MAT 167: Applied Linear Algebra

HANDWRITTEN DIGIT CLASSIFICATION USING MATRIX METHODS

December 10, 2019

Name: Patrick Soong
SID: 916220178

Patrick Soong
SID: 916220178 December 10, 2019

Introduction

This paper discusses the use of k-means and singular value decomposition (SVD) for the classification of hand written digits. The dataset used is the USPS.mat file, containing 9298 images of individual handwritten digits 0 to 9. The data was gathered at the Center of Excellence in Document Analysis and Recognition (CEDAR) at SUNY Buffalo as part of a project sponsored by the United States Postal Service.

Dataset: http://www.gaussianprocess.org/gpml/data/

1 Methodology

[5A] The Data Structure

Within the USPS.mat file are four matrices. The _digits matrices contains data for individual handwritten numbers or digits. The data is a raster scan of the 16×16 grey-level pixel intensities normalized to a range of [-1, 1]. The _labels matrices contain the true information about the digit images. Meaning the _labels matrices contain the images that corresponds to each raster scan in the _digits matrices.

Training data is data that is input into the algorithm to help fit the algorithm to the data. Once the algorithm has been fit to the data, the testing data is fed into the algorithm and the algorithm classifies each piece of testing data fed based on what it knows from the training data. In this project, We will use the training_digits and training_labels matrices so the algorithm learns what raster values are most likely to correspond to a specific handwritten digit. We then use the testing_digits and testing_labels matrices to see how our model performs classifying digits it has not seen before.

[5B] k-means (Euclidean Distance)

K-means is a popular and simple algorithm whose objective is to group similar data points together and use them to identify underlying patterns. K-means looks for a centroids, then classifies the data points by their cluster patterns around the centroid. Once the number of centroids has been set the algorithm allocates all data points to the nearest clusters while minimizing the size of the centroids.

We then use Euclidean (i.e. the l_2) distance: $d(x_i, y_j) = ||x_i - y_j||_2$ to compare the distance (error) between the mean digit image and the images in the training and testing datasets.

[5B] The "Mean Digit Image"

Classification becomes challenging without a distinct pattern. It means there are not clear clusters and a lot of variation in the way digits are written. We will be able to tell if there is clear clustering or not by looking at the mean digit image and seeing how diffuse the image is. Mean digits with little diffusion allow us to expect reasonably accurate results when using a distance-based method such as k-means.

To obtain the mean digit image, first, we take the columns of the images and stack them in a vector. We then identify all digits as a vector in $\mathbb{R}^{256}$. To obtain the average raster scans for each digit, sum up the raster information for each digit, then divide by the number of images to get the average raster scans for each digit. We have now obtained the mean digit image for each digit. See **Figure 2** in the **Appendix** for a plot of the 10 mean digit images.

[5C] Singular Value Decomposition (SVD)

The singular value decomposition (SVD) of any $m \times n$ matrix A is given by $A = U\Sigma V^T$ where U and V are $m \times n$ and $n \times n$ matrices, respectively. The diagonal entries of Σ give the singular values of A and are arranged so that $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_n \geq 0$. The columns of U and V are called the left and right singular vectors of A, respectively. The l_2 norm of a matrix is also calculated from the singular values, with $||A||_2 = \sigma_1$, meaning that the l_2 norm corresponds to the first diagonal value in Σ. The singular vectors U and V give the four fundamental subspaces of matrix A.

We can consider a digit image as an $m \times m$ matrix where each entry in said matrix is a pixel value. The columns of each image's matrix are stacked to form a $m^2 \times 1$ sized column vector. Then the stacked images for a single digit are concatenated into the matrix $A_j \in \mathbb{R}^{m^2 \times n}$, with the number of training images for a given digit represented by n and the specific digit as j (j = 0, 1,..., 9). By finding the SVD of A_j, we are also finding the fundamental subspaces of the digit. The left singular vectors form an orthogonal basis of the digit and are referred to as a "singular images". We did not use the singular values or right singular vectors in this project.

