Exploratory Data Mining via Search Strategies Lab #4

Ross Jacobucci & Kevin J. Grimm

Outline

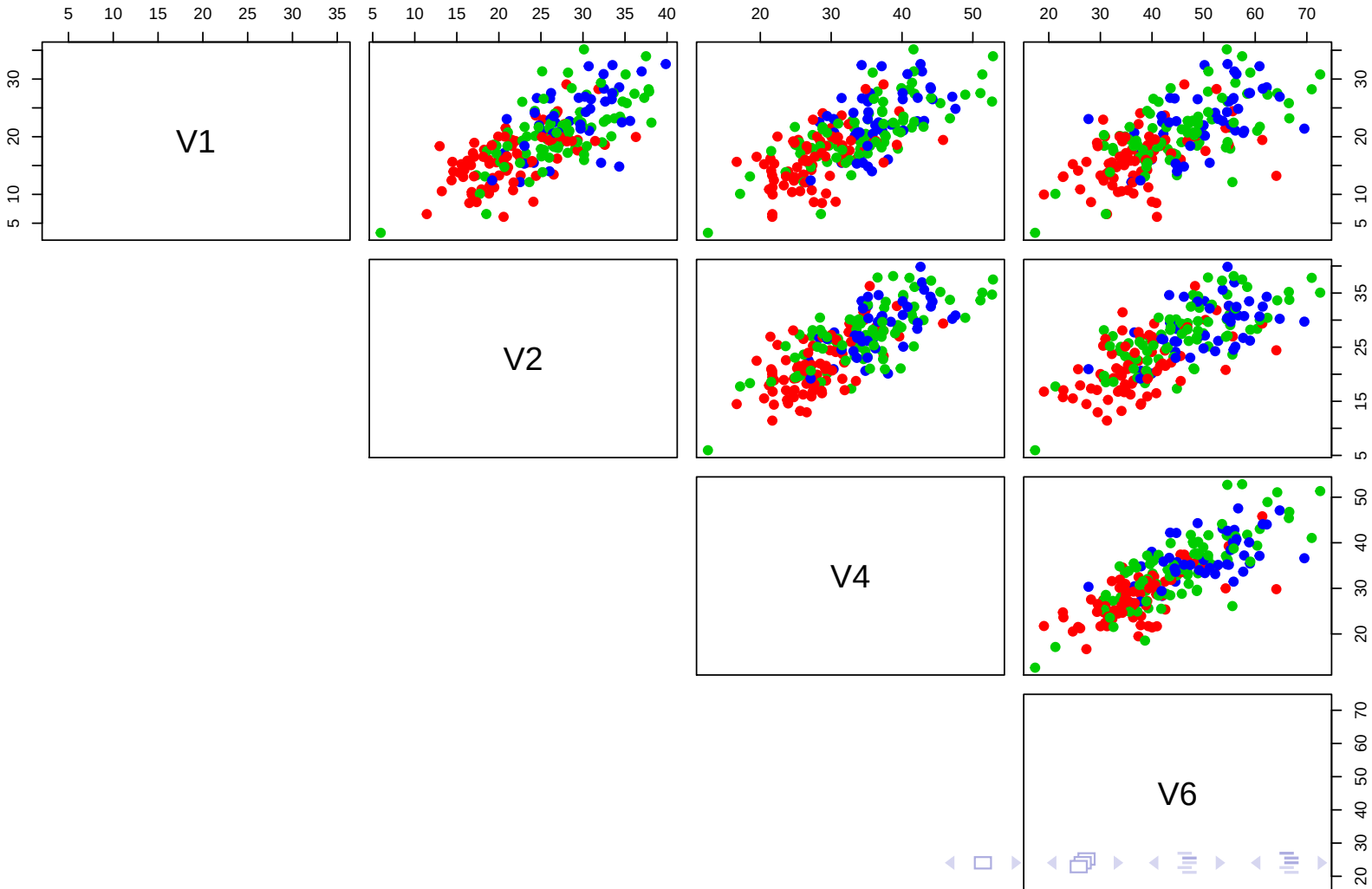
We will use some of the same packages used in the lectures to both clustering and finite mixture models.
To do this, we are going to use the WISC dataset that we will also use tomorrow. This is longitudinal data collected on kids in elementary school on verbal and performance scales along with mother's education.Data is WISC4VPE.DAT.

```
wisc <- read.table(
  "C:/Users/RJacobucci/Documents/GitHub/EDM_Labs/2015/wisc4vpe.dat")
names(wisc)<- c("V1","V2","V4","V6","P1","P2","P4", "P6", "Moeducat")
# note: V1 refers to verbal scores at grade 1, P is performance
```

Most analyses will not explicitly take into account the longitudinal nature of the data. However, we will look at a R package for longitudinal clustering at the end of the lab. In creating groups of individuals, we are going to compare these results to just classifying based on what their mother's education was.
First we will start with visualizing the data. Code adapted from : `https://cran.r-project.org/web/packages/dendextend/vignettes/Cluster_Analysis.html`

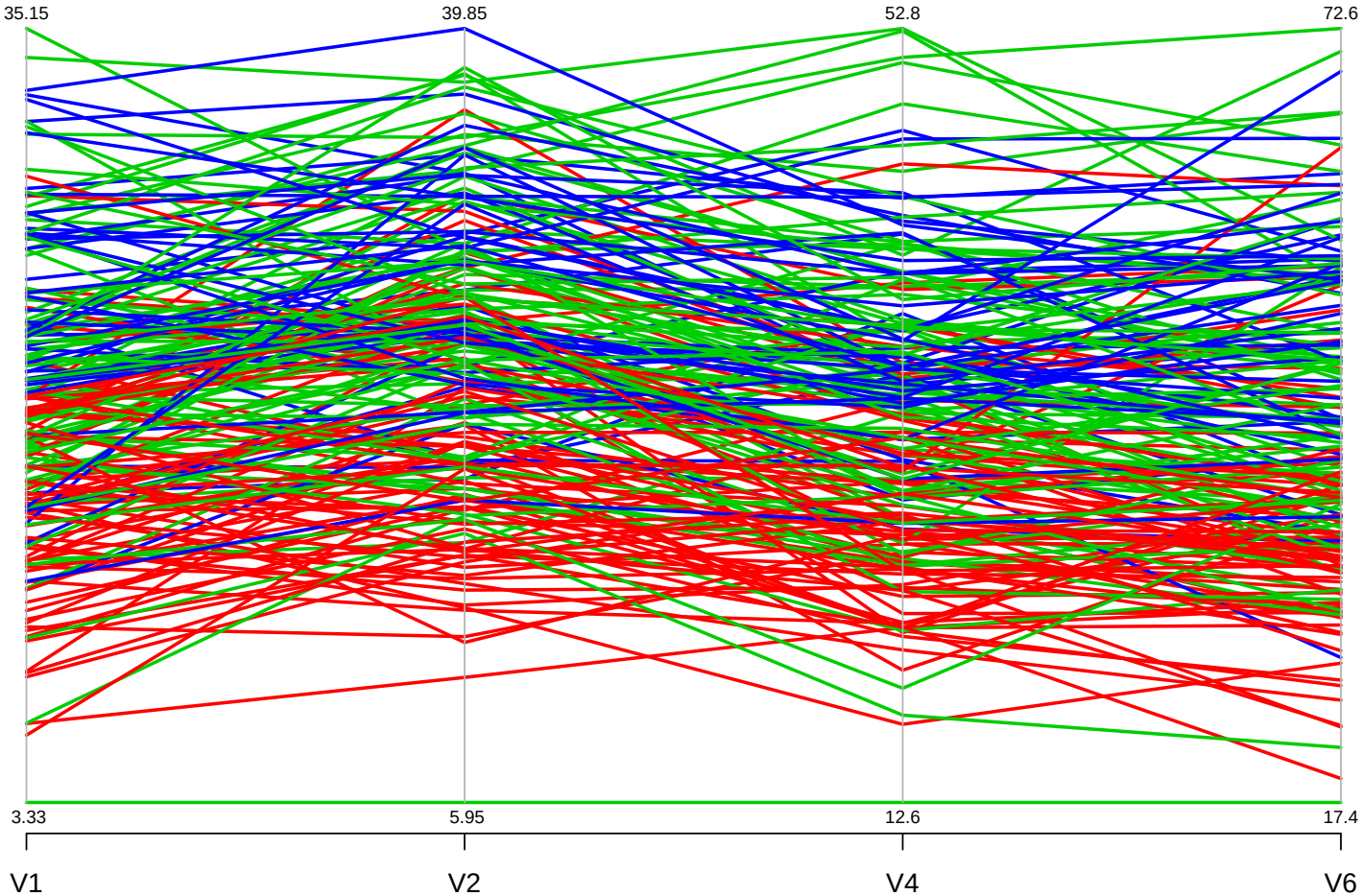
Visualize

```
col_class <- wisc$Moeducat + 2
# low = red, medium = green, high = blue
pairs(wisc[,1:4], col = col_class,lower.panel = NULL,
      cex.labels=2, pch=19, cex = 1.2)
```



