For this assignment iris.csv was selected as the dataset. The dataset contains four features (sepal length, sepal width, petal length, and petal width) for three different species of iris (Setosa, Versicolor, and Viginica). The data was imported and processed as shown in Figure 1.

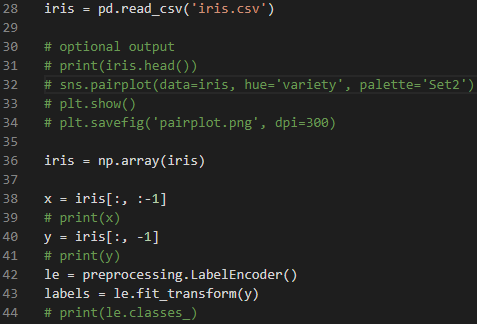


Figure - Importing the dataset.

The dataset was split 70% for training and 30% for testing as shown in Figure 2.



Figure - Splitting the dataset.

Three different types of classifiers are used: Support Vector Machine, K-Nearest Neighbors, and Logistic Regression.

The Support Vector Machine was used as shown in Figures 3 & 4.



Figure - Importing SVM from sklearn.

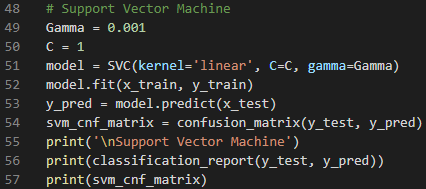


Figure - Code for SVM.

Use of the K-Nearest Neighbors classifier included validation of various hyperparameters as shown in Figures 5 & 6. Both Manhattan and Euclidian distances along with nearest neighbor values of 3, 5, and 7 were used in validating for best accuracy.



Figure - Importing KNN classifier from sklearn.

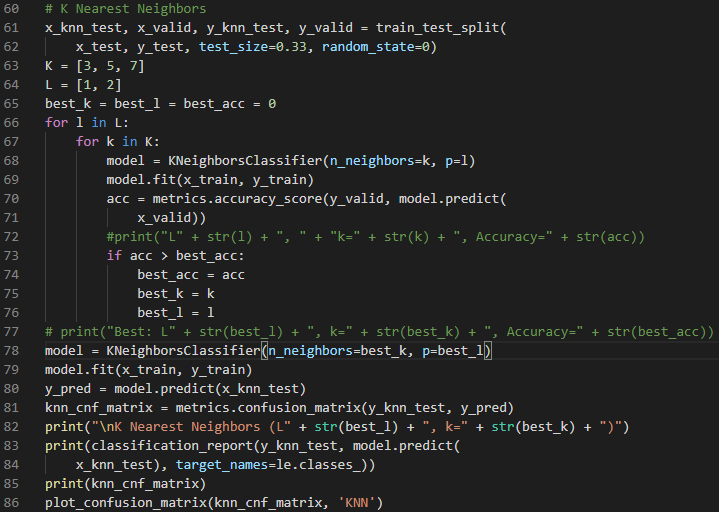


Figure - Code for KNN classifier.

Logistic regression was performed as shown in Figures 7 & 8.



Figure - Importing Logistic Regression from sklearn.

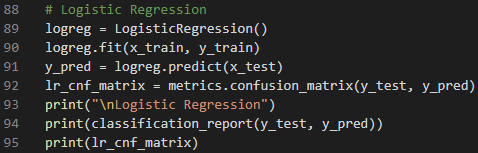


Figure - Code for Logistic Regression.

Using all four features from the dataset (sepal length, sepal width, petal length, and petal width), all classifiers perform very well.

As shown in Figure 9, the Support Vector Machine achieved accuracy of 98%. Only one iris was misclassified as shown in Figure 10.

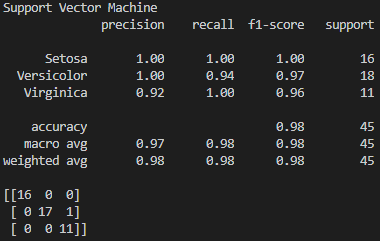


Figure - Metric and confusion matrix for SVM classifier with all dataset features.