SVD-based classification relies on the assumption that an unknown digit can be better approximated in a particular basis of singular images rather than in the bases of other digits. Computing the residuals between the unknown digit and a linear combination of singular images allows us to compare the bases'. The equation for this is given as: $min(||z - \sum_{i=1}^{k} \alpha_i u_i||) = min(||z - U_k\alpha||_2)$ where z is the unknown digit, u_i the singular images of a specific digit, k the number of bases used to approximate the image, and $U_k = (u_1, u_2, ..., u_k)$. Because the columns of U_k are orthogonal, the solution is given by $\alpha = U_K^T z$, meaning the residual is given by $||(I - U_k U_k^T)z||_2$.

Results

[5C] k-means (Euclidean Distance)

The overall classification accuracy using k-means based methods is 84.66%. However, the precision for each individual digit varies over a wide range with a minimum precision of 76.34% when classifying the digit "5" and a maximum of 99.54% for the digit "1". A table of the precision for each digit can be found in **Tables 3** and **Tables 4** in the **Appendix**.

The most difficult digit to identify was "5" while "1" was the easiest to identify. I believe "5" was the most difficult because it shares similarities in appearance to "3" and "8". "8" was the second hardest to identify with a precision of 76.44%. "1" was the easiest to identify because it is the only digit that is a straight line and the only digit that was oriented upright as both "7" and "9" contain a slanted line. For digit appearance reference, see **Figure 1** and **Figure 2** in **Appendix**.

The k-Means algorithm is not very effective for handwritten digit classification purposes. Some digits share similarities in their clustering patterns or are written very poorly. The variations in people's handwriting trigger a significant amount of false positives and negatives. A confusion matrix containing the results of the k-means algorithm can be found in **Table 1** in the **Appendix**.

[5C] Singular Value Decomposition (SVD) [5C]

The overall classification accuracy using SVD-based methods is 96.62%. the precision for each individual digit varies over a range with a minimum precision of 93.35% when classifying the digit "8" and a maximum of 99.85% for the digit "1". A table of the precision for each digit can be found in **Tables 3** and **Tables 4** in the **Appendix**.

The most difficult digit to identify was "8" while "1" was the easiest to identify. Interestingly, the most common false positive for "8" was "1". This is likely due to the digits sharing three points along a central line, marking the top-most, middle, and bottom-most points of the "digit". "3" and "9" both also had the next two highest amount of false positives for "8" and share the same similarity. For digit appearance reference, see **Figure 1** and **Figure 2** in **Appendix**.

SVD-based algorithms are much more effective for handwritten digit classification. However our algorithm still may not be as precise as needed for use in industry. A confusion matrix containing the results of the k-means algorithm can be found in **Table 2** in the **Appendix**.

[5D] Conclusion

In this project, we compared the performance of a SVD-based algorithm and a k-means based algorithm for handwritten digit classification. K-means methods use the mean digit images to determine the unknown digit by using the smallest Euclidean distance (error). SVD-based methods find the SVD for each unknown digit and use the resulting bases to determine the identity of the unknown digits. SVD is more accurate than k-means because we are approximating from the bases of the matrix instead of using the distance between two points. SVD out-performed k-means for each digit and in overall accuracy as the classification accuracy for k-means and SVD-based methods is 84.66% and 96.62%, respectively.

