Pre-Processing Approaches for Dimension Reduction via Singular Value Decomposition

Yuyang Chen, Evan Finken, Maneesh John

MATH:4840 Mathematics of Machine Learning

Prof. David Stewart

December 8, 2023

1 Introduction

The manifold hypothesis posits that real-world datasets, such as natural images, often lie on a low-dimensional manifold within a high-dimensional space. This is the motivation for dimension reduction, which is the problem of representing high-dimensional data using fewer dimensions.

In this project, we explore image pre-processing techniques for improving the effectiveness of dimension reduction via singular value decomposition (SVD). Our overall goal is reducing the number of significant singular values produced by SVD. The motivation is that fewer significant singular values translates to a more efficient representation of the data. We conducted our experiments on a subset of 10,000 images from the MNIST image dataset of hand-drawn digits.

The preprocessing techniques explored in this project include translation of the images to recenter them, rotations of the images to standardize the orientation, stratification of the data by its labels (i.e. applying SVD on each class separately), nonlinear transformations involving thresholding, a nonlinear transformation which weights pixel values by their grid position, and performing k-means clustering by some metric to further cluster the data. In our experiments, we tested each of these preprocessing techniques with SVD, to determine which technique was most effective in reducing the number of significant singular values.

2 Background

2.1 Singular Value Decomposition

Singular value decomposition, or SVD, is a matrix factorization method [4]. Any matrix **M** of dimensions $m \times n$ can be decomposed into three components $\mathbf{M} = \mathbf{U\Sigma V^*}$, where **U** is an $m \times m$ matrix, **V** is an $n \times n$ matrix, and $\mathbf{\Sigma}$ is an $m \times n$ matrix. Note that $\mathbf{V^*}$ refers to the conjugate transpose of **V**. If the matrix consists of strictly real values, as in the case of images, the components can be written as $\mathbf{M} = \mathbf{U\Sigma V^T}$. The original **M** can then be reconstructed as:

$$\mathbf{M} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^{\mathrm{T}} \quad (1)$$

Note that $\mathbf{u}_i$ and $\mathbf{v}_i$ are the columns of **U** and **V**, respectively. $\mathbf{\Sigma}$ is a matrix consisting of non-negative numbers in decreasing order along the diagonal, and zeros elsewhere. The values along the diagonal are the singular values, and the magnitude of each singular value corresponds to its significance. For the purposes of dimension reduction, $\mathbf{\Sigma}$ can be processed to remove smaller singular values, as they contribute less to **M**. For example, singular values very close to 0 are not significant, and can be discarded with very little loss of information in the reconstructed matrix. Thus, it would be valuable for dimension reduction if we preprocess the data so that performing SVD results in fewer significant singular values.

2.2 MNIST Dataset

The MNIST dataset consists of 70,000 images of hand-written digits [1]. Each image is 28×28 pixels, with pixel values ranging from 0 to 255. For our experiments, we decided to use a subset of the first 10,000 MNIST images for simplicity. We also scaled the pixel values to be between 0 and 1. In order to perform SVD, we reshaped the dataset into a 10000×784 matrix.

3 Methods

3.1 Recentering by Bounding Box Center

The MNIST data set is already centered by the image's center of mass [3]. However, we noticed that some images visually appeared to be off-center. This led us to explore more intuitive ways to center the image. We decided to recenter each image by its bounding box, which is the smallest rectangle that inscribes the non-zero values of the image. After identifying the bounding box, we moved the center of the bounding box to the center of the image.

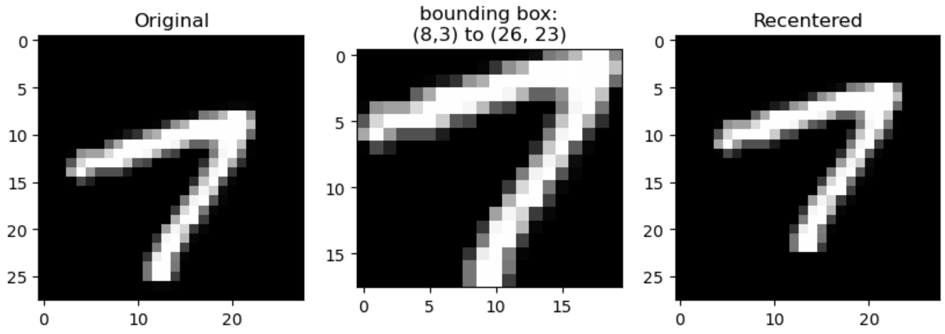

Figure 1: An example MNIST image centered by center of mass (left), the bounding box of the image (middle), and the image after recentering by the bounding box center (right).

