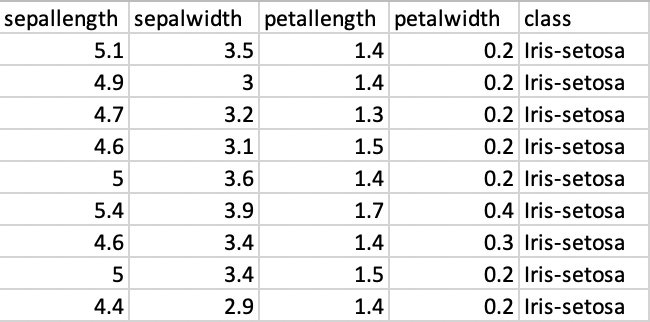
Machine learning cybersecurity

**iris DATASET classification**

# LAB 3: Writing a calssifier for iris plant

**Lab Description:** This lab is to classify the class of iris plants based on their features.

* The dataset contains 150 instances for iris plants with 4 features and 3 classes



* Use WEKA to input the iris dataset and perform the classification
* Write a python script based on sklearn library to implement the classifiers
* Write a python script based on Tensorflow framework to implement the classifier.

**Lab Environment:**

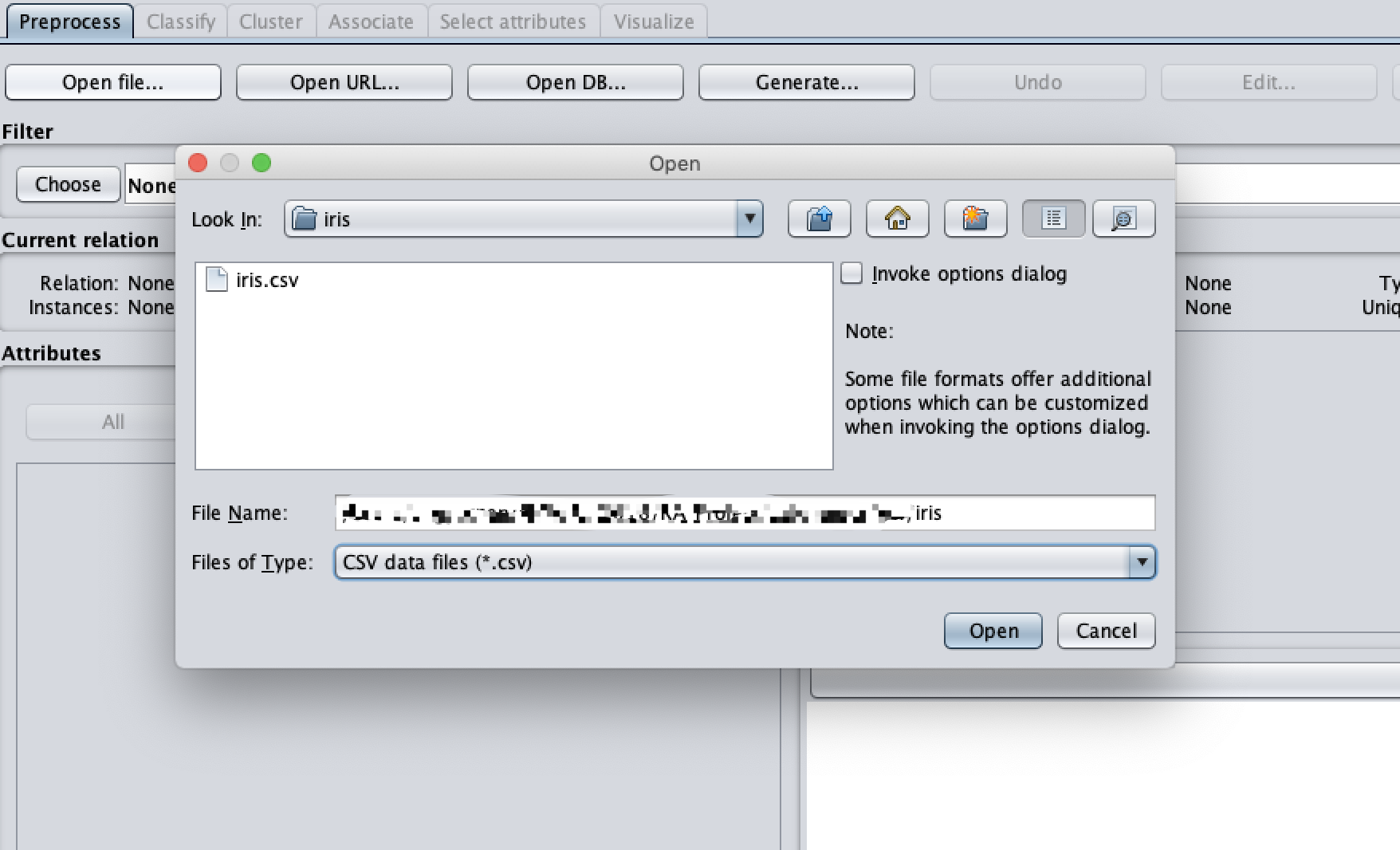
* The students should have access to a machine with Linux system or Windows system
* WEKA should be installed
* The environment for python is required as well as some packages such as numpy, tensorflow and sklearn.

