Assignment 2 glass

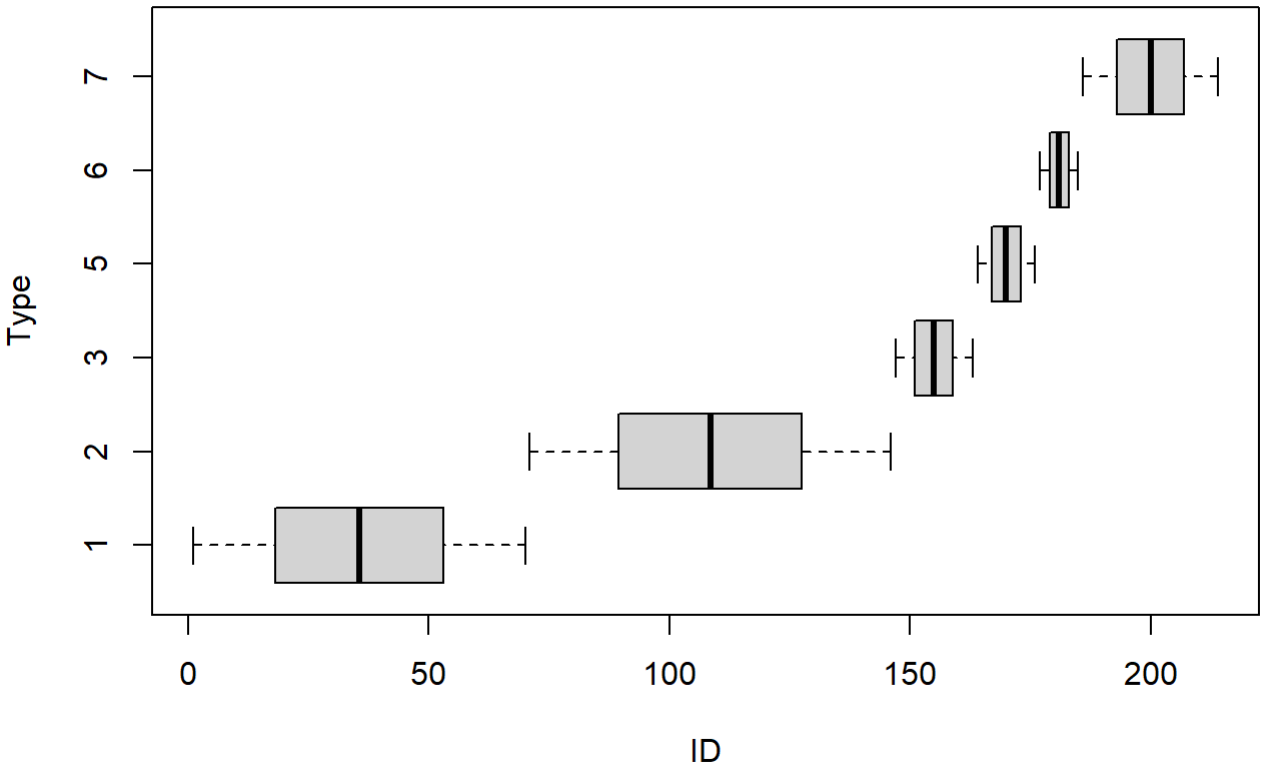
Drew Murray

10/1/2022

```
glass <- read.csv("C:/Users/dgmur/Downloads/glass.data", header = F)
View(glass)
colnames(glass) <- c('ID','RI','Na','Mg','Al','Si','K','Ca','Ba','Fe','Type')
```

#A

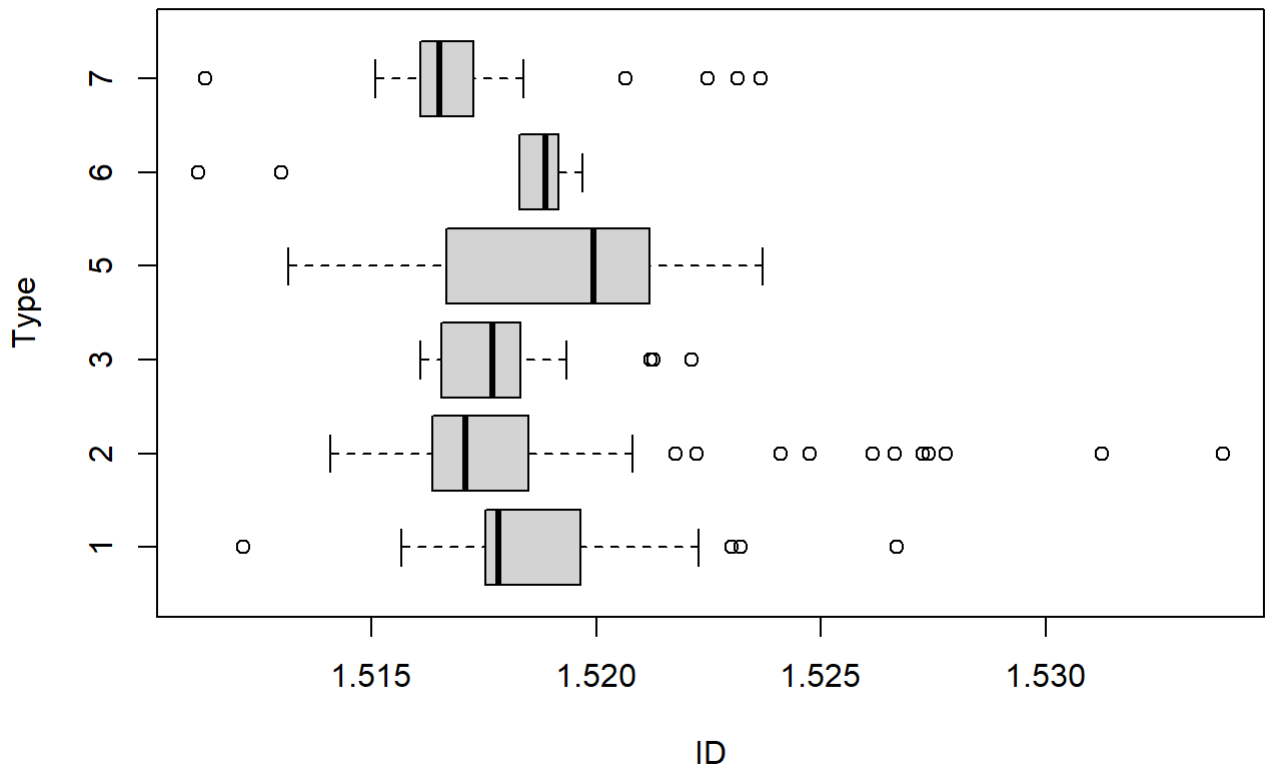
```
a <- boxplot(ID~Type, data = glass,xlab="ID",ylab="Type", horizontal = TRUE)
```




```
a
```

```
## $stats
##      [,1]  [,2] [,3] [,4] [,5] [,6]
## [1,]  1.0  71.0  147  164  177  186
## [2,] 18.0  89.5  151  167  179  193
## [3,] 35.5 108.5  155  170  181  200
## [4,] 53.0 127.5  159  173  183  207
## [5,] 70.0 146.0  163  176  185  214
##
## $n
## [1] 70 76 17 13  9 29
##
## $conf
##          [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
## [1,] 28.89039 101.6129 151.9343 167.3707 178.8933 195.8924
## [2,] 42.10961 115.3871 158.0657 172.6293 183.1067 204.1076
##
## $out
## numeric(0)
##
## $group
## numeric(0)
##
## $names
## [1] "1" "2" "3" "5" "6" "7"
```

