**Python Lab Assignment – 2**

**Name – Aravind Sheri Class ID – 31**

**Name – Rajiv Varma Class ID – 04**

**Team – 19**

1. **Implementing a Model using Linear Discriminant Analysis (LDA):**

**Introduction:** Linear Discriminant Analysis classifier is used to perform a supervised dimensionality reduction, by maximizing the space between the classes. Getting the test and train data and apply the LDA algorithm to plot the classes of the data for the prediction.

**Objectives:** Get the data from by any source and form a prediction model using the algorithm Linear Discriminant Analysis so that we can make predictions from the data provided.

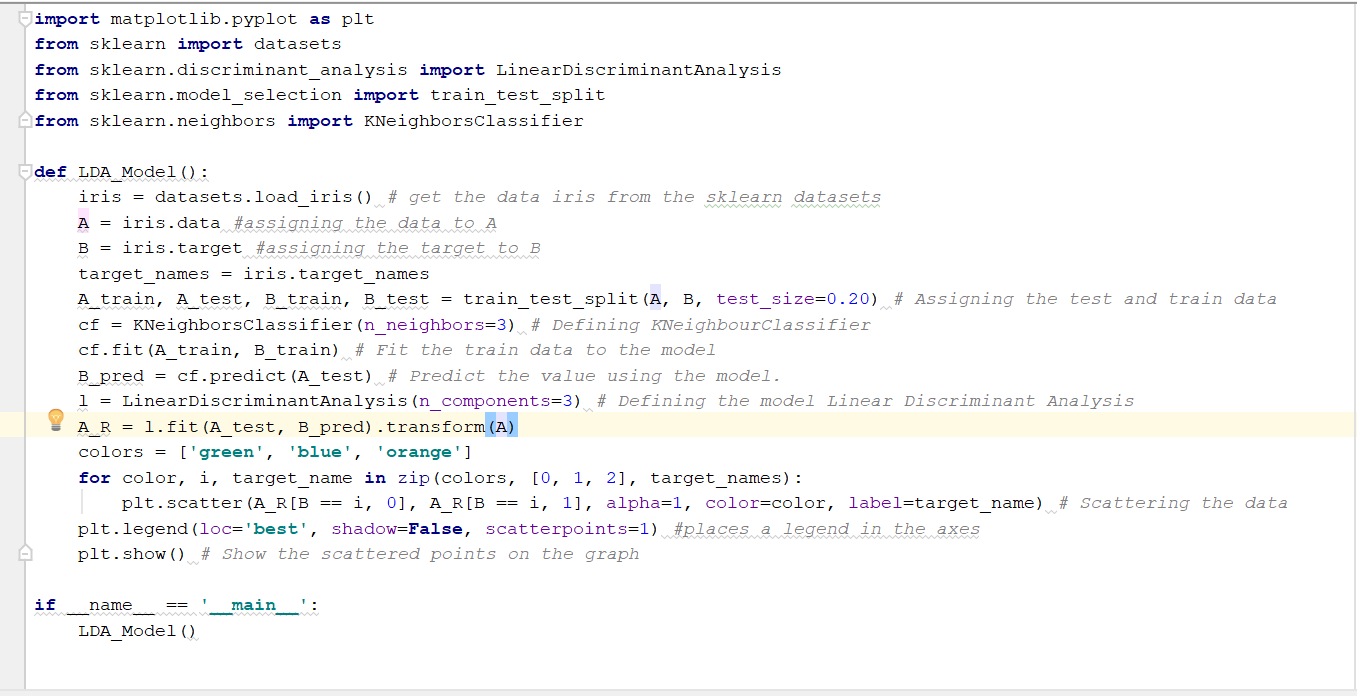
**Approach:** Got the data from the sklearn datasets and form the train and test data and fit the data into the LDA model and plot the data points using the model and form a prediction model.

