CSE574 Fall 2021 Introduction to Machine Learning

Programming Assignment 3

Classification and Regression

Group 26

NAME	UB PERSON ID:	UB IT NAME:
Akshaya Mohan	50419115	akshayam
Oviyaa Balamurugan	50418472	oviyaaba
Vaishnavi Ruppa Gangadharan	50418483	vruppaga

1. Logistic regression:

The total error and accuracy with respect to each category is reported below

	Training Data		Validation data		Testing data	
Class	**Accuracy %**	**Error %**	**Accuracy %**	**Error%**	**Accuracy %**	**Error%**
0	97.826	2.174	97.7	2.3	98.163	1.837
1	97.910	2.09	96.8	3.2	98.237	1.763
2	91.125	8.875	88.0	12	88.95	11.05
3	89.553	10.447	88.8	11.2	91.089	8.911
4	93.804	6.196	93.7	6.3	93.279	6.721
5	87.989	12.011	86.8	13.2	85.313	14.687
6	96.299	3.701	95.7	4.3	94.676	5.324
7	94.226	5.774	92.3	7.7	92.217	7.783
8	87.445	12.555	84.3	15.7	87.063	12.937
9	89.149	10.149	90.5	9.5	89.09	10.91

The overall training, validation and testing accuracy along with total error is reported below

	Training data	**Validation data**	**Testing data**
Accuracy (%)	92.67	91.46	91.94
Error(%)	7.33	8.54	8.06

Logistic regression considers all the data points to construct the hyperplane separating the data as per the classes. As such it works well on data with low input features. Here, we observe that though there is not a significant difference, training error is less than both validation and testing error.

2. Multi-class Logistic regression:

The total error and accuracy with respect to each category is reported below

	Training Data		**Validation data**		**Testing data**	
Class	**Accuracy %**	**Error %**	**Accuracy %**	**Error%**	**Accuracy %**	**Error%**
0	97.237	2.763	97.899	2.899	98.367	1.633
1	97.439	2.561	97.6	2.4	97.797	2.203
2	90.802	9.198	89.7	10.3	89.924	10.076
3	90.528	9.472	89.9	10.1	90.693	9.037
4	94.134	5.866	93.89	6.11	93.482	6.518
5	89.029	10.971	89.3	10.7	86.883	13.117
6	95.790	4.21	94.8	5.2	94.676	5.324
7	94.093	5.907	92.2	7.8	91.926	8.074
8	89.940	10.06	86.7	13.3	88.295	11.705
9	91.675	8.325	92.6	7.4	91.972	8.028

The overall training, validation and testing accuracy along with total error is reported below:

	Training data	**Validation data**	**Testing data**
Accuracy (%)	93.176	92.46	92.51
Error(%)	6.824	7.54	7.49

Similar to the values reported in the previous approach, training error is slightly better than validation and testing error. From the values reported, it is clear that Linear model performs well on known data compared to unseen data.

Comparison of Multi-class strategy with one-vs-all strategy:

	Training accuracy(%)	**Validation accuracy(%)**	**Test accuracy(%)**
Multi-class	93.176	92.46	92.51
One-vs-all	92.67	91.46	91.94

Here, we observe that out of the two approaches, the performance of multi-class regression is better. This is because, in the given dataset, each input belongs to one class and the parameters are also estimated independently thereby avoiding wrong classification.

3. Support Vector Machine:

Kernel	Gamma	Training accuracy(%)	Validation accuracy(%)	Test accuracy(%)
Linear	0	92.58	91.33	91.51
RBF	1	100	10	11.35
RBF	0 (default)	91.878	91.86	92.45

From the above table, we observe that using radial basis function with default gamma set of 0 performs better on test data than using linear kernel. We also observe that using a radial basis function with gamma set to 1 performs very poorly on validation and test data but gives 100% accuracy on training data. This can be attributed to the overfitting issue.

The below table reports accuracy obtained by setting the gamma value to default and varying the C value in the range 1 to 100 in steps of 10.

