IMPLEMENTATION OF LOGISTIC REGRESSION

Rohith Kumar Gadalay
University At Buffalo
rohithku@buffalo.edu

ABSTRACT

In this paper, the primary task is to implement logistic regression from scratch for a 2-class problem and classify suspected FNA cells(Fine Needle Aspirate) to Benign(class 0) or Malignant(class 1) using logistic regression as the classifier. The data is divided into training, validation and testing sets and the graphs are plotted by increasing the epochs and learning rates. The dataset in use is the Wisconsin Diagnostic Breast Cancer (WDBC dataset)

1 INTRODUCTION

Logistic Regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determines an outcome. In the given dataset, FNA cells classification in this project, implies comparing given set of values and mapping them to Benign or Malignant. This includes processing a huge dataset with a set of input features in the dataset. This paper discusses a diagnosis technique that uses the FNA cells with computational interpretation via machine learning and aims to create a classifier that provides a high-level accuracy with low rate of false- negatives.

2 DATASET

For this project, we use the data extracted from the Wisconsin Diagnostic Breast Cancer (WDBC) dataset. This dataset will be used for training, validation and testing .The dataset contains 569 instances with 32 attributes (ID, diagnosis(B/M),30-real valued input features).Features are computed from a digitized image of a fine needle aspirate(FNA) of a breast mass. Computed features describe the following characteristics of the cell nuclei present in the image:

1	radius (mean of distances from center to points on the perimeter)
2	texture (standard deviation of gray-scale values)
3	perimeter
4	area
5	smoothness (local variation in radius lengths)
6	compactness ($perimeter^2/area - 1.0$)
7	concavity (severity of concave portions of the contour)
8	concave points (number of concave portions of the contour)
9	symmetry
10	fractal dimension ("coastline approximation" - 1)

The mean, standard error, and worst or largest (mean of the three largest values) of these features were computed for each image resulting in 30 features.

3 PRE-PROCESSING

In this experiment we have some steps to follow in-order to get the required solution

1. Reading the dataset file
2. Processing the dataset
    1. Dropping the column id
    2. Map the label column to 0 and 1
3. Normalizing the dataset by using min and max function
4. Split the normalized data in such a way that training set has 80% of data, validation has 10% of data and testing has 10% of data.
5. Initilaize the weights, biases and learning rate respectively.
6. Calculate the loss function, sigmoid function, derivatives of weights and biases and pass the appropriate values to the logistic regression function in-order to get the desired outputs.

4 ARCHITECTURE

In this we use logistic regression to get the desired output. The algorithm is as follows: Given set of inputs X, assign them to one of the two classes or categories (0 or 1) by mapping probabilities for each input towards a particular category using a logistic function or sigmoid function.

The linear function is defined as follows:

$$Z = \theta^{\top} X + b$$

The above function denotes that each input goes through an activation function and gives the corresponding Z value. The activation function is

$$a = \sigma(Z)$$

where the value of σ(Z) is as follows:

$$\sigma(Z) = 1/\{1 + e^{-Z}\}$$

The loss function is calculated as

$$L = -((y \log a + (1 - y) \log (1-a))/m$$

$$= -\{y \log (\sigma(Z)) + (1 - y) \log (1- (\sigma(Z))) \} /m$$

Substituting the value of a in the above function and differentiating the above equation with respect to θ_i gives us

