Stat501_Homework8

Kelby Kies

4/27/2021

Question 1

Part A.) Evaluate if the five groups support multivariate normality distributional assumptions. [10 points]

```
# Read in the data
load('~/Desktop/stat_501/GRB-5groups.rda')
# Test whether the 5 groups follow a MVN

source('~/Desktop/stat_501/testnormality.R')

print(" Cramer Test Statistic")
```

```
## [1] " Cramer Test Statistic"
```

```
group1 <- dplyr::filter(GRB, GRB$class == 1)
group2 <- dplyr::filter(GRB, GRB$class == 2)
group3 <- dplyr::filter(GRB, GRB$class == 3)
group4 <- dplyr::filter(GRB, GRB$class == 4)
group5 <- dplyr::filter(GRB, GRB$class == 5)

#Cramer.test(GRB[,-1])
# group1
print("Class 1 q-value:")
```

```
## [1] "Class 1 q-value:"
```

```
Cramer.test(group1[,-1])
```

```
## [1] 1.133826e-09
```

```
# group2
print("Class 2 q-value:")
```

```
## [1] "Class 2 q-value:"
```

```
Cramer.test(group2[,-1])
```

```
## [1] 5.229242e-06
```

```
# group3
print("Class 3 q-value:")
```

```
## [1] "Class 3 q-value:"
```

```
Cramer.test(group3[,-1])
```

```
## [1] 7.204428e-10
```

```
# group4
print("Class 4 q-value:")
```

```
## [1] "Class 4 q-value:"
```

```
Cramer.test(group4[,-1])
```

```
## [1] 4.072973e-06
```

```
# group5
print("Class 5 q-value:")
```

```
## [1] "Class 5 q-value:"
```

```
Cramer.test(group5[,-1])
```

```
## [1] 2.033481e-05
```

If the q-value returned is less than 0.05, then normality is not good and is rejected. Thus we can not support Multivariate Normality Assumptions and the data should be transformed. The 5 classes would have to be transformed in a similar way which will require some more intense programming. For now we will leave the data how it is.

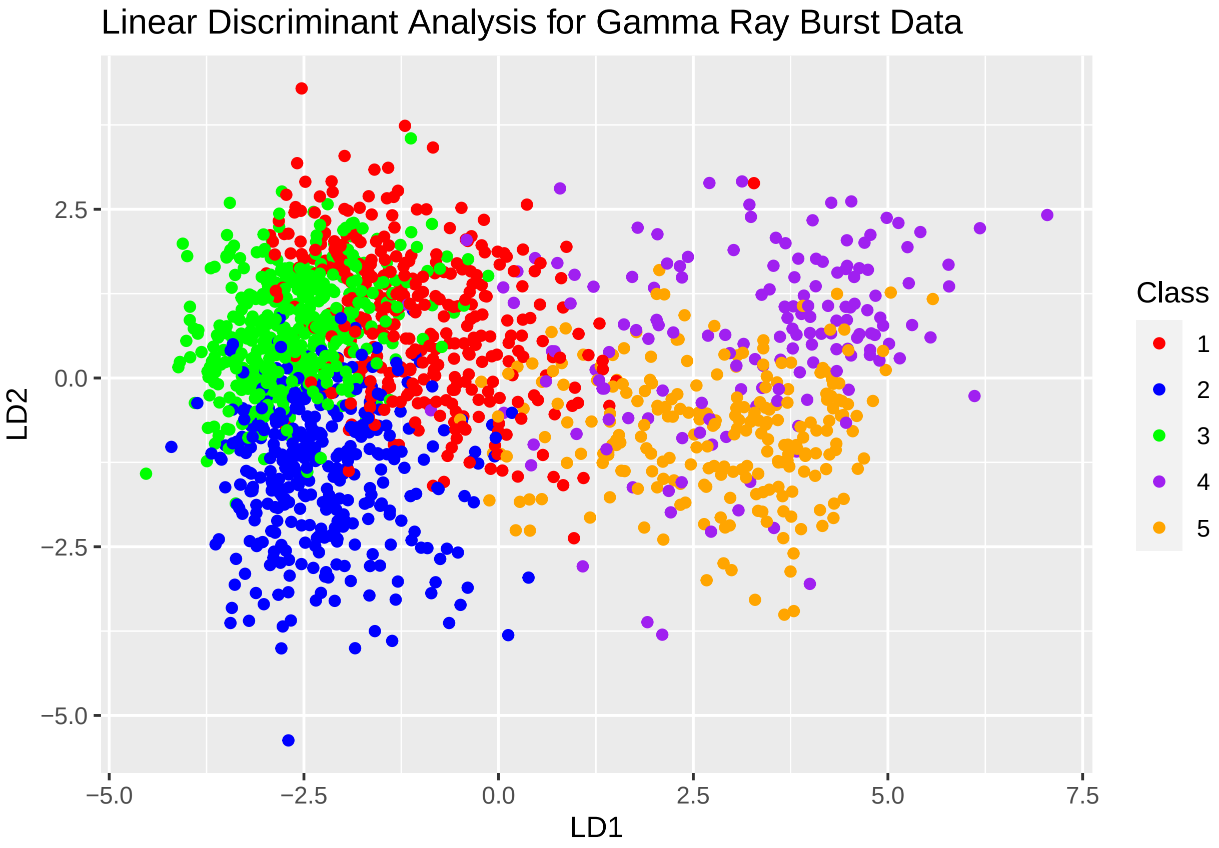
Part B.) Assuming equal prior probabilities and equal costs of misclassifcation, construct Fisher's linear discriminant function. [10 points]

```
## Call:
## lda(class ~ ., data = GRB, prior = c(0.2, 0.2, 0.2, 0.2, 0.2))
##
## Prior probabilities of groups:
##   1   2   3   4   5
## 0.2 0.2 0.2 0.2 0.2
##
## Group means:
```

```
##          T50         T90        F1        F2        F3        F4       P64
## 1  0.7161427  1.09999449 -6.866703 -6.755047 -6.303286 -5.970475 0.1076258
## 2  0.8767503  1.43435781 -5.911697 -5.764700 -5.272722 -5.167775 0.8160456
## 3  1.2404441  1.66702134 -6.271358 -6.178697 -5.776086 -5.860244 0.1346726
## 4 -0.6248785 -0.06790308 -7.979436 -7.770887 -7.052427 -6.489657 0.4285974
## 5 -0.7434956 -0.37041146 -7.901950 -7.607295 -6.820649 -6.444810 0.4985062
##           P256       P1024
## 1 -0.003654604 -0.15569922
## 2  0.781044750  0.68898510
## 3  0.068283039  0.01101267
## 4  0.127604005 -0.35581193
## 5  0.316373558 -0.12758329
##
## Coefficients of linear discriminants:
##               LD1        LD2         LD3        LD4
## T50   -0.11140679  0.5251030   1.9297123 -0.9854645
## T90   -0.03827327  0.3970489  -3.2310124  0.7749042
## F1    -0.46531093  0.5711469  -0.1632299  0.3176527
## F2    -1.19900658 -0.5821010   0.2746655 -1.2768597
## F3     0.26976994 -0.7932447   1.0083814  4.2040301
## F4     0.17174454  0.1467376  -0.1972441 -1.5753747
## P64    1.44760871  6.0739405 -10.6699427  0.0576076
## P256   6.18726628 -9.2732590  10.3833981  3.0383037
## P1024 -6.89112833  1.5003026  -2.0483281 -5.0926131
##
## Proportion of trace:
##    LD1    LD2    LD3    LD4
## 0.8292 0.1084 0.0537 0.0087
```

