CSE 574 Introduction to Machine Learning
Programming Assignment 2
Roshan Saundankar: 50419577
Sumeet Sahu: 50367891
Mrunmayee Rane: 50417094

LOGISTIC REGRESSION AND SUPPORT VECTOR MACHINE

Binary Logistic Regression (BLR):

Set	Accuracy	Error
Training	**81.876%**	**18.124%**
Validation	**80.58%**	**19.42%**
Testing	**81.61%**	**18.39%**

Training error is less than Testing error. Therefore it can be said that the Linear model performs better on already seen data, but when it gets the new data set it gives a small margin error. No huge difference.

Multiclass Logistic Regression (MLR):

Set	Accuracy	Error
Training	**93.176%**	**6.824%**
Validation	**92.46%**	**7.54%**
Testing	**92.51%**	**7.49%**

Training error is slightly less than Testing error. Therefore, we can conclude that this Linear model performs better on already seen data, but when it is tried on new data set it gives a small error. Errors are almost equal.

Performance Difference with Multiclass Strategy (MLR) with one v/s all Binary Logistic Regression (BLR) strategy:

Set	MLR Accuracy	BLR Accuracy
Training	**93.176%**	**81.876%**
Validation	**92.46%**	**80.58%**
Testing	**92.51%**	**81.61%**

- In multiclass logistic regression we have classified all the classes(total 10) of MNIST dataset at once, whereas, in one-vs-all(BLR) we only classify one class with respect to all other at a time.
- Therefore, it can be said that Multiclass has less time complexity.
- We observed the accuracy of the multiclass was better than the BLR classification. This is due to the parameters which are evaluated independently in multiclass, this helps in stopping wrong classification.

Support Vector Machine (SVM):

I. Using Liner Kernel:

Set	Accuracy
Training	**92.69%**
Validation	**91.58%**
Testing	**91.92%**

From the above results, we can infer that Linear Kernel works like a linear model, as the results are almost same as the previous linear model we trained.

II. Radial Basis Function:

a) Using Radial Basis Function (Gamma =1)

Set	Accuracy
Training	**100.0%**
Validation	**17.49%**
Testing	**19.0%**

b) Using Radial Basis Function with value of gamma setting to default (all other parameters kept as default)

Set	Accuracy
Training	**91.956%**
Validation	**92.0%**
Testing	**92.54%**

c) Using Radial Basis Function with value of gamma setting to default and varying value of C (1, 10, 20, 30, ...,100)

We go through the C values and take note of the optimum setting and then test the whole data on that setting. This variable controls the importance we are giving to the Slack variable. Therefore, there is a trade-off between the width of the margin and C value.

Below are the results for different values of C on Training, Testing and Validation data:

C	Training Accuracy	Validation Accuracy	Testing Accuracy
1	**96.502%**	**96.04%**	**96.26%**

