CS 440: Introduction to Artificial Intelligence

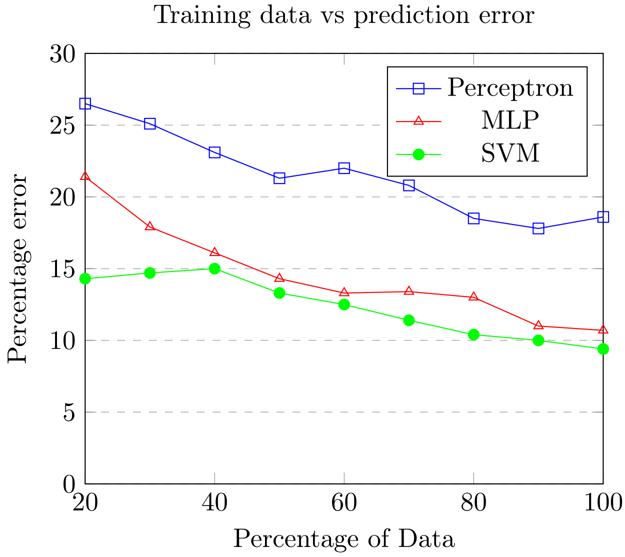
Brandon Smith & Nicholas Grieco

October 23, 2017

Question 1: Optical Character Recognition

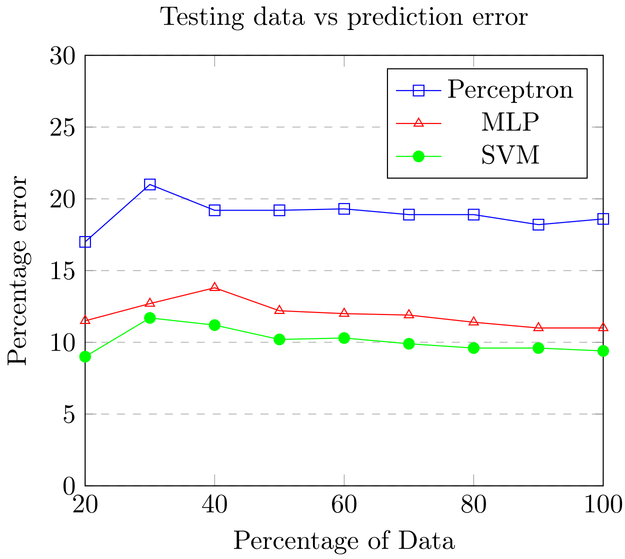
3. Partial Training Results

Results when using 20% to 100% of the training data, when validating and testing on 1000 images.



4. Partial Testing Results

Results when using 20% to 100% of the testing data, when training with 5000 images.



5. Qualitative Evaluation

One can see that the one layer perceptron does the worst. This makes sense since we know that a perceptron is equivalent to a non-linear line. Therefore, a single layer of perceptrons cannot make complex function. The SVM seems to do better than the MLP. SVM can often do better than MLP for lower dimensions such as this problem. Also using a premade library with default values may be tuned to these specific types of problems. I think if I play around a little more with the number of hidden nodes, and learning rate for MLP I can achieve better yields. For SVM using non default values may give me better results. Classical learning systems like neural networks suffer from their theoretical weakness, e.g. back-propagation usually converges only to locally optimal solutions. Here SVMs can provide a significant improvement.

For the training data size we see almost a linear decrease in error across all three learning methods. This shows that the more data we use to train the better our models become. Originally I had the learning rate as 0.5 for MLP and I was getting around 70-80% test data results. After I changed it to 0.05 I started getting 85-90%. Changing the number of hidden nodes and iterations did not have much effect on the error. As for SVM I thought the default values of gamma and C worked best.

