STAT488_HW1

2022-09-07

```
library(carData)
library(class)
```

1) Pg 52, Ex2: Classification vs Regression, Inference vs Prediction, find n and p

    a) regression problem, interested in inference because we are interested in the factors that affect salary, n = 500 since it is the top 500 firms in US, p would be all the firms in the US
    b) classification problem since "Success" and "Failure" are discrete variables, interested in prediction, n = 20 similar products and p would be all products
    c) regression problem, "predicting percentages", n = data per week and p is all the data for the year

2) Pg 52, Ex4: Real like applications for statistical learning

    a) Classification (Qualitative)
        - predicting whether someone would have a heart attack(Yes, No response) based on their sex, blood sugar, age, level of activity, average heart rate (predictors)
        - Predicting if an email is spam (Yes, No response) or not based on sender address, time sent, format of message, contents of message (key words), frequency of mail (predictors)
        - iris data set, predicting which species (Setosa, Versicolor, Virginica) depending on predictors like Sepal width/length, Petal width/length (predictors)
    b) Regression (Quantitative)
        - Predict the salary of teachers (response) based on how many classes they teach, level of degree, years of experience (predictors)
        - Predict price of a house (response) based on sqft, bedrooms/bathrooms, crime rate in neighborhood, distance from school(predictors)
        - Determine whether the displacement (predictor) of an engine in a car is a factor to the MSRP of the vehicle (response), example of inference

3) Pg 52, Ex5: Advantages and disadvantages of very flexible vs less flexible for regression or classification, what circumstances might a more flexible approach be preferred to less flexible approach, When might a less flexible approach be preferred?

    Flexible models can fit more complex problems but may fit too well (no errors = overfitting). Less flexible requires less parameters and can be interpreted easier, but it may not be the most accurate. If model is underfitted, then a more flexible apporoach would be preferred. If a dataset is smaller and has less parameters, then a less flexible approach is preferred.

4) Pg54, Ex8:

    a) get data into R, call loaded data "college"

```
data <- "/Users/melchorronquillo/Desktop/Data/College.csv"
college <- data.frame(read.csv(data))
```

```
    b) Look at data using View()
      ***The csv does not have a row with college names, getting error
```

```
#rownames(college) <- college[, 1]
#college <- college[, -1]
```

