**Making of Perceptron**

Made perceptron by initializing all weights and biases to 0. Then iterated over the number of epochs and added a second loop inside the previous one to iterate over all training samples. In a separate predict function, calculated the prediction value using y = wx +b and implemented sigmoid activation to give binary values.

NAND Using Perceptron

Trained the classifier to learn NAND operation. Used a learning rate of 0.005 and 5 epochs to get accurate result.

Linear Regression using Python:

Used an advertising dataset and applied perceptron to draw a best-fit line. For this purpose, removed the activation function and directly took the values of w and b to draw the best fit line. The result gave me an RMSE of 6.525267280338032. Trained perceptron over 25 epochs with a learning rate of 0.005.
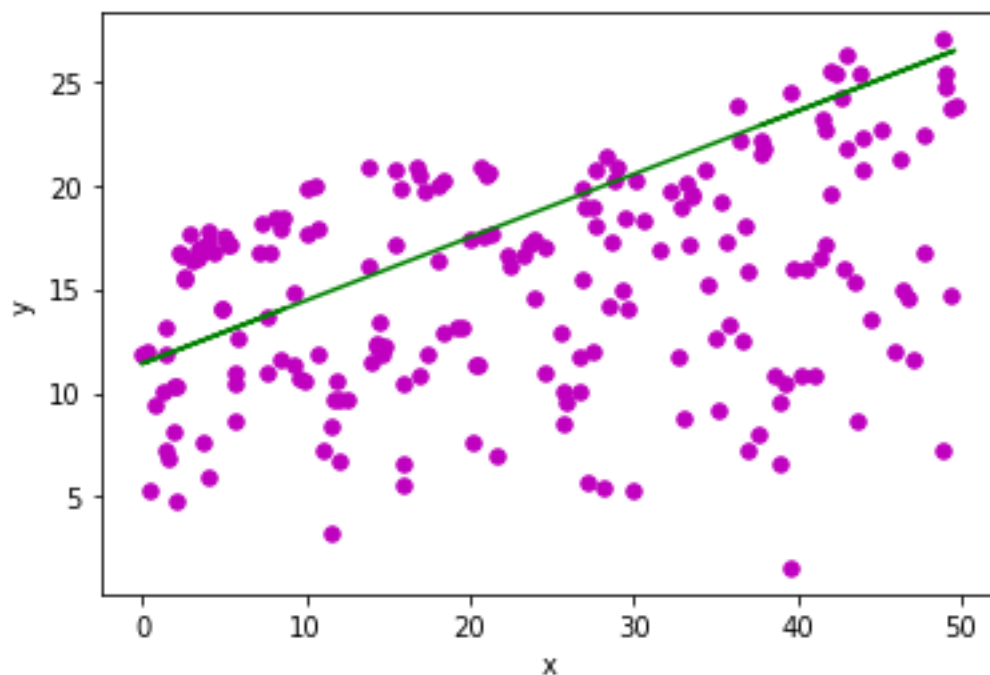
Classification using Python:

Used iris dataset for classification between iris-setosa and iris-versicolor. The prediction values and actual values are printed.

Confusion Matrix Generated:

```
[[37  0]
 [13 50]]

F-Score: 0.8849557522123894
Precision:  0.87
Recall 1.0
```