**Lab Files that are Needed:**

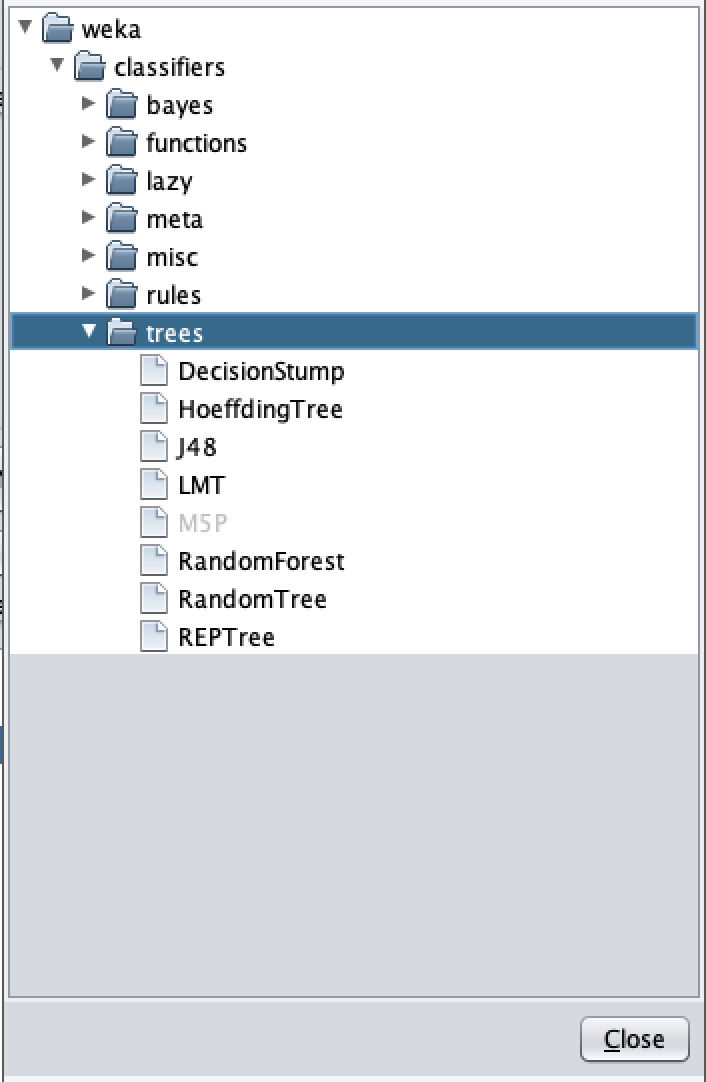
* For this lab you will need only one file (iris.csv) for both WEKA and python script.
* The last column is the class value, others are the features.