**Parameters:** Data is taken form the sklearn datasets and perform apply the LDA algorithm.

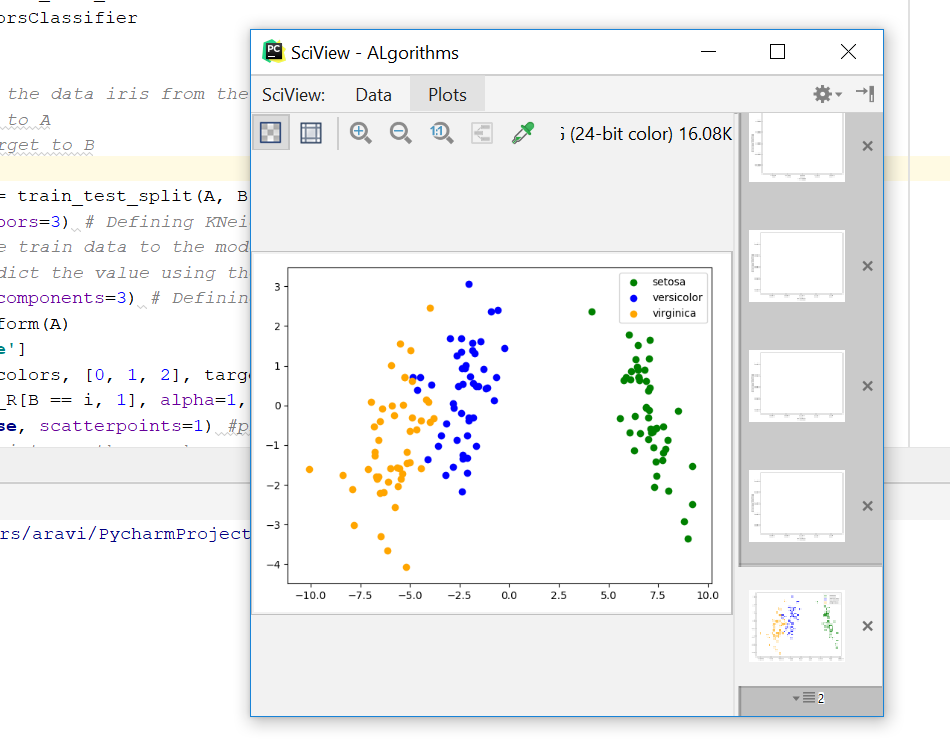
**Conclusion:** Applied the LDA algorithm on the Iris data and plotted the data points using the matplotlib and formed the classes of data with maximum distance so that we can predict using the data points and the class they belong to.

**Screenshots:**

**Code:**



**Output:**



**B:** Linear regression and Linear Discriminant Analysis are both statistical methods used widely for analyzing categories and classes of data. LDA makes more assumptions about the underlying data, so LR is more flexible and robust and easy to implement.

1. **Implementing Support Vector Machine Classification Algorithm:**

**Introduction:** SVC is a classifier that is used to create a boundary between classes of data for the better and correct visualization of data.

**Objectives:** Get the data from by any source and split the data into 20% test and 80% training data and apply SVC with Linear kernel and SVC with RBF kernel and find the accuracies of the predicted values using both the methods.

**Approach:** Got the data from the sklearn datasets and assigned the categories. Assigned the data to train and test using the train test split so that the values are assigned to the train and test. Created two models SVC Linear Kernel and SVC RBF Kernel and fit the data into the models and calculated the predicted and found the accuracy score for both values which are calculated by two separate models.

**Parameters:** Data is taken form the sklearn datasets and perform apply the SVC algorithm.

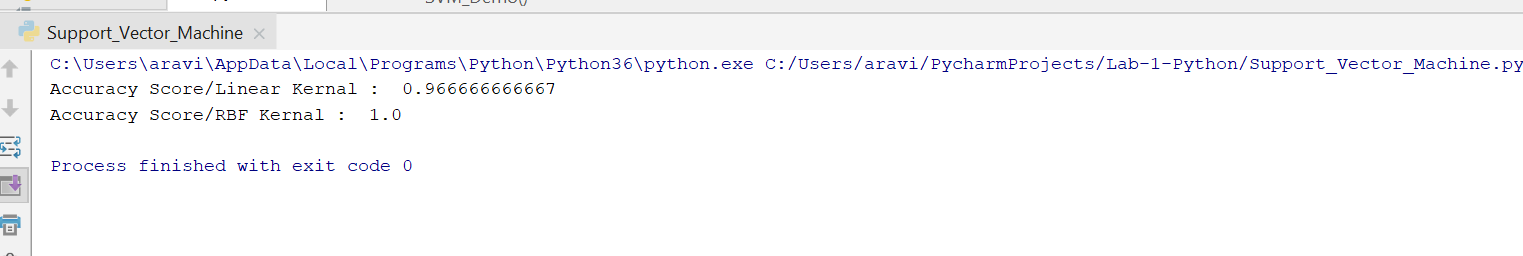
**Conclusion:** Applied the SVC algorithm on the data that is being collected from the sklearn data sets and calculated the predicted values using the two models SVC linear kernel and SVC RBF kernel and calculated the accuracies for the both predicted values. Accuracy score is very accurate when used RBF kernel compared to the Linear kernel.

**Screenshots:**

**Code:**



**Output:**



**Summary:** Accuracy can be increased by mapping the data into feature space and separating the data using a large margin hyperplane and by the aid of new kernels. RBF kernel is best suited for the model and the accuracy level is always maximum.

1. **Summarization of the data from an Input file:**

**Introduction:** Summarizing the data present in a file using Tokenization, Lemmatization and Bigrams in the NLP library.

