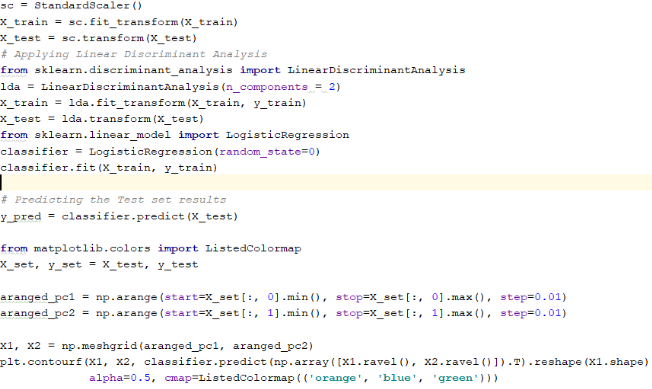
# LAB ASSIGNMENT 2

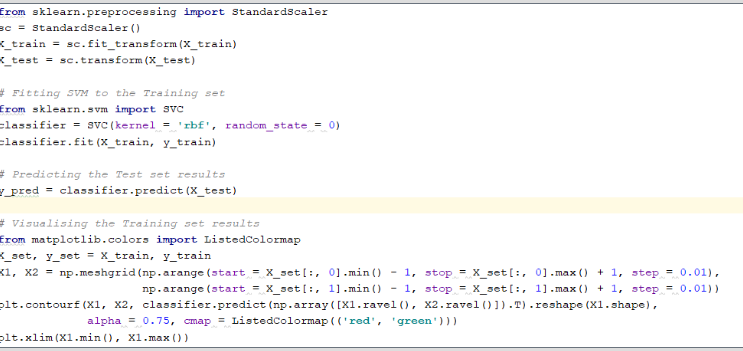
1.

All the required libraries are imported in the beginning of the program. Here I have applied linear discriminant analysis on Wine data. It depends on the remaining variables in the dataset. Different variables of values are provided, and output is checked. Here we have more than two classes so logistic regression cannot be applied.



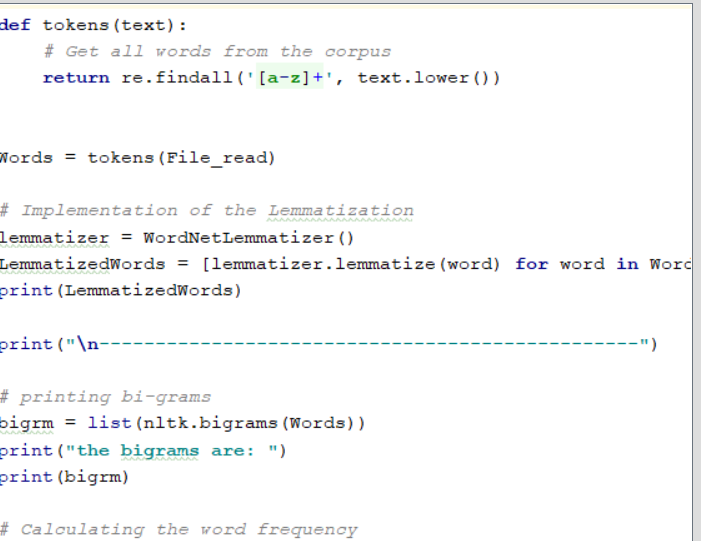
2.

Import all the required libraries for support vector machines. The dataset is loaded and split into two parts. The size of the test data is 20% and remaining is taken as training data. Then SVM model is applied with linear kernel and model is fit with training data. Then we will predict the future and accuracy is tested by using the remaining test data. In the similar way the model is applied with RBF kernel and accuracy is tested



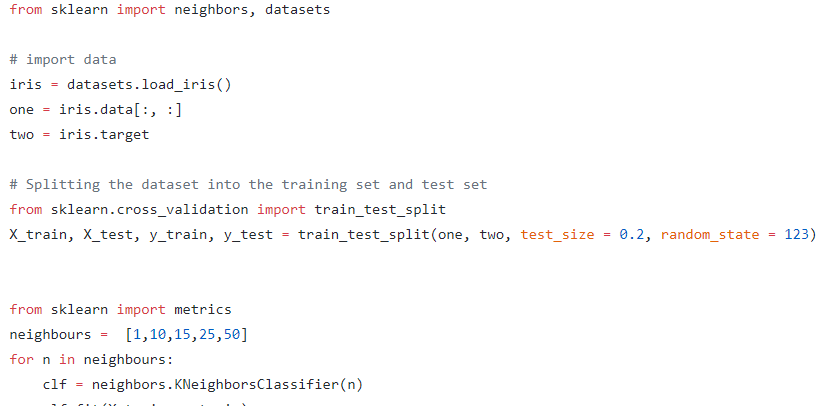
3.

All the required libraries are imported in the beginning of the program. Then open the file with read mode and print all the lines. Then all the text in the above file is divided into tokens. Then lemmatization can be done by using the parts of speech. Lemmatization removes the inflectional words and normalizes the words to the nearest form. Then bi-grams are produced by taking the n value as two and most frequent word is found in bi-grams and sentences with top five bi-grams are found by using all the predefined functions.



4.

K-nearest neighbors algorithm is a non-parametric method which is more used for regression and classification. It is much like the travelling sales person problem. The accuracy depends on the k value. A small value for K provides the most flexible fit. Larger values of K will have smoother decision boundaries. So to get the most flexible fit we should take lower values of k. Here in the above program, the accuracy is 0.9 when K=1 and accuracy is 0.866 when k=50. So the accuracy is high when the k values are low.



Video :

https://youtu.be/eLiec7IqD8M

GitHub :

https://github.com/BhargaviSiddipeta/Python/wiki/Lab-Assignment