Visualize

```
MASS::parcoord(wisc[,1:4], col = col_class, var.label = TRUE, lwd = 2)
```

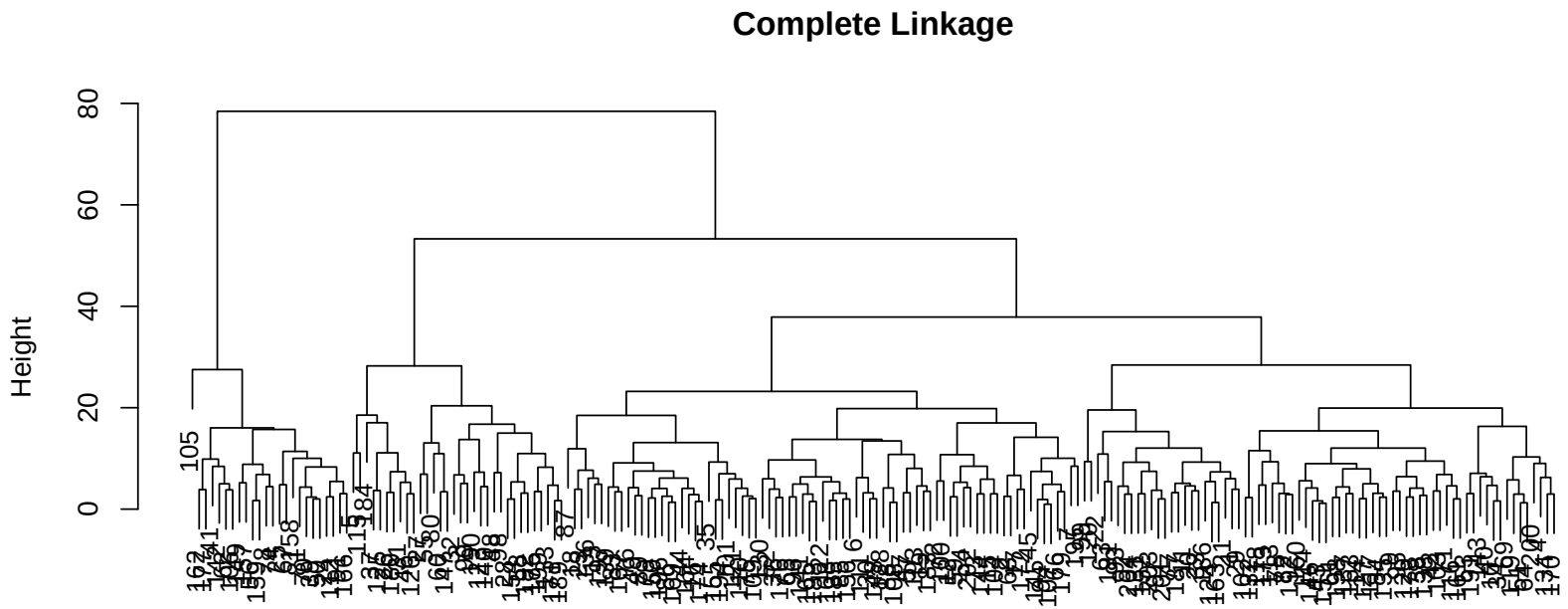


Hierarchical Clustering

Complete Linkage

Using hclust() that is built into R

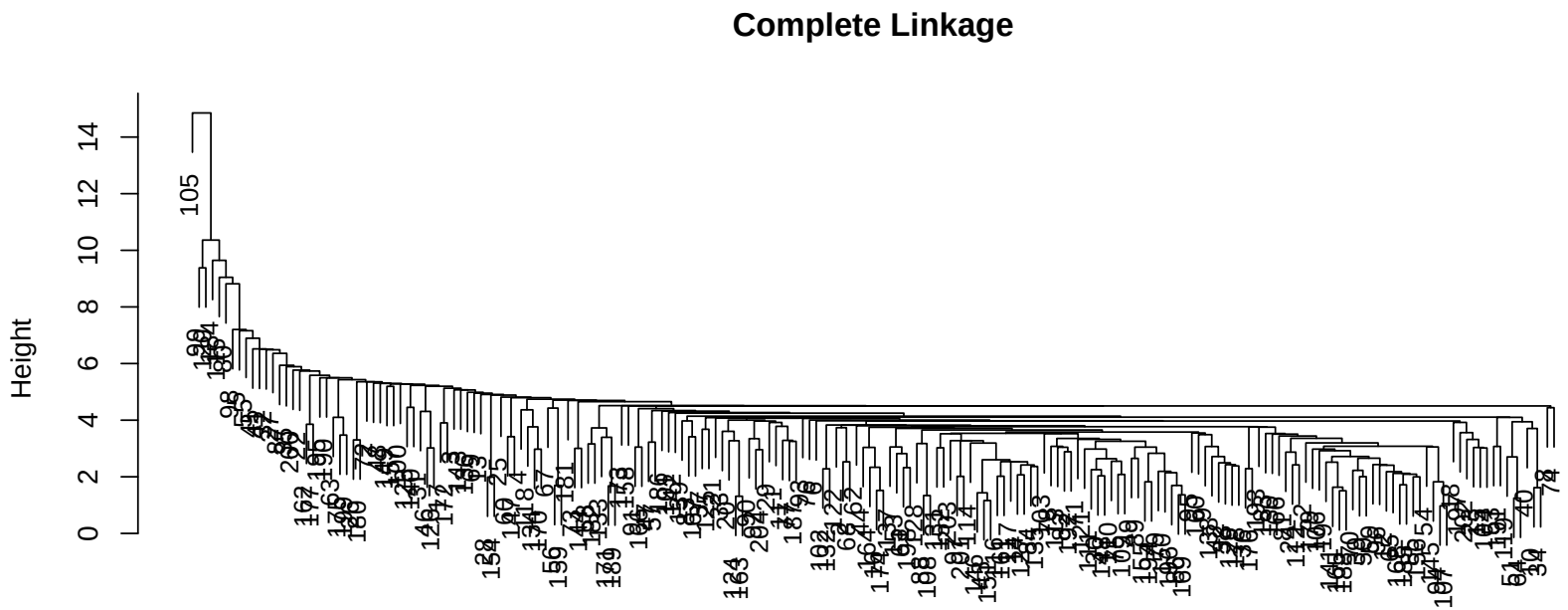
```
wisc.dist <- dist(wisc[,1:4])
hc.clust.1 = hclust(wisc.dist, method='complete')
plot(hc.clust.1, main='Complete Linkage', xlab='', sub='', cex=.9)
```



