# CS 180 MP 4-5 Analysis

Face Recognition using ANN Model

Using a two layer neural network model, with the number of hidden nodes equal to half of the input nodes, the ANN model got an accuracy of 100% for the training set, 95.0% for the test set. Training time took around 3.21667 minutes.

When training the model with varying number of hidden layer nodes, the performance of the model, tends to fluctuate within 90-95% for the test set. As the number of nodes increase the performance tends to drop (although only by a fraction) but still yield a decent accuracies as seen in the hidden nodes vs accuracy graph below.
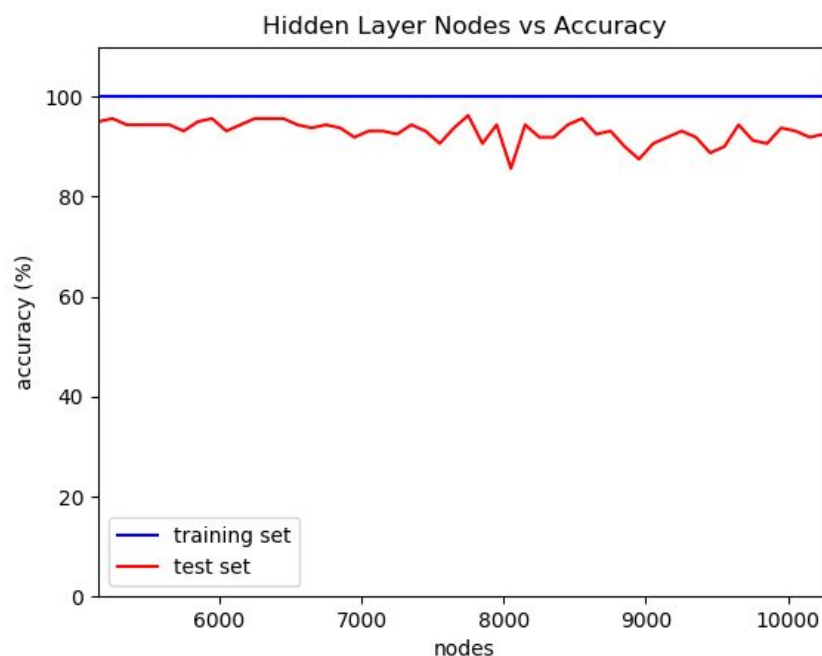

Using PCA and transforming the training data with respect to the 10 eigenvectors with largest eigenvalues and the number of hidden nodes is equal to half of the number of pixels of the image, the training time was faster, but the accuracy of the model decreased to 5.0% for the training set, 4.375% for the test set. When the number of hidden nodes is set to 10, which is the number of features of the training set when PCA is applied, the accuracy of the model dropped to 37.0833% for the training set, 35.0% for the test set.

Face Recognition using SVM Model

For the SVM model, with default parameters, the model got an accuracy of 100% for the training set, 95.625% for the test set. Training time took around 1 second (1.057 s)

Using a polynomial kernel with varying degrees from 1 to 5 for each run, the performance of the model is described by the degree vs accuracy plot below. Based on the graph, the accuracy of the model drops when the degree is increased.
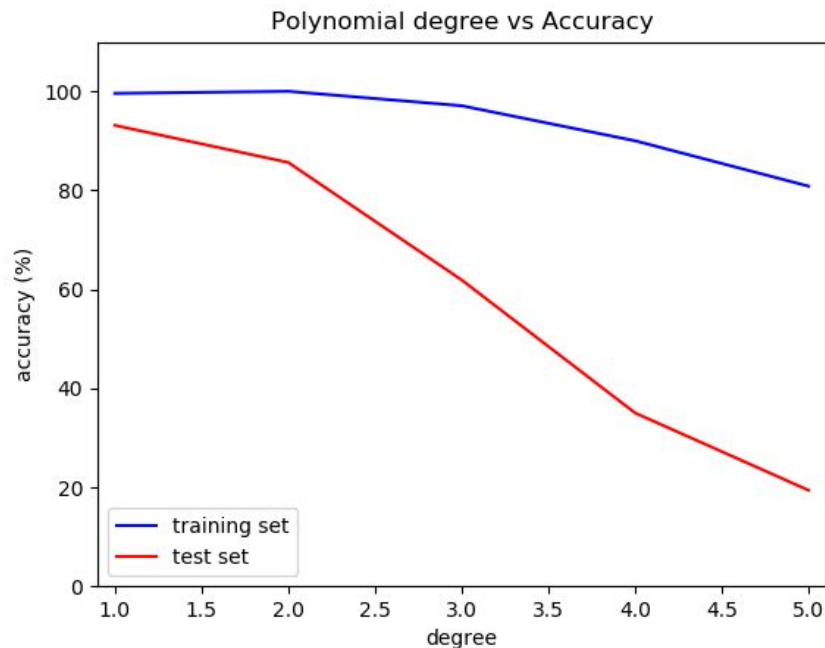
Polynomial Degree vs Accuracy Graph

Modifying the kernel into RBF and training the model with varying gamma values from 0.1 to 0.9 (0.1 increments) The accuracy of the model for the test set significantly drops as gamma reaches 1 as seen in the trend of the gamma vs accuracy plot.
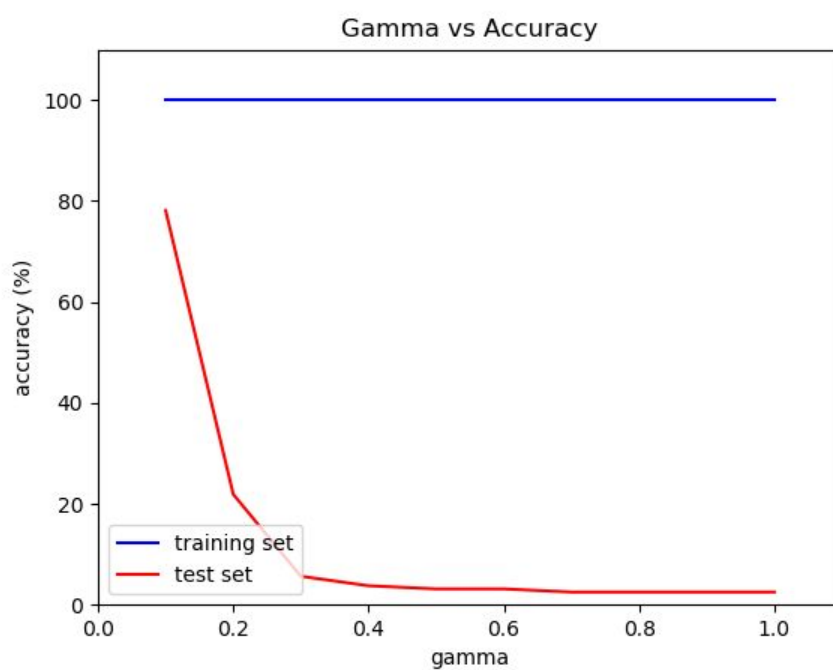
ANN vs SVM

In terms of accuracy, the difference between the ANN and SVM models are only by a fraction. Both models yield decent scores. What sets them apart is the training time. The ANN model takes about 3 mins to finish, while the SVM only takes about 1 second, which is significantly faster than the ANN model. To get a decent accuracy, there are a number of parameters to consider tuning for the ANN model, the activation function, number of hidden layers and hidden layer nodes, learning rate (depending on the solver), etc. The SVM model on the other hand, even with default parameters, the model managed to top the ANN's accuracy, although only by a fraction. For this dataset and problem, the SVM is clearly easier to use.