PA 3 - Classification and Regression

CSE 574: Introduction to Machine Learning

Team 7:

Firnaz Luztian Adiansyah - firnazlu@buffalo.edu

Sumedha Salil Nashte - sumedhas@buffalo.edu

Anushree Naveen Gupta - agupta38@buffalo.edu

UNIVERSITY AT BUFFALO

Activity 1 : Implement Logistic Regression.

Accuracies for classification using Logistic Regression :

	Training data	Validation data	Test data
Accuracies	86.336%	85.52%	85.63%

Time taken to complete : *368.1026077270508 seconds*, which is approximately 6 minutes.

Observation:

Logistic Regression considers all the points in a dataset and outputs any hyperplane which separates the data rather than the best hyperplane. Thus the accuracy given by Logistic Regression is less.

Activity 2: Use the SVM toolbox sklearn.svm. to perform classification.

Case 1: Using linear kernel (all other parameters are kept default).

Kernel	Training data	Validation data	Test data
linear	97.286%	93.64%	93.78%

Time taken to complete : *779.365645647049 seconds*, which is approximately 12 minutes.

Case 2: Using radial basis function with value of gamma setting to 1 (all other parameters are kept default).

Kernel	Gamma	Training data	Validation data	Test data
rbf	1	100.0%	15.48%	17.14%

Time taken to complete : *15441.136420488358 seconds*, which is approximately 4.3 hours.

Case 3: Using radial basis function with value of gamma setting to default (all other parameters are kept default).

Kernel	Gamma	Training data	Validation data	Test data
rbf	0	94.294%	94.02%	94.42%

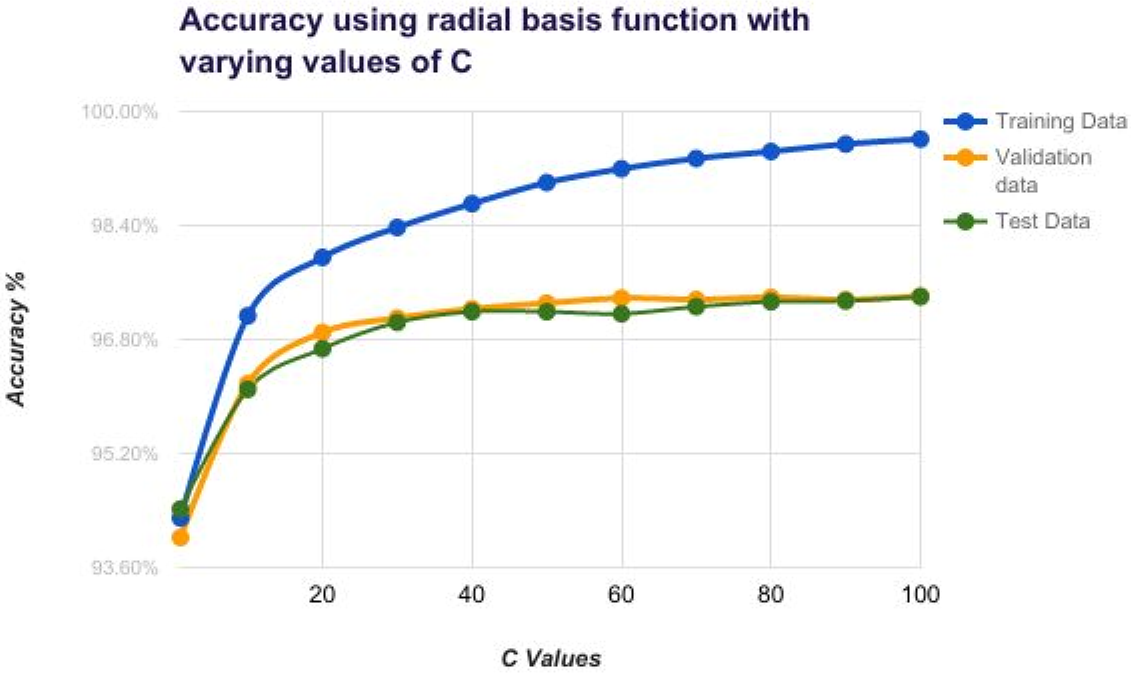
Time taken to complete : *1469.2295320034027 seconds*, which is approximately 24 minutes.