$$\Delta\theta_i = \delta L/\delta\theta_i$$

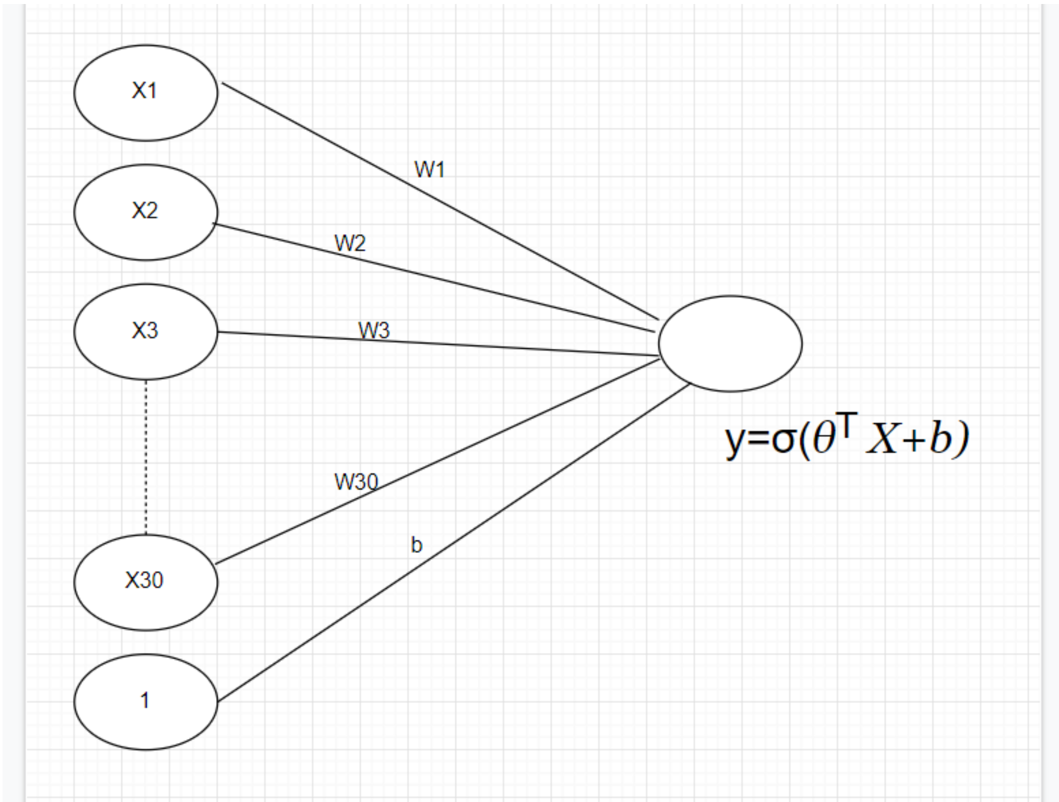
$$= -1/m\ \delta/\delta\theta_i\ \{ y \log (\sigma(Z)) + (1 - y) \log (1- (\sigma(Z)))\}$$

$$= -1/m\ \{y*1/ \sigma(Z) *\delta/\delta\theta_i\, \sigma(Z) + (1-y) *1/ (1-\sigma(Z)) * \delta/\delta\theta_i\, (1-\sigma(Z))\}$$

Simplifying the above equation, we get

$$\Delta\theta_i = -X_i\,(y- \sigma(Z))/m$$

For the given dataset, it has 30 different columns and can be mapped as below.



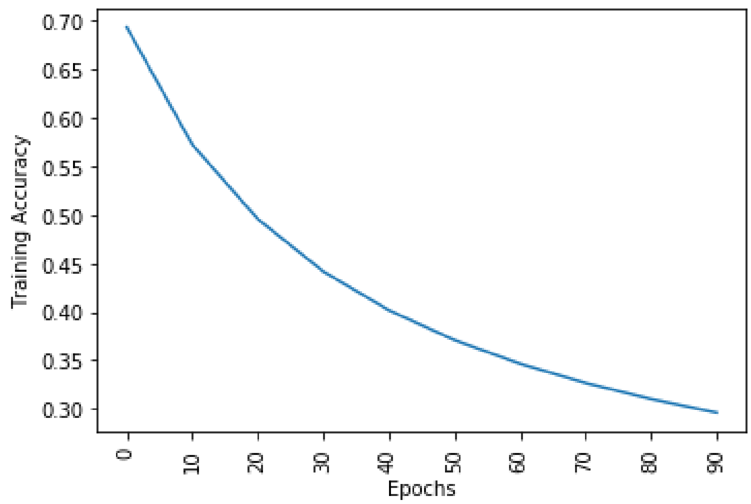
5 RESULTS

For this experiment, I have created data with training data as 80% of dataset, validation data as 10% of dataset and testing data as 10% of dataset.

