ASSIGNMENT 1 PERCEPTRON

CISC/CMPE 452/COGS400/CISC874

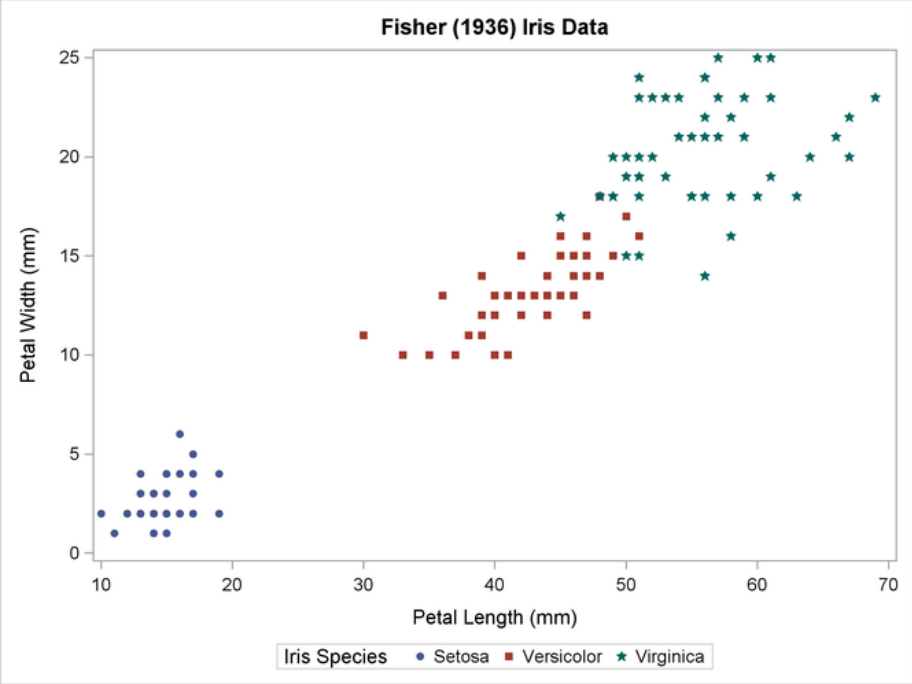
General Instructions for Code and Submission (for all assignments)

1. You can use any programming language (preferred Python, C, C++, Java – No GUI tool should be used)
2. Make one zip file named as Asgn1_studentID which should include
   a. *A text file containing the ANN design, accuracy in precision, recall, and confusion matrix*
   b. *Program code with comments in the code to explain what each of your program files and functions are for*
3. Upload zip file to the OnQ site

Assignment 1: The Data

- Iris data is one of the most popular datasets in machine learning tasks. Iris is a flowering plant with 260-300 different species.
- Iris data contains the measurements of 3 different species of Iris: Setosa, Versicolour and Virginia.
- The plot on the next slide shows the relation between Petal width and length.
- The data for this assignment is provided in two files: iris_train.txt and iris_test.txt.
  - *Each line in the files represents a separate data point where each data point includes the following comma separated values:*
    - Sepal length (cm), Sepal width (cm), Petal length (cm), Petal width (cm), class label
    - Example: 5.1,3.5,1.4,0.2,Iris-setosa
  - *Iris_train.txt contains 40 data points for each class. Use this to train your perceptron.*
  - *Iris_test.txt contains 10 data points for each class. Use this to test that your perceptron works correctly.*

Assignment 1 - Plot of Iris data (Petal width and length)



Assignment 1 –**Total 10 marks**

1. Program, Design and train a perceptron from scratch (**Do not use perceptrons implemented in other libraries**) using the data in iris_train.txt to predict the three classes of Iris using
   a. ***error correction learning with pocket algorithm.***

   Use iris_test.txt to test the accuracy of your ANN.
2. Submit the following
   a. *Program code with comments, any associated libraries needed to execute your code (and/or executables and instructions for execution) and the output labels generated by your code for the test data (i.e., test data with generated labels).*
   b. *Implement a function to create an output text file similar to the iris_test.txt file with your predicted output labels instead of the given class labels. Submit the text file.*
   c. *A report text/WORD file describing the structure of the ANN (diagram), parameter values used and accuracy of the model in terms of precision and recall for each class and all classes of data in the iris_test.txt file and a confusion matrix.*
3. Mark distribution – 7 marks
   a. *4 marks - for fully functional executable code (3 marks) and the output text file (1 mark)*
   b. *3 marks - for the report text file with ANN structure (0.5 mark), parameter values such as the initial and final weight vectors, training iterations and terminating condition, and other value choices made (1.5 marks), precision-recall and confusion matrix (1.5 marks).*

Use one of the following tools - 3 marks

- 2 marks - Use TensorFlow (with Keras) and Google Collab environment to create, train and test an ANN model. Explore built-in libraries for creating a perceptron, train and test the model.
- 1 mark - Submit a report file:
  - *Design choices for structure, function and parameter values used to create and train the model. (0.5 marks)*
  - *Compare the precision and recall values of the two ANNs (tool-based and your code) in the text file. (0.5 marks)*