Discriminant Analysis Homework – Alzheim Data

Discriminant Analysis Homework – Alzheim Data Nathan Kim, LJ Flores, Megan Zhang, Jinny Choi

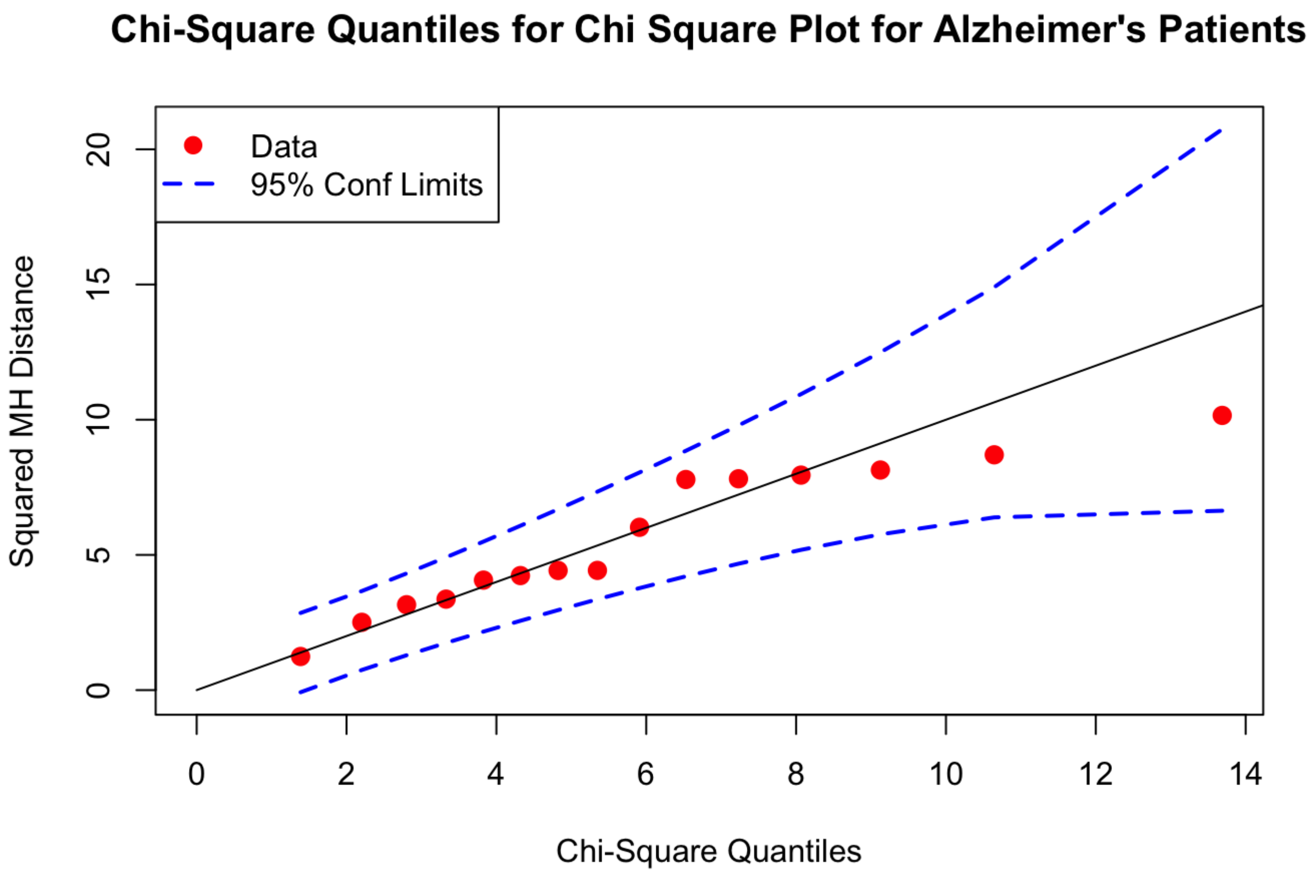
```
library(MASS)
library(DiscriMiner)
library(klaR)
```

```
alzheim <- read.csv("alzheim.csv", as.is = TRUE)
alzheim <- alzheim[,c(1,3,4,6,7,8,9,10)]
for (i in 2:7){
  alzheim[,i] <- (alzheim[,i]-mean(alzheim[,i]))/sqrt(var(alzheim[,i]))
}
alzheim$groupFactor <- as.factor(alzheim$group)
```

