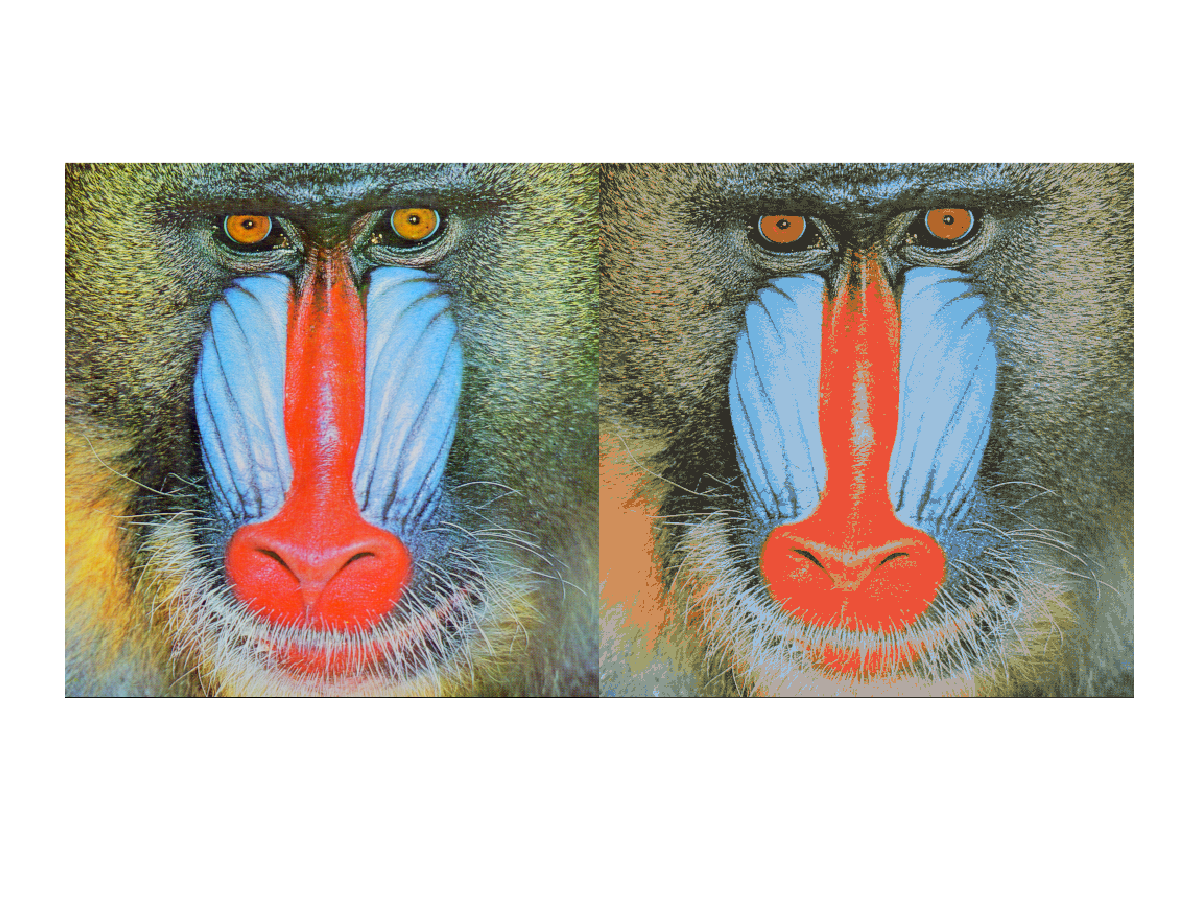
EECS 445 HW4

1.

Code is provided.