Appendix

Figures

Figure 1: The first 16 images in the training_digits matrix in USPS.mat

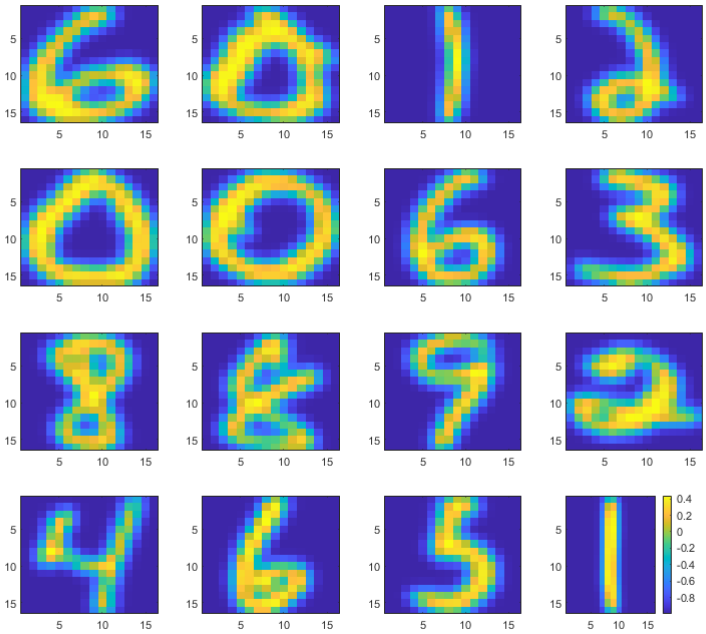
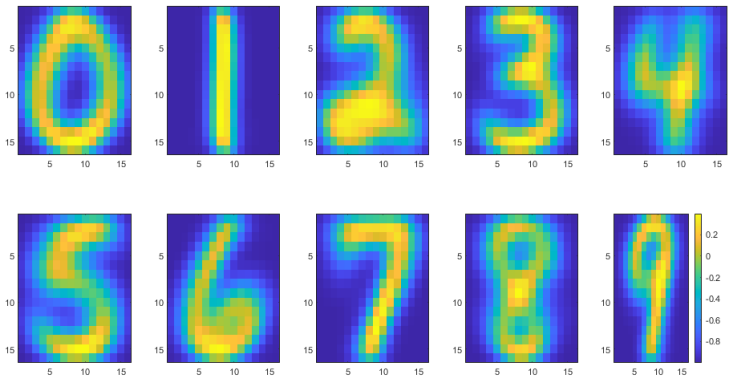


Figure 2: Images of the mean values of the 10 digits from training_averages



Tables

Note: For Tables 1 and 2, the columns contain the predicted result and rows the actual result. The precision rates in Tables 3 and 4 were calculated by dividing the confusion matrices' diagonal value for the row by the sum of that row.

Table 1: Confusion Matrix (k-means) — Accuracy = 84.66%

	0	1	2	3	4	5	6	7	8	9
0	656	1	3	4	10	19	73	2	17	1
1	0	646	0	1	0	0	1	0	1	0
2	14	4	362	13	25	5	4	9	18	0
3	1	3	4	368	1	17	0	3	14	7
4	3	16	6	0	363	1	8	1	5	4
5	13	3	3	20	14	271	9	0	16	6
6	23	11	13	0	9	3	354	0	1	0
7	0	5	1	0	7	1	0	351	3	34
8	9	19	5	12	6	6	0	1	253	20
9	1	15	0	1	39	2	0	24	3	314

Table 2: Confusion Matrix (SVD) — Accuracy = 96.62%

	0	1	2	3	4	5	6	7	8	9
0	772	2	1	3	1	1	2	1	3	0
1	0	646	0	0	0	0	0	0	0	1
2	3	6	431	6	0	3	1	2	2	0
3	1	1	4	401	0	7	0	0	4	0
4	2	8	1	0	424	1	1	5	0	1
5	2	0	0	5	2	335	7	1	1	2
6	6	4	0	0	2	3	399	0	0	0
7	0	2	0	0	2	0	0	387	0	11
8	2	9	1	5	1	1	0	0	309	3
9	0	5	0	1	0	0	0	4	1	388

Table 3: Digit Classification Precision (0-4)

	0	1	2	3	4
k-means	83.46%	99.54%	79.74%	88.04%	89.19%
SVD	98.22%	99.85%	94.93%	95.93%	95.71%

Table 4. Digit Classification Precision (5-9)

	5	6	7	8	9
k-means	76.34%	85.51%	87.31%	76.44%	78.69%
SVD	94.37%	96.38%	96.27%	93.35%	97.24%

Code

Please see Digit_Recognition_ppsoong.m