CSE-574
Introduction to Machine Learning
Programming Assignment 3
Logistic Regression and Support Vector Machines

submitted by:
Soham Gupte:50244656
Shreyas Kulkarni:50249367

Report

1. Logistic Regression:

We trained the logistic regression classifier using the entire MNIST training set and obtained the following accuracies:

Training set Accuracy:92.716%
Validation set Accuracy:91.44%
Testing set Accuracy :92.03%
The aforementioned accuracies were along expected lines.

Furthermore, we computed the train and test accuracies for each of the individual classes with the help of a confusion matrix to get a better picture of the error for each category. Following are the training and test accuracies for each of the classes:

Class 0:
Train:97.84684135689619%
Test:98.06%
Class 1:
Train: 97.91013584%
Test:98.15%
Class 2:
Train:91.10528438886648%
Test:89.05%
Class 3:
Train:89.80705515494056%
Test:91.089%
Class 4:
Train:93.90747624948%
Test:85.65%
Class 5:
Train:88.30581316444244%
Test:94.88%
Class 6:
Train:96.299308662%
Test:92.60%
Class 7:
Train:94.339981006%

Test:94.22%
Class 8:
Train:87.54895897%
Test:87.88%
Class 9:
Train:88.76540715%
Test:89.05%

2. Multi-Class Logistic Regression:
We trained a multi-class logistic regression classifier using the entire MNIST training set and obtained the following accuracies:
Training set Accuracy:92.072%

Validation set Accuracy:91.51%

Testing set Accuracy:91.75999999999999%
Using the same methodology used in Binary Logistic Regression, we calculated accuracies for each of the 10 classes. Following are the accuracy values:
Class 0:
Train:97.44058500%
Test:98.2653061224%
Class 1:
Train: 97.4573319%
Test:97.7092511013%
Class 2:
Train:90.2581686%
Test:89.2441860465%
Class 3:
Train:89.1639056%
Test:90.891089108%
Class 4:
Train:93.267244%
Test:93.2790224032%
Class 5:
Train:87.10698936%
Test:85.0896860986%
Class 6:
Train:95.9333062220%
Test:94.6764091858%
Class 7:
Train:93.84615384%
Test:91.7315175097%

Class 8:
Train:86.49762935%
Test:87.063655030%
Class 9:
Train:88.320872903%
Test:88.40436075%

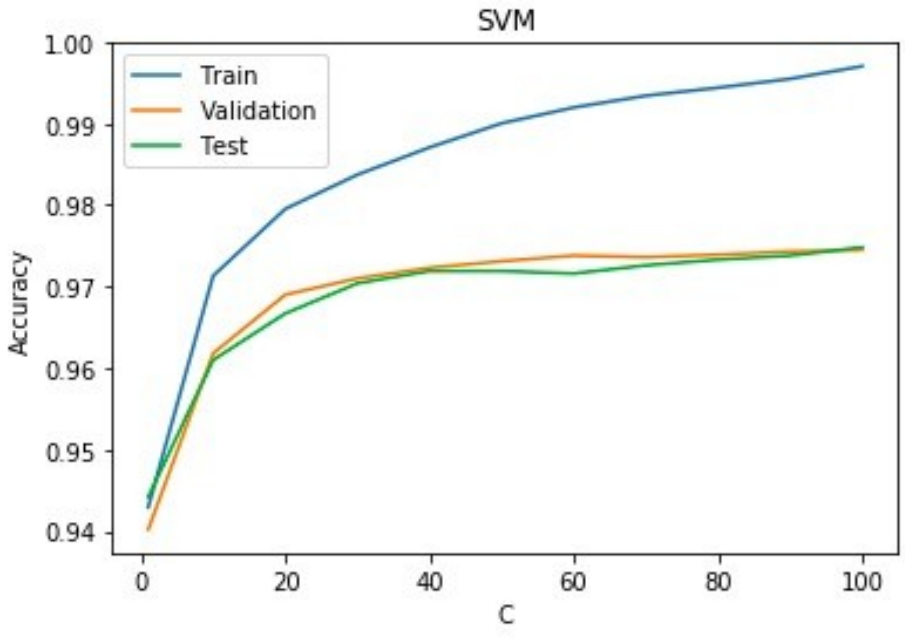
3. Support Vector Machines:

Following are the accuracies for the SVM trained on the entire training dataset for different parameters:

Training set accuracy for linear kernel: 0.97286
Validation set accuracy for linear kernel: 0.9364
Test set accuracy for linear kernel: 0.9378
Training set accuracy for SVM with RBF kernel and gamma=1: 1.0
Validation set accuracy for SVM with RBF kernel and gamma=1: 0.1548
Test set accuracy for SVM with RBF kernel and gamma=1: 0.1714
Training set accuracy for SVM with RBF kernel : 0.94294
Validation set accuracy for SVM with RBF kernel: 0.9402
Test set accuracy for SVM with RBF kernel: 0.9442
Training set accuracy for SVM with RBF kernel and C=1 : 0.94294
Validation set accuracy for SVM with RBF kernel and C=1: 0.9402
Test set accuracy for SVM with RBF kernel and C=1: 0.9442

From the above observations it is clear that using both linear and radial kernels both give similar testing accuracies of around 93-94 percent when other parameters are default. However, when RBF kernel is used with Gamma=1 the testing and validation accuracies drop dramatically to around 17 and 15 percent, respectively. This is because using gamma=1 led to overfitting of the model on the training data. Because we used a comparatively high Gamma as opposed to the sklearns default value(1/N) our learnt model became a high bias and low-variance model. Due to this our model has poor tolerance for data that is 'different' from the training data. As a result this model returned extremely low values of validation and test accuracies

3.1 SVM Accuracy with changing values of C:



From the above graph it is clear that as we increase the value of C the accuracy of the classifier generally increases. This is because C in our case controls the influence of individual support vectors which helps the model ‘ignore’ the error/cost of one or two misclassified vectors.