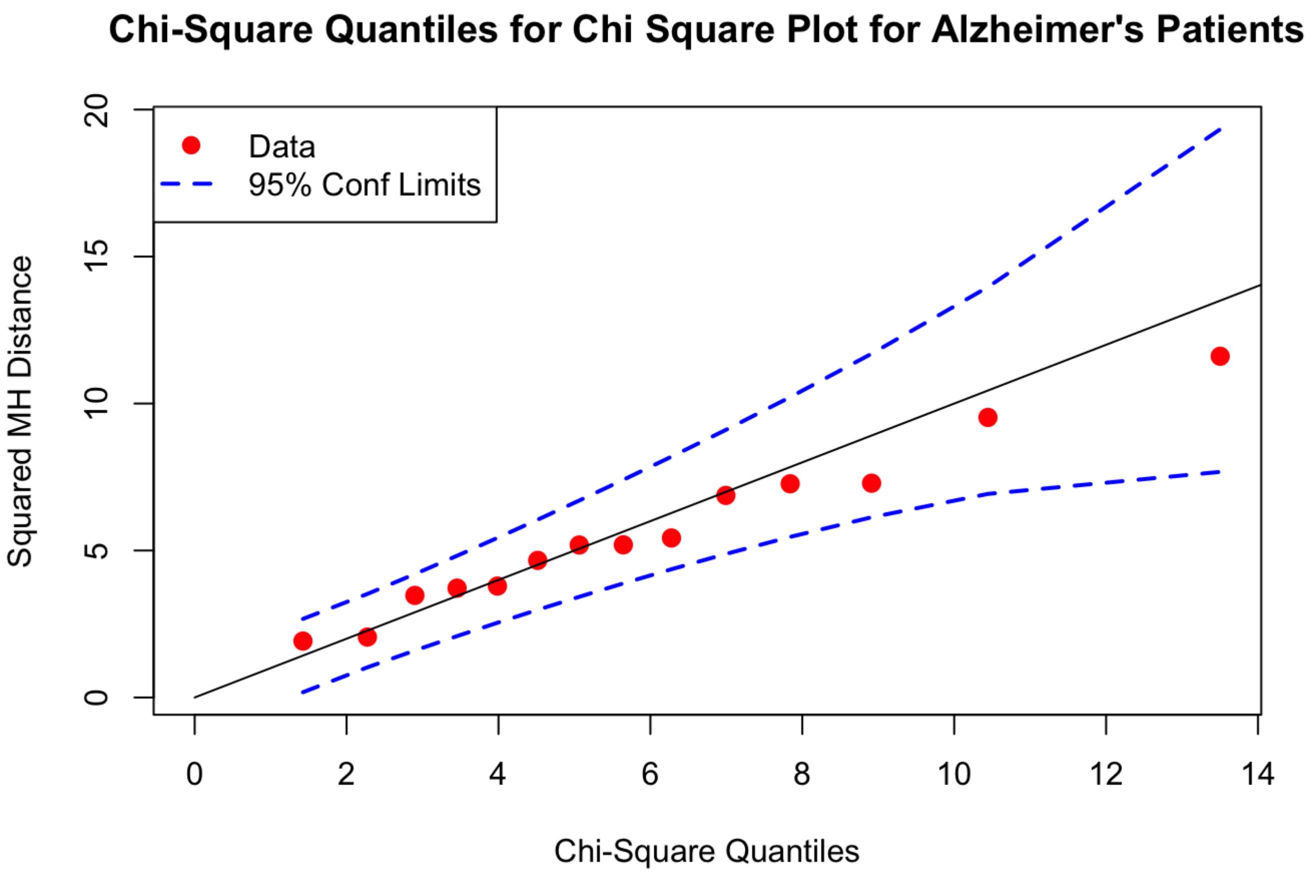
1. Evaluate the assumptions implicit to Discriminant Analysis for your data – multivariate normality WITHIN each group (i.e. chi-square quantile plots) and similarity of covariances matrices (look at Box's M or just look at raw standard deviations/covariance matrices). Comment on what you find. Comment on whether you think transformations might help your data to meet the assumptions of DA. If you think they might, make some transformations and find out! You might also want to make a matrix plot (or a pairs plot) to get a sense of what your data looks like two variables at a time (use different symbols for each group).

Checking for multivariate normality using Chi-square plots shows that all three groups have a multivariate normal distribution.

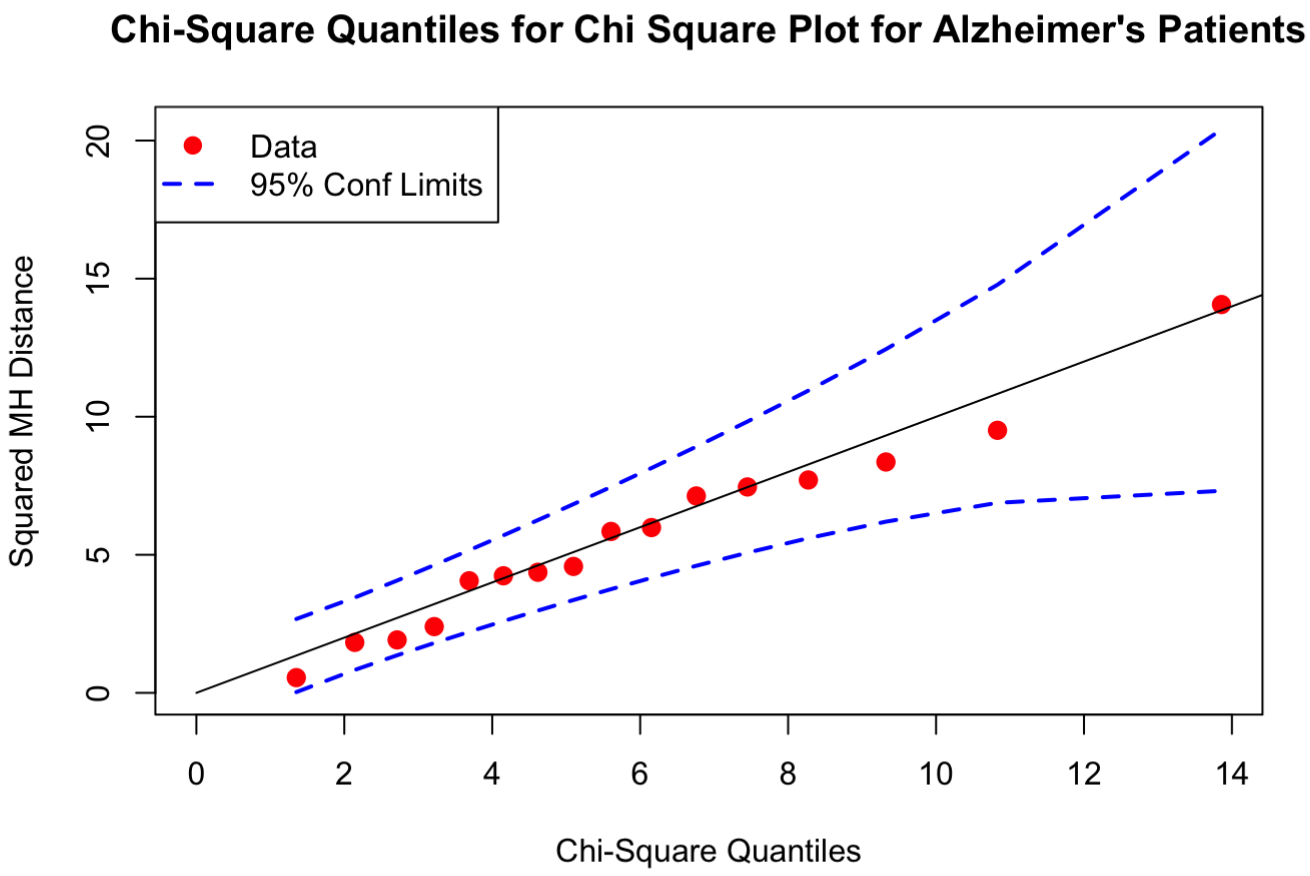
```
source("http://www.reuningscherer.net/STAT660/R/CSQPlot.r.txt")
CSQPlot(alzheim[alzheim$group == 1, c("hirecall", "lirecall", "hiunrem", "liunrem", "sto
re", "recog")], label = "Chi Square Plot for Alzheimer's Patients")
```



```
CSQPlot(alzheim[alzheim$group == 2, c("hirecall", "lirecall", "hiunrem", "liunrem", "sto
re", "recog")], label = "Chi Square Plot for Alzheimer's Patients")
```