**Objectives:** Read the data from an input file and summarizing the data using various steps and get the most frequent words that are present in the input file compared to the bigrams data which is calculated.

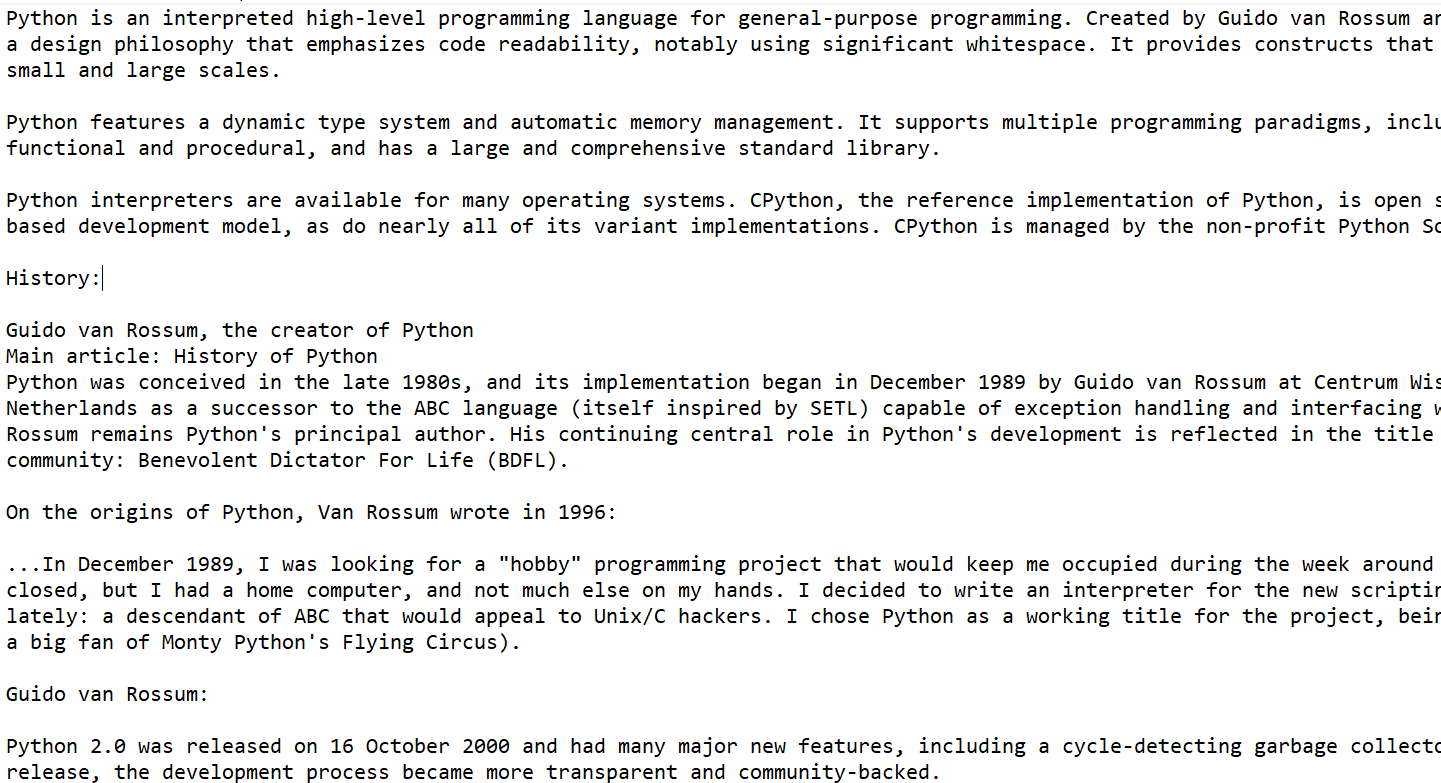
**Approach:** Read the data from an input file and apply Tokenization to split the data into words and apply bigrams so that combination of word pairs is formed and find the maximum frequency of the pair and comparing the words that are present in the input file and occur most frequently.