Figure 1 shows an example of centering by center of mass versus by the center of the bounding box. After centering by the bounding box, we see that the blank space around the digit is more evenly distributed. This allows for more of the values in the image vectors to be concentrated in the same area, thus reducing variation between digits. We later found that this approach has also been explored in other works, and their results align with ours [3].

3.2 Straightening via Radon Transform

The second approach that we tried was to rotate the images such that the images for each digit were straightened to similar orientations. The goal is to make the non-zero values more concentrated in approximately the same regions of the image for each digit. To accomplish this, we used the Radon transform (also referred to as the sinogram). The Radon transform of a function—in our case, an image—involves taking line integrals over various angles and offsets of the function [5]. We can then find lines in the function using the maximum values of the calculated sinogram. Finally, in order to standardize image orientation, we straightened each image by the opposite of the angle of the best line found by the image's sinogram. Example rotations are shown in Figure 2. Note that our approach was unable to standardize the orientation of the images, which made SVD less effective, as we discuss in our results.

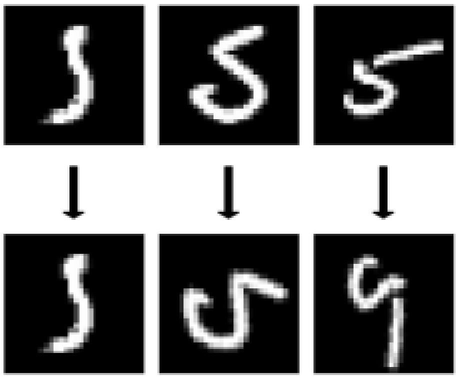

Figure 2: Examples of the digit 5, showing the original images (top) and the results of straightening via the Radon transform (bottom). Note that the straightening does not standardize the orientation.

3.3 Nonlinear Transformations

We also explored multiple nonlinear transforms. Our first approach involved removing unnecessary information from the image vectors, thus reducing the amount of information that would need to be represented by the singular values. The intuition behind this is that we can remove some information from the image without significantly affecting human perception of the digit. We noticed that the overall shape of the digit was nearly unaffected by the removal of pixels with low pixel values. Although some information is lost, as the overall shape of the digit is still recognizable, we are effectively reducing the dimensionality of the data.

We first used a threshold that removes values less than or equal to 0.5 and sets the remaining values to 1. Building upon this approach, we considered additional transformations before thresholding. Namely, we applied the square or square root to the pixel values before applying a threshold. Figure 3 shows an example image before the transformation, and the image after thresholding. We can see that the visual information still present is enough to identify the image as a 7. We also note that the removal of values close to 0 from the image would reduce variation between the digits, as the values are all uniformly 1's or 0's.

Another nonlinear transform we designed was to weight the pixel values by the pixel positions. The formula used for this transform was NewValue = OldValue * (Row/56) * (Column/112). After weighting the pixels, we applied a threshold of 0.25. Figure 4 shows how applying this transform results in the pixels being darker near the top, and brighter near the bottom. Our results show that this was the most effective nonlinear transform.

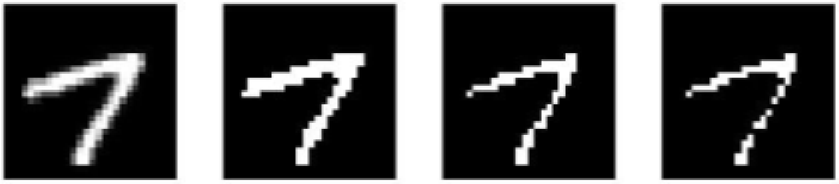

Figure 3: Nonlinear transformations. From left: the original image, the image after a threshold of 0.5, the image after squaring and threshold 0.75, and the image after applying square root and threshold 0.99.