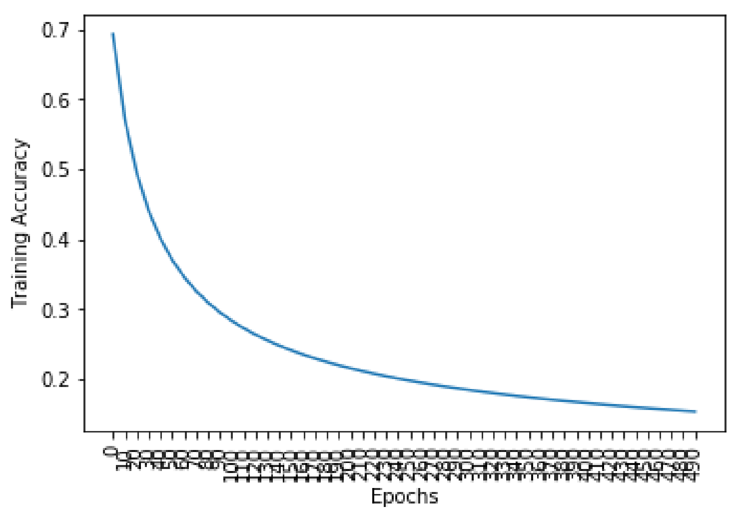
After implementing the logistic regression for different learning rates and epochs, the accuracy of training data is as follows:

5.1 Training Accuracy vs Epochs

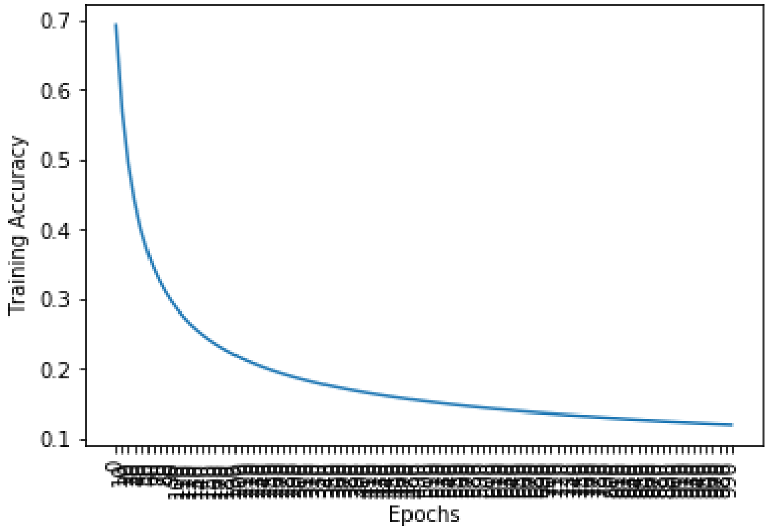
1) Learning rate=0.5 and epochs=100
   Training accuracy: 93.72294372294373 %