c) i: use summary() to produce numerical summary of variables in data set

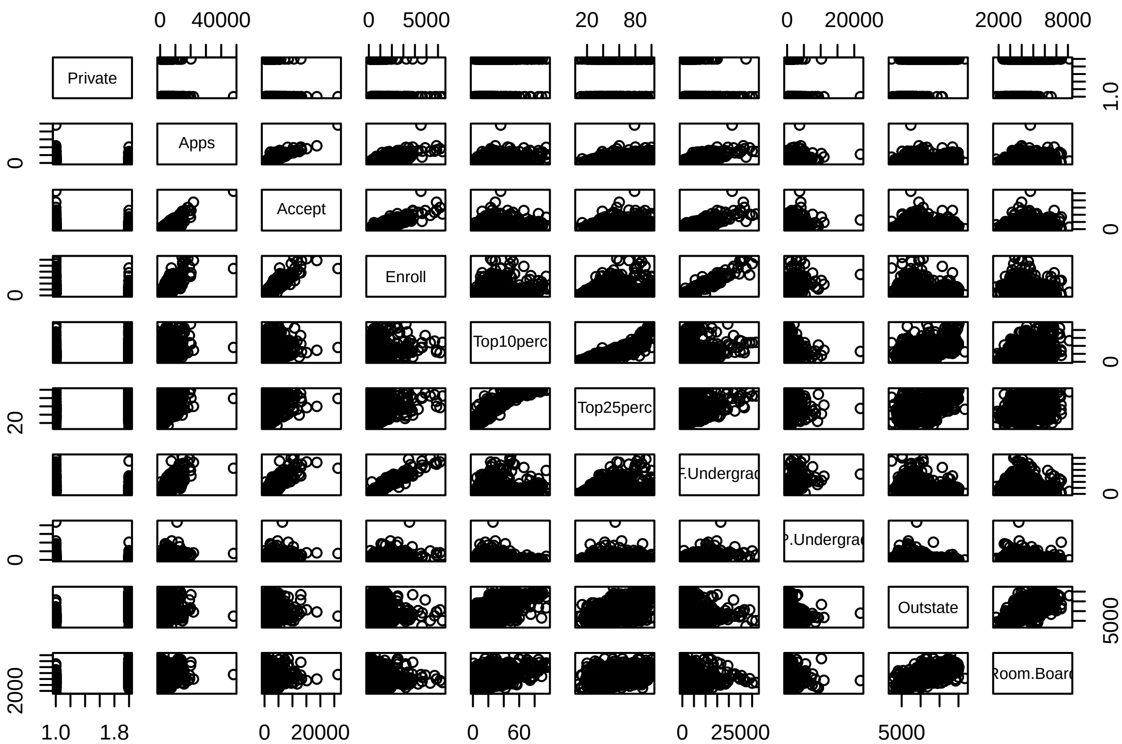
```
summary(college)
```

```
##    Private               Apps           Accept          Enroll
##  Length:777         Min.   :   81   Min.   :   72   Min.   :  35
##  Class :character   1st Qu.:  776   1st Qu.:  604   1st Qu.: 242
##  Mode  :character   Median : 1558   Median : 1110   Median : 434
##                     Mean   : 3002   Mean   : 2019   Mean   : 780
##                     3rd Qu.: 3624   3rd Qu.: 2424   3rd Qu.: 902
##                     Max.   :48094   Max.   :26330   Max.   :6392
##    Top10perc       Top25perc      F.Undergrad     P.Undergrad
##  Min.   : 1.00   Min.   :  9.0   Min.   :  139   Min.   :    1.0
##  1st Qu.:15.00   1st Qu.: 41.0   1st Qu.:  992   1st Qu.:   95.0
##  Median :23.00   Median : 54.0   Median : 1707   Median :  353.0
##  Mean   :27.56   Mean   : 55.8   Mean   : 3700   Mean   :  855.3
##  3rd Qu.:35.00   3rd Qu.: 69.0   3rd Qu.: 4005   3rd Qu.:  967.0
##  Max.   :96.00   Max.   :100.0   Max.   :31643   Max.   :21836.0
##     Outstate       Room.Board       Books           Personal
##  Min.   : 2340   Min.   :1780   Min.   :  96.0   Min.   : 250
##  1st Qu.: 7320   1st Qu.:3597   1st Qu.: 470.0   1st Qu.: 850
##  Median : 9990   Median :4200   Median : 500.0   Median :1200
##  Mean   :10441   Mean   :4358   Mean   : 549.4   Mean   :1341
##  3rd Qu.:12925   3rd Qu.:5050   3rd Qu.: 600.0   3rd Qu.:1700
##  Max.   :21700   Max.   :8124   Max.   :2340.0   Max.   :6800
##       PhD            Terminal       S.F.Ratio      perc.alumni
##  Min.   :  8.00   Min.   : 24.0   Min.   : 2.50   Min.   : 0.00
##  1st Qu.: 62.00   1st Qu.: 71.0   1st Qu.:11.50   1st Qu.:13.00
##  Median : 75.00   Median : 82.0   Median :13.60   Median :21.00
##  Mean   : 72.66   Mean   : 79.7   Mean   :14.09   Mean   :22.74
##  3rd Qu.: 85.00   3rd Qu.: 92.0   3rd Qu.:16.50   3rd Qu.:31.00
##  Max.   :103.00   Max.   :100.0   Max.   :39.80   Max.   :64.00
##      Expend        Grad.Rate
##  Min.   : 3186   Min.   : 10.00
##  1st Qu.: 6751   1st Qu.: 53.00
##  Median : 8377   Median : 65.00
##  Mean   : 9660   Mean   : 65.46
##  3rd Qu.:10830   3rd Qu.: 78.00
##  Max.   :56233   Max.   :118.00
```