Because we have 5 classes that the Gamma Ray Bursts can fall in, we have 4 sets of linear discriminant coefficients and thus 4 Fisher Linear Discriminant Functions: The Fisher Linear Discriminant functions are: 1.) $ Class = -0.11140679*T50 - 0.03827327*T90 - 0.46531093*F1 - 1.19900658*F2 + 0.26976994*F3 + 0.17174454*F4 + 1.44760871*P64 + 6.18726628*P256 - 6.89112833*P1024$ 2.) $ Class = 0.5251030*T50 + 0.3970489*T90 + 0.5711469*F1 - 0.5821010*F2 - 0.7932447*F3 + 0.1467376*F4 + 6.0739405*P64 - 9.2732590*P256 + 1.5003026*P1024$ 3.) $ Class = 1.9297123*T50 - 3.2310124*T90 - 0.1632299*F1 + 0.2746655*F2 + 1.0083814*F3 - 0.1972441*F4 - 10.6699427*P64 + 10.3833981*P256 - 2.0483281*P1024$ 4.) $ Class = -0.9854645*T50 + 0.7749042*T90 + 0.3176527*F1 - 1.2768597*F2 + 4.2040301*F3 - 1.5753747*F4 + 0.0576076*P64 + 3.0383037*P256 - 5.0926131*P1024$

Part Bi.) Display the first two linear discriminant coordinates. Do all the variables in the discriminant function appear to be important? [10 points]



```
## [1] "Linear Discriminant Coordinates:"

##               LD1        LD2         LD3        LD4
## T50   -0.11140679  0.5251030   1.9297123 -0.9854645
## T90   -0.03827327  0.3970489  -3.2310124  0.7749042
## F1    -0.46531093  0.5711469  -0.1632299  0.3176527
## F2    -1.19900658 -0.5821010   0.2746655 -1.2768597
## F3     0.26976994 -0.7932447   1.0083814  4.2040301
## F4     0.17174454  0.1467376  -0.1972441 -1.5753747
## P64    1.44760871  6.0739405 -10.6699427  0.0576076
## P256   6.18726628 -9.2732590  10.3833981  3.0383037
## P1024 -6.89112833  1.5003026  -2.0483281 -5.0926131

## [1] "Forward selection:"

## correctness rate: 0.55722;  in: "F2";  variables (1): F2
## correctness rate: 0.68358;  in: "P64";  variables (2): F2, P64
##
##  hr.elapsed min.elapsed sec.elapsed
##       0.000       0.000       1.367

## method      : lda
## final model : as.factor(GRB[, 1]) ~ F2 + P64
## <environment: 0x7fde6c5a4e98>
##
## correctness rate = 0.6836
```

```
## [1] "Backward selection:"
```

```
## correctness rate: 0.75735;  starting variables (9): T50, T90, F1, F2, F3, F4, P64, P256, P1024
## correctness rate: 0.76422;  out: "F1";  variables (8): T50, T90, F2, F3, F4, P64, P256, P1024
##
##  hr.elapsed min.elapsed sec.elapsed
##       0.000       0.000       3.041
```

```
## method      : lda
## final model : as.factor(GRB[, 1]) ~ T50 + T90 + F2 + F3 + F4 + P64 + P256 +
##     P1024
## <environment: 0x7fde6b3e2e10>
##
## correctness rate = 0.7642
```

