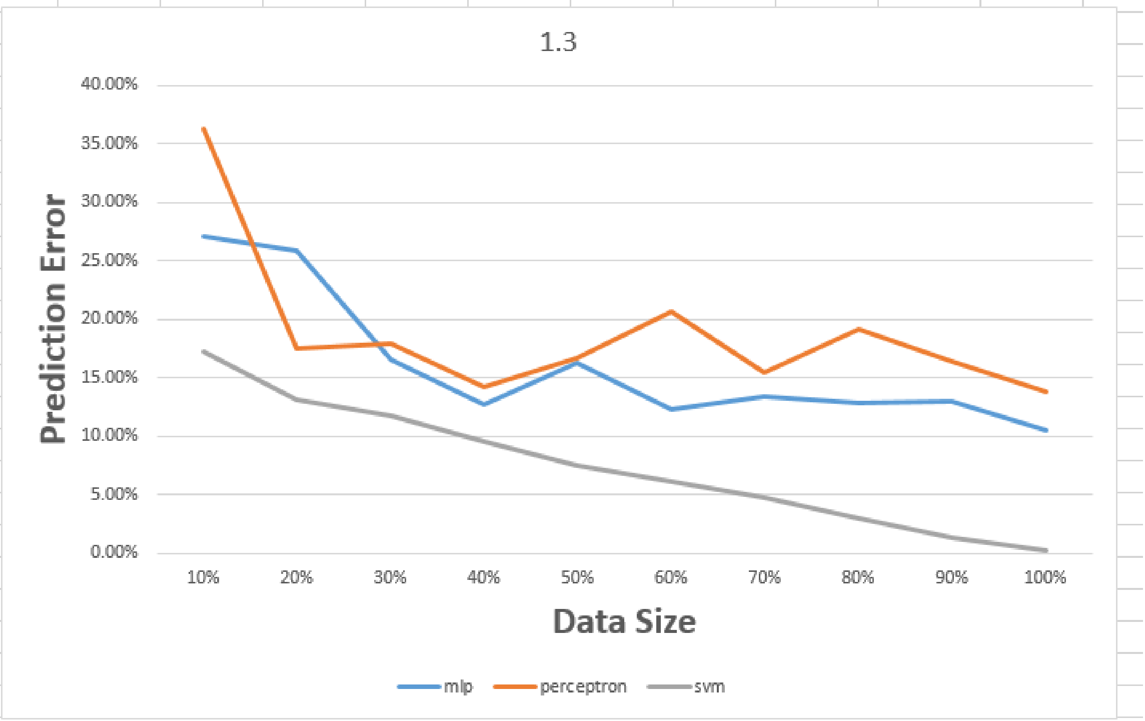
Assignment 2 Supervised Learning

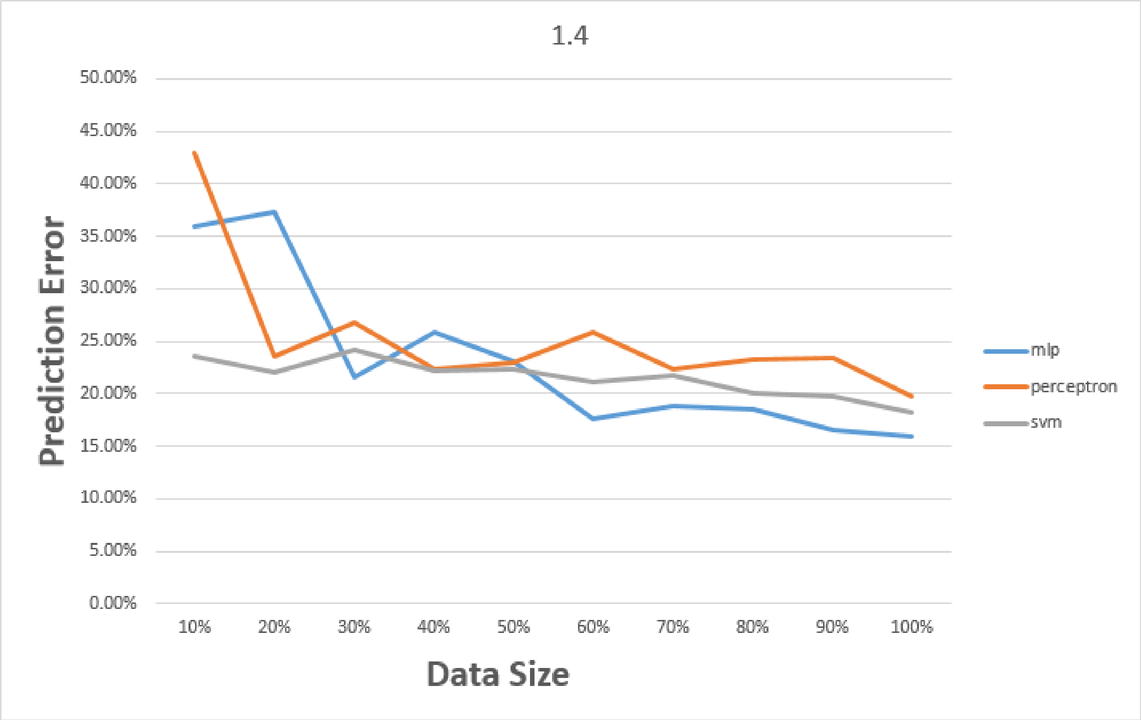
By Brandon Young and Ruicheng Wu

1 problem 1

1.3



1.4



1.5

From the graph above, with smaller training sets, the perceptron performs the worse (43% error) and the multi-layer neural network (MLP) does better with a 36% error rate and lastly SVM performs the best (24% error). Soon after the amount of training examples increases, the perceptron outperforms the neural network, perhaps because the perceptron can learn the weights quicker. However, the neural network overtakes the perceptron with just a slight increase in data.