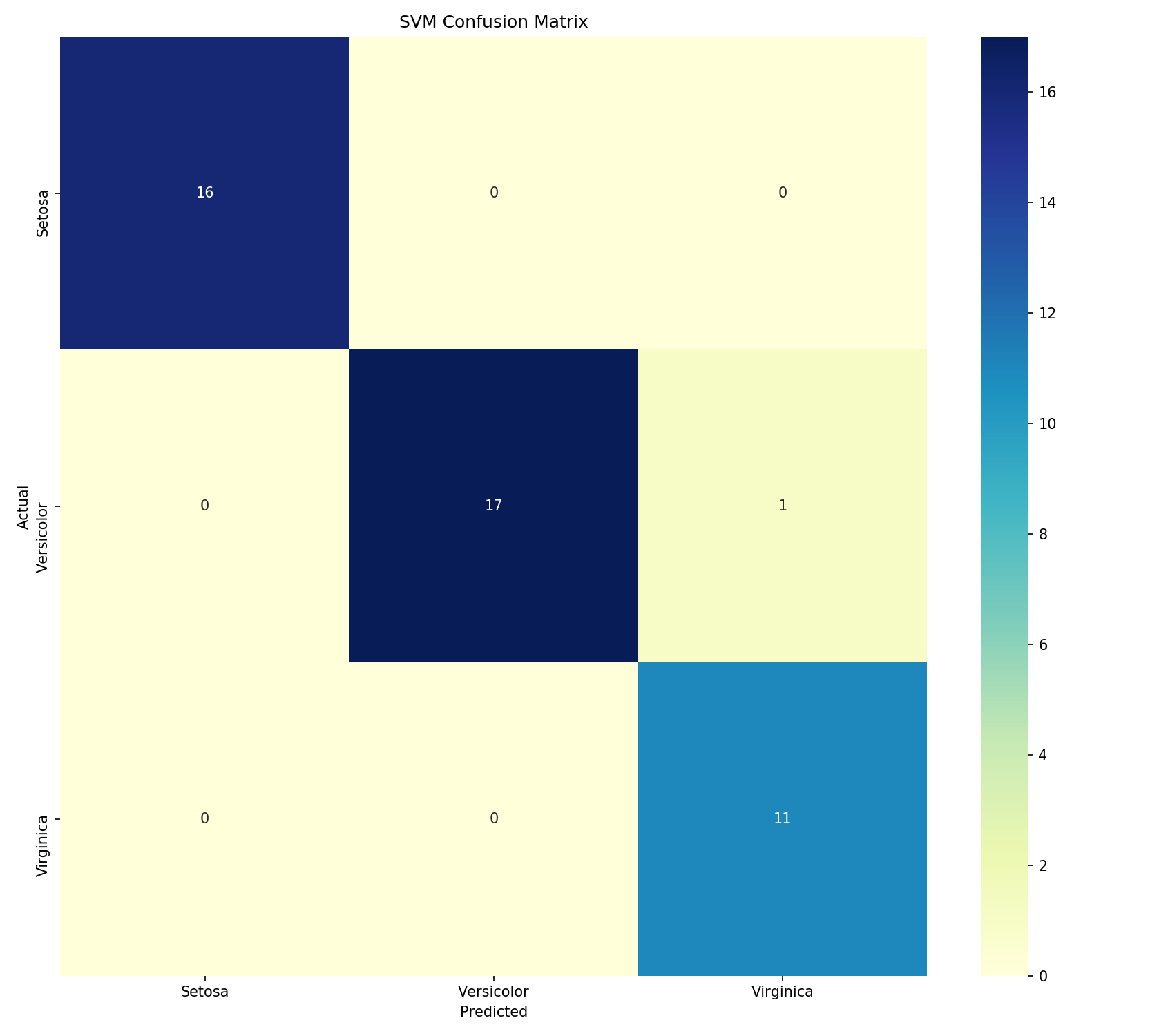


Figure - Confusion matrix for SVM using all dataset features.

