ECE 532 Final Project Update #2
Owen Seymour, Section 002
November 30, 2020

Project GitHub page: https://github.com/oseymour/ECE_532_Final_Project.git

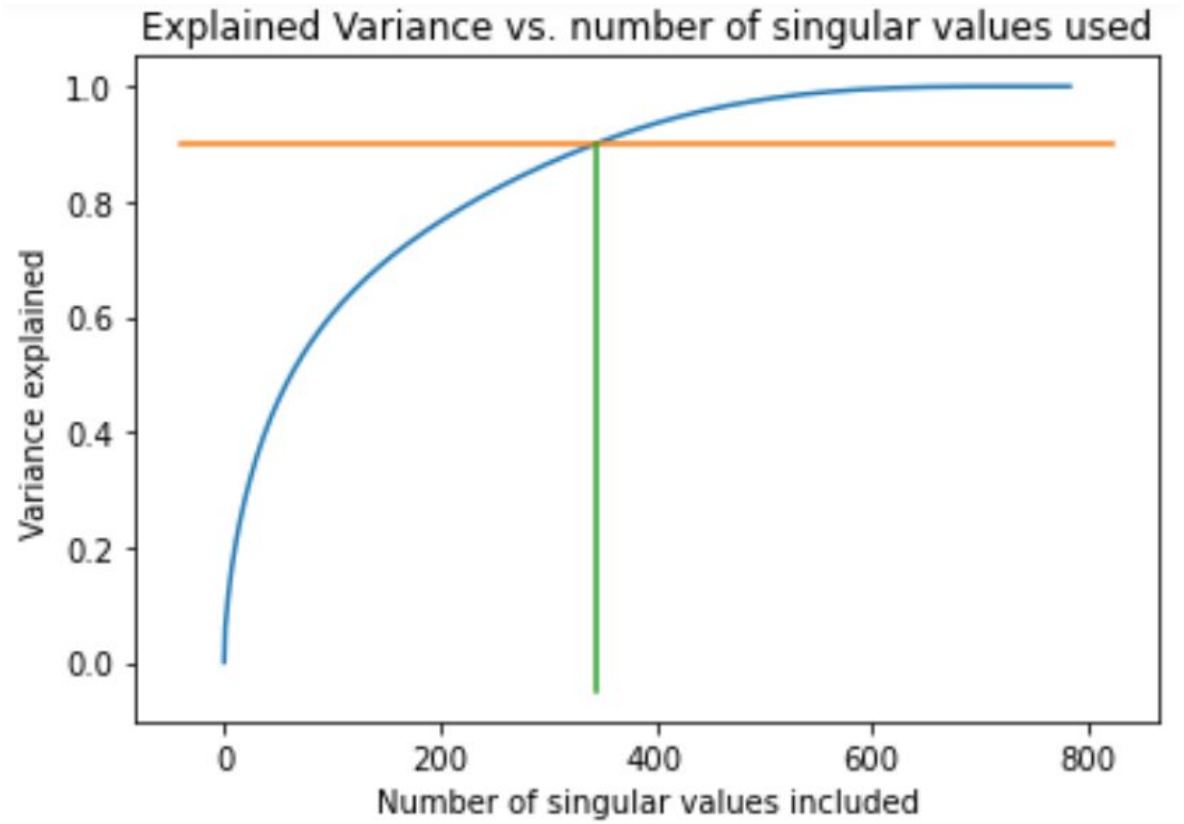
The linear regression algorithm has been developed since the last update and tested on unmodified test images, images formed via a truncated SVD method, two-feature images, and three-feature images. During training to find weights, ridge regularization was used for all data formats. Results are shown in the table below.

Data format	Training accuracy	Testing accuracy
Unmodified	7%	7%
Truncated via SVD	85%	85%
Two-feature	23%	24%
Three-feature	34%	35%

