Exercises in Learning Systems (DVA427)-Reference Solutions

Some questions from previous exams

1. Decision Tree Learning

Construct the decision tree for the set of training instances in the left table with attributes x_1, x_2 and classes +, –.

a) Which attribute should be used at the root node and why?

b) Show how the examples D_1, …., D_{10} are sorted down the tree and how they are classified.

Use the right table to look up the entropy for a subset of training instances. Take the entry that is closest to the decimal number for which you want to calculate the entropy (e.g. if you need to calculate the entropy of 0.33, take the entry for 0.3 in the table which is 0.9). It is sufficient to make approximate calculations rounded to one digit behind the decimal.

Training examples

No.	x_1	x_2	$c(x_1,x_2)$
D_1	0	0	+
D_2	0	0	+
D_3	0	0	+
D_4	0	0	+
D_5	0	1	-
D_6	0	1	-
D_7	0	1	-
D_8	1	0	-
D_9	1	1	+
D_{10}	1	1	+

p	entropy(p)
0.0	0.0
0.1	0.5
0.2	0.7
0.3	0.9
0.4	1.0
0.5	1.0
0.6	1.0
0.7	0.9
0.8	0.7
0.9	0.5
1.0	0.0

Answer:

Let G={D1, D2, D3, D4, D5, D6, D7, D8, D9, D10}.

If we divide G with x_1, we get two subgroups {D1~D7} and {D8~D10}

E{D1~D7}=1, E{D8~D10}=0.9

$$Gain(x_1, G) = 1 - \frac{7}{10} \times 1 - \frac{3}{10} \times 0.9 = 0.03$$

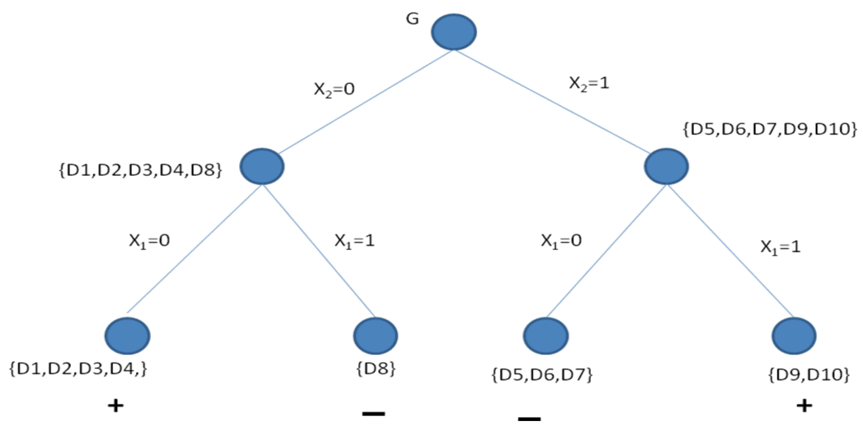
If we divide G with x_2, we get two subgroups {D1, D2, D3, D4, D8} and {D5, D6, D7, D9, D10 }

E{D1, D2, D3, D4, D8}=0.7, E{D5, D6, D7, D9, D10}=1

$$Gain(x_2, G) = 1 - \frac{5}{10} \times 0.7 - \frac{5}{10} \times 1 = 0.15$$

As x_2 has a larger information gain, x_2 is used at the root node.

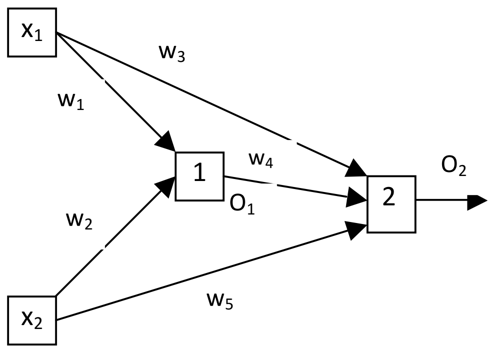
The training examples are sorted as follows:



2. Artificial Neural Network

Consider a neural network shown in the following figure. The outputs from the two units are given by

$$O_1 = sigmoid(w_1x_1 + w_2x_2 + c_1)$$
$$O_2 = sigmoid(w_3x_1 + w_4O_1 + w_5x_2 + c_2)$$



Now you are given a training example (x_{10}, x_{20}, t_0). Suppose that, under this training example, the outputs of the two units are O_{10} and O_{20} respectively, and the current value for weight w_4 is w_{40}. The question is how to update the weights and thresholds for the neuron units in light of this training example. Assume the learning rate is γ, please write out the formulas to calculate Δw_i and Δc_i in terms of the incremental BP algorithm.

Answer: $\delta_2 = (t_0 - O_{20})O_{20}(1 - O_{20})$

$$\Delta w_3 = \gamma\delta_2 x_{10}, \quad \Delta w_4 = \gamma\delta_2 O_{10}, \quad \Delta w_5 = \gamma\delta_2 x_{20}, \quad \Delta c_2 = \gamma\delta_2 \cdot 1$$
$$\delta_1 = O_{10}(1 - O_{10})w_{40}\delta_2$$
$$\Delta w_1 = \gamma\delta_1 x_{10}, \quad \Delta w_2 = \gamma\delta_1 x_{20}, \quad \Delta c_1 = \gamma\delta_1 \cdot 1$$

3. Genetic algorithms

a) Suppose a population has six individuals whose fitness values are illustrated in the table as follows

Individual	Fitness
1	6
2	4
3	5
4	10
5	12
6	13

What are the probabilities of selection for these individuals?

Ans. P1=12%, P2=8%, P3=10%, P4=20%, P5=24, P6=26%

b) Next consider the issue of individual selection based on these selection probabilities. For this purpose a uniform random number from [0, 1] is created. Suppose this created number is equal to 0.27, which individual in the population should be selected according to the roulette wheel scheme and why?

Answer: As P1+P2<0.27 and P1+P2+P3>0.27, individual 3 should be selected

c) Why is it required to have mutation in genetic algorithms?

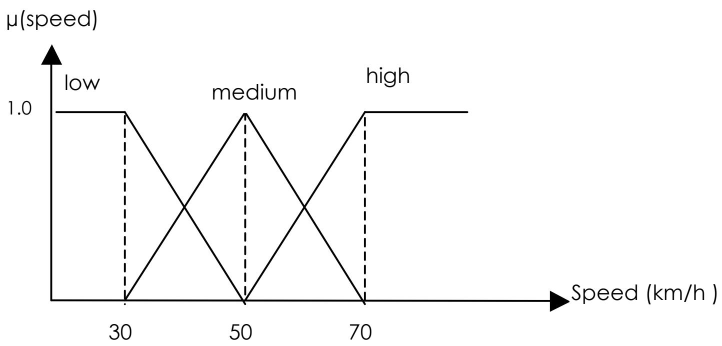
Answer: Mutation can increase the diversity of the population, prevent pre-maturity and avoid local optimum.

4. Fuzzy Rule-Based Reasoning

Consider three fuzzy rules to decide on the proper gears during driving a car in city traffic

R1:	If speed = *low*	Then gear two
R2:	If speed= *medium*	Then gear three
R3:	If speed =*high*	Then gear four

The membership functions for the linguistic terms *low*, *medium*, and *high* are defined as shown in the figure. Suppose the current speed is 54 km/hour.



a) What are the firing strengths of these fuzzy rules under the current speed?
Answer: t1= 0, t2=0.8, t3=0.2

b) What are the output fuzzy sets suggested by the fuzzy rules in the current situation?
Answer: Using MAX or product for fuzzy inference, we get

$$F_1 = \left\{ \frac{0}{Two}, \frac{0}{Three}, \frac{0}{Four} \right\}, \; F_2 = \left\{ \frac{0}{Two}, \frac{0.8}{Three}, \frac{0}{Four} \right\}, \; F_3 = \left\{ \frac{0}{Two}, \frac{0}{Three}, \frac{0.2}{Four} \right\}$$

c) What are the overall output fuzzy set according to the whole fuzzy rule set?
Answer: Using the MAX for aggregation, the overall output fuzzy set is:

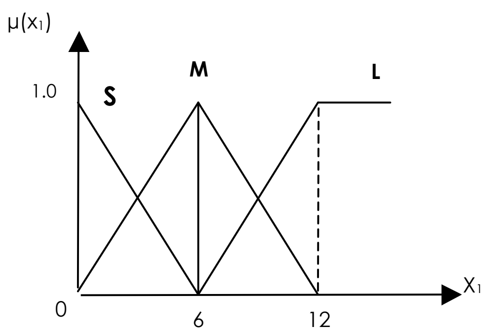
$$F = F_1 \vee F_2 \vee F_3 = \left\{ \frac{0}{Two}, \frac{0.8}{Three}, \frac{0.2}{Four} \right\}$$

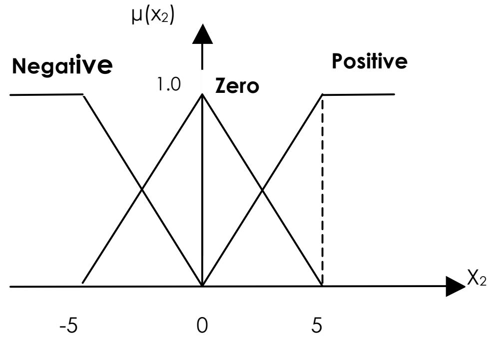
d) What is your final decision and why?
Answer: My final decision is gear three, as it has the highest membership degree in the overall fuzzy set F.

e) How does a fuzzy rule distinguish from a crisp rule? How can you change the above fuzzy rules into crisp rules?
Answer: A fuzzy rule uses fuzzy membership functions, so it can be matched with a partial degree. But a crisp rule does not have a partial matching degree, its condition is either satisfied or not. If we define the meaning of "low", "medium", and "high" as intervals having sharp boundaries, then the rules will become crisp rules.

5. Fuzzy systems learning

Suppose a fuzzy classification system with two inputs x_1 and x_2. The fuzzy subsets *S, M, and L* correspond to input x_1, and fuzzy subsets *Negative*, *Zero*, and *Positiv*e correspond to input x_2. The fuzzy set membership functions of the inputs are depicted in the figures below.





Now suppose there are four training examples as follows:

x_1	x_2	Class
2	-1	A
10	4	B
6	1	C
2	-2	B

Please generate a fuzzy rule set from the above training examples using the Wang-Mendel algorithm.
Answer:
First we create the following four rules from the training examples:

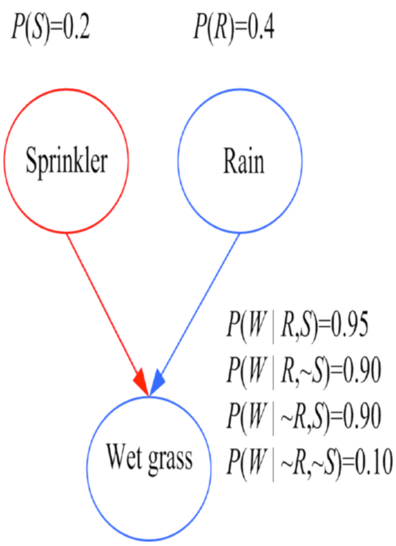
R1: If (x1=S) and (x2=Zero) then class A
R2: If (x1=L) and (x2=Positive) then class B
R3: If (x1=M) and (x2=Zero) then class C
R4: If (x1=S) and (x2=Zero) then class B

Rules R1 and R4 conflict with each other, we compare their degrees of truth:

$$truth(R1) = \frac{2}{3} \times \frac{4}{5} = \frac{8}{15}, \quad truth(R4) = \frac{2}{3} \times \frac{3}{5} = \frac{6}{15}$$

Because truth(R4) is smaller than truth(R1), rule R4 is finally deleted from the rule base.