Hierarchical Clustering

Single Linkage

```
wisc.dist <- dist(wisc[,1:4])
hc.clust.2 = hclust(wisc.dist, method='single')
plot(hc.clust.2, main='Complete Linkage', xlab='', sub='', cex=.9)
```



Hierarchical Clustering

Who is case #105?

```
wisc[105,1:4]
```

```
##            V1       V2       V4       V6
## 105 3.333333 5.952381 12.60417 17.35119
```

```
summary(wisc[,1:4])
```

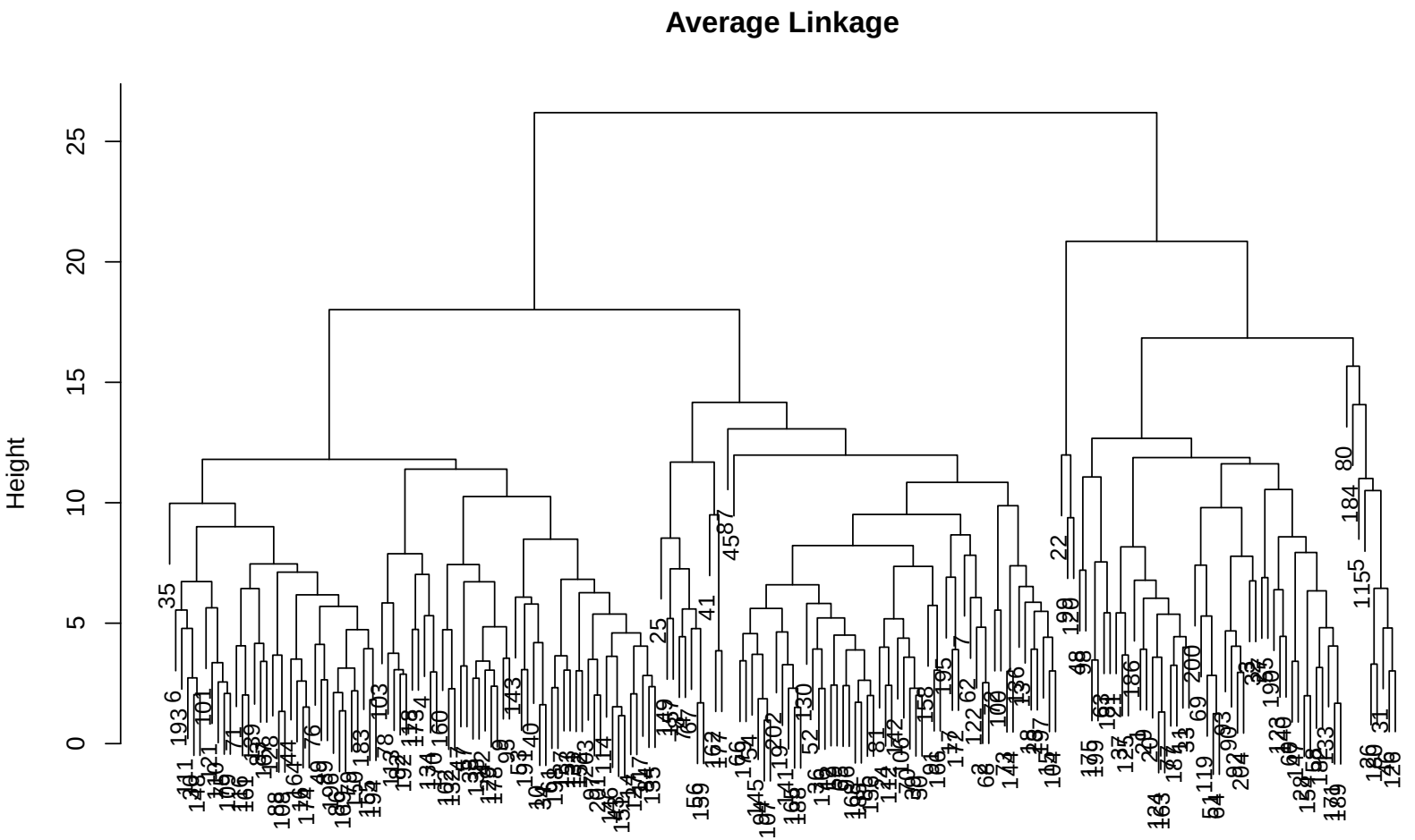
```
##        V1                V2               V4               V6
##  Min.   : 3.333   Min.   : 5.952   Min.   :12.60   Min.   :17.35
##  1st Qu.:15.636   1st Qu.:20.778   1st Qu.:27.24   1st Qu.:35.97
##  Median :19.330   Median :25.982   Median :32.82   Median :42.54
##  Mean   :19.585   Mean   :25.415   Mean   :32.61   Mean   :43.75
##  3rd Qu.:22.839   3rd Qu.:29.695   3rd Qu.:37.22   3rd Qu.:51.00
##  Max.   :35.149   Max.   :39.851   Max.   :52.84   Max.   :72.59
```

Hey, we found an outlier!

