Machine Learning: Handwritten Digit Recognition Using Singular Value Decomposition

Webster Gordon, Justin Nichols, Mollee Swift

Overseen by: Dr. Frederic Marazzato & Dr. Peter Wolenski
Department of Mathematics, Louisiana State University, Spring 2022

Introduction

Machine learning is a computer procedure in which an algorithm improves its accuracy through the process of iterative data analysis. Machine learning algorithms build a model on a given dataset (training data) to make predictions on an unseen dataset (testing data). Singular Value Decomposition (SVD) is an algebraic method that simplifies high-dimensional data for use in these algorithms. In our analysis, we used Singular Value Decomposition to reduce the dimensionality of our data and used Linear Least Squares as our classification method.

Data Set & Objective

Using handwritten digits from the United States Postal Service, we plan to use SVD to train our model to learn the numbers 0-9. The goal is to then give our model a new data sample containing an unseen handwritten digit and test if the model can accurately predict the value of the digit. We decided to use SVD over Principal Component Analysis (PCA), because SVD is more efficient than PCA when working with dense data.

Figure 1: *Original image samples of each digit from dataset.*

Our dataset comes from the USPS.[1] The dataset contains 7291 training and 2007 test images in h5 format, represented as 16 by 16 matrices of 8-bit (0-255) grayscale values. Our training set represents approximately 80 percent of the data, and the testing set is around 20 percent.

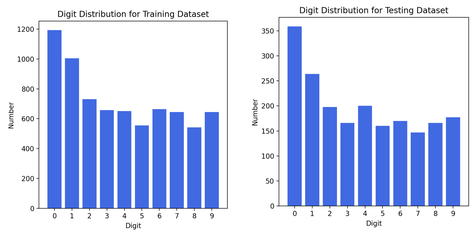

Figure 2 & 3: *Digit distribution for training and testing dataset, respectively.*

We organized the data into different digit sets and performed SVD on each digit set and computed the rank-12 matrix approximations (see explanation in methods).

Figure 4: *Rank-12 matrix approximations images for each digit.*

1 - https://www.kaggle.com/bistaumanga/usps-dataset

Methods

Singular Value Decomposition is the factorization of a rectangular matrix into three components:

$$A = U * \Sigma * V^T$$

Where Σ is a diagonalized matrix containing singular values in descending order, U is an orthogonal matrix composed of the left eigenvectors, and V^T is an orthogonal matrix composed of the right eigenvectors.

The matrices A, which we decomposed with SVD, were the result of grouping the 16x16 matrices from the training set by their label (0-9), converting to 256x1 row vectors, and stacking them into an Nx256 matrix, where N = number of data points with each label.

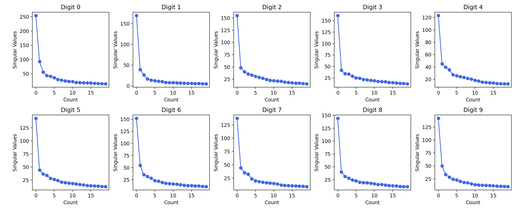

Figure 5: *Plot of first 20 singular values for each digit, 0-9.*

After reviewing the singular value plots, we sought to reduce the dimensionality of our data without sacrificing accuracy. Based on the plots, we conjectured that our desired peak accuracy would be within the first 20 approximations. We used Linear Least Squares as a classification model by finding the minimum of the following quantity across the 10-digit classes:

$$b = \frac{\|(I - U_k U_k^T) X_i\|}{\|X_i\|}$$

Where I denotes the identity matrix, X_i denotes the i-th image in the test set, U_k denotes the orthonormal matrix of eigenvectors for digit k obtained by SVD, and the norm used is the usual Frobenius norm. We calculated this classifier for all images in the test set and for the increasing rank approximations. We then computed the percentage that were properly classified and found that the accuracy peaked at rank-12.

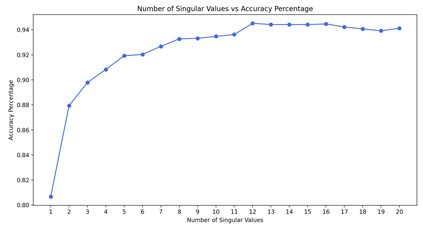

Figure 6: *Plot comparing the classification accuracy percentage to the rank of the singular value approximations.*

Results

Technologies and packages used: Python using the Anaconda distribution, SciPy, Scikit-Learn, Numpy, matplotlib, and h5py.

We applied the classification model, Linear Least Squares, to achieve an accuracy of 94.52% on the prediction of digit values from the testing dataset. Thus, out of 2007 test values, 1897 were classified correctly.

The individual success rate of each digit differs in accordance with the complexity of the digit, along with the cleanliness of the rank-12 matrix approximations. The similarity of a digit's characteristics to another digit also impacted its success rate.

Digit 0, while having the largest population in the dataset, also performed the best with an accuracy of 98.60%. However, digit 3 had the lowest accuracy of 86.70%.

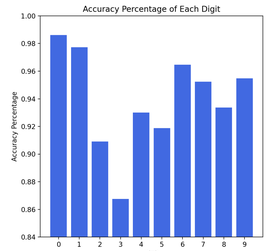

Figure 7: *Plot of accuracy values for each digit.*

There were a total of the 110 misclassified digits when using the rank-12 matrix approximation. When comparing the correct digit to the predicted values, similar patterns in the misclassified digits were apparent. For instance, digits 3, 5, and 8 were often misclassified as one another as well as digits 1 and 7. Also, many of the misclassifications were a result of misplacement and skewedness.

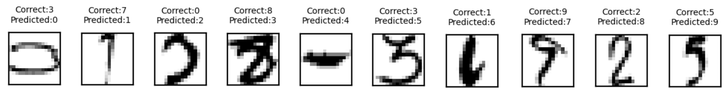

Figure 7: *10 Misclassified digits.*

Conclusion

Our machine learning model of SVD and Linear Least Squares was able to successfully predict handwritten digits with a high accuracy (>94%). Ultimately, the model can be used in a variety of applications, such as quickly interpreting postal codes and addresses on mail and packages. In general, machine learning algorithms such as ours can be used to recognize any handwritten digits, like financial documents, mathematical writing, etc.

Additionally, the model can be augmented to learn not only digits, but also the modern Latin alphabet, or any language's alphabet, for purposes of digitizing handwritten text.

We would like to extend our gratitude to Dr. Frederic Marazzato for his assistance.