HW 2 Student

Andy Ackerman

10/17/2023

This homework is meant to illustrate the methods of classification algorithms as well as their potential pitfalls. In class, we demonstrated K-Nearest-Neighbors using the `iris` dataset. Today I will give you a different subset of this same data, and you will train a KNN classifier.

Above, I have given you a training-testing partition. Train the KNN with $K = 5$ on the training data and use this to classify the 50 test observations. Once you have classified the test observations, create a contingency table – like we did in class – to evaluate which observations your algorithm is misclassifying.

```r
set.seed(123)
#STUDENT INPUT

model_1 <- knn(iris_train, iris_test, cl = iris_target_category, k = 5)
tableresult <- table(model_1, iris_test_category)
tableresult
```

```
##               iris_test_category
## model_1        setosa versicolor virginica
##   setosa            5          0         0
##   versicolor        0         25         0
##   virginica         0         11         9
```

Discuss your results. If you have done this correctly, you should have a classification error rate that is roughly 20% higher than what we observed in class. Why is this the case? In particular run a summary of the `iris_test_category` as well as `iris_target_category` and discuss how this plays a role in your answer.

Below you can see a summary of the `iris_test_category` as well as `iris_target_category` and it's results. After looking at these results, the classification error rate is higher than what we observed in class for a couple reasons that are mainly related to the uneven distribution of categories in the training and testing sets. Particularly with versicolor, we see that the data was trained on 14 observations, but tested on 36 observations. Obviously, there is less training data than we would like for versicolor and this will likely lead to inaccuracy because we know versicolor and virginica often oftenlap and since it was only trained on 14 versicolor examples, it might have a difficult time differentiating observations between those two categories. For setosa and virginica, these categories have more training than testing points, which can lead to a better performance. The thing is, even though there are fewer observations in the testing set, making it easier *to classify these observations, with there being fewer observations in the testing set, the contribution to the overall accuracy is not as high as versicolor. To fix this we would want to try and adjust our training and testing data to hopefully have a more balanced distribution between the three different categories.*

```
summary(iris_test_category)
```

```
##     setosa versicolor  virginica
##          5         36          9
```

```
summary(iris_target_category)
```

```
##     setosa versicolor  virginica
##         45         14         41
```

Choice of K can also influence this classifier. Why would choosing $K = 6$ not be advisable for this data?

$K = 6$ would not be advisable for this data and would be an inappropriate choice of K. The reason for this is the number of K should be indivisble by the number of categories. In our example above we have three categories - setosa, versicolor, and virginica. Obviously, $6/3 = 2$ and the reason we want our number for K to be indivisble is because if it was divisible, we would be unable to break any ties within the data.

Build a github repository to store your homework assignments. Share the link in this file.

https://github.com/kburr3/STOR390-HW/tree/HW-2