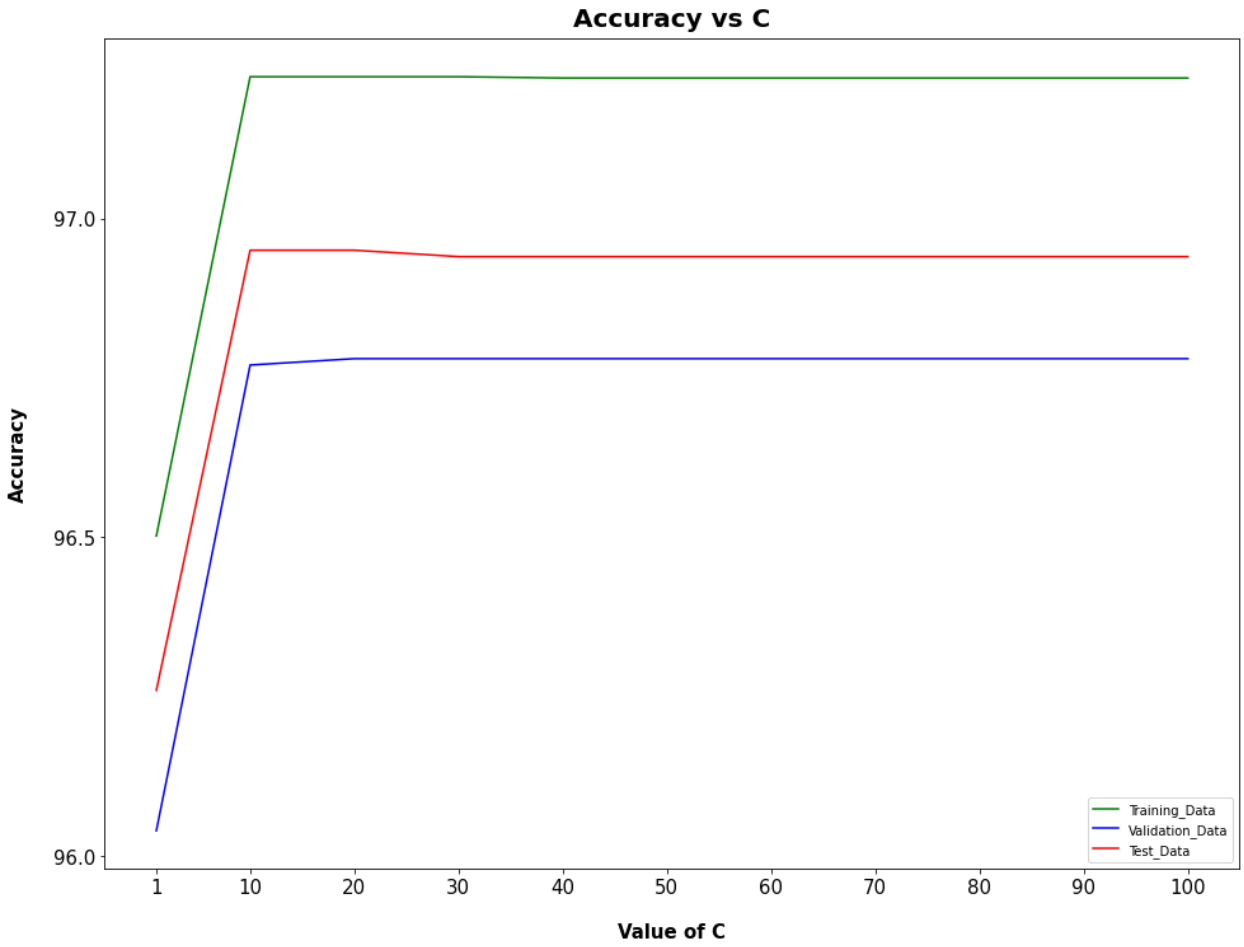
10	**97.222%**	**96.77%**	**96.95%**
20	**97.567%**	**96.78%**	**96.95%**
30	**98.21%**	**97.48%**	**97.64%**
40	**98.54%**	**97.78%**	**97.94%**
50	**98.725%**	**97.93%**	**97.95%**
60	**97.821%**	**97.81%**	**97.93%**
70	**97.891%**	**97.952%**	**97.96%**
80	**97.22%**	**97.78%**	**97.84%**
90	**97.17%**	**97.48%**	**96.24%**
100	**97.20%**	**97.51%**	**97.54%**

we can conclude that we are getting the best result by setting gamma to default and taking C = 70.

Results for the whole dataset using optimal parameters:

Kernel	**C**	**Training Accuracy**	**Validation Accuracy**	**Testing Accuracy**
RBF(Gamma=default)	**70**	**99.33999999999999%**	**97.36%**	**97.26%**

Plot of accuracy obtained on each of Training, Testing and Validation dataset with respect to various values of C:



It can be concluded that the dataset is non-linear as it gives better result on this non-linear model.