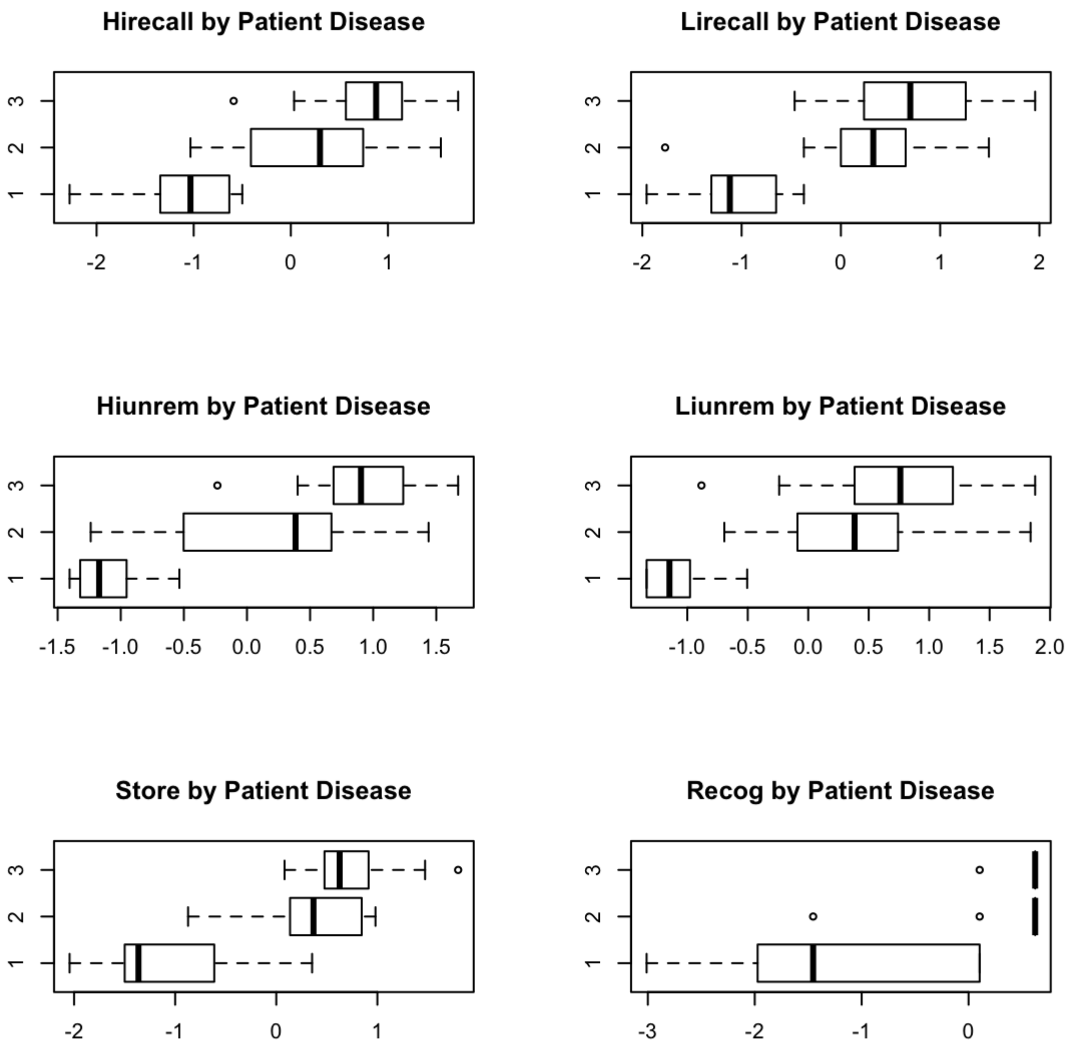


```
CSQPlot(alzheim[alzheim$group == 3, c("hirecall", "lirecall", "hiunrem", "liunrem", "sto
re", "recog")], label = "Chi Square Plot for Alzheimer's Patients")
```