With enough training examples, the perceptron performs the worse, with a 25% error rate and SVM comes in at 2nd with an error rate of about 18%. The neural network performs the best with a 16% error rate.

All three methods have trouble identifying similar-looking numbers. For example, with a large amount of training samples, the perceptron identifies a 7 as a 9 and the SVM predicts a 9 to be a 3.

To fine-tune the perceptron, we tried using several learning rates, from 0.1 to 0.0001 and settled with $\alpha = 0.01$, where the algorithm tends to perform the best. For the multi neural-network, there was choice of how many nodes should be used in the hidden layer. There was no fine-tuning needed for SVM, since the sci-kit learn package was used.

With additional time and computational resources we would add more training examples and especially more 3s, 7s and 9s. Then the learning algorithms can better distinguish identifying features in similar looking numbers. For the multi-layer neural network, we would add more hidden layers to allow the network to classify/be more specific about the features of a number. Also, we would add a validation set as part of the algorithms.

In short, the neural network performs the best, although all three methods improve with larger training sets and perform reasonably well with enough data.

2 problem 2

2.1 a.

We claim that the provided tree correctly categorizes the provided examples since every example's class can be determined from this decision tree. Examples 1, 2 and 3 with a GPA above 3.8 are in P and those with a GPA below 3.3 (examples 9, 10, 11 and 12) are in N.

For the rest of the examples, examples 4 and 6 had prior research and they are in P, otherwise check their University Rank. That leaves examples 5, 7 and 8. Examples 8 and 7 have rank 1 and 3 respectively and are correctly placed in the N class. Lastly example 5 has rank 2 and is accepted. Recommendation doesn't matter according to this tree.

2.2 b.

Step I

I($\frac{6}{12}$, $\frac{6}{12}$)=1

GPA: $[3.9, 4.0]$ 3(PPP) , $(3.2, 3.9)$ 5$(PPPNN)$, $[3.0, 3.2]$ 4$(NNNN)$
University: Rank 1— 5(PPPNN),Rank 2— 3(PPN) , Rank 3— 4(PNNN)
Publication: Yes 5(PPPNN) , No 7(PPPNNNN)
Recommendation: good 8(PPPPPNNN) , normal 4(PNNN)

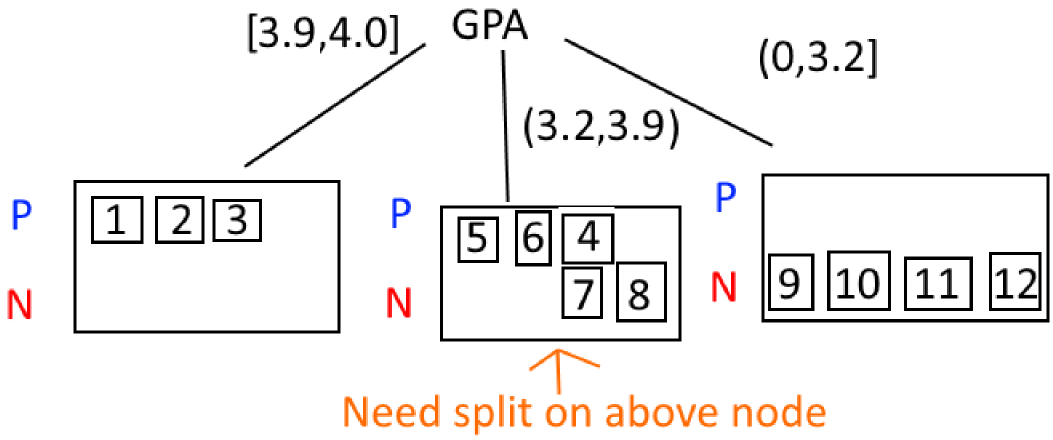
Gain(GPA)=1$-$[$\frac{3}{12}$ B($\frac{3}{3}$)+$\frac{5}{12}$ B($\frac{3}{5}$)+$\frac{4}{12}$ B($\frac{0}{4}$)]=1-[0.0+0.405+0.0] = 0.595

Gain(University)=1$-$[$\frac{5}{12}$ B($\frac{3}{5}$)+$\frac{3}{12}$ B($\frac{2}{3}$)+$\frac{4}{12}$ B($\frac{1}{4}$)]= 1-[0.405+0.2306+0.2704] =0.09544

Gain(Publication)=1-[$\frac{5}{12}$ B($\frac{3}{5}$)+$\frac{7}{12}$ B($\frac{3}{7}$)]=1-[0.405+0.575]= 0.0207

Gain(Recommendation)=
1-[$\frac{8}{12}$ B($\frac{5}{8}$)+$\frac{4}{12}$ B($\frac{1}{4}$)]=1-[0.636+0.270]= 0.0933