6. Bayesian network and probabilistic reasoning
   (1) Consider a Bayesian network with 3 nodes as follows, what is the probability for Rain if you see wet grass?



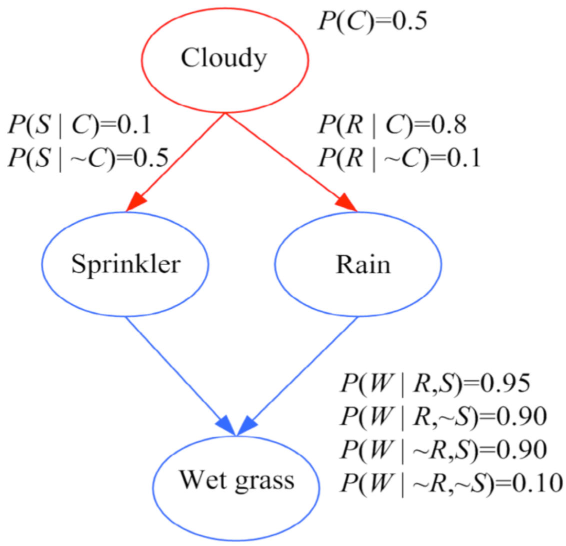
Solution:
P(W|R) = P(W|R,S) P(S|R) +P(W|R,~S) P(~S|R)
= P(W|R,S) P(S) +P(W|R,~S) P(~S)
= 0.95* 0.2 + 0.9 *0.8

P(W|~R) = P(W|S, ~R) P(S| ~R) +P(W|~R, ~S) P(~S| ~R)
= P(W|S, ~R) P(S) +P(W|~R, ~S) P(~S)
= 0.9* 0.2 + 0.1 *0.8

P(R|W)=P(R)P(W|R)/[P(R)P(W|R)+P(~R)P(W|~R)]

= 0.4P(W|R)/[0.4P(W|R)+0.6P(W|~R)]

(2) Consider a Bayesian network with 4 nodes below, how do you calculate the probability of cloudy weather if grass is not wet?



Solution:

$$
\begin{aligned}
P(\sim W|C) = {} & P(\sim W|R,S)\,P(R,S|C) + P(\sim W|\sim R,S)\,P(\sim R,S|C) + P(\sim W|R,\sim S)\,P(R,\sim S|C) + \\
& P(\sim W|\sim R,\sim S)\,P(\sim R,\sim S|C) \\
= {} & P(\sim W|R,S)\,P(R|C)\,P(S|C) + P(\sim W|\sim R,S)\,P(\sim R|C)P(S|C) + P(\sim W|R,\sim S)\,P(R|C)\,P(\sim S|C) + \\
& P(\sim W|\sim R,\sim S)\,P(\sim R|C)\,P(\sim S|C) \\
= {} & 0.05*0.8*0.1+0.1*0.2*0.1+0.1*0.8*0.9+0.9*0.2*0.9
\end{aligned}
$$

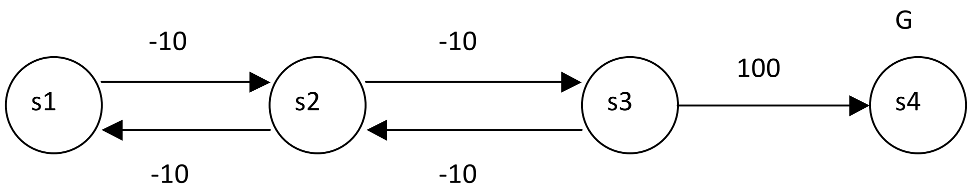
Similarly we calculate P(~W|~C) by considering all situations with Sprinkler and Rain.