c) ii: use pairs() to produce scatterplot matrix of first 10 columns

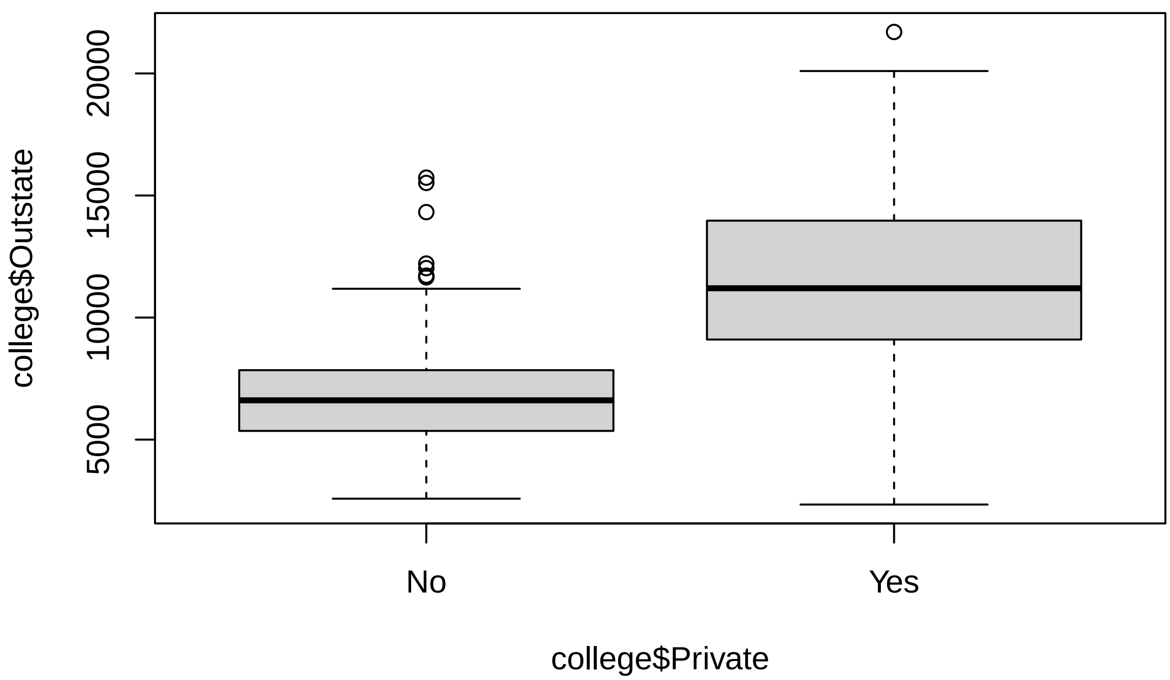
```
#Error in pairs.default(college[, 1:10]) : non-numeric argument to 'pairs'

#change private variable to numeric values (0,1) instead of (no, yes)
college[,1] = as.factor(college[,1])
pairs(college[,1:10])
```



c) iii: use plot() to produce side-by-side boxplot of Outstate vs Private

```
boxplot(college$Outstate~college$Private)
```



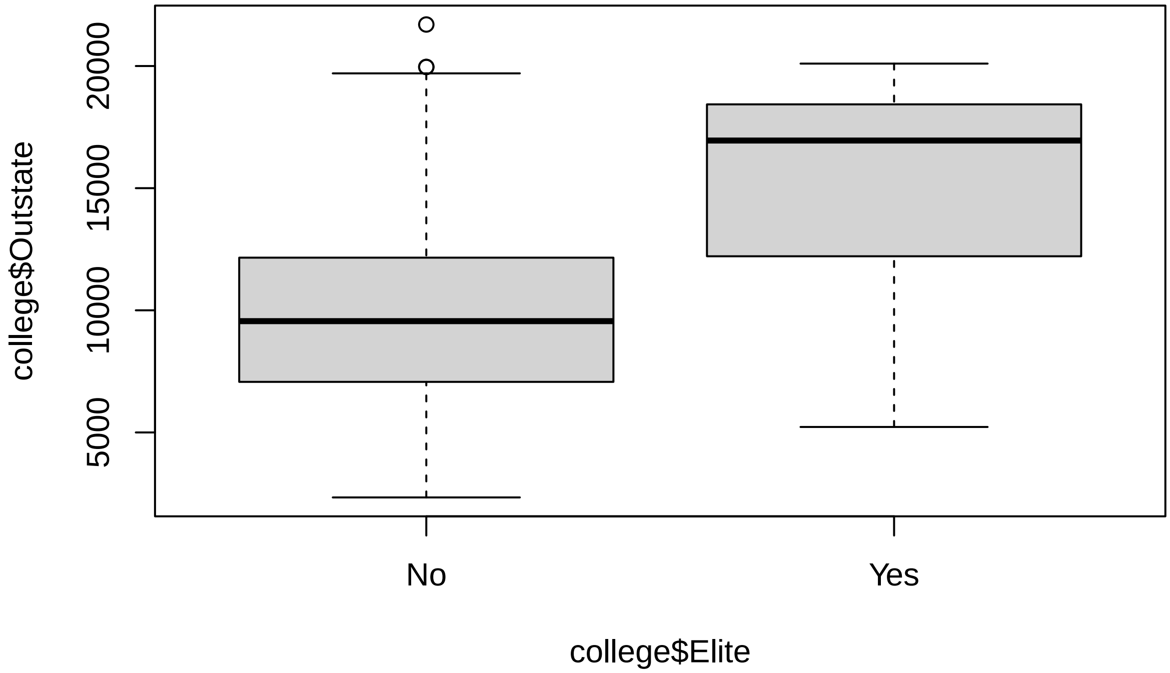
c) iv: create Elite variable, bin Top10Perc variable, use summary function to see how many elite universities there are, plot Outstate vs Elite

```
Elite <- rep("No", nrow(college))
Elite[college$Top10perc > 50] <- "Yes"
Eltie <- as.factor(Elite)
```

```r
college <- data.frame(college, Elite)
colelite <- college[college$Elite == "Yes", ]
nrow(colelite) # 78 elite colleges
```

```
## [1] 78
```

```r
boxplot(college$Outstate ~ college$Elite)
```



