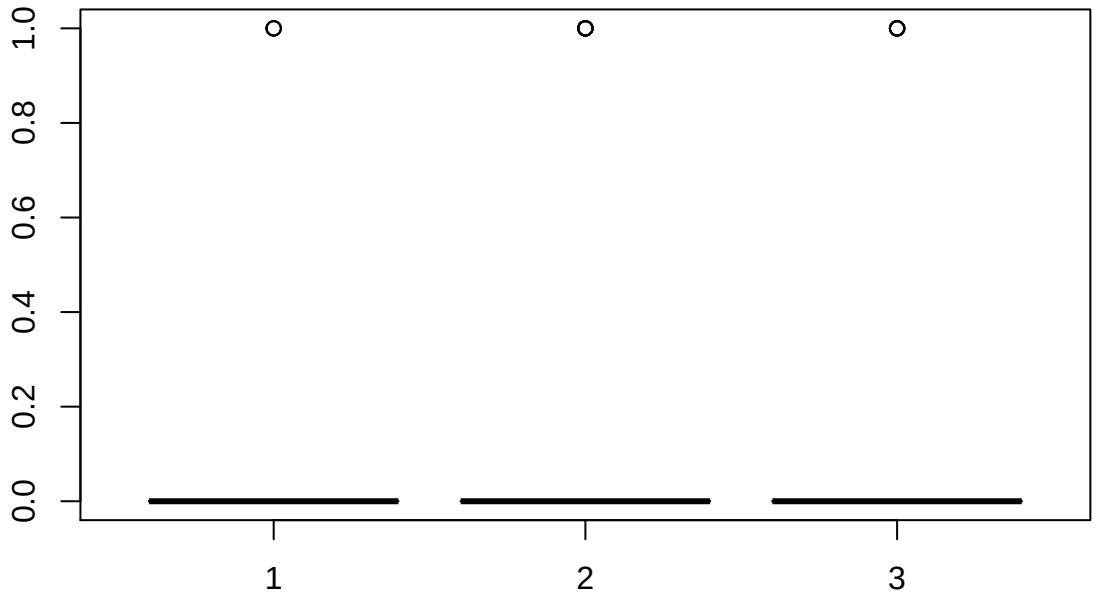
Supervised learning, Learnin in high dimention

Supervised learning

The general procedure

let's condider some classification problem with the iris data

```
library(MASS);library(class); library(e1071)
x=iris[,-5]
y=iris[,5]
E=matrix(NA,3,nrow(x))
# I would like to compare several classification technique (LDA, KNN, SVM)
#use leave one out method
for (b in 1:nrow(x)){
  xl=x[-b,];yl=y[-b]
  xv=x[b,];yv=y[b] #leave one out
  #LDA
  y1=predict(lda(xl,yl),xv)
  #KNN
  y2=knn(xl,xv,yl)
  #SVM
  y3=predict(svm(xl,yl),xv)

  E[1,b]= as.numeric(y1$class != yv)
  E[2,b]= as.numeric(y2 != yv)
  E[3,b]= as.numeric(y3 != yv)
}

#pick the most appropriate method
boxplot(t(E))
```




```r
apply(E,1,mean)
```

```
## [1] 0.02000000 0.04000000 0.03333333
```

```r
apply(E,1,sd)
```

```
## [1] 0.1404690 0.1966157 0.1801069
```

```r
#smallest mean and smallest sd, lda is the best method

#my final pick lda
clf=lda(x,y)

#in the future when i have new observation
xstar=c(0.5, 8,56,4)
predict(clf,xstar)
```

```
## $class
## [1] virginica
## Levels: setosa versicolor virginica
##
## $posterior
##      setosa    versicolor virginica
## [1,]      0 9.493336e-184         1
```