Average Linkage

```
wisc.dist2 <- dist(wisc[-105,1:4])
hc.clust.3 = hclust(wisc.dist2, method='average')

plot(hc.clust.3, main='Average Linkage', xlab='', sub='', cex=.9)
```



Comparing Clusters to Mother's Education

Are we really just clustering people based on the family they come from?

```
pred.3 = cutree(hc.clust.3,3)
table(pred.3, wisc$Moeducat[-105]+1)
```

```
##
## pred.3  1  2  3
##      1  4 26 23
##      2 70 54 23
##      3  2  1  0
```

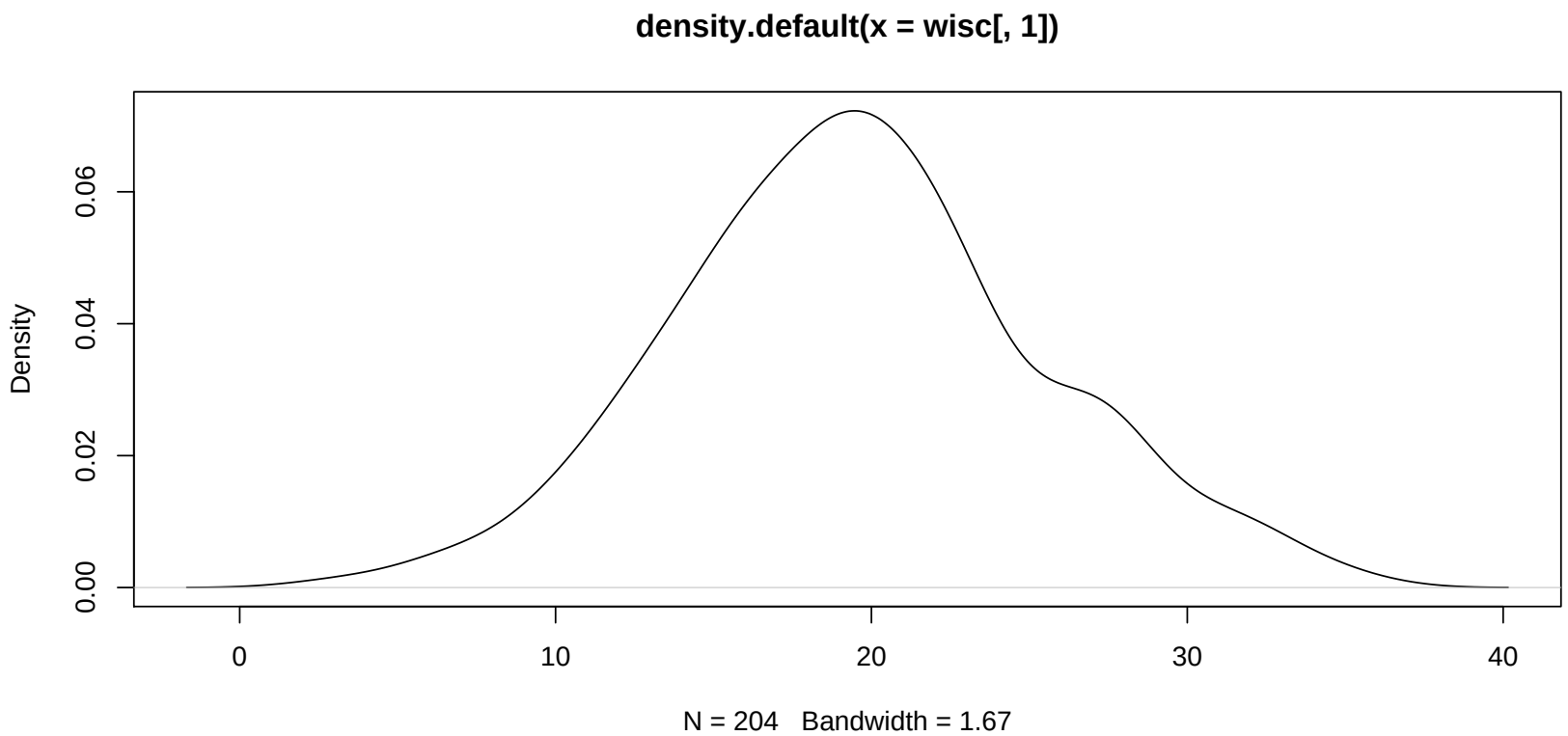
Looks like there is another factor involved other than mother's education

Finite Mixtures

Univariate Visualization

First Score

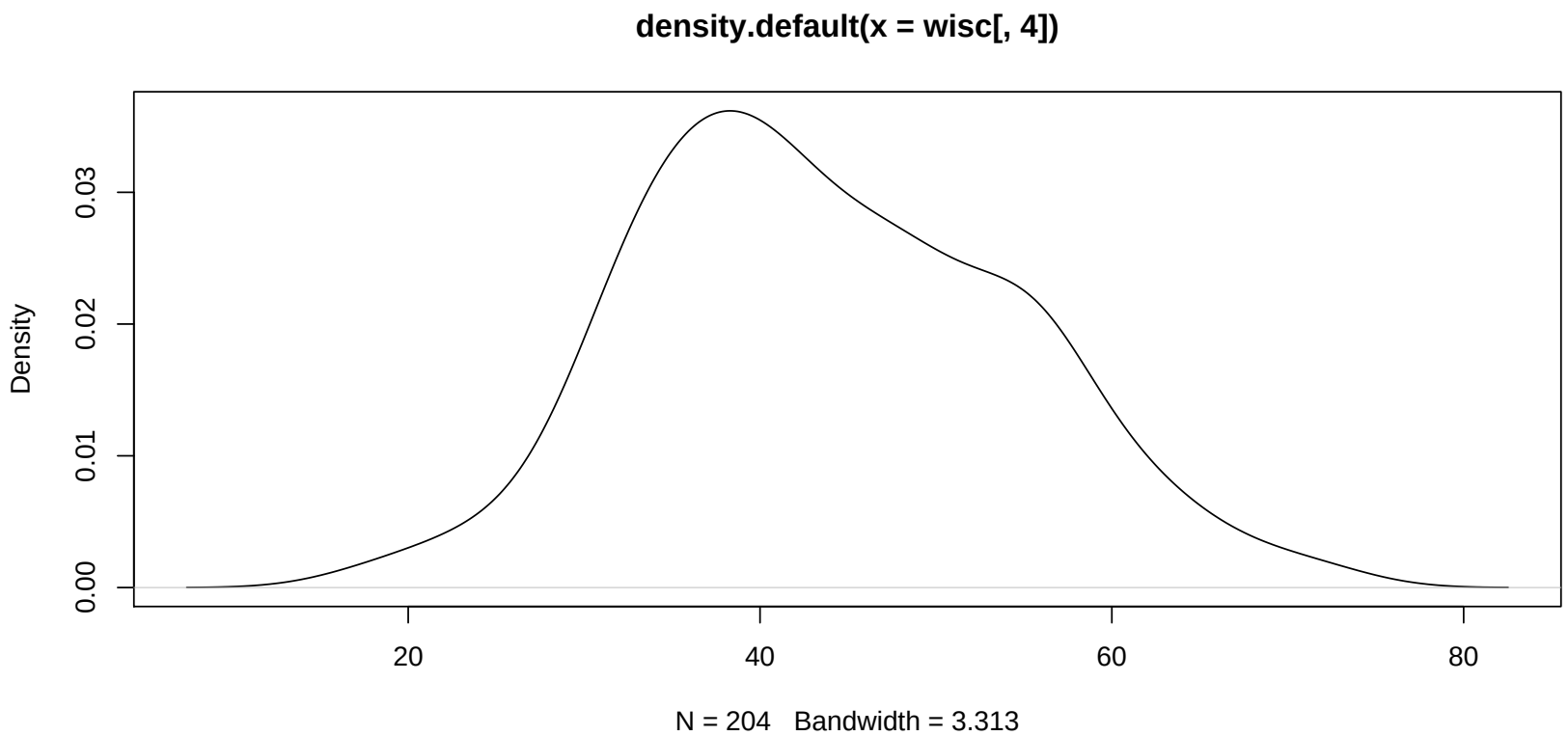
```
dd <- density(wisc[,1])
plot(dd)
```



