Task 1:ADALINE & SLP

Neural networks and deep learning

Department: SC

Team members:

1st member: Manar Ibrahim Abd-El-Wahed Sweilem Hadhoud.
ID: 20201700852
2nd member: Manar Mohamed Mahdy Meselhy
ID: 20201700854
3rd member: Manar Alaa Azoz Mostafa
ID: 20201700853
4th member: Malk Ismail Mohamed Mahmoud
ID: 20201700850
5th member: Mennatallah Ibrahim Kamal Khalil
ID: 20201700856
6th member: Fady Makram Migar Shedada
ID: 20201700562

Task parts:

1- Read data (selected features, selected classes, selected method, method parameters) from Gui.
2- Read data from excel sheet depending on data entered through Gui.
3- Perform preprocessing on the selected features and classes.
4- Implement NN methods (single layer perceptron, Adaline)
5- Output train line equation, accuracy, and confusion matrix

1- GUI:

- Enter two features.
- Enter two classes to perform classification between.
- Enter parameters needed to be initialized in the methods
  1- Epochs
  2- Learning rate
  3- MSE
  4- Bias to be included or not choice.
- Enter to be trained method.

2- Read excel sheet:

- Read specific columns according to selected features.
- Read specific rows according to selected classes.

3- Preprocessing:

I. Encoding

- Convert the categorical values of Two classes to

First class: -1
second class: 1

II. Scaling
- Normalization
- Standardization

4- Splitting data:
- Separate the rows of each class from the other one.
- Shuffle the rows of each class.
- Randomly select 30 rows from each class for training method
- Use the remaining rows for testing.

5- implement class for SLP:
- Training part
  - Initialize weights of two features by randomly selected number.
  - Initialize bias with random number in case it's decided to be used otherwise its value is 0.
  - Loop over rows of data selected for training and calculate the net value by values of features then calculate the actual value by signum function.
  - Check if the output value of signum function equals the actual value in the data.
    If not: update values of the weights
    Else: continue to the next row with the same weights.
  - Repeat the third step according to the epochs number entered by user in Gui.
- Draw the line equation with the final values of weights.
- Testing part:
  - Given x values of data selected for testing, predict the value of class (y) and compare it with the actual value.
  - Do this for all rows of the testing data and calculate four parts of the confusion matrix

TP: right predicted value cases for class 1
TN: right predicted value cases for class 2
FP: wrong predicted value cases for class 1
FN: wrong predicted value cases for class 2

by these values calculate accuracy of trained model.

5- implement class for ADALINE:

- Training part

  - Initialize weights of two features by randomly selected number.
  - Initialize bias with random number in case it's decided to be used otherwise its value is 0.
  - Loop over rows of data selected for training and calculate the net value by values of features then calculate the actual value by linear activation function.
  - Calculate the error for the current row and update values of the weights then calculate the total error as Adaline learn by stochastic gradient descent.
  - Repeat the third step according to the epochs number entered by user in Gui.

- Draw the linear decision boundary equation with the final values of weights.

- Testing part:

  - Given x values of data selected for testing, predict the value of class (y) and compare it with the actual value.
  - Do this for all rows of the testing data and calculate four parts of the confusion matrix.

TP: right predicted value cases for class 1
TN: right predicted value cases for class 2
FP: wrong predicted value cases for class 1
FN: wrong predicted value cases for class 2

by these values calculate accuracy of trained model.