2) Learning rate=0.5 and epochs=500
   Training accuracy: 96.53679653679654 %

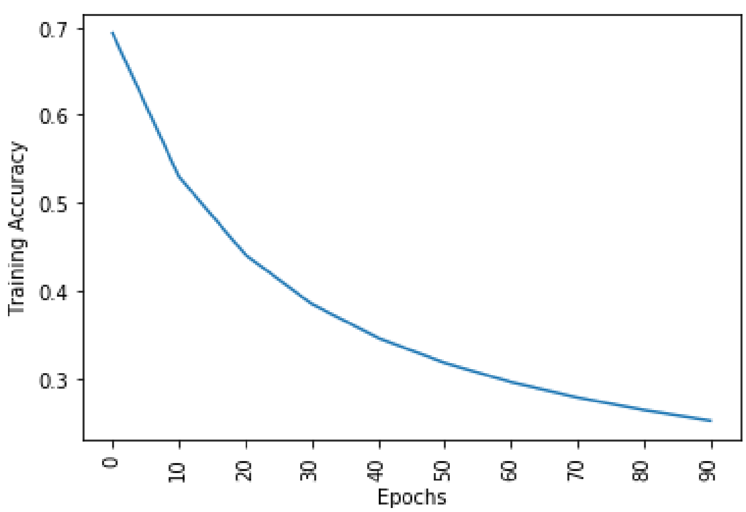


3) Learning rate=0.5 and epochs=1000
   Training accuracy: 96.96969696969697 %

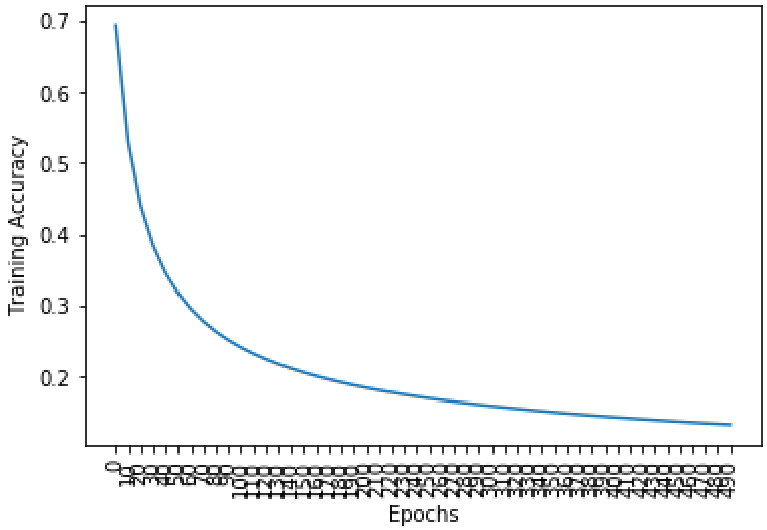


For learning rate=0.5 and change in the number of epochs from 100 to 1000 the training accuracy increases.

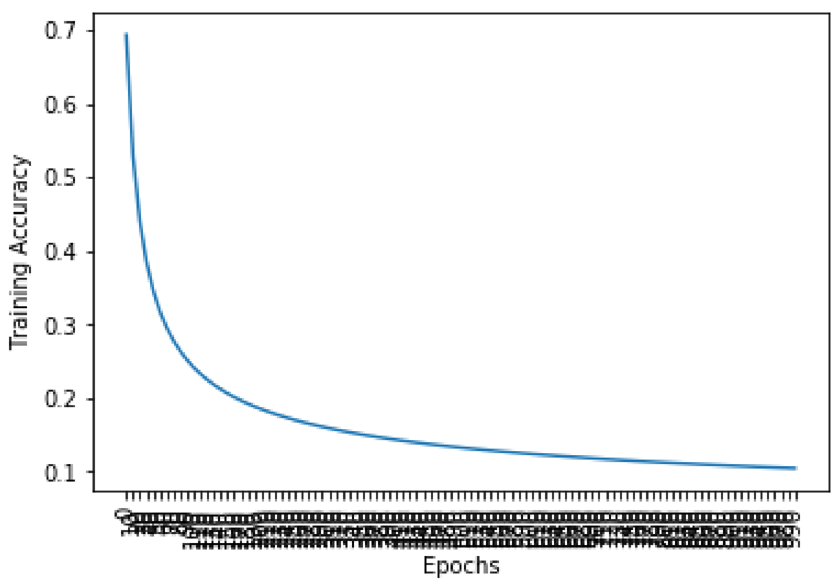
4) Learning rate=0.75 and epochs=100
   Training accuracy: 94.15584415584415 %