### **Lab exercise 1**

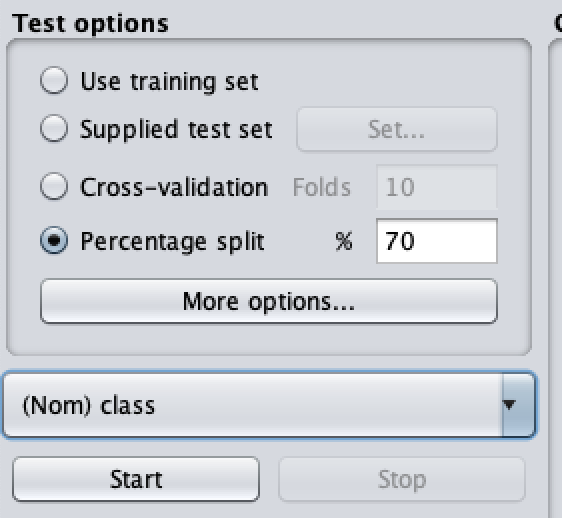
* Import data into WEKA (explorer), the files of type should be specified (csv).



* Choose a proper classifier, such as RandomForest

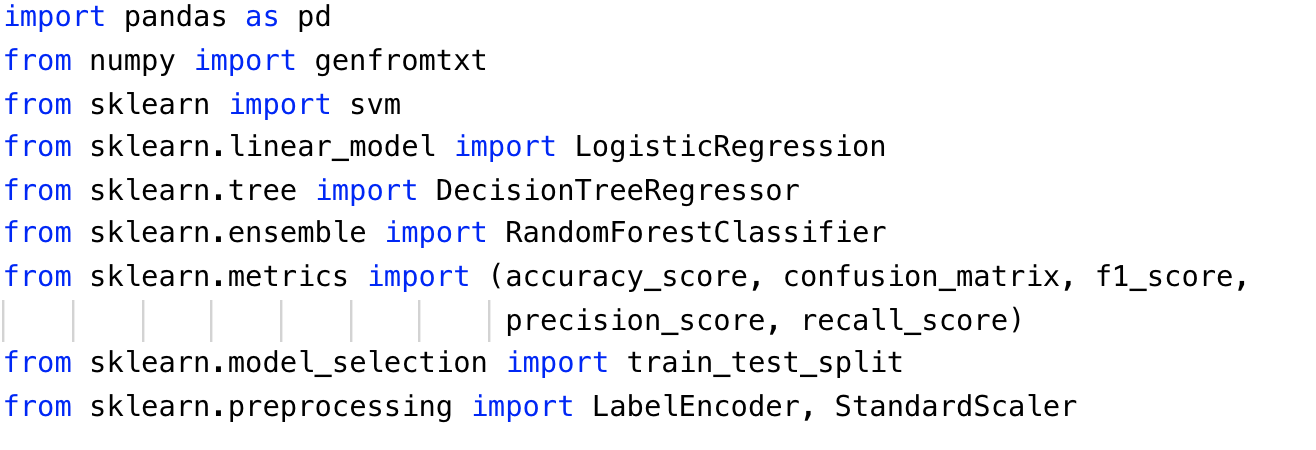


* Specify the test option and the column of class

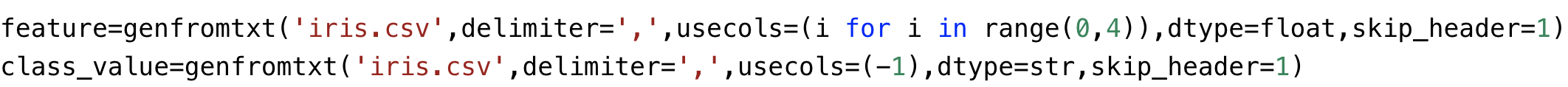


### **Lab exercise 2**

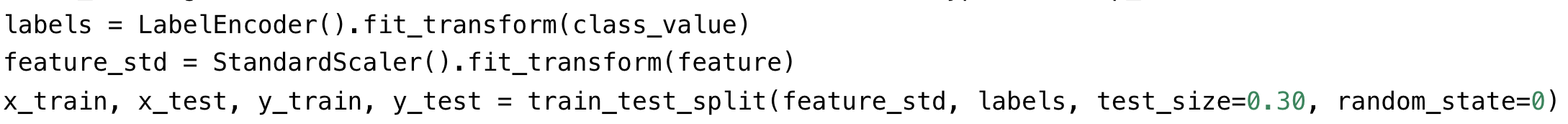
* Iris dataset is a basic but representative dataset for machine learning.
* In this exercise, you need to implement several classifiers with the use of sklearn.
* Import the required libraries



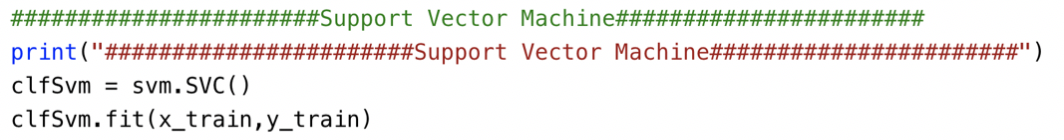
* Read the features and class values from iris dataset with proper method



