CSE 417T Machine Learning

Hw7, Due: Dec.1, 2016

Chih Yun Pai

1.

The greedy decision tree learning algorithm might not return the exactly same tree, because the decision tree algorithm generates decision logic based on training data set rather than the entire structure of the tree. So this does not guarantee getting back the same tree. However, the decision learning algorithm does guarantee the exactly same “decision logic”, if training error is zero.

2.

$\mathrm{S} \ = \ [4-,2+]$

$\mathrm{Value(Cylinders)} \ = \ [4{,}8]$

$\mathrm{S}_4 = \ [2-,1+]$

$\mathrm{S}_8 = \ [2-,1-]$

$Gain(S, Cylinder) \ = \ Entropy(\mathrm{S}) - \ \sum_{v \in value(Cylinders)} \frac{|S_v|}{|S|} Entropy(S_v)$

$= Entropy(\mathrm{S}) - \frac{|S_4|}{|S|} Entropy(S_4) - \frac{|S_8|}{|S|} Entropy(S_8)$

$= \left[-\frac{4}{4+2}\log_2\frac{4}{4+2} - \frac{2}{4+2}\log_2\frac{2}{4+2}\right] - \frac{2+1}{6}\left[-\frac{2}{2+1}\log_2\frac{2}{2+1} - \frac{1}{2+1}\log_2\frac{1}{2+1}\right] - \frac{2+1}{6}\left[-\frac{2}{2+1}\log_2\frac{2}{2+1} - \frac{1}{2+1}\log_2\frac{1}{2+1}\right]$

$= 0.9183 - 0.3993 - 0.3993$

$= 0.1197$

$\mathrm{Value(Displacement)} \ = \ [\mathrm{Low, Medium, High}]$

$S_{Low} = [1-,1+]$

$S_{Medium} = [1-,0+]$

$S_{High} = [2-,1+]$

$Gain(S, Displacement) \ = \ Entropy(\mathrm{S}) - \ \sum_{v \in value(Displacement)} \frac{|S_v|}{|S|} Entropy(S_v)$

$= Entropy(\mathrm{S}) - \frac{|S_{Low}|}{|S|} Entropy(S_{Low}) - \frac{|S_{Medium}|}{|S|} Entropy(S_{Medium}) - \frac{|S_{High}|}{|S|} Entropy\left(S_{High}\right)$

$= 0.9183 - \frac{2}{6}\left[-\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}\right] - \frac{1}{6}[-(1)\log_2(1) - (0)\log_2(0)] - \frac{3}{6}\left[-\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}\right]$

$= 0.9183 - 0.3333 - 0 - 0.0119$

$= 0.5731$

$\mathrm{Value(Horsepower)} \ = \ [\mathrm{Low, Medium, High}]$

$S_{Low} = [1-,0+]$

$S_{Medium} = [1-,2+]$

$S_{High} = [2-,0+]$

$Gain(S, Horsepower) \ = \ Entropy(\mathrm{S}) - \ \sum_{v \in value(Horsepower)} \frac{|S_v|}{|S|} Entropy(S_v)$

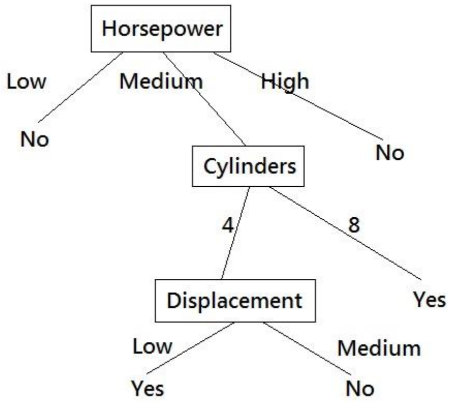
$$= Entropy(S) - \frac{|S_{Low}|}{|S|} Entropy(S_{Low}) - \frac{|S_{Medium}|}{|S|} Entropy(S_{Medium}) - \frac{|S_{High}|}{|S|} Entropy(S_{High})$$

$$= 0.9183 - \frac{1}{6}[-(1)\log_2(1)] - \frac{3}{6}\left[-\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}\right] - \frac{2}{6} - (1)\log_2(1)]$$

$$= 0.9183 - 0 - 0.129 - 0$$

$$= 0.7893$$

$\Rightarrow$ The first feature should choose "Horsepower"



3.