When we display the first 2 linear discriminate coordinates we can see that the 5 classes do have some separation in the scatter plot. The first discriminant coordinate (x-axis) separates class 1,2,3 and class 4,5 pretty well, although there is still some overlap. The second discriminat coordinate(y-axis) shows minimal separation between class 1,3,4 and class 2,5. But the classes are not perfectly separate even when using the first 2 discriminate coordinates.

Do all the variables in the discriminant function appear to be important? To determine what variables are important we can interpret the linear discriminant coordinates similar to how we would interpret the PC's of a principal component analsysis. In our linear discriminant analysis, we can see that the first discriminant coordinate shows a contrast between (T50,T90,F1,F2, P1024) and (F3, F4,P64 and P256).The second discriminant coordinate shows a contrast between (F2,F3, P256) and all other variables. The third discriminant coordinate shows a contrast between (T90,F1, F4,P64, P1024) and (T50, F2,F3,P256). The fourth discriminant coordinate shows a contrast between (T50,F2,F4, P1024) and (T90, F1, F3,P64,P256). After looking at the discriminate coordinates, it appears to me that all of the variables do appear to be important.

Another way we can check is to use the stepclass method found in the klaR package. This function allows us to doa forward/backward variable selection for classification. When I run the forward selection, it appears that the most important variables to predict class are F2 and P64. When I ran the backward selection, the most important variables to predict class are T50, T90, F2, F3, F4, P64, P256, P1024, thus most of the variables are important when using the backwards variable selection method.

Part Bii.) Calculate the misclassification rates using the AER and the leave-one-out cross-validation method. [10 points]

```
lda_2 <- lda(class ~., data=GRB, prior=c(.2,.2,.2,.2,.2), CV=TRUE)
print("Prediction of Class Membership:")
```

```
## [1] "Prediction of Class Membership:"
```

```
table(GRB$class, lda_2$class)
```

```
##
##       1   2   3   4   5
##   1 248  19  75   4  14
##   2  19 297  47   0   0
##   3  70  27 382   0   0
##   4  11   3   0 113  33
##   5  14   4   0  36 183
```

```
print("Error Rate:")
```

```
## [1] "Error Rate:"
```

```
mean(GRB$class != lda_2$class)
```