So we pick GPA as the best Gain attribute in this level



Step II

$I(\frac{2}{5},\frac{3}{5})$=0.971

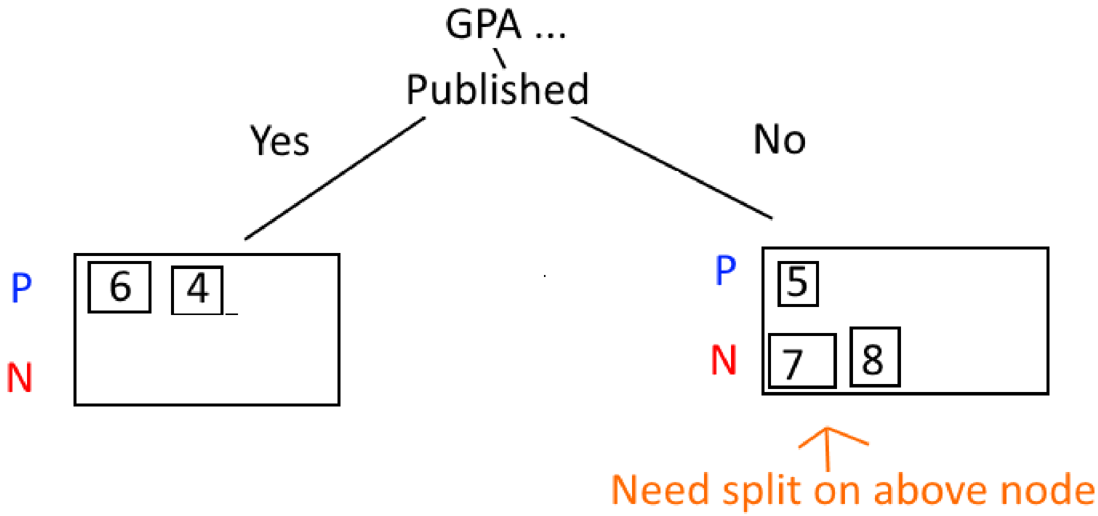
University: Rank 1— 2(PN),Rank 2— 1(P) , Rank 3— 2(PN)
Publication: Yes 2(PP) , No 3(PNN)
Recommendation: good 5(PPPNN)

Gain(University)=0.971-[$\frac{2}{5}$ B($\frac{1}{2}$)+$\frac{1}{5}$ B($\frac{1}{1}$)+$\frac{2}{5}$ B($\frac{1}{2}$)]= 0.971-[0.4+0.0+0.4] = 0.171

Gain(Publication)=0.971-[$\frac{3}{5}$ B($\frac{1}{3}$)+$\frac{2}{5}$ B($\frac{2}{2}$)]=0.971-[0.551+0.0]= 0.420

Gain(Recommendation)=0.971-[$\frac{5}{5}$ B($\frac{3}{5}$)]=0.971-[0.971+0.0+0.0] = 0

So we pick Publication as the best Gain attribute in this level



Step III

I($\frac{1}{3}$, $\frac{2}{3}$)=0.9183

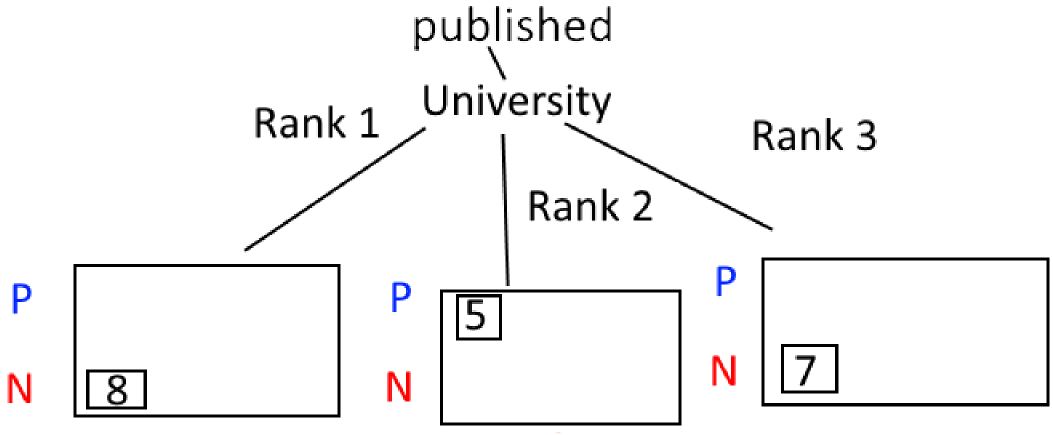
University: Rank 1 — 1(N),Rank 2 — 1(P) , Rank 3 — 1(N)

Recommendation: good 3(PNN)

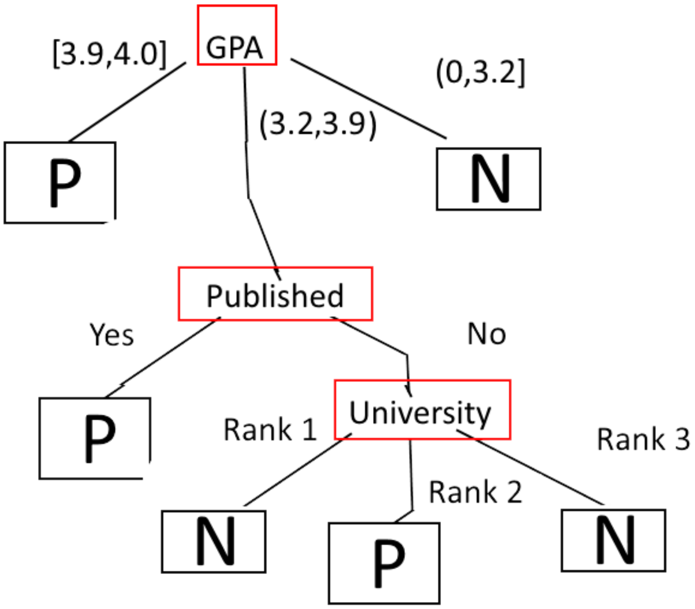
Gain(University)=0.9183-[$\frac{1}{3}$ B($\frac{0}{1}$)+$\frac{1}{3}$ B($\frac{1}{1}$)+$\frac{1}{3}$ B($\frac{0}{1}$)]=0.9183-[0.0+0.0+0.0]=0.9183