As shown in Figures 11 & 12, the K-Nearest Neighbors achieved accuracy of 100%.

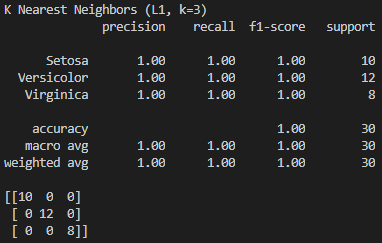


Figure - Metrics and confusion matrix for KNN classifier using all dataset features .

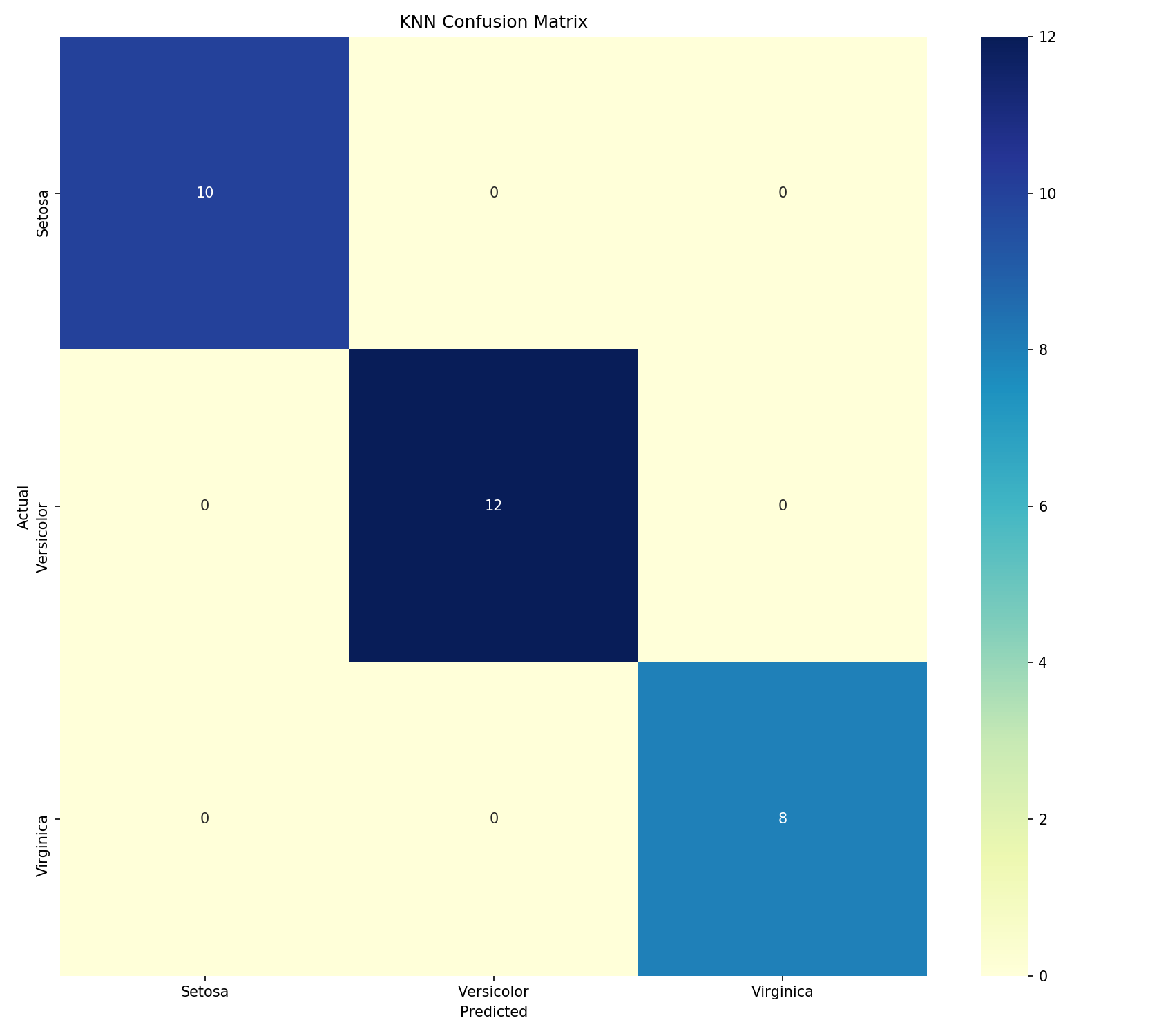


Figure - Confusion matrix for KNN classifier using all dataset features.

As shown in Figure 13, the Logistic Regression achieved accuracy of 98%. Only one iris was misclassified as shown in Figure 14.

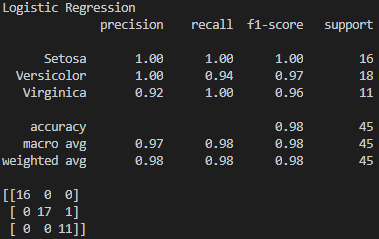


Figure - Metrics for Logistic Regression classifier using all dataset features.