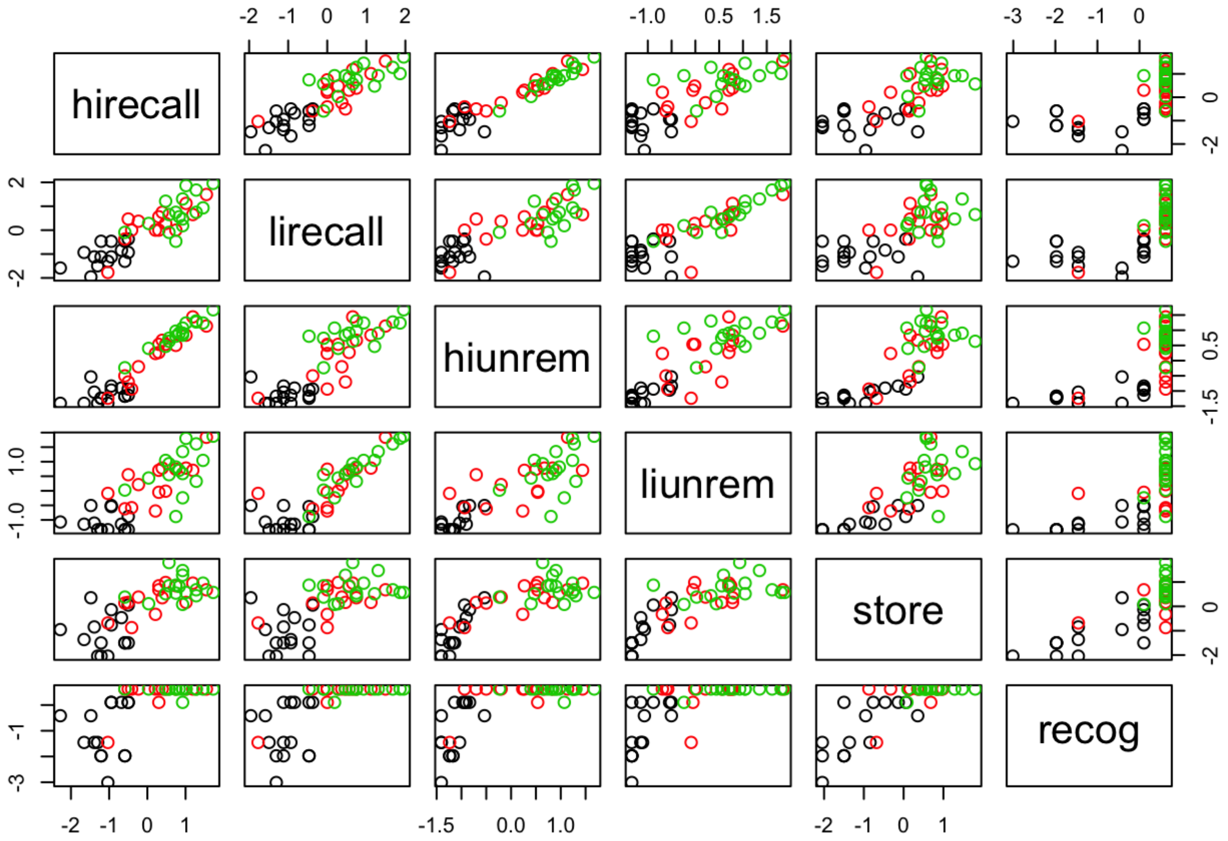
Boxplots of data divided by the different variables show that there is significant overlap within each variable for the Depressed and Control groups, whereas there is clear separation between these groups from Alzheimer’s patients.

```
par(mfrow=c(3,2))
boxplot(hirecall ~ group, data = alzheim, horizontal = T, main = "Hirecall by Patient Di
sease")
boxplot(lirecall ~ group, data = alzheim, horizontal = T, main = "Lirecall by Patient Di
sease")
boxplot(hiunrem ~ group, data = alzheim, horizontal = T, main = "Hiunrem by Patient Dise
ase")
boxplot(liunrem ~ group, data = alzheim, horizontal = T, main = "Liunrem by Patient Dise
ase")
boxplot(store ~ group, data = alzheim, horizontal = T, main = "Store by Patient Disease"
)
boxplot(recog ~ group, data = alzheim, horizontal = T, main = "Recog by Patient Disease"
)
```



The matrix plot of test variables plotted pairwise and colored by group shows similar results, with depression and control groups occupying more-or-less the same score ranges while the group for Alzheimer's disease shows consistent separation from the other two groups.

```
plot(alzheim[,c("hirecall", "lirecall", "hiunrem", "liunrem", "store", "recog")], col =
 alzheim$group, pch = 1, cex=1.2)
```



Comparing the covariance matrices of both “Alzheimer’s Disease”, “Depression” and “Control” categories shows large differences between these categories.

```
round(cov(alzheim[alzheim$group == 1, 2:7]),2)
```

```
##          hirecall lirecall hiunrem liunrem store recog
## hirecall     0.25     0.13    0.04   -0.02  0.01  0.06
## lirecall     0.13     0.21    0.00   -0.01 -0.03  0.05
## hiunrem      0.04     0.00    0.06    0.06  0.17  0.16
## liunrem     -0.02    -0.01    0.06    0.10  0.21  0.22
## store        0.01    -0.03    0.17    0.21  0.60  0.60
## recog        0.06     0.05    0.16    0.22  0.60  1.09
```

```
round(cov(alzheim[alzheim$group == 2, 2:7]),2)
```

```
##          hirecall lirecall hiunrem liunrem store recog
## hirecall     0.54     0.44    0.56    0.36  0.29  0.20
## lirecall     0.44     0.58    0.44    0.36  0.24  0.33
## hiunrem      0.56     0.44    0.63    0.34  0.35  0.21
## liunrem      0.36     0.36    0.34    0.49  0.24  0.08
## store        0.29     0.24    0.35    0.24  0.35  0.14
## recog        0.20     0.33    0.21    0.08  0.14  0.32
```

```
round(cov(alzheim[alzheim$group == 3, 2:7]),2)
```

```
##          hirecall lirecall hiunrem liunrem store recog
## hirecall     0.30     0.24    0.24    0.19  0.04  0.00
## lirecall     0.24     0.51    0.19    0.49  0.02  0.02
## hiunrem      0.24     0.19    0.20    0.15  0.01 -0.01
## liunrem      0.19     0.49    0.15    0.55  0.07  0.03
## store        0.04     0.02    0.01    0.07  0.22  0.02
## recog        0.00     0.02   -0.01    0.03  0.02  0.02
```