Gain(Recommendation)=0.9183-[$\frac{3}{3}$ B($\frac{1}{3}$)]=0.9183-[0.9183]= 0

So we pick University as the best Gain attribute in this level



And the final tree to be returned is



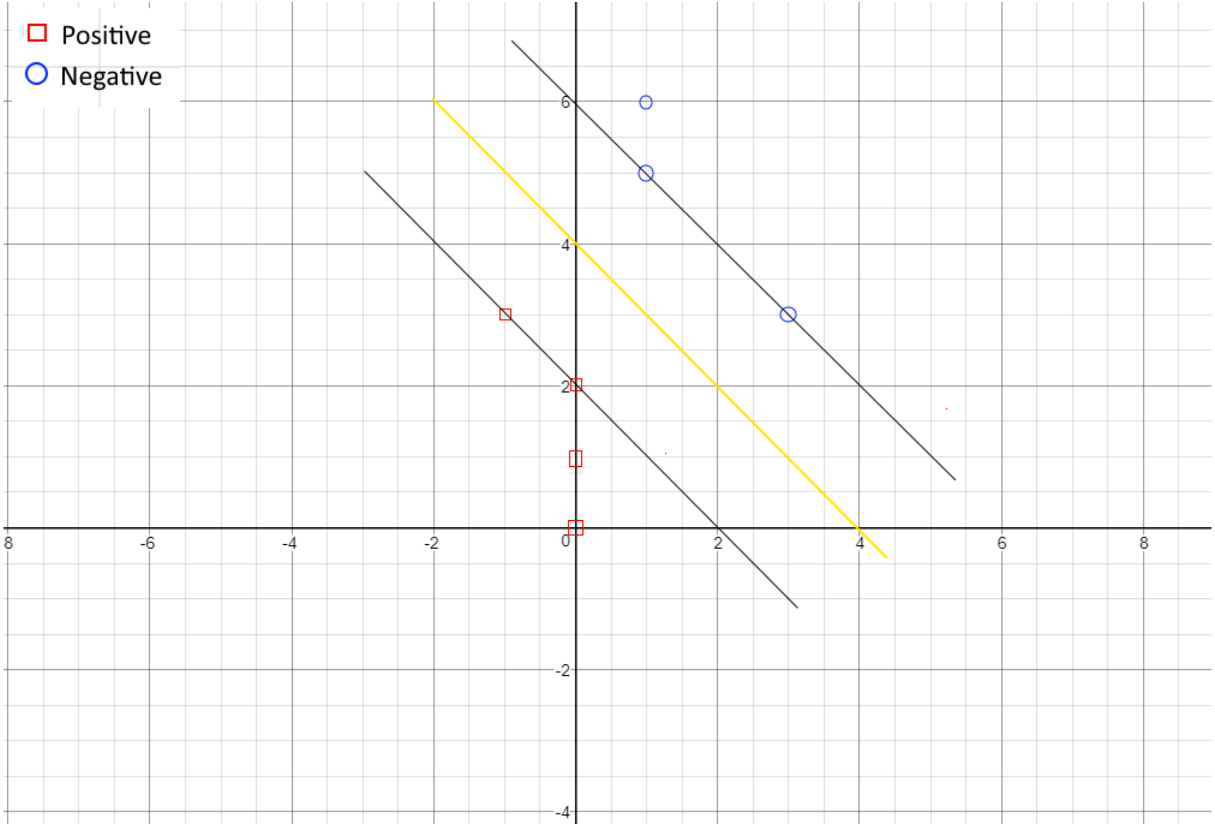
2.3 c.

The decision tree we got from (b) is the same as the provided tree. This is not coincidence, the decision tree algorithm shown in class always generates the same tree from the same data.

3 problem 3

3.1 a.

The horizontal axis is x_1, the vertical axis is x_2. The linear classifier is the yellow line.



3.2 b.

s_1 : $(1,5)$, s_2 : $(-1,3)$, w : $(1,1)$, So decision boundary is: $y = (1,1)x - 4$ Assume there is (a,a) for w^T, we will have

$$a + 5a + b = -1$$

$$-a + 3a + b = 1$$

Solve those equations for :

$$a = -0.5, b = 2$$

So we have $w^T : (-0.5, -0.5)$ and $b : 2$ for parameters. We can confirm the max margin is $\frac{2}{||w^T||} = \frac{2}{(\frac{\sqrt{2}}{2})} = 2\sqrt{2}$, which is the same as the distance between $s_1 : (1, 5)$ and $s_2 : (-1, 3)$: $\sqrt{(1-(-1))^2 + (5-3)^2} = 2\sqrt{2}$. The linear SVM is:

$$h(x) = (-0.5, -0.5)x + 2.$$

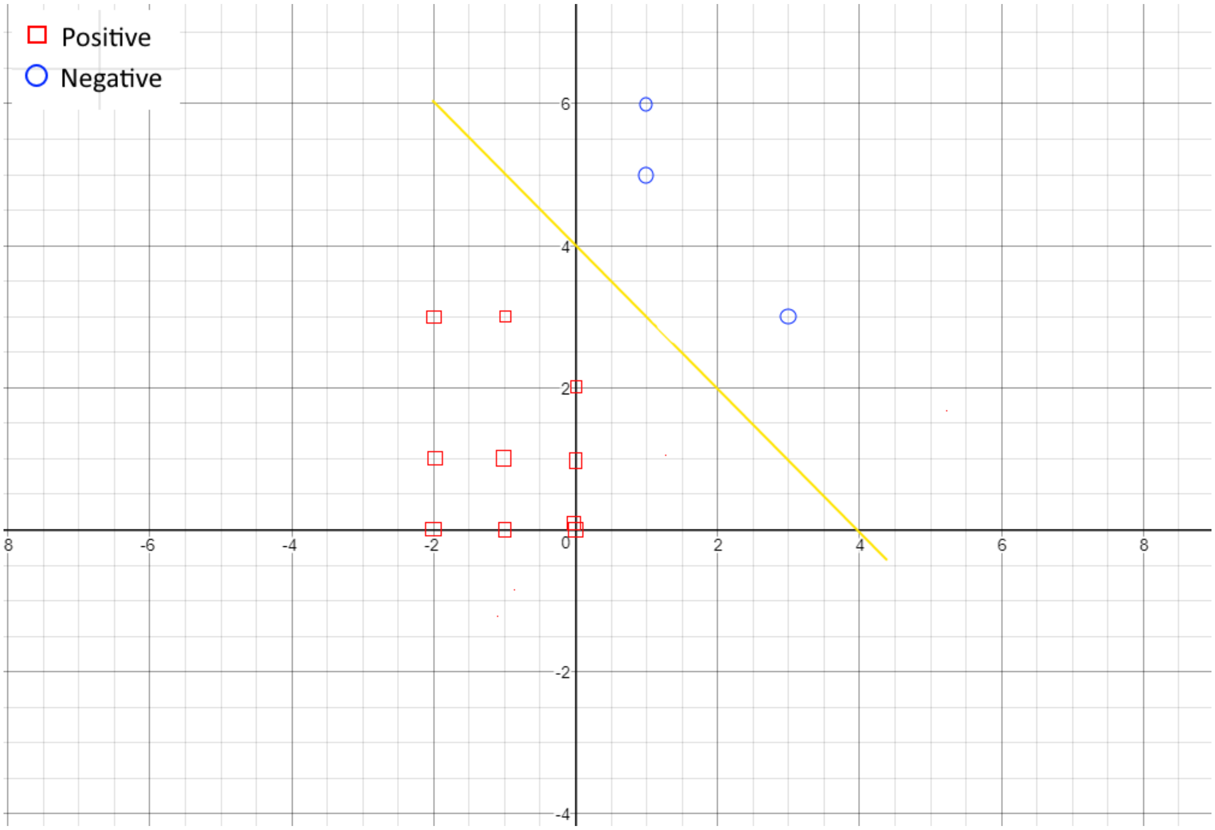
Checking with the given data:
[-1 3]^T [0 2]^T [0 1]^T [0 0]^T, plug in: all positive , so those points are in class 1 (the positive class). [1 5]^T [1 6]^T [3 3]^T, plug in: all negative, so those points are in class -1 (the negative class).

