Q5

For documented code, please see Q5.py and PCA.py

Algorithm for reconstruction:
Borrowing terminology from the report for Q2, the projection matrix $P_{N\times R}$ holds the projections of instances from our sample ($S_{N\times M}$) onto the R ($R = 84$) principle modes of variation, represented by the matrix $E_{M\times R}$, where M is the original number of dimensions (784 in this case). As $E_{M\times R}$ represents the principle modes of variation in the M-basis, multiplying each row-vector $E_{M\times 1}$ by the projection of an instance (say, $S_{1\times M}$, which has projection $P_{1\times R}$), should return the instance's representation in M-basis, shifted by the sample mean ($\mu_{1\times M}$). Reconstruction (denoted by $X_{1\times M}$) of instance $S_{1\times M}$:

$$X_{1\times M} = P_{1\times R} \times (E^T)_{R\times M} + \mu_{1\times M}$$

If applied to all instances:

$$X_{N\times M} = P_{N\times R} \times (E^T)_{R\times M} + \mu_{N\times M}$$

where $\mu_{N\times M}$ is obtained by replicated $\mu_{1\times M}$ across N rows.
Note: Each of the modes of variation $E_{M\times 1}$ is a unit vector in our program, hence projections don't involve division by their magnitude. Also, the program follows an opposite dimension order for E. ($R \times M$ instead of $M \times R$)

Here are the images comparing original and reconstructed values:

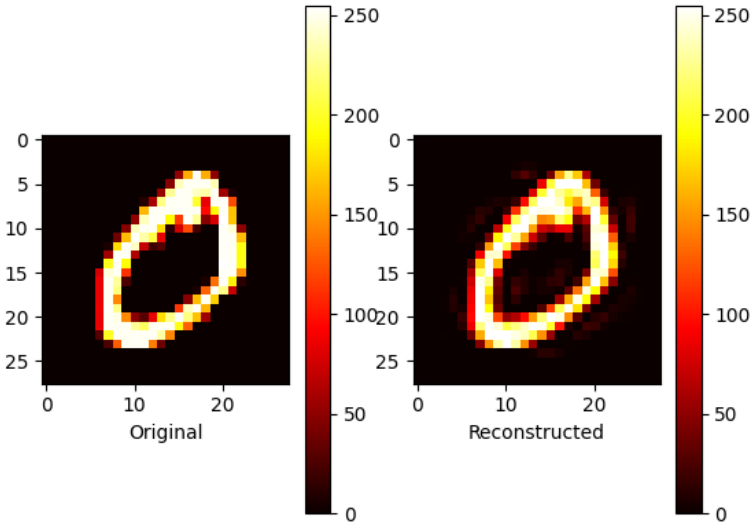

Figure 1: Comparing images for digit 0 data.

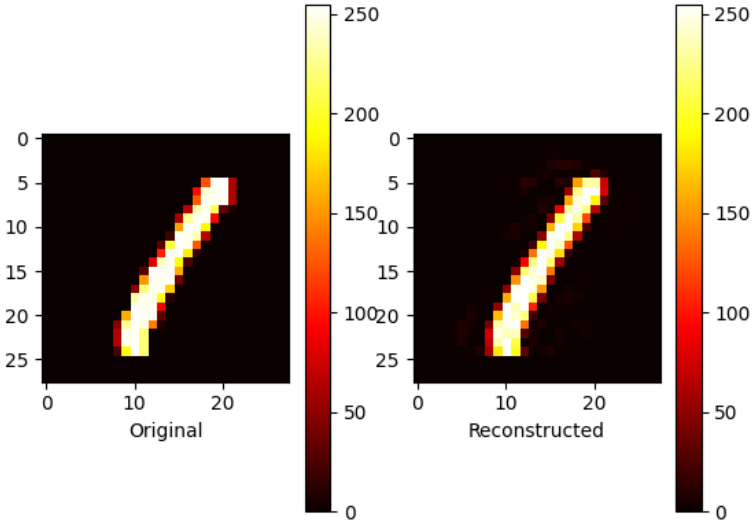

Figure 2: Comparing images for digit 1 data.

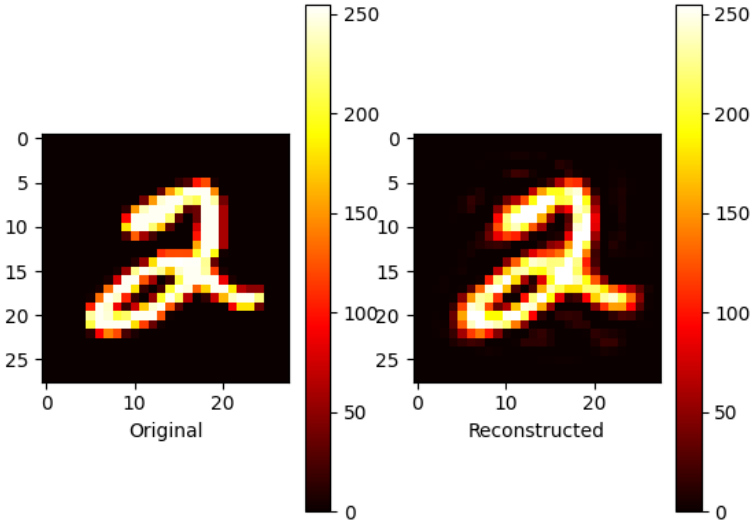

Figure 3: Comparing images for digit 2 data.