**Parameters:** Data is an input file which contains any information or anything.

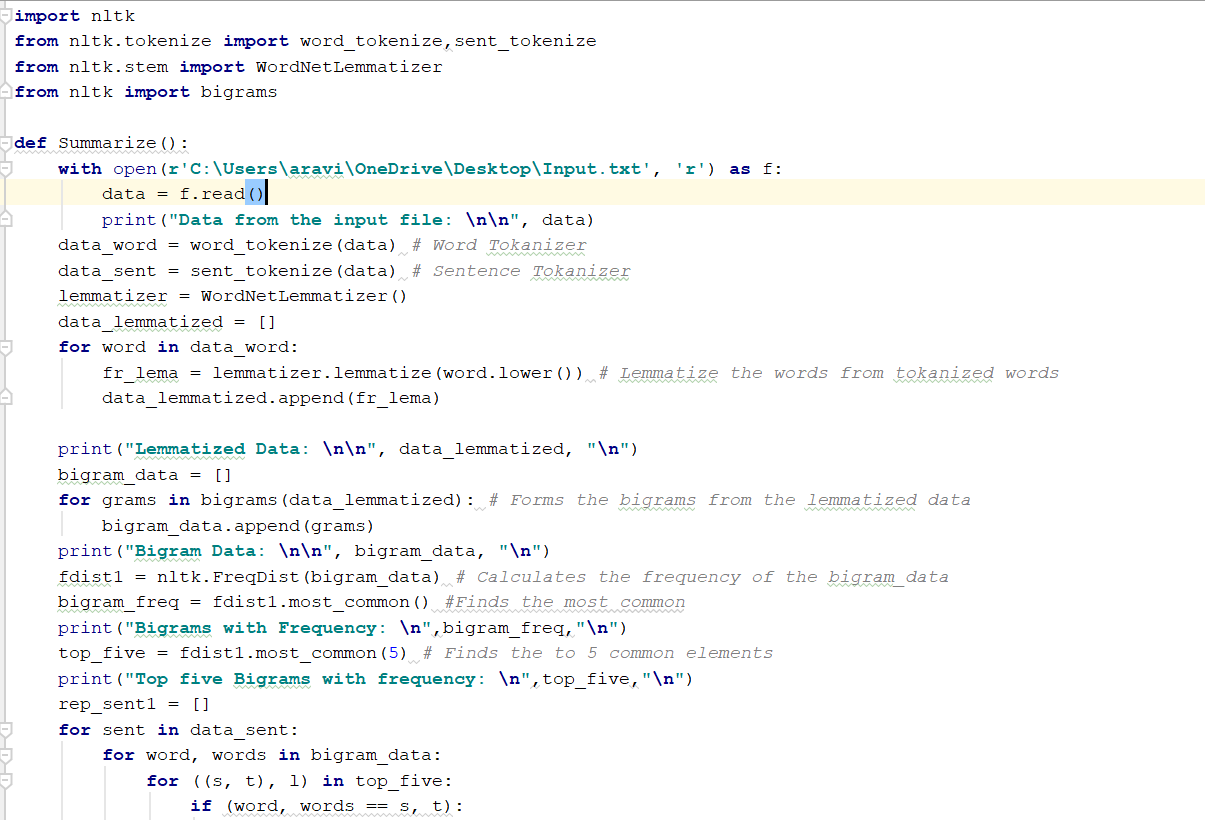
**Conclusion:** Applied the above approach and formed the summary of the data present in a file.

