HW 2 Student

Sam Reid

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)
pr <- knn(iris_train, iris_test, iris_target_category, k=5)
tab <- table(pr, iris_test_category)
accuracy <- function(x){
  sum(diag(x))/sum(rowSums(x))
}
tab
```

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

```r
accuracy(tab)
```

```
## [1] 0.78
```

```r
#STUDENT INPUT
```

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.

STUDENT INPUT

```
summary(iris_target_category)
```

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

```
summary(iris_test_category)
```

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

Our accuracy suffered because our sample was not chosen randomly, which led to it being unrepresentative of the test data. As seen above, the training data contained much more setosa and virginica, and much less versicolor than the test data, which led to it misclassifying versicolor more frequently. Specifically, due to the small data size if a versicolor iris in the testing set was near the 'outside' of it's group in terms of the predictors, it would have a higher likelihood of getting caught in another species' group due to those group's having higher densities. The solution to this problem is just to use randomized training and testing data like we did in class. #

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

STUDENT INPUT https://github.com/SamReid4321/SamRSTOR390