Homework, $n-$armed bandit

Bandit Questions

1. Is the $n-$armed bandit problem an example of supervised or unsupervised learning?

2. The goal of the $n-$armed bandit problem is to devise a strategy to play the $n-$arms so that we maximize our returns. To start the algorithm, suppose we begin by getting estimates of the returns of the n slots.

   (a) What is the greedy algorithm, and why would it not necessarily produce the maximum return?

   (b) What is the ϵ greedy algorithm?

3. The "softmax" function is designed to take a vector of returns (the current estimates of the expected returns of the n slots), and return probabilities so that machine i is played with probability p_i.

   (a) What restrictions are placed on a set of numbers before we can say that they represent a set of probabilities for some action (suppose the numbers are $p_1, p_2, \ldots, p_n$).

   (b) Why do we apply the exponential function to a set of returns?

4. To finish converting the set of returns (Q) to probabilities (P), we also introduce a "temperature" parameter τ. The final version of the probability to play the k^{th} slot machine at time t is given by:

   $$P_t(k) = \frac{\exp\left(\frac{Q_t(k)}{\tau}\right)}{\sum_{j=1}^{n} \exp\left(\frac{Q_t(j)}{\tau}\right)}$$

   Now suppose we have two probabilities, $P(1)$ and $P(2)$ (we left off the time index since it won't matter in this problem). Furthermore, suppose $P(1) > P(2)$. Compute the limits of $P(1)$ and $P(2)$ as τ goes to zero. Compute the limits as τ goes to infinity (Hint on this part: Use the definition, and divide numerator and denominator by $\exp(Q(1)/\tau)$ before taking the limit).

   (NOTE: This is the same exercise as on pg 10-11 of the notes.

5. Work through the 5 questions at the bottom of pg 12 (Pursuit Strategy, or "Win-Stay, Lose-Shift").

Matlab/Octave Questions

1. Suppose A and B are two vectors of the same size. What is the difference between: A/B and $A./B$? For help, you might look at the help file for `mrdivide` (short for matrix right divide).

2. Give the result of the computations on lines 2-4 of the code below (the line numbers are not part of the actual commands, there are there only for reference). You should be able to do this without typing in the code to Matlab/Octave, but you can check your work that way.

```
1  x=[1 2 3 0 3];
2  max(x)
3  x==max(x)
4  find(x==max(x))
```

3. Given a vector of real numbers stored in variable Q, and positive scalar t, what is the following set of commands doing?

```
P=exp(Q/t)
P=P./sum(P)
```

4. Run the two scripts in Matlab/Octave (we ran one of them in class).

   For Octave Online, you'll need an account on Octave, then drag/drop the four $m-$files (you'll need to download them from our class website). The two main files are `ScriptSoftMax.m` and `ScriptPursuit.m`, each uses the function files `softmax.m` and `winstay.m`, respectively.

   In Octave Online (or Matlab), you should only need to type in the file name of the script you want to run (for example, `ScriptSoftMax.m`).

   Report on the results.