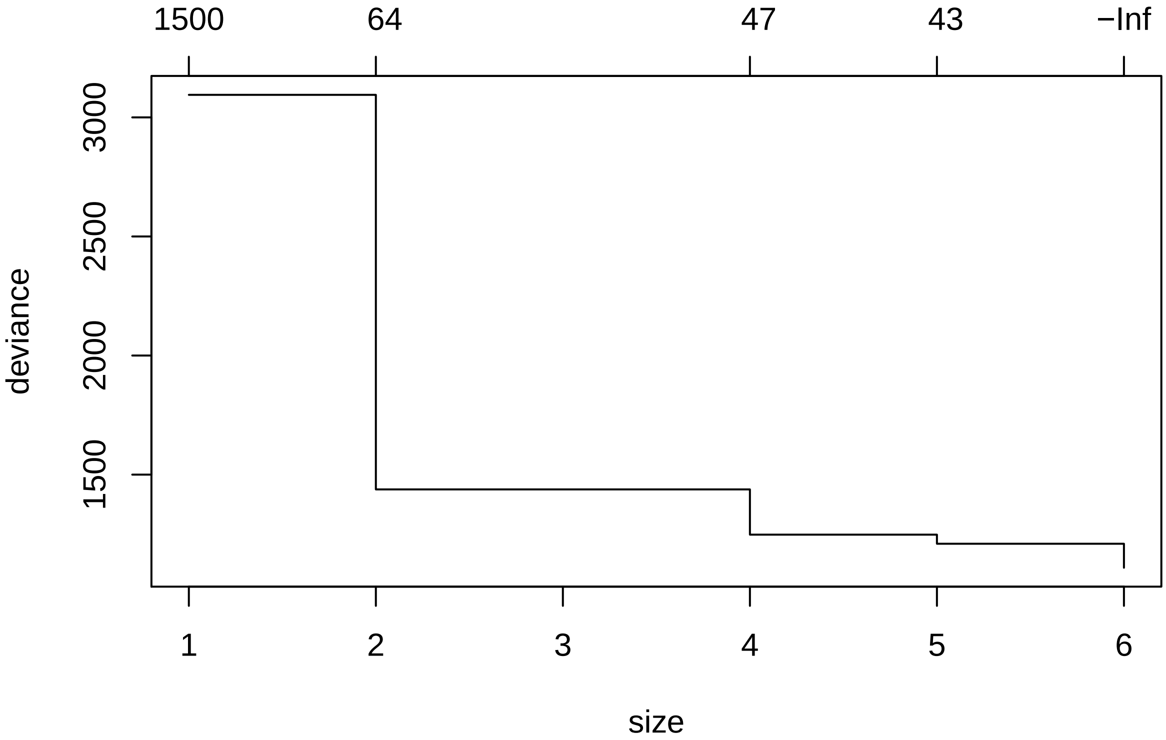
```
## [1] 0.235147
```

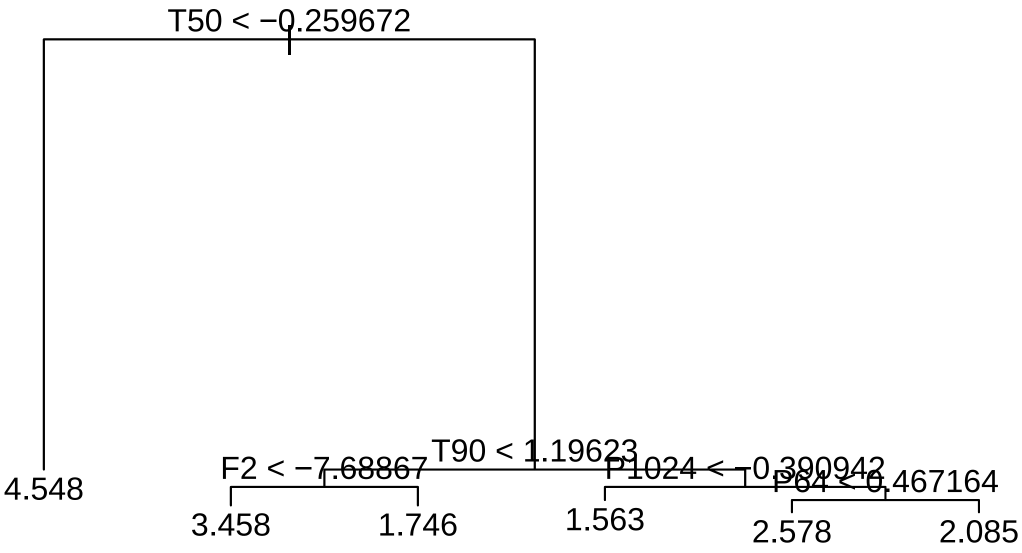
Part C.) Repeat the same exercise as in (b) but using quadratic discriminant analysis, CART and k-nearest neighbors. Choose the best tree or the number of nearest neighbors by cross-validation. Summarize the performance of the results for all four cases (LDA, QDA, CART and k-NN). [10 + 15+10+5 points]

```
## [1] "Prediction of Class Membership:"
```

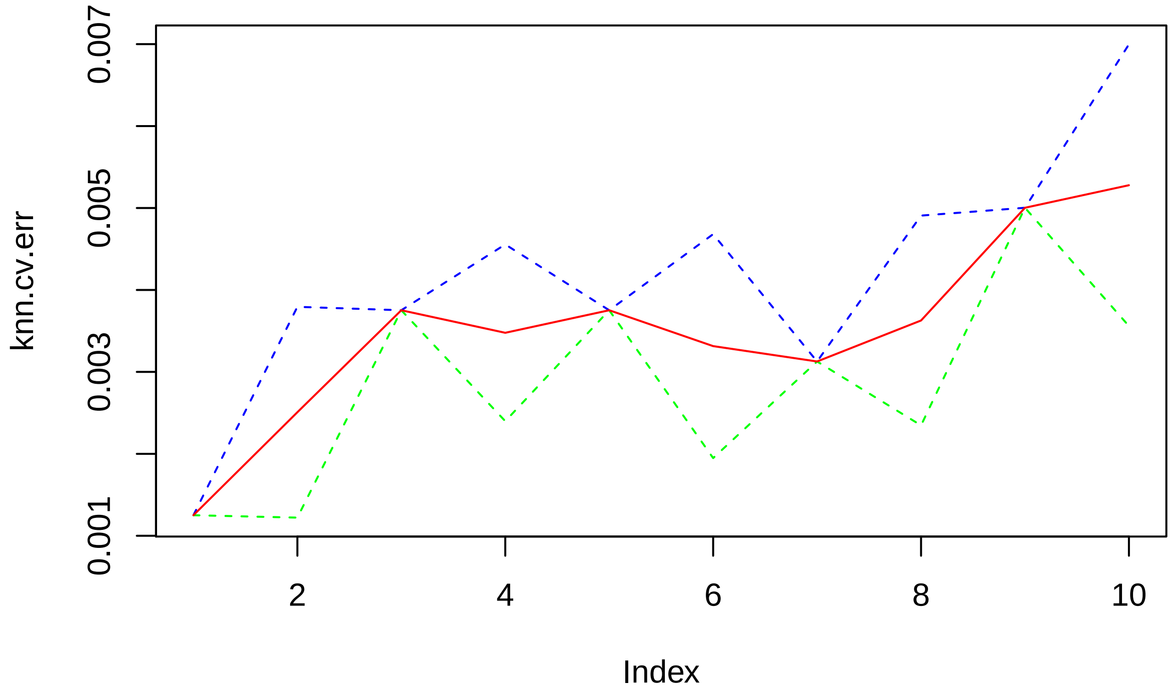
```
##
##       1   2   3   4   5
##   1 344   6   8   1   1
##   2   4 352   6   1   0
##   3  13   9 454   3   0
##   4   0   0   0 159   1
##   5   1   0   0   5 231
```

```
## [1] 0.03689806
```