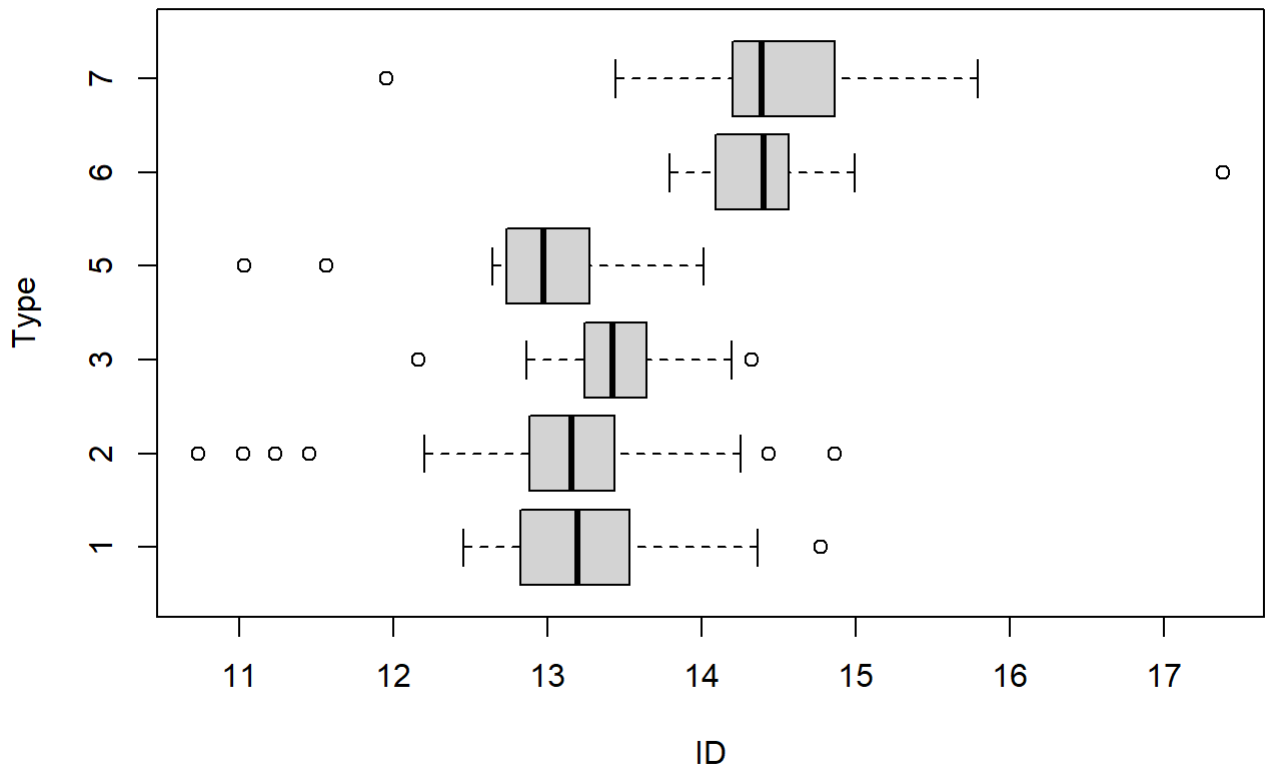
```
b<-boxplot(RI~Type, data = glass,xlab="ID",ylab="Type", horizontal = TRUE)
```



b

```
## $stats
##          [,1]     [,2]    [,3]    [,4]    [,5]    [,6]
## [1,] 1.515670 1.514090 1.51610 1.51316 1.51829 1.51508
## [2,] 1.517540 1.516355 1.51655 1.51666 1.51829 1.51609
## [3,] 1.517835 1.517100 1.51769 1.51994 1.51888 1.51651
## [4,] 1.519660 1.518495 1.51832 1.52119 1.51916 1.51727
## [5,] 1.522270 1.520810 1.51934 1.52369 1.51969 1.51838
##
## $n
## [1] 70 76 17 13  9 29
##
## $conf
##          [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
## [1,] 1.517435 1.516712 1.517012 1.517955 1.518422 1.516164
## [2,] 1.518235 1.517488 1.518368 1.521925 1.519338 1.516856
##
## $out
##  [1] 1.52667 1.52320 1.51215 1.52300 1.52725 1.52410 1.52475 1.53125 1.53393
## [10] 1.52222 1.52664 1.52739 1.52777 1.52177 1.52177 1.52614 1.52127 1.52121
## [19] 1.52211 1.51299 1.51115 1.51131 1.52315 1.52247 1.52365 1.52065
##
## $group
##  [1] 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 3 3 3 5 5 6 6 6 6 6
##
## $names
## [1] "1" "2" "3" "5" "6" "7"
```