c) v: use hist() to produce histograms with differing number of bins for quantitative variables

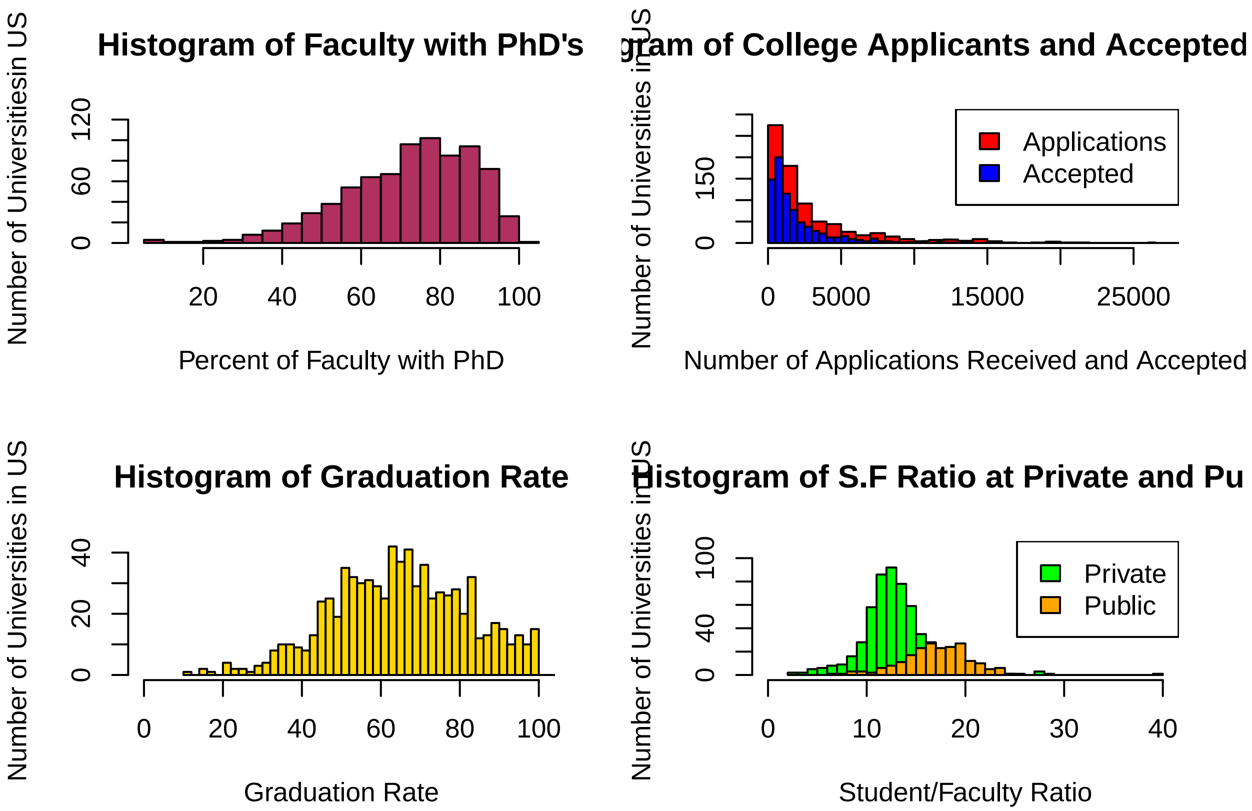
```r
par(mfrow = c(2, 2))
hist(college$PhD, breaks = 25, xlab = "Percent of Faculty with PhD",
     ylab = "Number of Universitiesin US", main = "Histogram of Faculty with PhD's",
     col = "maroon",ylim = c(0,125))

#histogram of applications vs accepted
#max(college$Accept)
#max(college$Enroll)
hist(college$Apps, breaks = 50, xlim = c(0,27000), ylim = c(0,300), col='red',
     main='Histogram of College Applicants and Accepted Students',
     xlab = "Number of Applications Received and Accepted" ,
     ylab = "Number of Universities in US")
hist(college$Accept, breaks = 50, col='blue', add=TRUE)
legend('topright', c('Applications', 'Accepted'), fill = c('red','blue'))

hist(college$Grad.Rate, xlab = "Graduation Rate", main = "Histogram of Graduation Rate",
     ylab = "Number of Universities in US",
     col = "gold", breaks = 75, xlim = c(0,100))


colpriv <- college[college$Private=="Yes", ]
colreg <- college[college$Private=="No", ]
```

```r
hist(colpriv$S.F.Ratio, breaks = 50, xlim = c(0,40), ylim = c(0,110), col='green',
     main='Histogram of S.F Ratio at Private and Public', xlab = "Student/Faculty Ratio",
     ylab = "Number of Universities in US")
hist(colreg$S.F.Ratio, breaks = 25, col='orange', add=TRUE)
legend('topright', c('Private', 'Public'), fill = c('green','orange'))
```