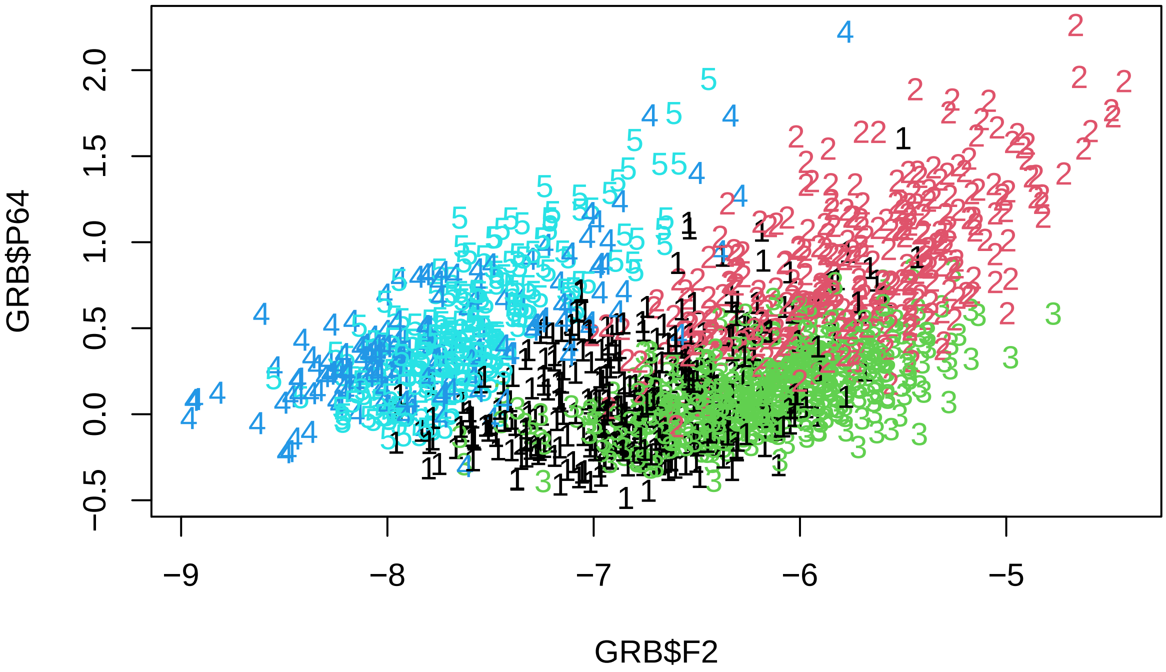

1500
64
47
43
−Inf
3000
2500
2000
1500
deviance
1
2
3
4
5
6
size



```
##
##  Done i=  1
##  Done i=  2
##  Done i=  3
##  Done i=  4
##  Done i=  5
##  Done i=  6
##  Done i=  7
##  Done i=  8
##  Done i=  9
##  Done i=  10
```



```
##  [1] 0.001250782 0.002507817 0.003752345 0.003477173 0.003752345 0.003314572
##  [7] 0.003126954 0.003627267 0.005003127 0.005278299
```



Summarize the performance of the results for all four cases (LDA, QDA, CART and k-NN).

The LDA result in 4 linear discriminant functions. The first two discriminant coefficients showed very little separation in the data. When selecting which variables to use it appears that all of the variables are important when looking at the LD's, but when we use a formal method (i.e. stepclass()) we see that all of the variables are important to predict the class the GRB falls into except for the F1 variable representing the time integrate fulences in the 20-50 keV spectral channel. The misclassification error rate was 0.235147.

The QDA results show that it performs better than the LDA. The class predictions have smaller numbers on the off diagonal, which I believe means that the GRBs are being classified into the correct class. This is also supported with a misclassification error rate of 0.03689806, which is smaller than the LDA's error rate.

Using the Classification and Regression Tree method, I found that the best tree has 6 clusters.

Here I tried to determine what the best k value to use was. Looking at the cross-validate error rates we see that k = 1 is best because all 3 The error rate is 0.001250782. If we look at the plot of the F2 variable vs the P64 variable we can see very little separation between the classes. Possibly I am not comparing the most important variables though.