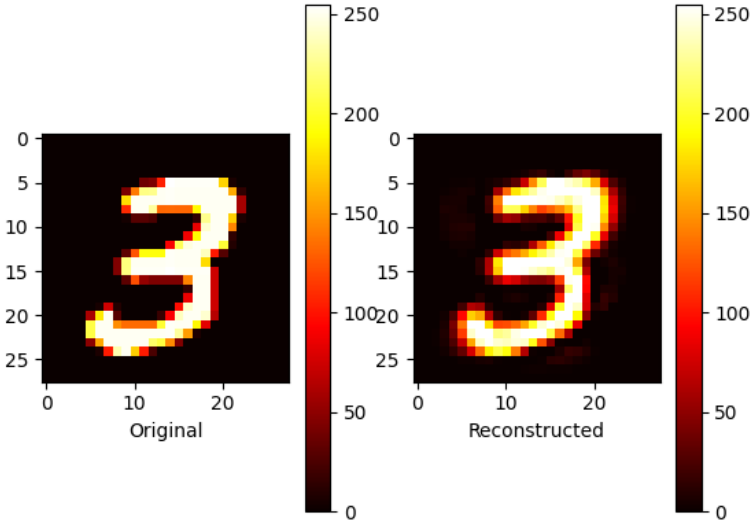

Figure 4: Comparing images for digit 3 data.

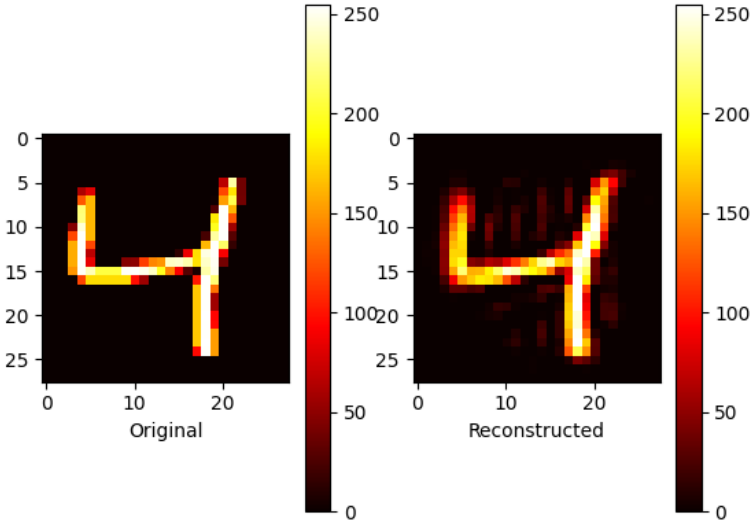

Figure 5: Comparing images for digit 4 data.

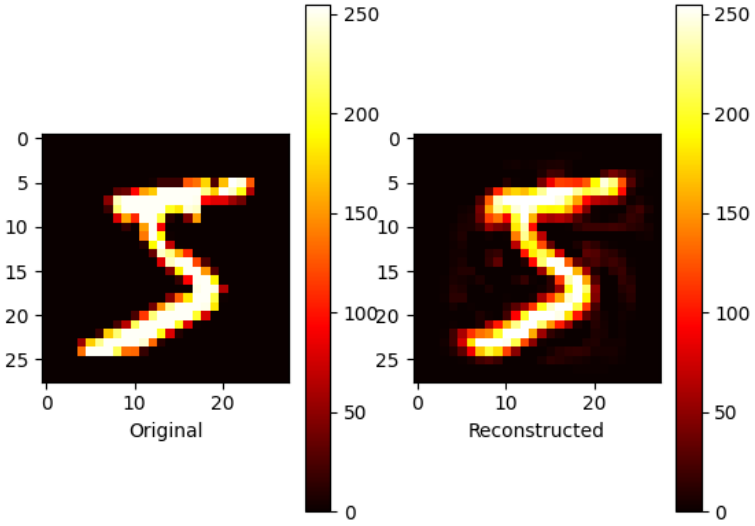

Figure 6: Comparing images for digit 5 data.