c) vi: Continue exploring the data, and provide a brief summary of what you discover.

```r
colpriv <- college[college$Private=="Yes", ]
nrow(colpriv)
```

```
## [1] 565
```

```r
colreg <- college[college$Private=="No", ]
nrow(colreg)
```

```
## [1] 212
```

```r
mean(colpriv$Grad.Rate) - mean(colreg$Grad.Rate)
```

```
## [1] 12.95578
```

```r
mean(colpriv$Apps-colpriv$Accept) - mean(colreg$Apps-colreg$Accept)
```

```
## [1] -1138.406
```

```
* There are 212 Private schools and 565 public schools in this dataset
* The average graduation rate at private schools are around 13% higher than public schools
* Public schools accept around an average of 1,139 more applicants than private schools
```

5) The training data set contains 175 observations classified as red or green. The test data set contains 1750 observations classified as either red or green.
    a) Perform k-nearest neighbor classification using the training data with k = 1. Use this model to predict the class of each observation in the training data set. How many observations were incorrectly classified? Is this good?

```
data2 = "/Users/melchorronquillo/Desktop/Data/PA_HW1_train.csv"
rgtrain<- data.frame(read.csv(data2, row.names=NULL))
k <- 1
greg <- knn(train = rgtrain[,1:2], test = rgtrain[,1:2], cl = rgtrain[,3], k = k)
#confusion matrix, diagonals should have most values correctly
table(rgtrain$col, greg)
```

```
##        greg
##         green red
##   green    75   0
##   red       0 100
```

```
#avg of trues and falses, gives misclassification rate
mean(rgtrain$col != greg)
```

```
## [1] 0
```

Perfect classification but it means nothing.The error rate is 0, and it classified 100% of the greens and reds correctly. However, with no errors, it means the model is overfitted.This is not a good model to use with other data since it is trained so specifically to this training data.

```
b) Again using k = 1, build a classification model with the training data set and use it to
classify the observations in the test data set. How many observations were incorrectly
classified? Is this good?
```

```
data3 = "/Users/melchorronquillo/Desktop/Data/PA_HW1_test.csv"
rgtest<- data.frame(read.csv(data3, row.names=NULL))
greg <- knn(train = rgtrain[,1:2], test = rgtest[,1:2], cl = rgtrain[,3], k = k)
table(rgtest$col, greg)
```

```
##        greg
##         green red
##   green   398 352
##   red     367 633
```