5) Learning rate=0.75 and epochs=500
Training accuracy: 96.96969696969697 %

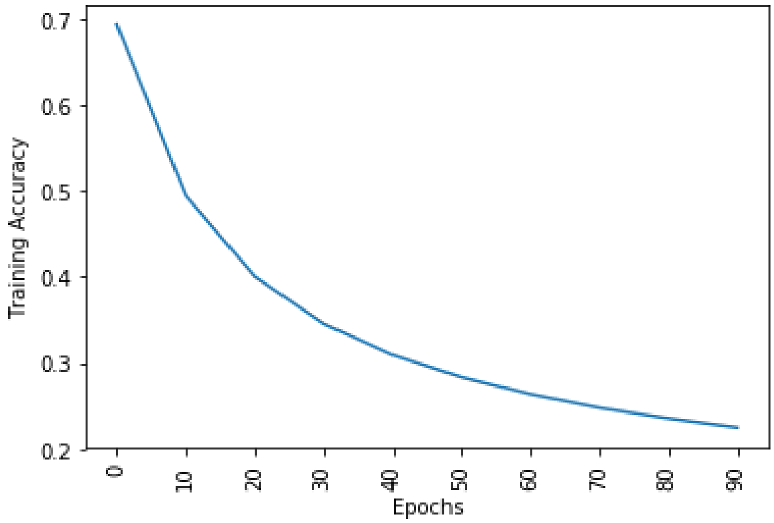


6) Learning rate=0.75 and epochs=1000
Training accuracy: 97.40259740259741 %

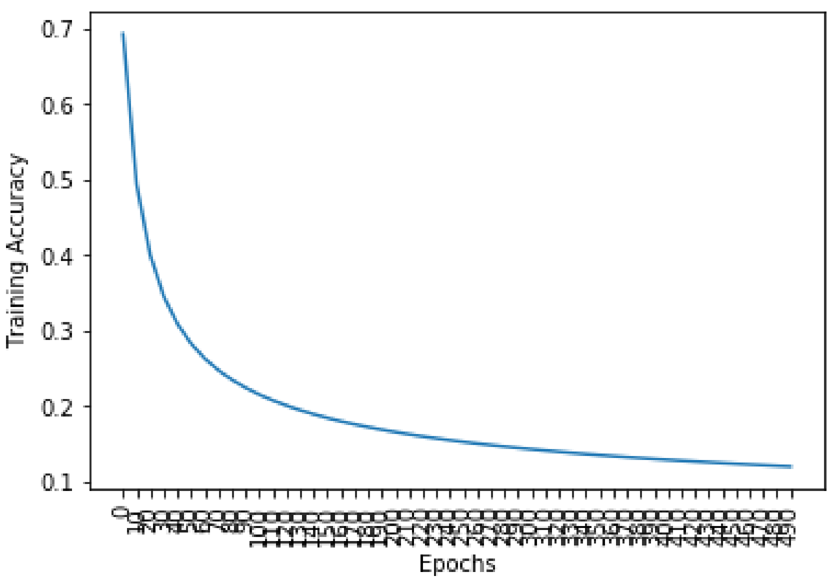


For learning rate=0.75 and change in the number of epochs from 100 to 1000 the training accuracy increases.

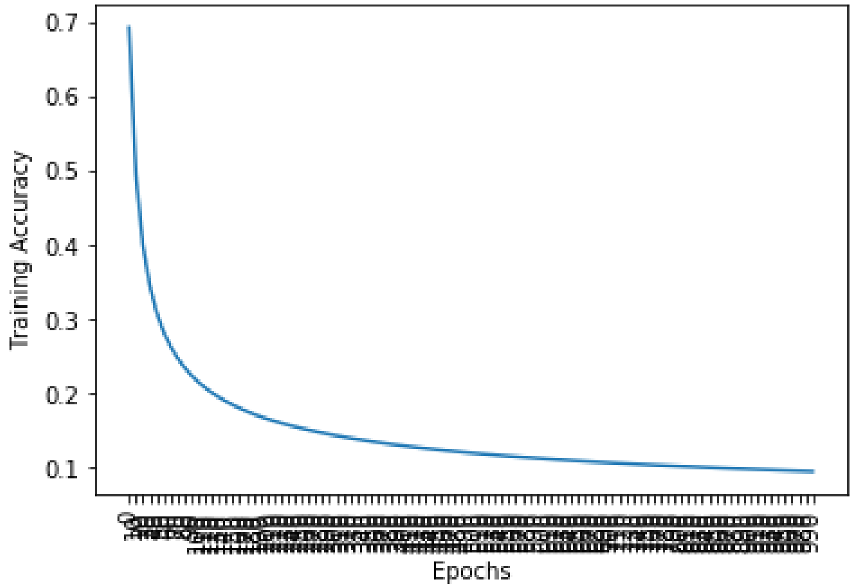
7) Learning rate=1 and epochs=100
Training accuracy: 94.8051948051948 %