Part D.) For each of the groups, test the hypothesis that a fewer number of factors is adequate to express the variables in the dataset. [15 points]

```
# use a likelihood Ratio Test
library(fad)
```

```
## Loading required package: RSpectra
## Please cite the paper: Dai, F., Dutta, S., and, Maitra, R. (2020). A Matrix-Free Likelihood Method fo
```

```
# Class 1
# factanal(x=as.matrix(group1[,-1]), factors=1, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group1[,-1]), factors=2, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group1[,-1]), factors=3, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group1[,-1]), factors=4, method = "mle", scale=T, center=T)
factanal(x=as.matrix(group1[,-1]), factors=5, method = "mle", scale=T, center=T)
```

```
## 
## Call:
## factanal(x = as.matrix(group1[, -1]), factors = 5, method = "mle",     scale = T, center = T)
## 
## Uniquenesses:
##   T50   T90    F1    F2    F3    F4   P64  P256 P1024 
## 0.045 0.005 0.141 0.005 0.005 0.684 0.031 0.005 0.005 
## 
## Loadings:
##       Factor1 Factor2 Factor3 Factor4 Factor5
## T50   -0.138   0.914   0.301                 
## T90   -0.149   0.945   0.271                 
## F1     0.241   0.681   0.144   0.562         
## F2     0.294   0.647   0.385   0.585         
## F3     0.362   0.421   0.817   0.134         
## F4     0.171   0.202   0.491                 
## P64    0.959           0.177                 
## P256   0.980           0.161                 
## P1024  0.935           0.272   0.134   0.168 
## 
##                Factor1 Factor2 Factor3 Factor4 Factor5
## SS loadings      3.100   2.830   1.372   0.729   0.042
## Proportion Var   0.344   0.314   0.152   0.081   0.005
## Cumulative Var   0.344   0.659   0.811   0.892   0.897
## 
## Test of the hypothesis that 5 factors are sufficient.
## The chi square statistic is 7.86 on 1 degree of freedom.
## The p-value is 0.00506
```

```
# No

# Class 2
# factanal(x=as.matrix(group2[,-1]), factors=1, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group2[,-1]), factors=2, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group2[,-1]), factors=3, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group2[,-1]), factors=4, method = "mle", scale=T, center=T)
factanal(x=as.matrix(group2[,-1]), factors=5, method = "mle", scale=T, center=T)
```

```
## 
## Call:
## factanal(x = as.matrix(group2[, -1]), factors = 5, method = "mle",     scale = T, center = T)
## 
## Uniquenesses:
##   T50   T90    F1    F2    F3    F4   P64  P256 P1024 
## 0.018 0.239 0.005 0.005 0.005 0.034 0.005 0.005 0.016
##
## Loadings:
##       Factor1 Factor2 Factor3 Factor4 Factor5
## T50            0.969   0.122   0.141
## T90   -0.128   0.817   0.217   0.168
## F1     0.515   0.465   0.238   0.670
## F2     0.533   0.458   0.360   0.597   0.120
## F3     0.517   0.449   0.562   0.435   0.145
## F4     0.437   0.412   0.749   0.209
## P64    0.967           0.174   0.163
## P256   0.964           0.180   0.176
## P1024  0.929           0.243   0.242
##
##                Factor1 Factor2 Factor3 Factor4 Factor5
## SS loadings      3.756   2.416   1.247   1.203   0.050
## Proportion Var   0.417   0.268   0.139   0.134   0.006
## Cumulative Var   0.417   0.686   0.824   0.958   0.964
##
## Test of the hypothesis that 5 factors are sufficient.
## The chi square statistic is 391.53 on 1 degree of freedom.
## The p-value is 3.85e-87
```

```
# No

# Class 3
# factanal(x=as.matrix(group3[,-1]), factors=1, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group3[,-1]), factors=2, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group3[,-1]), factors=3, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group3[,-1]), factors=4, method = "mle", scale=T, center=T)
factanal(x=as.matrix(group3[,-1]), factors=5, method = "mle", scale=T, center=T)
```

```
##
## Call:
## factanal(x = as.matrix(group3[, -1]), factors = 5, method = "mle",     scale = T, center = T)
##
## Uniquenesses:
##   T50   T90    F1    F2    F3    F4   P64  P256 P1024
## 0.101 0.026 0.080 0.005 0.005 0.530 0.005 0.005 0.005
##
## Loadings:
##       Factor1 Factor2 Factor3 Factor4 Factor5
## T50            0.199   0.911   0.170
## T90            0.213   0.942   0.201
## F1     0.311   0.866   0.237   0.117
## F2     0.336   0.856   0.264   0.279
## F3     0.370   0.612   0.277   0.624   0.132
## F4     0.212   0.168   0.273   0.567
## P64    0.955   0.233           0.157
## P256   0.954   0.240           0.169
## P1024  0.940   0.275           0.187
##
##                Factor1 Factor2 Factor3 Factor4 Factor5
## SS loadings      3.095   2.159   1.995   0.959   0.031
## Proportion Var   0.344   0.240   0.222   0.107   0.003
## Cumulative Var   0.344   0.584   0.805   0.912   0.915
##
## Test of the hypothesis that 5 factors are sufficient.
## The chi square statistic is 30.38 on 1 degree of freedom.
## The p-value is 3.56e-08
```

```
# Class 4
# factanal(x=as.matrix(group4[,-1]), factors=1, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group4[,-1]), factors=2, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group4[,-1]), factors=3, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group4[,-1]), factors=4, method = "mle", scale=T, center=T)
factanal(x=as.matrix(group4[,-1]), factors=5, method = "mle", scale=T, center=T)
```

```
##
## Call:
## factanal(x = as.matrix(group4[, -1]), factors = 5, method = "mle",     scale = T, center = T)
##
## Uniquenesses:
##   T50   T90    F1    F2    F3    F4   P64  P256 P1024
## 0.145 0.005 0.196 0.052 0.026 0.245 0.051 0.011 0.005
##
## Loadings:
##       Factor1 Factor2 Factor3 Factor4 Factor5
## T50           0.915
## T90    0.243   0.950   0.174
## F1     0.730   0.393   0.284   0.188
## F2     0.902   0.319   0.106   0.131
## F3     0.743   0.307   0.562
## F4     0.397   0.156   0.757
## P64    0.860  -0.161   0.348  -0.245
## P256   0.885   0.105   0.421  -0.120
## P1024  0.844   0.299   0.409           0.154
##
##                Factor1 Factor2 Factor3 Factor4 Factor5
## SS loadings      4.356   2.242   1.482   0.145   0.038
## Proportion Var   0.484   0.249   0.165   0.016   0.004
## Cumulative Var   0.484   0.733   0.898   0.914   0.918
##
## Test of the hypothesis that 5 factors are sufficient.
## The chi square statistic is 1.2 on 1 degree of freedom.
## The p-value is 0.272
```