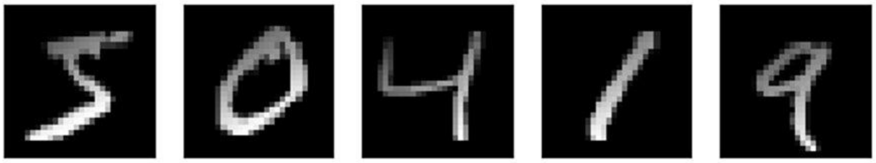

Figure 4: Examples of images with pixel values weighted by the position of the pixels.

3.4 K-Means Clustering

K-means clustering is an algorithm that partitions a dataset into *k* subsets [2]. The motivation behind using this for data preprocessing is that separating the data into clusters may give SVD more information which would reduce the number of singular values necessary to represent the data. We performed k-means clustering on a two-dimensional vector representing some property of the image vector. The two properties used were the coordinates of the center of mass along with the height and width of the bounding box. The cluster that the data point was assigned was then added as an additional dimension, resulting in a vector of length 785.

As mentioned in [3], MNIST data is already centered by center of mass. However, there is slight variation in the centers of mass, and we were curious to see if any useful information exists in that variation. Thus, we recomputed the centers of mass from the image vectors. The coordinates of the centers of mass were then used as data points for k-means clustering. However, as Figure 5 shows, we could not identify any significant clustering. The data points seem to be uniformly distributed along each axis.

We attempted the same clustering approach but instead used the height and width of the bounding box, based on the intuition that the same digits may have roughly the same bounding box. From Figure 5 we see that the algorithm clustered all possible values to one cluster. This implies that this orange cluster includes bounding boxes of any height and width. This approach was ineffective for the same reason as the first, as the clusters offer no useful information.

The method may potentially produce significant results if the clustering is performed on the image vector itself. However, due to the nature of the k-means clustering algorithm (NP hard), running many iterations on such a large dataset would be computationally impractical.

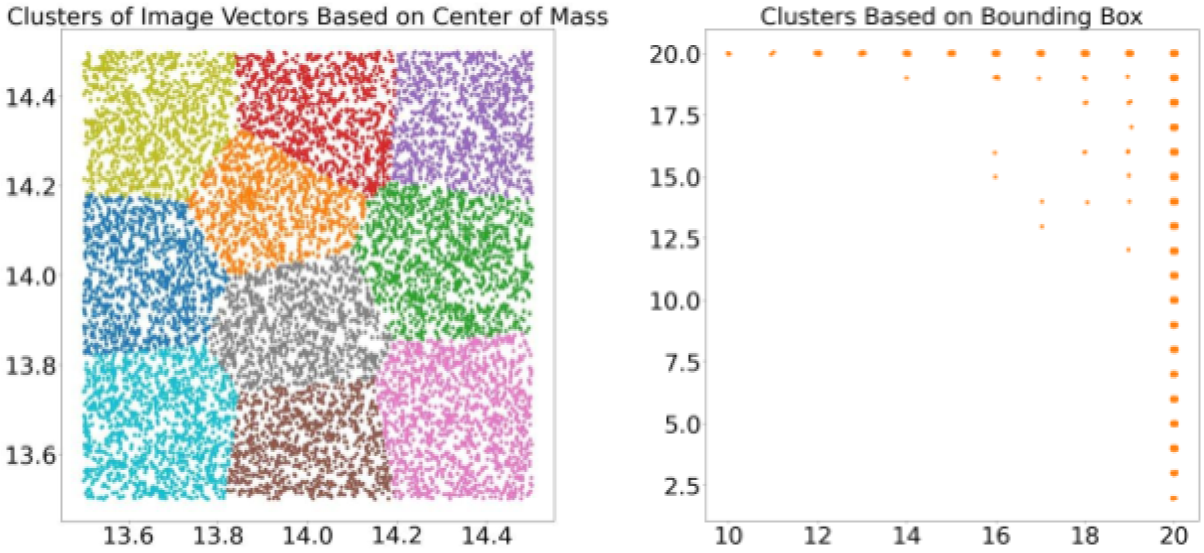

Figure 5: K-means clustering with 10 clusters, applied to the centers of mass of the images (left) and bounding box (right). Bounding box data points have random variation added for visual effect.

3.5 Stratification by Class

As we discuss in our results, we found that recentering was the most effective approach to reduce the number of significant singular values. We attempted to improve these results by stratification of the dataset. We split the recentered data into subsets according to the digit label (i.e. 0s, 1s, etc.), and then applied SVD on each subset. This approach was motivated by the assumption that images of the same digit will generally be more similar-looking than images of different digits.