8) Learning rate=1 and epochs=500
Training accuracy: 96.96969696969697 %



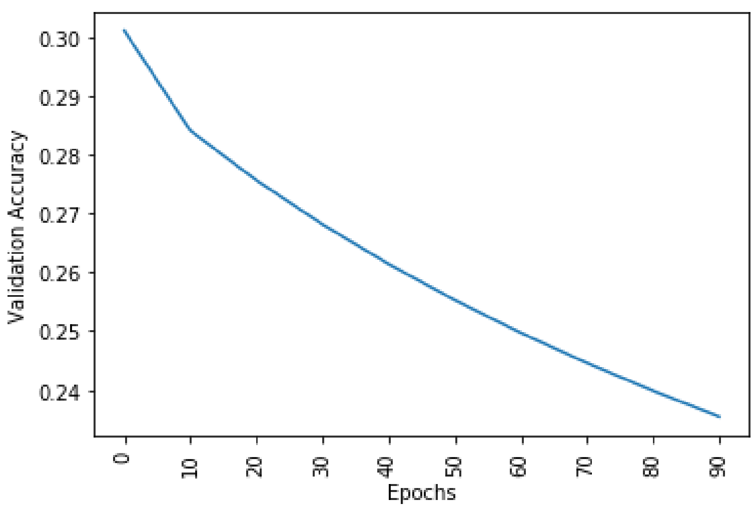
9) Learning rate=1 and epochs=1000
Training accuracy: 97.83549783549783 %



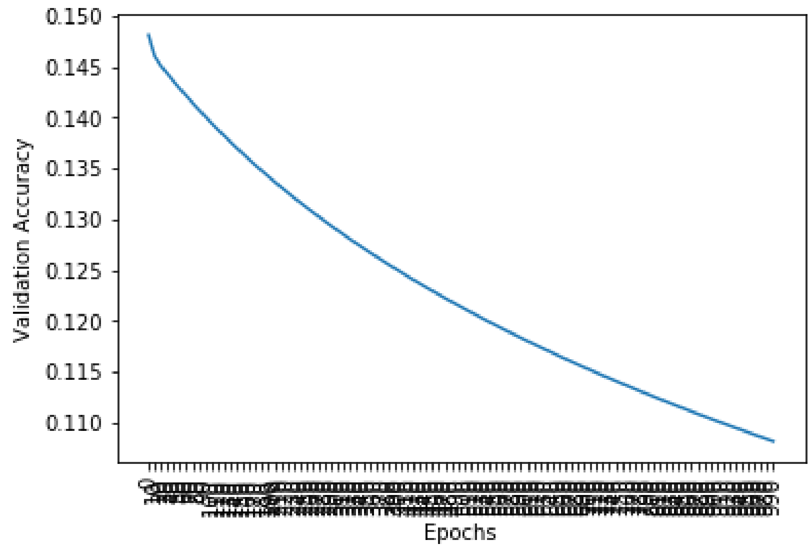
For learning rate=1 and change in the number of epochs from 100 to 1000 the training accuracy increases.

5.2 Validation Accuracy vs Epochs

1) Learning rate=0.5 and epochs=100
Validation accuracy: 94.11764705882354 %



2) Learning rate=0.5 and epochs=1000
   Validation accuracy: 98.03921568627452 %



For learning rate=0.5 and change in the number of epochs from 100 to 1000 the validation accuracy increases.

5.3 Accuracy, Precision and Recall on testing data

The values for accuracy, precision and recall are calculated by using the confusion matrix and finding out the values of True Positive (TP),True Negative(TN),False Positive(FP) and False Negative(FN). The values in the confusion matrix varies based on the epochs and learning rates.

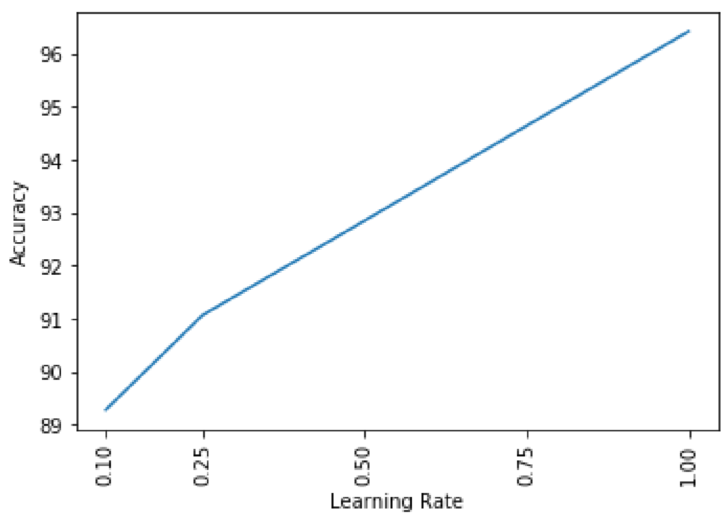
Accuracy=TP+TN/(TP+TN+FP+FN)
Precision= TP/(TP+FP)
Recall=TP/(TP+FN)

