1. UNIT 1 Q#14 Solution - Decision tree classification

Consider the dataset below

A	B	Class Label
T	F	+
T	T	+
T	T	+
T	F	−
T	T	+
F	F	−
F	F	−
F	F	−
T	T	−
T	F	−

(a) Calculate the information gain when splitting on A and B. Which attribute would the decision tree induction algorithm choose?

Answer:

The contingency tables after splitting on attributes A and B are:

	$A = T$	$A = F$
+	4	0
−	3	3

	$B = T$	$B = F$
+	3	1
−	1	5

The overall entropy before splitting is:

$$E_{orig} = -0.4 \log 0.4 - 0.6 \log 0.6 = 0.9710$$

The information gain after splitting on A is:

$$\begin{aligned} E_{A=T} &= -\frac{4}{7} \log \frac{4}{7} - \frac{3}{7} \log \frac{3}{7} = 0.9852 \\ E_{A=F} &= -\frac{3}{3} \log \frac{3}{3} - \frac{0}{3} \log \frac{0}{3} = 0 \\ \Delta &= E_{orig} - 7/10 E_{A=T} - 3/10 E_{A=F} = 0.2813 \end{aligned}$$

The information gain after splitting on B is:

$$\begin{aligned}
E_{B=T} &= -\frac{3}{4}\log\frac{3}{4} - \frac{1}{4}\log\frac{1}{4} = 0.8113 \\
E_{B=F} &= -\frac{1}{6}\log\frac{1}{6} - \frac{5}{6}\log\frac{5}{6} = 0.6500 \\
\Delta &= E_{orig} - 4/10 E_{B=T} - 6/10 E_{B=F} = 0.2565
\end{aligned}$$

Therefore, attribute A will be chosen to split the node.

7. Consider the data set shown in Table 5.1

Table 5.1. Data set for Exercise 7.

Record	A	B	C	Class
1	0	0	0	+
2	0	0	1	−
3	0	1	1	−
4	0	1	1	−
5	0	0	1	+
6	1	0	1	+
7	1	0	1	−
8	1	0	1	−
9	1	1	1	+
10	1	0	1	+

(a) Estimate the conditional probabilities for $P(A|+)$, $P(B|+)$, $P(C|+)$, $P(A|-)$, $P(B|-)$, and $P(C|-)$.

Answer:

$P(A = 1|-) = 2/5 = 0.4$, $P(B = 1|-) = 2/5 = 0.4$,
$P(C = 1|-) = 1$, $P(A = 0|-) = 3/5 = 0.6$,
$P(B = 0|-) = 3/5 = 0.6$, $P(C = 0|-) = 0$; $P(A = 1|+) = 3/5 = 0.6$,
$P(B = 1|+) = 1/5 = 0.2$, $P(C = 1|+) = 2/5 = 0.4$,
$P(A = 0|+) = 2/5 = 0.4$, $P(B = 0|+) = 4/5 = 0.8$,
$P(C = 0|+) = 3/5 = 0.6$.

(b) Use the estimate of conditional probabilities given in the previous question to predict the class label for a test sample $(A = 0, B = 1, C = 0)$ using the naïve Bayes approach.

Answer:

Let $P(A = 0, B = 1, C = 0) = K$.

$$\begin{aligned}
& P(+|A = 0, B = 1, C = 0) \\
= & \frac{P(A = 0, B = 1, C = 0|+) \times P(+)}{P(A = 0, B = 1, C = 0)} \\
= & \frac{P(A = 0|+)P(B = 1|+)P(C = 0|+) \times P(+)}{K} \\
= & 0.4 \times 0.2 \times 0.6 \times 0.5/K \\
= & 0.024/K.
\end{aligned}$$

$$\begin{aligned}
& P(-|A = 0, B = 1, C = 0) \\
= & \frac{P(A = 0, B = 1, C = 0|-) \times P(-)}{P(A = 0, B = 1, C = 0)} \\
= & \frac{P(A = 0|-) \times P(B = 1|-) \times P(C = 0|-) \times P(-)}{K} \\
= & 0/K
\end{aligned}$$

The class label should be '+'.