Checking standard deviations of diseases, categorized as “Alzheimer’s Disease”, “Depression” and “Control” also shows that for variables like hiunrem, liunrem, and recog, the largest standard deviation is over twice that of the smallest entry.

```
sumstats <- round(sqrt(aggregate(alzheim[,2:7],by=list(alzheim$group),FUN=var)),2)[,-1]
rownames(sumstats) = c("Alzheimer's", "Depression", "Control")
print("Standard Deviations by Group")
```

```
## [1] "Standard Deviations by Group"
```

```
sumstats
```

	hirecall <dbl>	lirecall <dbl>	hiunrem <dbl>	liunrem <dbl>	store <dbl>	recog <dbl>
Alzheimer's	0.50	0.46	0.25	0.32	0.77	1.04
Depression	0.74	0.76	0.79	0.70	0.59	0.56
Control	0.55	0.72	0.45	0.74	0.46	0.13

3 rows

We conclude that although the data is multivariate normal within each group based on the chi square plots, the covariance matrices are not the same between groups, so quadratic DA will be applied.

2. Perform stepwise discriminant analysis on your data. Comment on which model seems the best. Use quadratic discriminant analysis if appropriate. If you end up with only one significant discriminating variable, you might want to just force a second variable in the model (i.e. add a technically ‘non-significant’ discriminator).

Stepwise discriminant analysis on the data using quadratic discriminant analysis advises to use only hiunrem, whereas linear discriminant analysis says to use only hirecall. We note however that the quadratic analysis had a higher correctness rate, and is appropriate for the data given the covariance matrices are not equal.

```
alzheimStepLDA <- stepclass(group ~ hirecall + lirecall + hiunrem + liunrem + store + re
cog, data = alzheim, method = "lda", direction = 'both', fold = nrow(alzheim))
```

```
##  `stepwise classification', using 45-fold cross-validated correctness rate of method
lda'.
```

```
## 45 observations of 6 variables in 3 classes; direction: both
```

```
## stop criterion: improvement less than 5%.
```

```
## correctness rate: 0.71111;  in: "hirecall";  variables (1): hirecall
## 
##  hr.elapsed min.elapsed sec.elapsed 
##       0.000       0.000       0.878
```

```
alzheimStepQDA <- stepclass(group ~ hirecall + lirecall + hiunrem + liunrem + store + re
cog, data = alzheim, method = "qda", direction = 'both', fold = nrow(alzheim))
```

```
##  `stepwise classification', using 45-fold cross-validated correctness rate of method
qda'.
```

```
## 45 observations of 6 variables in 3 classes; direction: both
```

```
## stop criterion: improvement less than 5%.
```

```
## Warning in cv.rate(vars = c(model, tryvar), data = data, grouping =
## grouping, : error(s) in modeling/prediction step
```

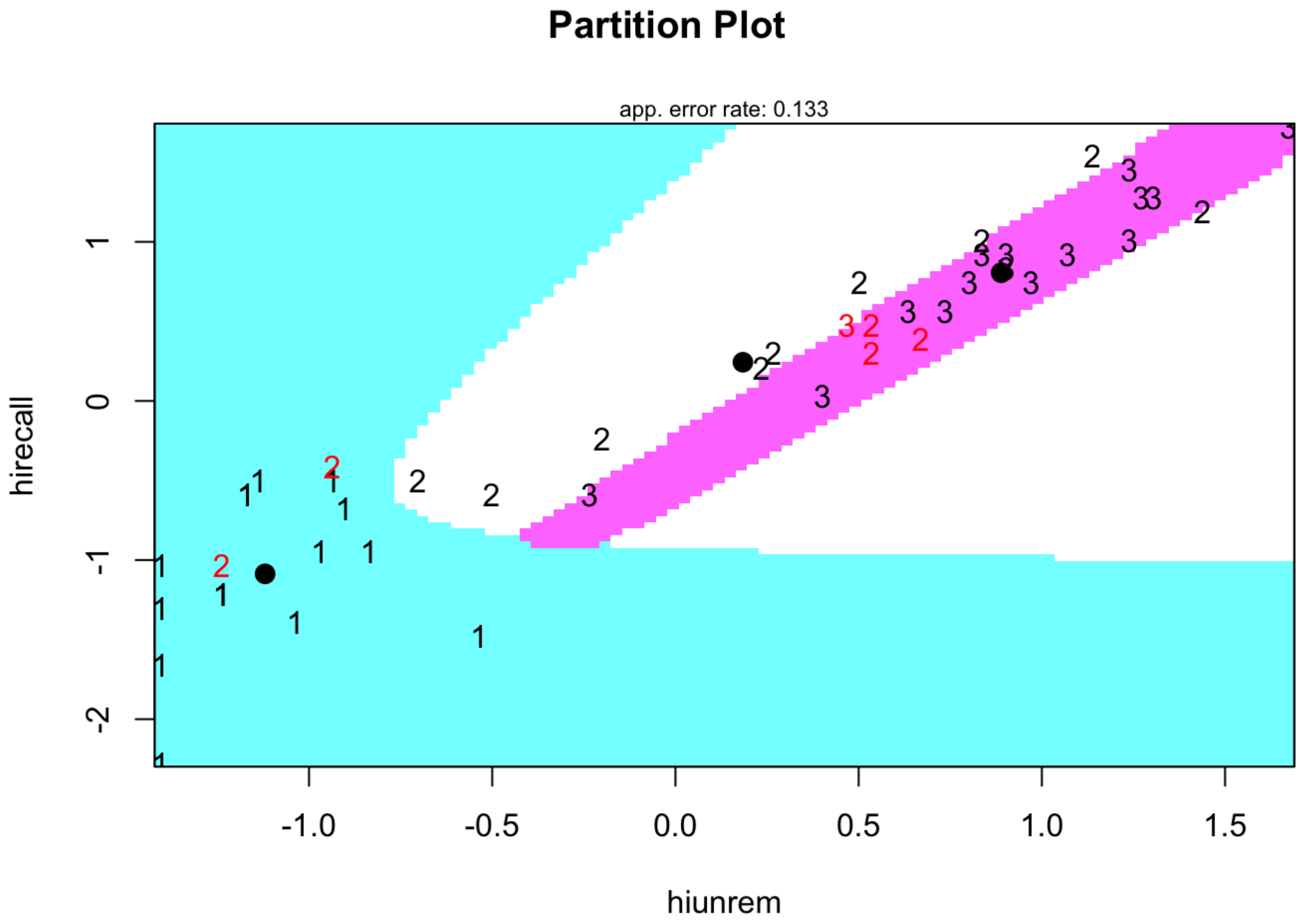
```
## correctness rate: 0.73333;  in: "hiunrem";  variables (1): hiunrem
```

```
## Warning in cv.rate(vars = c(model, tryvar), data = data, grouping =
## grouping, : error(s) in modeling/prediction step
```

```
## 
##  hr.elapsed min.elapsed sec.elapsed 
##        0.00        0.00        0.83
```

