Rohit Dewan

**Project Neural Net v SVM**

**Task 1:**

for task 1 we first open the regression training file and copy the data to the x and t vectors, making a vector array for both variables for n observations,

wherein n is dynamically calculated based on the number of iterations of 15

entries, with 8 inputs and 7 outputs, that are detected.

After reading the data, the built in MATLAB train command is used to train a neural network

with one hidden layer based on this training data set.

The training output is then compared with the training data

set and the least-squares error is calculated for each observation, summed up

over all observations, and normalized.

This process is repeated where the hidden layer has 1, 4, and 8 hidden units respectively

**The output from the MATLAB program for task 1 was as follows:**

Least-Squares Training Error for regression.tra over all observations with 1 hidden unit is 4.870496e-09 for 1768 observations

Least-Squares Training Error for regression.tra over all observations with 4 hidden units is 5.286821e-09 for 1768 observations

Least-Squares Training Error for regression.tra over all observations with 8 hidden units is 9.720274e-10 for 1768 observations

Least-Squares Training Error for regression.tst over all observations with 1 hidden unit is 1.423191e-08 for 1000 observations

Least-Squares Training Error for regression.tst over all observations with 4 hidden units is 1.221127e-08 for 1000 observations

Least-Squares Training Error for regression.tst over all observations with 8 hidden units is 1.288624e-08 for 1000 observations

**Task 2:**

%we then move to task 2, to design a three layer neural network for

%classification, with 2 inputs and 2 classes, with 1, 2, and 4 hidden units

%in the hidden layer of our neural network.

%for task 2 we repeat essentially the same operation as task 1 for reading the data into the vector arrays,

%however we have to perform an extra step for classes to translate them from discrete to vector outputs, where '1'

%becomes [1 0] and '2' becomes [0 1] respectively. Then the patternnet command is used to define a classification neural network with the train command used as in task 1 for training, and through the vec2ind function the indices are then obtained and compared

**The output from the MATLAB program for task 2 was as follows:**

Training Accuracy for classification.tra over all observations with 1 hidden unit is 9.475000e+01 percent for 400 observations

Training Accuracy for classification.tra over all observations with 2 hidden units is 9.350000e+01 percent for 400 observations

Training Accuracy for classification.tra over all observations with 4 hidden units is 9.250000e+01 percent for 400 observations

Testing Accuracy for classification.tst over all observations with 1 hidden unit is 9.550000e+01 percent for 400 observations

Testing Accuracy for classification.tst over all observations with 2 hidden units is 9.575000e+01 percent for 400 observations

Testing Accuracy for classification.tst over all observations with 4 hidden units is 9.425000e+01 percent for 400 observations

**Task 3:**

%{

for task 3 we repeat essentially the same operation as task 2, where we now

have 16 inputs and 10 classes, and we use 5, 10, and 13 hidden units in the

hidden layer of the MLP

%}

**The output from the MATLAB program for task 3 was as follows:**

Training Accuracy for zipcode.tra over all observations with 5 hidden unit is 9.563333e+01 percent for 3000 observations

Training Accuracy for zipcode.tra over all observations with 10 hidden units is 9.930000e+01 percent for 3000 observations

Training Accuracy for zipcode.tra over all observations with 13 hidden units is 9.906667e+01 percent for 3000 observations

Testing Accuracy for zipcode.tst over all observations with 5 hidden unit is 9.426667e+01 percent for 3000 observations

Testing Accuracy for zipcode.tst over all observations with 10 hidden units is 9.720000e+01 percent for 3000 observations

Testing Accuracy for zipcode.tst over all observations with 13 hidden units is 9.740000e+01 percent for 3000 observations

**Task 4:**

%{

for task 4 we repeat essentially the same operation as tasks 2 and 3, where we now

use the SVM classifier instead. For task 2 we use the fitcsvm function

for binary classification utilizing kernel functions, and then for task 3

we use the fitcecoc function which is a combination of binary SVM

classifiers, where the default kernel function used is linear

%}

**The output from the MATLAB program for task 4 was as follows:**

Training Accuracy for classification.tra over all observations with the binary SVM classifier is 9.375000e+01 percent for 400 observations

Testing Accuracy for classification.tst over all observations with the binary SVM classifier is 96 percent for 400 observations

Training Accuracy for zipcode.tra over all observations with the multiclass SVM classifier is 100 percent for 3000 observations

Testing Accuracy for zipcode.tst over all observations with the multiclass SVM classifier is 9.993333e+01 percent for 3000 observations.