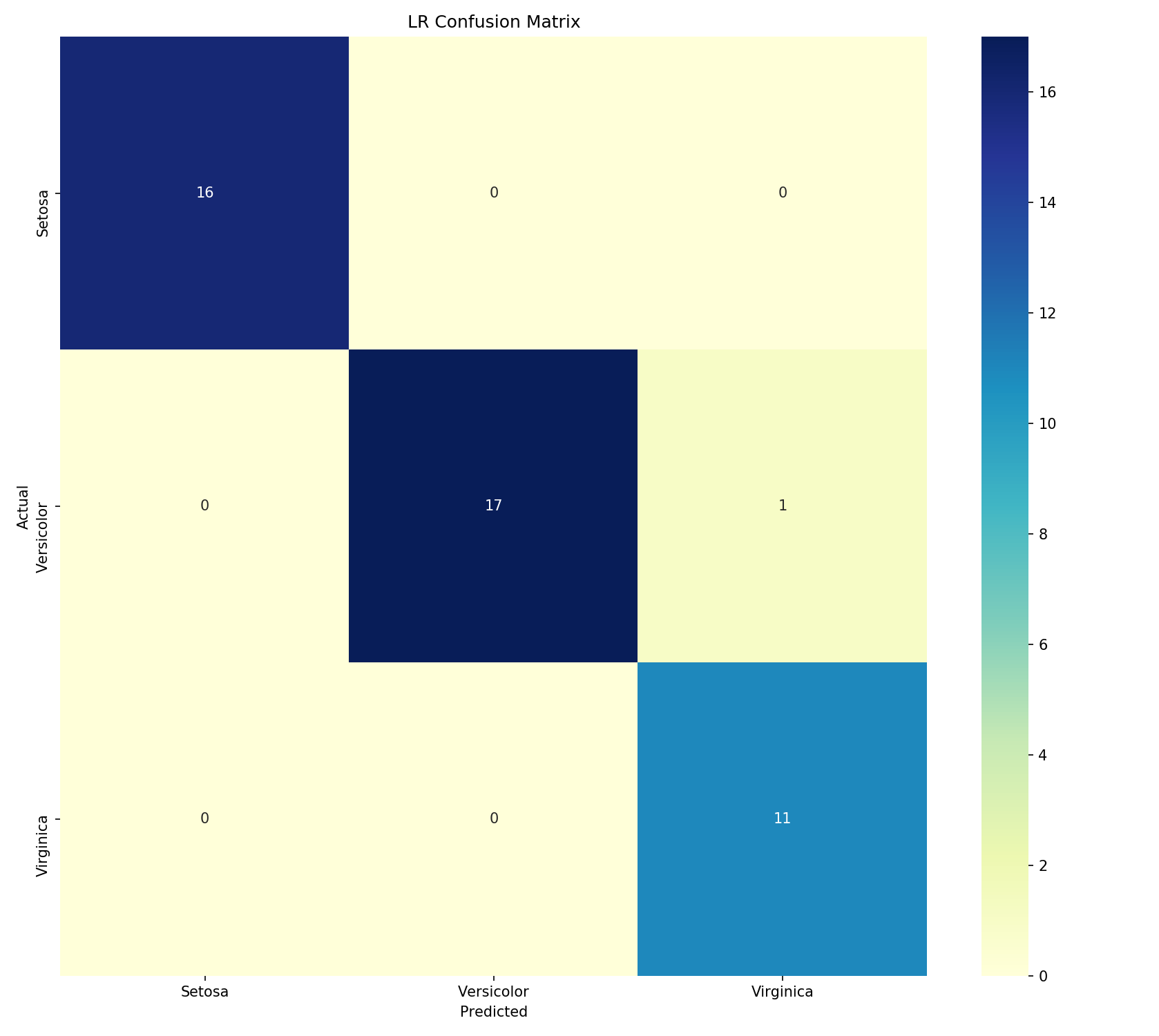


Figure - Confusion matrix for Logistic Regression classifier using all dataset features.

K-Nearest Neighbors performed the best using all features of the dataset. Additional tests were run using individual features of the dataset.

As shown in Figure 15, the data was modified to classify using only the sepal length feature.



Figure - Using only sepal length feature.

Metrics for each classifier using only the sepal length feature are shown in Figures 16 through 18.

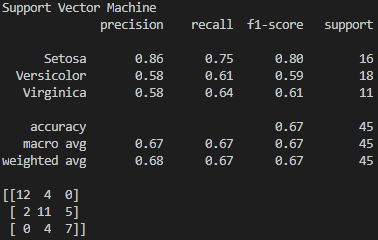


Figure - SVM metrics using only sepal length.

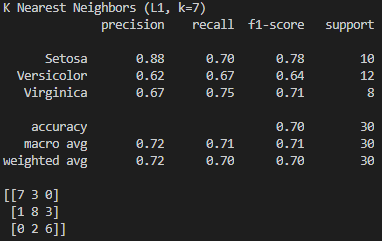


Figure -KNN metrics using only sepal length