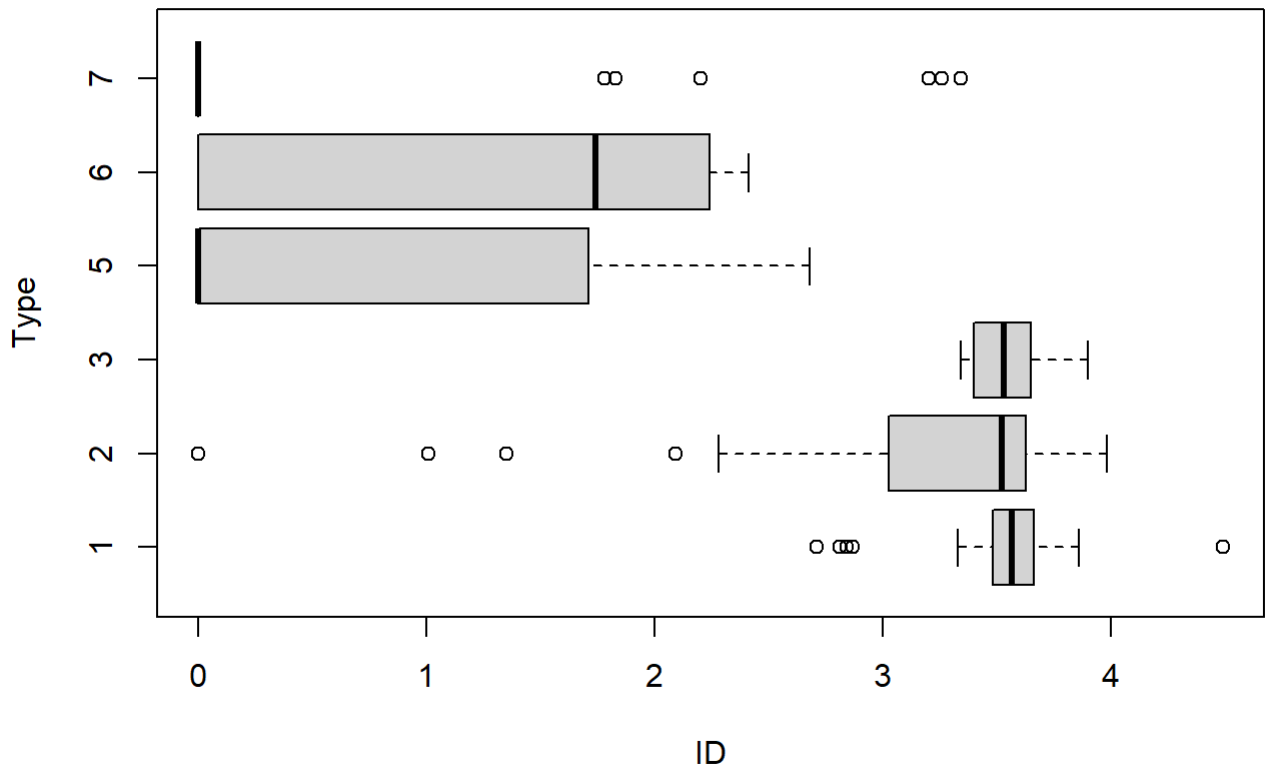
```
c<-boxplot(Na~Type, data = glass,xlab="ID",ylab="Type", horizontal = TRUE)
```




```
c
```

```
## $stats
##        [,1]   [,2]  [,3]  [,4]  [,5]  [,6]
## [1,] 12.450 12.200 12.86 12.64 13.79 13.44
## [2,] 12.820 12.880 13.24 12.73 14.09 14.20
## [3,] 13.195 13.155 13.42 12.97 14.40 14.39
## [4,] 13.530 13.435 13.64 13.27 14.56 14.86
## [5,] 14.360 14.250 14.19 14.01 14.99 15.79
##
## $n
## [1] 70 76 17 13  9 29
##
## $conf
##          [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
## [1,] 13.06092 13.05441 13.26672 12.73336 14.15247 14.19636
## [2,] 13.32908 13.25559 13.57328 13.20664 14.64753 14.58364
##
## $out
##  [1] 14.77 14.86 11.45 10.73 14.43 11.23 11.02 12.16 14.32 11.56 11.03 17.38
## [13] 11.95
##
## $group
##  [1] 1 2 2 2 2 2 2 3 3 4 4 5 6
##
## $names
## [1] "1" "2" "3" "5" "6" "7"
```

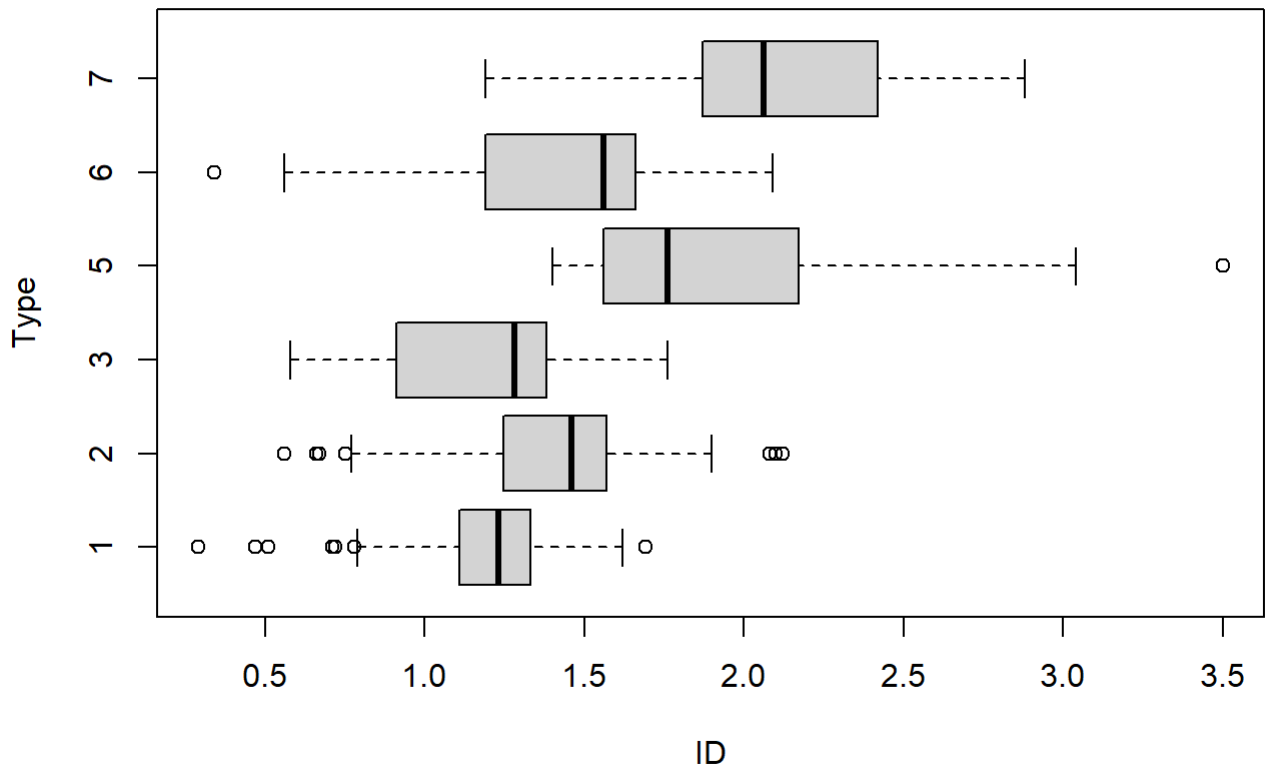
```
d<-boxplot(Mg~Type, data = glass,xlab="ID",ylab="Type", horizontal = TRUE)
```



d

```
## $stats
##       [,1]  [,2] [,3] [,4] [,5] [,6]
## [1,] 3.330 2.280 3.34 0.00 0.00    0
## [2,] 3.480 3.025 3.40 0.00 0.00    0
## [3,] 3.565 3.520 3.53 0.00 1.74    0
## [4,] 3.660 3.625 3.65 1.71 2.24    0
## [5,] 3.860 3.980 3.90 2.68 2.41    0
##
## $n
## [1] 70 76 17 13  9 29
##
## $conf
##          [,1]     [,2]     [,3]       [,4]      [,5] [,6]
## [1,] 3.531008 3.411257 3.434198 -0.7493445 0.5602667    0
## [2,] 3.598992 3.628743 3.625802  0.7493445 2.9197333    0
##
## $out
##  [1] 4.49 2.87 2.84 2.81 2.71 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 2.09 1.35
## [16] 1.01 0.00 3.20 3.26 3.34 2.20 1.83 1.78
##
## $group
##  [1] 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 6 6 6 6 6 6
##
## $names
## [1] "1" "2" "3" "5" "6" "7"
```