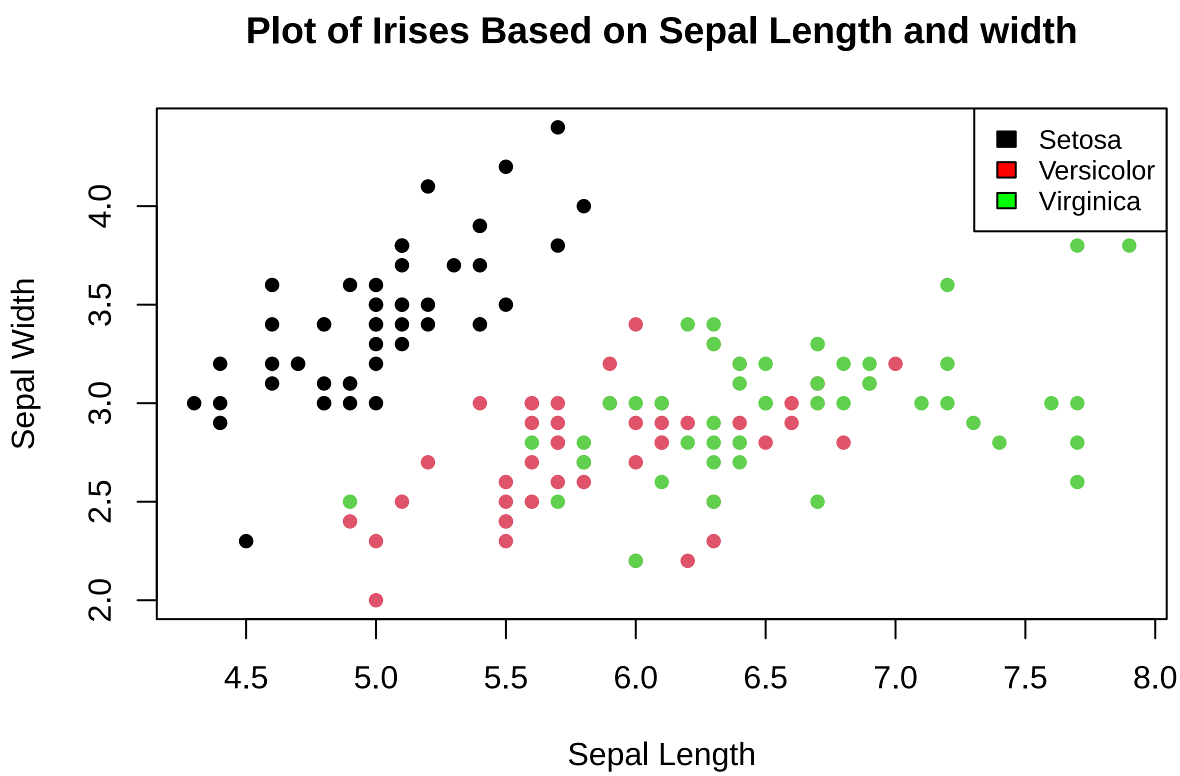
```
mean(rgtest$col != greg)
```

```
## [1] 0.4108571
```

With the test data, we get a misclassification rate of around 41%. Although the rate is lot higher than the training data, it is much better than just zero. The model is not overfitted and can still classify some of the data correctly (especially with keeping k = 1). With more training and finding a better value for K, this model will improve classifying the data.