Question 2: University of Excellence

a)

The provided tree is classified correctly.

b)

GPA: ≥ 3.9, $3.2 < \text{GPA} < 3.9$, ≤ 3.2
Research: Yes, No
Rank: 1, 2, 3

Level 1

GPA		
	Positive	Negative
≥ 3.9	3	0
$3.2 < \text{GPA} < 3.9$	3	2
≤ 3.2	0	4

Remainder $= \frac{3}{12}$ B(1) + $\frac{5}{12}$ B($\frac{3}{5}$) + $\frac{1}{3}$ B(0)= 0.4046

Research		
	Positive	Negative
Yes	3	2
No	3	4

Remainder $= \frac{5}{12}$ B($\frac{3}{5}$) + $\frac{7}{12}$ B($\frac{3}{7}$) = 0.9792

Rank		
	Positive	Negative
Rank 1	3	2
Rank 2	2	1
Rank 3	1	3

Remainder $= \frac{5}{12}$ B($\frac{3}{5}$) + $\frac{1}{4}$ B($\frac{2}{3}$) + $\frac{1}{3}$ B($\frac{1}{4}$)= 0.9046

Reccomendation		
	Positive	Negative
Good	5	3
Normal	1	3

Remainder $= \frac{8}{12}$ B($\frac{5}{8}$) + $\frac{4}{12}$ B($\frac{1}{4}$) = 0.9067

GPA has the lowest remainder so we choose that option for the first level.
Level 2

Research		
	Positive	Negative
Yes	2	0
No	1	2

Remainder $= \frac{2}{5}$ B(1) + $\frac{3}{5}$ B($\frac{1}{3}$) = 0.5510

Rank		
	Positive	Negative
Rank 1	1	1
Rank 2	1	0
Rank 3	1	1

Remainder $= \frac{2}{5}$ B(1) + B(1) + $\frac{2}{5}$ B($\frac{1}{2}$) = 0.80

Reccomendation		
	Positive	Negative
Good	3	2
Normal	0	0

Remainder = B($\frac{3}{5}$) + B(1) + B(1) = 0.9710

Research has the lowest remainder so we choose that option for the second level.
Level 3

Rank		
	Positive	Negative
Rank 1	0	1
Rank 2	1	0
Rank 3	0	1

Remainder = B(1) + B(1) + B(1) = 0

Reccomendation		
	Positive	Negative
Good	1	2
Normal	0	0

Remainder =B(1) + B($\frac{1}{3}$) = 0.9183

University ranking has the lowest remainder so we choose that option for the last level.

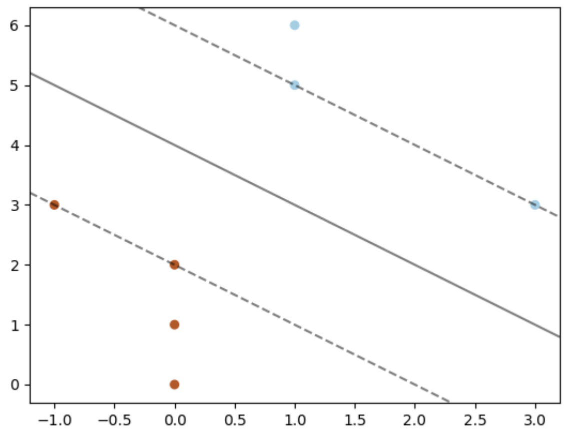
c)

Yes they classify the same way because it's the same tree. It is not a coincidence. At each state the best attribute was chosen. The only time you can have different trees from the same data is at any given 2 attributes have the same remainder and you can choose.

Question 3: SVM

a+b)

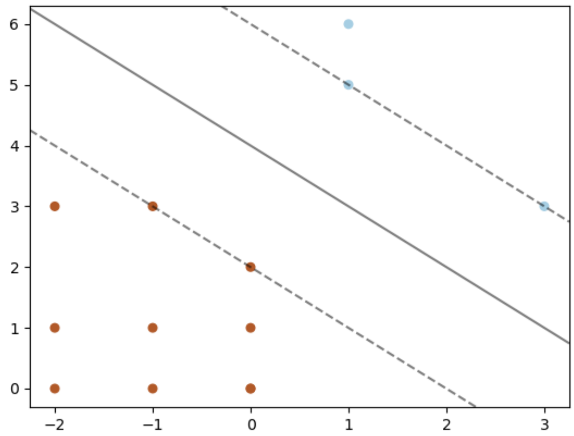
Figure 1: Linear SVM by inspection



Parameters w_1 = -0.5 w_2 = -0.5 b = 2
to fit the equation $w_1x_1 + w_2x_2$ + b = 0

c)

