CPSC 633-600 (Total 100 points)
Decision Tree Learning and Reinforcement Learning

See course web page for the **due date**.
Use **csnet** to submit your assignments.

Instructor: Yoonsuck Choe

February 18, 2009

1 Entropy

Given a random variable X that can take on values $\{\oplus, \ominus\}$, the entropy is defined as:

$$E(X) = -\sum_{x \in \{\oplus,\ominus\}} P(X = x) \log_2 P(X = x).$$

Since $P(X = \oplus) + P(X = \ominus) = 1$, $E(X)$ can be rewritten as a function of $P(X = \oplus)$: Letting $p_\oplus = P(X = \oplus)$:

$$E(X) = f(p_\oplus) = -p_\oplus \log_2 p_\oplus - (1 - p_\oplus) \log_2 (1 - p_\oplus).$$

Figure 1 shows how $f(p_\oplus)$ behaves as $p_\oplus$ changes.

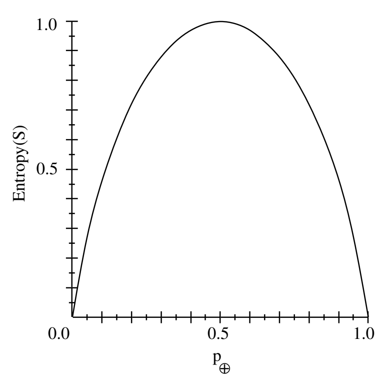

Figure 1: **Entropy.**

Problem 1 (Written: 5 pts): Extend the above analysis to a random variable Y that can take on values $\{\alpha, \beta, \gamma\}$. Given $p_\alpha = P(Y = \alpha)$, etc.,

1. Derive $E(Y)$ as a function of p_α and p_β:

$$E(Y) = f(p_\alpha, p_\beta) = ...$$

Note: $p_\alpha + p_\beta + p_\gamma = 1.0$.

2. For which values of p_α and p_β does $E(Y)$ become maximal (no need to derive it exactly from $f(p_\alpha, p_\beta)$–consider when it is maximal in the 2-value case)?

3. Explain why.

Problem 2 (Program: 15 pts): Write a short program to calculate $f(p_\alpha, p_\beta)$ derived above, and obtain the $E(Y) = f(p_\alpha, p_\beta)$ values for all combinations of $p_\alpha, p_\beta \in \{0.0, 0.1, 0.2, ..., 0.9, 1.0\}$, and plot in 3D (Octave: use `gsplot`; Matlab: use `surf`; or draw by hand).

2 Decision Tree Learning (ID3)

Problem 3 (Written: 15 pts): You are trying to decide whether you want to submit your paper to a particular conference or not, based on three criterions: location of the conference, prestige of the conference, and chance of getting your paper accepted.

(1) Calculate the entropy of the following training set. (2) Calculate the information gain for each of the three attributes, $Location$, $Prestige$, and $Chance$. (3) Which one is the best attribute to test first?

Example#	Location	Prestige	Chance	Submit?
1	Near	High	High	Y
2	Far	High	Medium	Y
3	Near	Okay	High	Y
4	Far	High	High	N
5	Near	Okay	Low	N
6	Far	High	High	Y
7	Far	Okay	High	N
8	Far	Okay	Medium	N
9	Near	High	High	Y
10	Far	High	Medium	Y

3 Reinforcement Learning: Deterministic Case

Problem 4 (Written: 20 pts): Solve exercise 13.2 in the textbook (p. 388). (Note the typo in the textbook: $V^*(s, a)$ should be $V^*(s)$.)

4 Reinforcement Learning: Nondeterministic Case

Suppose we modified the grid world in exercise 13.2 so that the reward function $r(s, a)$ and the state transition function $\delta(s, a)$ are probabilistic:

- For the actions leading to the goal (those that currently have a reward of 10), the reward has the following probability distribution:

Reward r	P(r)
6	0.2
8	0.3
10	0.5

  and for the rest all rewards are zero.

- The state transition $\delta(s, a)$ occurs with a probability of 70% for the intended direction, and 30% for the unintended direction if there is only one alternative direction, and 15% each for each unintended direction if there are two alternative directions.

Problem 5 (Program: 20 pts): Write a program to learn the Q values for the task above, according to the description in 13.4 (p. 381–382). Use a random policy.

Report (1) the $Q(s, a)$ values, and also (2) the expected value of the reward $r(s, a)$

$$E[r(s,a)] = \frac{\sum_t immediate\_reward_t(s,a)}{visits(s,a)}.$$

Problem 6 (Written: 15 pts): (1) Using the learned $Q(s, a)$ values, calculate the $V^*(s)$ for each state (note that $V^*(s) = \max_{a'} Q(s, a')$). (2) Verify that equation 13.8 holds, using the $V^*(s)$ just calculated, and $E[r(s, a)]$ calculated from the simulation. (3) Analytically derive $E[r(s, a)]$, and compare with the results from (2).

Problem 7 (Written: 10 pts): For easier reference, let us name the states as follows:

s_1	s_2	s_3
s_4	G	s_5

Given the following assumptions (which are reasonable), derive the $V^*(s)$ and $Q(s, a)$ values by hand, using equation 13.8 (and the fact that $V^*(s) = \max_{a'} Q(s, a')$):

- $V^*(s_1) = V^*(s_3)$ and $V^*(s_4) = V^*(s_5)$.
- For s_2, s_4, and s_5, $\mathrm{argmax}_{a'} Q(s, a')$ is the direction going into the goal G.