3.3 c.

By inspection, the new added points are not support vectors (i.e. the closest points to separating line)

We claim that $w^T : (-0.5, -0.5)$ and $b : 2$ are still same, even with the new data.

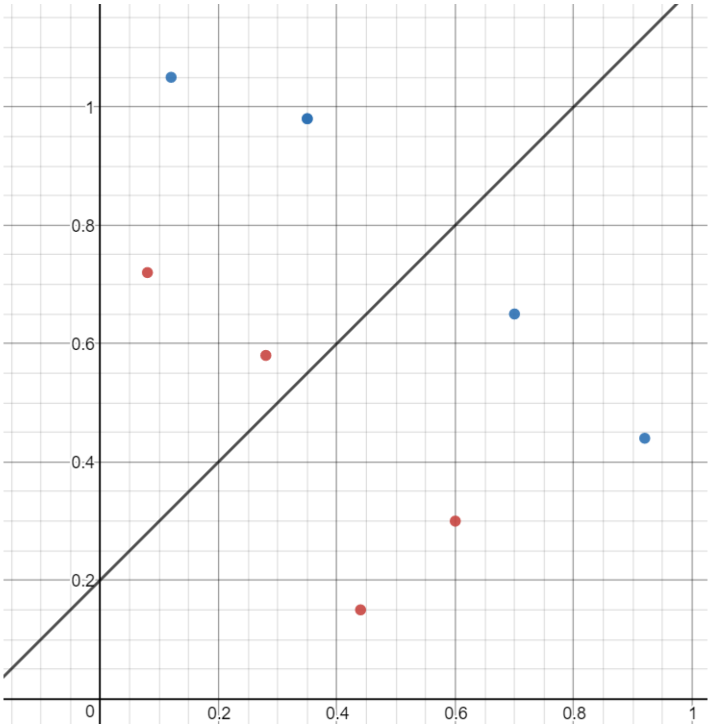
[-2 0]^T [-2 1]^T [-2 3]^T [-1 0]^T [-1 1]^T [0 0]^T, plug in: All positive , so those are predicted to be in class 1, which is correct.


□ Positive
○ Negative
6
4
2
0
-2
-4
8
-6
-4
-2
2
4
6
8

4 problem 4

4.1 a.

The initial linear separator (line 1) is $i_2 = -\frac{i_1}{i_2} - \frac{0.2}{i_2}$ or $i_2 = i_1 + 0.2$:



Assume that the points are:

A:	$(0.08, 0.72)$
B:	$(0.28, 0.58)$
C:	$(0.44, 0.15)$
D:	$(0.6, 0.3)$
E:	$(0.12, 1.05)$

- F: $(0.35, 0.98)$
- G: $(0.7, 0.65)$
- H: $(0.92, 0.44)$

where A, B, C and D are in class 1 and E, F, G and H are in class -1.

Use $f(x) = w_1 i_1 + w_2 i_2 + 0.2$ to classify the samples. If $f(x) < 0$ then x is in class -1 and $h_w(x) = 0$. If $f(x) \geq 0$ then x is in class 1 and $h_w(x) = 1$. By the initial linear separator, 4 samples are misclassified (A, B, G and H):

- A: $(0.08) - (0.72) + 0.2 = -0.44 < 0, h_w(A) = 0$
- B: $(0.28) - (0.58) + 0.2 = -0.1 < 0, h_w(B) = 0$
- C: $(0.44) - (0.15) + 0.2 = 0.49 > 0, h_w(C) = 1$
- D: $(0.6) - (0.3) + 0.2 = 0.5 > 0, h_w(D) = 1$
- E: $(0.12) - (1.05) + 0.2 = -0.73 < 0, h_w(E) = 0$
- F: $(0.35) - (0.98) + 0.2 = -0.43 < 0, h_w(F) = 0$
- G: $(0.7) - (0.65) + 0.2 = 0.25 > 0, h_w(G) = 1$
- H: $(0.92) - (0.44) + 0.2 = 0.68 > 0, h_w(H) = 1$

