Duru Naz Han
21902864
CS464-02

CS464: Homework 2

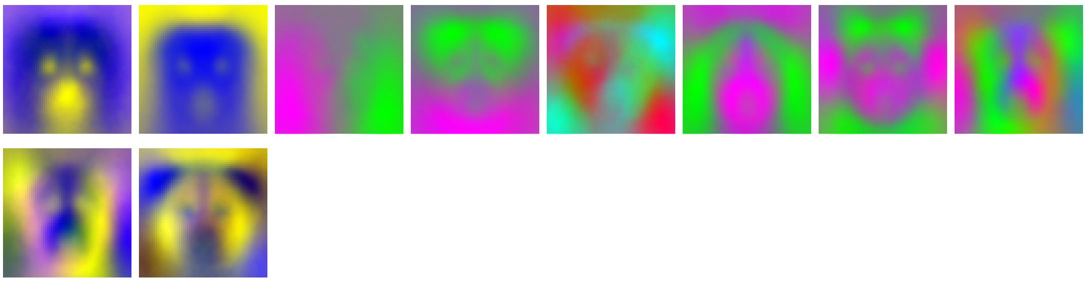
Question 1.1:

First 10 PCA vectors of R,G,B channels are displayed in the jupyter notebook provided for this homework.

For R channel, the minimum number of principal components that are required
to obtain at least 70% PVE is k = 17 (PVE = 70.05665%), for G channel k = 18 (PVE = 70.002110%) and for B channel it is k = 17 (PVE = 70.417384%).

Question 1.2:

Images obtained from stacking R,G,B channels for the first 10 PCA vectors:



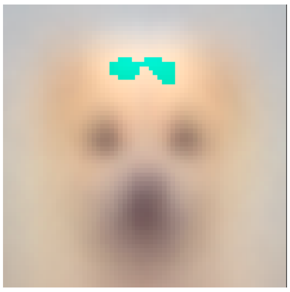
Question 1.3:

Image reconstruction from PCAs can be done in several steps. First, we need to take the dot product of the PCAs with the image that we want to reconstruct. Next, using this new matrix we just obtained, we need to take another dot product with the PCAs to project the image back. While applying these steps, subtracting and adding the mean of all data is essential for better reconstruction.

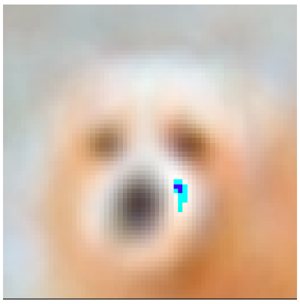
Images obtained from reconstruction with k PCAS where k = [1,50,250,500,1000,4096]:

Original Image:

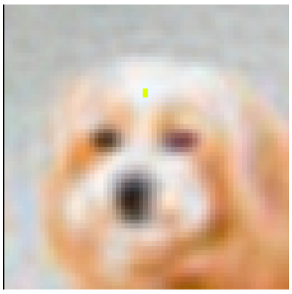
k = 1:



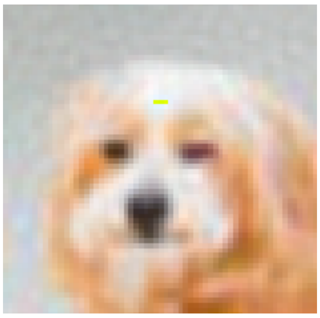
k = 50:



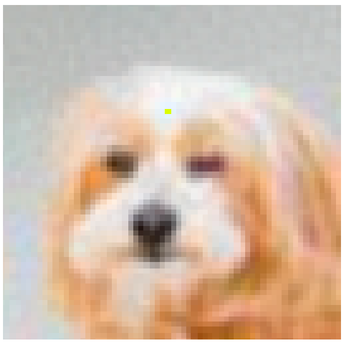
k = 250:



k = 500:



k = 1000:



k = 4096:

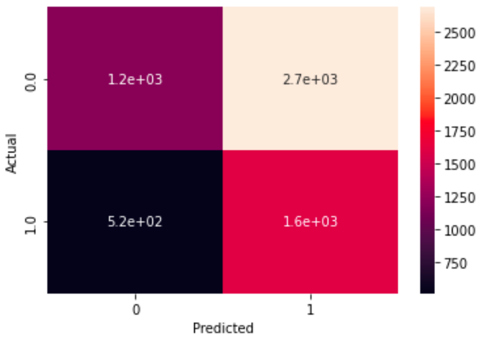
From the above figures, we can see that the quality of the picture got better as the number of PCAs increased. This is because we are increasing the number of futures that we use for reconstruction. However, when we compare the original image and the image with k = 4096, we see that they are still not the same because even though we keep the valuable information (so that we can still understand we are looking at a dog picture), we still leave out some features that enhance the quality of the picture.

Question 2.1:

For mini-batch (batch size = 64) gradient ascent, the accuracy turned out to be 46.65% whereas for full-batch gradient descent it is 35.17%. Confusion matrix and its heatmap for mini-batch is:

```
Accuracy of batch size = 64 is 0.4665

Confusion matrix of batch size = 64:
 (Predicted     0     1
Actual
0.0         1198  2684
1.0          517  1601, <AxesSubplot:xlabel='Predicted', ylabel='Actual'>)
```



The model performance gets better when the batch size decreases.

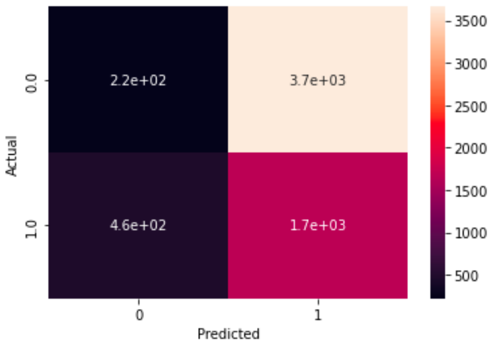
Question 2.1:

For uniform distribution, the accuracy decreases to 31.37% and with zeros, it is higher than gaussian with 64.7% but the confusion matrix shows that it only predicts 0 so this accuracy is not completely reliable as it will only show the amount of actual 0s in the whole dataset. Therefore, best practice seems to be implementing gaussian weights.

Below is the confusion matrix and heatmap for uniform distribution and zeros respectively:

```
Accuracy of batch size = 64 is 0.31366666666666665

Confusion matrix of batch size = 64:
 (Predicted    0     1
Actual
0.0         220  3662
1.0         456  1662, <AxesSubplot:xlabel='Predicted', ylabel='Actual'>)
```




```
Accuracy of batch size = 64 is 0.647

Confusion matrix of batch size = 64:
 (Predicted      0
Actual
0.0         3882
1.0         2118, <AxesSubplot:xlabel='Predicted', ylabel='Actual'>)
```