Univariate Visualization

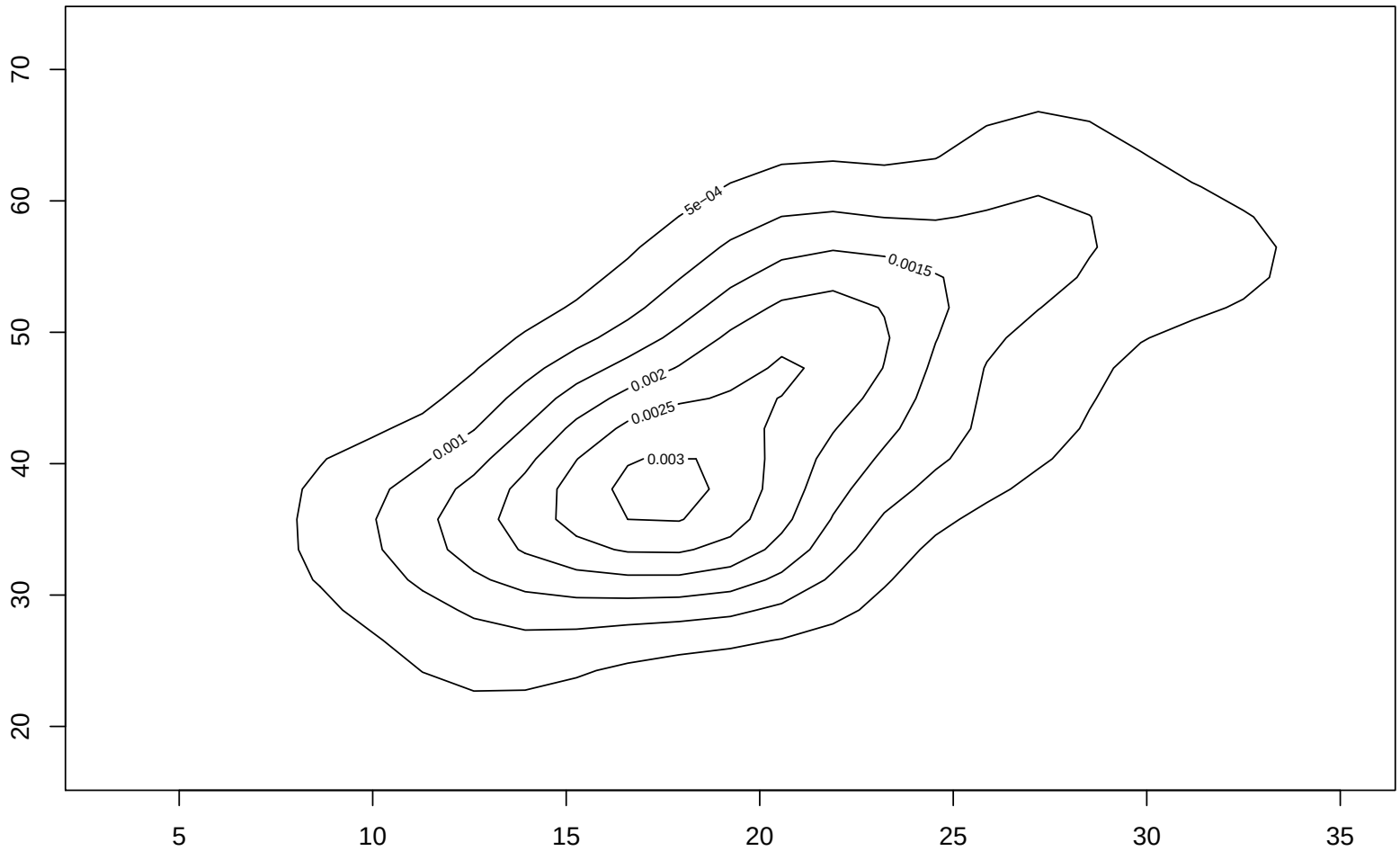
Last Score

```
dd <- density(wisc[,4])
plot(dd)
```



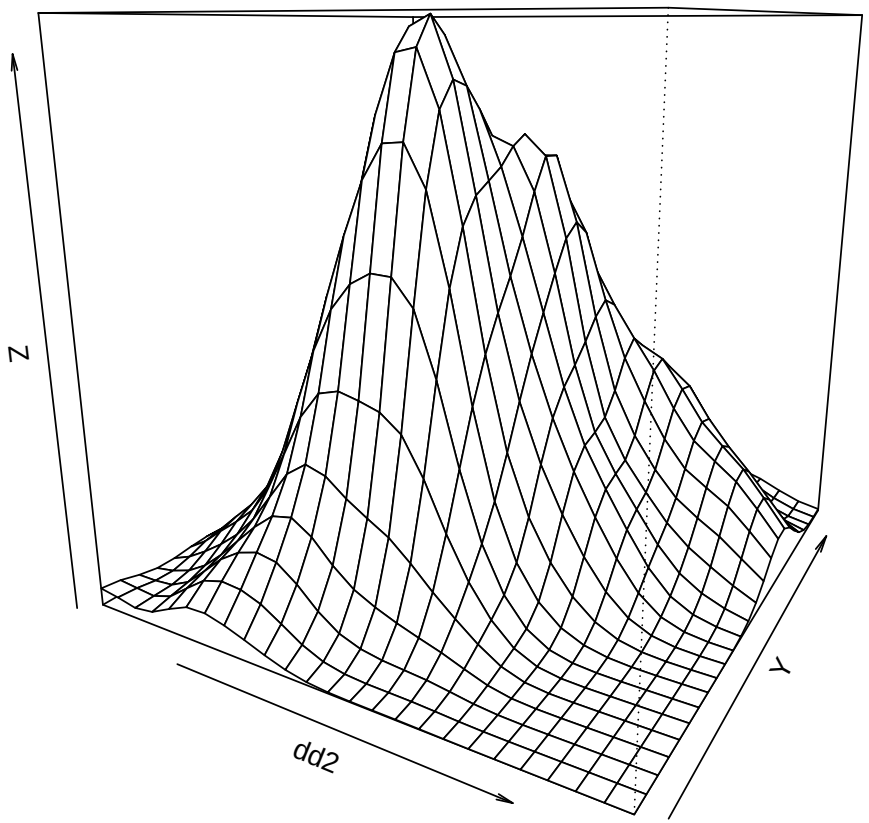
Bivariate Visualization

```
dd2 <- MASS::kde2d(wisc[,1],wisc[,4])
contour(dd2)
```



Bivariate Visualization

```
persp(dd2,theta=30,phi=15)
```



Mixture Visualization

What did we learn?