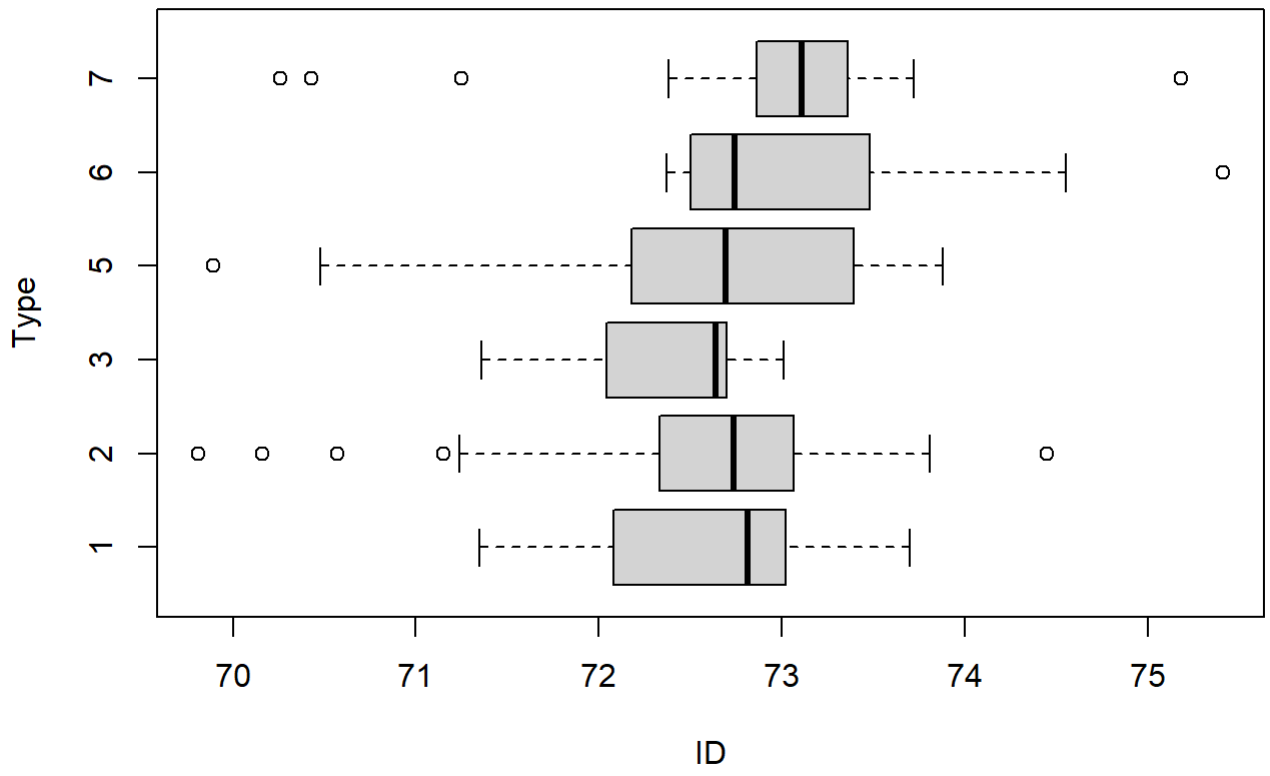
```
e<-boxplot(Al~Type, data = glass,xlab="ID",ylab="Type", horizontal = TRUE)
```




```
e
```

```
## $stats
##      [,1]  [,2] [,3] [,4] [,5] [,6]
## [1,] 0.79 0.770 0.58 1.40 0.56 1.19
## [2,] 1.11 1.245 0.91 1.56 1.19 1.87
## [3,] 1.23 1.460 1.28 1.76 1.56 2.06
## [4,] 1.33 1.570 1.38 2.17 1.66 2.42
## [5,] 1.62 1.900 1.76 3.04 2.09 2.88
##
## $n
## [1] 70 76 17 13  9 29
##
## $conf
##          [,1]     [,2]     [,3]    [,4]     [,5]     [,6]
## [1,] 1.188454 1.401098 1.099893 1.49269 1.312467 1.898631
## [2,] 1.271546 1.518902 1.460107 2.02731 1.807533 2.221369
##
## $out
##  [1] 1.69 0.29 0.47 0.47 0.72 0.71 0.51 0.78 2.12 2.08 0.66 2.10 0.56 0.75 0.67
## [16] 3.50 0.34
##
## $group
##  [1] 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 4 5
##
## $names
## [1] "1" "2" "3" "5" "6" "7"
```