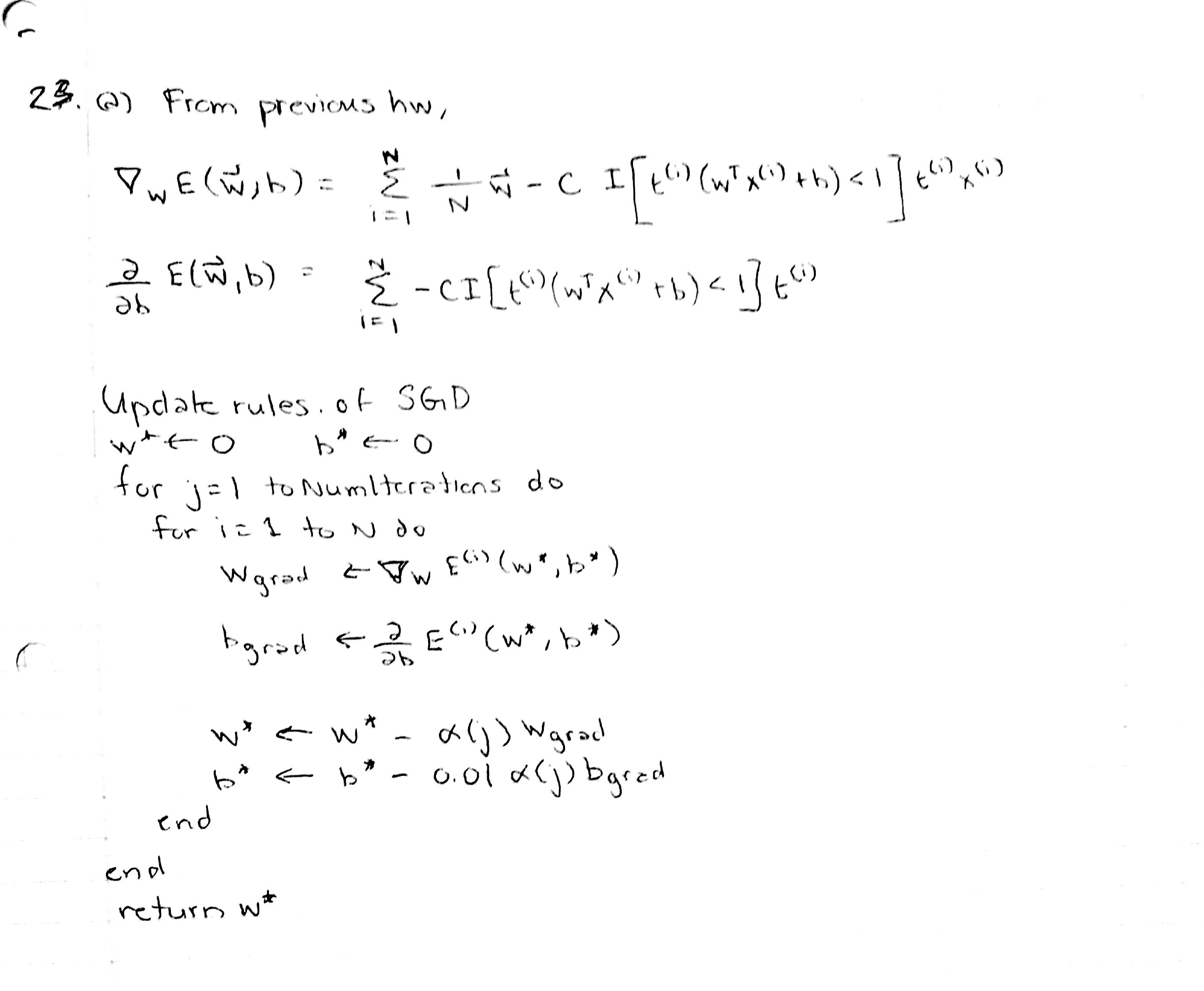
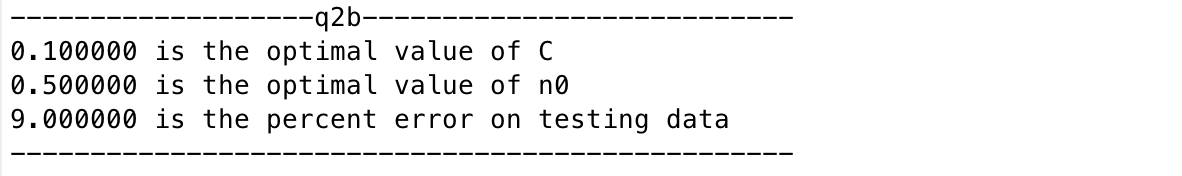


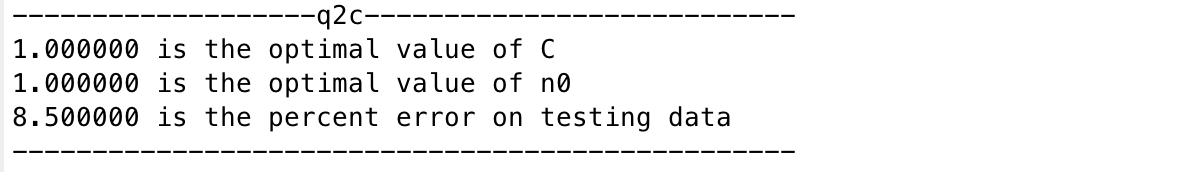
1d. For each pixel in our uncompressed image, RGB can take any value. Thus, we have 0-255 values for each of R, G, and B. Each of these can be represented with 8 bits, so we have a total of

