Assignment 2

Priyanka Dhulkhed
Liam O'Brien

October 25, 2017

1 Question 1

For the perceptron function we set each element of the training data to *datum* and each trainingLabel to *correct* We then go through the classify function with each datum and increment our guess array. If the guess is incorrect we increase the weights of the correct element and sum it with the datum while simultaneously decreseasing the weight of the guess. For the perceptron these are the following results for each respective percentage of training data:

10%: Validating... 59 correct out of 100 (59.0%). Testing... 59 correct out of 100 (59.0%).
20%: Validating... 105 correct out of 200 (52.5%). Testing... 98 correct out of 200 (49.0%).
30%: Validating... 188 correct out of 300 (62.7%). Testing... 162 correct out of 300 (54.0%).
40%: Validating... 247 correct out of 400 (61.8%). Testing... 241 correct out of 400 (60.2%).
50%: Validating... 333 correct out of 500 (66.6%). Testing... 303 correct out of 500 (60.6%).
60%: Validating... 419 correct out of 600 (69.8%). Testing... 405 correct out of 600 (67.5%).
60%: Validating... 451 correct out of 700 (64.4%). Testing... 458 correct out of 700 (65.4%).
70%: Validating... 549 correct out of 800 (68.6%). Testing... 514 correct out of 800 (64.2%).
80%: Validating... 686 correct out of 900 (76.2%). Testing... 649 correct out of 900 (72.1%).
90%: Validating... 731 correct out of 900 (81.2%). Testing... 725 correct out of 900 (80.6%).
100%: Validating... 820 correct out of 1000 (82.0%). Testing... 783 correct out of 1000 (78.3%).

For the SVM portion we used sklearn to automatically classify the data points. We used the SVM built in function *fit* to our points and training labels. In the classify function, we put our points into an array called *guess* and implement the *predict* on our values. For the SVM these are the following results for each respective percentage of training data:

10%: Validating... 75 correct out of 100 (75.0%). Testing... 75 correct out of 100 (75.0%).
20%: Validating... 163 correct out of 200 (81.5%). Testing... 153 correct out of 200 (76.5%).
30%: Validating... 247 correct out of 300 (82.3%). Testing... 222 correct out of 300 (74.0%).
40%: Validating... 310 correct out of 400 (77.5%). Testing... 304 correct out of 400 (76.0%).
50%: Validating... 412 correct out of 500 (82.4%). Testing... 385 correct out of 500 (77.0%).
60%: Validating... 493 correct out of 600 (82.2%). Testing... 475 correct out of 600 (79.2%).
70%: Validating... 584 correct out of 700 (83.4%). Testing... 563 correct out of 700 (80.4%).
80%: Validating... 658 correct out of 800 (82.2%). Testing... 614 correct out of 800 (76.8%).
90%: Validating... 731 correct out of 900 (81.2%). Testing... 725 correct out of 900 (80.6%).
100%: Validating... 820 correct out of 1000 (82.0%). Testing... 783 correct out of 1000 (78.3%).

2 Question 2

a.) Yes this tree correctly categorizes the provided information. For each of the 12 cases in the previous year, all of the cases are correctly represented on the tree diagram.

b.)

$$I(\frac{6}{12}, \frac{6}{12}) = -(\frac{6}{12})\log_2(\frac{6}{12}) - (\frac{6}{12})\log_2(\frac{6}{12}) = 1$$

For each entropy, we must find the fraction of those who were accepted less those who were declinded for each . First we must calculate the entropy for the three breakdowns in GPA:

$$E(GPA \geq 3.9) = -(\frac{3}{3})\log_2(\frac{3}{3}) - (\frac{0}{3})\log_2(\frac{0}{3}) = 0$$

$$E(3.9 > GPA > 3.2) = -(\frac{3}{5})\log_2(\frac{3}{5}) - (\frac{2}{5})\log_2(\frac{2}{5}) = 0.292$$

$$E(GPA \leq 3.2) = -(\frac{0}{4})\log_2(\frac{0}{4}) - (\frac{4}{4})\log_2(\frac{4}{4}) = 0$$

Next we need calculate the entropy of whether or not the applicant has prior research:

$$E(Resarch = "Yes") = -(\frac{3}{5})\log_2(\frac{3}{5}) - (\frac{2}{5})\log_2(\frac{2}{5}) = 0.292$$

$$E(Research = "No") = -(\frac{3}{7})\log_2(\frac{3}{7}) - (\frac{4}{7})\log_2(\frac{4}{7}) = 0.019$$