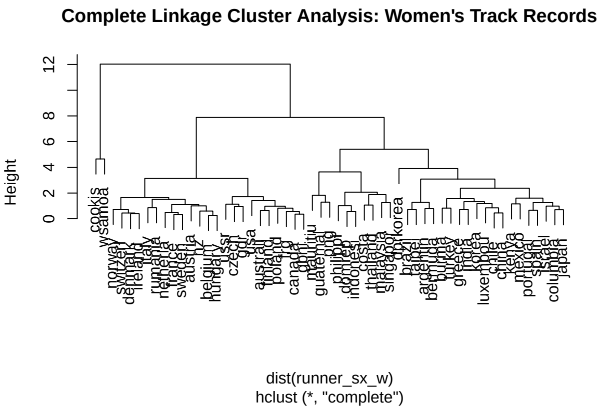
```
# Yes 4 factors!

# Class 5
# factanal(x=as.matrix(group5[,-1]), factors=1, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group5[,-1]), factors=2, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group5[,-1]), factors=3, method = "mle", scale=T, center=T)
# factanal(x=as.matrix(group5[,-1]), factors=4, method = "mle", scale=T, center=T)
factanal(x=as.matrix(group5[,-1]), factors=5, method = "mle", scale=T, center=T)
```

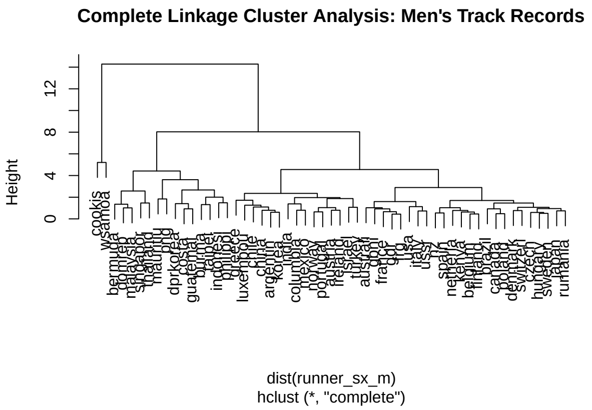
```
##
```

```
## Call:
## factanal(x = as.matrix(group5[, -1]), factors = 5, method = "mle",     scale = T, center = T)
##
## Uniquenesses:
##   T50   T90    F1    F2    F3    F4   P64  P256 P1024
## 0.102 0.154 0.005 0.005 0.005 0.350 0.037 0.005 0.005
##
## Loadings:
##       Factor1 Factor2 Factor3 Factor4 Factor5
## T50            0.940                  -0.103
## T90            0.916
## F1     0.810           0.250   0.523
## F2     0.972   0.135   0.163
## F3     0.754           0.645
## F4     0.329           0.732
## P64    0.811  -0.422   0.323           0.135
## P256   0.853  -0.249   0.423           0.156
## P1024  0.860           0.493
##
##                Factor1 Factor2 Factor3 Factor4 Factor5
## SS loadings      4.406   2.000   1.571   0.291   0.068
## Proportion Var   0.490   0.222   0.175   0.032   0.008
## Cumulative Var   0.490   0.712   0.886   0.919   0.926
##
## Test of the hypothesis that 5 factors are sufficient.
## The chi square statistic is 84.22 on 1 degree of freedom.
## The p-value is 4.43e-20
```