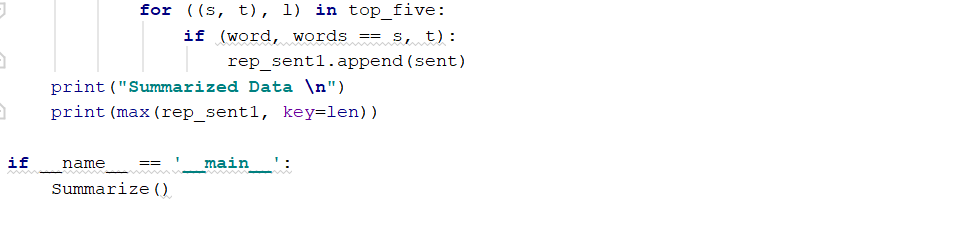
**Screenshots:**

**Input:**

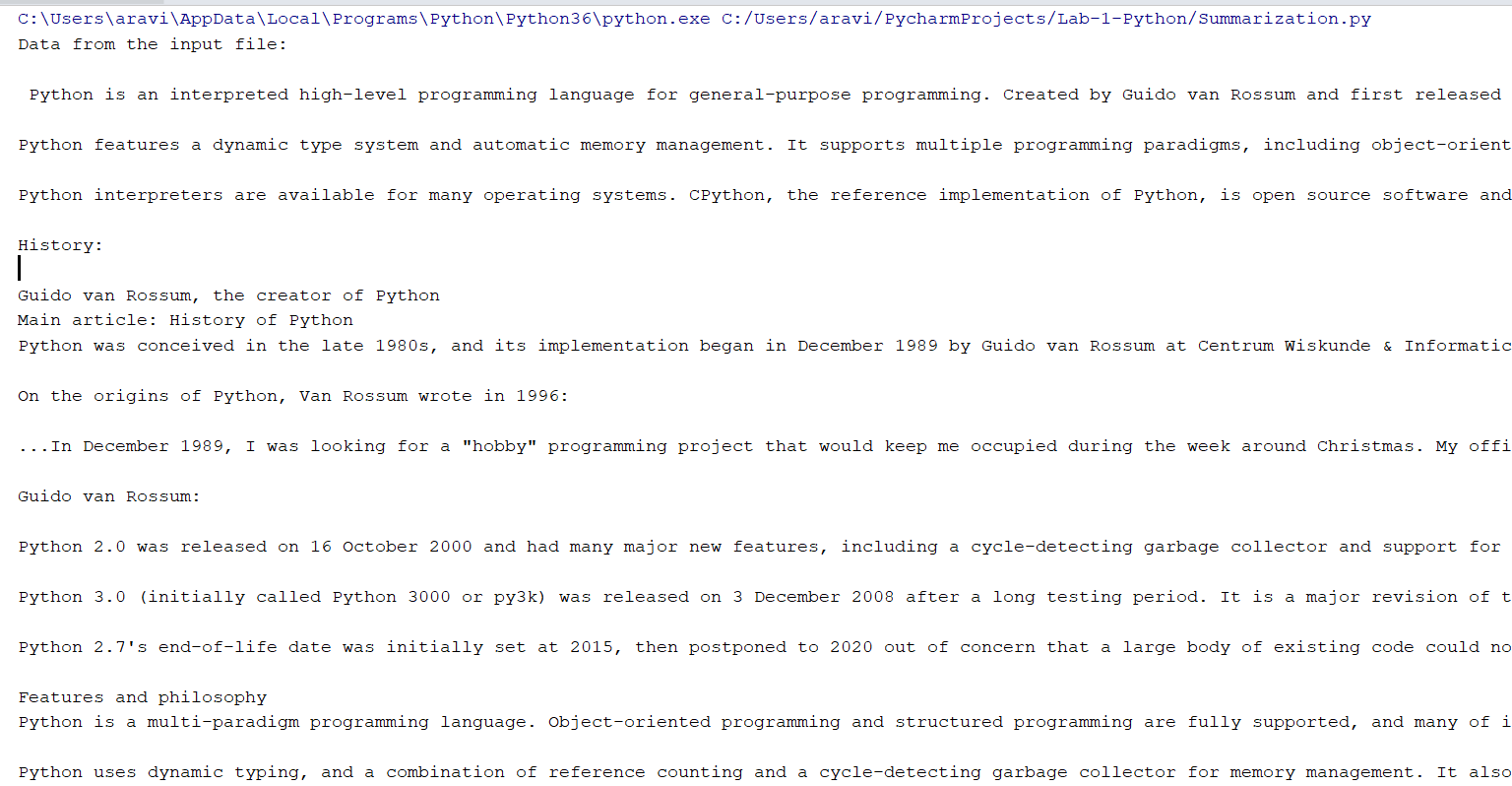


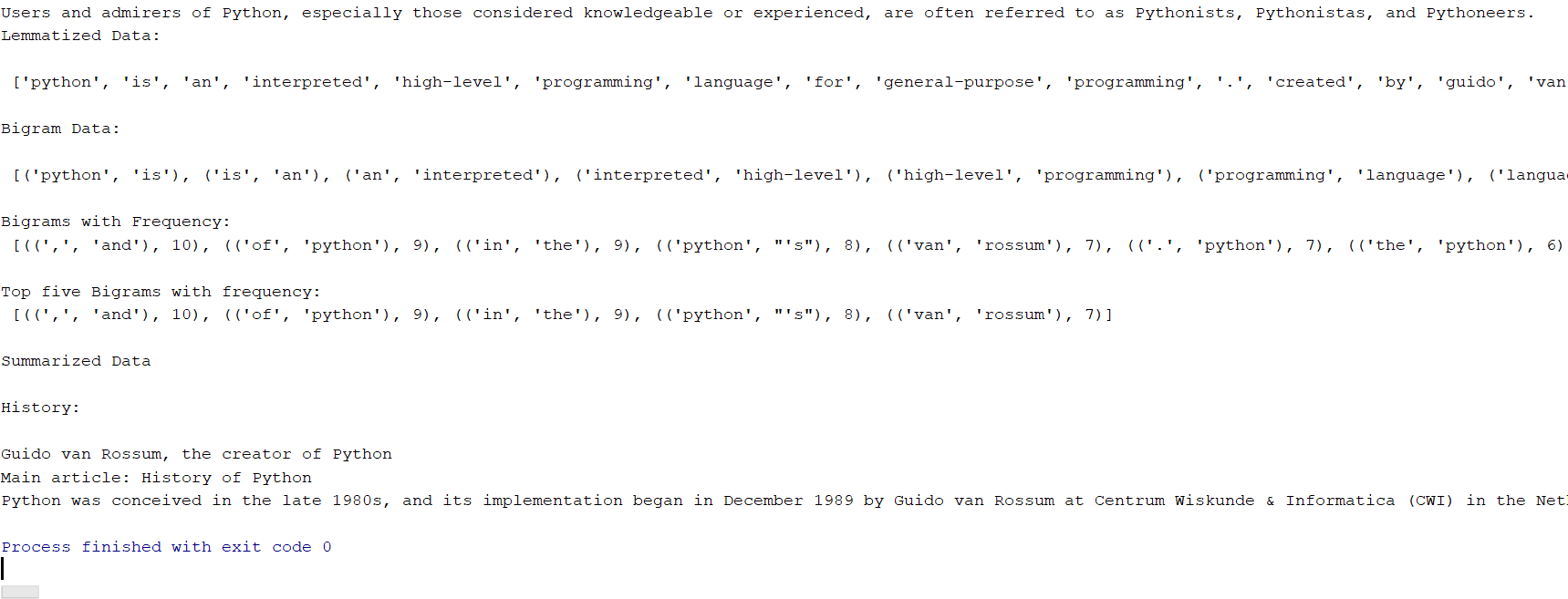
**Code:**





**Output:**





1. **K Nearest Neighbor Algorithm implementation:**

**Introduction:** When an unseen instance of data is seen K nearest Neighbor