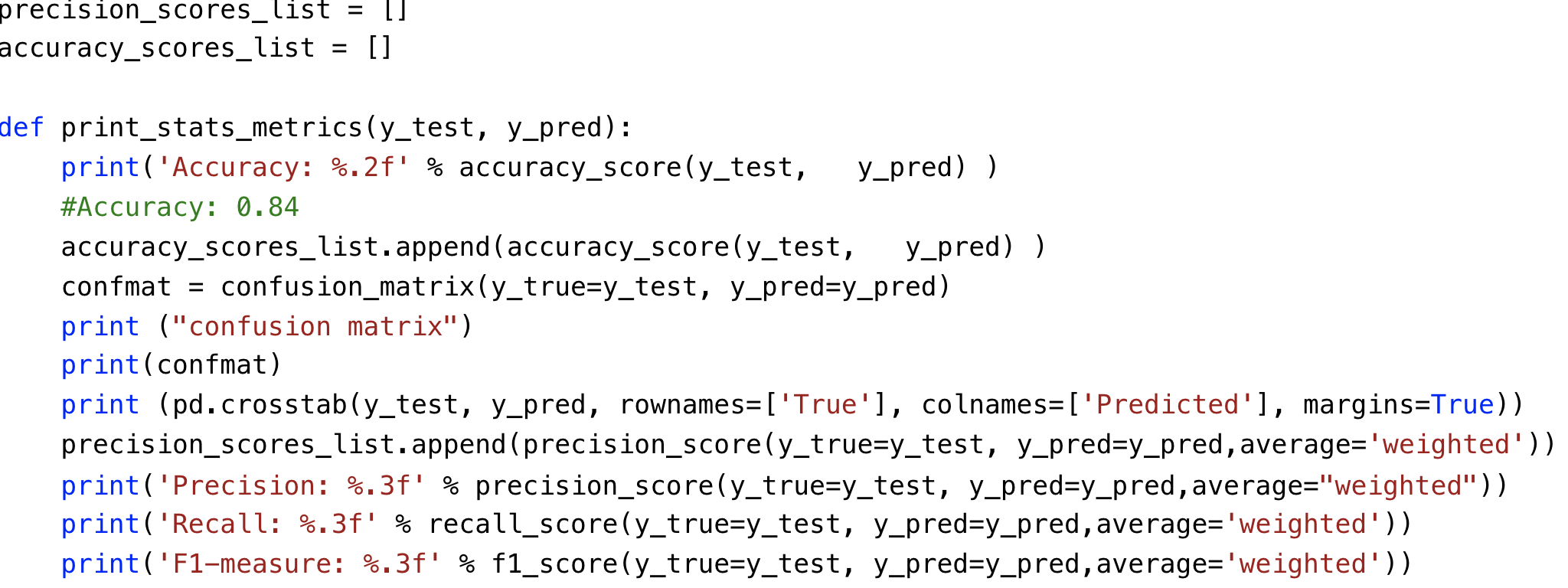
* iris.csv is the name of the file.
* delimiter indicates the character to split the data in a row.
* usecols indicates which columns will be read. For features, the first 4 columns of the rows will be read. For class values, the last columns of the rows will be read.
* dtype indicates the type of data to read
* Since the first line of the file is names for each column, we set skip\_header to 1 to avoid read the first row.
* You may need to create labels for each of iris classes and then split the dataset. When you finish the preprocess step, you can write the python script with the use of sklearn package to build your architecture of classifier.



