**Group26\_HW1**

**Group member:**

Beshay, Sandy

Fana, Phoebe

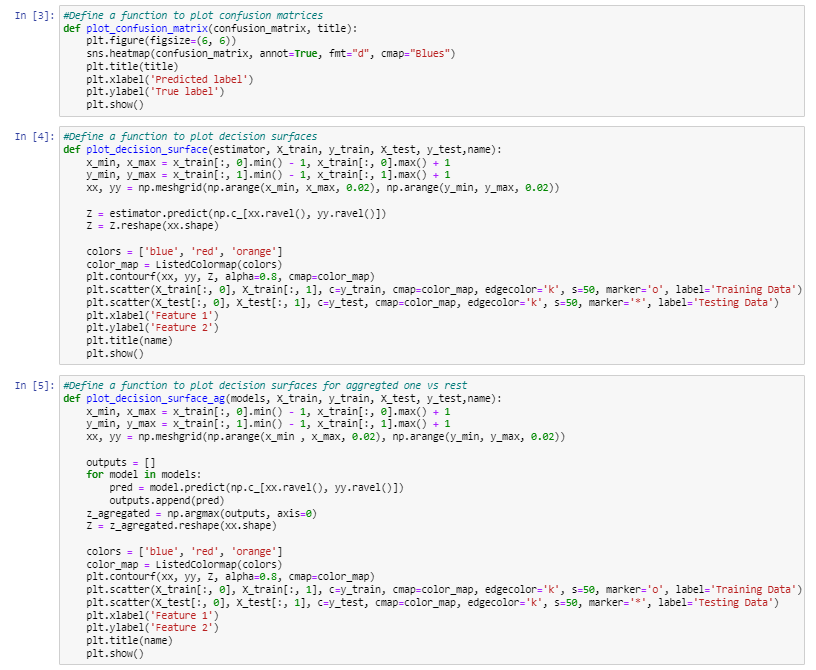
Ali, Muhammad

**Problem 1:**

A)

* First, we import the important libraries we use.
* Then we used the training and testing data.
* Then we built a three function for drawing.

(Confusion matrices, decision surfaces, decision surfaces aggregated one vs rest)



* Then we trained the SVM model, and the accuracy was 100%.
* Then we obtain confusion matrices for training and testing datasets.

A picture containing text, screenshot, diagram, rectangle

Description automatically generatedA picture containing text, screenshot, diagram, rectangle

Description automatically generated

* Then we visualize decision surfaces for multi-class classification.

A picture containing text, screenshot, diagram

Description automatically generated

B)

* We label training and testing data for one vs rest as binary classifier.

A picture containing text, font, screenshot, line

Description automatically generated

* Then we use the one-vs-rest strategy for SVM.A picture containing text, screenshot, font, document

  Description automatically generated
* Then we get accuracy, Confusion matrices, decision surfaces for each class (0,1,2).
* The result of class0:



A picture containing text, screenshot, diagram, rectangle

Description automatically generatedA picture containing text, screenshot, diagram

Description automatically generated

A picture containing text, screenshot, line, plot

Description automatically generated

* The result of class1:



A picture containing text, screenshot, diagram

Description automatically generatedA picture containing text, screenshot, diagram, rectangle

Description automatically generated

A picture containing text, screenshot, diagram, line

Description automatically generated

* The result of class2:



A picture containing text, screenshot, diagram

Description automatically generated A picture containing text, screenshot, diagram, rectangle

Description automatically generated

A picture containing text, screenshot, line, diagram

Description automatically generated

* Then we repeat all this steps with Perceptron.
* Then we use the one-vs-rest strategy for Perceptron.A screenshot of a computer program

  Description automatically generated with medium confidence
* Then we get accuracy, Confusion matrices, decision surfaces for each class (0,1,2)
* The result of class0:



A picture containing text, screenshot, diagram

Description automatically generated A picture containing text, screenshot, diagram

Description automatically generated

A picture containing text, screenshot, line, plot

Description automatically generated

* The result of class1:



A picture containing text, screenshot, diagram, rectangle

Description automatically generated A picture containing text, screenshot, diagram, rectangle

Description automatically generated

A blue and orange graph

Description automatically generated with low confidence

* The result of class2:



A picture containing text, screenshot, diagram

Description automatically generated A picture containing text, screenshot

Description automatically generated

A picture containing text, screenshot, diagram, line

Description automatically generated

* Then we Compare and analyze SVM and Perceptron results:

by plots that compare the accuracy between (SVM and Perceptron).

