Gerom Pagaduan
Dr. Jiang
CS 3120 – Machine Learning
April 5, 2020

Midterm Report

What I Did

For the project, I chose to base my model off the `iris.csv` dataset. I split the data into three different sections. The 70% was used for training, 20% for testing, and 10% for validation. I used the following three models to classify the information given: Logistic Regression, K Nearest Neighbor, and the Support Vector Machine.

Their performances did not vary much from each other, so I ended up making a few changes to *what* data was being used. First, I changed the `random_factor` parameter for when I partitioned the data; however, this had minimal impact and did not change anything meaningful. Then, I changed the training and testing sizes, which affected the performance of all three. To experiment further, I started removing different feature-columns from the dataset and I had very different results, depending on which one I removed.

What I Noticed

I realized that the petal features were the most important features of all. You could read *only* the petal information—mainly the petal width—and get the same accuracy as you would when you use the whole dataset. Conversely, if you take away the petal information from the training data, the performance of all classifiers became much worse. There were a few wrong classifications in the dataset, but those did not impact the results much.

Through changing different variables, I noticed that for every single test, the results of the Logistic Regression and the SVM classifiers were nearly identical, only being off by a maximum of around 5%. The KNN classifier outshined the other two when only the information of sepals was provided. On the other hand, SVM performed the worst.

What Changed The results

The results changed when I adjusted the features used or when I adjusted the way the data was partitioned. When I increased the training size and decreased the testing size, the accuracy of all classifiers decreased slightly. When I decreased the training size and increased the testing size, the accuracy slightly increased.

The largest change happened when I started to omit features. When I omitted the petal length and width features, I noticed that it decreased the accuracy by about 25%. If I used any information of the petals, my program would yield at least a 96% accuracy. Removing the sepal features did not affect the overall accuracy very much. It lowered the recall and precision, but the f-1 score and accuracy remained the same. However, if I did not use the petal features, its accuracy would drop and the KNN classifier would outperform the other two each test.

Conclusion

I think the reason my model was very accurate was because the dataset was very small, and my performance was very high. Because of that, I am concerned that I could have overfitted my model to the specific dataset to where it is very good at handling that specific data. The result was that my accuracies were very similar. It was hard to judge which one did the best since each classifier had the exact same precision, recall, f1-score, and accuracy when all the data was used. I determined that they were all equally efficient; but when dealing with a small dataset, the KNN classifier outshined the other two when crucial features such as the petals were not used. I believe that if I had a much larger dataset, the SVM classifier would be the most useful because it can separate the data more efficiently.

Output

Using all data: Same accuracy of 96% for all classifiers.

Logistic Regression

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20
1	1.00	0.88	0.94	17
2	0.88	1.00	0.93	14
accuracy			0.96	51
macro avg	0.96	0.96	0.96	51
weighted avg	0.97	0.96	0.96	51

K Nearest Neighbors

	precision	recall	f1-score	support
Setosa	1.00	1.00	1.00	20
Versicolor	1.00	0.88	0.94	17
Virginica	0.88	1.00	0.93	14
accuracy			0.96	51
macro avg	0.96	0.96	0.96	51
weighted avg	0.97	0.96	0.96	51

Support Vector Machine

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20
1	1.00	0.88	0.94	17
2	0.88	1.00	0.93	14
accuracy			0.96	51
macro avg	0.96	0.96	0.96	51
weighted avg	0.97	0.96	0.96	51

Using only petal width: Same accuracy of 96% for all classifiers.

Logistic Regression

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20
1	1.00	0.88	0.94	17
2	0.88	1.00	0.93	14
accuracy			0.96	51
macro avg	0.96	0.96	0.96	51
weighted avg	0.97	0.96	0.96	51

K Nearest Neighbors

	precision	recall	f1-score	support
Setosa	1.00	1.00	1.00	20
Versicolor	1.00	0.88	0.94	17
Virginica	0.88	1.00	0.93	14
accuracy			0.96	51
macro avg	0.96	0.96	0.96	51
weighted avg	0.97	0.96	0.96	51

Support Vector Machine

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20
1	1.00	0.88	0.94	17
2	0.88	1.00	0.93	14
accuracy			0.96	51
macro avg	0.96	0.96	0.96	51
weighted avg	0.97	0.96	0.96	51

Using only petal length: Same accuracy of 94% for all classifiers.

