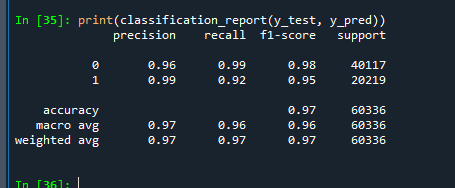
Lab 4 – Cole Daley

1. Execute the code in the file SpamFilterLogisticRegression.py and note the performance metrics. Change the training: test ratio to 80:20 and note the following performance metrics: confusion matrix, accuracy, recall, precision, and f1 score.

After executing the first original file and using logistic regression we obtain these results. Me and my group worked though the code and discovered a few errors that were ultimately fixed up and were able to run the program.



1. Change the code provided to perform classification using a decision tree classifier and note the performance metrics.

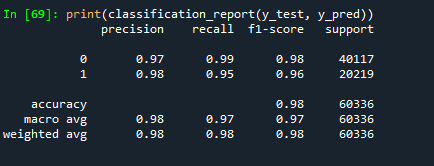
In the second question we are asked to use a decision tree. We used these trees for previous exercises and found out running the new import could take some time. Since we did it as a group, we took this wait time to focus on making sure everyone’s code was running together. The results for decision tree are shown below.

A screenshot of a computer program

Description automatically generated

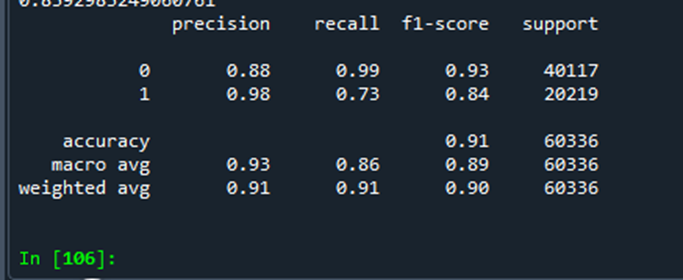
1. Change the code provided to perform classification using a support vector machine classifier and note the performance metrics.

At this point in the lab we understood most of the code was run the same and a few lines were changed. This allowed the process to run smoother and take less time. The results below are for the SVM classifier.



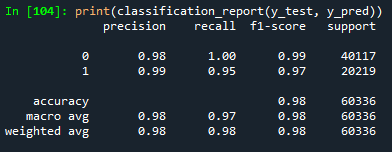
1. Change the code provided to perform classification using a random forest classifier and note the performance metrics.

Since we had this process down now, we were efficiently running through the code and finding our results. With the table shown below it is easy to compare and contrast the different results shown which I will do at the end of the document. The picture shown below is the results of random forest classifier.



1. Change the code provided to perform classification using an ANN classifier (multilayer perceptron - MLP) and note the performance metrics.

The final task we were asked to do was the MLP test. With this we understood the objective and the lines that needed to be changed. After making the changes we got the results shown below.



1. Finally, write a half-page reflection on the process and results from this classification activity.

Throughout this lab we have been asked to run different tests to see what produces the best results for catching spam emails. Some of the tests like the regression and decision tree tests we have run in the past but the remining three were new. We started with regression, and we changed the test to .2 like asked in the first question. We say the results for regression were quite strong but now we would like to compare those numbers with the remaining four tests. After a few changes to the source code, we were able to find the results for the decision tree classifier. These results were encouraging and were very close with regression metrics. What followed was the MLP, SVC, and the random forest tests which showed the MLP being the strongest overall giving us really had percentage results. The lowest test we saw was actually the random forest which gave us only a 91% overall accuracy.