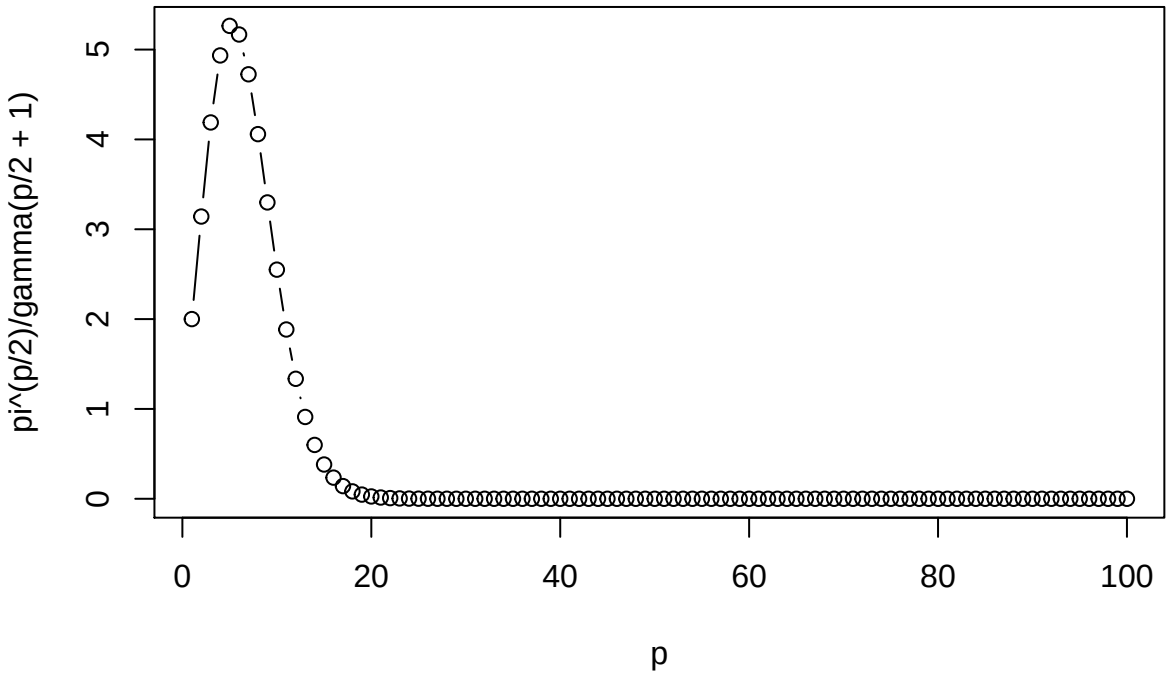
```
##
## $x
##            LD1       LD2
## [1,] -119.7141 -30.16409
```

Learning in high-dimensional spaces

The curs of dimensionality

The volume of the hyper-sphere is $V(p) = \frac{\pi^{p/2}}{\Gamma(p/2+1)}$.

```
p = 1:100
plot(p,pi^(p/2) / gamma(p/2+1), type='b')
```



High-dimensional data clustering (HDDC)

HDDC is implemented in the `HDclassif` library (which also pˆroposes the HDDA method for classification).

```
#install.packages('HDclassif')
library(HDclassif)
```

```
## Warning: package 'HDclassif' was built under R version 3.6.1
```

```
#?hddc
```

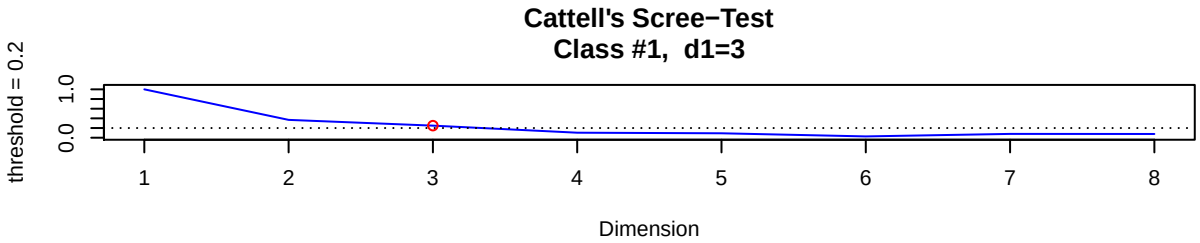
We may try HHDC on the wine data set:

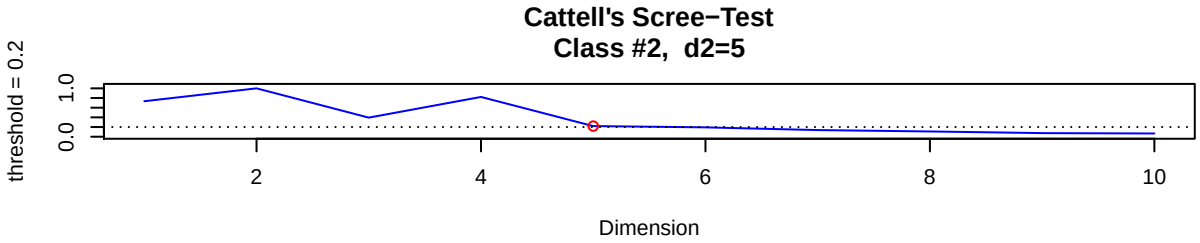
```
data(wine)
X = scale(wine[,-1])
Y = wine$class

out = hddc(X,K=1:10)#,threshold = (1:10)/100)
```

