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.

1

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.

```
set.seed(123)
knn <- knn(train=iris_train, test=iris_test, cl=iris_target_category, k=5)

contingency_table <- table(knn, iris_test_category)

contingency_table
```

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

```
accuracy <- function(x){
  sum(diag(x)/(sum(rowSums(x)))) * 100
}

sprintf("Accuracy is %.2f%%", accuracy(contingency_table))
```

```
## [1] "Accuracy is 78.00%"
```

```
print("Summary of Test and Train Categories")
```

```
## [1] "Summary of Test and Train Categories"
```

```
summary(iris_target_category)
```

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

```
summary(iris_test_category)
```

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

2

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.

The accuracy of our KNN model is 78%, which is much lower as compared to the 96% accuracy we found in the lecture code. This discrepency can be explained by the `iris_test_Category` and `iris_train_category` tables. A summary of these two tables show us a substantial difference in the categorical distributions of the flowers. The training data had the least number of versicolor flowers at just 14 while the versicolor made up the majority of the test data with 36 entries. This shows that the model did not have enough training data with versicolor to make the correct prediction as the training data was largely populated with setosa and virginica categories, which likely led to the incorrect classification of versicolor being predicted as virginica.

3

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

Choosing `K = 6` would not be ideal for this data as the training data is predominantly composed of setosa and virginica category flowers. By choosing `K = 6`, we will likely increase the error rate as the versicolor flowers will be incorrectly classified as one of the other two categories due to the higher probability that the surrounding points will also be majorly of setosa and virginica types.

4

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

https://github.com/aarjavjain2002/stor390-f24 (https://github.com/aarjavjain2002/stor390-f24)