Software Project Milestone Two

Ryan Garry

Milestone two: kNN classification evaluation

The second milestone involves evaluating the kNN classifier for k=1 and k=3. For the remainder of the project, only the third and fourth features (petal length and petal width) will be used. For milestone two, all 150 datapoints will be used as training data. The test vectors to be used are given in Table B.

Table B. Iris dataset test vectors (petal length and petal width) for milestone two.

Test case	Petal length	Petal width	True Class	Pred Class k=1	Pred Class k=3
1	2.0	0.8	Setosa		
2	4.0	0.8	Versicolor		
3	6.5	2.5	Virginica		
4	4.5	1.7	Virginica		
5	4.8	1.8	Virginica		
6	5.0	1.8	Versicolor		
7	5.0	1.5	Virginica		

Provide the following.

1. Complete Table B with the kNN classifier results for k=1 and k=3.
2. For both k=1 and k=3, provide confusion matrices.
3. For both k=1 and k=3, fill in Table C, providing the overall probability of classification error as well as conditional classification error probabilities, conditioned on the true class.
4. Answer the following question. **For test case 7, why did the k=3 classifier choose versicolor?** The scatter plot in Fig. B suggests that for k=3, it should choose virginica. Hint: take a closer look at the dataset.

Note: test case 5 can be used to verify the tie-breaking rule. For k = 1, there is a tie between versicolor and virginica. According to the "smallest" rule, versicolor (2nd class) should be selected instead of virginica (3rd class).

Note: test case 6 reveals a roundoff error problem with the kNN classifier in MATLAB®® (Python may not have this). For k = 1, there should be a tie, in which case Versicolor should be the detected (predicted) class. However, due to roundoff, you may get Virginica as the detected class. To ensure that roundoff error does not bypass the tie-breaking rule, **you should feed the classifier integer features** (e.g. feature_int = round(10*feature)).

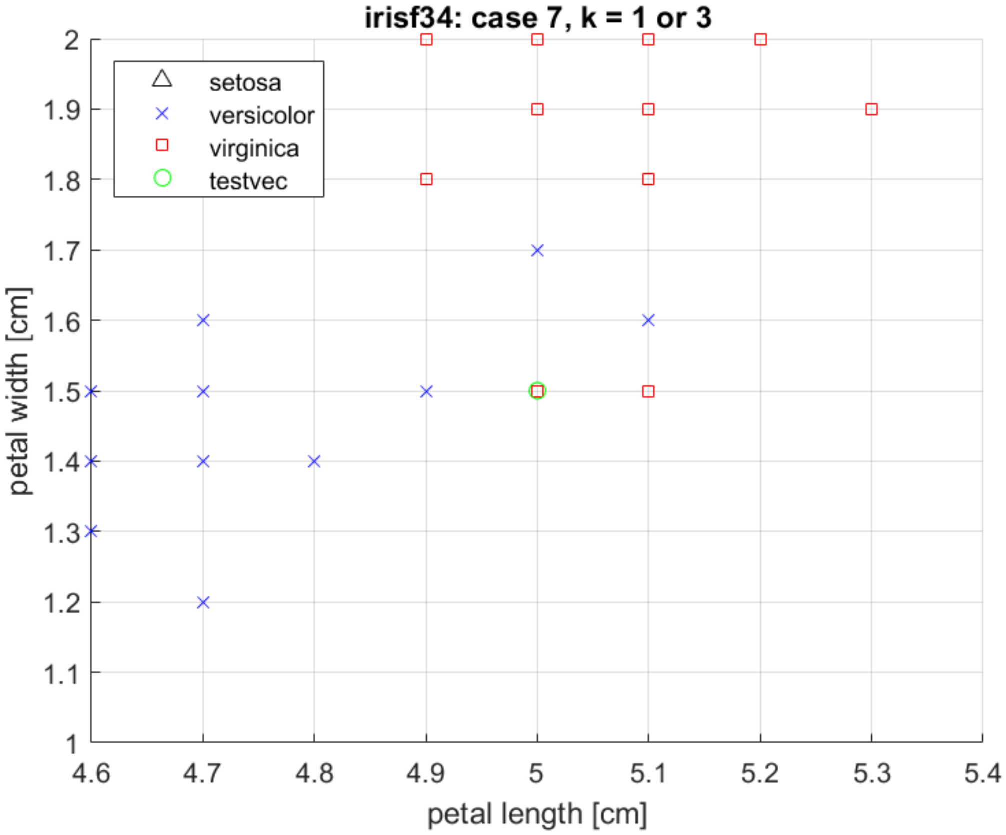

Figure B. Scatter plot of fisheriris dataset (features petal length and petal width only) and test vector 7.

Table C. Overall & conditional classification error probabilities for kNN classification trained with the irisf34 dataset.

kNN k value	Overall Pe	Pe | setosa	Pe | versicolor	Pe | virginica
1				
3				

Solution

1. Results for the kNN classifier for k=1 and k=3 are given in Table 1.
2. Confusion matrices for k=1 and k=3 are given in Figs. 1 and 2, respectively.
3. Overall and conditional probability of classification error values are given in Table 2.