If the data set is linearly separable, a decision tree might and might not correctly separate the data. For example, two clusters might be inclined to x1, x2 axis and overlap over a certain cutoff value either on x1 or x2. (See, fig2 and fig3)

If the data set is not linearly separable, a decision tree still might and might not correctly separate the data. (See fig4 and fig5)

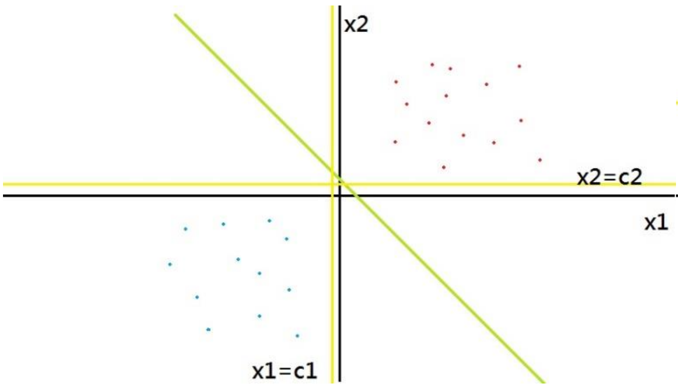


Fig2: linearly separable & can be correctly classified

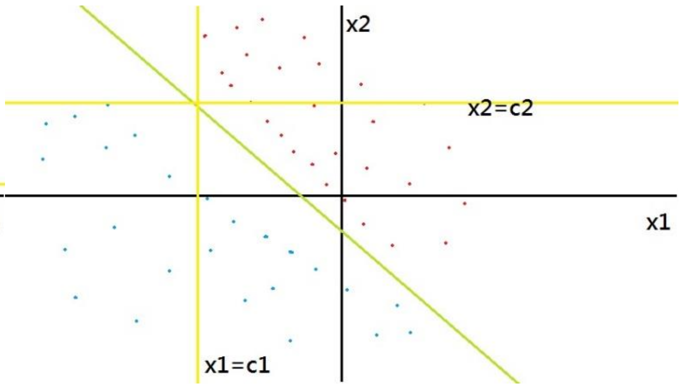


Fig3: linearly separable & can not be correctly classified

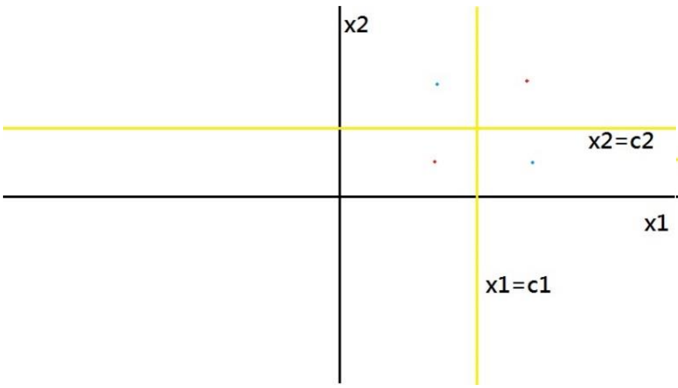


Fig4: not linearly separable & can be correctly classified

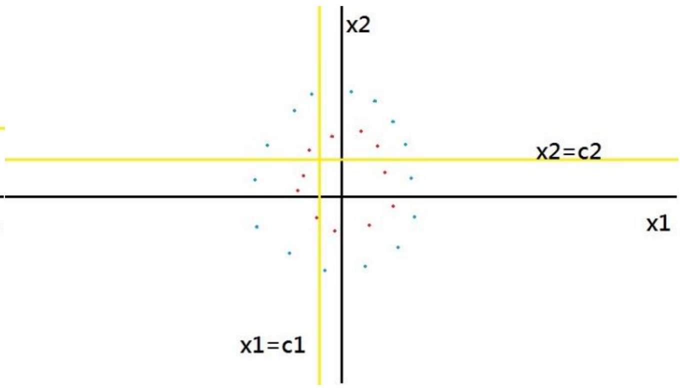


Fig5: not linearly separable & can not be correctly classified

4.