* random\_state is the seed used by the random number generator



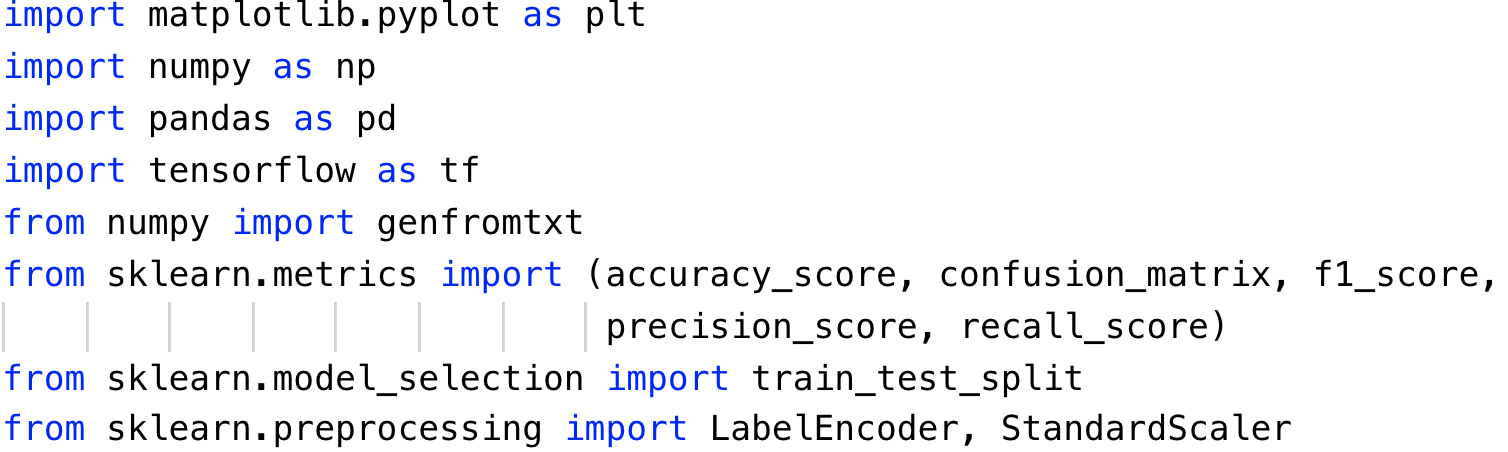
* Please print the statistics metrics such as accuracy, recall, precision and f1 score.



* Implement the classifiers based on Logistic Regression, Decision Tree, Support Vector Machine and Random Forest