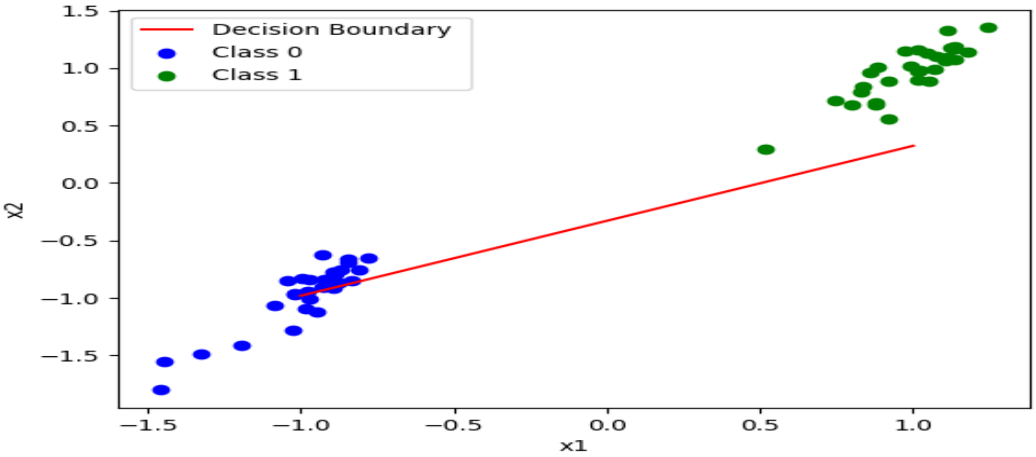
Results of models

classes possible combinations:

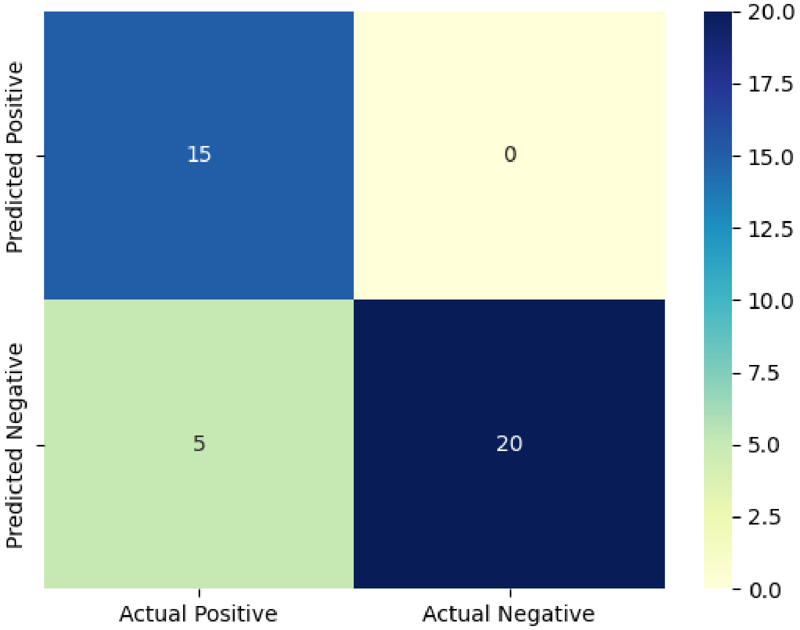
1- BOMBAY & CALI:

For both model…

- Highest accuracy = 87.5% by feature1: MajorAxisLength , feature2: Perimeter
- Parameters: learning rate: 0.0000001 , epochs: 100 , MSE : 0.5 , bias: True



decision boundary line by two models
weights: w1:1.00391556, w2: -1.5387499033471634, bias: 0.23641754350293442

