Applications of PCA and Image Compression (Lab 1)

Samuil Dichev, mbaxtsd2
The University of Manchester

November 21, 2017

1 Applications of PCA

The iris dataset becomes easier to analyse in lower dimensions without the redundancy. Figure 1 shows clearly linearly separable data, as does Figure 2, which also shows tighter clustering for parts of the data. Not much can be analysed without the labels, but with them one could draw better conclusions from seeing the data represented in the these subspaces.
Figure 3 on the other hand, doesn't reveal much information about the data. There is no clear separation and there is little to no clustering to speak of. From observing the 3 figures, we can clearly see the principal components' variance reduces from first to last, as it should.

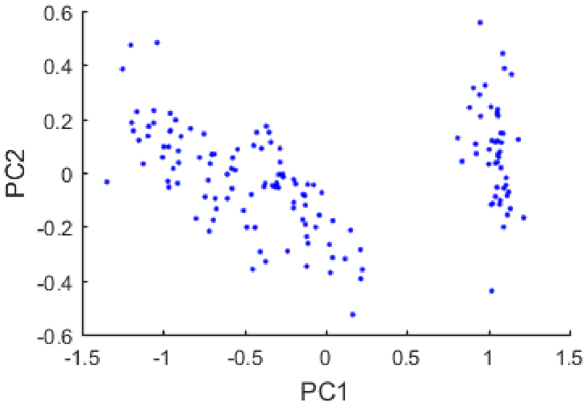

Figure 1: Results in subscape PC_1-PC_2.

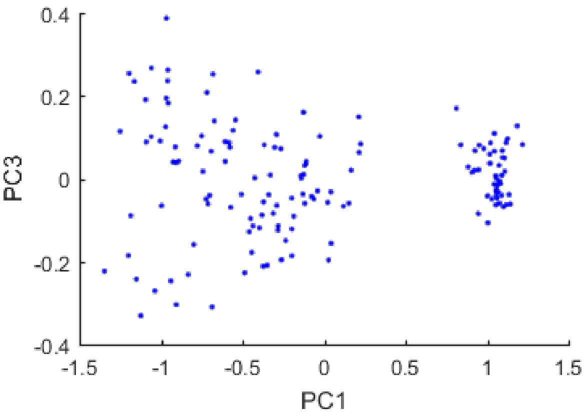

Figure 2: Results in subscape PC_1-PC_3.

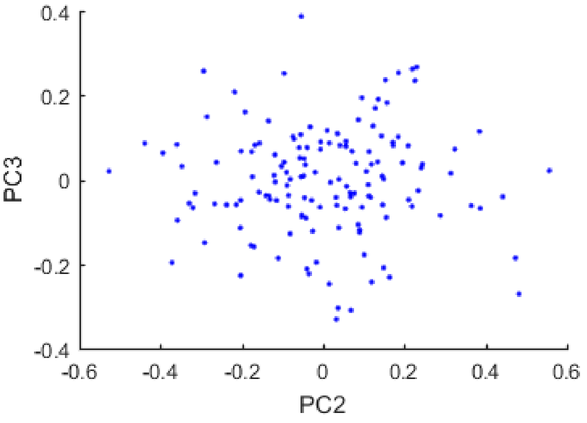

Figure 3: Results in subscape PC_2-PC_3.

2 Image Compression

For the image compression of the digits, I used PCA with SVD because the data has more dimensions (features) than examples. Since our examples are images, each pixel of the images is a feature. At a width by height of 28 by 28, that's 784 pixels or 784 features. The images (examples), however, were only 300 in the train dataset.

For the implementation of this compression system, I designed a function to represent the training data as a matrix of size 784x300, where each column is each 28x28 image represented as a column vector instead of a matrix. This was then passed though the pca2 function, which returned the PCs. The number of PCs to use was chosen based on Proportion of Variance $PoV >= 90\%$, calculated via:

$$PoV = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{d} \lambda_i}$$

This condition is first satisfied at 47 principal components with $PoV = 90.05\%$. By comparison, at $k = 46$ we only get a PoV of 89.77%, so $k = 47$ is the first value at which we achieve a satisfactory $PoV >= 90\%$ as shown in Figure 4.