Next we need to calculate the entropy of the three ranks:

$$E(Rank = "1") = -(\frac{3}{5})\log_2(\frac{3}{5}) - (\frac{2}{5})\log_2(\frac{2}{5}) = 0.276$$

$$E(Rank = "2") = -(\frac{2}{3})\log_2(\frac{2}{3}) - (\frac{1}{3})\log_2(\frac{1}{3}) = 0.244$$

$$E(Rank = "3") = -(\frac{1}{4})\log_2(\frac{1}{4}) - (\frac{3}{4})\log_2(\frac{3}{4}) = 0.292$$

Lastly, we need to calcuate the entropy of the quality of the recommendation letter:

$$E(Recommendation = "Good") = -(\frac{5}{8})\log_2(\frac{5}{8}) - (\frac{3}{8})\log_2(\frac{3}{8}) = 0.287$$

$$E(Recommendation = "Normal") = -(\frac{1}{4})\log_2(\frac{1}{4}) - (\frac{3}{4})\log_2(\frac{3}{4}) = 0.292$$

Now, we need to find the I for each student with a P:

$$I(GPA) = (\frac{3}{12}) * (0) + (\frac{5}{12}) * (0.292) + (\frac{0}{12}) * (0) = .122$$

$$I(Research) = (\frac{5}{12}) * (0.292) + (\frac{7}{12}) * (0.019) = .133$$

$$I(Rank) = (\frac{5}{12}) * (0.276) + (\frac{3}{12}) * (0.244) + (\frac{4}{12}) * (0.292) = .273$$

$$I(Recommendation) = (\frac{8}{12}) * (0.287) + (\frac{4}{12}) * (0.292) = .289$$

Finally we need to calculate the Gain for each of these weights:

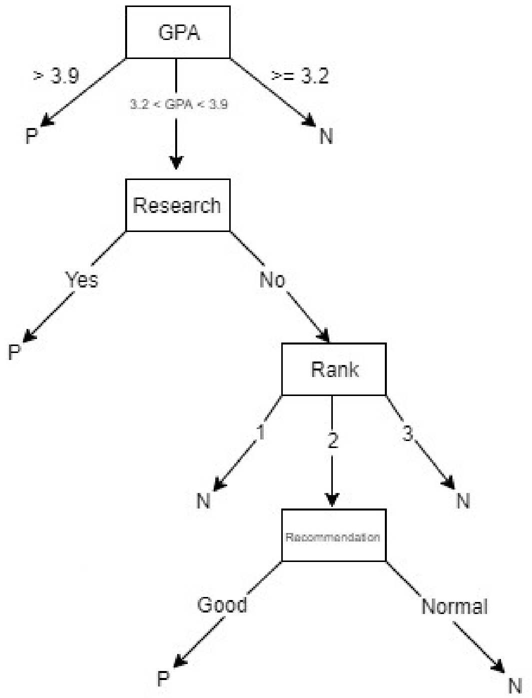
$$Gain(Students, GPA) = 1 - .122 = .878$$

$$Gain(Students, Research) = 1 - .133 = .867$$

$$Gain(Students, Rank) = 1 - .273 = .727$$

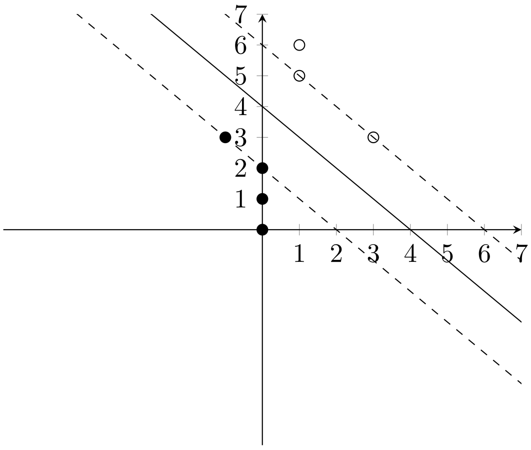
$$Gain(Students, Recommendation) = 1 - .289 = .711$$

Based on our gain, and ordering the tree diagram in the order of weights, this is what the diagram now looks like:



c) The tree we got in part b) is equivalent to the tree provided but it is not exactly the same. It is equivalent in the sense that the weights are the same but the only differences are instead of prior research, the university is looking for publication. Second, there is an extra node called "recommendation" that has the least amount of gain which is why it is the last node on the list. This is not a coincidence because we are not changing the attributes but we are adding additional ones. This additional data can be used to help classify applicants who fall in the middle numbers (from the previous year, it would help to better classify students near No. 6-8).