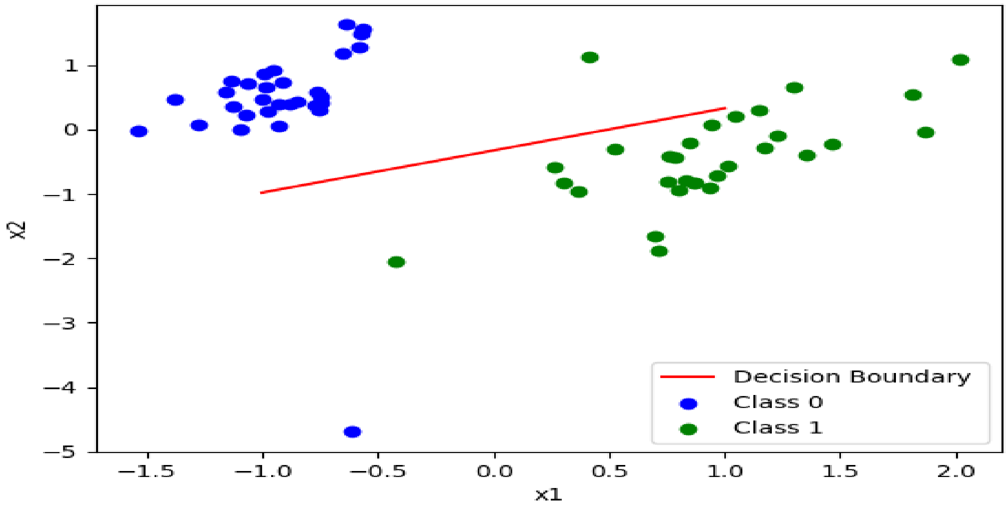


Confusion matrix ----> TP: 15 FP: 0 FN: 5 TN: 20

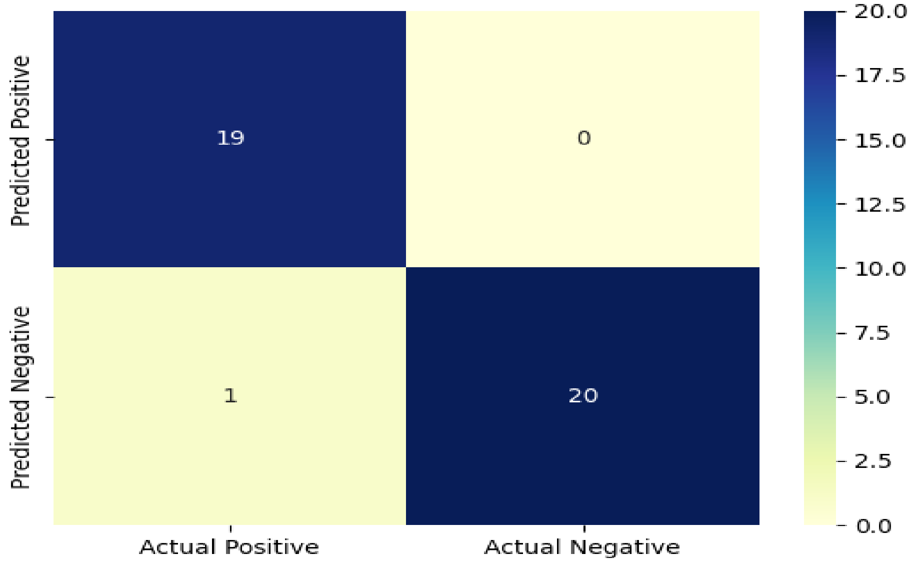
2- *SIRA & CALI:*

For both model…

- Highest accuracy = 97.5% by feature1: MinorAxisLength, feature2: roundnes
- Parameters: learning rate: 0.0000001, epochs: 100, MSE: 0.5, bias: True



decision boundary line by two models
weights: w1: -1.5387499, w2: 1.0039155603, bias: -0.50570106692