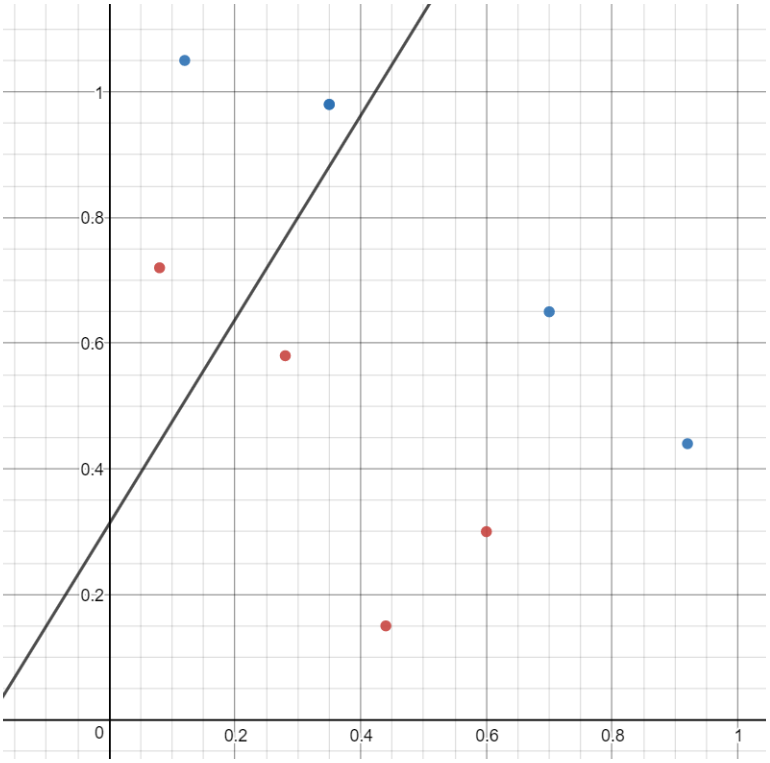
Let the learning rate, $\alpha = 0.5$ and use A to update the weights. Then $Err_A = y - h_w(A) = 1 - 0 = 1$.

$$w_1' \leftarrow w_1 + \alpha * Err_A * i_1$$

$$w_1' \leftarrow 1 + (0.5)(1)(0.08)$$
$$w_1' = 1.04$$

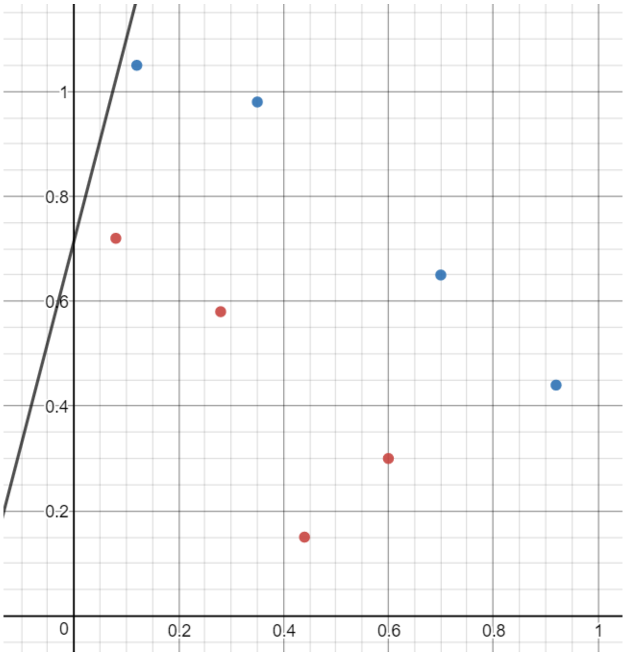
$$w'_2 \leftarrow (-1) + (0.5)(1)(0.72)$$
$$w'_2 = -0.64$$

Hence the new weights are $w_1 = 1.04$ and $w_2 = -0.64$. Line 2, then, is: $i_2 = 1.625(i_1) + 0.3125$, with 3 misclassified points:

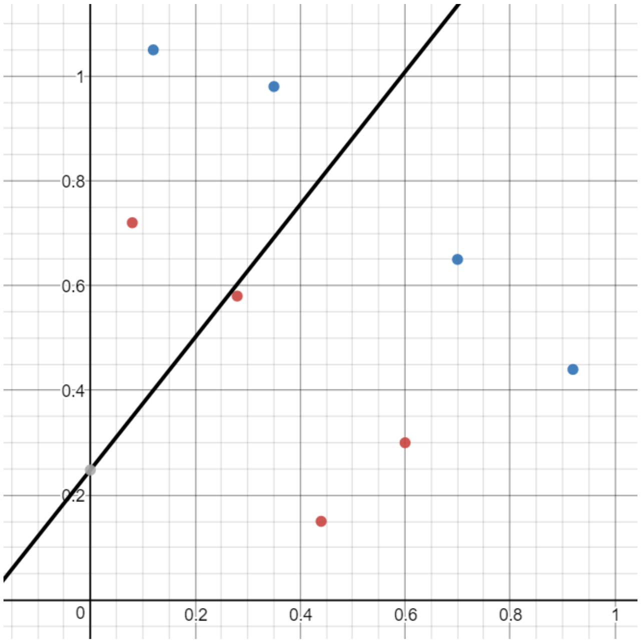


Following a similar process as above, the next few iterations generate the following plots:

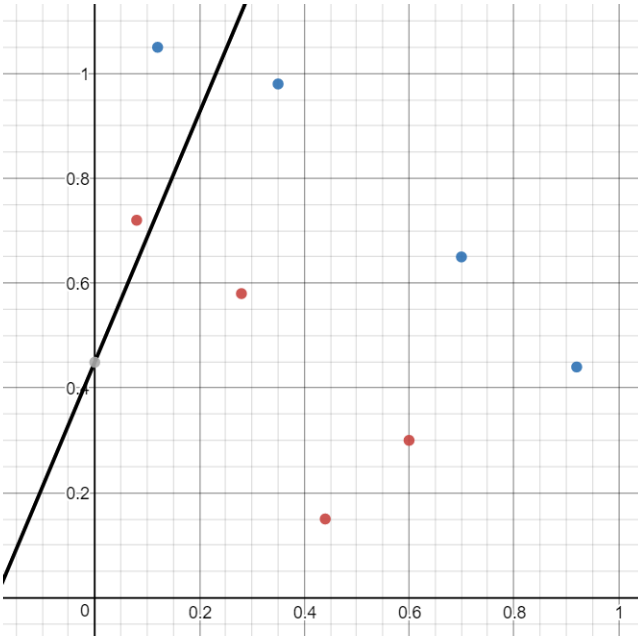
Update Line 2 with A to get $w_1 = 1.08$, $w_2 = -0.28$. Line 3: $i_2 = 3.857i_1 + 0.714$, with 4 misclassified points:



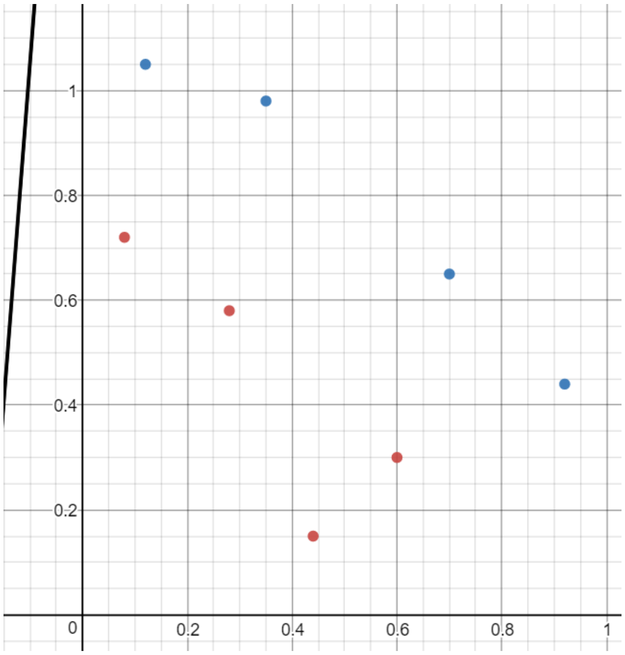
Update Line 3 with E to get $w_1 = 1.02$, $w_2 = -0.805$. Line 4, $i_2 = 1.267(i_1) + 0.248$, with 3 misclassified points:



Update Line 4 with A to get $w_1 = 1.06$, $w_2 = -0.445$. Line 5, $i_2 = 2.382(i_1) + 0.449$, with 4 misclassified points:



Update Line 5 with A to get $w_1 = 1.1$, $w_2 = -0.085$. Line 6, $i_2 = 12.941(i_1) + 2.35$, with 4 misclassified points:

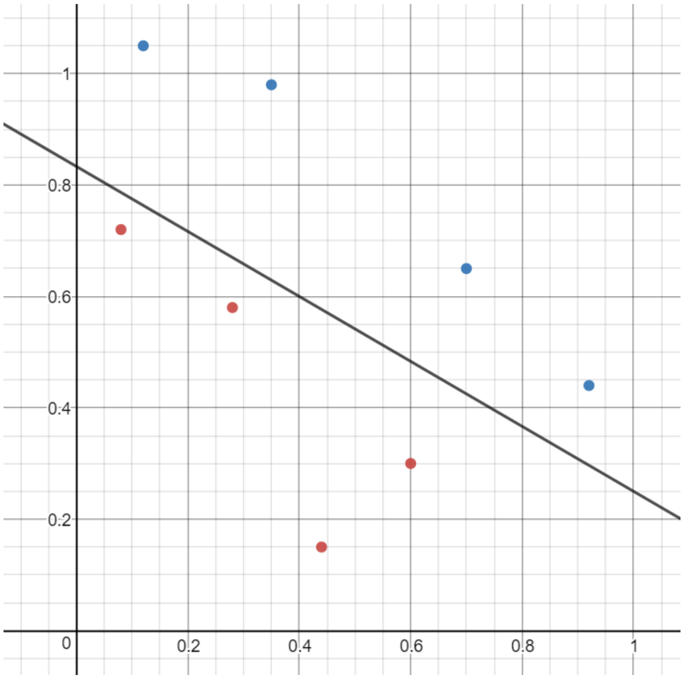


4.2 b.

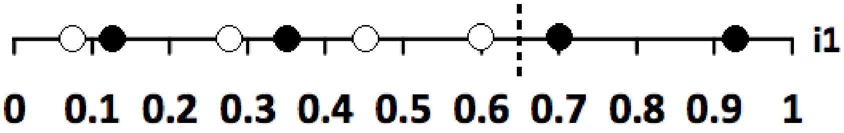
The linear separator that achieves perfect classification (found after 172 iterations) is:

$$i_2 = -0.58333 i_1 + 0.83333$$

where $w_1 = -0.14$ and $w_2 = -0.24$. The line in the graph:



4.3 c.



The diagram above shows only the i_1 value from each sample. The dotted line represents the location of the best possible split, which is at $i_1 = 0.65$, with only 2 misclassified points and a 25% error.

A separator in 1D is $w_1 i_1 + w_0 i_0$ or $w_1 i_1 + 0.2$ (assuming $w_0 = 0.2$) which can be used to solve for w_1:

$$w_1 = -\frac{0.2}{i_1} = -\frac{0.2}{0.65} = -0.3077$$

As a result, $w_1 = -0.3077$ is the weight that best classifies the samples.

5 problem 5

5.1 a.