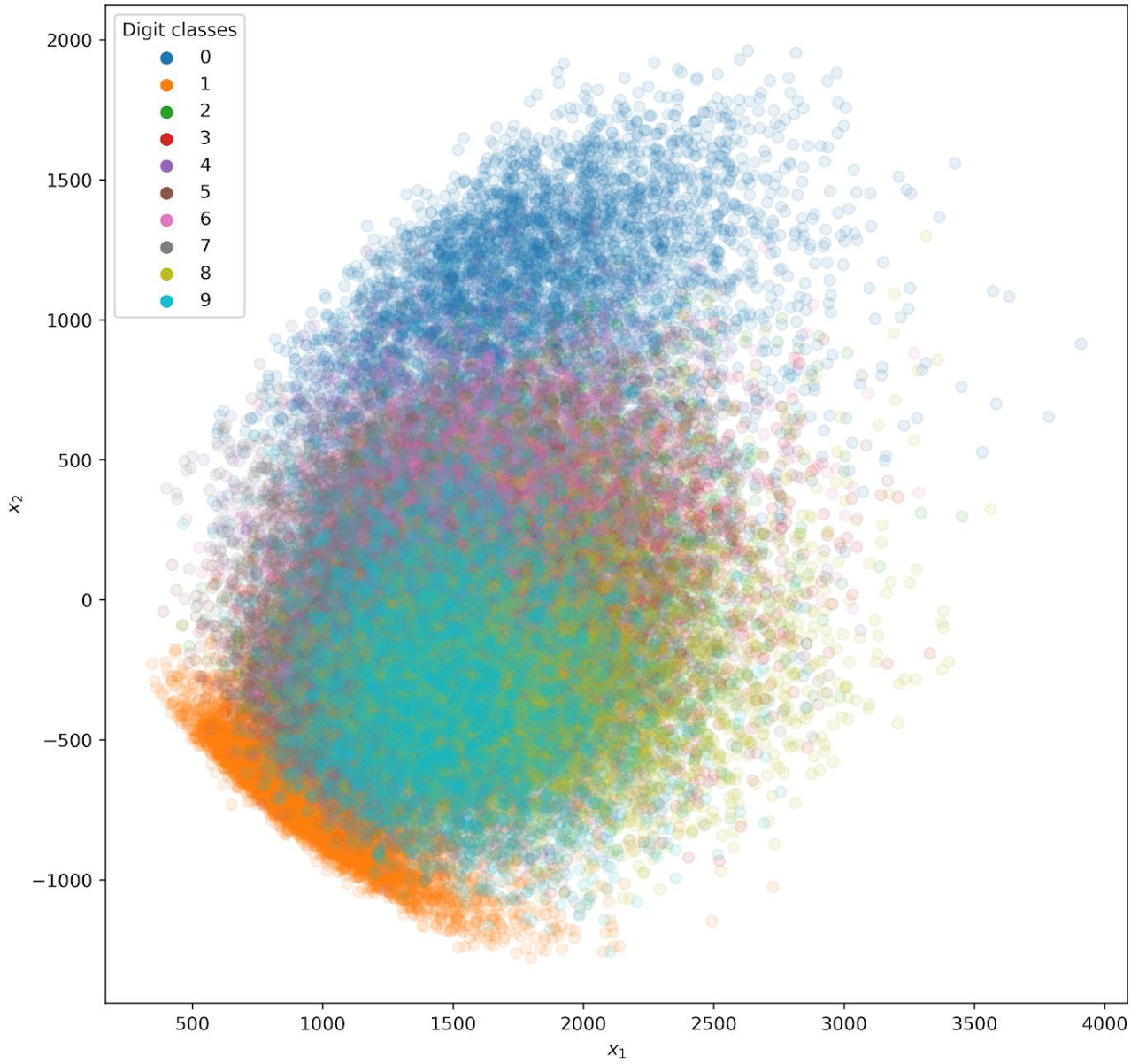
And here is another table showing test data precision by digit for each data format.