A screenshot of a computer program

Description automatically generated with low confidence

* We draw one plot for train dataset and one for test dataset accuracy prediction.

A picture containing text, line, screenshot, plot

Description automatically generatedA picture containing text, screenshot, line, plot

Description automatically generated

* The plots show that SVM has better results than Perceptron.

C)

* Then we Aggregate results from the one-vs-rest strategy for SVM.

A picture containing text, font, screenshot

Description automatically generated

* Then we get accuracy, Confusion matrices, decision surface for aggregated svm classes.



A picture containing text, screenshot, diagram, rectangle

Description automatically generated A picture containing text, screenshot, diagram, rectangle

Description automatically generated

A picture containing text, screenshot, diagram, line

Description automatically generated

* Then we Aggregate results from the one-vs-rest strategy for perceptron.

A picture containing text, screenshot, font

Description automatically generated

* Then we get accuracy, Confusion matrices, decision surface for aggregated svm classes.



A picture containing text, screenshot, diagram, rectangle

Description automatically generated A picture containing text, screenshot, diagram

Description automatically generated

A picture containing text, screenshot, line, plot

Description automatically generated

* we compere the SVM aggregated classes and default SVM, then we discover that the accuracy is very close but default SVM is better than SVM aggregated classes , because default SVM accuracy is 100%.

D)

* The default model of SVM is a linear model, which means that it separates the data points by a straight line. This is a simple and efficient model, but it can be too simple for some data sets. In these cases, the aggregated performance of SVM can be better than the default model.
* Then we Refine the default SVM by selecting the appropriate parameter and train the model.

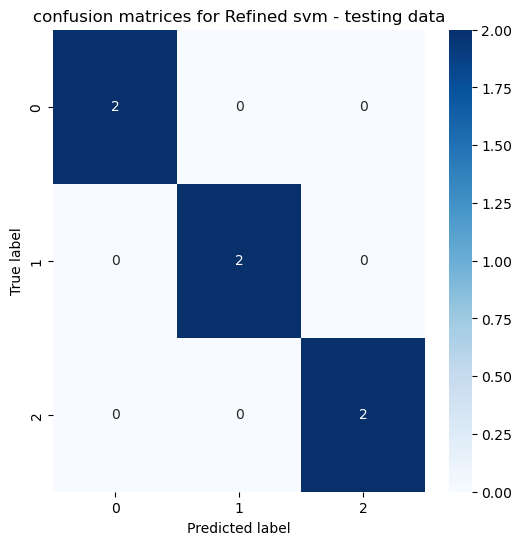
A screenshot of a computer

Description automatically generated with medium confidence

* Then we get accuracy, Confusion matrices, decision surface for refined SVM.



A picture containing text, screenshot, diagram, rectangle

Description automatically generated

A picture containing text, screenshot, diagram

Description automatically generated

* We compere the refined SVM and default SVM, then we discover that the two model have the same accuracy , but the decision surface for refined SVM is more accurate .

**Problem 2:**

**A)**

* First, we import the important libraries we use.
* Then we load the dataset from a CSV file and define the column names.

A screenshot of a computer

Description automatically generated with medium confidence

* Then we shuffle and split the dataset into a training set with 1000 samples, a validation set with 300 samples, and a testing set with 428 samples.

A picture containing text, font, screenshot, line

Description automatically generated

B)

* Then we change string values into numeric by Encoder after divide the data.

A close-up of a white rectangular object

Description automatically generated with low confidence

C)

* Then we use different number of training samples to show the impact of number of training samples and Use 10%, 20%, 30%, 40%, 50%,60%, 70%, 80%, 90% and 100% of the training set for 10 separate KNN classifiers and show their performance (accuracy score) on the validation set and testing set.

A screenshot of a computer program

Description automatically generated with medium confidence

A picture containing text, diagram, line, plot

Description automatically generated

D)

* Then we use 100% of training samples, to find the best K value, and show the accuracy curve on the validation set when K varies from 1 to 10 and the best k value is 5.

A picture containing text, font, screenshot

Description automatically generated

A picture containing text, diagram, line, plot

Description automatically generated

E)

* The conclusion:
* When the percentage of training data is increased, the accuracy increases.
* We should train the KNN model over all available k value to choose the best k value that suits our dataset, in our dataset the best k value is five.