Figure 2: Linear SVM by inspection



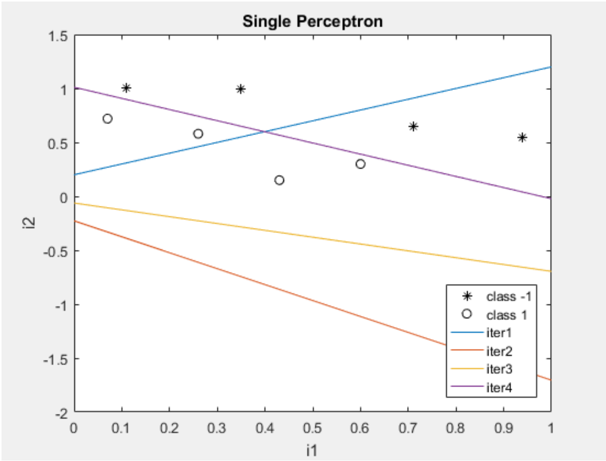
Parameters w_1 = -0.5 w_2 = -0.5 b = 2
to fit the equation $w_1x_1 + w_2x_2$ + b = 0

The parameters are still the same because the new points do not change the margins, so the hyperplane stays the same.

Question 4: Perceptrons

a)

Figure 3: Single Perceptron



By iteration 4 all the points were being perfectly classified.

Listing 1: Perceptron Code

```python
from random import choice
from numpy import array, dot, random

unit_step = lambda x: -1 if x < 0 else 1

training_data = [
    (array([1,0.1,0.72]), -1),
    (array([1,0.15,1.01]), 1),
    (array([1,0.25,0.55]), -1),
    (array([1,0.32,0.95]), 1),
    (array([1,0.45,0.12]), -1),
    (array([1,0.6,0.3]), -1),
    (array([1,0.7,0.6]), 1),
    (array([1,0.9,0.4]), 1),
]

w = [0.2, 1, -1]
errors = []
eta = 1
epoch = 5

print("Initial Weights: " + str(w))
for z in range(epoch):
    print("Epoch: " + str(z + 1))
    for i in xrange(len(training_data)):
        x, expected = training_data[i]
        result = dot(w, x)
        error = expected - unit_step(result)
        errors.append(error)
        w += eta * error * x
    print("Updated weights: " + str(w))
    tmp_err = 0
    for l in range(len(training_data)):
        x, expected = training_data[l]
        result = dot(w, x)
        if expected != unit_step(result):
            tmp_err = tmp_err + 1
    print("Missclassified: " + str(tmp_err))
```

Listing 2: Code output

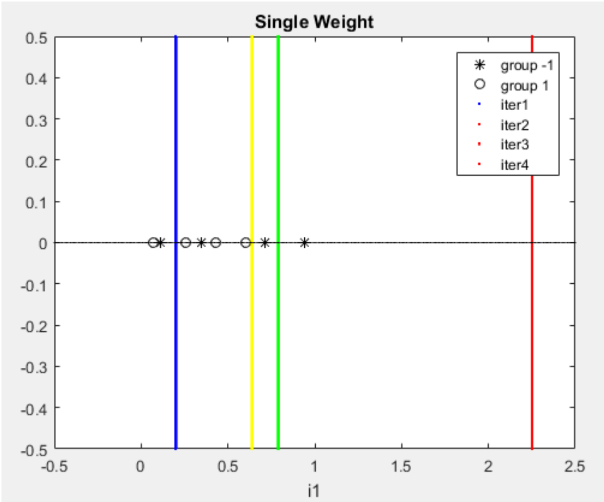
```
Epoch: 1
Updated weights: [ 0.2 1.3  0.88]
Missclassified: 4
Epoch: 2
Updated weights: [ 0.2 2.04 3.22]
Missclassified: 4
Epoch: 3
Updated weights: [-1.8 1.84 1.78]
Missclassified: 0
Epoch: 4
Updated weights: [-1.8 1.84 1.78]
Missclassified: 0
```

b)

