HW5

David DeStephano

May 5, 2020

```
library(ISLR)
library(tidyverse)
library(e1071)
library(caret)
library(kernlab)
```

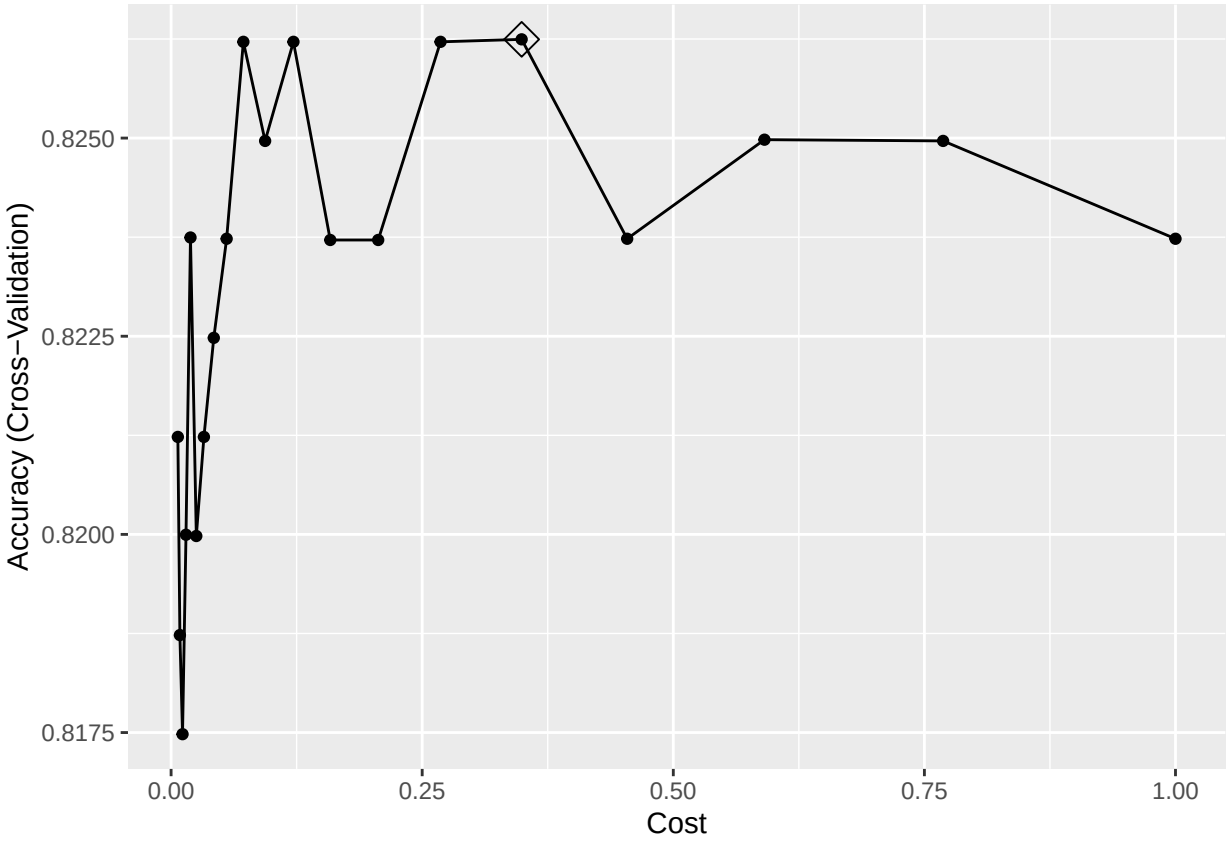
This problem involves the OJ data set which is part of the ISLR package. The data contains 1070 purchases where the customer either purchased Citrus Hill or Minute Maid Orange Juice. A number of characteristics of the customer and product are recorded. Create a training set containing a random sample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

```
data(OJ)

set.seed(1)
rowTrain = createDataPartition(OJ$Purchase,
                               p=799/1070,
                               list=F)

train <- OJ[rowTrain, ]
test <- OJ[-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)

svml.fit <- train(Purchase~.,
                  data = train,
                  method = "svmLinear2",
                  preProcess = c("center", "scale"),
                  tuneGrid = data.frame(cost =
                                        exp(seq(-5,0,len=20))),
                  trControl = ctrl)

ggplot(svml.fit, highlight = TRUE)
```




```
getTrainPerf(svml.fit)
```

```
##   TrainAccuracy TrainKappa     method
## 1     0.8262457  0.6286221 svmLinear2
```

```
pred.linear <- predict(svml.fit, newdata = test)

confusionMatrix(data = pred.linear,
                reference = test$Purchase)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  CH  MM
##         CH 144  20
##         MM  21  85
##
##                Accuracy : 0.8481
##                  95% CI : (0.7997, 0.8888)
##     No Information Rate : 0.6111
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.6811
##
##  Mcnemar's Test P-Value : 1
```

```
##
##             Sensitivity : 0.8727
##             Specificity : 0.8095
##          Pos Pred Value : 0.8780
##          Neg Pred Value : 0.8019
##              Prevalence : 0.6111
##          Detection Rate : 0.5333
##    Detection Prevalence : 0.6074
##       Balanced Accuracy : 0.8411
##
##        'Positive' Class : CH
##
```