C	Training accuracy(%)	Validation accuracy(%)	Test accuracy(%)
1	96.526	96.14	96.37
10	**97.225**	**96.67**	**97.0**
20	97.224	96.64	96.98
30	97.224	96.64	96.98
40	97.224	96.64	96.98
50	97.224	96.64	96.98
60	97.224	96.64	96.98

70	97.224	96.64	96.98
80	97.224	96.64	96.98
90	97.224	96.64	96.98
100	97.224	96.64	96.98

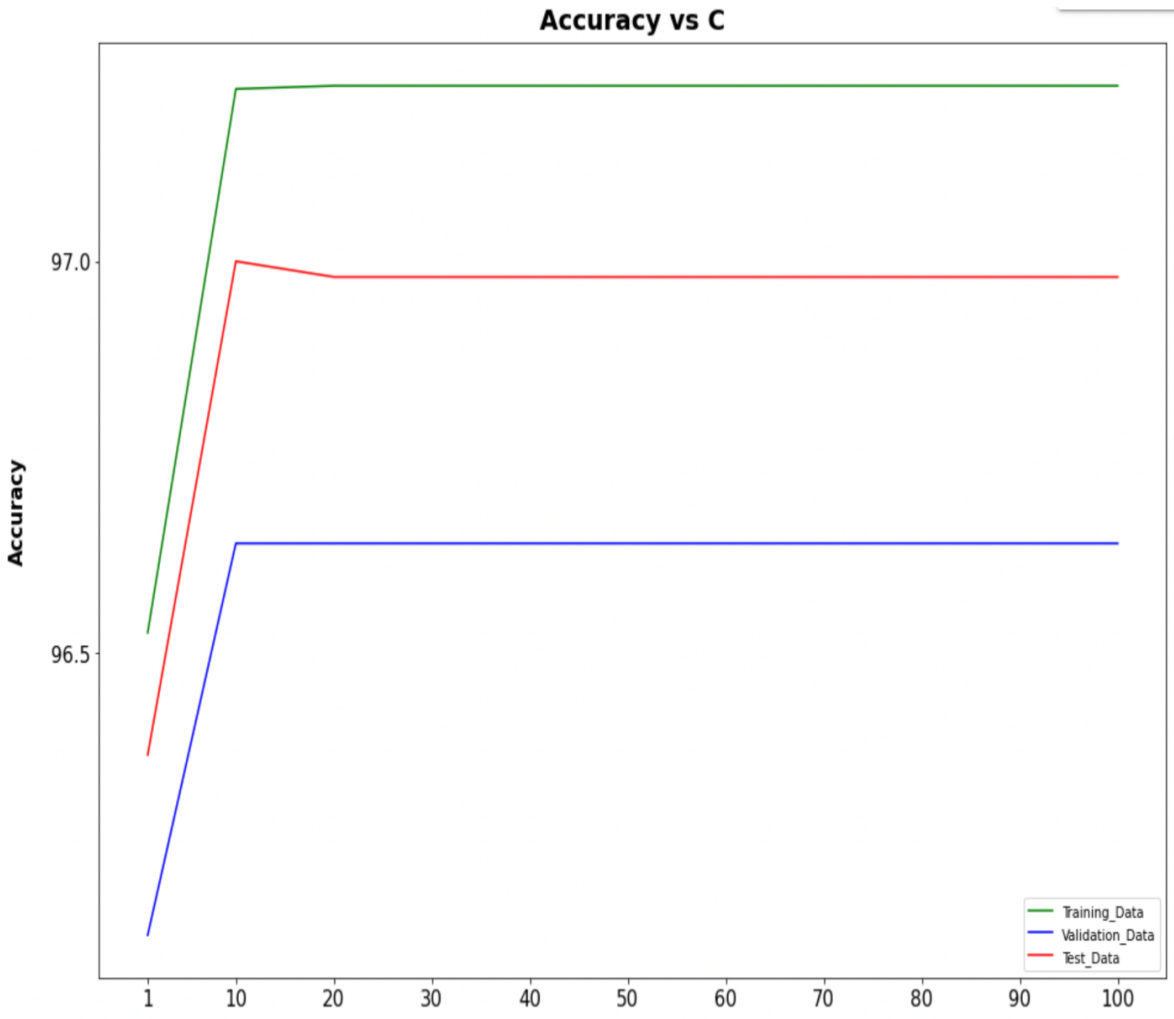
From the above table, it is clear that the optimal value for C is 10 since the accuracy of that setting is the highest.

With this optimal setting, i.e. gamma set to default and C set to 10, we train the entire dataset and observe the accuracies.

Kernel	**Optimal C**	**Training accuracy(%)**	**Validation accuracy(%)**	**Test accuracy(%)**
RBF (default)	10	98.13	96.98	97.3

From the table it is clear that, properly tuned RBF kernel performs better than the Linear kernel for the given data set.

The observed accuracies are also plotted in the graph below:


Accuracy vs C
Accuracy
97.0
96.5
1
10
20
30
40
50
60
70
80
90
100
Training_Data
Validation_Data
Test_Data