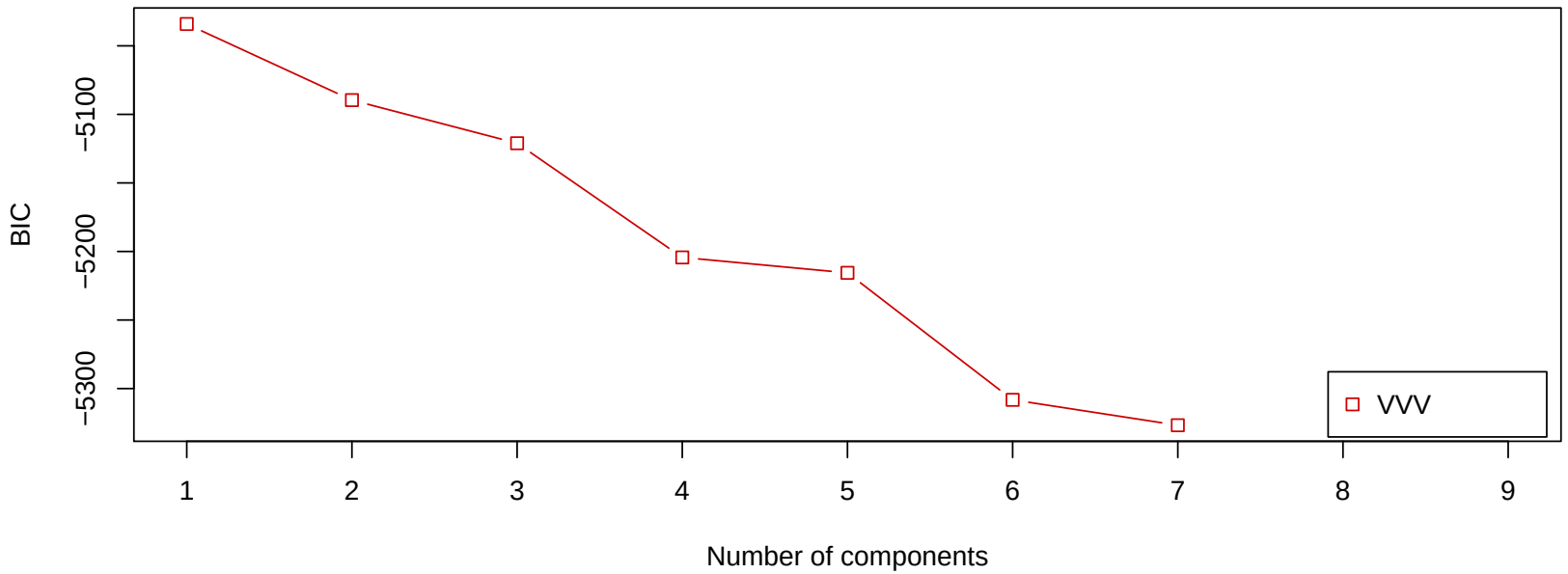
There seems to be some non-normality to the univariate and bivariate distributions. This means that finite mixtures are likely to find 2 or more classes underlying the multivariate distribution

Test Finite Mixtures

Vignette for Mclust
https://cran.r-project.org/web/packages/mclust/vignettes/mclust.html

```
library(mclust)
library(mixtools)

# for multivariate, N > d, and spherical i.e. latent class would be "VVI"
Mix.1 = Mclust(wisc[,1:4], G = 1:9,modelNames=c("VVV"))
plot(Mix.1,"BIC")
```



Look at density

Try different mixture

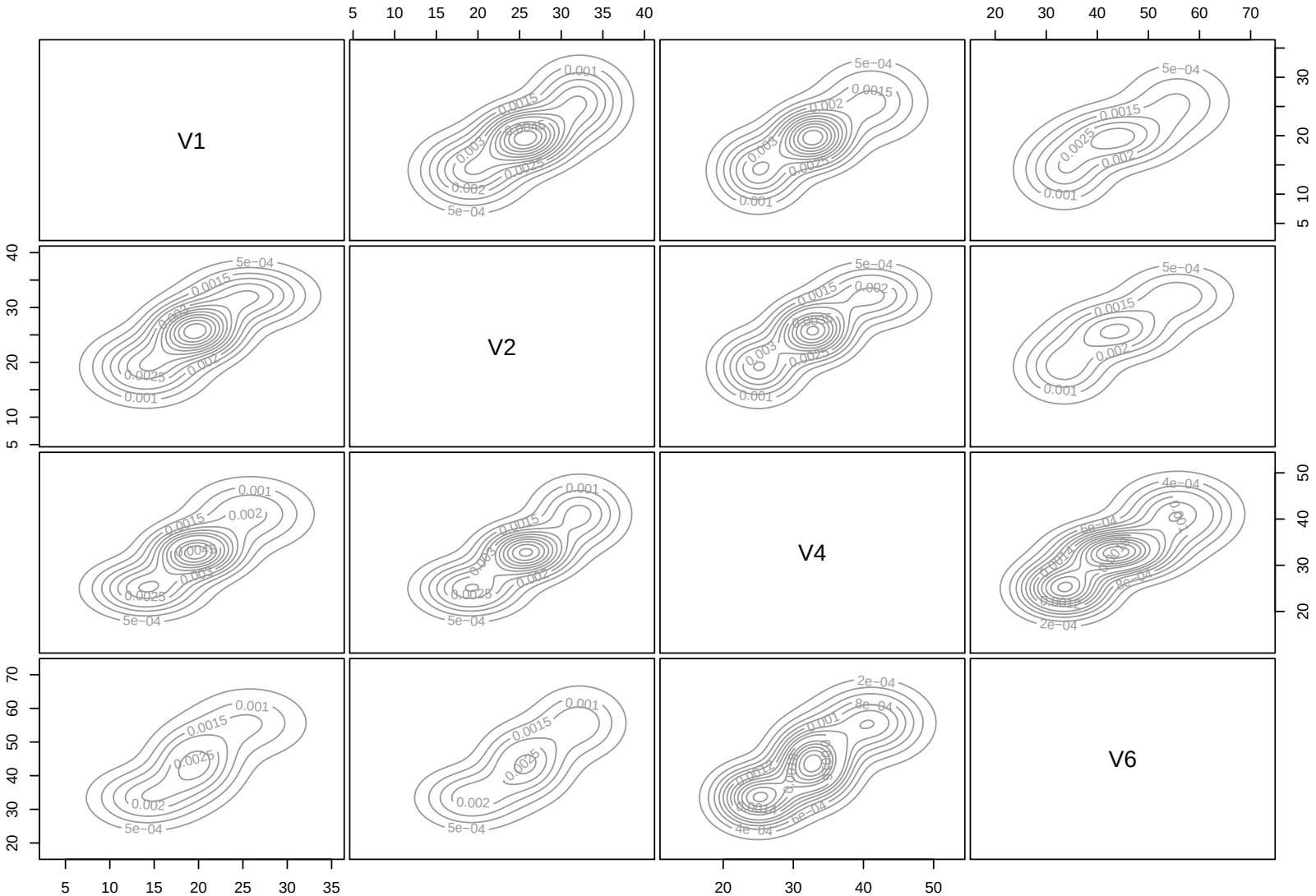
```
## assume same within class variance
Mix.1 = Mclust(wisc[,1:4], G = 1:9,modelNames=c("VVI"))

summary(Mix.1, parameters=TRUE)
```

```
## ----------------------------------------------------
## Gaussian finite mixture model fitted by EM algorithm
## ----------------------------------------------------
##
## Mclust VVI (diagonal, varying volume and shape) model with 3 components:
##
##  log.likelihood   n df       BIC       ICL
##       -2512.103 204 26 -5162.476 -5192.434
##
## Clustering table:
##  1  2  3
## 84 63 57
##
## Mixing probabilities:
##         1         2         3
## 0.4116910 0.3143575 0.2739515
##
## Means:
##        [,1]     [,2]     [,3]
## V1 19.67358 14.06274 25.78869
## V2 25.73517 19.10639 32.17316
```