### **Lab exercise 3**

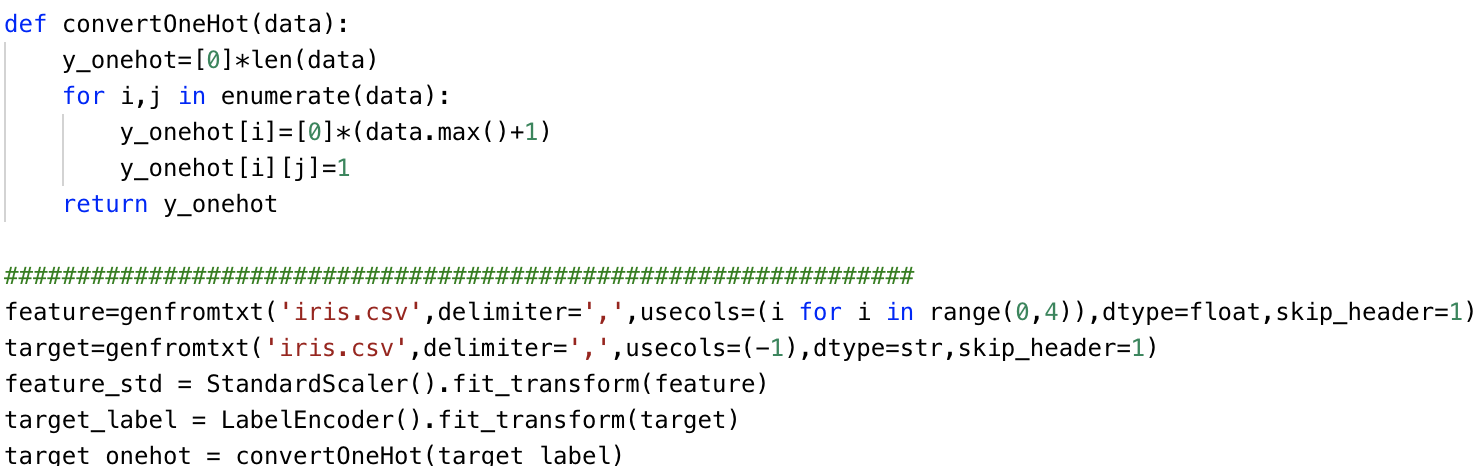
* Use the same data you use in the exercise 1 and 2.
* In this exercise, you will implement an artificial neural network classifier based on Tensorflow
* Import the required libraries



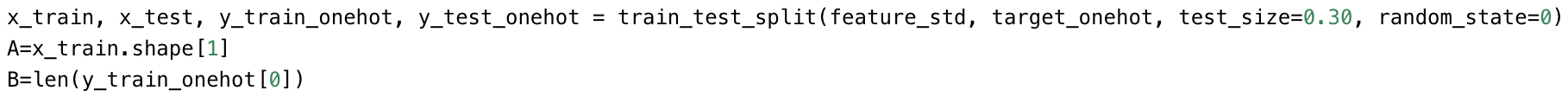
* Repeat the same steps to preprocess the data as Exercise 2. Read the data, standard scale the feature and encode the labels.
* Define the learning rate and number of epochs for artificial neural network



* An extra step in preprocess is to perform the one-hot encoding for the labels.



