Exam in Learning Systems (DVA427)

Date	2018-03-22
Time	8:10 – 11:30
Allowed material	None
Examiner	Ning Xiong (Phone 151716)

Good Luck!!!

1. Decision Tree Learning (1+4+2p)

In the table below you see a set of training examples for deciding whether a type of food is appealing based on temperature, taste and size

Appealing	Temperature	Taste	Size
No	Hot	Salty	Small
No	Cold	Sweet	Large
No	Cold	Sweet	Large
Yes	Cold	Sour	Small
Yes	H	Sour	Small
No	H	Salty	Large
Yes	H	Sour	Large
Yes	Cold	Sweet	Small
Yes	Cold	Sweet	Small
No	H	Salty	Large

a) What is the Entropy of this set of training examples?
b) What are the information gains of Temperature, Taste and Size, given this training set?
c) What is the root node for the decision tree and why?

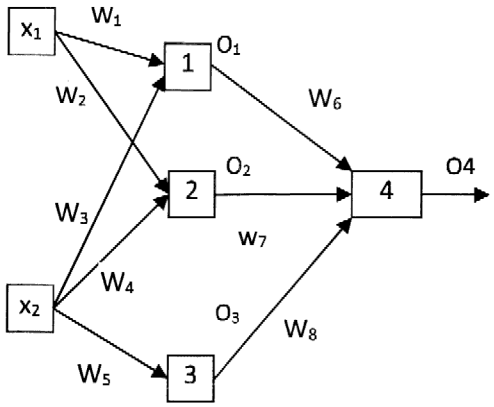
Use the following table to look up the entropy for a subset of training instances. Take the entry that is closest to the decimal number for which you want to calculate the entropy (e.g. if you need to calculate the entropy of 0.33, take the entry for 0.3 in the table which is 0.9).

p	entropy(p)
0.0	0.0
0.1	0.5
0.2	0.7
0.25	0.8
0.3	0.9
0.4	1.0
0.5	1.0
0.6	1.0
0.7	0.9
0.75	0.8
0.8	0.7
0.9	0.5
1.0	0.0

2. Artificial Neural Network (6p)

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

$$O_1 = sigmoid(w_1 x_1 + w_3 x_2 + c_1)$$
$$O_2 = sigmoid(w_2 x_1 + w_4 x_2 + c_2)$$
$$O_3 = sigmoid(w_5 x_2 + c_3)$$
$$O_4 = sigmoid(w_6 O_1 + w_7 O_2 + w_8 O_3 + c_4)$$



Now you are given a training example (x_{10}, x_{20}, t_0). Suppose that, under this training example, the outputs of the four units are O_{10}, O_{20}, O_{30} and O_{40} respectively, and the current values for weights of W_6, W_7 and W_8 are W_{60}, W_{70} and W_{80} respectively. Assume the learning rate is defined as γ. Your task is to update the weights and thresholds for the neuron units in light of this training example. Please write out the formulas to calculate ΔW_i and Δc_i in terms of the incremental BP algorithm.

3. Genetic algorithms (1+1+2+2+2p)

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

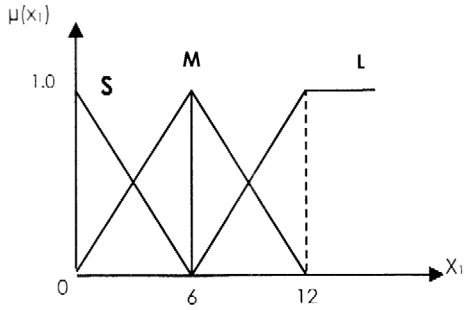
Individual	Fitness
1	12
2	5
3	6
4	8
5	5
6	14