Plot density

```
plot(Mix.1,"density")
```



Mixtools

```
mix.3 = mvnormalmixEM(wisc[,1:4], lambda = c(.3, .3, .4), k = 3,
                      epsilon=1e-3,arbmean=TRUE, arbvar=TRUE)
```

```
## number of iterations= 64
```

```
summary(mix.3)
```

```
## Warning in rbind(x$lambda, matrix(unlist(x$mu), byrow = TRUE, nrow =
## length(x$lambda))): number of columns of result is not a multiple of vector
## length (arg 1)

## summary of mvnormalmixEM object:
##           comp 1    comp 2    comp 3    comp 4
## lambda  0.425731  0.126644  0.447624  0.425731
## mu1    15.732879 20.610825 26.876220 35.922611
## mu2    19.558280 24.512715 33.377798 38.836696
## mu3    23.256302 30.239624 37.840417 52.584330
## loglik at estimate:  -2452.851
```

Compare to Mothers Education

```
head(round(mix.3$posterior,3),3)
```

```
##      comp.1 comp.2 comp.3
## [1,]  0.022      0  0.978
## [2,]  0.996      0  0.004
## [3,]  0.043      0  0.957
```

```
max <- apply(mix.3$posterior, 1, max)


class = which(mix.3$posterior == max,arr.ind=T)
class2 <- rep(NA,nrow(wisc))
for(i in 1:nrow(wisc)){
  class2[i] <- class[class[,1] == i,2]
}

table(class2,wisc$Moeducat+1)
```

```
##
## class2  1  2  3
##      1 57 16  6
##      2  7 20  5
##      3 12 46 35
```

Two class

```
mix.2 = mvnormalmixEM(wisc[,1:4], k=2,epsilon=1e-3)
```

```
## number of iterations= 92
```

```
mix.2$loglik
```

```
## [1] -2460.975
```

```
mix.3$loglik
```

```
## [1] -2452.851
```

Four class

```
mix.4 = mvnormalmixEM(wisc[,1:4], k=4,epsilon=1e-3)
```

```
## number of iterations= 91
```

```
mix.4$loglik
```

```
## [1] -2428.218
```

```
mix.2$loglik
```

```
## [1] -2460.975
```

```
mix.3$loglik
```

```
## [1] -2452.851
```

Longitudinal Clustering

Longitudinal Clustering

Although we can use regular clustering package to create classes with longitudinal data, it is best to use procedures that take into account the dependency of the data. This usually invloves changing the distance criterion.

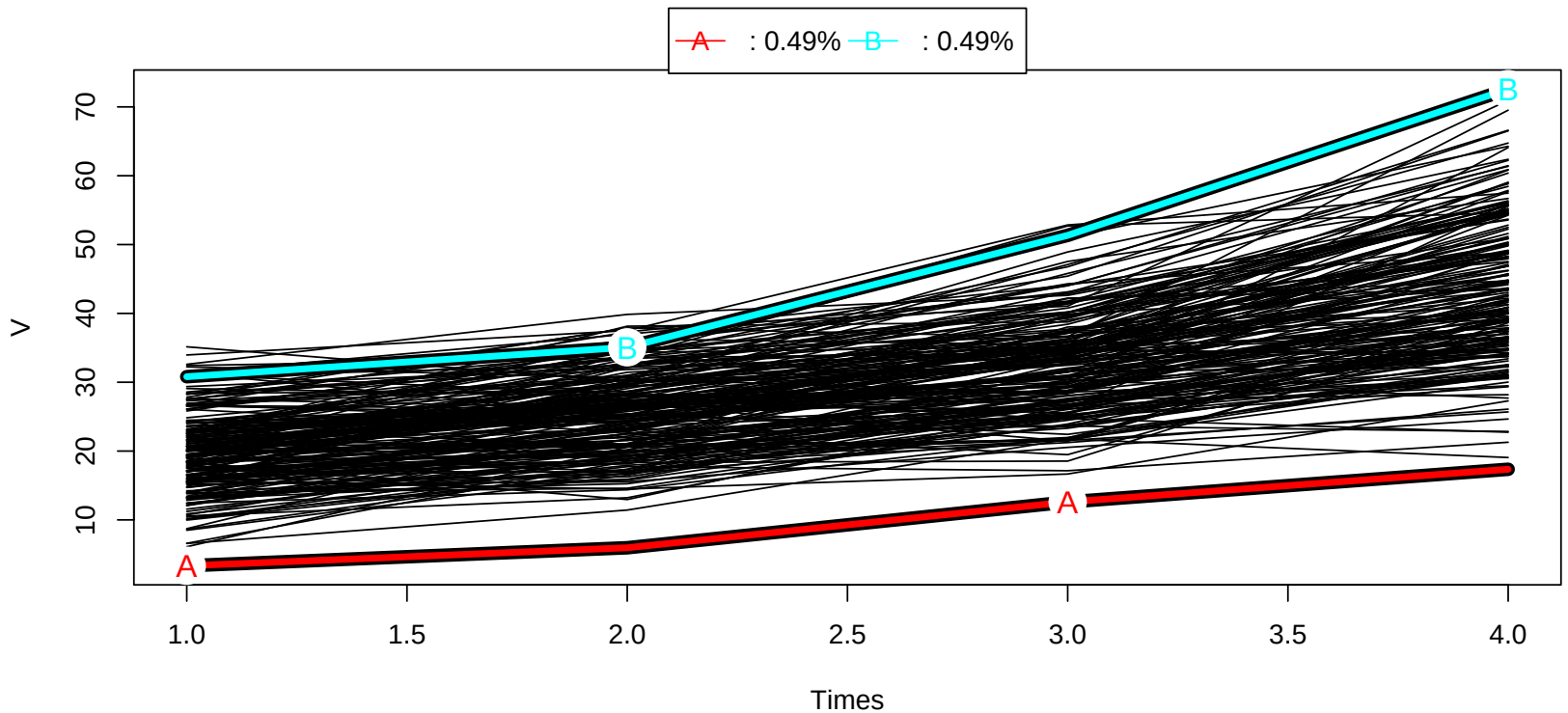
There are a number of packages in R to do longitudinal clustering, we will just look at two: kml and traj

