HW 2 Student

Neel Singh

9/27/2024

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.

```
set.seed(123)

# implement knn classifier
knn1 <- knn(iris_train, iris_test, cl = iris_target_category, k = 5)

# create contingency table
table1 <- table(knn1, iris_test_category)
table1
```

```
##             iris_test_category
## knn1         setosa versicolor virginica
##   setosa          5          0         0
##   versicolor      0         25         0
##   virginica       0         11         9
```

```
# calculate classification accuracy
class_accuracy <- sum(diag(table1) / (sum(rowSums(table1))))
class_accuracy
```

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

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.

```r
# summary of 'iris_test_category'
summary(iris_test_category)
```

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

```r
# summary of 'iris_target_category'
summary(iris_target_category)
```

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

There is a stark difference in the representation of different classes between the test and target subsets. The test subset has the majority class 'versicolor', while 'setosa' and 'virginica' have much lower representation than the majority class. The target subset has the majority class 'setosa' (with 'virginica' coming in at a close second). The target subset has a much lower representation of 'versicolor' compared to 'setosa' and 'virginica' Thus, the data that the KNN classifier was "trained" on happens to underrepresent the majority class label of the test subset.

Since KNN relies on the distance between the testing observations and target subset observations to classify testing observations, the mismatch in representation between the different classes in both subsets contributes to the higher classification error. Since there are fewer 'versicolor' observations in the target subset, the KNN classifier may not be able to discern a clear boundary between 'versicolor' and other the other two classes. This causes poor generalization to the test subset, leading to the observed decrease in classification accuracy.

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

We are attempting to discern between three class labels. If $K = 6$, then K is divisible by the number of class labels we have. There is a possibility that we may encounter a situation where the $K = 6$ nearest neighbors are two 'setosa', two 'versicolor', and two 'virginica'. This would result in a three-way tie (and no clear class label for the test observation). A more probable scenario is a two-way tie between any two class labels.

A choice of K that is not divisible by the number of class labels could reduce the chance of a tie. It would still be possible to tie if we had, for instance, a scenario where the $K = 5$ nearest neighbors are two 'setosa', two 'versicolor', and one 'virginica' (thus causing a two-way tie). But it is less likely for this to happen than if K is divisible by the number of class labels we have.

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

GitHub Repo (https://github.com/nisingh1/STOR-390-Homework-2.git)