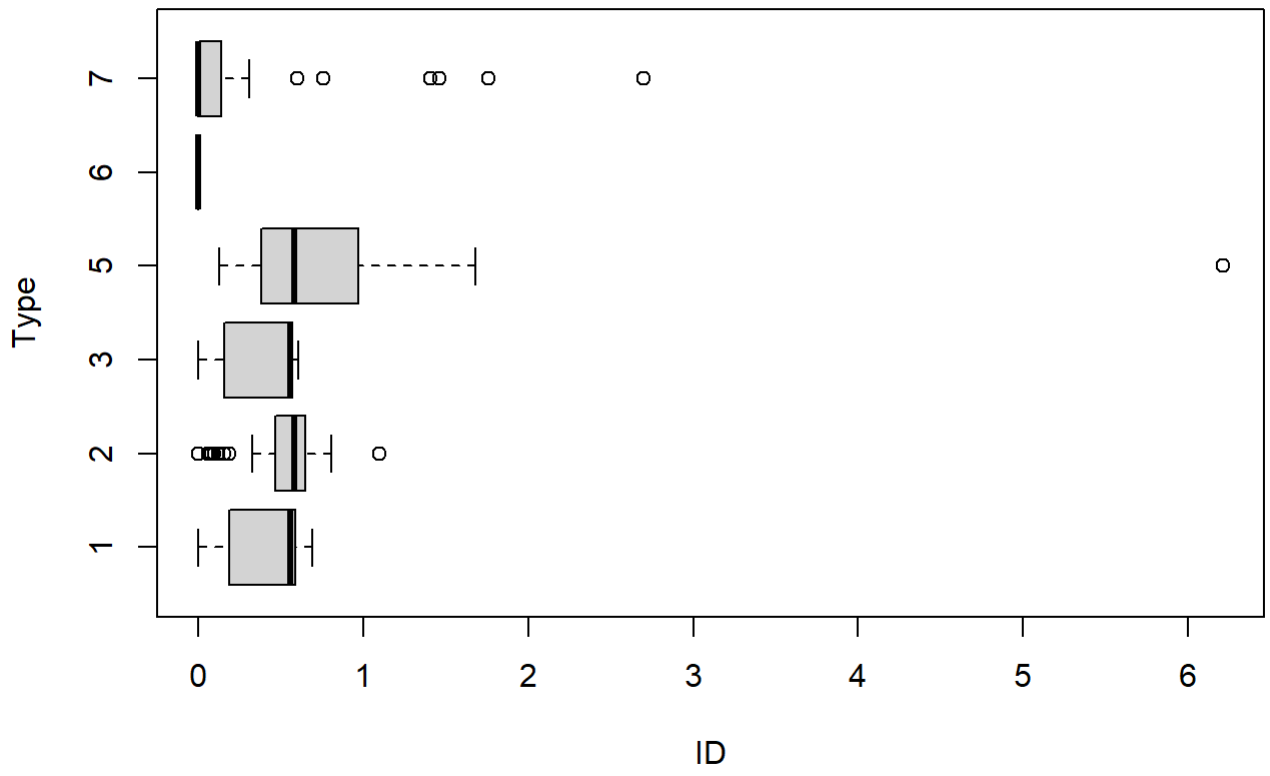
```
f<-boxplot(Si~Type, data = glass,xlab="ID",ylab="Type", horizontal = TRUE)
```




```
f
```

```
## $stats
##        [,1]   [,2]  [,3]  [,4]  [,5]  [,6]
## [1,] 71.350 71.240 71.36 70.48 72.37 72.38
## [2,] 72.080 72.330 72.04 72.18 72.50 72.86
## [3,] 72.815 72.735 72.64 72.69 72.74 73.11
## [4,] 73.020 73.065 72.70 73.39 73.48 73.36
## [5,] 73.700 73.810 73.01 73.88 74.55 73.72
##
## $n
## [1] 70 76 17 13  9 29
##
## $conf
##          [,1]     [,2]     [,3]     [,4]     [,5]    [,6]
## [1,] 72.63748 72.60179 72.38708 72.15976 72.22387 72.9633
## [2,] 72.99252 72.86821 72.89292 73.22024 73.25613 73.2567
##
## $out
##  [1] 70.57 71.15 69.81 70.16 74.45 69.89 75.41 71.25 70.26 70.43 75.18
##
## $group
##  [1] 2 2 2 2 2 4 5 6 6 6 6
##
## $names
## [1] "1" "2" "3" "5" "6" "7"
```

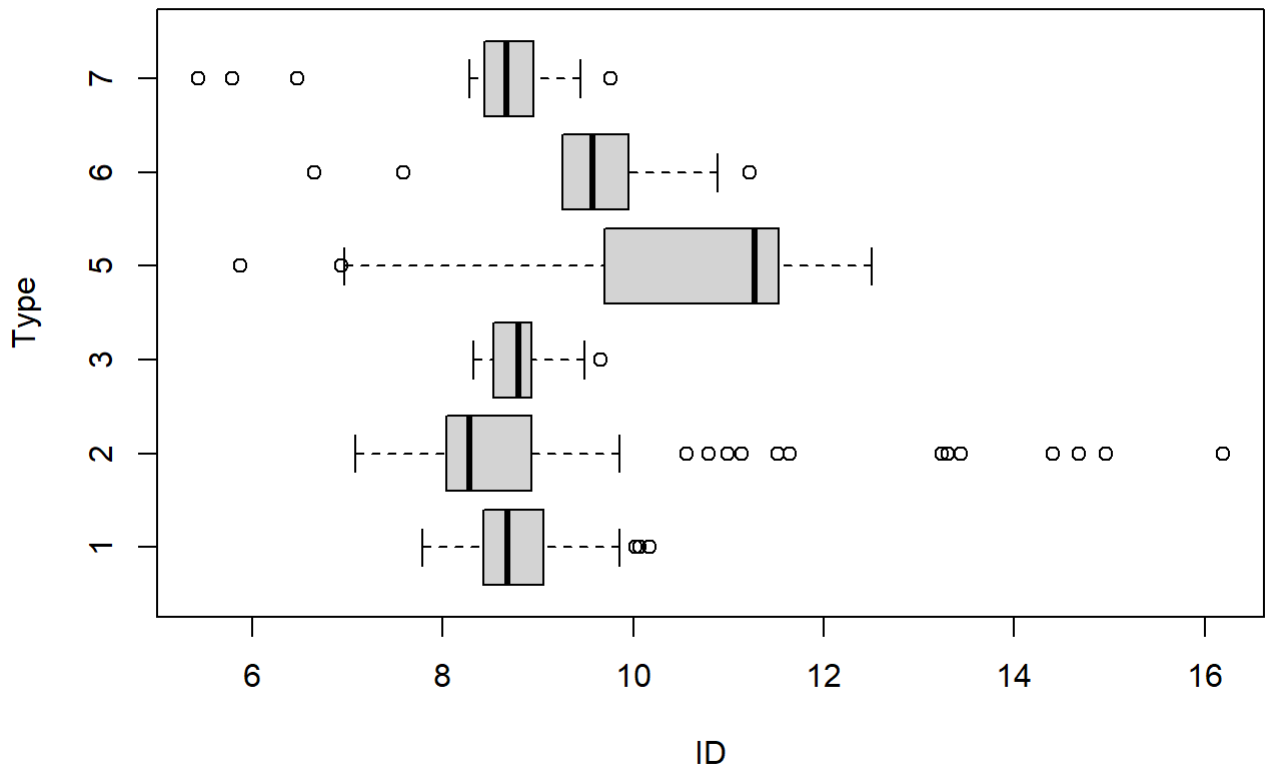
```
g<-boxplot(K~Type, data = glass,xlab="ID",ylab="Type", horizontal = TRUE)
```




```
g
```

```
## $stats
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,] 0.00 0.33 0.00 0.13    0 0.00
## [2,] 0.19 0.47 0.16 0.38    0 0.00
## [3,] 0.56 0.58 0.56 0.58    0 0.00
## [4,] 0.59 0.65 0.57 0.97    0 0.14
## [5,] 0.69 0.81 0.61 1.68    0 0.31
##
## $n
## [1] 70 76 17 13  9 29
##
## $conf
##           [,1]      [,2]      [,3]      [,4] [,5]        [,6]
## [1,] 0.4844616 0.5473771 0.4028854 0.3214542    0 -0.04107581
## [2,] 0.6355384 0.6126229 0.7171146 0.8385458    0  0.04107581
##
## $out
##  [1] 0.16 1.10 0.07 0.08 0.08 0.12 0.10 0.00 0.00 0.00 0.06 0.19 6.21 6.21 1.76
## [16] 1.46 0.60 0.76 2.70 1.41
##
## $group
##  [1] 2 2 2 2 2 2 2 2 2 2 2 4 4 6 6 6 6 6 6
##
## $names
## [1] "1" "2" "3" "5" "6" "7"
```

