- Can we do better than 'two bits' in the three message or do we need one bit '0' to say no rain everyday?
- Every system of discrete signals has some 'information' and we need those many bits to represent the system

- Entropy = $\sum_i p_i * \log(\frac{1}{p_i})$

Information Gain

- Entropy $H(S) = \sum_{i \in [K]} \frac{S_i}{S} * \log \frac{S}{S_i}$

- Information Gain due to A
  - $IG(S, A) = H(S) - H(S|A)$

    $= H(S) - \sum_t \frac{S_t}{S} H(S_t)$

    where, $t \in T$ is the different values present in S for A

Detail Compact Column

5 of 5 columns

Outlook	Temperature	Humidity	Wind	Play Tennis
Sunny 36% Rain 36% Other (4) 29%	Mild 43% Hot 29% Other (4) 29%	2 unique values	Weak 57% Strong 43%	 true 9 64% false 5 36%
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

$$H(S/A)$$

$$= \frac{5}{14} H(S_S) + \frac{4}{14} \times H(S_O) + \frac{5}{14} H(S_R)$$

$$= 0.693$$

$$I(S, A{=}\text{outlook}) = 0.247$$

$A = \text{wind}$, $T = \{S, W\}$

$$H(S_S) = \frac{1}{2}\log 2 + \frac{1}{2}\log 2 = 1$$

$$H(S_W) = \frac{6}{8}\log\frac{8}{6} + \underbrace{\frac{2}{8}\log 4}_{0.5}$$

$$=$$

Yes, No

$H(S) = 0.94$

A = outlook

$T = \{S, O, R\}$

$H(S_{sunny}) = \frac{2}{5}\log\frac{5}{2} + \frac{3}{5}\log\frac{5}{3}$

$= 0.971$

$H(S_O) = 0$

$H(S_R) = \frac{3}{5}\log\frac{5}{3}$

$+ \frac{2}{5}\log\frac{5}{2}$

$= 0.971$

$H(S/A)$

$= \frac{5}{14} H(S_S) + \frac{4}{14} \times H(S_O)$

$= 0.693$

$I(S, A{:}\text{outlook}) = 0$

CART

- Classification and Regression Tree
  - Key idea:
  - Only Two Children
  - Some goodness criterion to split (Typically minimize Ginni Index based)
    - $\sum p_t(1 - p_t)$
  - Pruning to avoid overfitting (Typically Information Gain)

In general,
Decision trees, good for explaining decisions
However, very sensitive to noise in the data, small change in the data can lead to a very different different tree

Bayes vs Naïve Bayes vs Decision Tree

- Optimal
- This is ultimate goal
- practically difficult to learn Bayes classifier directly from the data
- Curse of Dimensionality

- Easy to build
- Not all features independent
- Explainability is poor

- Easy to build
- Explainability is good
- Overfitting
- sensitivity

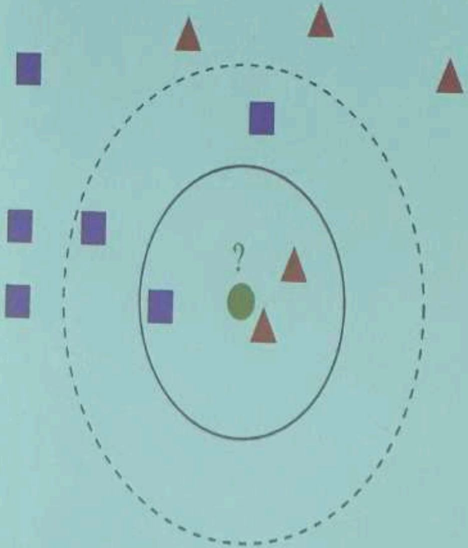
kNN

- 'k' nearest neighbours
- Data set $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$

- Given a point x, find out k nearest point from D to x
- E.g., say $|x_{\sigma(1)} - x| <= |x_{\sigma(2)} - x| <= \dots <= |x_{\sigma(n)} - x|$
- Collect labels of $x_{\sigma(1)}, x_{\sigma(2)}, \dots, x_{\sigma(k)}$
- $\hat{y}$= the most frequently occurring label in $x_{\sigma(1)}, x_{\sigma(2)}, \dots, x_{\sigma(k)}$

What will be label if $k = 3$?

What will be label if $k = 5$?



- How to choose right k?
- On the given data,
  - estimate error on different k values
  - select the one with least error
  - Choose the k value where a small decrease in k causes a large increase in error and increase in k results small decrease in the error

- Running time, linear in $n \times d$
- As $n \to \infty$, 1-NN error rate is at most $2 \times Bayes\ error\ rate$
- How to measure nearness?
  - If features are continuous, Euclidian metric
  - Discrete features: hamming distance

What is metric?

- Drawback
  - Frequent classes dominate