2. UNIT 1: Q#15 - Bayes optimal classifier

In general, the most probable classification of the new instance is obtained by combining the predictions of all hypotheses, weighted by their posterior probabilities. If the possible classification of the new example can take on any value ***vj*** from some set V, then the probability P(vjlD) that the correct classification for the new instance is v;, is just

$$P(v_j|D) = \sum_{h_i \in H} P(v_j|h_i)P(h_i|D)$$

The optimal classification of the new instance is the value ***v,,*** for which
P (v; 1 D) is maximum

Bayes optimal classification:

$$\underset{v_j \in V}{\operatorname{argmax}} \sum_{h_i \in H} P(v_j|h_i)P(h_i|D)$$

To illustrate in terms of the above example, the set of possible classifications
of the new instance is V = (+,-)

$$P(h_1|D) = .4,\ P(\ominus|h_1) = 0,\ P(\oplus|h_1) = 1$$
$$P(h_2|D) = .3,\ P(\ominus|h_2) = 1,\ P(\oplus|h_2) = 0$$
$$P(h_3|D) = .3,\ P(\ominus|h_3) = 1,\ P(\oplus|h_3) = 0$$

$$\sum_{h_i \in H} P(\oplus|h_i)P(h_i|D) = .4$$
$$\sum_{h_i \in H} P(\ominus|h_i)P(h_i|D) = .6$$

$$\underset{v_j \in \{\oplus,\ominus\}}{\operatorname{argmax}} \sum_{h_i \in H} P(v_j|h_i)P(h_i|D) = \ominus$$

3. UNIT 2 : Q#20: Back Propagation Weights and Bias Calculation

Sample calculations for learning by the backpropagation algorithm. Figure 9.5 shows a multilayer feed-forward neural network. Let the learning rate be 0.9. The initial weight and bias values of the network are given in Table 9.1, along with the first training tuple, $X = (1, 0, 1)$, with a class label of 1.

This example shows the calculations for backpropagation, given the first training tuple, X. The tuple is fed into the network, and the net input and output of each unit are computed. These values are shown in Table 9.2. The error of each unit is computed and propagated backward. The error values are shown in Table 9.3. The weight and bias updates are shown in Table 9.4. ■

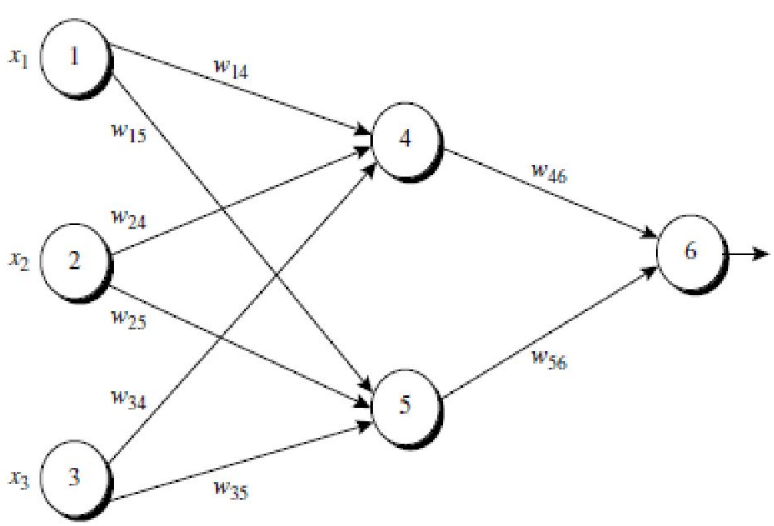

Figure 9.5 Example of a multilayer feed-forward neural network.

