p8106_hw5_yg2625

Yue Gu

April 27, 2019

This problem involves the OJ data set which is part of the ISLR package. The data contains 1070 purchases where the customer either purchased Citrus Hill or Minute Maid Orange Juice. A number of characteristics of the customer and product are recorded. Use set.seed() for reproducibility. Create a training set containing a random sample of 800 observations, and a test set containing the remaining observations.

load data

```
data(OJ)
oj_data = OJ %>%
  janitor::clean_names()


# create a training set containing 800 obs, and a test set containing the remaining obs
set.seed(1)
rowTrain = createDataPartition(y = oj_data$purchase,
                               p = 799/1070,
                               list = F)
train_data = oj_data[rowTrain, ]
test_data = oj_data[-rowTrain, ]
```

(a) Fit a support vector classifier (linear kernel) to the training data with Purchase as the response and the other variables as predictors. What are the training and test error rates?

```
ctrl <- trainControl(method = "cv")

set.seed(1)
# fit model
svml.fit <- train(purchase ~ .,
                  data = train_data,
                  method = "svmLinear2",
                  preProcess = c("center", "scale"),
                  tuneGrid = data.frame(cost = exp(seq(-5,-1,len=50))),
                  trControl = ctrl)
# model output
svml.fit$finalModel
```

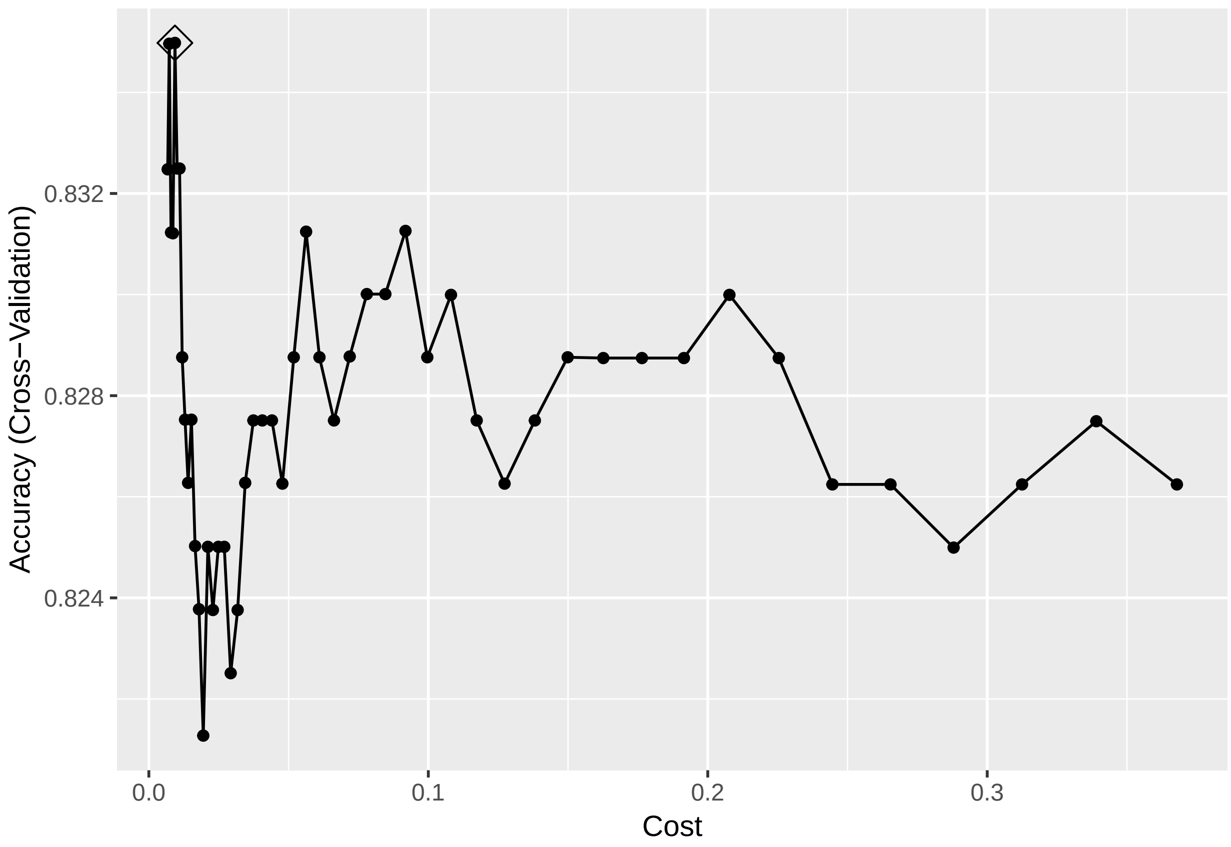
```
##
## Call:
## svm.default(x = as.matrix(x), y = y, kernel = "linear", cost = param$cost,
##     probability = classProbs)
##
##
## Parameters:
```

```
##    SVM-Type:  C-classification
##  SVM-Kernel:  linear
##        cost:  0.00933981
##       gamma:  0.05882353
##
## Number of Support Vectors:  444
```

```
# best tunning parameter
svml.fit$bestTune
```

```
##         cost
## 5 0.00933981
```

```
# Accuracy plot
ggplot(svml.fit, highlight = TRUE)
```



```
# training error rate
pred_train = predict(svml.fit)
mean(train_data$purchase != pred_train)
```

```
## [1] 0.16125
```

```
# test error rate
pred_test = predict(svml.fit, newdata = test_data, type = "raw")
mean(test_data$purchase != pred_test)
```

```
## [1] 0.1703704
```

The training error is 0.161, the test error is 0.170.