We add hirecall onto the model with hiunrem, and generate a partition plot as follows, with 1 for Alzheimer’s, 2 for Depression, and 3 for Control.

```
partimat(groupFactor ~ hirecall + hiunrem, data = alzheim, method = "qda")
```



3. Comment on whether there is statistical evidence that the multivariate group means are different (i.e. Wilks Lambda test).

Here we perform Wilk's Lambda Test

```
alzheimManova <- manova(as.matrix(alzheim[,2:7]) ~ alzheim$group)
summary.manova(alzheimManova, test = "Wilks")
```

```
##                 Df   Wilks approx F num Df den Df    Pr(>F)    
## alzheim$group  1 0.27349   16.824      6     38 2.307e-09 ***
## Residuals     43                                               
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary.aov(alzheimManova)
```

```
##  Response hirecall :
##               Df Sum Sq Mean Sq F value    Pr(>F)
## alzheim$group  1 27.515 27.5145  71.767 1.029e-10 ***
## Residuals     43 16.485  0.3834
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response lirecall :
##               Df Sum Sq Mean Sq F value    Pr(>F)
## alzheim$group  1 24.234 24.2344  52.722 5.401e-09 ***
## Residuals     43 19.766  0.4597
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response hiunrem :
##               Df Sum Sq Mean Sq F value    Pr(>F)
## alzheim$group  1 31.064 31.0640  103.26 5.294e-13 ***
## Residuals     43 12.936  0.3008
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response liunrem :
##               Df Sum Sq Mean Sq F value    Pr(>F)
## alzheim$group  1 25.784 25.7840  60.865 9.071e-10 ***
## Residuals     43 18.216  0.4236
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response store :
##               Df Sum Sq Mean Sq F value    Pr(>F)
## alzheim$group  1 25.645 25.6451  60.079 1.071e-09 ***
## Residuals     43 18.355  0.4269
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response recog :
##               Df Sum Sq Mean Sq F value    Pr(>F)
## alzheim$group  1  20.22  20.220  36.562 3.128e-07 ***
## Residuals     43  23.78   0.553
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The Wilk’s Lambda test shows that across all six variables, there is sufficient evidence to reject the null hypothesis at a significance level of 0.01, and conclude that the multivariate group means are different amongst Alzheimer’s, Depression, and Control cases.

4. How many discriminant functions are significant? What is the relative discriminating power of each function?

Upon running Linear Discriminant Analysis, we see that LDA1 is more significant than LDA2; while this is not a significance test, we observe that 95.7% of the variability was explained by LD1 while 4.7% of the variability was explained by LD2.

```
alzheimLDA <- lda(alzheim$group ~ alzheim$hirecall + alzheim$lirecall + alzheim$hiunrem
+ alzheim$liunrem + alzheim$store + alzheim$recog, prior = c(.33, .33, .34))
alzheimLDA
```

```
## Call:
## lda(alzheim$group ~ alzheim$hirecall + alzheim$lirecall + alzheim$hiunrem +
##     alzheim$liunrem + alzheim$store + alzheim$recog, prior = c(0.33,
##     0.33, 0.34))
##
## Prior probabilities of groups:
##    1    2    3
## 0.33 0.33 0.34
##
## Group means:
##   alzheim$hirecall alzheim$lirecall alzheim$hiunrem alzheim$liunrem
## 1       -1.0866317       -1.0313607      -1.1199193      -1.0844734
## 2        0.2431515        0.2529585       0.1840505       0.3051694
## 3        0.8059597        0.7455619       0.8888802       0.7496706
##   alzheim$store alzheim$recog
## 1    -1.0830832    -1.0384755
## 2     0.3076351     0.4376432
## 3     0.7462098     0.5906329
##
## Coefficients of linear discriminants:
##                         LD1        LD2
## alzheim$hirecall 0.29708915 -1.9382268
## alzheim$lirecall 0.02420258  0.7658663
## alzheim$hiunrem  0.77286983  3.0186925
## alzheim$liunrem  0.41139636 -0.9758841
## alzheim$store    0.32034343  0.1597913
## alzheim$recog    0.37315884 -1.2507103
##
## Proportion of trace:
##    LD1    LD2
## 0.9578 0.0422
```

5. Use classification, both regular and leave-one-out (or cross-validation) to evaluate the discriminating ability of your functions.

Performing Quadratic Discriminant Analysis yields 93% accuracy using regular and 71% using cross-validation.