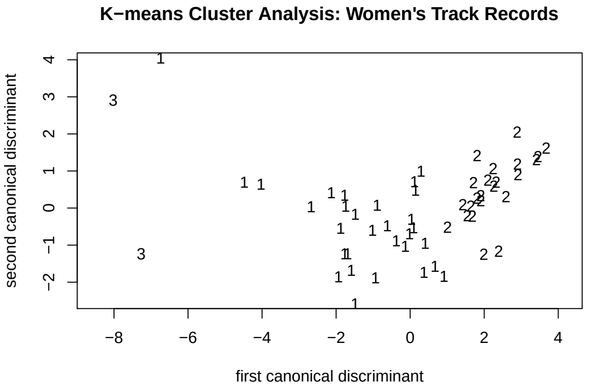
Here we tested the hypothesis that for each class of GRB, the variables can be represented for a fewer number of factors. For each class I performed a factor discriminatory analysis using factanal(). I did this for 5,4,3,2,1 factors. What I found is that most of the classes can not be explained by fewer than 5 factors except for Class 4 which can be explained by 4 factors! We know this because when I performed and fda on the data from class 4, the resulting p-value was 0.272 which is not significant enought to reject the null hypothesis.

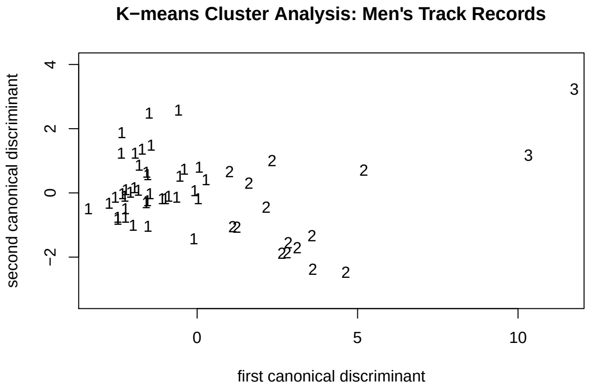
Question 2: Compare the results of clustering obtained using the Women's and Men's track records results. Use hierarchical clustering with the correlation similarity matrix, k-means and model-based clustering. Display the results using appropriate graphical aids. Comment. [30 points]



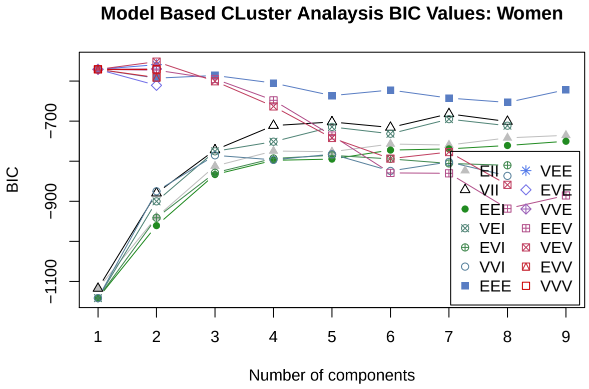


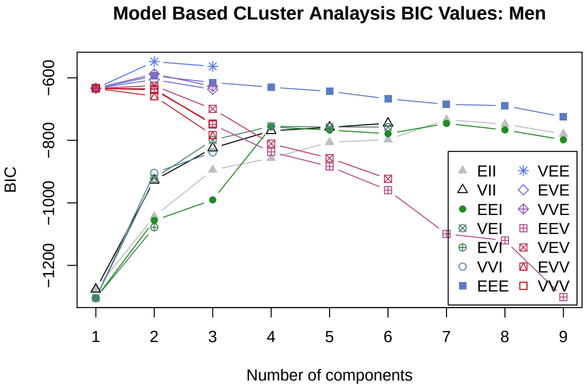
Comment: What is interesting is that we can see the division of largely 2 clusters for both the male and the female track records. Even more so one of those clusters is the same between male and female. The other main group is similar between the two sexes as well with some slight differences. Based on this I would probably cut this tree into 3 main clusters for both sexes.





Comment: When we plot the clusters using the K-Means Analysis we see roughly a similar pattern. The two largest clusters 1 and 2 are generally clustered tighter together while, the 3rd cluster is very small and very sparse. Maybe the samples in this cluster could be an outlier. I would say overall there still isn't great separation between the clusters.


Model Based CLuster Analaysis BIC Values: Women


Model Based CLuster Analaysis BIC Values: Men

Comment:I had trouble plotting the actual data after determing the clusters via model based clustering. Instead I showed a graph of the BIC values to show that the women's data is best displayed with a VVV model with 2 clusters and the men's data is best displayed with the VEE model and 2 clusters as well.

I ran in to trouble plotting the data because I tried to use the classification from mclust() to run an lda, but because the optimum # of clusters is 2 this results in 1 linear discriminant function and I need at least 2 to plot on the x and y axes.