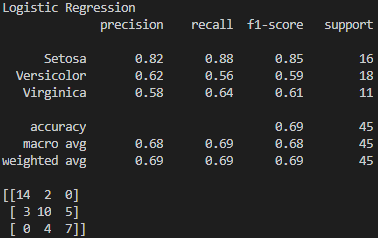


Figure - Logistic Regression metrics using only sepal length.

K-Nearest Neighbors performed best when using only sepal length as a classifier.

As shown in Figure 19, the data was modified to classify using only the sepal width feature.



Figure - Using only sepal width feature.

Metrics for each classifier using only the sepal width feature are shown in Figures 20 through 22.

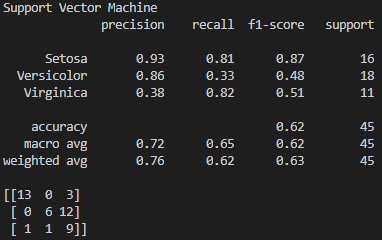


Figure - SVM metrics using only sepal width.

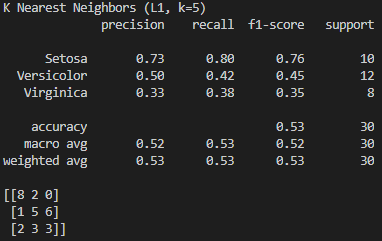


Figure - KNN metrics using only sepal width.

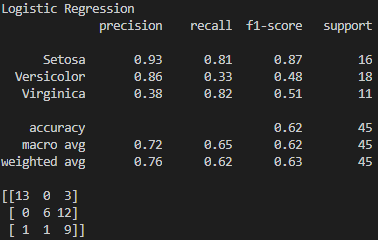


Figure - Logistic Regression metrics using only sepal width.