```
alzheimQDA1 <- qda(alzheim$group ~ alzheim$hirecall + alzheim$lirecall + alzheim$hiunrem
+ alzheim$liunrem + alzheim$store + alzheim$recog, prior = c(.33, .33, .34))
alzheimRaw <- table(alzheim$group, predict(alzheimQDA1)$class)
round(sum(diag(prop.table(alzheimRaw))),2)
```

```
## [1] 0.93
```

```
alzheimQDA2 <- qda(alzheim$group ~ alzheim$hirecall + alzheim$lirecall + alzheim$hiunrem
+ alzheim$liunrem + alzheim$store + alzheim$recog, prior = c(.33, .33, .34), CV = TRUE)
alzheimCV <- table(alzheim$group, alzheimQDA2$class)
round(sum(diag(prop.table(alzheimCV))),2)
```

```
## [1] 0.71
```

6. Provide some evidence as to which of your original variables are the 'best' discriminators amongst your groups (look at standardized discriminant coefficients).

The data was standardized at the beginning of the data analysis, thus these coefficients serve as standardized discriminant coefficients. We see here that hiunrem, hirecall, and recog (in this order, with the first two being variables recommended by previous tests) are the 'best' discriminators.

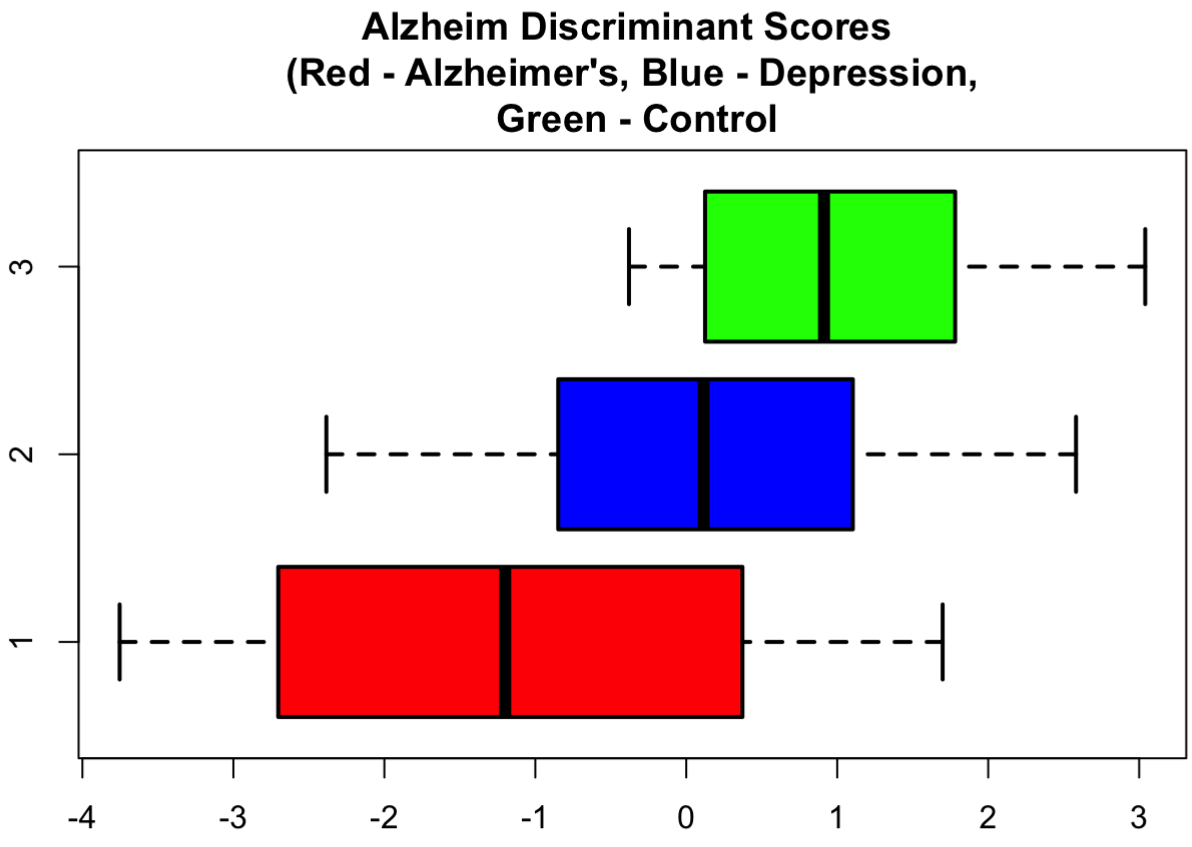
```
alzheimLDA$scaling
```

