Applied Data Mining: Homework #7

Due on December 6,2017

Instructor: Hasan Kurban

Naimesh Chaudhari

December 6, 2017

Problem 1

In this problem, you are asked to use SVM to predict whether a given car gets high or low gas mileage based on the Auto data set. The data set can be obtained as follows:

```
> library(ISLR)
> View(Auto)
```

1.1 Create a binary variable that takes on a 1 for cars with gas mileage above the median, and a 0 for cars with gas mileage below the median. Add this variable to the data as a new variable and name it as "mpglevel" (mpglevel is the response variable for questions 1.2 and 1.3).

R Code

Listing 1: Sample R Script With Highlighting

```
df = Auto
set.seed(1234)
md <- median(df$mpg)
df$mpglevel <- ifelse(df$mpg >= md,1,0)
```

1.2 Fit a linear support vector classifier to the data with various values of cost (cost = c(0.01, 0.1, 1, 5, 10, 100)), in order to predict whether a car gets high or low gas mileage. Report the cross-validation errors associated with different values of this parameter. Comment on your results, i.e., what is the cost value for the model that has the lowest cross-validation error?

R Code

Listing 2: Sample R Script With Highlighting

```
rndSample <- sample(1:nrow(df), 100)
tr <- df[rndSample, ]
ts <- df[-rndSample, ]
tune.out = tune(svm, mpglevel ~ ., data = tr,ranges = list(cost = c(0.01, 0.1,
    1, 5, 10, 100)))
summary(tune.out) #cost = 100, error = 8.04%
```

Cross-validation Errors and Discussion of the Results

As the cost went higher the error rate seems to have gone down, and at cost 100 we see the lowerst error rate of 8.04 percent.

1.3 Now repeat (1.2), this time using SVMs with radial and polynomial basis kernels, with different values of gamma (c(0.01, 0.1, 1, 5, 10, 100)) and degree (c(2, 3, 4)) and cost (c(0.1, 1, 5, 10)). Use the cost and degree parameters values for polynomial kernels. The cost and gamma parameters values are given for radial basis kernels. Comment on your results, i.e., what are the parameters values (cost, degree, gamma) for the model that has the lowest cross-validation error?

R Code

Listing 3: Sample R Script With Highlighting

```
tune.out = (tune(svm, mpglevel ~ ., data = tr,kernel="radial",
                 ranges = list(cost = c(0.01, 0.1, 1, 5, 10, 100),degree = c
                     (2, 3, 4))))
summary(tune.out) #cost = 100, degree = 2, error = 8.38%
tune.out = (tune(svm, mpglevel ~ ., data = tr,kernel="polynomial",
                 ranges = list(cost = c(0.01, 0.1, 1, 5, 10, 100),gamma = c
                     (0.01, 0.1, 1, 5, 10, 100))))
summary(tune.out)  #cost = 5, gamma = 0.1, error = 12.58%
```

Discussion of Results

For the radial test it seems that the best model is where the cost is 100 and degree is 2 which resuted in an error rate of 8.35 percent and for the poly model it seems at cost 5 and gamma 0.1 we see the best result with an error rate of 12.58 percent. It seems that the result we got in step 1.2 is the model with the lowest error rate.

Problem 2

Load the Caravan data set as follows and answer the questions below.

```
> library(ISLR)
> View(Caravan)
```

2.1 Create a training set consisting of the first 1,000 observations, and a test set consisting of the remaining observations. The class variable is "Purchase" whose values are "No" and "Yes". Transform "No" to "0" "Yes" to 1. Place the R code below.

R Code

Listing 4: Sample R Script With Highlighting

```
df <- Caravan
set.seed(1234)
df$Purchase <- ifelse(df$Purchase == "Yes",1,0)
tr <- df[1:1000,]
ts <- df[1001:5822,]
```

2.2 Fit a boosting model to the training set with Purchase as the response and the other variables as predictors. Use 1,000 trees, and a shrinkage value of 0.01. Which predictors appear to be the most important? (R package for boosting: "gbm".)

