**Module 4 Exercises**

1. (15 points) Simulate a single predictor and a nonlinear relationship, such as a sin wave shown in Fig. 7.7 of textbook, and investigate the relationship between the cost, 𝜺, and kernel parameters for a support vector machine model:

set.seed(100)

x <- runif(100, min = 2, max = 10)

y <- sin(x) + rnorm(length(x)) \* .25

sinData <- data.frame(x = x, y = y)

plot(x, y)

## Create a grid of x values to use for prediction > dataGrid <- data.frame(x = seq(2, 10, length = 100)).

(a) Fit different models using a radial basis function and different values of the cost (the C parameter) and 𝜺. Plot the fitted curve. For example:

library(kernlab)

rbfSVM <- ksvm(x = x, y = y, data = sinData,

kernel ="rbfdot", kpar = "automatic",

C = 1, epsilon = 0.1)

modelPrediction <- predict(rbfSVM, newdata = dataGrid)

## This is a matrix with one column. We can plot the

## model predictions by adding points to the previous plot points(x = dataGrid$x, y = modelPrediction[,1], type = "l", col = "blue")

(b) The σ parameter can be adjusted using the kpar argument, such as kpar = list(sigma = 1). Try different values of σ to understand how this parameter changes the model fit. How do the cost, 𝜺, and σ values affect the model?

For this problem, you will perform your analysis in R.

1. (15 points) For the Tecator data described in chapter 6 of the textbook, build SVM, neural network, MARS, and KNN models. Since neural networks are especially sensitive to highly correlated predictors, does pre-processing using PCA help the model?

For this problem, you will perform your analysis in R.

1. (15 points) Start R and use these commands to load the data:

library(AppliedPredictiveModeling)

data(chemicalManufacturingProcess)

The matrix processPredictors contains the 57 predictors (12 describing the input biological material and 45 describing the process predictors) for the 176 manufacturing runs. yield contains the percent yield for each run. This describes the data for a chemical manufacturing process. Use data imputation, data splitting, and pre-processing steps and train several nonlinear regression models.

(a) Which nonlinear regression model gives the optimal resampling and test set performance?

(b) Which predictors are most important in the optimal nonlinear regression model? Do either the biological or process variables dominate the list? How do the top ten important predictors compare to the top ten predictors from the optimal linear model?

(c) Explore the relationships between the top predictors and the response for the predictors that are unique to the optimal nonlinear regression model. Do these plots reveal intuition about the biological or process predictors and their relationship with yield?

For this problem, you will perform your analysis in R.

1. (15 points) Recreate the simulated data as shown below:

library(mlbench)

set.seed(200)

simulated <- mlbench.friedman1(200, sd = 1) > simulated <- cbind(simulated$x, simulated$y)

simulated <- as.data.frame(simulated)

colnames(simulated)[ncol(simulated)] <- "y"

(a) Fit a random forest model to all of the predictors, then estimate the variable importance scores:

library(randomForest)

library(caret)

model1 <- randomForest(y ~ ., data = simulated, importance = TRUE,

ntree = 1000)

rfImp1 <- varImp(model1, scale = FALSE)

Did the random forest model significantly use the uninformative predictors (V6 – V10)?

(b) Now add an additional predictor that is highly correlated with one of the informative predictors. For example:

simulated$duplicate1 <- simulated$V1 + rnorm(200) \* .1 > cor(simulated$duplicate1, simulated$V1)

Fit another random forest model to these data. Did the importance score for V1 change? What happens when you add another predictor that is also highly correlated with V1?

For this problem, you will perform your analysis in R.

1. (15 points) Use a simulation to show tree bias with different granularities. For this problem, you will perform your analysis in R.
2. (15 points) In stochastic gradient boosting, the bagging fraction and learning rate will govern the construction of the trees as they are guided by the gradient. Although the optimal values of these parameters should be obtained through the tuning process, it is helpful to understand how the magnitudes of these parameters affect magnitudes of variable importance. Figure 8.24 below provides the variable importance plots for boosting using two extreme values for the bagging fraction (0.1 and 0.9) and the learning rate (0.1 and 0.9) for the solubility data. The left-hand plot has both parameters set to 0.1, and the right-hand plot has both set to 0.9:

(a) Why does the model on the right focus its importance on just the first few of predictors, whereas the model on the left spreads importance across more predictors?

(b) Which model do you think would be more predictive of other samples?

(c) How would increasing interaction depth affect the slope of predictor importance for either model in Fig. 8.24?