```
##                         LD1        LD2
## alzheim$hirecall 0.29708915 -1.9382268
## alzheim$lirecall 0.02420258  0.7658663
## alzheim$hiunrem  0.77286983  3.0186925
## alzheim$liunrem  0.41139636 -0.9758841
## alzheim$store    0.32034343  0.1597913
## alzheim$recog    0.37315884 -1.2507103
```

7. Make score plots for the first two or three DA function scores (be sure to use different symbols/colors for each group). Comment on what you see.

Upon performing discriminant analysis, we see better separation between the three groups, especially between the Depression and Control groups, which had high areas of overlap prior to discriminant analysis.

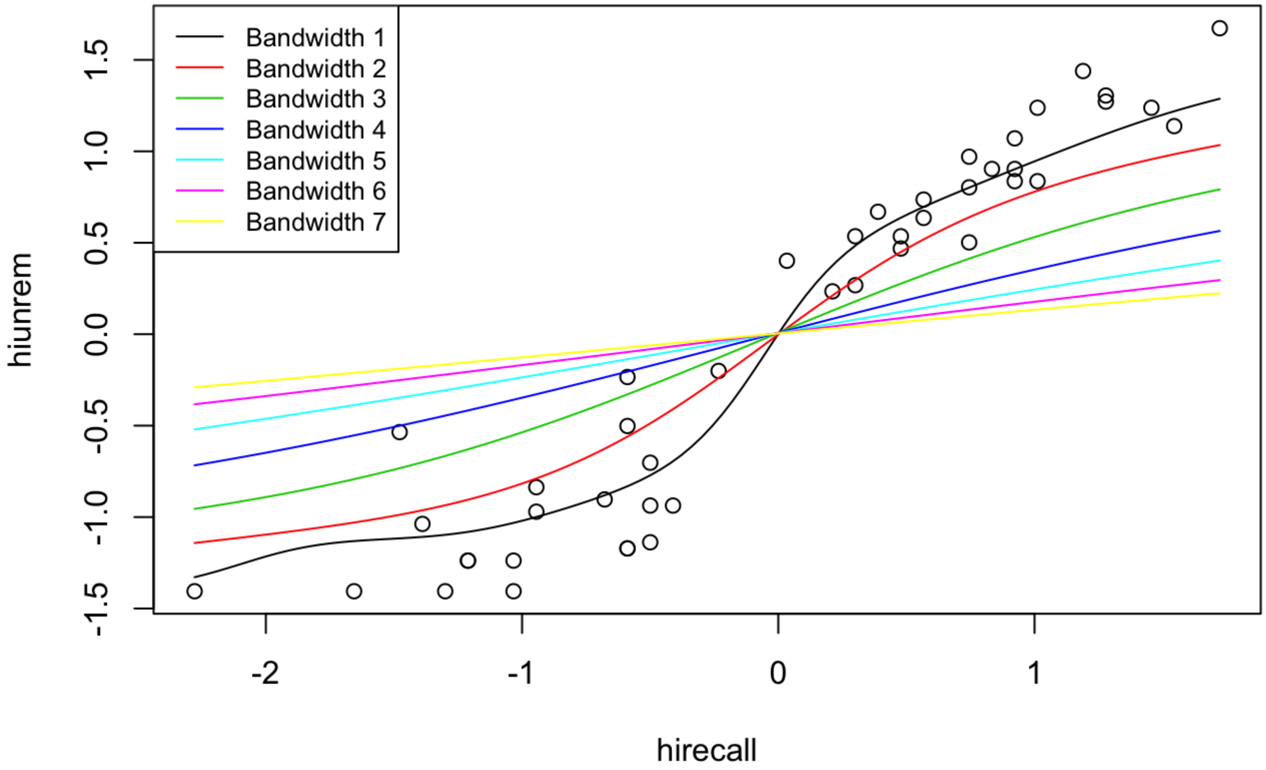
```
scores1 <- as.matrix(alzheim[,c(2:7)])%*%matrix(c(alzheimLDA$scaling), ncol=2)
boxplot(scores1 ~ alzheim$group, lwd=2, col=c("red","blue","green"), horizontal=T, main=
"Alzheim Discriminant Scores \n (Red - Alzheimer's, Blue - Depression, \n Green - Contro
l")
```



8. Bonus (and optional)– try kernel smoothing or k-nearest neighbors and get the admiration of your professor and TA (and some extra credit)! You'll have to use SAS or R for this.

Below is the k-smoothing graph with bandwidths indicated on the top left.

```
with(alzheim,{
  plot(hirecall, hiunrem)
  lines(ksmooth(alzheim$hirecall, alzheim$hiunrem, "normal", bandwidth = 1), col = 1)
  lines(ksmooth(alzheim$hirecall, alzheim$hiunrem, "normal", bandwidth = 2), col = 2)
  lines(ksmooth(alzheim$hirecall, alzheim$hiunrem, "normal", bandwidth = 3), col = 3)
  lines(ksmooth(alzheim$hirecall, alzheim$hiunrem, "normal", bandwidth = 4), col = 4)
  lines(ksmooth(alzheim$hirecall, alzheim$hiunrem, "normal", bandwidth = 5), col = 5)
  lines(ksmooth(alzheim$hirecall, alzheim$hiunrem, "normal", bandwidth = 6), col = 6)
  lines(ksmooth(alzheim$hirecall, alzheim$hiunrem, "normal", bandwidth = 7), col = 7)
  legend("topleft", legend=c("Bandwidth 1", "Bandwidth 2", "Bandwidth 3", "Bandwidth 4",
"Bandwidth 5", "Bandwidth 6", "Bandwidth 7"), col=c(1:7), lty=1, cex=0.8)
})
```


Bandwidth 1
Bandwidth 2
Bandwidth 3
Bandwidth 4
Bandwidth 5
Bandwidth 6
Bandwidth 7
hiunrem
-1.5
-1.0
-0.5
0.0
0.5
1.0
1.5
-2
-1
0
1
hirecall