Algorithm will search through the training data set for the k-most similar instances and find the accurate class for the data instance.

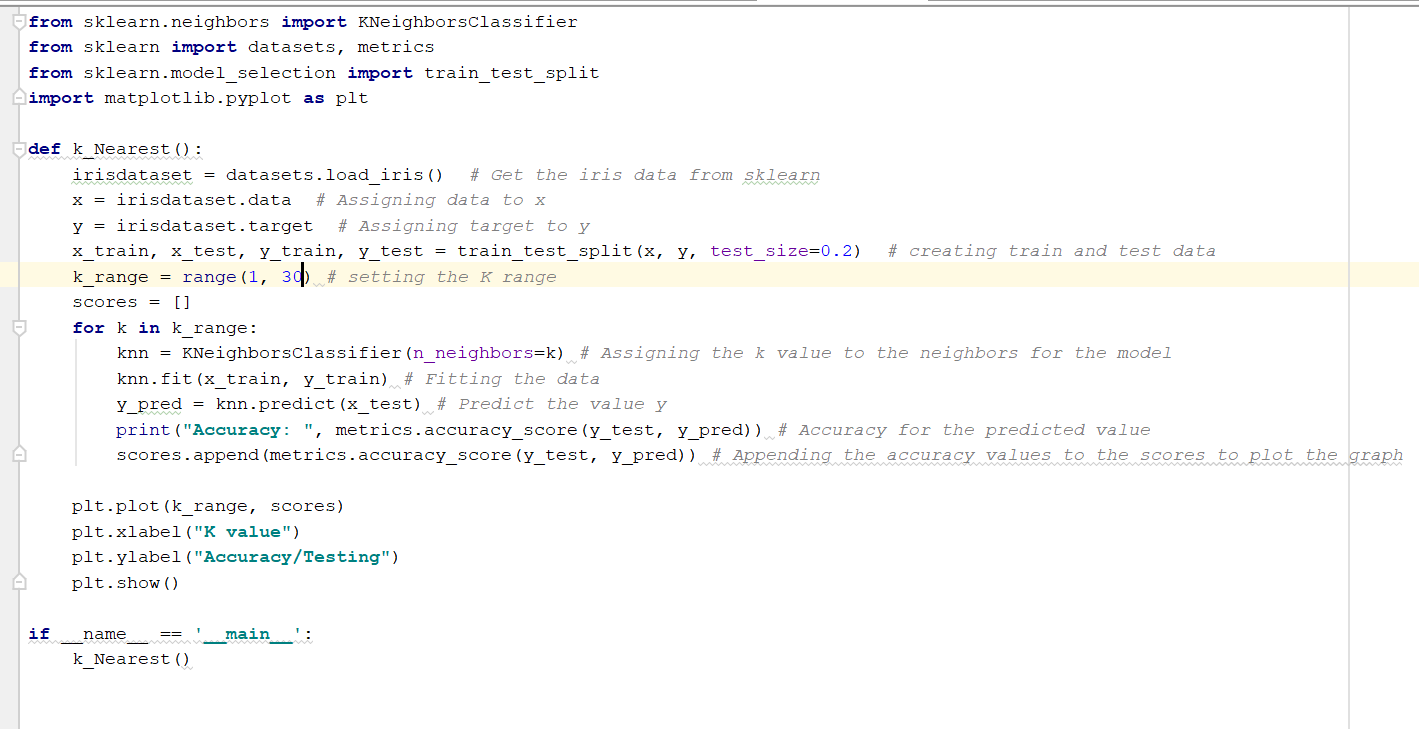
**Objectives:** Plot the graph and find the accuracy using K Nearest Algorithm and find the accuracy difference when K value is changed.