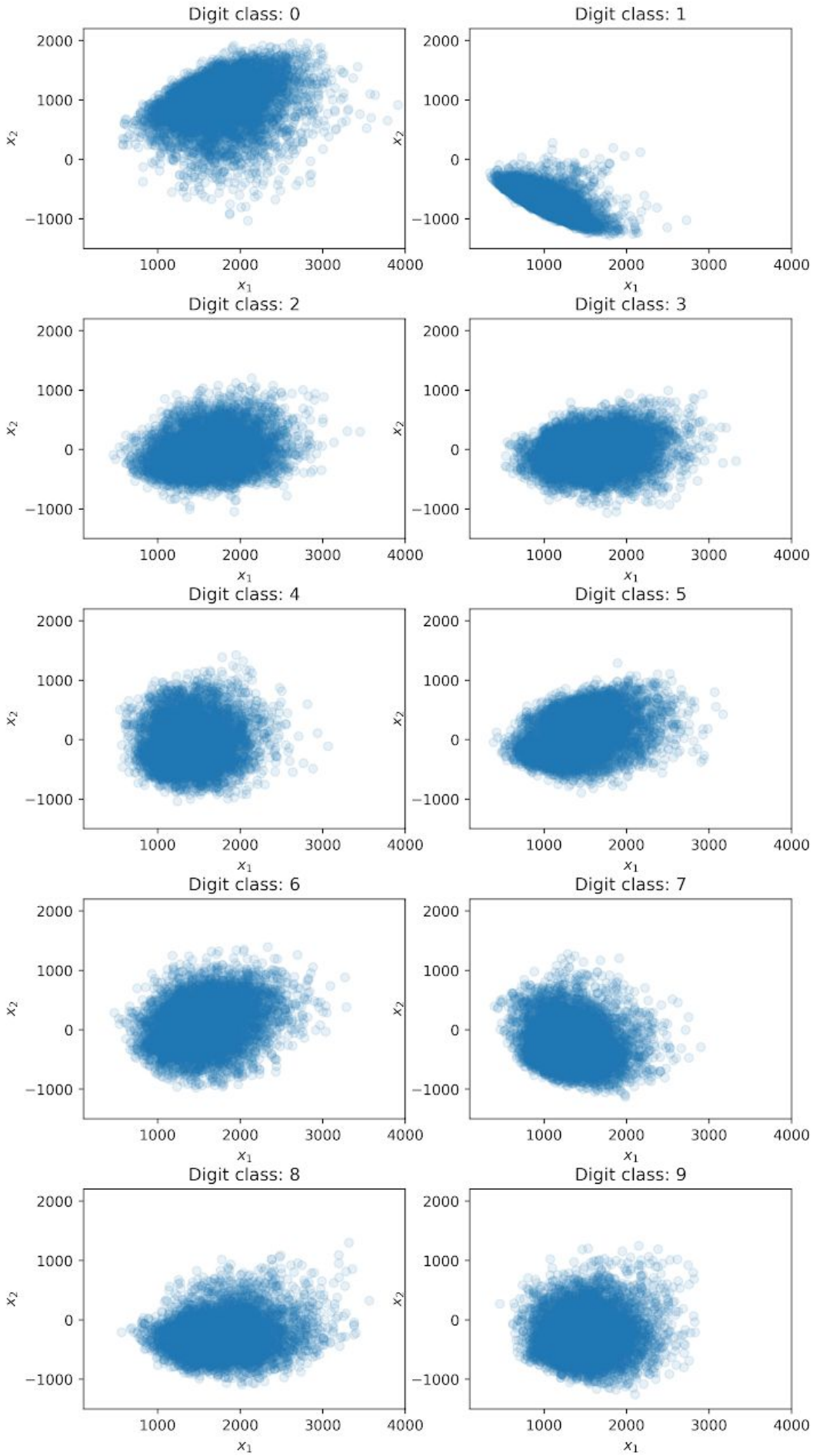
Digit	Unmodified	Truncated via SVD	Two-feature	Three-feature
0	6%	90%	24%	34%
1	0%	83%	26%	37%
2	12%	91%	0%	0%
3	7%	84%	0%	47%
4	15%	84%	0%	42%
5	0%	87%	0%	0%
6	1%	89%	0%	0%
7	1%	86%	0%	32%
8	3%	80%	16%	36%
9	10%	82%	0%	4%

The unmodified data was as it sounds, each image was simply a vectorized numpy array. The data truncated by the SVD was formed by using the first 345 singular values in the SVD of the unmodified training images. By using 340 singular values, the truncated SVD images were able to account for 90% of the variance in the unmodified training images, as shown by the figure below. On the horizontal axis is the number of singular values and on the vertical axis is the variation explained by that number of singular values. The explained variance was calculated by normalizing the singular values relative to the sum of all singular values and then summing the number of singular values on the horizontal axis. The horizontal orange line lies at 0.9 explained variance and the vertical green lines lies at 345 singular values.



The two-feature and three-feature data were primarily made as a way of trying to visualize the separation of digit classes and formed by multiplying the unmodified images by the appropriate numbers of columns in V, from the SVD of the unmodified data, since the vectorized images were arranged in rows. The 2-D plots of two-feature images are shown below, color-coded by digit class in the first and split up into subplots by digit in the second. As we can see, most of the digits are very closely clustered to each other while others are clearly separable from the rest (0 and 1).


Digit classes
0
1
2
3
4
5
6
7
8
9
2000
1500
1000
500
0
−500
−1000
x_2
500
1000
1500
2000
2500
3000
3500
4000
x_1


Digit class: 0
Digit class: 1
Digit class: 2
Digit class: 3
Digit class: 4
Digit class: 5
Digit class: 6
Digit class: 7
Digit class: 8
Digit class: 9
2000
1000
0
−1000
1000
2000
3000
4000
x_1
x_2

Remaining goals for the project include:

- Allow for other norms to be used in the K-means algorithm besides the Frobenius norm. The one-norm and two-norm will be additional options.
- Develop, test, and refine the neural net.