Support Vector Machine and Logistic Regression tied for best performance when using only sepal width as a feature.

As shown in Figure 23, the data was modified to classify using only the petal length feature.



Figure - Using only petal length feature.

Metrics for each classifier using only the petal length feature are shown in Figures 24 through 26.

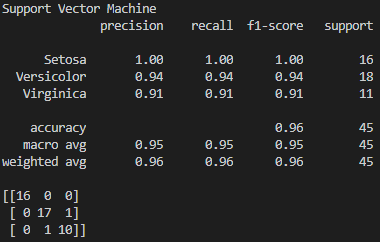


Figure - SVM metrics using only petal length.

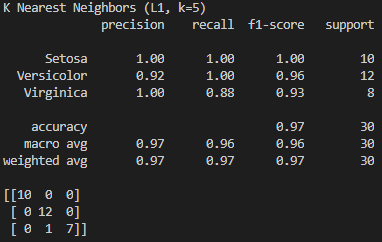


Figure - KNN metrics using only petal length.

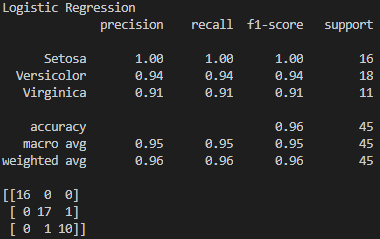


Figure -Logistic Regression metrics using only petal length.

K-Nearest Neighbors performed best when using only petal length as a classifier.

As shown in Figure 27, the data was modified to classify using only the petal width feature.



Figure - Using only petal width feature.

Metrics for each classifier using only the petal width feature are shown in Figures 28 through 30.

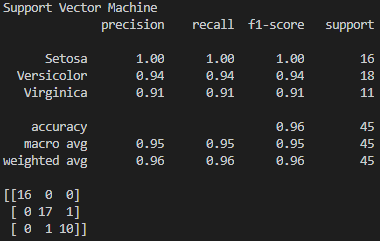


Figure - SVM metrics using only petal width.

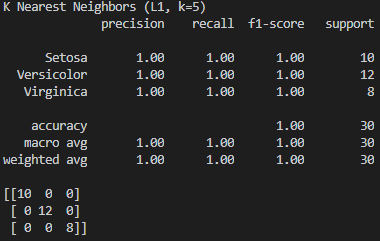


Figure - KNN metrics using only petal width.

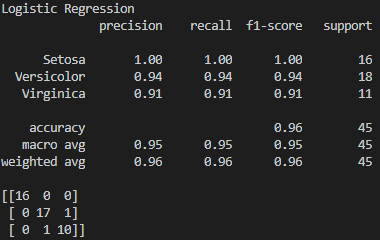


Figure - Logistic Regression metrics using only petal width.

K-Nearest Neighbors performed best when using only petal width as a classifier.

K-Nearest Neighbors performed better than SVM for most of the tests run. Different hyperparameters were tested for K-Nearest Neighbors, but not for the Support Vector Machine. It is possible that testing different hyperparameters for the Support Vector Machine may have improved performance. It is also possible that using static hyperparameters for the K-Nearest Neighbors classifier could have resulted in lower classification accuracy.

While using single features of the dataset did impact classification performance for each classifier, validation used to select hyperparameters for K-Nearest Neighbors seemed to have the greatest impact on performance. Overall, the small dataset created high performance for each classifier and tuning hyperparameters for K-Nearest Neighbors during each test run seems to have improved the classifier enough to outperform the others in most of the tests. K-Nearest neighbors has most likely performed as the best classifier because of the validation used to tune the hyperparameters during each test. This analysis seems apparent after examining the pair plot constructed from the dataset. As shown in Figure 31, the data for each feature of the iris dataset clusters together which seems to make it ideal for K-Nearest Neighbors and slightly less ideal for Support Vector Machine due to the slight overlap of some features.

