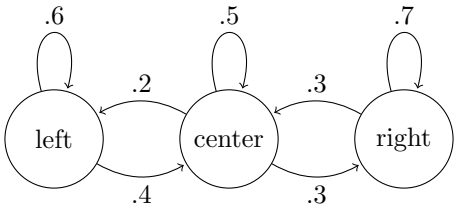
Intro to A.I.
Final Exam

Ozaner Hansha

May 10, 2020

Question 1

Part a: Below is the transition diagram of the Markov chain:



Part b: Note that the Markov chain's transition matrix M is given by:

$$M = \begin{matrix} \\ l \\ c \\ r \end{matrix} \begin{matrix} \begin{matrix} l & c & r \end{matrix} \\ \begin{bmatrix} .6 & .4 & 0 \\ .2 & .5 & .3 \\ 0 & .3 & .7 \end{bmatrix} \end{matrix}$$

And our initial state is given by $x_0 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$. With this information, the (transposed) state at time step 3 $x_3^\top$ is given by:

$$\begin{aligned} x_3^\top &= x_0^\top M^3 \\ &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} .6 & .4 & 0 \\ .2 & .5 & .3 \\ 0 & .3 & .7 \end{bmatrix}^3 \\ &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} .352 & .432 & .216 \\ .216 & .406 & .378 \\ .108 & .378 & .514 \end{bmatrix} \\ &= \begin{bmatrix} .108 & .378 & .514 \end{bmatrix} \end{aligned}$$

And so, at step 3, the probability that the car is on the left lane is 10.8%, on the center lane is 37.8%, and on the right lane is 51.4%.

Part c: Note that we assume that the car was in the right lane in step 0, as in part b. Also note that for the Markov chain $(X_i)_{i \in I}$ we denote the event $X_i = s$ as s_i where $s \in \{r, c, l\}$. Now, before we calculate the conditional distribution of X_3 given r_0 and c_4, we must first compute the following:

$$\begin{aligned} P(c_4 | r_0) &= (x_0^\top M^4)_2 && \text{(distr. of Markov chain)} \\ &= (x_3^\top M)_2 && (x_i^\top := x_0^\top M^i) \\ &= \left(\begin{bmatrix} .108 & .378 & .514 \end{bmatrix} \begin{bmatrix} .6 & .4 & 0 \\ .2 & .5 & .3 \\ 0 & .3 & .7 \end{bmatrix} \right)_2 \\ &= \left(\begin{bmatrix} .1404 & .3864 & .4732 \end{bmatrix} \right)_2 \\ &= .3864 \end{aligned}$$

We can now compute our desired probabilities. For each lane $s \in \{l, c, r\}$ we have:

$$\begin{aligned}
P(s_3|r_0c_4) &= \frac{P(c_4s_3r_0)}{P(r_0c_4)} && \text{(def. of conditional prob.)} \\
&= \frac{P(c_4|s_3r_0)P(s_3|r_0)P(r_0)}{P(c_4|r_0)P(r_0)} && \text{(chain rule)} \\
&= \frac{P(c_4|s_3)P(s_3|r_0)P(r_0)}{P(c_4|r_0)P(r_0)} && \text{(Markov property)} \\
&= \frac{P(c_4|s_3)P(s_3|r_0)\cdot 1}{P(c_4|r_0)\cdot 1} && \text{(initial distribution)} \\
&= \frac{P(c_4|s_3)P(s_3|r_0)}{.3864} && \text{(calculated above)}
\end{aligned}$$

Now we plug in the corresponding probabilities for each $s \in \{l, c, r\}$:

$$\begin{aligned}
P(l_3|r_0c_4) &= \frac{P(c_4|l_3)P(l_3|r_0)}{.3864} \\
&= \frac{P(c_4|l_3)(.108)}{.3864} && \text{(part a)} \\
&= \frac{M_{12}(.108)}{.3864} && \text{(def. of transition prob.)} \\
&= \frac{18}{161} \approx .1118 \\
P(c_3|r_0c_4) &= \frac{P(c_4|c_3)P(c_3|r_0)}{.3864} \\
&= \frac{P(c_4|l_3)(.378)}{.3864} && \text{(part a)} \\
&= \frac{M_{22}(.378)}{.3864} && \text{(def. of transition prob.)} \\
&= \frac{45}{92} \approx .4891 \\
P(r_3|r_0c_4) &= \frac{P(c_4|r_3)P(r_3|r_0)}{.3864} \\
&= \frac{P(c_4|l_3)(.514)}{.3864} && \text{(part a)} \\
&= \frac{M_{32}(.514)}{.3864} && \text{(def. of transition prob.)} \\
&= \frac{257}{644} \approx .3991
\end{aligned}$$

Given it was on the center lane at step 4, the probability that at step 3 the car is on the left lane is 11.18%, on the center lane is 48.91%, and on the right lane is 39.91%.

