CSE574-ASSIGNMENT 3 – GROUP 52
CLASSIFCATION AND REGRESSION

May 3, 2017

Name	UBIT Name	Person #
Sourav Puri	Souravpu	50206918
Shaleen Somani	Shaleens	50207380
Amardeep Virdy	Avirdy	50208831

Binomial Logistic Regression

Logistic regression is probabilistic classification statistical model. Logistic regression can either be binomial or multinomial. Binomial or binary logistic regression (Blr) is only applicable in situations in which the outcome for a variable can have only two possible values, and thus can be dealt with binary values by using 0 to specify one condition and 1 to specify another condition.

With the weight values and the input data, Theta was calculated with the following formula:

$$\theta_n = \sigma(\mathbf{w}^T \mathbf{x}_n)$$

Using the Theta values calculated above, Error function was calculated by using the following formula:

$$E(\mathbf{w}) = -\frac{1}{N} \ln p(\mathbf{y}|\mathbf{w}) = -\frac{1}{N} \sum_{n=1}^{N} \{y_n \ln \theta_n + (1 - y_n) \ln(1 - \theta_n)\}$$

where y is the value of label and N is the total number of inputs.

After computing the error, the gradient of error function with respect to w, was computed by using the following formula:

$$\nabla E(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} (\theta_n - y_n) \mathbf{x}_n$$

Upon implementing Binomial logistic regression in the code, the Accuracies that were obtained were as follows:

Training set Accuracy:86.182%

Validation set Accuracy:85.36%

Testing set Accuracy:85.38%

and the time taken for the computation to finish on Metallica server was, 12 minutes and 9 seconds.

Multinomial Logistic Regression

Training set Accuracy:92.846%

Validation set Accuracy:92.39%

Testing set Accuracy:92.39%

Multi-class classification is an extension of Logistic Regression. it is a model that is used to predict the probabilities of the different possible outcomes of a categorically distributed dependent variable, given a set of independent variables (which may be real-valued, binary-valued, categorical-valued, etc.). With this method, we don't need to build 10 classifiers like before. Instead, we now only need to build 1 classifier that can classify 10 classes at the same time.

The posterior probabilities are given by a softmax transformation of linear functions of the feature variables as below:

$$P(y = C_k|\mathbf{x}) = \frac{\exp(\mathbf{w}_k^T\mathbf{x})}{\sum_j \exp(\mathbf{w}_j^T\mathbf{x})}$$

We perform a one-to-many (1-of-K classes) coding scheme in which the target vector y_n for a feature vector x_n belonging to class C_k is a binary vector with all elements zero except for element k. We compute the cross-entropy error function for the multi-class classification problem as:

$$E(\mathbf{w}_1, \cdots, \mathbf{w}_K) = -\ln\ P(\mathbf{Y}|\mathbf{w}_1, \cdots, \mathbf{w}_K) = -\sum_{n=1}^{N}\sum_{k=1}^{K} y_{nk} \ln\ \theta_{nk}$$

And the error gradient function can be calculated as:

$$\frac{\partial E(\mathbf{w}_1, \cdots, \mathbf{w}_K)}{\partial \mathbf{w}_k} = \sum_{n=1}^{N}(\theta_{nk} - y_{nk})\mathbf{x}_n$$

MLR is an effective method to predict 1 out of K classes for a given data and its accuracy is around 93% for all training, testing, and validation data.

Performance and Accuracy of Binomial Logistic Regression Vs Multinomial Logistic Regression:

Binomial Logistic Regression estimates the probability that a characteristic is present (e.g. estimate probability of "success") given the values of explanatory variables, in this case a single categorical variable; $\pi = P_r\,(Y = 1|X = x)$. It shows BLR is a good method to predict the output class of input data with **accuracy around 86%.**

Multinomial Logistic Regression is nothing but a combination of many Binary Logistic Regression. Generally, MLR estimates the different probabilities of a given input data (with D features) to fall under a given category out of K classes. Based on the highest probability, it assigns the respective class to the input data. MLR is an effective method to predict 1 out of K classes for a given data and its **accuracy is around 93%.**

Performance of MLR seems to be better than BLR as the time elapsed in predicting the outcomes for same dataset was less in case of MLR than BLR. Also, the accuracy of MLR is better than BLR as well. Hence, we can conclude that MLR is better than BLR in terms of performance.

Support Vector Machines.