6) Plot all irises based on their Sepal.Length and Sepal.Width values using different colors for each species.

```
iris <- data.frame(iris)
plot(iris$Sepal.Length, iris$Sepal.Width, col = iris$Species, pch = 16,
     xlab = "Sepal Length",
     ylab = "Sepal Width",
     main ="Plot of Irises Based on Sepal Length and width")
legend('topright', c('Setosa', 'Versicolor', 'Virginica'),
       fill = c('black','red', 'green'), cex = .8)
```



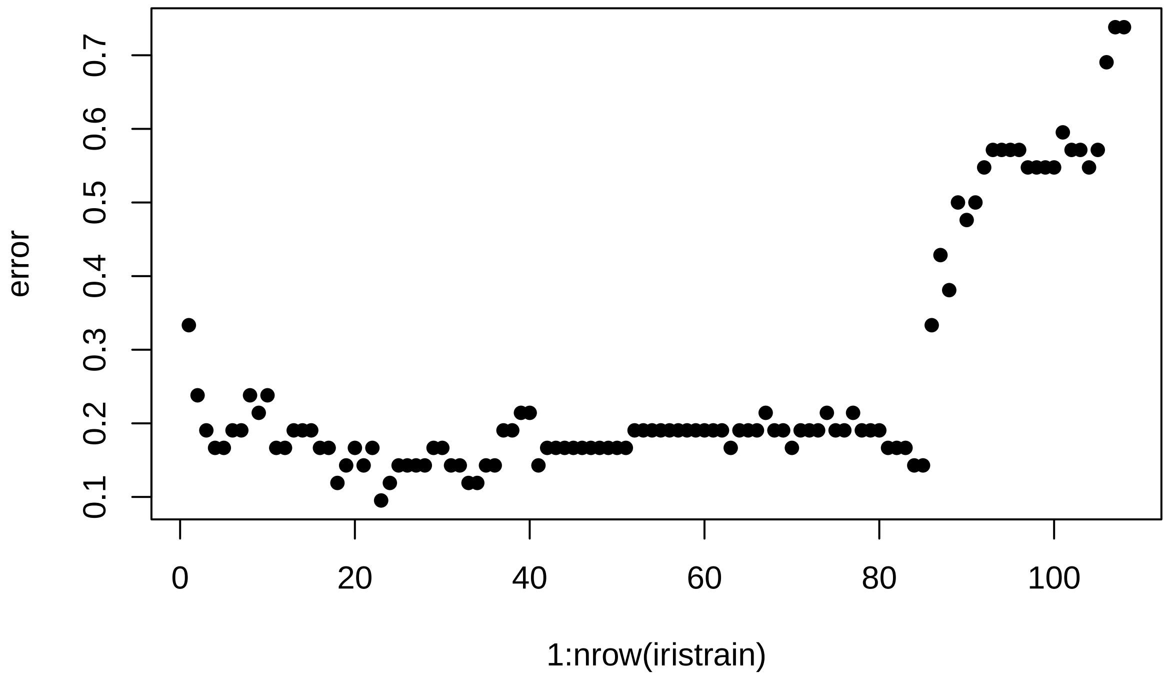
7) Perform knn analysis using the iris data with only Sepal.Length and Sepal.Width as predictors. Make predictions about the species of each iris and create a confusion matrix for this predictions.

```
k <- 1
# goes through every row of data set,generates uniform random numbers, scramble our data
u <- runif(nrow(iris))
iristrain <- iris[u <= .70,] # training set has 70% of data
irisholdout <- iris[u > .70,] # holdout set has 30% of data
#nrow(iristrain)
#nrow(irisholdout)
greggie <- knn(train = iristrain[,1:2], test = irisholdout[,1:2], cl = iristrain[,5], k = k)
table(irisholdout$Species, greggie)#confusion matrix
```

```
##             greggie
##              setosa versicolor virginica
##   setosa         17          0         1
##   versicolor      0          9         4
##   virginica       0          6         5
```

***trying to find lowest k and using it in model to reduce misclassification rate and improve confusion matrix

```
error <- c()
for (k in 1:nrow(iristrain)){
gregk <- knn(train = iristrain[,1:2], test = irisholdout[,1:2], cl = iristrain[,5], k = k)
error[k] <- mean(irisholdout[,5] != gregk)
}
#View(error)
plot(1:nrow(iristrain), error, pch = 16) # the value where k is the lowest is 20
```




```
gregnewk <- knn(train = iristrain[,1:2], test = irisholdout[,1:2], cl = iristrain[,5], k = 37)
table(irisholdout$Species, gregnewk)
```

```
##             gregnewk
##              setosa versicolor virginica
##   setosa         18          0         0
##   versicolor      2          8         3
##   virginica       0          3         8
```

```
mean(irisholdout$Species != gregnewk)
```

```
## [1] 0.1904762
```