Case 4: Using radial basis function with value of gamma setting to default and varying value of C (1; 10; 20; 30;......; 100)

Kernel	C Values	Training data	Validation data	Test data
rbf	1.0	94.294%	94.02%	94.42%
rbf	10.0	97.132%	96.18%	96.1%
rbf	20.0	97.952%	96.9%	96.67%
rbf	30.0	98.372%	97.1%	97.04%
rbf	40.0	98.706%	97.23%	97.19%
rbf	50.0	99.002%	97.31%	97.19%
rbf	60.0	99.196%	97.38%	97.16%
rbf	70.0	99.34%	97.36%	97.26%
rbf	80.0	99.438%	97.39%	97.33%
rbf	90.0	99.542%	97.36%	97.34%
rbf	100.0	99.612%	97.41%	97.4%

Time taken to complete : *9028.078711986542 seconds*, which is approximately 2.5 hours.

Graph of accuracy with respect to C values :



Observations:

1. **Case 1: Linear kernel is useful for multi-dimensional data and where original data is very informative. In our case, MNIST dataset consists of digit images which are high dimensional but the pixels are not very informative hence the accuracy is not as high as it would be for nonlinear models.**
2. **In Case 2 where gamma = 1 we can clearly see the overfitting problem when compared to Case 3 where gamma = 0.**
3. **For Case 4 where we vary the C parameter with rbf kernel, which C parameter tells the SVM optimization how much we want to avoid misclassifying each training example. We can observe the below points:**
    a. **From graph we can say that on increasing C value, accuracy increases.**
    b. **For large values of C like 50 and above, the optimization will choose a smaller-margin hyperplane which classifies more of training data points correctly.**

c. **Conversely, a very small value of C like less that 50, will cause the optimizer to choose a larger-margin separating hyperplane, even if it misclassifies more data points.**

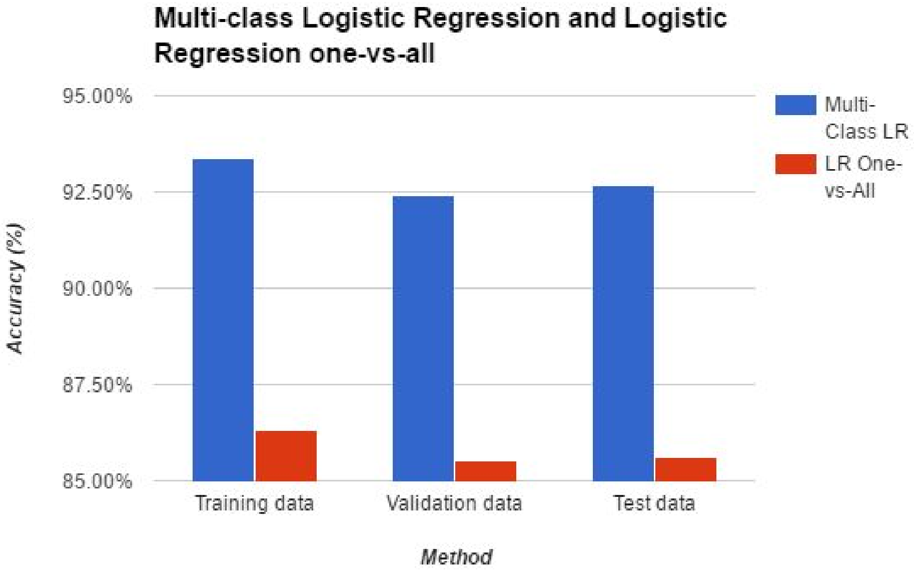
CONCLUSION

Thus from Activity 1 and 2 we can conclude that Logistic Regression works better when we have a dataset with fewer of features whereas SVM (with rbf kernel) works better for multi-dimensional data. Our MNIST dataset has more number of dimensions and thus SVM gives better accuracy for it.

Activity 3 : Implement gradient descent minimization of Multi-class Logistic Regression

Accuracies for classification using Multi-class Logistic Regression :

Method	Training data	Validation data	Test data	Completion time (seconds)
Multi-class Logistic Regression	93.39%	92.43%	92.67%	184.5301752090454
Logistic Regression one-vs-all	86.336%	85.52%	85.63%	368.1026077270508



CONCLUSION

Logistic Regression traditionally is used for binary classification. Multi-class Logistic Regression can be used to build a classifier that can classify 10 classes at the same time. In term of performance from the table above we are able to observe that the accuracy obtained in Multi-class Logistic Regression is higher comparing to one-vs-all Logistic Regression strategy. Moreover, since we are able to classify 10 classes simultaneously, the time to complete for Multi-class Logistic Regression is less.