The training accuracy was 82.6% and the test accuracy was 84.8%

The error rates are:

```
# training
pred.svml.train <- predict(svml.fit)
mean(pred.svml.train != train$Purchase)
```

```
## [1] 0.16875
```

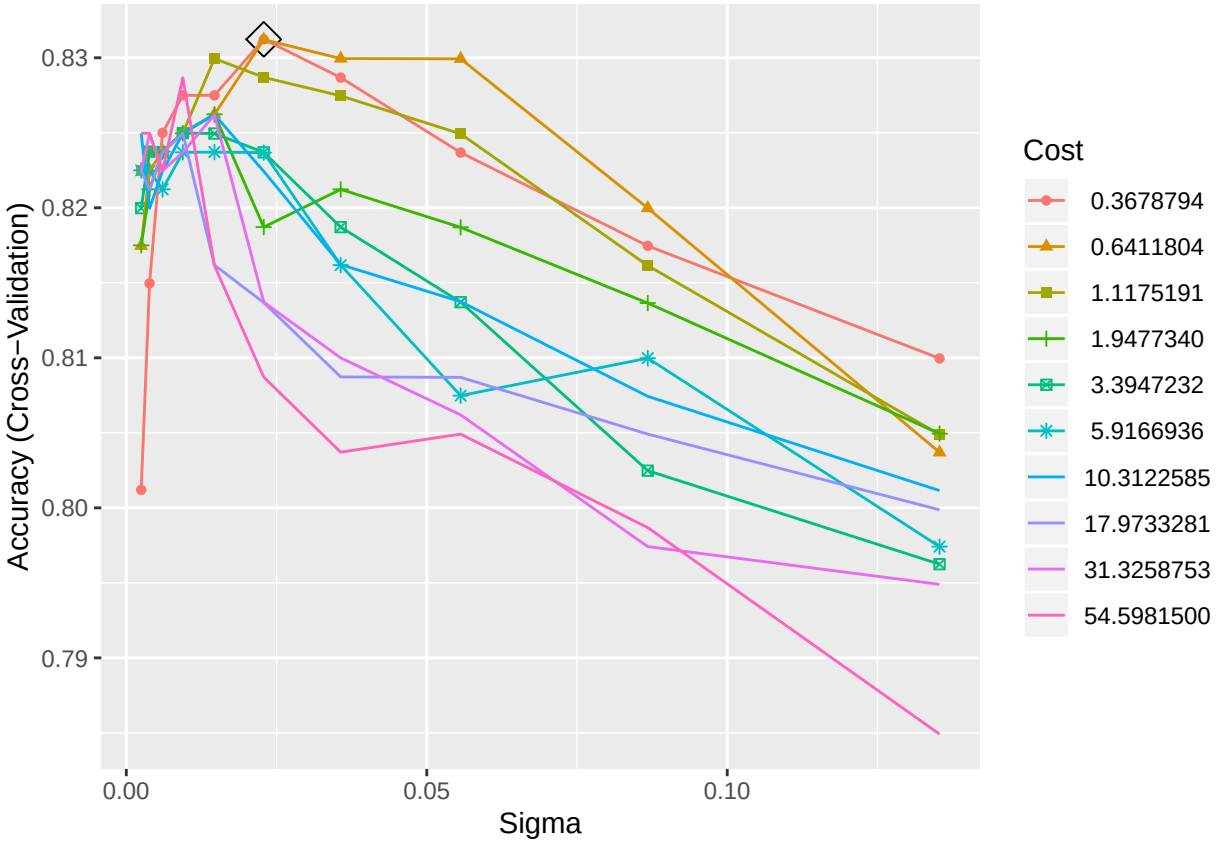
```
# test
pred.svml.test <- predict(svml.fit, newdata = test)
mean(pred.svml.test != test$Purchase)
```

```
## [1] 0.1518519
```

