hmwk3.1a

student

2023-09-05

Cross Validation

Summary

Number of Nearest Neighbors: best k: 7 neighbors yielded the highest accuracy.

Accuracy of best k: 85.9% of the test samples the model made correct prediction.

Model Accuracy: 82.23% of the time the model is correct.

Confidence Interval: 95% interval accuracy falls in.

No Information Rate: 0.5584, success rate with no model prediction.

It appears the classifier performs reasonably well given the context of this data set.

##Begin Code

Load the packages

```r
knitr::opts_chunk$set(echo = TRUE)

library(kknn)
library(kernlab)
library(caTools)
library(caret)
```

```
## Loading required package: ggplot2
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following object is masked from 'package:kernlab':
##
##     alpha
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:kknn':
##
##     contr.dummy
```

Load and View Data Structure

```
# load data
data <- read.csv("C:\\Users\\Public\\Documents\\gatech\\hw1\\credit_card_with_headers.csv")

# check data structure for data frame
str(data)
```

```
## 'data.frame':    654 obs. of  11 variables:
##  $ A1 : int  1 0 0 1 1 1 1 0 1 1 ...
##  $ A2 : num  30.8 58.7 24.5 27.8 20.2 ...
##  $ A3 : num  0 4.46 0.5 1.54 5.62 ...
##  $ A8 : num  1.25 3.04 1.5 3.75 1.71 ...
##  $ A9 : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ A10: int  0 0 1 0 1 1 1 1 1 1 ...
##  $ A11: int  1 6 0 5 0 0 0 0 0 0 ...
##  $ A12: int  1 1 1 0 1 0 0 1 1 0 ...
##  $ A14: int  202 43 280 100 120 360 164 80 180 52 ...
##  $ A15: int  0 560 824 3 0 0 31285 1349 314 1442 ...
##  $ R1 : int  1 1 1 1 1 1 1 1 1 1 ...
```

Split data into two distinct data sets, using random sampling without replacement

```
set.seed(123)
indexes <- sample(1:nrow(data), size = nrow(data) * 0.7)
data1 <- data[indexes,]
data2 <- data[-indexes,]
```

Perpare the data

```
library(caret)

# Convert the target variable to factor
data1$R1 <- as.factor(data1$R1)
data2$R1 <- as.factor(data2$R1)
```

Set up cross-validation, split data1 into different pieces (folds).

```
# Setting up k-fold cross-validation
cvCtrl <- trainControl(method = "cv", number = 10)  # 10-fold Cross validation
```

Train K-NN model: the model will take turns using 9 of the folds for training and 1 fold for validation, iterating this process 10 times so that each fold is used as a validation set exactly once.

```
# Define the k values you want to try
k_values = expand.grid(k = seq(from = 1, to = 10, by = 1))

# Train the k-NN model with different k values
knnFit <- train(
  R1 ~ .,
  data = data1,
  method = "knn",
  trControl = cvCtrl,
  preProcess = c("center", "scale"),
  tuneGrid = k_values
)

# Summary of the k-NN model
# print(knnFit)

best_k <- knnFit$bestTune$k
best_accuracy <- max(knnFit$results$Accuracy)
cat("Best k:", best_k, "\n")
```

```
## Best k: 7
```

```
cat("Best accuracy:", best_accuracy, "\n")
```

```
## Best accuracy: 0.8595169
```

Report Acuracy: The best number of neighbors, 7, has 0.859 accuracy, or how often the model correctly classifies the samples in the evaluation set. Evaluate the model on data2

```
# Evaluate the model on the test set (data2)
predictions <- predict(knnFit, newdata = data2)
```

Summary

84 (TN): The number of times the model correctly predicted class '0'.

78 (TP): The number of times the model correctly predicted class '1'.

9 (FP): The number of times the model incorrectly predicted class '0' as class '1'.

26 (FN): The number of times the model incorrectly predicted class '1' as class '0'.

Model Accuracy: 82.23%

Confidence Interval: 95%

No Information Rate: 0.5584, success rate with no model prediction.

P-Value: indicates model is significantly better than a no-information rate.

Mcnemar's Test P-Value: Used to compare the performance of two classifiers.

It appears the classifier performs reasonably well given the context of this data set.

```
confusionMatrix(predictions, data2$R1)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 84  9
##          1 26 78
##
```

```
##                Accuracy : 0.8223
##                  95% CI : (0.7617, 0.873)
##     No Information Rate : 0.5584
##     P-Value [Acc > NIR] : 4.081e-15
##
##                   Kappa : 0.647
##
##  Mcnemar’s Test P-Value : 0.006841
##
##             Sensitivity : 0.7636
##             Specificity : 0.8966
##          Pos Pred Value : 0.9032
##          Neg Pred Value : 0.7500
##              Prevalence : 0.5584
##          Detection Rate : 0.4264
##    Detection Prevalence : 0.4721
##       Balanced Accuracy : 0.8301
##
##        ’Positive’ Class : 0
##
```