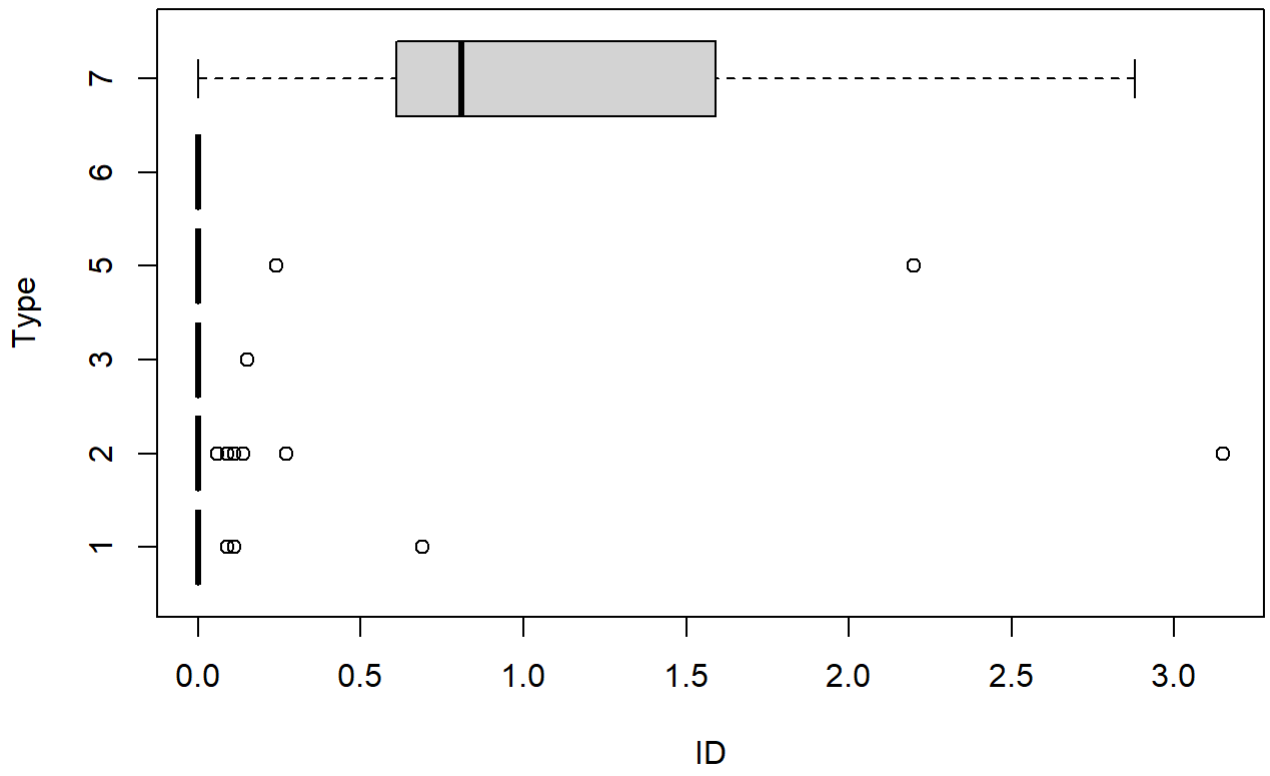
```
h<-boxplot(Ca~Type, data = glass,xlab="ID",ylab="Type", horizontal = TRUE)
```



h

```
## $stats
##       [,1]  [,2] [,3]  [,4]  [,5] [,6]
## [1,] 7.780 7.080 8.32  6.96  9.26 8.28
## [2,] 8.430 8.035 8.53  9.70  9.26 8.44
## [3,] 8.675 8.275 8.79 11.27  9.57 8.67
## [4,] 9.060 8.930 8.93 11.53  9.95 8.95
## [5,] 9.850 9.850 9.49 12.50 10.88 9.45
##
## $n
## [1] 70 76 17 13  9 29
##
## $conf
##          [,1]     [,2]     [,3]     [,4]   [,5]     [,6]
## [1,] 8.556027 8.112792 8.636717 10.46807 9.2066 8.520367
## [2,] 8.793973 8.437208 8.943283 12.07193 9.9334 8.819633
##
## $out
##  [1] 10.02 10.06 10.17 11.64 10.79 13.24 13.30 16.19 11.52 10.99 14.68 14.96
## [13] 14.40 10.56 11.14 13.44  9.65  5.87  6.93  7.59 11.22  6.65  5.43  5.79
## [25]  9.76  6.47
##
## $group
##  [1] 1 1 1 2 2 2 2 2 2 2 2 2 2 2 3 4 4 5 5 5 6 6 6 6
##
## $names
## [1] "1" "2" "3" "5" "6" "7"
```