Boosting is linear.

Bagging is not linear.

5.

X1	X2	X1 XOR x2	X1X2
1	1	0	1
1	-1	1	-1
-1	1	1	-1
-1	-1	0	1

In X1, X1X2 mapping, the separator is line: X1X2=0, and its margin is 1.

Transforming the separator from X1, X1X2 space to the original space X1, X2, the separator becomes the axises, (X1=0||X2=0)

6.

$$
\begin{aligned}
&\mathrm{K}(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j) \\
&D_z = (z_i - z_j)^2 \\
&= (\Phi(x_i) - \Phi(x_j))^2 \\
&= (\Phi(x_i))^2 - 2\Phi(x_i) \cdot \Phi(x_j) + (\Phi(x_j))^2 \\
&= \Phi(x_i) \cdot \Phi(x_i) - 2\mathrm{K}(x_i, x_j) + \Phi(x_j) \cdot \Phi(x_j) \\
&= \mathrm{K}(x_i, x_i) - 2\mathrm{K}(x_i, x_j) + \mathrm{K}(x_j, x_j)
\end{aligned}
$$

Bonus Problem

(a)

$$p_n(1,1) = 1$$

$$p_n(m, 1) = \begin{cases} 0, m > 1 \\ 1, m = 1 \end{cases}$$

$$p_n(1, k) = \begin{cases} 1, k = 1 \\ \left(\frac{1}{n}\right)^{k-1}, k > 1 \end{cases}$$

(b)

Not increase m : $p_n = \frac{m}{n}$

Will increase m: $p_n = \frac{n-m}{n}$

(c)

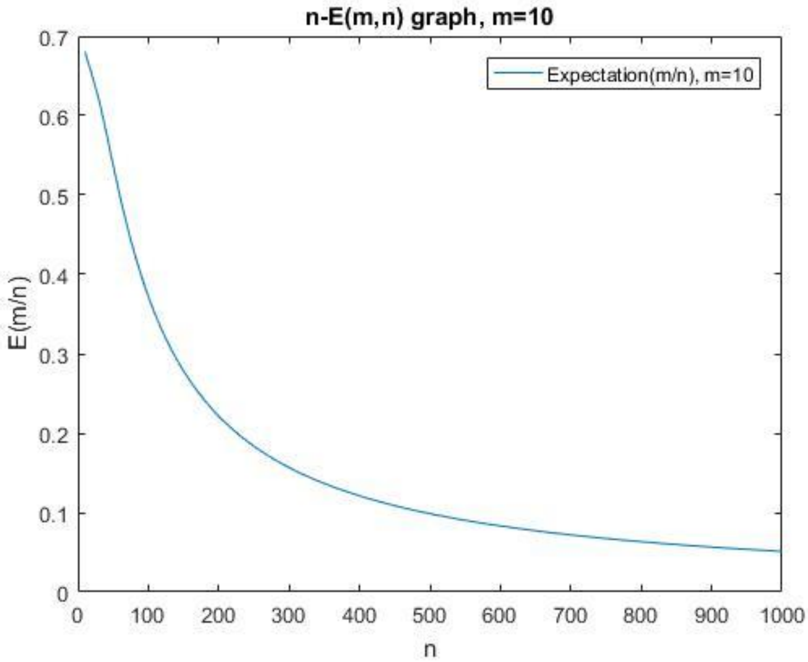
$$p_n(m, k) = \frac{m}{n} p_n(m, k-1)$$

$$= \frac{m-m+1}{n} p_n(m-1, k-1)$$

(d)

$$E_{k=n}\left[\frac{m}{n}\right] = \frac{1}{n}\sum_{i=1}^{n} i\left(\frac{i}{n}p_n(i, n-1) + \frac{n-i+1}{n}p_n(i-1, n-1)\right)$$

(e)



As n increases, E(10,n) converges around 0.05

(f)