**DEPARTMENT OF ELECTRONIC AND TELECOMMUNICATION ENGINEERING**

**UNIVERSITY OF MORATUWA**



**EN3150 Assignment 02**

Learning from data and related

challenges and classification

**PUSHPAKUMARA H.M.R.M.**

**200488E**

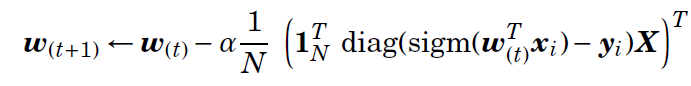
October 02, 2023

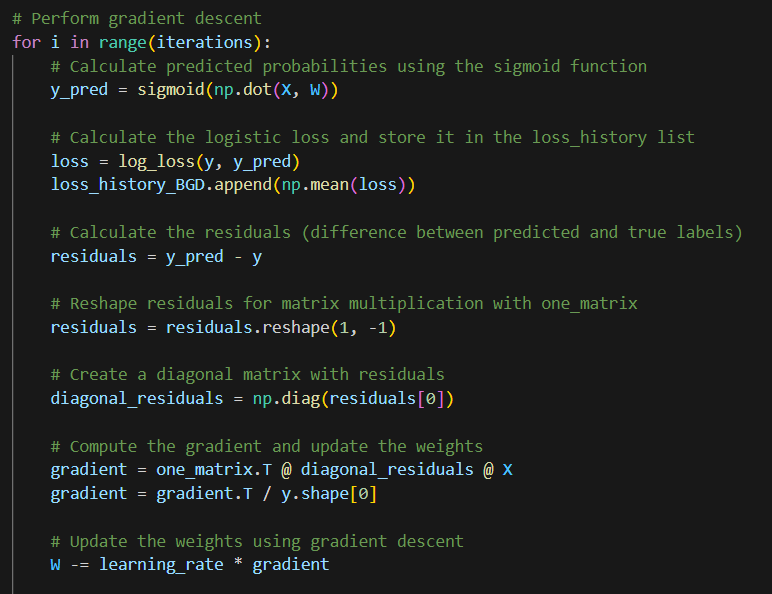
# 1. Logistic regression weight update process

This question asks us to perform a gradient descent-based weight update for the given data using binary cross entropy as a loss function. We are given the following:

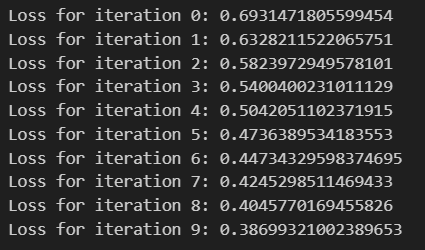
* Data matrix X of dimension N×(D+1), where N is the total number of data samples and D is the number of features.
* Learning rate α = 0.1
* Number of iterations t = 10

Batch gradient descent weight update is given as follows:





Losses after each iteration are as follows:

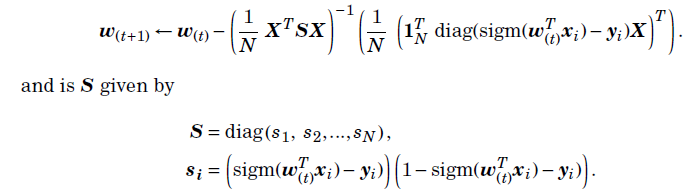


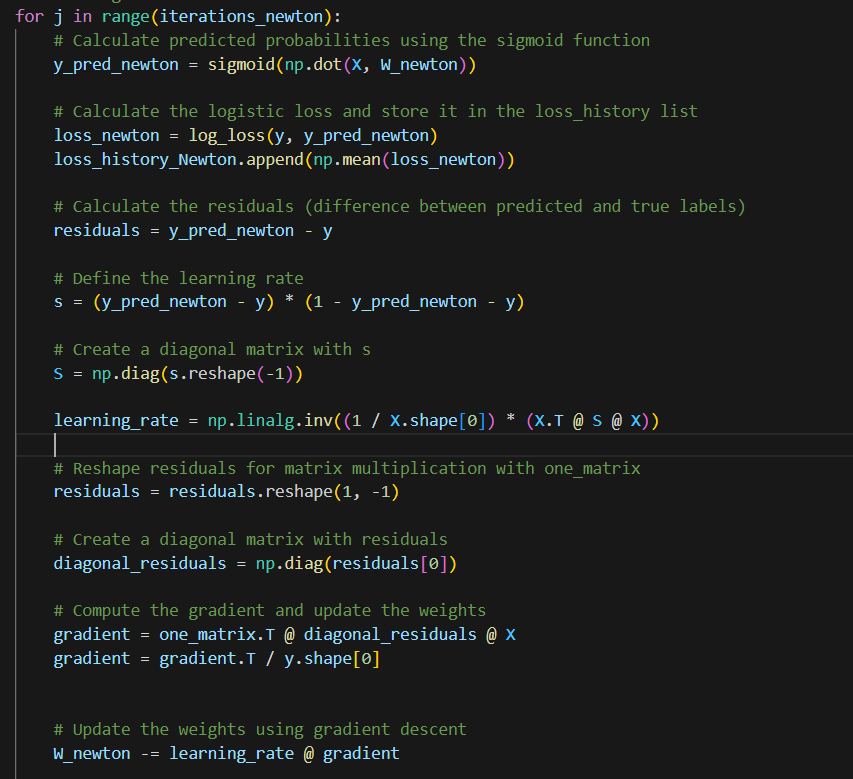
A graph with a blue line

Description automatically generated

This question asks us to perform weight updates using Newton's method for binary cross-entropy as the loss function. We initialize the weights as zeros and conduct 10 iterations to update them using this advanced optimization technique.

Batch Newton’s method weight update is given below.





Losses after each iteration are follows:

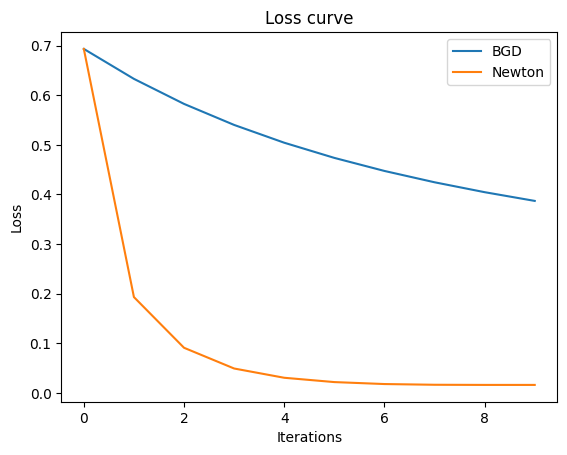
A screen shot of a computer

Description automatically generated

A graph with a line

Description automatically generated

Loss with respect to number of iterations for both Gradient descent and Newton method’s



In both approaches, the loss starts at a value of 0.6931 since the initial W values are zero.

In the Batch Gradient Descent approach, the loss decreases significantly from iteration 0 to iteration 5, indicating rapid early learning, and after the 5th iteration, the rate of decrease slows down. After the 10th iteration, the loss is approximately 0.3869.

In Newton’s method approach, the loss drops significantly from iteration 0 to iteration 1. It continued to decrease rapidly over the next few iterations, reaching a low loss value of approximately 0.013 after the 10th iteration.

When we compare these two approaches, we can identify that the Batch Gradient Descent shows a slower convergence rate when compared with Newton’s method. Newton’s method shows rapid convergence due to its ability to take curvature information into account.

# 2. Perform grid search for hyper-parameter tuning

This task involves using Lasso logistic regression for image classification with specific settings: 'penalty' set to 'l1', 'solver' set to 'liblinear', and 'multi\_class' set to 'auto'. To optimize the model's hyperparameter 'C', we created a machine learning pipeline that includes data scaling, the Lasso logistic regression estimator, and a parameter grid for hyperparameter tuning. We employ GridSearchCV to systematically explore a range of 'C' values spanning from 0.01 to 100 in order to find the optimal value for this hyperparameter. The goal is to identify the 'C' value that maximizes the model's performance for image classification.

A screen shot of a computer code

Description automatically generated

A black screen with white text

Description automatically generated

Plot of classification accuracy with respect to hyperparameter C

A graph with a line

Description automatically generated

**Confusion matrix**

A diagram of a confusion matrix

Description automatically generated

**Precision, Recall, and F1-score**

A black background with white numbers

Description automatically generated

* The confusion matrix provides a detailed breakdown of the classification results for each class. Each row corresponds to the actual class, while each column represents the predicted class.
* Precision (0.826) measures the accuracy of positive predictions. It indicates that out of all the samples predicted as positive, approximately 82.6% were correct.
* Recall (0.79) measures the ability of the model to correctly identify all relevant instances. It suggests that the model correctly identified about 79% of all positive instances.
* The F1-Score (0.794) is the harmonic mean of precision and recall and provides a balanced measure of a model's performance. An F1-Score of 0.794 indicates a good overall performance of the model.
* Overall, the model appears to perform well with relatively high precision, recall, and F1-Score values. However, further domain-specific considerations might be needed to assess whether these metrics meet the desired level of performance for the specific application.

# 3. Logistic regression

1. To estimate the probability that a student, who has studied for 40 hours (x1) and has an undergraduate GPA of 3.5 (x2), will receive an A+ in the class (y = 1),

we can use the logistic regression equation:

Where:

Now here:

We can calculate:

Then,

1. To achieve a 50% chance of receiving an A+ in the class,

The student has a 3.5 GPA. Therefore . We have to find ;