R Code

Listing 5: Sample R Script With Highlighting

```
bst <- gbm(Purchase ~ ., data = tr, distribution = "gaussian"
           , n.trees = 1000, shrinkage = 0.01)
summary(bst)  # Variable PPERSAUT & MKOOPKLA have the most significanve
```

Most Important Predictors

Variable PPERSAUT nad MKOOPKLA have the most significance ...

2.3 Use the boosting model to predict the response on the test data. Predict that a person will make a purchase if the estimated probability of purchase is greater than 20%. Form a confusion matrix. What fraction of the people predicted to make a purchase do in fact make one?

R Code

Listing 6: Sample R Script With Highlighting

```
prob <- predict(bst, ts, n.trees = 1000, type = "response")
pred <- ifelse(prob > 0.2, 1, 0)
table(ts$Purchase, pred)
#25% of the people predicted to make a purchase actually did make a purchase
```

Results

25 percent of the people predicted to make a purchase actually did make a purchase

Problem 3

In this question, you are asked to compare Naive Bayes and K-nearest Neighbors (KNN) algorithms over Ionosphere Data Set. Here is the beginning of an R session that allows us to read this data from the web into our local R session:

```
> library(data.table)
> library("curl")
> mydata <- fread("https://archive.ics.uci.edu/ml/machine-learning-databases/
                   ionosphere/ionosphere.data")
> mydata <- as.data.frame(mydata)
> mydata <- mydata[,-2] #remove the second variable
```

3.1 Create a training data set containing a random sample of 300 data points and a test set containing the remaining observations. Name the training data and test data as mydata.training and mydata.testing, respectively. Place the R code below. You will use mydata.training and mydata.testing to answer rest of the questions. Thus, create them once and use mydata.training to train the models (classifiers) and mydata.testing to test the models. The last variable variable (35th variable in the data) is the response and the other variables are predictors.

R Code

Listing 7: Sample R Script With Highlighting

```
set.seed(1234)
rndSample <- sample(1:nrow(mydata), 300)
mydata.training <- mydata[rndSample, ]
mydata.testing <- mydata[-rndSample, ]
```

3.2 Train a naive bayes classifier using 10-fold cross-validation over mydata.training. Use this model to predict the observations in mydata.testing. Form a confusion matrix and report the error rate of the classifier over mydata.testing.

R Code

Listing 8: Sample R Script With Highlighting

```
x <- mydata.training[,-34]
y <- mydata.training$V35
model = train(x,y,'nb',trControl=trainControl(method='cv',number=10))
table(predict(model$finalModel,x)$class,y)  # 8% Error on train data
table(predict(model$finalModel,mydata.testing[,-34])$class,mydata.testing$V35)
    #3.92% Error on test data
```

Confusion Matrix and Error Rate

The error rate is 3.92 percent ...

3.3 Perform KNN on mydata.training, with several values of k (k= (2,5,10,50), in order to classify radar returns from the ionosphere. What test errors do you obtain over my mydata.testing? Report the confusion matrices. Which value of k seems to perform the best on the test data.

R Code

Listing 9: Sample R Script With Highlighting

```
set.seed(1)
x <- mydata.training[,-34]
y <- mydata.training$V35

test.x = mydata.testing[,-34]
test.y = mydata.testing$V35

knn.pred=knn(x,test.x,y,k=50)
table(knn.pred,test.y)

(13+31)/51
# k = 2 9.80% Error
# k = 5 11.76% Error
# k = 10 11.76% Error
# k = 50 31.37% Error
```

Discussion and Results

At K = 2 BB = 16, BG = 0, GB = 4, GG = 31 Error Percent 9.8
At K = 5 BB = 14, BG = 0, GB = 6, GG = 31 Error Percent 11.76
At K = 10 BB = 14, BG = 0, GB = 6, GG = 31 Error Percent 11.76
At K = 50 BB = 4, BG = 0, GB = 16, GG = 31 Error Percent 31.37

3.4 Compare Naive Bayes classifier with KNN, i.e., Which one performed better?

Comparison of the Algorithms

It seens the Naive Bayes classifier model performed best with a error percent of 3.91 percent.