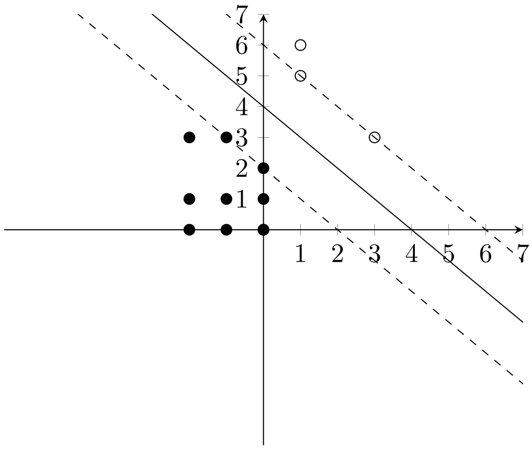
3 Question 3



a) The above graph represents our linear classifier by inspection. Note that positive plots are solid and negative plots are hollow. Our linear separator is indicated by the solid line, and our margin lines are dashed.

b) w = [-1, -1], and our b (y intercept) = 4. This leads us to the final equation:
h(x) = sign([-1, -1] * $[x_i, x_j]^T + 4$)

c)



The above graph shows the additional positive points included. For part c), our linear SVM is the same as in section b) because the additional points were all positive (i.e., to the left of the linear separator) and they did not lie within the left margin, so these additional points because they are of all the same class, have no effect to the linear SVM.

4 Question 4

In order to answer this question, the hypothesis function that is used is:

$h_w(i_1, i_2) = w_0(1) + w_1(i_1) + w_2(i_2)$

The threshold function is as follows:

If $h_w(i_1, i_2) < 0$ –> Fires 0
If $h_w(i_1, i_2) \geq 0$ –> Fires 1

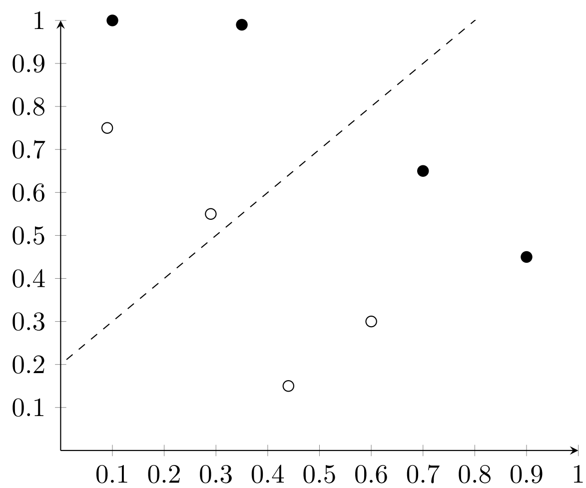
The points that are approximated from the given graph:

The negative points in the form (i_0, i_1):
(0.09, 0.75)
(0.29, 0.55)
(0.44, 0.15)
(0.6, 0.3)

The positive points in the form (i_0, i_1):
(0.1, 1)
(0.35, 0.99)
(0.7, 0.65)
(0.9, 0.45)

The values approximated from the points above are:
$h_w(0.09, 0.75) = (0.2)(1) + (1)(0.09) + (-1)(0.75) = -0.46$ –> Fires 0 (Calculated Correctly)
$h_w(0.29, 0.55) = (0.2)(1) + (1)(0.29) + (-1)(0.55) = -0.06$ –> Fires 0 (Calculated Correctly)
$h_w(0.44, 0.15) = (0.2)(1) + (1)(0.44) + (-1)(0.15) = 0.49$ –> Fires 1 (Calculated Incorrectly)
$h_w(0.6, 0.3) = (0.2)(1) + (1)(0.6) + (-1)(0.3) = 0.5$ –> Fires 1 (Calculated Incorrectly)
$h_w(0.1, 1.00) = (0.2)(1) + (1)(0.1) + (-1)(1.00) = -0.7$ –> Fires 0 (Calculated Incorrectly)
$h_w(0.35, 0.99) = (0.2)(1) + (1)(0.35) + (-1)(0.99) = -0.44$ –> Fires 0 (Calculated Incorrectly)
$h_w(0.7, 0.65) = (0.2)(1) + (1)(0.7) + (-1)(0.65) = 0.25$ –> Fires 1 (Calculated Correctly)
$h_w(0.09, 0.45) = (0.2)(1) + (1)(0.09) + (-1)(0.45) = 0.65$ –> Fires 1 (Calculated Correctly)

This is the original plot that has been generated by using the initial weights $w_0 = 0.2$, $w_1 = 1$, and $w_2 = -1$