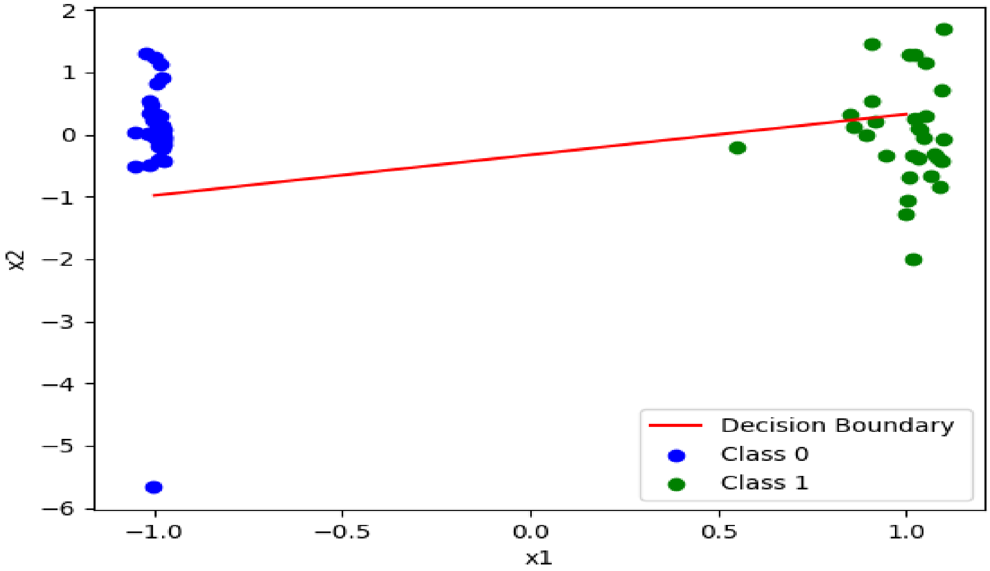


Confusion Matrix -------> TP: 19 FP: 0 FN: 1 TN: 20

3- *BOMBAY & SIRA*:

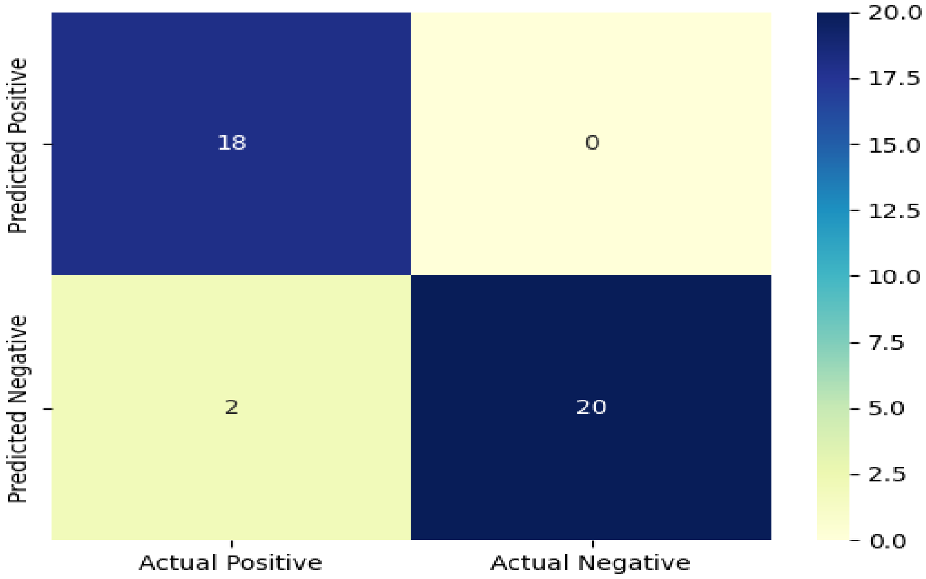
For both model…

- Parameters: learning rate: 0.0000001, epochs: 100, MSE: 0.5, bias: True
- Highest accuracy = 95%



by feature1: Area, feature2: roundness

decision boundary line by two models
weights: w1: -1.5387499, w2: 1.0039155603, bias: -0.50570106692



Confusion Matrix -------> TP: 18 FP: 0 FN: 2 TN: 20

by feature1: Perimeter, feature2: roundness ---> same results

by feature1: MinorAxisLength, feature2: roundness ----> same results

Conclusion:

Both model “SLP” & “ADALINE” give the same results of accuracy but not the same weights.
Both models are used in binary classification task.
The most suitable feature for training for classification task between *BOMBAY & SIRA is* roundness, for *SIRA & CALI as well.*
For BOMBAY & CALI the most suitable features are MajorAxisLength and Perimeter.