Logistic Regression

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20
1	0.94	0.88	0.91	17
2	0.87	0.93	0.90	14
accuracy			0.94	51
macro avg	0.93	0.94	0.94	51
weighted avg	0.94	0.94	0.94	51

K Nearest Neighbors

	precision	recall	f1-score	support
Setosa	1.00	1.00	1.00	20
Versicolor	0.94	0.88	0.91	17
Virginica	0.87	0.93	0.90	14
accuracy			0.94	51
macro avg	0.93	0.94	0.94	51
weighted avg	0.94	0.94	0.94	51

Support Vector Machine

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20
1	0.94	0.88	0.91	17
2	0.87	0.93	0.90	14
accuracy			0.94	51
macro avg	0.93	0.94	0.94	51
weighted avg	0.94	0.94	0.94	51

Using only sepal width: KNN Classifier performed the best at 61% accuracy.

Logistic Regression

	precision	recall	f1-score	support
0	0.81	0.65	0.72	20
1	0.80	0.47	0.59	17
2	0.36	0.64	0.46	14
accuracy			0.59	51
macro avg	0.66	0.59	0.59	51
weighted avg	0.68	0.59	0.61	51

K Nearest Neighbors

	precision	recall	f1-score	support
Setosa	0.69	0.90	0.78	20
Versicolor	0.86	0.35	0.50	17
Virginica	0.39	0.50	0.44	14
accuracy			0.61	51
macro avg	0.65	0.58	0.57	51
weighted avg	0.66	0.61	0.59	51

Support Vector Machine

	precision	recall	f1-score	support
0	0.81	0.65	0.72	20
1	0.80	0.47	0.59	17
2	0.36	0.64	0.46	14
accuracy			0.59	51
macro avg	0.66	0.59	0.59	51
weighted avg	0.68	0.59	0.61	51

Using only sepal length: KNN classifier performed the best at 80% accuracy.

Logistic Regression

	precision	recall	f1-score	support
0	0.95	0.90	0.92	20
1	0.69	0.53	0.60	17
2	0.63	0.86	0.73	14
accuracy			0.76	51
macro avg	0.76	0.76	0.75	51
weighted avg	0.78	0.76	0.76	51

K Nearest Neighbors

	precision	recall	f1-score	support
Setosa	0.95	0.90	0.92	20
Versicolor	0.90	0.53	0.67	17
Virginica	0.64	1.00	0.78	14
accuracy			0.80	51
macro avg	0.83	0.81	0.79	51
weighted avg	0.85	0.80	0.80	51

Support Vector Machine

	precision	recall	f1-score	support
0	0.94	0.80	0.86	20
1	0.60	0.53	0.56	17
2	0.63	0.86	0.73	14
accuracy			0.73	51
macro avg	0.72	0.73	0.72	51
weighted avg	0.74	0.73	0.73	51

Using only petal data: All results were identical at 96% accuracy.

Logistic Regression

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20
1	1.00	0.88	0.94	17
2	0.88	1.00	0.93	14
accuracy			0.96	51
macro avg	0.96	0.96	0.96	51
weighted avg	0.97	0.96	0.96	51

K Nearest Neighbors

	precision	recall	f1-score	support
Setosa	1.00	1.00	1.00	20
Versicolor	1.00	0.88	0.94	17
Virginica	0.88	1.00	0.93	14
accuracy			0.96	51
macro avg	0.96	0.96	0.96	51
weighted avg	0.97	0.96	0.96	51

Support Vector Machine

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20
1	1.00	0.88	0.94	17
2	0.88	1.00	0.93	14
accuracy			0.96	51
macro avg	0.96	0.96	0.96	51
weighted avg	0.97	0.96	0.96	51

Using only sepal data: SVM classifier performed the worst at 82% accuracy.

Logistic Regression

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20
1	0.86	0.71	0.77	17
2	0.71	0.86	0.77	14
accuracy			0.86	51
macro avg	0.85	0.85	0.85	51
weighted avg	0.87	0.86	0.86	51

K Nearest Neighbors

	precision	recall	f1-score	support
Setosa	1.00	1.00	1.00	20
Versicolor	1.00	0.59	0.74	17
Virginica	0.67	1.00	0.80	14
accuracy			0.86	51
macro avg	0.89	0.86	0.85	51
weighted avg	0.91	0.86	0.86	51

Support Vector Machine

	precision	recall	f1-score	support
0	1.00	1.00	1.00	20
1	0.83	0.59	0.69	17
2	0.63	0.86	0.73	14
accuracy			0.82	51
macro avg	0.82	0.82	0.81	51
weighted avg	0.84	0.82	0.82	51