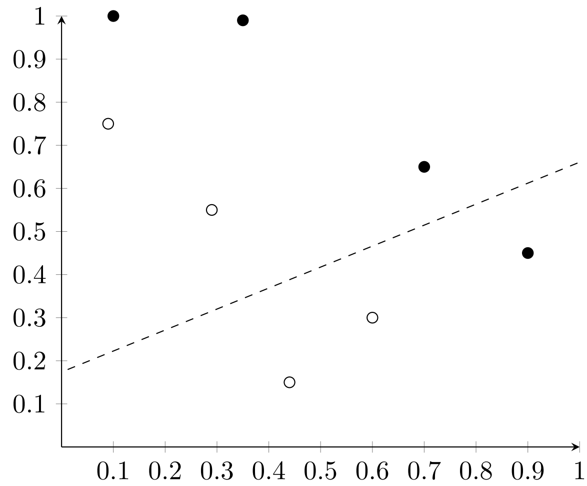
So, we apply the weight update rule:
$w_i := w_i + \alpha(y - h_w(x))(x_i)$, with learning rate $\alpha = 1.0$

We arbitrarily chose one of the misclassifed ones to change our weights:
The misclassified point that we will choose to update our weights is $(0.44, 0.15)$.
The actual output (y) is 0
Out hypothesis function fires 1 for this point.
Applying the weight update rule to these two points, we get:
$w_1 := w_1 + (1)(y - h_w(i_1, i_2))(i_1)$
$w_1 := 1 + (1)(0 - 1)(0.44) = 0.56$
$w_2 := w_2 + (1)(y - h_w(i_1, i_2))(i_2)$
1 $w_2 := -1 + (1)(0 - 1)(0.15) = -1.15$

The new hypothesis function we have is:
$h_w(i_1, i_2) = (0.2)(1) + (0.56)(i_1) + (-1.15)(i_2)$

Now, we calculate the output for all the points and find out the number of points misclassified.

The values approximated from the points above are:
$h_w(0.09, 0.75) = (0.2)(1) + (0.56)(0.09) + (-1.15)(0.75) = -0.6121$–>Fires 0 (Calculated Correctly)
$h_w(0.29, 0.55) = (0.2)(1) + (0.56)(0.29) + (-1.15)(0.55) = -0.2701$ –> Fires 0 (Calculated Correctly)
$h_w(0.44, 0.15) = (0.2)(1) + (0.56)(0.44) + (-1.15)(0.15) = 0.2739$ –> Fires 1 (Calculated Incorrectly)
$h_w(0.6, 0.3) = (0.2)(1) + (0.56)(0.6) + (-1.15)(0.3) = 0.191$ –> Fires 1 (Calculated Incorrectly)
$h_w(0.1, 1.00) = (0.2)(1) + (0.56)(0.1) + (-1.15)(1.00) = -0.894$ –> Fires 0 (Calculated Incorrectly)
$h_w(0.35, 0.99) = (0.2)(1) + (0.56)(0.35) + (-1.15)(0.99) = -0.7425$–> Fires 0 (Calculated Incorrectly)
$h_w(0.7, 0.65) = (0.2)(1) + (0.56)(0.7) + (-1.15)(0.65) = -0.155$ –> Fires 0 (Calculated Incorrectly)
$h_w(0.09, 0.45) = (0.2)(1) + (0.56)(0.09) + (-1.15)(0.45) = -0.2671$ –> Fires 0 (Calculated Incorrectly)

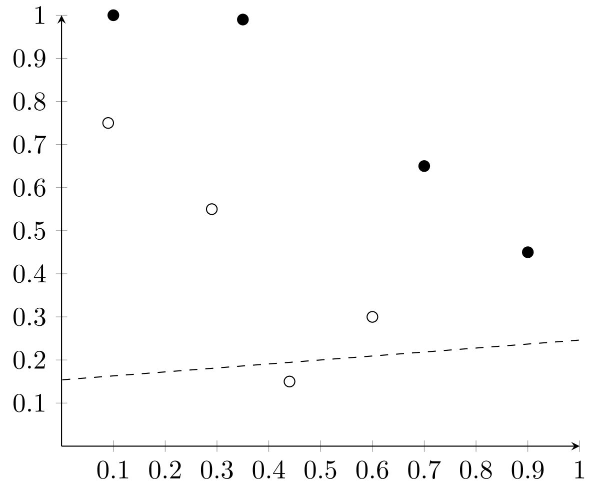


As we see from the above plot, the points above the line should be negative (open circles) and points above the line should be positive. We can see from the graph and the description above that 6 points have been labelled incorrectly by the hypothesis function.

We will run another iteration of updating the weights with the same $\alpha = 1$
The misclassified point that we will choose to update our weights is $(0.44, 0.15)$.