Accuracy:

For the testing data with different learning rates, iterations=100, the accuracy is as follows:

Learning Rate	Accuracy
0.1	89.28%
0.25	91.07 %
0.5	92.85 %
0.75	94.64 %
1	96.42%

The graph between learning rate and accuracy is as follows:



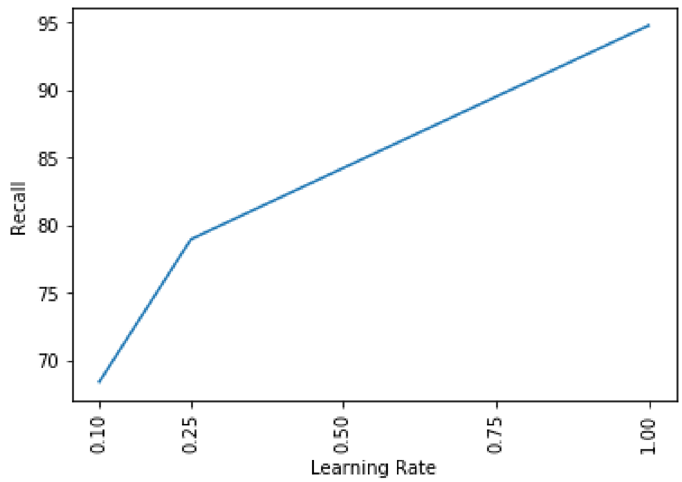
Based on the above table and the graph, it can be clearly said that the accuracy increases as the learning rate increases.

Recall:

For the testing data with different learning rates, iterations=100, the recall is as follows:

Learning Rate	Recall
0.1	68.42%
0.25	78.94%
0.5	84.21%
0.75	89.47%
1	94.73%

The graph between learning rate and recall is as follows:



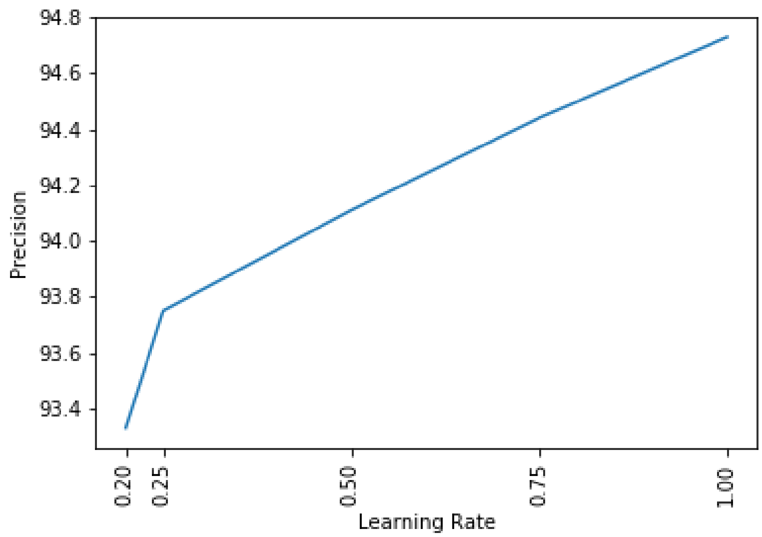
Based on the above table and the graph, it can be clearly said that the recall increases as the learning rate increases.

Precision:

For the testing data with different learning rates, iterations=100 the precision is as follows:

Learning Rate	Precision
0.20	93.33 %
0.25	93.75%
0.5	94.11 %
0.75	94.44%
1	94.73%

The graph between learning rate and precision as follows:



Based on the above table and the graph, it can be clearly said that the precision increases as the learning rate increases.

6 CONCLUSION

As per this experiment, the logistic regression for the given dataset has been implemented. The models are trained with different learning rates. It is observed that the training accuracy of the dataset increases when the epoch and learning rate increases. When the learning rate is increased to a very high value the accuracy values increase dramatically which is not ideal.

7 REFERENCES

*Used the docx file in the sit https://nips.cc/Conferences/2015/PaperInformation/StyleFiles
*Used the existing documentation as a reference https://www.researchgate.net/publication/311950799_Analysis_of_the_Wisconsin_Breast_Cancer_Dataset_and_Machine_Learning_for_Breast_Cancer_Detection

-