.2) = 2.32$$

$$H_1 = \frac{1}{2}\begin{pmatrix}(-0.4\ log_2 0.4) + (-0.4\ log_2 0.4) \\ + (-0.2\ log_2 0.2)\end{pmatrix} + \frac{1}{2}\begin{pmatrix}(-0.4\ log_2 0.4) + (-0.4\ log_2 0.4) \\ + (-0.2\ log_2 0.2)\end{pmatrix} = 1.52$$

$$\text{Gain}(S, S_v) = H - H_1 = 2.32 - 1.52 = 0.8$$



Best? Maximum Information Gain

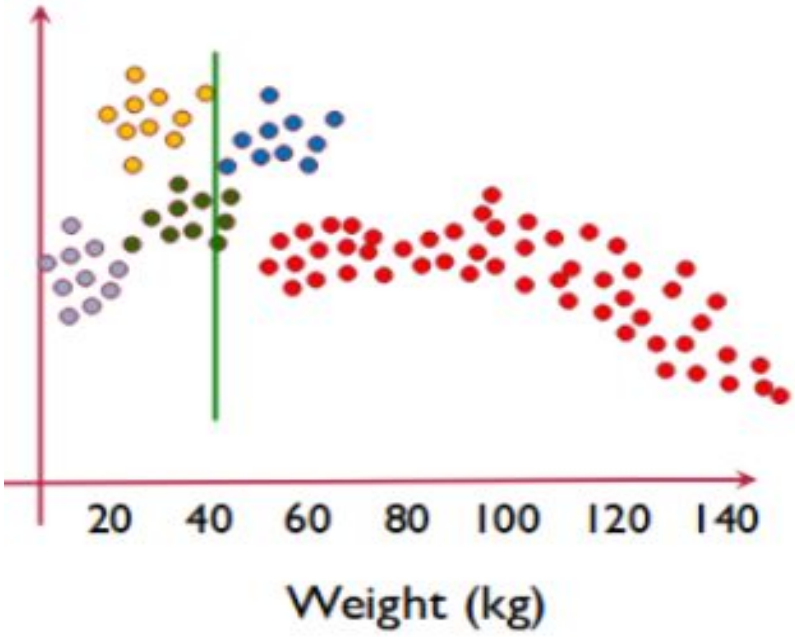
Initial Entropy: $5 \times (-0.2\, log_2\, 0.2) = 2.32$

$$H_1 = \frac{1}{2}\begin{pmatrix}(-0.4\, log_2 0.4) + (-0.4\, log_2 0.4) \\ + (-0.2\, log_2 0.2)\end{pmatrix} + \frac{1}{2}\begin{pmatrix}(-0.4\, log_2 0.4) + (-0.4\, log_2 0.4) \\ + (-0.2\, log_2 0.2)\end{pmatrix} = 1.52$$

Information Gain = 0.8

$$H_2 = \frac{1}{6}(-1.0\ \ log_2 1.0) + \frac{5}{6}\begin{pmatrix}(-0.22\, log_2 0.22) + (-0.22\, log_2 0.22) + \\ (-0.22\, log_2 0.22) + (-0.22\, log_2 0.22) + \\ (-0.12\, log_2 0.12)\end{pmatrix} = 1.91$$

Information Gain = 0.41



ID3 Algorithm

- Consider the training data and compute the impurity
- At start, all samples are at the root node
- At each step:
  - Inspect all possible features
  - Compute the information gain for each
  - Select the feature that maximizes information gain
  - Distribute data into child nodes
  - Do 1-4 recursively for each child node until pure leaf nodes arrive

Applications of Decision Trees

- Medical diagnosis
- Credit risk analysis

Advantages of Decision Trees

- Fast, Compact and Effective
- Handles categorical variables
- Interpretable as a set of rules
- Can indicate the most useful features

Disadvantages of Decision Trees

- Not suitable for prediction of continuous attribute
- Does not handle non-rectangular regions well
- Computationally expensive to train
- Tends to overfit

Splitting Data at a Node

- Random Split
- N-way split on value (categorical features)
- Binary split on threshold (continuous features)

Impurity Metrics

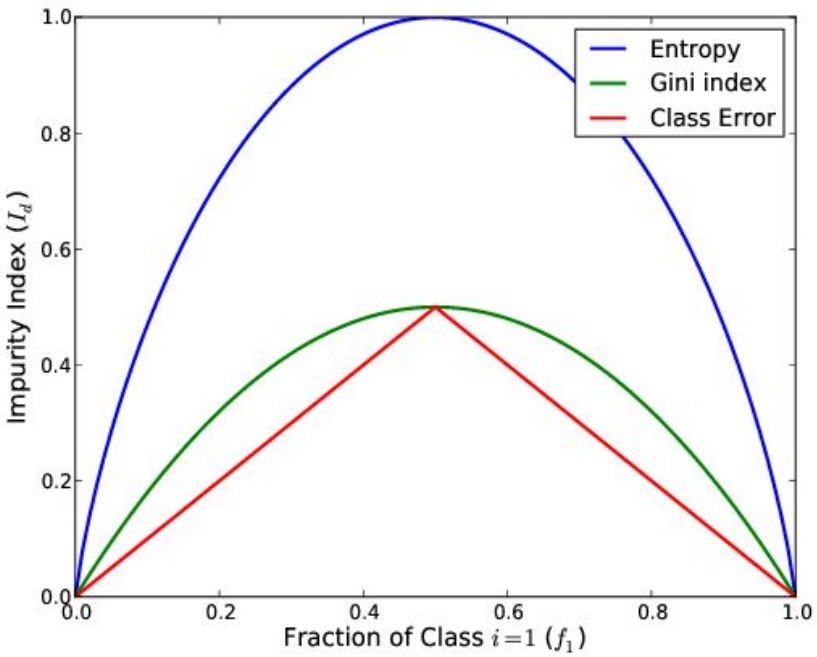
- Entropy:

$$H(x) = -\sum_{0}^{j} P(i) \log_2 P(i)$$

- GINI:

$$I(P) = \sigma P_i(1 - P_i) = 1 - \sum_{0}^{j} P_i^2$$

- Misclassification Error



(imagesource:https://www.commonlounge.com/discussion/ee8b074936a041f2b5a57d2054dc3701

CART Algorithm

- Classification and Regression Trees (Leo Breiman)
- Recursive Binary Splitting: Greedy Algorithm
  - All values of an attribute are sorted and all split points are tested
  - Test all such attributes and select the split with lowest cost
- Cost Functions:
  - Regression: MSE
  - Classification: Gini

Early Stopping and Pruning

- Decision trees are notorious for overfitting
- Early Stopping
  - Do not split beyond a point
- Pruning
  - Once the tree is formed, remove weakest branches
  - Use validation set to decide when to stop
- Both approaches also reduce the depth and improve classification speed

Handling missing values

- Why did the data go missing?
  - Random, but dependent on observed variables
  - Random, but dependent on unobserved variables
  - Dependent on the value itself!!
- Throw out data with missing values
  - What are the implications?

Handling missing values

- Impute values
  - Impute 0
  - Mean/median imputation
  - Impute from observed values: build predictor
  - Impute from last observation

DT in Scikit Learn

Training :

```
from sklearn.tree import DecisionTreeClassifier
clf_gini = DecisionTreeClassifier(criterion = “gini”,
                                  random_state = 100,
                                  max_depth = 5,
                                  min_samples_leaf = 1)
clf_gini.fit(X_train, y_train)
Z = clf_gini.predict(x,y)
```

Thanks

Questions?