(b) Fit a support vector machine with a radial kernel to the training data. What are the training and test error rates?

```
svmr.grid <- expand.grid(C = exp(seq(-1,4,len=10)),
                         sigma = exp(seq(-6,-2,len=10)))

set.seed(1)

svmr.fit <- train(Purchase~.,
                  data=train,
                  method = "svmRadial",
                  preProcess = c("center", "scale"),
                  tuneGrid = svmr.grid,
                  trControl = ctrl)

ggplot(svmr.fit, highlight = TRUE)
```

```
## Warning: The shape palette can deal with a maximum of 6 discrete values
## because more than 6 becomes difficult to discriminate; you have
## 10. Consider specifying shapes manually if you must have them.

## Warning: Removed 40 rows containing missing values (geom_point).
```




```
getTrainPerf(svmr.fit)
```

```
##   TrainAccuracy TrainKappa    method
## 1     0.8312295  0.6393021 svmRadial
```

```
pred.rad <- predict(svmr.fit, newdata = test)

confusionMatrix(data = pred.rad,
                reference = test$Purchase)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  CH  MM
##         CH 147  24
##         MM  18  81
##
##                Accuracy : 0.8444
##                  95% CI : (0.7956, 0.8855)
##     No Information Rate : 0.6111
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.6693
##
##  Mcnemar's Test P-Value : 0.4404
```

```
##
##             Sensitivity : 0.8909
##             Specificity : 0.7714
##          Pos Pred Value : 0.8596
##          Neg Pred Value : 0.8182
##              Prevalence : 0.6111
##          Detection Rate : 0.5444
##    Detection Prevalence : 0.6333
##       Balanced Accuracy : 0.8312
##
##        'Positive' Class : CH
##
```

The training accuracy was 83.1% and the test accuracy was 84.4%

The error rates are:

```
# training
pred.svmr.train <- predict(svmr.fit)
mean(pred.svmr.train != train$Purchase)
```

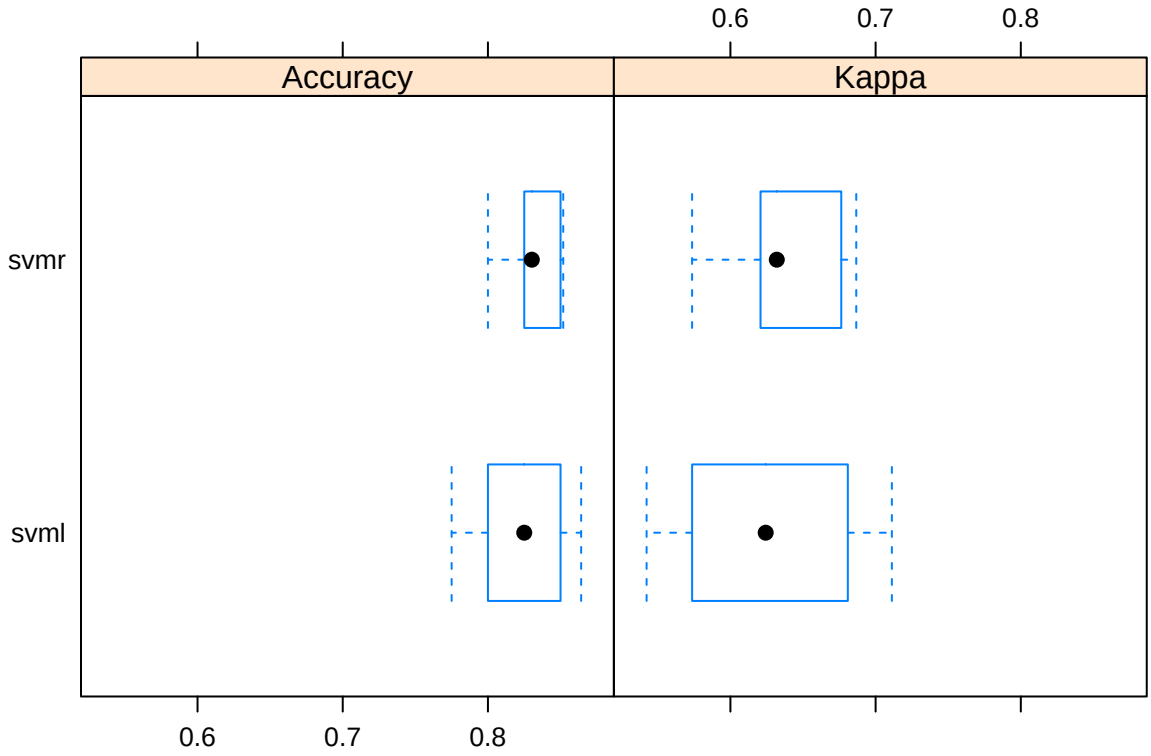
```
## [1] 0.16
```

```
# test
pred.svmr.test <- predict(svmr.fit, newdata = test)
mean(pred.svmr.test != test$Purchase)
```

```
## [1] 0.1555556
```

Compare the two models:

```
resamp <- resamples(list(svmr = svmr.fit, svml = svml.fit))
bwplot(resamp)
```


Accuracy
Kappa
svmr
svml
0.6
0.7
0.8
0.6
0.7
0.8