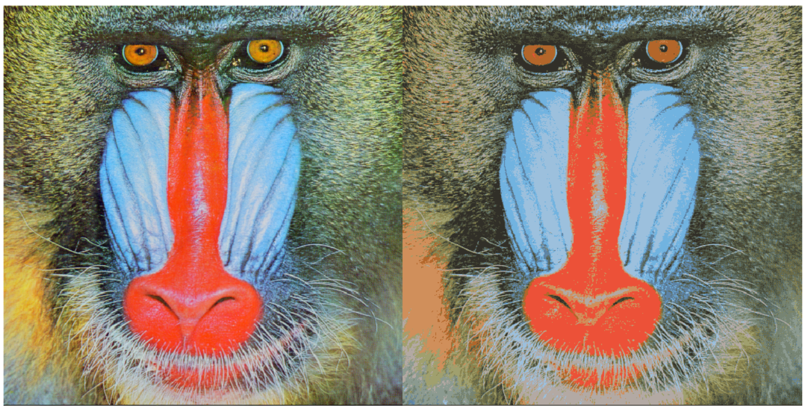
EECS 445 HW4

1.

Code is provided.



1d. For each pixel in our uncompressed image, RGB can take any value. Thus, we have 0-255 values for each of R, G, and B. Each of these can be represented with 8 bits, so we have a total of

$$numerator = 512\ x\ 512\ x\ 24\ bits$$

to store the uncompressed image.

For the compressed image, RGB can only take one of 16 values. Of course, if we want to store this, we need to create a mapping from each point to its nearest centroid, which will require a constant C bits to store that mapping, in addition to the 4-bits required to store the data for the centroids. Thus, for the compressed image, we have a total of

$$denominator = (512\ x\ 512\ x\ 4) + \ C\ bits$$

Thus, our compression factor is

$$\frac{numerator}{denominator} = \frac{512\ x\ 512\ x\ 24}{(512\ x\ 512\ x\ 4) + C} = 6x\ factor\ approximately$$

2 3. (a) From previous hw,

$$\nabla_w E(\vec{w}, b) = \sum_{i=1}^{N} \frac{1}{N}\vec{w} - C\, I\left[t^{(i)}(w^T x^{(i)} + b) < 1\right] t^{(i)} x^{(i)}$$

$$\frac{\partial}{\partial b} E(\vec{w}, b) = \sum_{i=1}^{N} -C I\left[t^{(i)}(w^T x^{(i)} + b) < 1\right] t^{(i)}$$

Update rules. of SGD

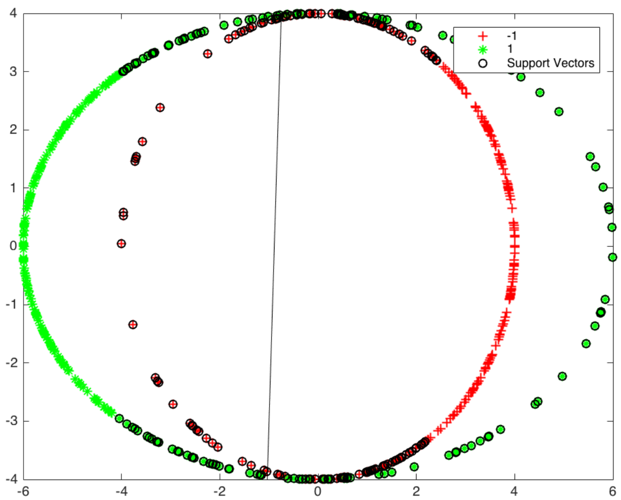
```
w* ← 0        b* ← 0
for j=1 to NumIterations do
    for i=1 to N do
        wgrad ← ∇w E(i)(w*, b*)
        bgrad ← ∂/∂b E(i)(w*, b*)
        w* ← w* - α(j) wgrad
        b* ← b* - 0.01 α(j) bgrad
    end
end
return w*
```

```
--------------------q2b----------------------------
0.100000 is the optimal value of C
0.500000 is the optimal value of n0
9.000000 is the percent error on testing data
---------------------------------------------------
```

```
--------------------q2c----------------------------
1.000000 is the optimal value of C
1.000000 is the optimal value of n0
8.500000 is the percent error on testing data
---------------------------------------------------
```

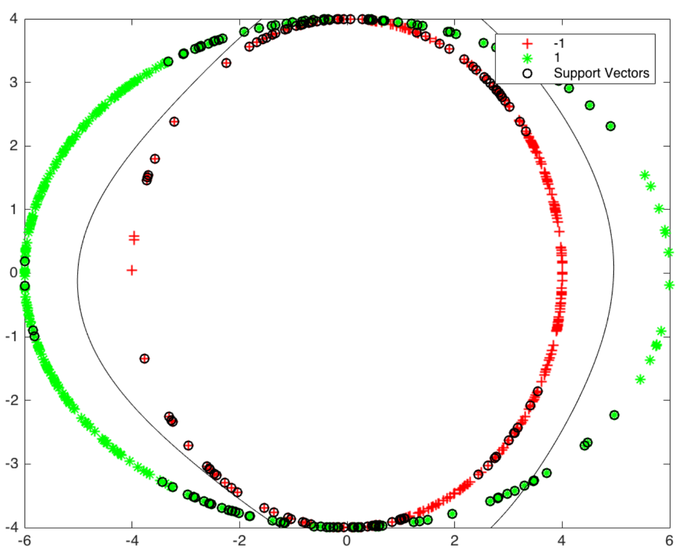
Q3a.

88.0 percent classification accuracy




Q3b.

100.0 percent classification accuracy




Q3c.

Using a Gaussian kernel function allows us to more richly classify our data from not using a kernel, thus making it easier to classify data using an SVM model.

Q3d.

Optimal value of sigma = 0.2
Accuracy = 100%

```
-------------------q3c--------------------------
0.200000 is the optimal value of sigma
-------------------------------------------------
```