**Objective**:

To Ensure correct installation and configuration of Python 3.6 and to gain familiarity with the environment, python language and libraries. To write a python program that uses the Iris dataset provided in the scikit-learn package to train a k-nearest neighbor classifier and to plot the data using the matplotlib graphics library.

**Procedure:**

1. Download Anaconda 5.0.1 for windows and use Spyder IDE to write the python program.

2.Load the Irisdata using Pandas library.

3. Using Scikit learn train the K-nearest neighbor classifier (K=5 in my case)

4. Using the Matplotlib graphics library plot the decision boundaries and training sets

**Technology used:** Python. Libraries Used are pandas, numpy, sklearn, matplotlib

**Tool used:** Spyder

**Source Code Filename:**

Classification\_Irisdata.py

**Program Explanation:**

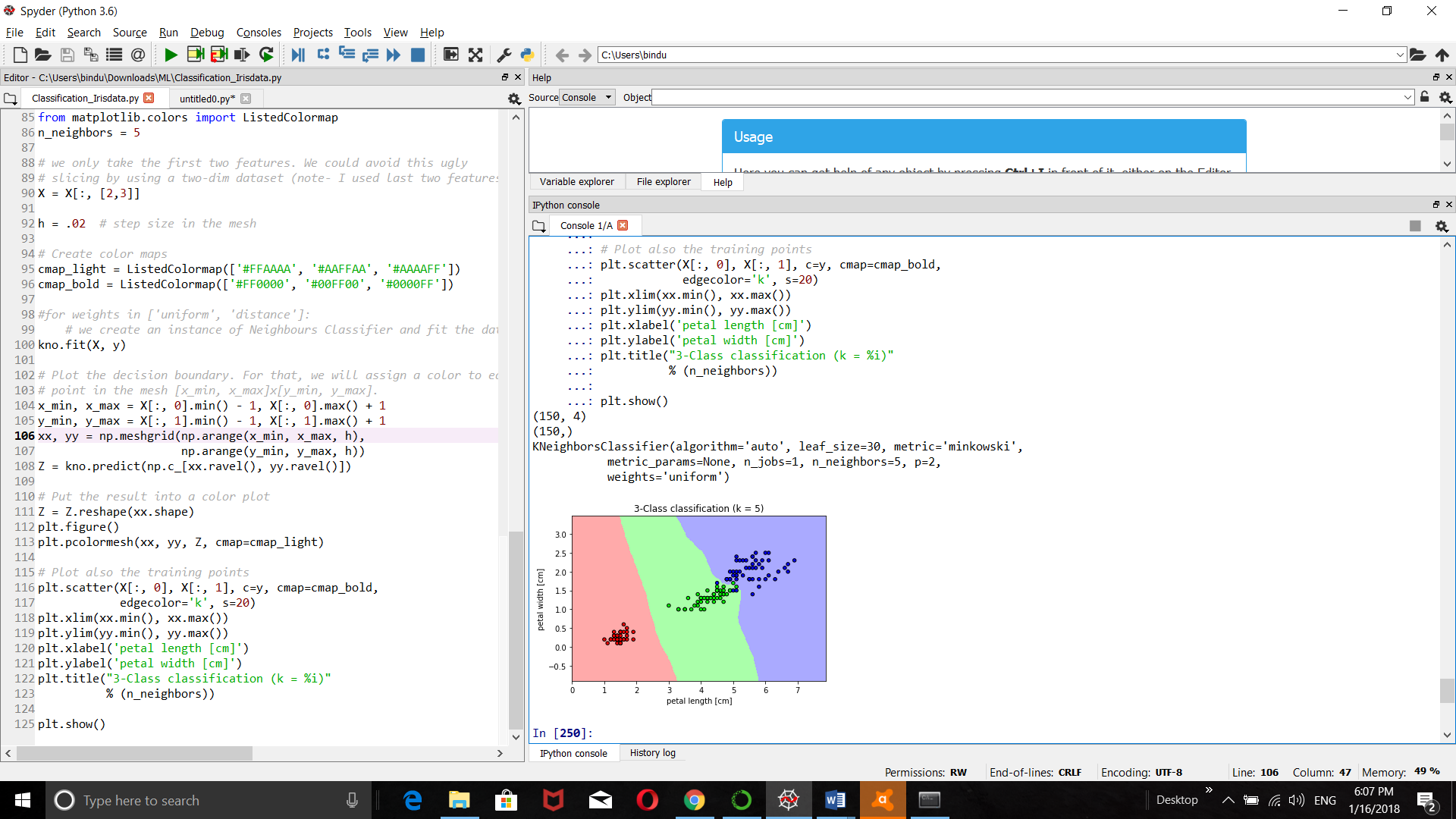
1. The program is to load the Iris data, train the model and plot the decision boundaries, training set using pandas, scikit learn and matplotlib packages respectively.
2. The program takes csv file as input (the Iris data that is downloaded from the official website).