**Approach:** Got the data from the sklearn datasets and fit the data into KNeighborClassifier and find the predicted value by changing the K value and find the difference.

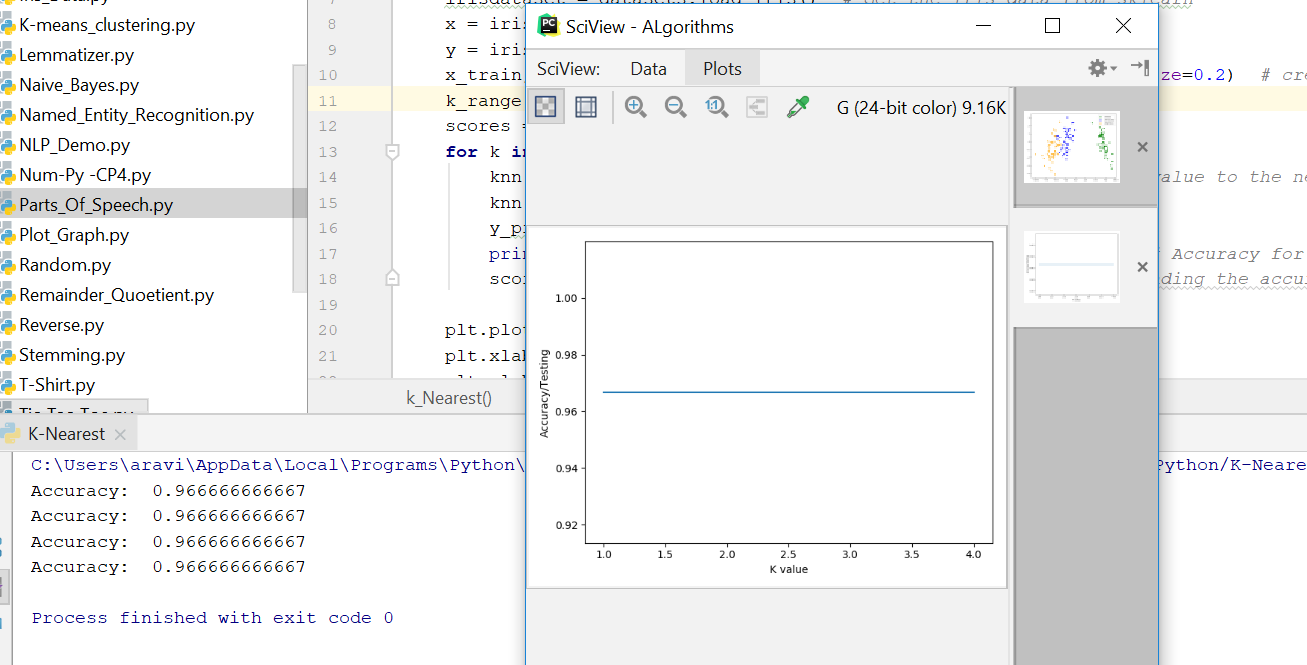
**Parameters:** Data is taken form the sklearn datasets.