4 Results

4.1 Comparison of Log of Singular Values

For each preprocessing technique, we apply SVD and plot the log base-10 of the singular values against the index of the singular values. For the original data with no preprocessing, the index where the singular values decrease drastically is around 670, as shown in Figure 6.

Figure 6 demonstrates the effectiveness of recentering. The recentered data was able to achieve around 400 significant singular values, which is roughly a 40% reduction. We attribute the success of the recentering approach to its ability to distribute blank space more evenly around the images. Figure 7 shows that stratification was also effective. However, Figures 6 and 7 show that combining stratification with recentering only offers improvement with images of class label ‘1’. With stratification and recentering, the singular value cutoff for most classes remains around 400, which can already be achieved by recentering the data without stratification.

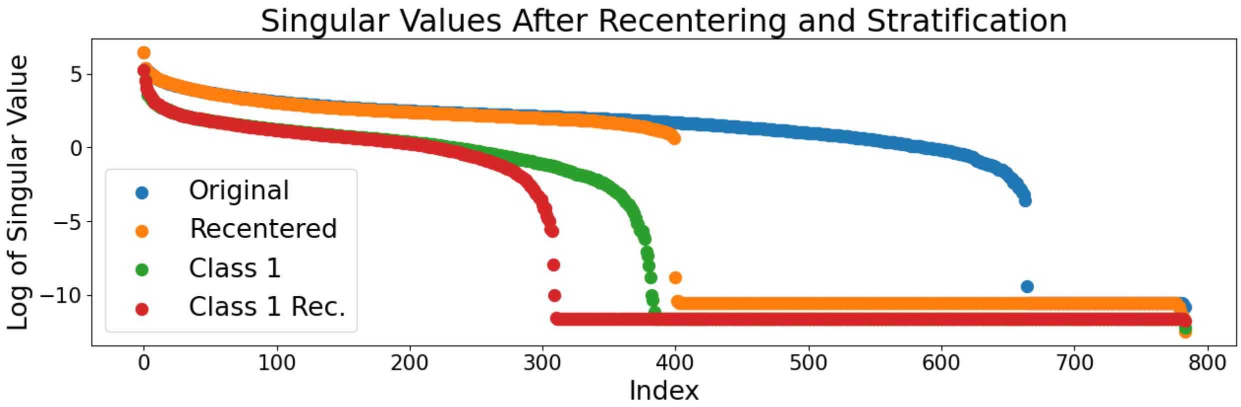

Figure 6: Singular values for the original data, recentered data, data for the digit 1, and recentered data for the digit 1. The indices of the singular values are plotted against the log of the singular values.

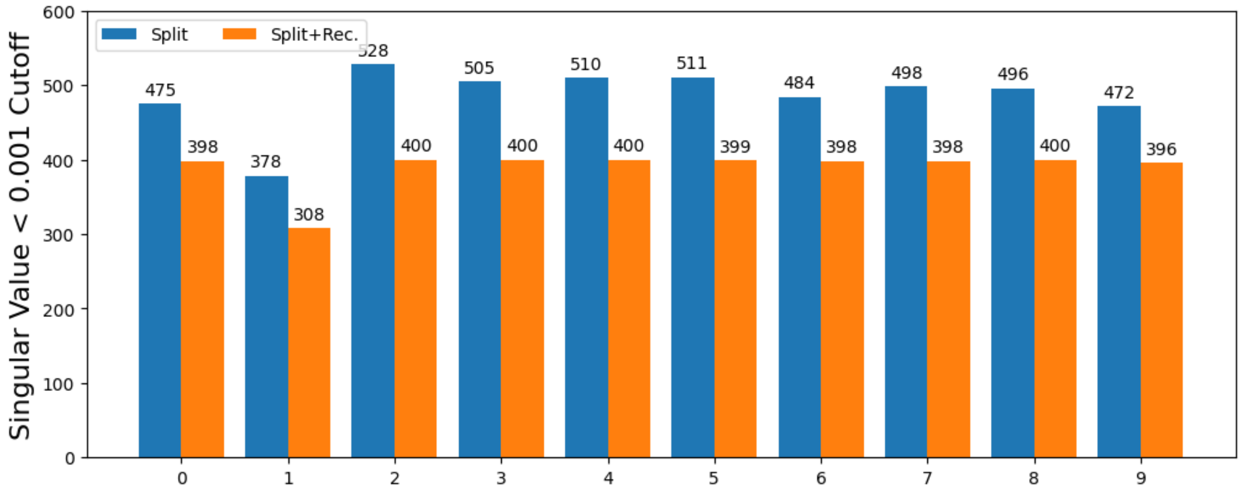

Figure 7: Comparison of stratification by class, and stratification by class with recentering.