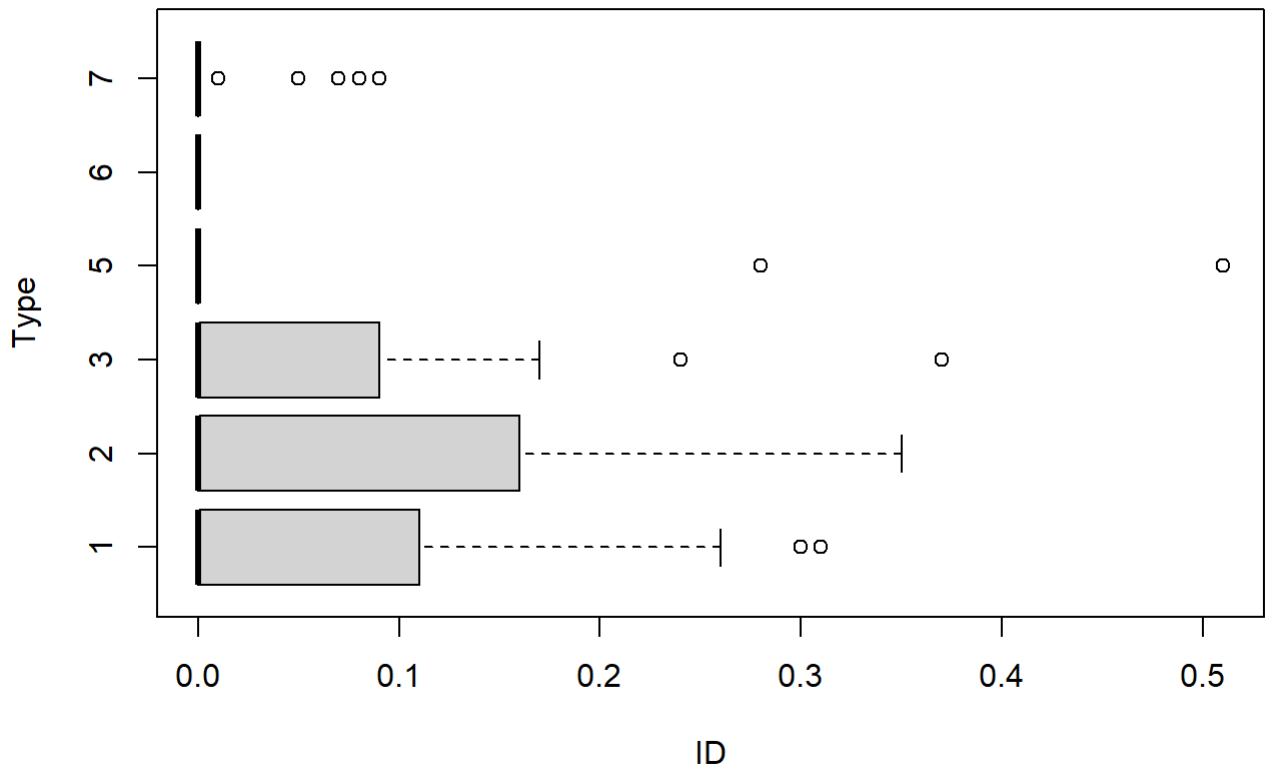
```
i<-boxplot(Ba~Type, data = glass,xlab="ID",ylab="Type", horizontal = TRUE)
```




```
i
```

```
## $stats
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    0    0    0    0    0 0.00
## [2,]    0    0    0    0    0 0.61
## [3,]    0    0    0    0    0 0.81
## [4,]    0    0    0    0    0 1.59
## [5,]    0    0    0    0    0 2.88
##
## $n
## [1] 70 76 17 13  9 29
##
## $conf
##      [,1] [,2] [,3] [,4] [,5]      [,6]
## [1,]    0    0    0    0    0 0.5224693
## [2,]    0    0    0    0    0 1.0975307
##
## $out
##  [1] 0.09 0.11 0.69 0.14 0.11 3.15 0.27 0.09 0.06 0.15 2.20 0.24
##
## $group
##  [1] 1 1 1 2 2 2 2 2 2 3 4 4
##
## $names
## [1] "1" "2" "3" "5" "6" "7"
```

```
j<-boxplot(Fe~Type, data = glass,xlab="ID",ylab="Type", horizontal = TRUE)
```



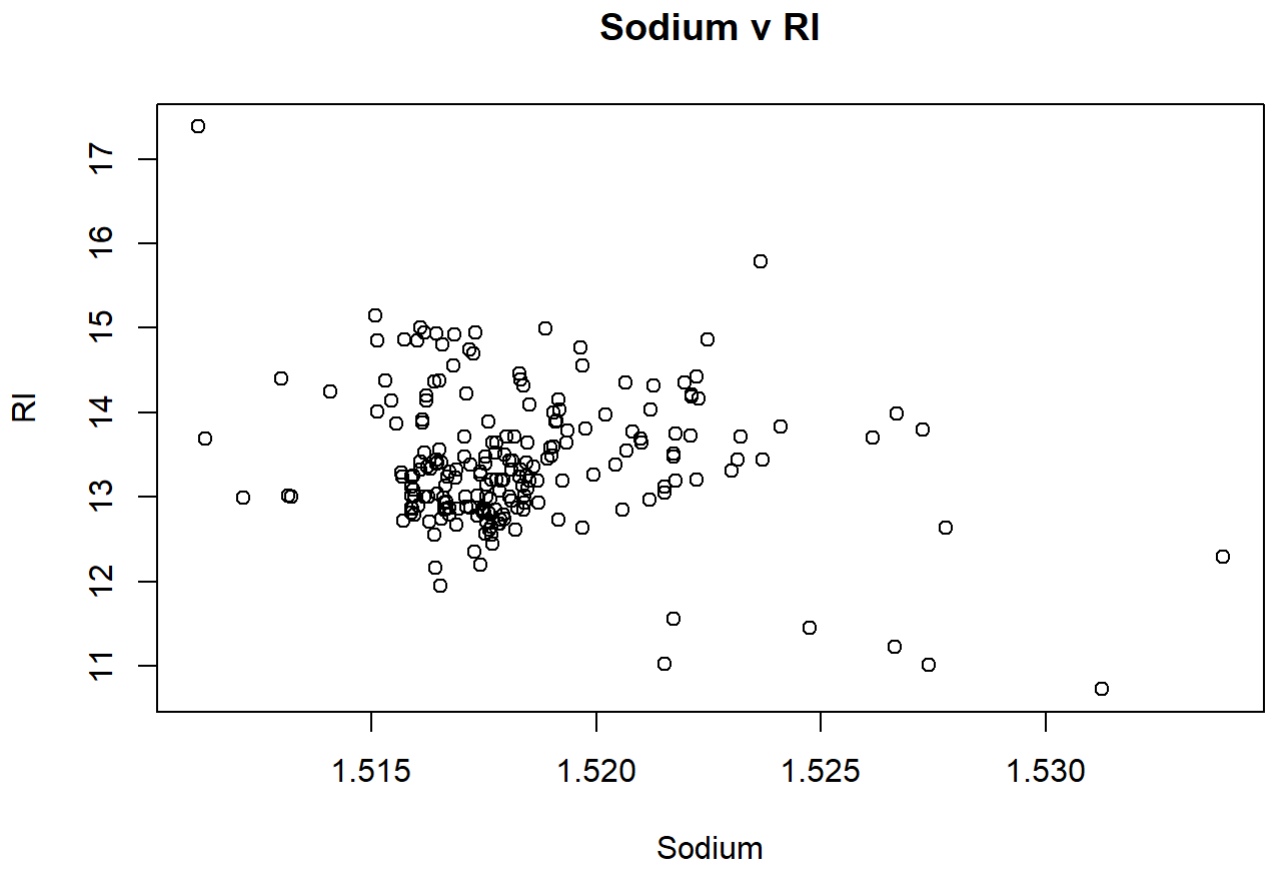

```
j
```

```
## $stats
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,] 0.00 0.00 0.00    0    0    0
## [2,] 0.00 0.00 0.00    0    0    0
## [3,] 0.00 0.00 0.00    0    0    0
## [4,] 0.11 0.16 0.09    0    0    0
## [5,] 0.26 0.35 0.17    0    0    0
##
## $n
## [1] 70 76 17 13  9 29
##
## $conf
##             [,1]        [,2]        [,3] [,4] [,5] [,6]
## [1,] -0.02077307 -0.02899815 -0.03448857    0    0    0
## [2,]  0.02077307  0.02899815  0.03448857    0    0    0
##
## $out
##  [1] 0.30 0.31 0.24 0.37 0.51 0.28 0.09 0.09 0.08 0.07 0.05 0.01
##
## $group
##  [1] 1 1 3 3 4 4 6 6 6 6 6 6
##
## $names
## [1] "1" "2" "3" "5" "6" "7"
```