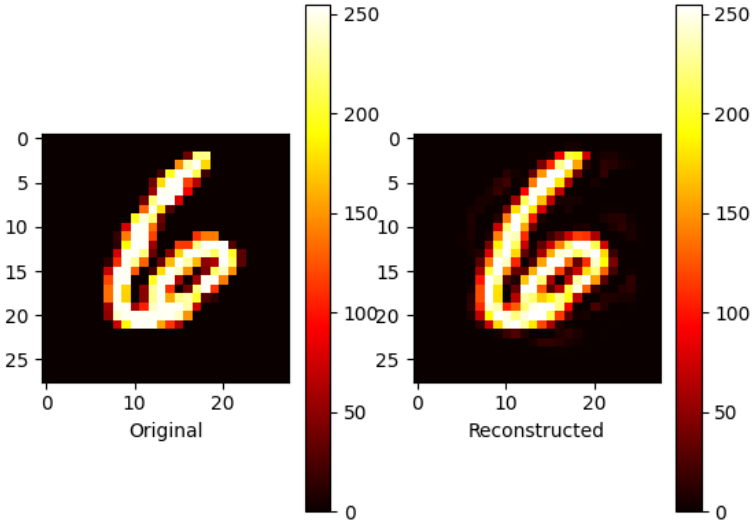

Figure 7: Comparing images for digit 6 data.

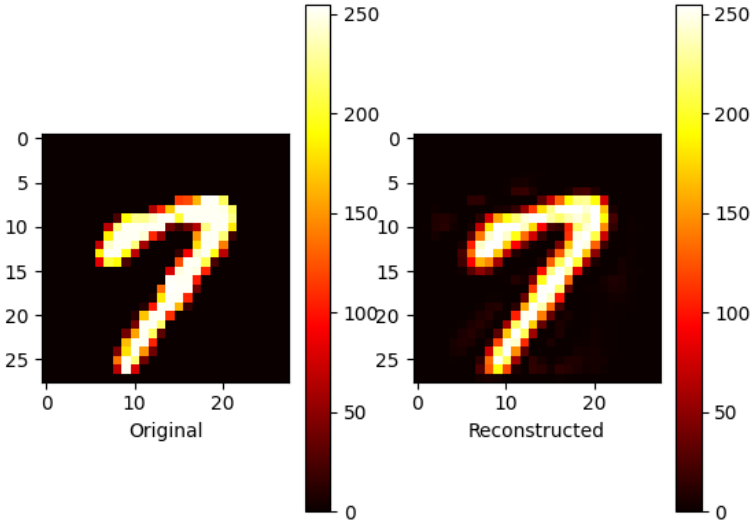

Figure 8: Comparing images for digit 7 data.

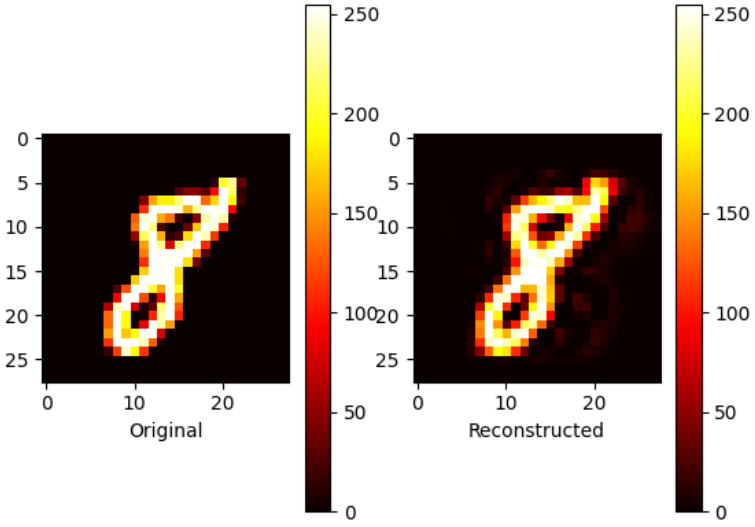

Figure 9: Comparing images for digit 8 data.

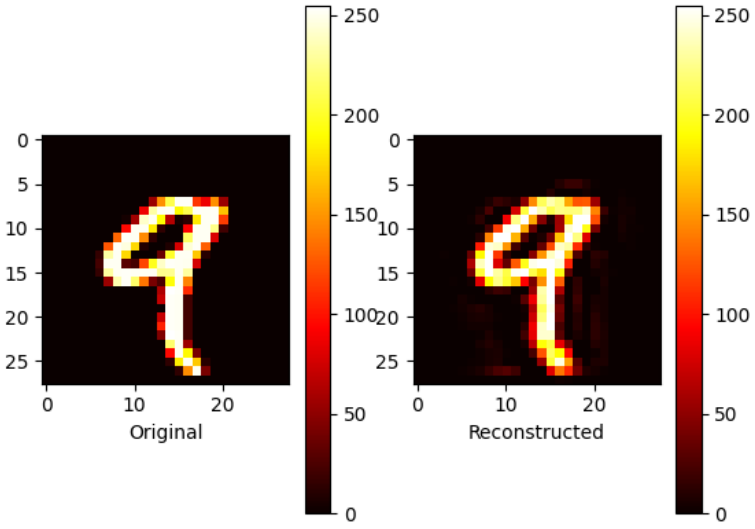

Figure 10: Comparing images for digit 9 data.