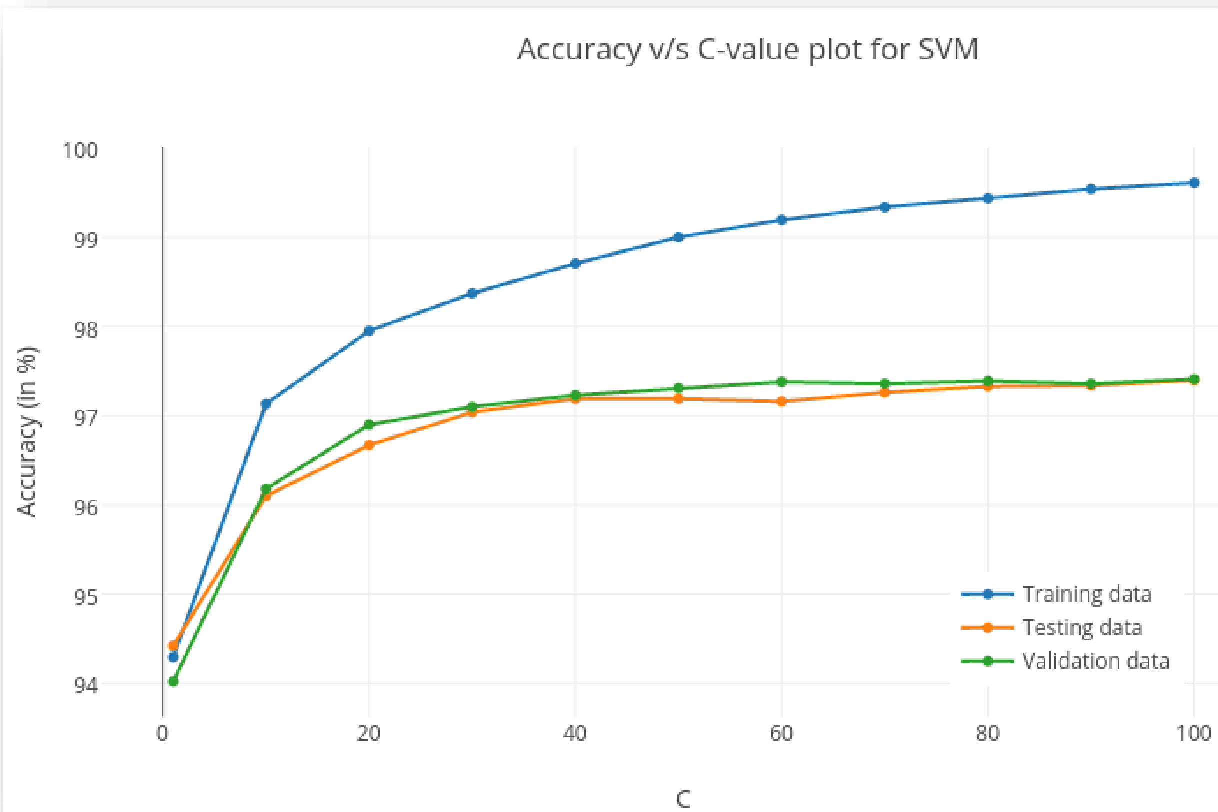
Below are the results for the following cases:

- Using linear kernel (all other parameters are kept default).
- Using radial basis function with value of gamma setting to 1 (all other parameters are kept default).
- Using radial basis function with value of gamma setting to default (all other parameters are kept default).
- Using radial basis function with value of gamma setting to default and varying value of C (1; 10; 20; 30; ……….; 100)

S No.	Kernel	Gamma value	C-value	Training set accuracy(%)	Test set accuracy(%)	Validation Accuracy(%)
1	Linear	*default*	*default*	97.286	93.78	93.64
2	RBF	*default*	*default*	94.294	94.42	94.02
3	RBF	1	*default*	100.00	17.14	15.48
4	RBF	*default*	1	94.294	94.42	94.02
5	RBF	*default*	10	97.132	96.10	96.18
6	RBF	*default*	20	97.952	96.67	96.90
7	RBF	*default*	30	98.372	97.04	97.10
8	RBF	*default*	40	98.706	97.19	97.23
9	RBF	*default*	50	99.002	97.19	97.31
10	RBF	*default*	60	99.196	97.16	97.38
11	RBF	*default*	70	99.340	97.26	97.36
12	RBF	*default*	80	99.438	97.33	97.39
13	RBF	*default*	90	99.542	97.34	97.36
14	RBF	*default*	100	99.612	97.40	97.41

It can be inferred from the above observations that:

- Increasing the value of gamma leads to **overfitting** which leads to a 100% accuracy in case of training data set but a very poor accuracy in case of test and validation dataset. This is due to the fact that gamma affects the contribution by each training sample on the learned boundary.
- Generally, one would expect more reliable accuracy by the radial basis function, which is used to learn non-linear boundaries. But in this particular case, linear kernel provides a better accuracy. It must also be noticed that the time taken by radial basis function is comparatively more than the time taken by linear kernel.



Inference from the above graph:

- With lower value of C, the margin created is wider which leads to more false samples to fit in and therefore the accuracy is low.
- As the value of C increases, the margin becomes narrower while training, and therefore the accuracy increases.
- But, one important point to be noticed from above plot is that for test and validation data set, the accuracy increases up to a certain level and then saturates. This indicates that there might be a case of overfitting at higher values of C.

References:

1. *https://en.wikipedia.org/wiki/Multinomial_logistic_regression*
2. *https://onlinecourses.science.psu.edu/stat504/node/150*
3. *http://www.sciencedirect.com/science/article/pii/S0169743916303318*
4. *https://en.wikipedia.org/wiki/Logistic_regression*