Question 2

Part a: The expected reward given b (i.e. that James buys the textbook) is given by:

$$\begin{aligned}
E[R|b] &= \overbrace{2300}^{\text{pass}-\text{book}} P(p|b) && \text{(def. of conditional expectation)} \\
&= 2300(P(p|m,b)P(m|b) + P(p|\neg m,b)P(\neg m|b)) && \text{(chain rule \& total probability)} \\
&= 2300((.95)(.9) + (.6)(.1)) \\
&= 2104.5
\end{aligned}$$

The expected reward given $\neg b$ (i.e. that James doesn't buy the textbook) is given by:

$$\begin{aligned}
E[R|\neg b] &= \overbrace{2500}^{\text{pass}} P(p|\neg b) && \text{(def. of conditional expectation)} \\
&= 2500(P(p|m,\neg b)P(m|\neg b) + P(p|\neg m,\neg b)P(\neg m|\neg b)) && \text{(chain rule \& total probability)} \\
&= 2500((.8)(.7) + (.3)(.3)) \\
&= 1625
\end{aligned}$$

Since $E[R|b] > E[R|\neg b]$, James should buy the textbook as that will maximize his net reward.

Part b: Note that to make this situation more exact we denote having the book at the beginning b_1 and at the end b_2. Clearly $P(b_2|b_1) = 1$.

First we compute the expected reward given that James defers buying the book and then buys it (i.e. $\neg b_1, b_2$) is given by:

$$\begin{aligned}
E[R|\neg b_1, b_2] &= \overbrace{2300}^{\text{pass}-\text{book}} P(p|\neg b_1, b_2) && \text{(def. of conditional expectation)} \\
&= 2300(P(p|m,b_2)P(m|\neg b_1) + P(p|\neg m,b_2)P(\neg m|\neg b_1)) && \text{(chain rule \& total probability)} \\
&= 2300((.95)(.7) + (.6)(.3)) \\
&= 1943.5
\end{aligned}$$

Now we recall that the expected reward given that James defers buying the book and then doesn't buy it (i.e. $\neg b_1, \neg b_2$) is the same as in part a (i.e. $\neg b$):

$$E[R|\neg b_1, \neg b_2] = E[R|\neg b] = 1625$$

Likewise, the expected reward given that James doesn't defer buying the book (i.e. b_1) is also the same (i.e. b):

$$E[R|b_1] = E[R|b_1, b_2] = E[R|b] = 2104.5$$

We can now express the expected reward given James defers the decision (i.e. $\neg b_1$) like so:

$$\begin{aligned}
E[R|\neg b_1] &= E[R|\neg b_1, b_2]p(b_2|\neg b_1) + E[R|\neg b_1, \neg b_2]p(\neg b_2|\neg b_1) \\
&= 1943.5P(b_2|\neg b_1) + 1625P(\neg b_2|\neg b_1)
\end{aligned}$$

You'll note, however, that we cannot determine the probability of $P(b_2|\neg b_1)$ or $P(\neg b_2|\neg b_1)$. This is not a problem though, as we can still bound the expected reward:

$$\begin{aligned}
E[R|\neg b_1] &\leq \max(E[R|\neg b_1, b_2], E[R|\neg b_1, \neg b_2]) \\
&= E[R|\neg b_1, b_2] \\
&= 1943.5
\end{aligned}$$

As a result, we know that the expected reward of deferring to buy the book is less than the expected reward of buying it right away:

$$E[R|\neg b_1] \leq 1943.5 \leq E[R|b_1] = 2104.5$$

Now, to compute our policy, we simply have to determine whether buying or not buying the book before the test (i.e. b_2) maximizes our reward in the case that James has mastered/hasn't mastered the material:

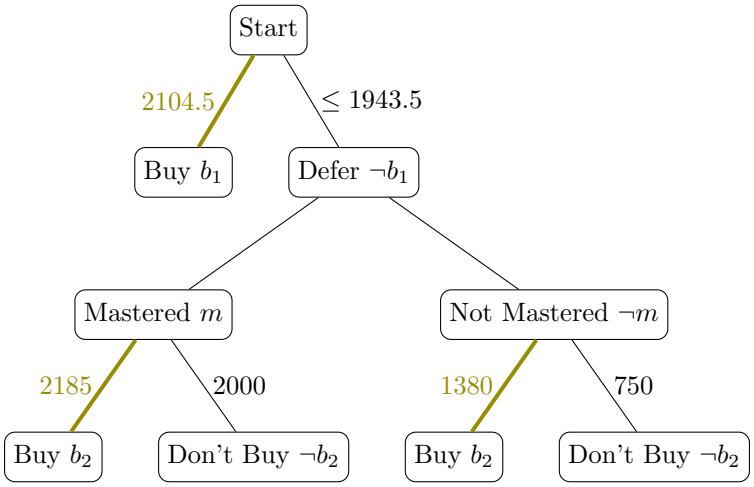
$$E[R|m, \neg b_1, b_2] = \overbrace{2300}^{\text{pass}-\text{book}} P(p|m, b_2) = 2185$$

$$E[R|m, \neg b_1, \neg b_2] = \overbrace{2500}^{\text{pass}} P(p|m, \neg b_2) = 2000$$

$$E[R|\neg m, \neg b_1, b_2] = \overbrace{2300}^{\text{pass}-\text{book}} P(p|\neg m, b_2) = 1380$$

$$E[R|\neg m, \neg b_1, \neg b_2] = \overbrace{2500}^{\text{pass}} P(p|\neg m, \neg b_2) = 750$$

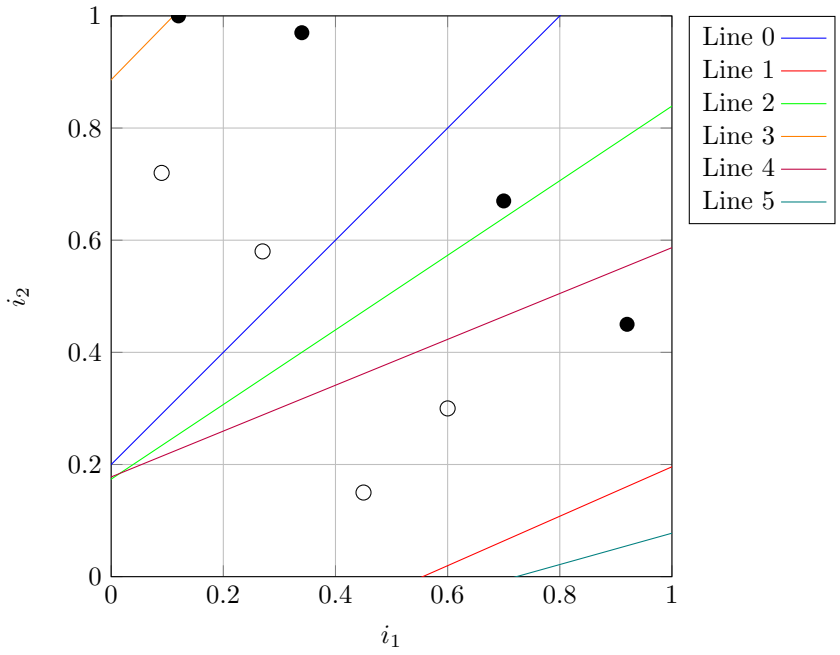
It is now clear what our policy should be:



We highlight the optimal choice in the decision tree above with olive branches, with the edge weights denoting the expected reward given that choice. You'll note that, even though the choice to buy or not buy the book before the test should never be encountered in an optimal setting (since the optimal first action is to simply buy the book at the beginning), we include it as the definition of a policy is a mapping of *all* states to actions.

Question 3

Part a:



Step	Misclassified
0	4
1	4
2	3
3	4
4	3
5	4

We use a learning rate of $\alpha = .5$. Note that the boundary is given by $y = mx + b$ where:

$$m = -\frac{w_1}{w_2}$$
$$b = -\frac{w_0}{w_2}$$

The calculations for the weights at each step are given below:

$$\mathbf{w}^{(0)} = \begin{bmatrix} .2 \\ 1 \\ -1 \end{bmatrix}$$

$$\mathbf{w}^{(1)} = \begin{bmatrix} .2 \\ 1 \\ -1 \end{bmatrix} + .5(0-1)\begin{bmatrix} 1 \\ .92 \\ .45 \end{bmatrix} = \begin{bmatrix} -.3 \\ .54 \\ -1.225 \end{bmatrix}$$