kml package

Vignette: http://christophe.genolini.free.fr/recherche/aTelecharger/genolini2011.pdf

```
library(kml)
cld.wisc <- clusterLongData(wisc[,1:4],rep(1:nrow(wisc)))
kml(cld.wisc,nbClusters=2:4,toPlot="traj")
```

```
##  ~ Slow KmL ~
```




```
## *
```

traj package

Vignette:
https://cran.r-project.org/web/packages/traj/vignettes/trajVignette.pdf

```
library(traj)
time = matrix(rep(c(1,2,4,6),nrow(wisc)),nrow(wisc),4,byrow=TRUE)
# computes all distance measures
s1 = step1measures(wisc[,1:4],time)
```

```
## [1] "Correlation of m5 and m6 : 1"
## [1] "Correlation of m11 and m15 : 1"
```

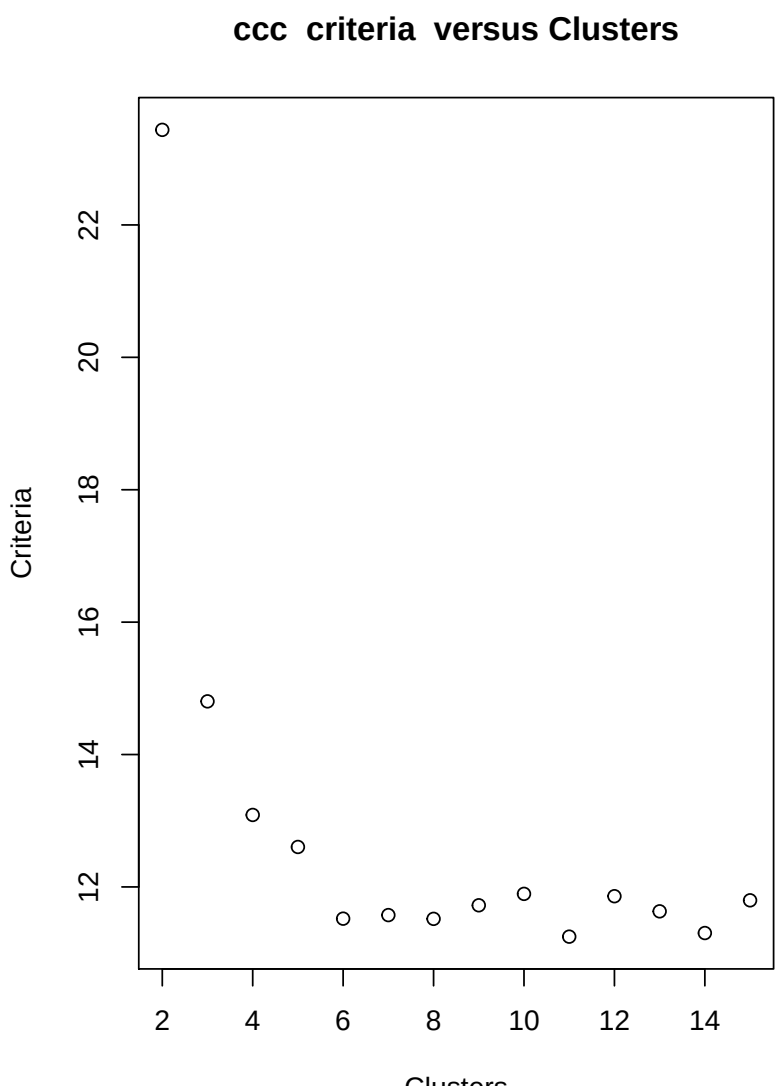
```
# dimension reduction of all measures
s2 = step2factors(s1)
```

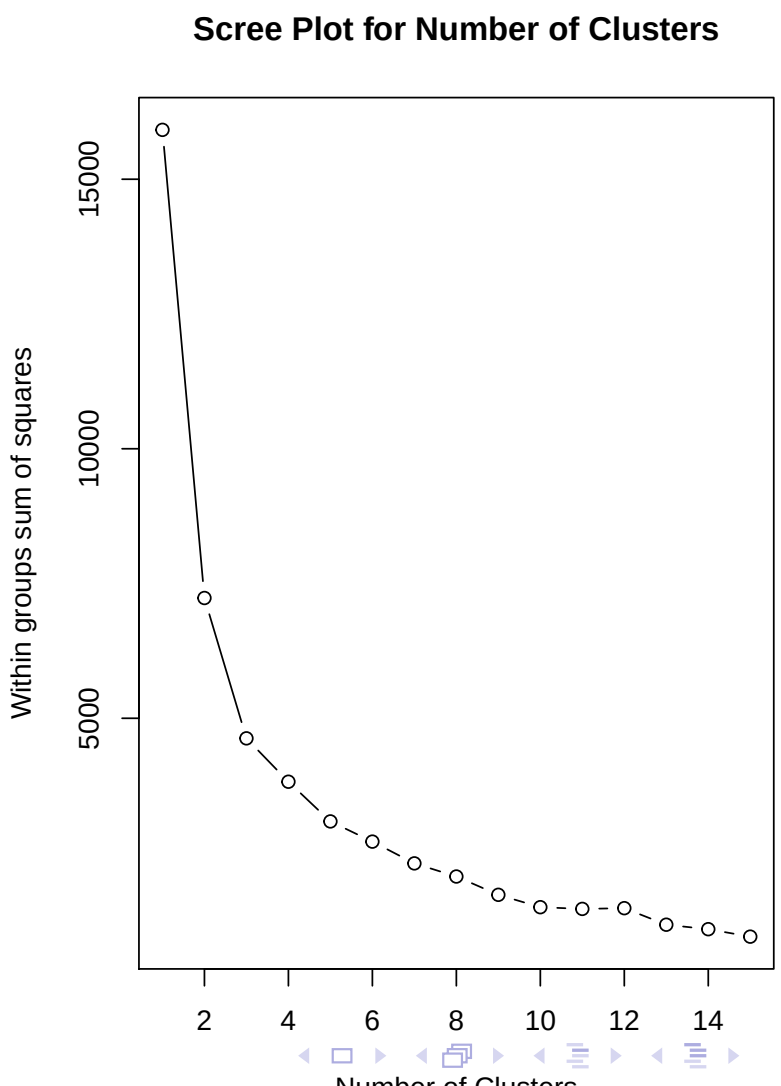
```
## [1] "m6 is removed because it is perfectly correlated with m5"
## [2] "m15 is removed because it is perfectly correlated with m11"
## [1] "Computing reduced correlation e-values..."
```

traj Continued

Create clusters

```
s3 = step3clusters(s2)
```





Plot Sample Trajectories

```
plot(s3)
```