With regards to straightening the images via Radon transform, Figure 8 demonstrates that straightening the images increased the cutoff for significant singular values. We believe there could be multiple reasons for this. First, as seen in Figure 2, the transform tends to unpredictably rotate images of the same class due to the variation in how the digits are written. Second, rotating the images requires interpolation as well, therefore some data is lost in the process.

Nonlinear transformations are a close second to the most effective preprocessing method for reducing the number of significant singular values. Figure 9 shows that the thresholding methods improve the cutoff, but are not nearly as effective as the weighting-by-position transform, which decreased the number of significant singular values to approximately 470.

The k-means clustering approaches, as shown in Figure 10, neither increased nor decreased the number of significant singular values. However, they had an unexpected effect of greatly reducing the magnitude of the insignificant singular values. For the other methods, the log of the smallest singular values was approximately -10, whilst k-means clustering decreased that to -30.

However, this offers no advantage, as these singular values were already very close to 0 and would be discarded even without this decrease in significance.

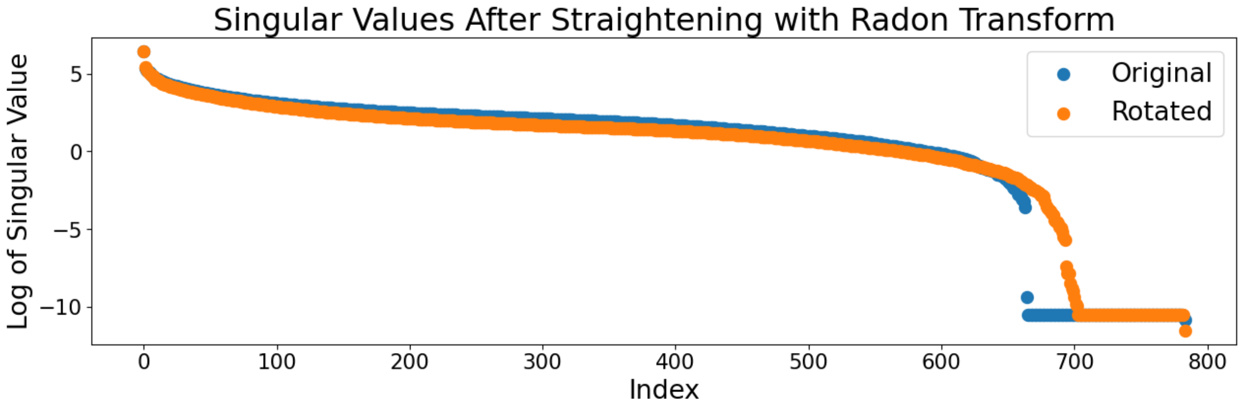

Figure 8: Log of the singular values before and after straightening the data using the Radon transform.