4. **For test case 7, why did the k=3 classifier choose versicolor?** The scatter plot in Fig. B suggests that for k=3, it should choose virginica.
    a. **Answer**: The scatter plot suggests it should be virginica for k=3; however, the plot does not show that there are two Versicolor vectors at (4.9,1.5). This means that for the volume that encloses at least 3 vectors would require a 0.1 Manhattan block radius which encloses 4 vectors: 2 Versicolor and 2 Virginica. A tie has occurred and from earlier in the document, "according to the "smallest" rule, versicolor (2nd class) should be selected instead of virginica (3rd class)." Therefore, **Versicolor** is chosen for k=3.

Table 1. Iris dataset test vectors (petal length and petal width) for milestone two.

Test case	Petal length	Petal width	True Class	Pred Class k=1	Pred Class k=3
1	2.0	0.8	Setosa	Setosa	Setosa
2	4.0	0.8	Versicolor	Versicolor	Versicolor
3	6.5	2.5	Virginica	Virginica	Virginica
4	4.5	1.7	Virginica	Virginica	Versicolor
5	4.8	1.8	Virginica	Versicolor	Virginica
6	5.0	1.8	Versicolor	Virginica	Virginica
7	5.0	1.5	Virginica	Virginica	Versicolor

Table 2. Overall & conditional classification error probabilities for kNN classification trained with the irisf34 dataset.

kNN k value	Overall Pe	Pe | setosa	Pe | versicolor	Pe | virginica
1	2/7	0	1/2	1/4
3	3/7	0	1/2	2/4

* I think the kNN implementation in python may be subject to roundoff error due as I give it integer features, but it still guesses Virginica for case 6 with k=1. My guess is it introduces floating point math internally. Giving it integer features did not change the test cases for k=1 or k=3.

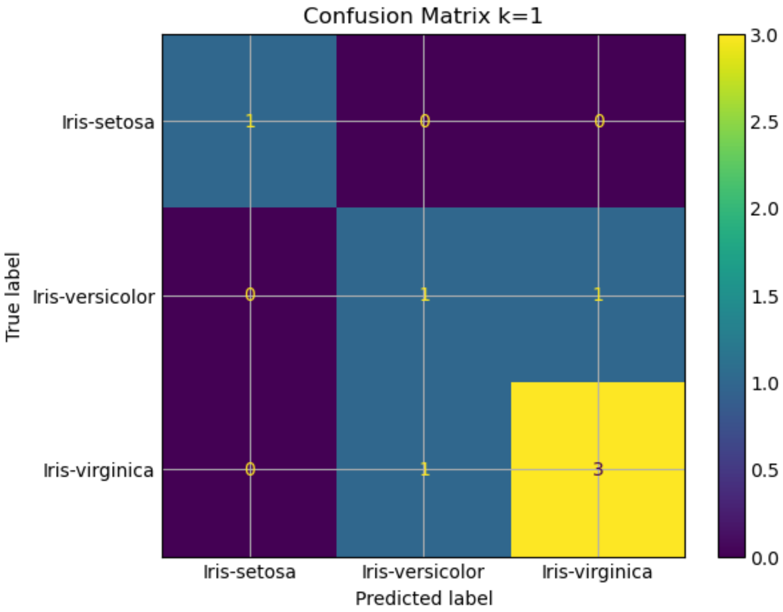

Figure 1. Confusion matrix for kNN classifier and k=1.

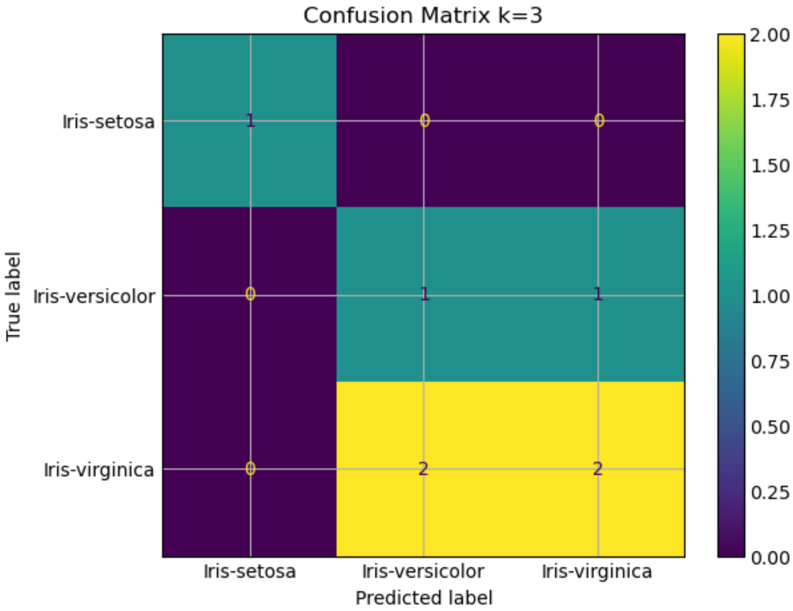

Figure 2. Confusion matrix for kNN classifier and k=3.