ECE 532 Final Project Proposal
Owen Seymour, Section 002
October 22, 2020

In this project, the publicly available MNIST dataset (http://yann.lecun.com/exdb/mnist/) will be used to train and test three classification algorithms learned in the Fall 2020 semester of ECE 532 Matrix Methods in Machine Learning at the University of Wisconsin - Madison. The algorithms developed will be: 1) K-Nearest Neighbors, 2) truncated SVD, and 3) a neural network. The final report will be available by December 12th, 2020 at the latest, in addition to two updates on November 17th, 2020 and December 1st, 2020. All progress, code, and results will be available on a GitHub page (https://github.com/oseymour/ECE_532_Final_Project.git). All algorithms will be developed in Python.

The MNIST dataset is an open-source dataset of digitized handwritten images, zero through nine. There are 60,000 training images and 10,000 test images, with all of them being 28 pixels by 28 pixels. All of the images in the training and the test data are labeled, as well. The dataset and more information are available at http://yann.lecun.com/exdb/mnist/. The three algorithms proposed in the introduction will be developed exclusively with the training data and modified based on results from cross-validation with the training data before being run on the test images for the final results, which will be given in the final report.

The three algorithms being proposed for development are K-Nearest Neighbors (KNN), truncated SVD, and a neural network. All of these algorithms, except KNN, were taught in the ECE 532 Matrix Methods in Machine Learning course at the University of Wisconsin-Madison during the Fall 2020 semester.

For KNN, the number of nearest-neighbors, k, will be changed and the results compared via cross-validation with the training data to find the best value. The value for k that results in the lowest cross-validation error is the value that will be used when the algorithm is run on the test images. Also, various methods of calculating the distance between the two images will be tested. Possible methods for determining the distance could include the one-norm, the two-norm, the Frobenius norm, etc.

The second algorithm, truncated SVD, will technically be a collection of ten classifiers since a decision boundary will be needed for each of the ten possible classes. In the ideal scenario, only one of the ten classifiers will classify an image as a number, and the other nine will classify it as *not* the class associated with that decision boundary. In the case that more than one classifier labels the image as a number, the result from the KNN algorithm will be used to decide the class of the image. The most important parameter to modify for this algorithm will be the regularization parameter, i.e. the rank of the approximation of the image that is used. It will be possible for each classifier within the truncated SVD algorithm to use a different rank approximation. Again, the regularization parameter will be chosen via cross-validation on subsets of the training data before being applied to the test data.

For the final algorithm, the neural network, the easiest parameter to change will be the number of layers. Networks for two through five layers will all be tested via cross-validation with the training data and the most accurate layer count will be used in the final testing phase, with the test data.

The deadline for submission of the final project report is December 12, 2020 and the report will be available on the GitHub page (https://github.com/oseymour/ECE_532_Final_Project.git) by this date.

In the interim, two project updates will be published, one by November 17th, 2020 and another by December 1st, 2020. Both updates will describe what progress has been made since the last update as well as goals for what will be accomplished by the next update. These updates will also be available on the project's GitHub page. In addition, reviews of two final reports written by peers in the course will be completed by December 17th, 2020. These will not be posted on the GitHub page in order to protect the authors' intellectual property.