Link to download: https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data

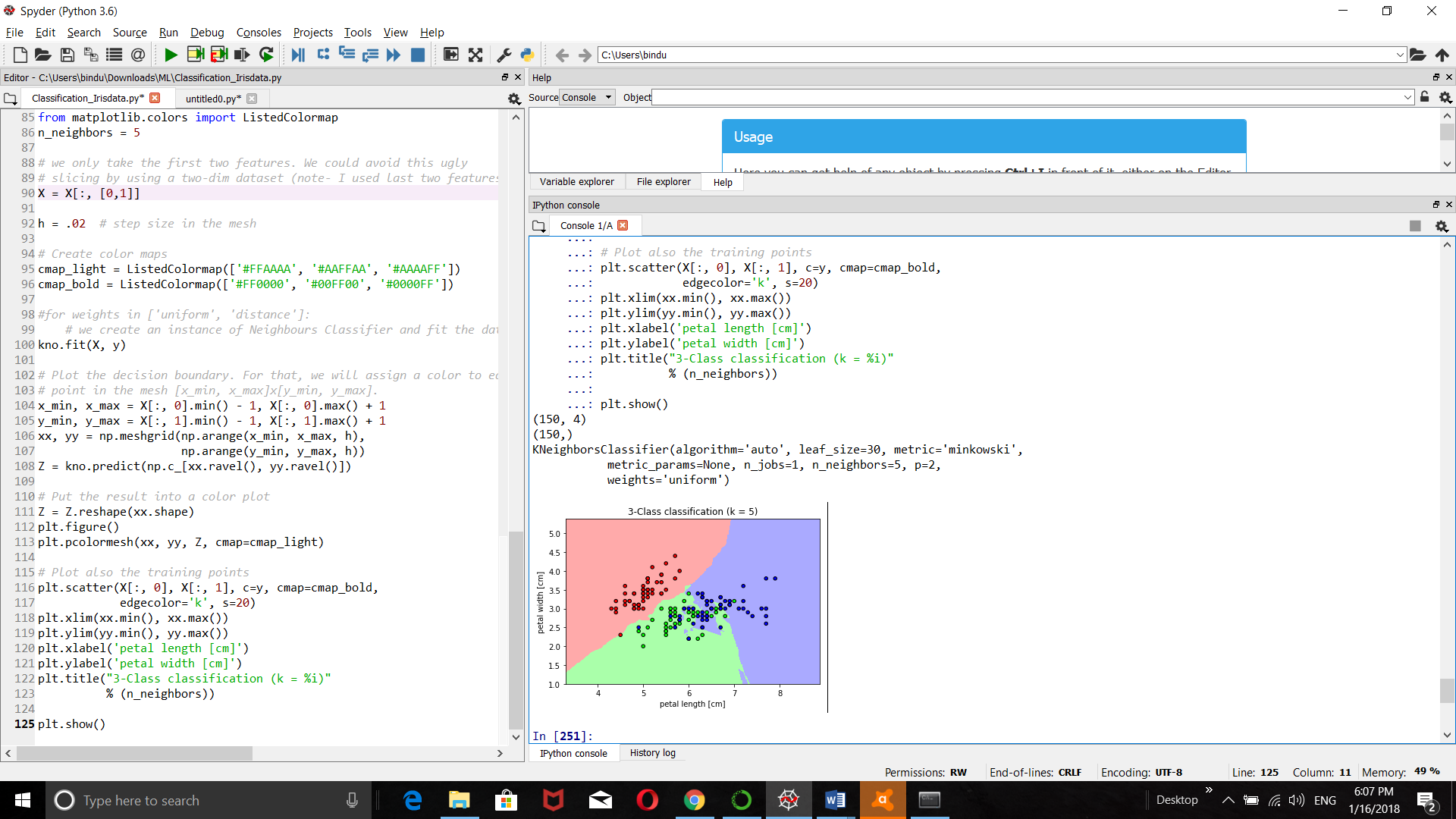
Import this from the excel and changes the column names as sepal length, sepal width, petal length, petal width, target and save it as .csv file

1. Or else we can directly use the Iris dataset from the Scikit learn sample datasets. (code for this alternative is commented in the program for reference).
2. Unlike the dataset that is available from the scikit learn Our dataset that is downloaded from the site will be different so use .as\_matrix method to convert the first 4 columns data as matrix and last column will be the target
3. convert the target in the string format to float format to apply the classifier. To do it we will be using a map that converts 'Iris-setosa' to 0.0, 'Iris-versicolor' to 1.0, 'Iris-virginica' to 2.0
4. Now the dataset is ready to train the nearest neighbor classifier. I chose K=5 because it has high testing accuracy.
5. For plotting I have used a part of code from the official scikit learn website. The URL for the link is: <http://scikit-learn.org/stable/auto_examples/neighbors/plot_classification.html#sphx-glr-auto-examples-neighbors-plot-classification-py>
6. The above code used the sepal measurements here in my code petal measurements are used
7. Finally, the decision boundaries and training sets are plotted.

**Screen Shots of output:**

1.Where x and y-axis are two features petal length and petal width respectively. 

2.Where x and y-axis are two features sepal length and sepal width respectively.



**Conclusion:**

Python 3.6 (Anaconda 5.0.1) has been successfully installed and configured in the pc. Got familiar with the python language and libraries.