Na, RI,Si,Ca,Mg

#B

```
par(mfrow=c(1,1))
k<-plot(Na~RI, data=glass, xlab="Sodium", ylab = "RI", main="Sodium v RI")
```



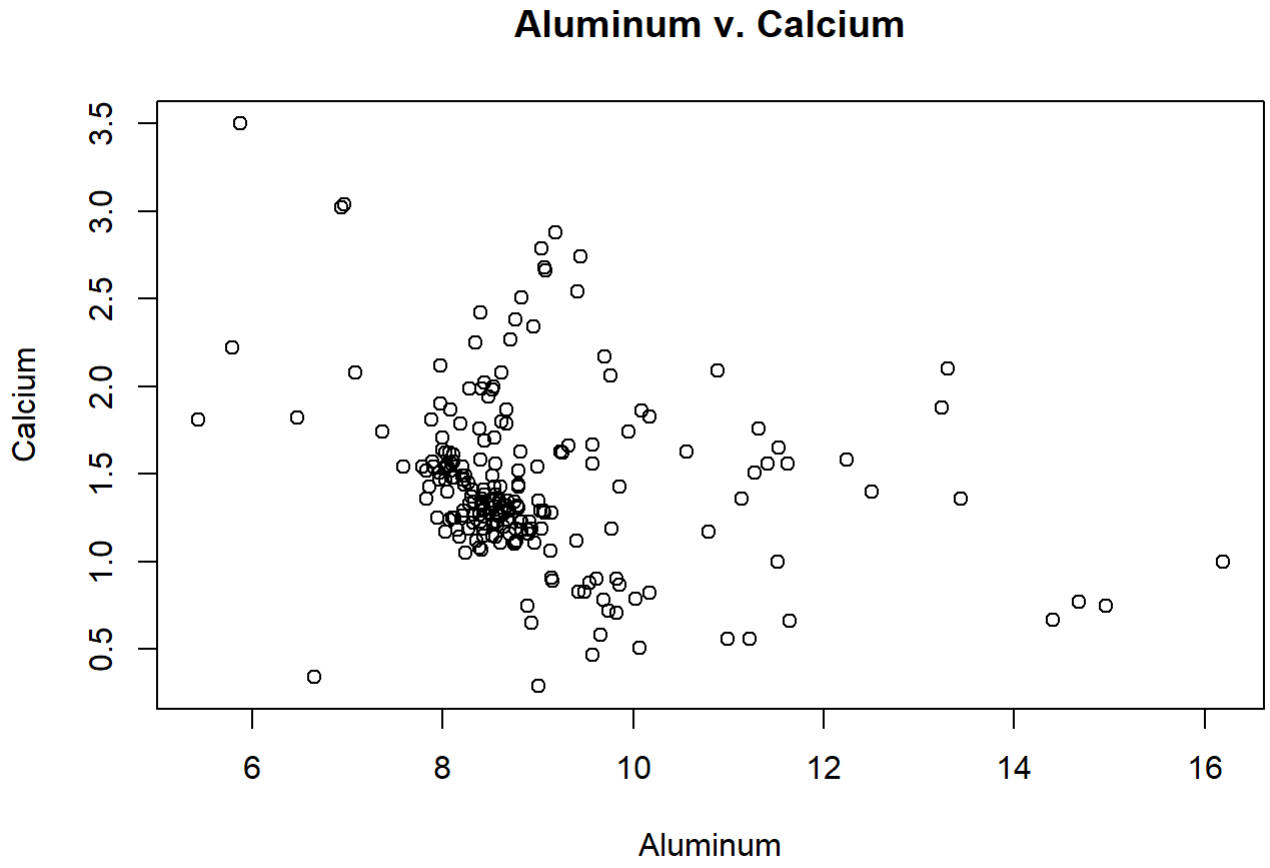

```
k
```

```
## NULL
```

No, many of the observations are closed together making it hard to discriminate

#C

```
l<-plot(Al~Ca,data=glass, xlab="Aluminum", ylab="Calcium", main="Aluminum v. Calcium")
```




```
l
```

```
## NULL
```

#D

```
newglass <- c("RI","Na","Al","Ca","Type")
newdata <- glass[newglass]
View(newdata)
library(MASS)
fit.cv <- lda(Type~., data=newdata, CV=TRUE)
t1<- table(newdata$Type,fit.cv$class)
t1
```

```
##
##      1  2  3  5  6  7
##   1 44 26  0  0  0  0
##   2 20 47  0  4  2  3
##   3  7 10  0  0  0  0
##   5  0  4  0  5  1  3
##   6  0  2  0  0  3  4
##   7  3  3  0  0  0 23
```

```
aper1 <- 92/214
aper1
```

```
## [1] 0.4299065
```

#E

```
fit2 <- qda(Type~., data=newdata, CV=TRUE)
t2<-table(newdata$Type,fit2$class)
t2
```

```
##
##      1  2  3  5  6  7
##   1 56  5  8  0  0  1
##   2 47 17  1  3  5  3
##   3 11  3  3  0  0  0
##   5  0  5  0  8  0  0
##   6  0  5  0  0  0  4
##   7  2  2  0  0  0 25
```

```
aper2 <- 105/214
aper2
```

```
## [1] 0.4906542
```

linear discriminant analysis is more effective

#F

```
set.seed(12)

train<- sample(1:214,107)

fit.train <- lda(Type~., data = newdata, subset = train, CV=TRUE)

t3<-table(newdata$Type[-train],fit.train$class)
t3
```

```
##
##      1  2  3  5  6  7
##   1 10 14  1  5  2  6
##   2 12 18  0  1  0  8
##   3  3  3  0  1  1  0
##   5  1  1  0  1  0  0
##   6  1  3  0  0  0  0
##   7  3  8  0  2  0  2
```

```
aper3<- 72/107
aper3
```

```
## [1] 0.6728972
```

#G

```
fit3 <- lda(Type~., data = glass, CV=TRUE)
t4<-table(glass$Type,fit3$class)
t4
```

```
##
##      1  2  3  5  6  7
##   1 69  1  0  0  0  0
##   2  1 71  3  1  0  0
##   3  0  2 15  0  0  0
##   5  0  0  2 10  0  1
##   6  0  0  1  0  7  1
##   7  0  0  2  2  0 25
```

```
aper4 <- 17/214
aper4
```

```
## [1] 0.07943925
```

Using all the numerical variables was the most effective because it had the lowest aer

#H

```
newdata2 <- subset(glass, Type < 4,)
```