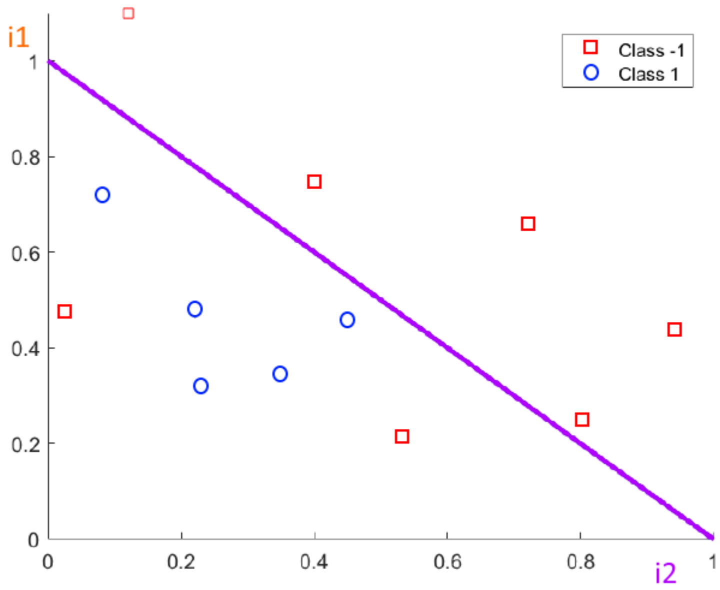
We define those points by inspection.

Class 1: (0.03,0.50),(0.11,1.2),(0.5,0.75),(0.54,0.23),(0.7,0.65),(0.8,0.25),(0.91,0.4

Class -1: (0.07,0.73),(0.23,0.49),(0.24,0.33),(0.35,0.35),(0.46,0.46);

The line can be observed from the graph, and we found the best single perceptron will have two unclassied points. So the minimum error will be $\frac{2}{12} = 0.167$;

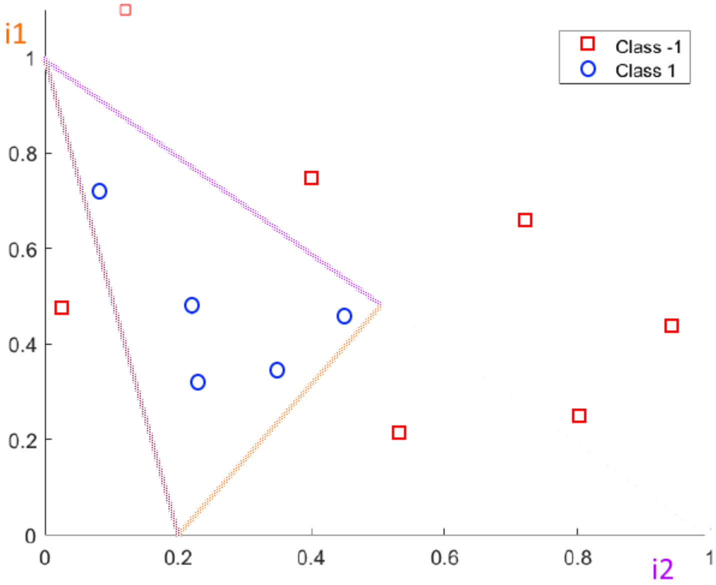
The formula for this dividing line is : $1 - 1 *^T x_1 - 1 * x_2 = 0$



5.2 b.

We claim that we need 3 separate lines to completely separate two classes. classes. So we need at least 3 perceptrons to compute classification functions. We will just use 3 perceptrons for simplicity.

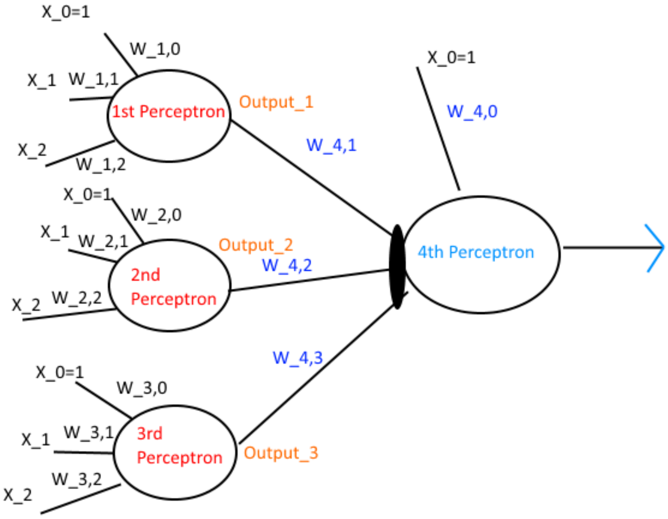
Formula for line 1:
$1 - 1 * x_1 - 1 * x_2 = 0$
$W_{1,0} = 1, W_{1,1} = -1, W_{1,2} = -1$
Formula for line 2:
$0.2 - 1 * x_1 + 0.6 * x_2 = 0$
$W_{1,0} = 0.2, W_{1,1} = -1, W_{1,2} = 0.6$
Formula for line 3:
$-0.2 + 1 * x_1 + 0.2 * x_2 = 0$
$W_{1,0} = -0.2, W_{1,1} = 1, W_{1,2} = 0.2$



As we can see, a data point gives output as positive with all three perceptrons is class 1 and same applies to class -1. So in the next layer, we can simply apply "and" operation to the results of all three perceptrons. In that case, we only need one unit. And the output will be 1 if and only if all perceptrons in the 1st layer output 1.

Hence we set

$$W_{4,0} = -2.5, W_{4,1} = 1, W_{4,2} = 1, W_{4,3} = 1$$


X_0=1
W_1,0
X_1
W_1,1
1st Perceptron
Output_1
X_2
W_1,2
W_4,1
X_0=1
W_2,0
X_1
W_2,1
2nd Perceptron
Output_2
W_4,2
X_2
W_2,2
X_0=1
W_4,0
4th Perceptron
W_4,3
X_0=1
W_3,0
X_1
W_3,1
3rd Perceptron
Output_3
X_2
W_3,2