$$\mathbf{w}^{(2)} = \begin{bmatrix} -.3 \\ .54 \\ -1.225 \end{bmatrix} + .5(1-0)\begin{bmatrix} 1 \\ .45 \\ .15 \end{bmatrix} = \begin{bmatrix} .2 \\ .765 \\ -1.15 \end{bmatrix}$$

$$\mathbf{w}^{(3)} = \begin{bmatrix} .2 \\ .765 \\ -1.15 \end{bmatrix} + .5(1-0)\begin{bmatrix} 1 \\ .09 \\ .72 \end{bmatrix} = \begin{bmatrix} .7 \\ .81 \\ -.79 \end{bmatrix}$$

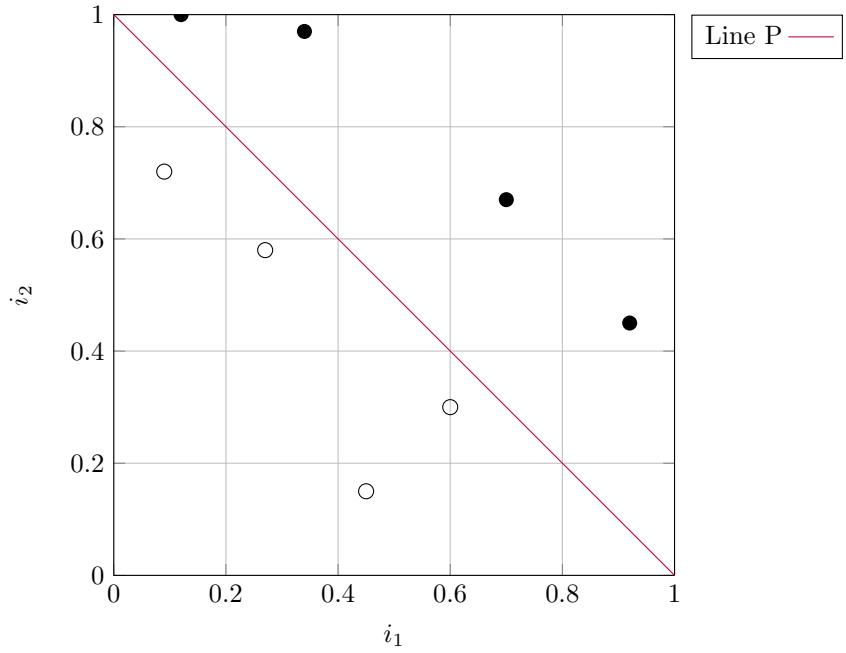
$$\mathbf{w}^{(4)} = \begin{bmatrix} .7 \\ .81 \\ -.79 \end{bmatrix} + .5(0-1)\begin{bmatrix} 1 \\ .7 \\ .67 \end{bmatrix} = \begin{bmatrix} .2 \\ .46 \\ -1.125 \end{bmatrix}$$

$$\mathbf{w}^{(5)} = \begin{bmatrix} .2 \\ .46 \\ -1.125 \end{bmatrix} + .5(0-1)\begin{bmatrix} 1 \\ .09 \\ .72 \end{bmatrix} = \begin{bmatrix} -.3 \\ .415 \\ -1.485 \end{bmatrix}$$

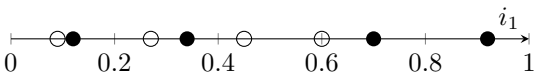
Part b: Our perceptron did not achieve perfect classification in 5 iterations. One set of weights $\mathbf{w}^{(p)}$ that would perfectly classify the data is given below:

$$\mathbf{w}^{(p)} = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} \implies \begin{cases} m = -1 \\ b = 1 \end{cases}$$

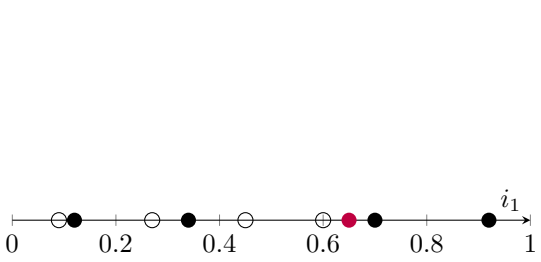
Graphing the corresponding decision boundary we have:



Part c: Such a modified problem is equivalent to projecting the dataset onto the i_1 axis like so:



The boundary problem is now to find the point on the axis that separates the black and white data the most. One such boundary is at $i_1 = .65$, where the left of this point represents white samples and the right represents black ones. Only 2 samples are misclassified by this:



Question 4

Part a: A sinusodial regression would try to fit the weights w_0, w_1, w_2, w_3 on the following function:

$$f_{\mathbf{w}}(x) = w_0 \sin(w_1 x + w_2) + w_3$$

Fitting these 4 variables corresponds to fitting the sine wave's amplitude (w_0), frequency (w_1), horizontal translation (w_2), and vertical translation (w_3).