**Conclusion:** Accuracy will change when the K value is changed and the plotted graph.

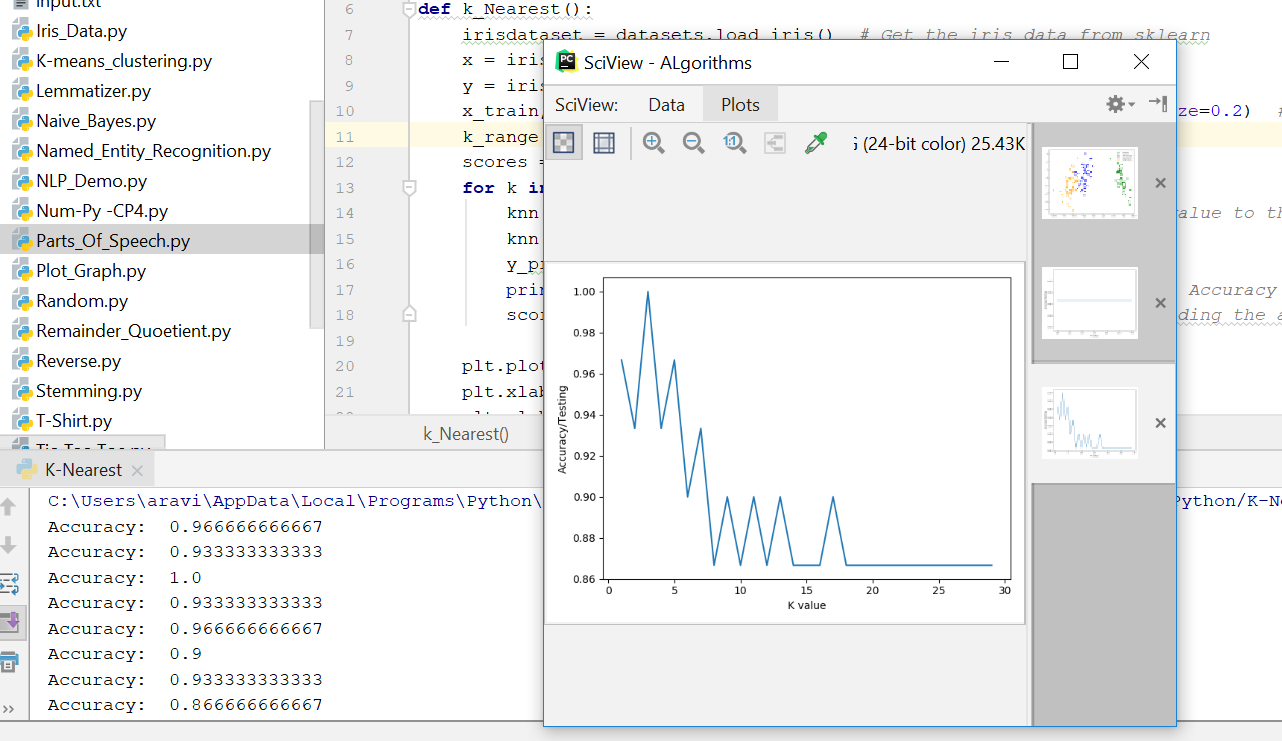
**Screenshots:**



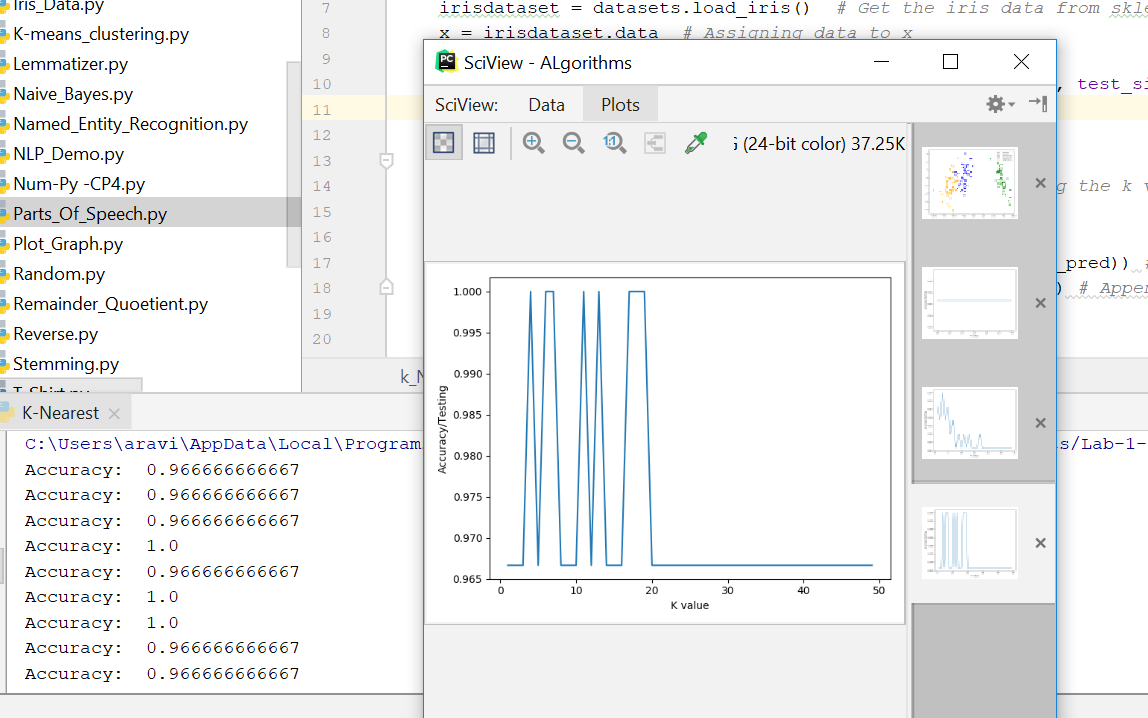
Output when K range is between 1 and 5:



**Output When K range is 1 and 30:**



**Output when K range is 1 and 50:**



**B.** Accuracy changes when the K value range is changed because of the following reasons.

**Smaller Values of K**: A small value for K provides the most flexible fit, which will have low bias but high variance.

**Higher Values of K**: Larger values of K will have smoother decision boundaries which means lower variance but increased bias.