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

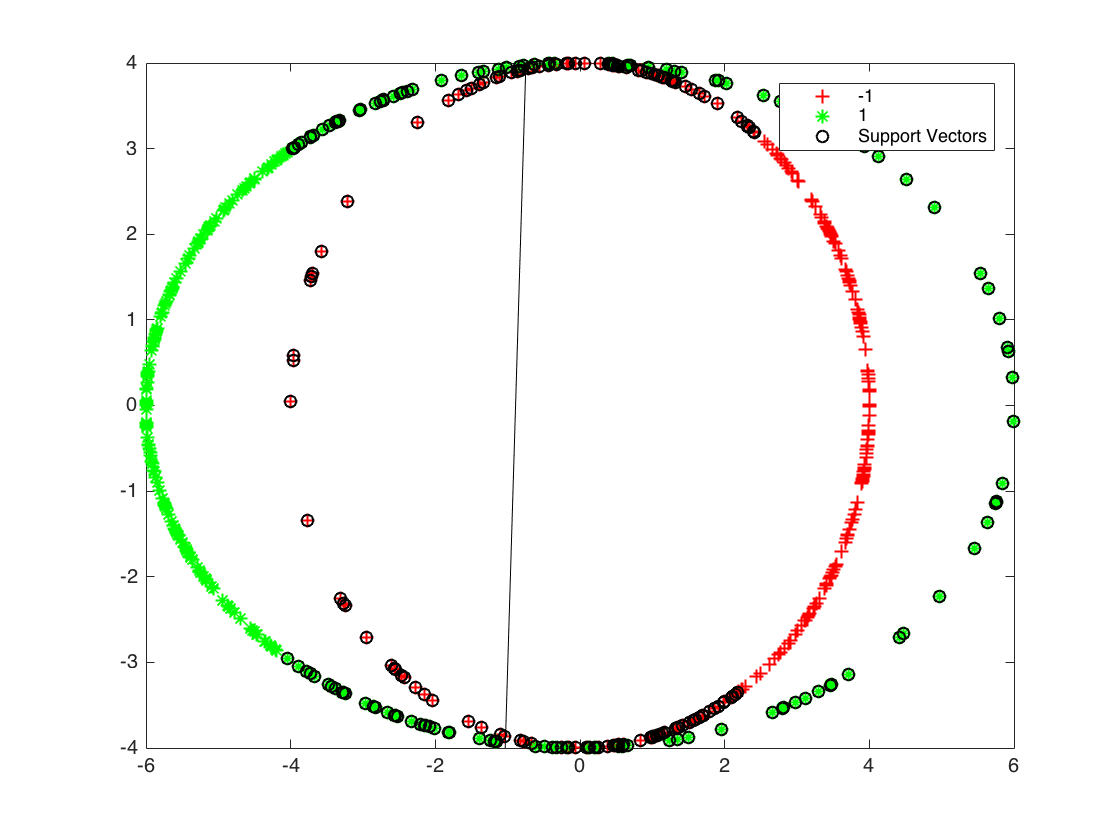
Thus, our compression factor is





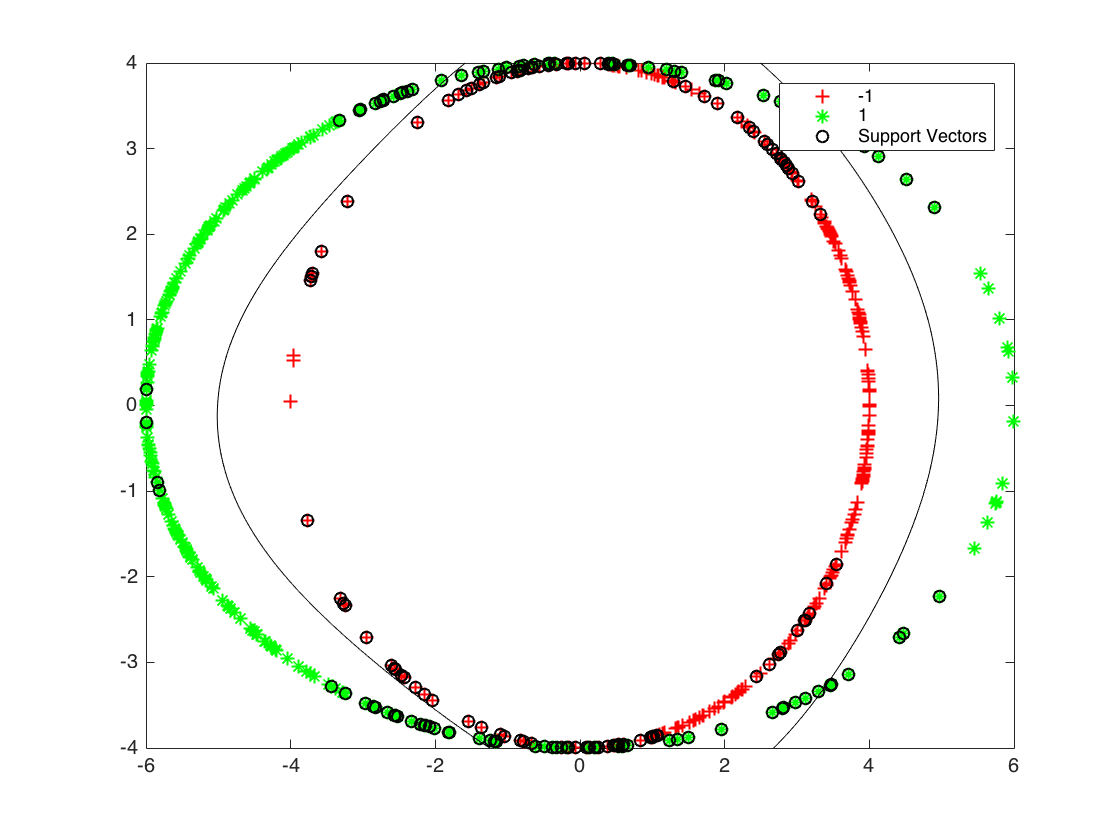
Q3a.

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%