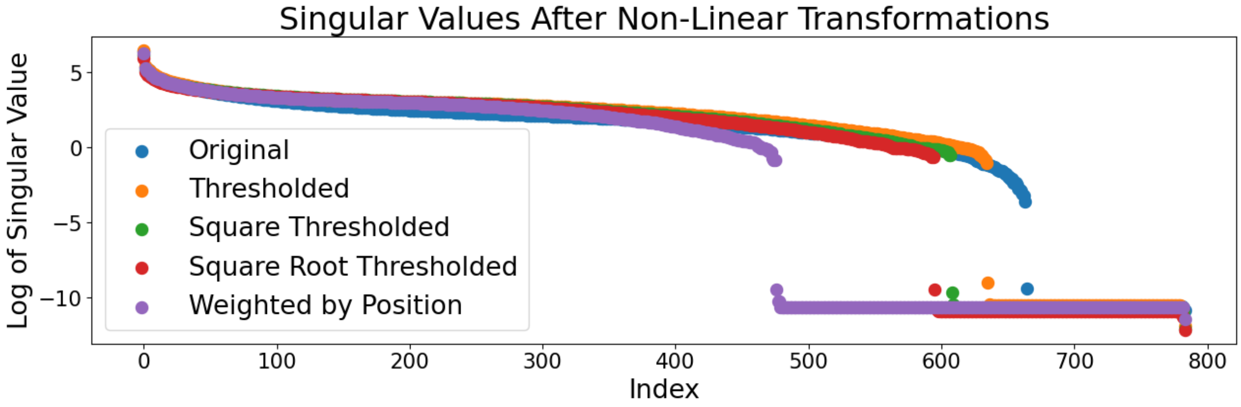

Figure 9: Log of the singular values before and after applying various nonlinear transforms.

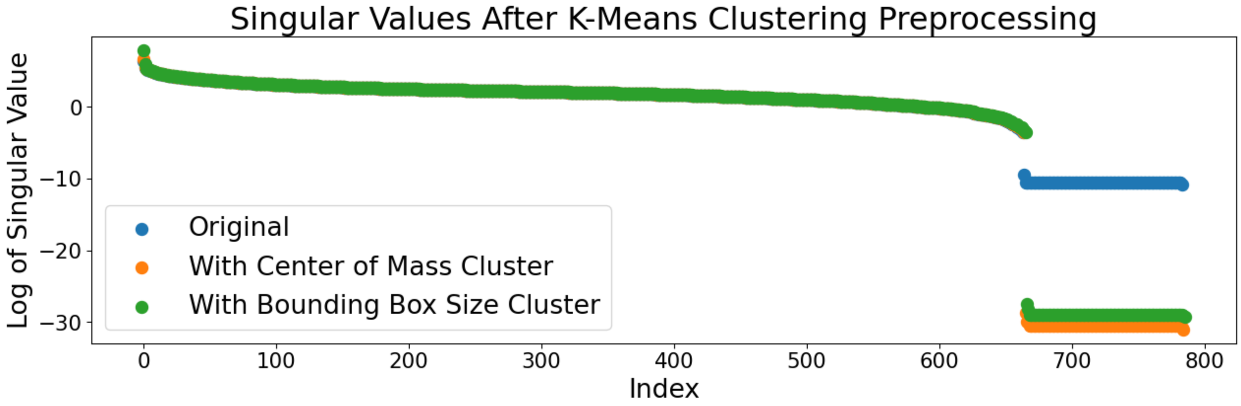

Figure 10: Log of the singular values before and after applying k-means clustering preprocessing.

4.2 Comparison of Image Reconstructions

After applying SVD, we can reconstruct the original matrix with fewer singular values while retaining a reasonable amount of information. Figure 11 shows an example image from the reconstructed matrix with truncated singular values.