The perceptron perfectly classified the set of points.

c)

Figure 4: Single layer perceptrons

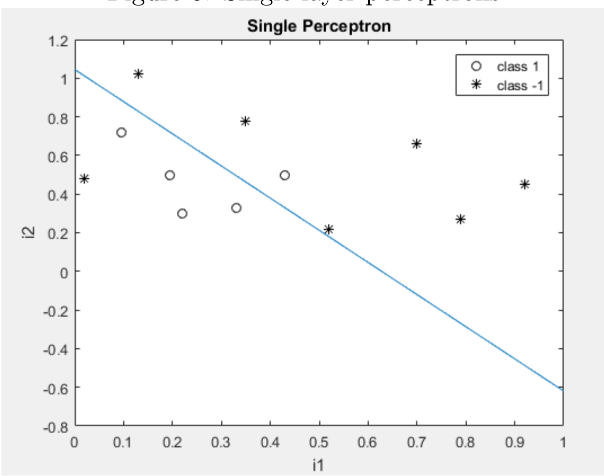


The best we can linearly separate the data points without i_2 is 2 classification errors. We get $b = -1.8$ and $w_1 = 2.82$ making the point it separates at $i_0 = 0.63$

Question 5: Classification

a)

Figure 5: Single layer perceptrons



Using the same code for the perceptron as in question 4, I was able to get a minimum of 2 classification errors. This means the data points are non-linearly separable. A multi-layer perceptron would be better in classifying this set of data.

b)

Figure 6: Multi layer perceptrons

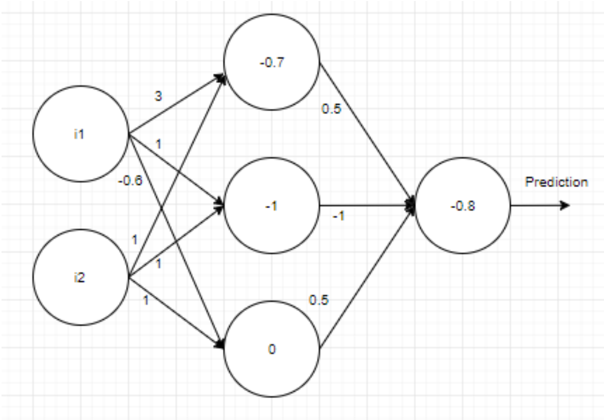
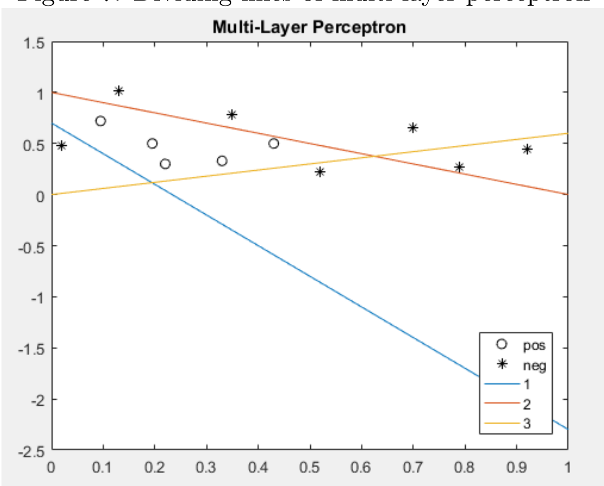


Figure 7: Dividing lines of multi layer perceptron



$w_{11} = 3, w_{12} = 1, w_{13} = -0.6$
$w_{21} = 1, w_{22} = 1, w_{23} = 1$
$b_1 = -0.7, b_2 = -1, b_3 = 0, b_4 = -0.8$