Part b:

$$\nabla \operatorname{Cost}_{\mathbf{w}}(D) = \begin{bmatrix} \frac{\partial \operatorname{Cost}_{\mathbf{w}}(D)}{\partial w_0} \\ \frac{\partial \operatorname{Cost}_{\mathbf{w}}(D)}{\partial w_1} \\ \frac{\partial \operatorname{Cost}_{\mathbf{w}}(D)}{\partial w_2} \\ \frac{\partial \operatorname{Cost}_{\mathbf{w}}(D)}{\partial w_3} \end{bmatrix} \qquad \text{(def. of gradient)}$$

$$= \begin{bmatrix} \frac{\partial (y - f_{\mathbf{w}}(x))^2}{\partial w_0} \\ \frac{\partial \sum_{(x,y)\in D} (y - f_{\mathbf{w}}(x))^2}{\partial w_1} \\ \frac{\partial \sum_{(x,y)\in D} (y - f_{\mathbf{w}}(x))^2}{\partial w_2} \\ \frac{\partial \sum_{(x,y)\in D} (y - f_{\mathbf{w}}(x))^2}{\partial w_3} \end{bmatrix} \qquad \text{(def. of Cost)}$$

$$= \sum_{(x,y)\in D} \begin{bmatrix} \frac{\partial (y - f_{\mathbf{w}}(x))^2}{\partial w_0} \\ \frac{\partial (y - f_{\mathbf{w}}(x))^2}{\partial w_1} \\ \frac{\partial (y - f_{\mathbf{w}}(x))^2}{\partial w_2} \\ \frac{\partial (y - f_{\mathbf{w}}(x))^2}{\partial w_3} \end{bmatrix} \qquad \text{(linearity)}$$

$$= \sum_{(x,y)\in D} \begin{bmatrix} 2(w_0 \sin(w_1 x + w_2) - y + w_3) \sin(w_1 x + w_2) \\ 2(w_0 \sin(w_1 x + w_2) - y + w_3) \cos(w_1 x + w_2) w_0 x \\ 2(w_0 \sin(w_1 x + w_2) - y + w_3) \cos(w_1 x + w_2) w_0 \\ 2(w_0 \sin(w_1 x + w_2) - y + w_3) \end{bmatrix} \qquad \text{(calculate partial derivatives)}$$

Extra Credit: The amount of daylight any particular location on earth receives changes over time. By recording the amount of daylight for a few different days, we can fit a sinusodial curve and extrapolate farther into the future/past.

Another example is with modeling circular motion. The position of a cart on a ferris wheel, for example, can be modeled with a sinusoidal curve due to its recurrent motion and constant frequency (rotational speed).

Question 5

Below we give the input function S and activation function f for an n-input neuron that implements an n-input xor gate:

$$S(x_1, \cdots, x_n) = \sum_{1 \le i \le n} x_i$$

$$f(x) = \begin{cases} 1, & x = 1 \\ 0, & \text{otherwise} \end{cases}$$

Note that we set all weights w_i of this neuron to 1.

Question 6

Part a: First recall that the complexity of matrix-vector multiplication with a matrix of size $m \times m$ and a vector of size $m \times 1$ is $O(m^2)$ (lower bounds can be found but we use the standard matrix multiplication algorithm). Also note that the activation function applied at each layer, presumably, takes constant time for any given input.

Since each of the n layers takes $O(m^2)$ time for the matrix multiplication (i.e. summation function of entire layer) and $O(m)$ time for the activation function of each entry, we have that the entire network takes $O(n(m^2 + m)) = \mathbf{O(m^2 n)}$ time to be queried. Note that the first and last layers take les time to perform matrix multiplication since they are smaller, but this does not affect the overall complexity.

Part b: Each of the n layers has m^2 weights (since they are represented by $m \times m$ matrices). The first layer actually only has mi weights and the output has only mj, with $i, j \le m$, but this does not affect the asymptotic complexity. Put together this totals a spatial complexity of $\mathbf{O(m^2 n)}$.