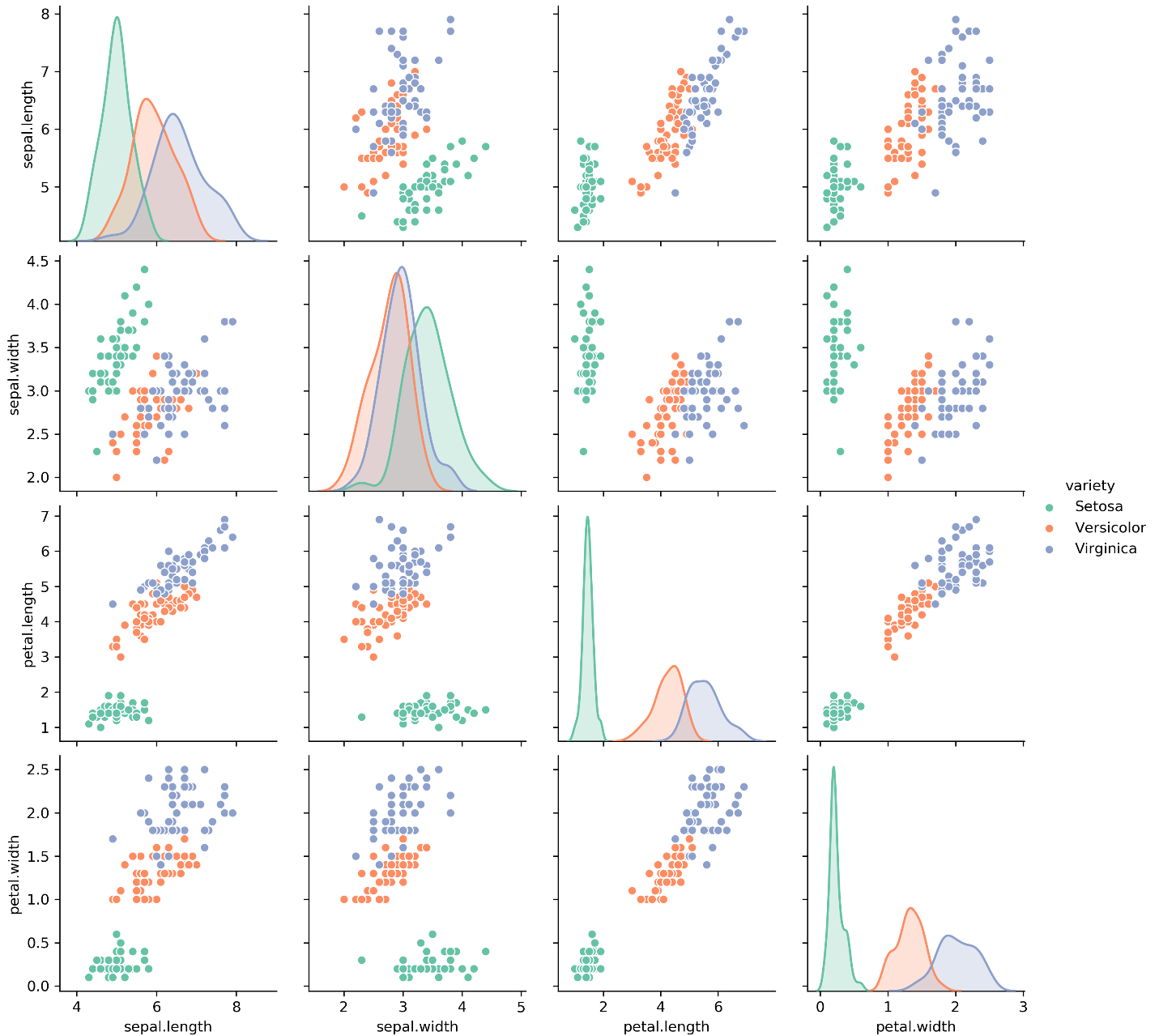


Figure – Pair plot of the iris dataset.