Finally

$$
\begin{aligned}
P(C|\sim W) &= P(C)P(\sim W|C) /[\, P(C)P(\sim W|C)+ P(\sim C)P(\sim W|\sim C)] \\
&= 0.5P(\sim W|C) /[\, 0.5P(\sim W|C)+ 0.5P(\sim W|\sim C)]
\end{aligned}
$$

7. Reinforcement Learning in Deterministic Environments

Consider the deterministic environment of four states s1, …., s4 as shown in the figure. The state s4 is an absorbing terminal state. In state s1 there is only one way of move till right state s2, while for states s2 and s3 the agent can choose between two alternative actions (right or left move). The agent obtains a reward of +100 when entering into the terminal state s4 and a penalty of -10 for all other moves. Let the discount factor be 0.8.



a) Suppose a policy π of actions as follows:
   π(S1)= →, π(S2)=←, π(S3)=→,

What are the values of the states under this policy? For states S1 and S2, you only need to write formulas for calculation rather than exact results.

Answer:

$V^{\pi}(s3) = 100$

$V^{\pi}(s2) = -10 + 0.8 \cdot (-10) + 0.8^2 \cdot (-10) + \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$

$V^{\pi}(s1) = -10 + 0.8 \cdot (-10) + 0.8^2 \cdot (-10) + \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$

b) What are the optimal values of the states s1, s2 and s3? How do you get the optimal actions at states from these optimal state values?

Answer:

$V^*(s3) = 100$

$V^*(s2) = -10 + 0.8 \cdot 100 = 70$

$V^*(s1) = -10 + 0.8 \cdot (-10) + 0.8^2 \cdot 100 = 46$

$Q^*(s3, \rightarrow) = 100$

$Q^*(s3, \leftarrow) = -10 + 0.8 V^*(s2) = -10 + 0.8 \times 70 = 46$

As $Q^*(s3, \rightarrow) > Q^*(s3, \leftarrow)$, the optimal action at state s3 is $\rightarrow$

$Q^*(s2, \rightarrow) = -10 + 0.8 V^*(s3) = -10 + 0.8 \times 100 = 70$

$Q^*(s2, \leftarrow) = -10 + 0.8 V^*(s1) = -10 + 0.8 \times 46 = 26.8$

As $Q^*(s2, \rightarrow) > Q^*(s2, \leftarrow)$, the optimal action at state s2 is $\rightarrow$

c) Suppose that the previous estimates for Q* values are zero for all state-action pairs, how can you update some estimates of the Q* values using the recorded sequence s1 → s2 → s3 → s4?

Answer:

$Q(s3, \rightarrow) = 100$

$Q(s2, \rightarrow) = -10 + 0.8 \max_{a'} Q(s3, a') = -10 + 0.8 \cdot 100 = 70$

$Q(s1, \rightarrow) = -10 + 0.8 \max_{a'} Q(s2, a') = -10 + 0.8 \cdot 70 = 46$

8. Reinforcement Learning in Stochastic Environments

Write out the Q-learning rule in a stochastic environment. What is its difference with the Q-learning rule for a deterministic environment? Please also justify the Q-learning rule you have presented.

Reference answer:

Suppose action a_t is performed at state s_t leading to a new state s_{t+1} and new reward r_{t+1}, the Q-learning rule for a stochastic environment is as follows:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)\right]$$

where Q is the estimate of Q* values, and $\alpha = \frac{1}{k}$ (k is the total number of times that the pair (s_t, a_t) has been visited so far, including this latest visit used for learning).

Here we treat next state s_{t+1} and new reward r_{t+1} as samples in the sense that outcomes can be different by taking same action in a given state. But, in deterministic environment, the next state and reward by taking one action are fixed outcomes.

Back to stochastic environment, if we observe next state s_{t+1} and new reward r_{t+1}, we consider $r_{t+1} + \gamma \max_{a'} Q(s_{t+1}, a')$ as a new sample for estimating Q*. And we use this new sample to update the current estimate of Q* via an incremental averaging method, which leads to the Q-learning rule in stochastic environments.