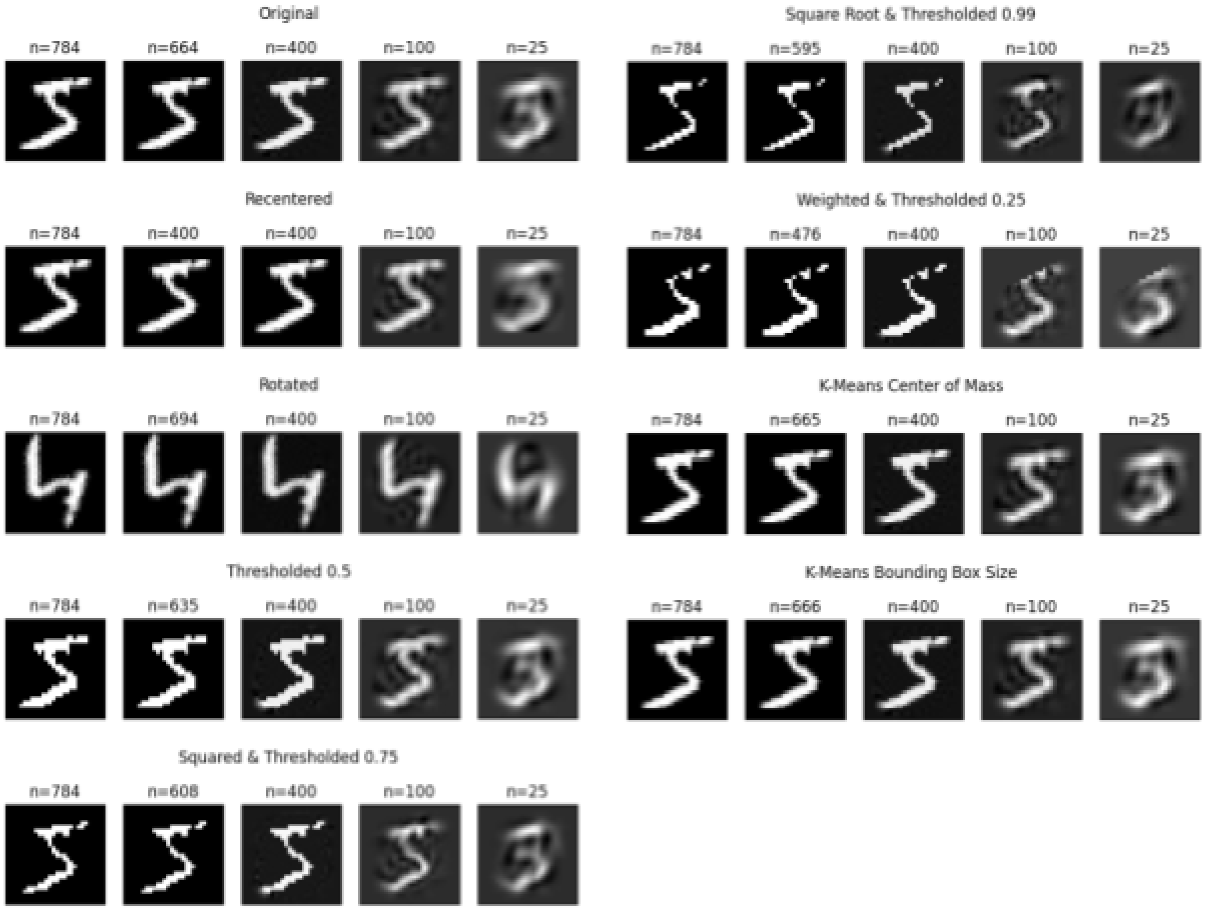

Figure 11: Reconstruction of the image using the first n singular values obtained after different preprocessing methods. n is chosen to be 784 values, the cutoff where the values are less than 0.001, the minimum cutoff among all methods (400), 100 values, and 25 values.

After reconstructing the matrix from the truncated singular values, we can see that after discarding the singular values after the significance drop-off we are able to recreate the image in full with very little loss of information. We note that when discarding all singular values after the significance cutoff, the reconstructed image is nearly identical to the original. However, even when significant values are removed, the reconstruction still appears reasonably similar to the original. Notably, the recentering method yields the best results. At 400 singular values, the recentering method produces an identical reconstruction, while the other methods show some minor degradations in image quality.

5 Conclusion

In this project, we explored multiple methods for reducing the number of dimensions needed to represent the first 10,000 images of the MNIST dataset. We used translation, rotation, nonlinear transformations, and k-means clustering as preprocessing techniques for improving the effectiveness of SVD. The methods which most effectively reduced the number of significant singular values were centering by bounding box, splitting the data by class, weighting pixel values by pixel position, and nonlinear transformations with thresholding of pixel values. The other methods were not effective at reducing the dimensionality of the images.

References

1. LeCun, Y., Cortes, C. and Burges, C.J.C. (1998) The MNIST Database of Handwritten Digits. New York, USA. Retrieved from yann.lecun.com/exdb/mnist/

2. Sharma, S. (2022, September 22). *Coding K-means clustering using Python and NumPy*. DEV Community. Retrieved from https://dev.to/sajal2692/coding-k-means-clustering-using-python-and-numpy-fg1

3. Vanschoren, J. (2014, September 29). *mnist_784*. OpenML. Retrieved from www.openml.org/search?type=data&sort=runs&id=554&status=active

4. Wikipedia contributors. (2023, December 6). Singular value decomposition. In *Wikipedia, The Free Encyclopedia*. Retrieved from en.wikipedia.org/w/index.php?title=Singular_value_decomposition

5. Wikipedia contributors. (2023, September 11). Radon transform. In *Wikipedia, The Free Encyclopedia*. Retrieved from en.wikipedia.org/w/index.php?title=Radon_transform