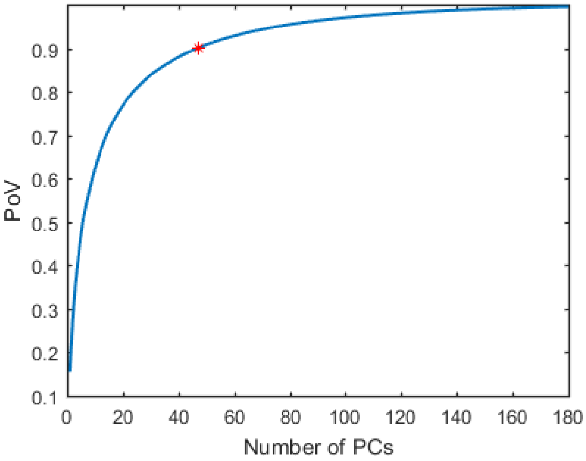

Figure 4: PoV for each k. Marker at $k = 47$.

At 47 principal components, we also observe good size compression on disk. After being saved on disk, the default uncompressed test dataset takes 4985 bytes, while the compressed version is only 3837 bytes, which is a size compression of roughly 23.03%.

Each datapoint is compressed via $z = U_M^T(x - \vec{x})$ and is then reconstructed via $x' = \vec{x} + U_M z$. Finally, the reconstruction error is calculated by summing the unused components' eigenvalues $E = \sum_{i=M+1}^{d} {}_{\lambda_i}$, which results in $E = 3.9419$.

On Figure 5 we can see the original test dataset images, before any compression and on Figure 6 we can see the same images after reconstruction using 47 PCs.

On Figure 7 we see the reconstructed images. The left column contains the images compressed and reconstructed using 1 to 25 PCs and the right column shows the images made with PCs between 26 and 50.

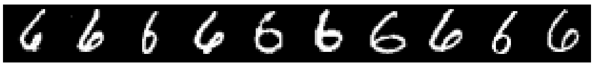

Figure 5: Original uncompressed test images.

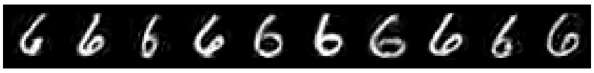

Figure 6: Reconstructed images using 47 PCs.

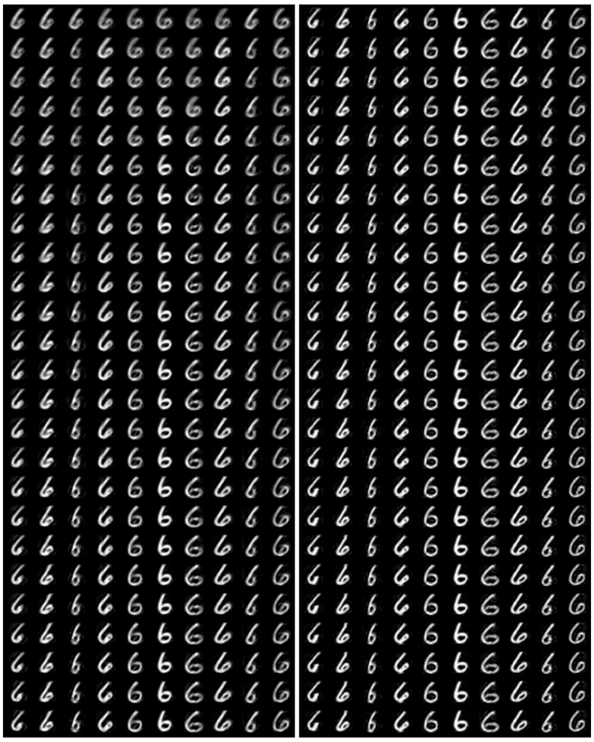

Figure 7: Reconstructed images in order of number of PCs used from 1 to 50.