Table 9.1 Initial Input, Weight, and Bias Values

x_1	x_2	x_3	w_{14}	w_{15}	w_{24}	w_{25}	w_{34}	w_{35}	w_{46}	w_{56}	θ_4	θ_5	θ_6
1	0	1	0.2	−0.3	0.4	0.1	−0.5	0.2	−0.3	−0.2	−0.4	0.2	0.1

Table 9.2 Net Input and Output Calculations

Unit, j	*Net Input, I_j*	*Output, O_j*
4	$0.2+0-0.5-0.4=-0.7$	$1/(1+e^{0.7})=0.332$
5	$-0.3+0+0.2+0.2=0.1$	$1/(1+e^{-0.1})=0.525$
6	$(-0.3)(0.332)-(0.2)(0.525)+0.1=-0.105$	$1/(1+e^{0.105})=0.474$

Table 9.3 Calculation of the Error at Each Node

Unit, j	*Err_j*
6	$(0.474)(1-0.474)(1-0.474)=0.1311$
5	$(0.525)(1-0.525)(0.1311)(-0.2)=-0.0065$
4	$(0.332)(1-0.332)(0.1311)(-0.3)=-0.0087$

Table 9.4 Calculations for Weight and Bias Updating

Weight or Bias	*New Value*
w_{46}	$-0.3+(0.9)(0.1311)(0.332)=-0.261$
w_{56}	$-0.2+(0.9)(0.1311)(0.525)=-0.138$
w_{14}	$0.2+(0.9)(-0.0087)(1)=0.192$
w_{15}	$-0.3+(0.9)(-0.0065)(1)=-0.306$
w_{24}	$0.4+(0.9)(-0.0087)(0)=0.4$
w_{25}	$0.1+(0.9)(-0.0065)(0)=0.1$
w_{34}	$-0.5+(0.9)(-0.0087)(1)=-0.508$
w_{35}	$0.2+(0.9)(-0.0065)(1)=0.194$
θ_6	$0.1+(0.9)(0.1311)=0.218$
θ_5	$0.2+(0.9)(-0.0065)=0.194$
θ_4	$-0.4+(0.9)(-0.0087)=-0.408$

4. Unit 2: Q#15 : Two Input Perceptron - XOR

4.2. Design a two-input perceptron that implements the boolean function $A \wedge \neg B$. D network of perceptrons that implements A XOR B.

Ans. We assume 1 for true, -1 for false.

(1) $A \wedge \neg B$: w0 = -0.8. w1 = 0.5, w2 = -0.5.

x1(A)	x2(B)	w0+w1x1+w2x2	output
-1	-1	-0.8	-1
-1	1	-1.8	-1
1	-1	0.2	1
1	1	-0.8	-1

5. Unit 3: Q#18 - Optimal Decision Rule - Bayesian Decision Theory

. *In a two-class, two-action problem, if the loss function is* $\lambda_{11} = \lambda_{22} = 0$, $\lambda_{12} = 10$, *and* $\lambda_{21} = 1$, *write the optimal decision rule.*

We calculate the expected risks of the two actions:

$$
\begin{aligned}
R(\alpha_1|x) &= \lambda_{11}P(C_1|x) + \lambda_{12}P(C_2|x) = 10P(C_2|x) \\
R(\alpha_2|x) &= \lambda_{21}P(C_1|x) + \lambda_{22}P(C_2|x) = P(C_1|x)
\end{aligned}
$$

and we choose C_1 if $R(\alpha_1|x) < R(\alpha_2|x)$, or if $P(C_1|x) > 10P(C_2|x)$, $P(C_1|x) > 10/11$. Assigning accidentally an instance of C_2 to C_1 is so bad that we need to be very sure before assigning an instance to C_1.

6. Unit 5 Q#13 – Q-Learning algorithm

Q learning algorithm

For each *s* , a initialize the table entry Q~(s,a) to zero.

Observe the current state s

Do forever:

Select an action a and execute it

Receive immediate reward r

Observe the new state s'

Update the table entry for Q$(s,$a$,)$ as follows:

$$Q(s,a) = r + R\max Q(s',a')$$

$x=s'$