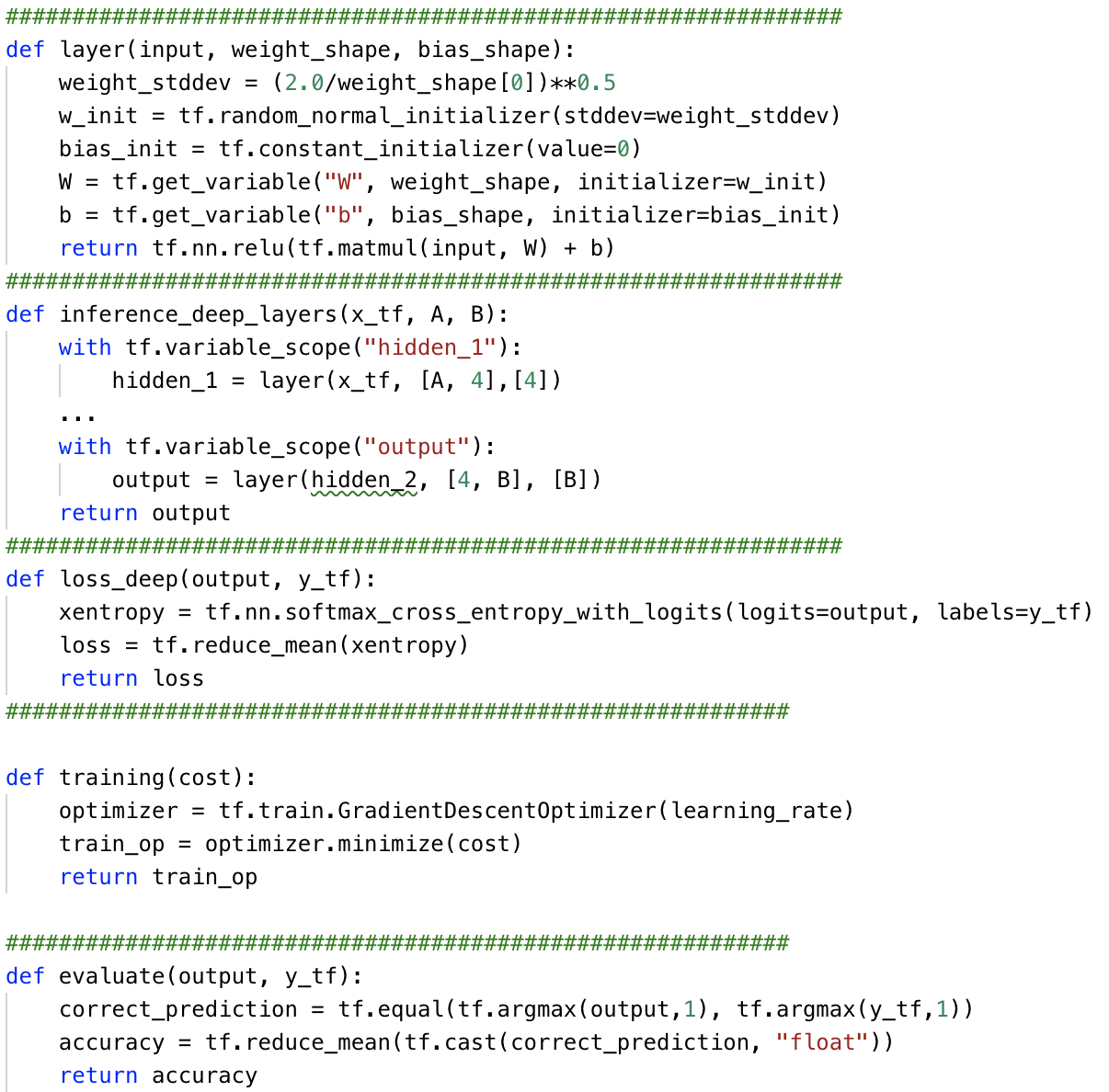
* Split the dataset after preprocessing and define the parameters to store the shape of placeholder.