We use the same misclassified point (0.44,0.15)
The actual output (y) is 0
Out hypothesis function fires 1 for this point.
Applying the weight update rule to these two points, we get:
$w_1 := w_1 + (1)(y - h_w(i_1, i_2))(i_1)$
$w_1 := 0.56 + (1)(0 - 1)(0.44) = 0.12$
$w_2 := w_2 + (1)(y - h_w(i_1, i_2))(i_2)$
1 $w_2 := -1.15 + (1)(0 - 1)(0.15) = -1.3$

The new hypothesis function we have is:
$h_w(i_1, i_2) = (0.2)(1) + (0.12)(i_1) + (-1.3)(i_2)$

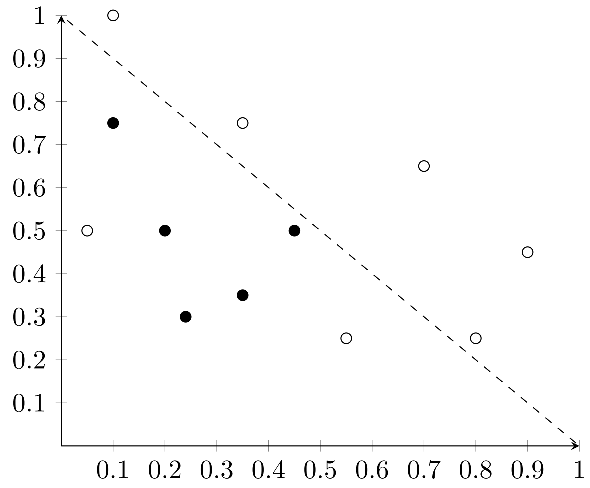


5 Question 5

For this question, Class 1 are the solid points, and Class -1 are hollow points. These are our estimated values for each class:

For Class 1:
(0.1, 0.75)
(0.2, 0.50)
(0.24, 0.30)
(0.35, 0.35)
(0.45, 0.50)

For Class -1:
(0.05, 0.50)
(0.1, 1.00)
(0.35, 0.75)
(0.55, 0.25)
(0.70, 0.65)
(0.80, 0.25)
(0.90, 0.45)

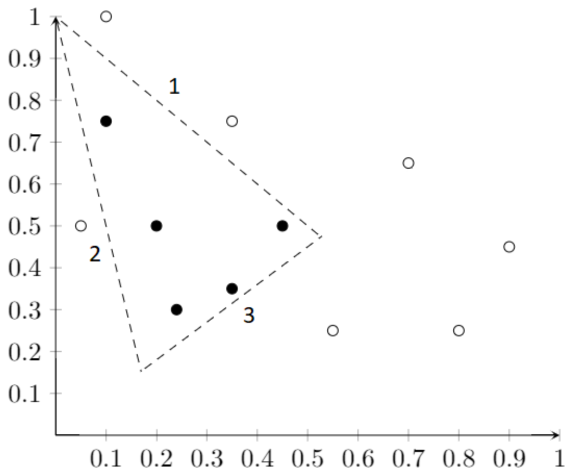


For the line y = -x + 1 we were able to have only just 2 points of error out of 12. This means that one perceptron has an error of .167. The two points that have these errors are both of class -1 and have the following coordinates:

(0.05, 0.50)
(0.55, 0.25)

From our initial line, to figure out the weights, we have:
1 - 1*$i_1 - 1 * i_2 = 0$
Solving this shows the values of our weights as:
$w_0 = 1$
$w_1 = -1$
$w_2 = -1$

b)



The above triangle formed by the three dashed lines shows that with 3 perceptrons we can correctly classify the positive and negative class. Everything inside of this triangle will be Class 1 and everything else outside will be of Class -1. To determine the weights of the perceptrons we must solve three equations from the three lines.

For our first line labeled 1 we can use the equation from above for our first perceptron:
1 - 1*$i_1 - 1 * i_2 = 0$
So the weights for our first perceptron are:
$w_0 = 1$
$w_1 = -1$
$w_2 = -1$

For our second line labeled 2, the equation looks like:
-0.2 $+ 1 * i_1 + 0.2 * i_2 = 0$

Solving this shows that the weights for our second perceptron are:
$w_0 = -0.2$
$w_1 = 1$
$w_2 = 0.2$

For our third line labeled 3, the equation looks like:
0.2 - $1 * i_1 + 0.55 * i_2 = 0$

Solving this shows that the weights for our third perceptron are:
$w_0 = -0.2$
$w_1 = -1$
$w_2 = 0.55$