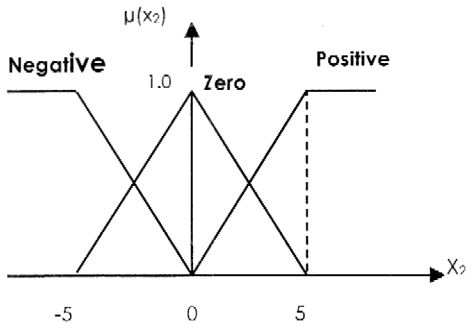
What are the probabilities of selection for these individuals based on their fitness?

b) Now your task is to select individuals based on these selection probabilities. For this purpose a uniform random number from [0, 1] is created. Suppose this created number is equal to 0.3,

which individual in the population should be selected according to the roulette wheel scheme and why?
c) What is the role of mutation in genetic algorithms?
d) How to do mutation on a real-valued string?
e) What do you think would be a weakness of the roulette wheel scheme and what alternative method can be helpful to enhance the population diversity compared to roulette wheel?

4. Fuzzy systems and learning (4+2+2+1p)
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 *Positive* correspond to input x_2. The fuzzy set membership functions of the inputs are depicted in the figures below.



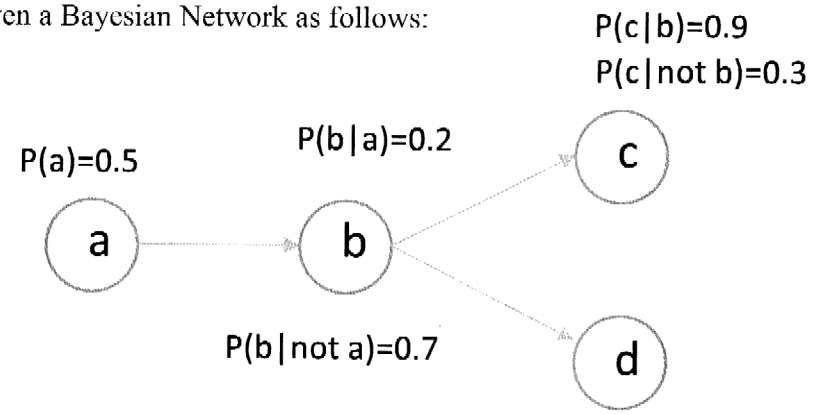


Now suppose there are four training examples as follows:

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

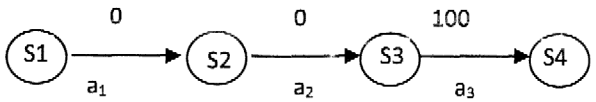
a) Please generate a fuzzy rule set from the above training examples using the Wang-Mendel algorithm.
b) What are firing strengths of the generated fuzzy rules given the input situation x_1=5, x_2=2?
c) What are the output fuzzy sets suggested by the generated fuzzy rules in the current situation with x_1=5, x_2=2?
d) How can you make final decision in terms of these fuzzy rules in the current situation with x_1=5, x_2=2?

5. Bayesian Networks (3+1p)
Given a Bayesian Network as follows:



a) What are the probabilities of c when a is true and when a is not true?
b) How do you calculate the probability of a when c is known to be true?

6. Reinforcement Learning without model (3+3p)
Given a sequence of the agent's interaction with the environment as shown in the figure as follows, where rewards are indicated above the arrows and actions are indicated below the arrows.



Let Q be the estimate of the best action values Q*. Assume the current Q estimates are given as follows:

$Q(S1,a_1)=7.0$, $Q(S1, a)=0$ for all other actions different from a_1
$Q(S2,a_2)=5.0$, $Q(S2, a)=0$ for all other actions different from a_2
$Q(S3,a_3)=3.0$, $Q(S3, a)=0$ for all other actions different from a_3
$Q(S4, a)=0$ for any action a

So far the numbers of visits for the state-actions pairs $(S1,a_1)$, $(S2,a_2)$, and $(S3,a_3)$ are N_1, N_2, and N_3 respectively. The discounting factor is 0.8.

1) How can you update the Q estimates using the Q-learning rule if the environment is deterministic?
2) How can you update the Q estimates using the Q-learning rule if the environment is stochastic?