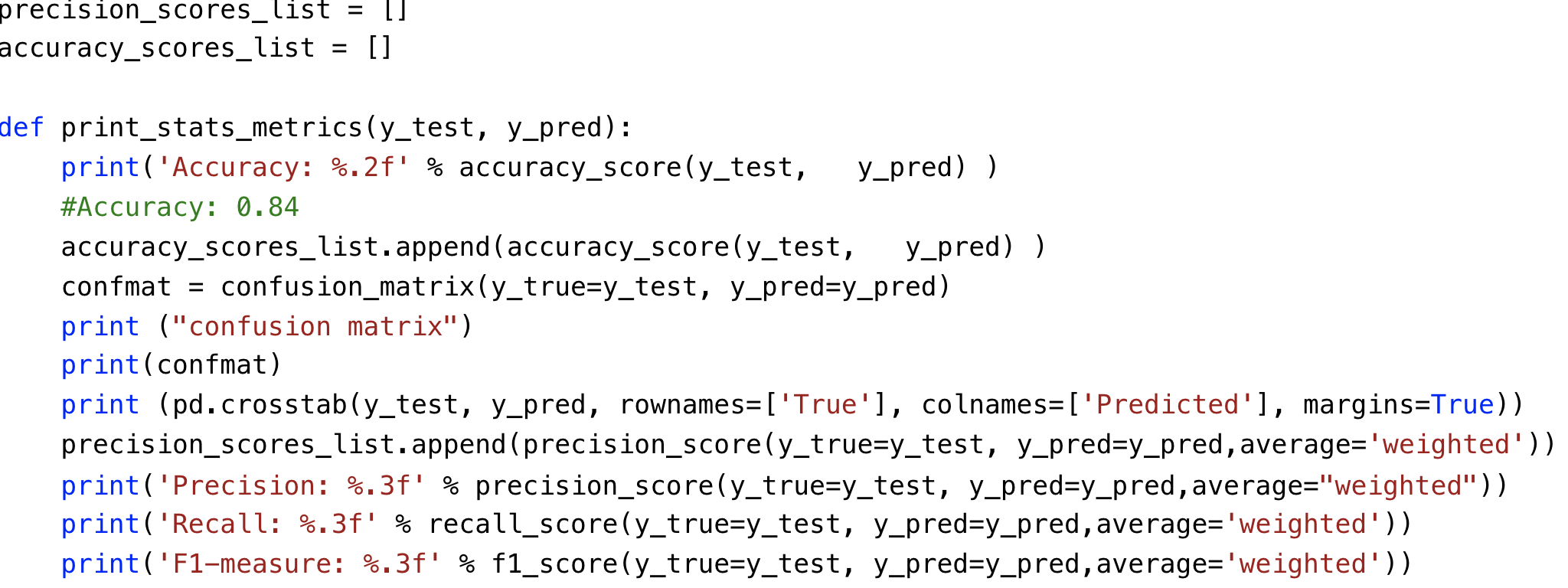
* Define the function to draw the plot of performance



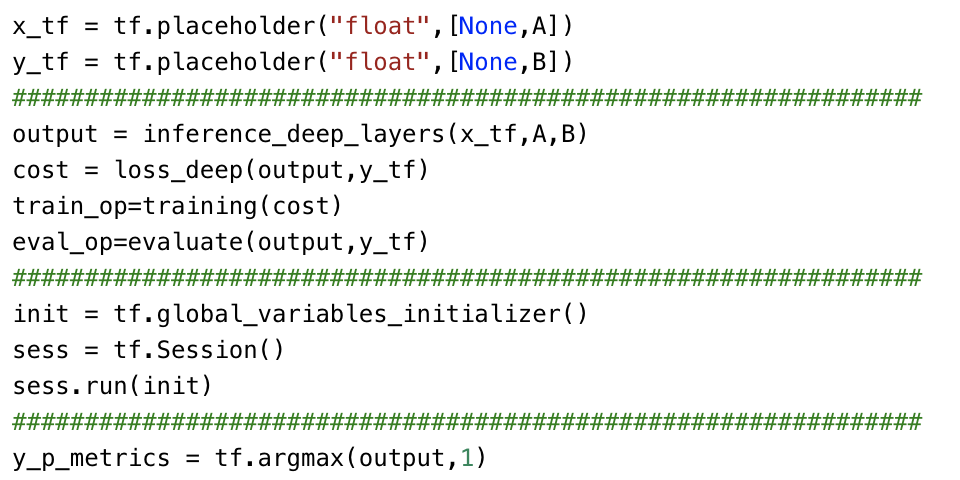
* Define your own architecture of neural network



* Please print the statistics metrics such as accuracy, recall, precision and f1 score.



* Initialize the variables and placeholders. Then perform the training and testing on iris dataset.



## What to Submit

You should submit a lab report file which includes:

* + The steps you preprocessed data
  + The necessary code snippet of your classifier and architecture.
  + The screenshot of the results
  + You can name your report "Lab3\_iris\_yourname.doc".