COMPSCI 570 **Instructor:** Ronald Parr

Homework 4 **Due:** 4/23/25

Note: You may **not** search the internet, use ChatGPT/ChatGPT-like systems, or use any other (generative)AI/foundation/large-language models to help answer these questions. You may use the internet to look up basic definitions and methods.

You may have high level discussions with your classmates about understanding the underlying concepts and about how to approach these problems in general, but may not share anything resembling a complete solution. What you write up and turn in should be your own.

Note that, unlike previous assignments we will not accept submissions that are more than one day late. This is to ensure that we are able to provide grades and distribute solutions in a timely manner.

1 HMMs (30 points)

Consider a hidden Markov model with two states, x and y, two observations a and b, and the following somewhat unusual parameters. We use the random variable S_t for the state at time t:

- $P(S_{t+1} = x|S_t = x) = P(S_{t+1} = y|S_t = y) = 1$
- $P(a|S_t = x) = P(b|S_t = x) = 0.5$
- $P(b|S_t = y) = 1.0$

These parameters were chosen to help you realize some things about HMMs as you work through the basic algorithms. Assume that at time 0, the distribution over states is: $P(S_0 = x) = 0.6$, and that there are no observations at time 0. At time 1, the observation is b. At time 2, the observation is a.

a) Show the work to run the forward algorithm (AKA monitoring or tracking) and compute the distribution over states at time steps 1 and 2, i.e., $P(S_1|O_1 = b)$ and $P(S_2|O_1 = b, O_2 = a)$.

b) Show the work to compute the smoothed distribution over states at time step 1, i.e., $P(S_1|O_1 = b, O_2 = a)$, where O_t is the observation at time t. (Hint: You should see a big difference between the forward probabilities at time 1 and the smoothed probabilities.)

c) Show the work to compute the Viterbi path given the same initial distribution at time 0, and same observatins, $O_1 = b$, then extend your calculation to include $O_2 = a$. Note that final your answers should be a sequence of 2 states up to $O_1 = b$, and then a sequence of 3 states when you extend your answer. You should observe something interesting about how the optimal path changes when you get the observation at time 2.

2 Linear Regression (20 points)

Consider a linear regression problem with feature set $\Phi = \phi_1 \dots \phi_k$ and regression target t, and let w^* be the least squares solution, i.e., the minimizer of:[1]

$$E = \|\Phi w^* - t\|^2 = \sum_{i=1}^{n} (t^{(i)} - \sum_{j=1}^{k} \phi_j(x^{(i)}) w_j^*)^2$$

Now, consider $\Phi' = \phi'_1 \dots \phi'_k$, with $\phi'_i = a * \phi_i$ for all i, where a is a positive constant. Let $w^{*'} = w^*/a$.

1. Prove that $E' = \|\Phi' w^{*'} - t\| = E$. (10 points)
2. Prove that $w^{*'}$ is the minimizer of E'. (10 points)

3 Neural Weight Design (10 points)

Consider the XOR problem illustrated in the graph below, where points $(0, 1)$ and $(1, 0)$ are treated as negative and points $(0, 0)$ and $(1, 1)$ are treated as positive.

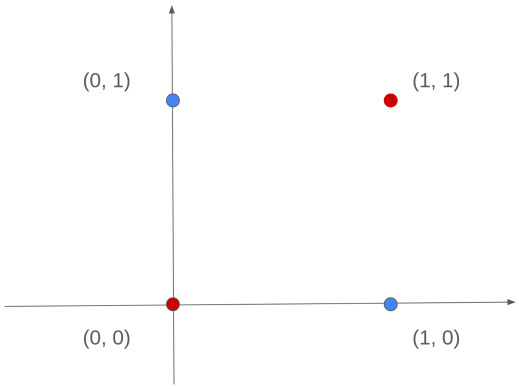

Figure 1: Illustration for problem 1

Construct a two-layer network of perceptrons (step activation function, 3 neurons total) that can classify the points perfectly. Assume that your step function fires if the weighted sum of inputs is greater than or equal to 0. The inputs to your network should be x_1 and x_2. If you wish, you can assume that each node has an additional input which is always 1.

[1] We accidentally left the square off of $\|\Phi w^* - t\|^2$ in an earlier version of this PDF. It doesn't matter if work with the sum of squared errors or the square root of this quantity, so long as you are consistent.