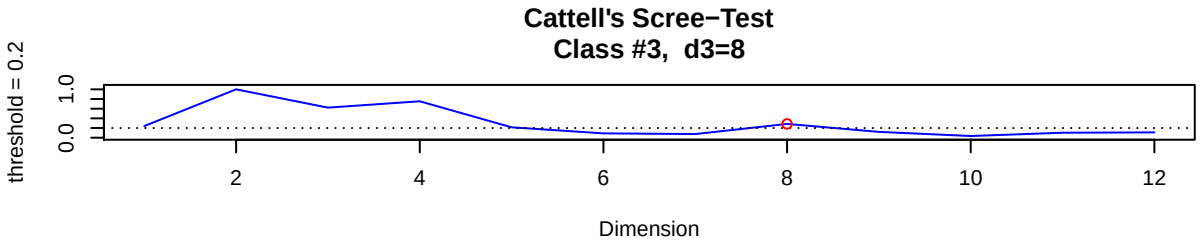
```
## Warning in hddc_main(model = model, K = K, threshold = threshold, ...):
## Maximum iterations reached (200).
```

```
## HDDC:
##        model  K threshold       BIC
## 1  AKJBKQKDK  3       0.2 -5,608.69
## 2  AKJBKQKDK  2       0.2 -5,637.93
## 3  AKJBKQKDK  4       0.2 -5,731.81
## 4  AKJBKQKDK  5       0.2 -5,802.62
## 5  AKJBKQKDK  1       0.2 -5,841.07
## 6  AKJBKQKDK  6       0.2 -6,005.42
## 7  AKJBKQKDK  7       0.2 -6,175.51
## 8  AKJBKQKDK  8       0.2 -6,475.91
## 9  AKJBKQKDK  9       0.2 -6,655.57
## 10 AKJBKQKDK 10       0.2 -6,776.24
##
## SELECTED: model  AKJBKQKDK  with  3  clusters.
## Selection Criterion: BIC.
```

```
plot(out)
```








```
out$d
```

```
##       Intrinsic dimensions of the classes:
##        1 2 3
##   dim: 3 5 8
```

```
table(out$cl,Y)
```

```
##    Y
##      1  2  3
##   1 59  2  0
##   2  0  0 48
##   3  0 69  0
```

Comparison with Mclust

```
library(mclust)
```

```
## Package 'mclust' version 5.4.3
## Type 'citation("mclust")' for citing this R package in publications.
```

```
out1 = hddc(X,K=3)
```

```
##       model K threshold       BIC
## 1 AKJBKQKDK 3       0.2 -5,608.69
##
## SELECTED: model  AKJBKQKDK  with  3  clusters.
## Selection Criterion: BIC.
```

```
table(out1$cl,Y)
```

```
##    Y
##      1  2  3
##   1  0 69  0
##   2  0  0 48
##   3 59  2  0
```

```
out2 = Mclust(X,G=3)
table(out2$cl,Y)
```

```
##    Y
##      1  2  3
##   1 56  0  0
##   2  3 70  0
##   3  0  1 48
```

Let's now move to higher dimensions:

```
#install.packages('MBCbook')
library(MBCbook)
```

```
## Warning: package 'MBCbook' was built under R version 3.6.1
```

```
## Loading required package: Rmixmod
```

```
## Loading required package: Rcpp
```

```
## Rmixmod v. 2.1.2.2 / URI: www.mixmod.org
```

```
## Loading required package: mvtnorm
```

```
data("usps358")
X = usps358[,-1]
Y = usps358$cls

system.time(out1 <- hddc(X,K=3))
```

```
##       model K threshold         BIC
## 1 AKJBKQKDK 3       0.2 -613,317.19
##
## SELECTED: model  AKJBKQKDK  with  3  clusters.
## Selection Criterion: BIC.
```

```
##    user  system elapsed
##   10.06    0.86   11.30
```

```r
table(out1$cl,Y)
```

```
##    Y
##       1   2   3
##   1 596   5  24
##   2  43 549  26
##   3  19   2 492
```

```r
#system.time(out2 <- Mclust(X,G=3,modelNames = 'VVV'))
#table(out2$cl,Y)
```

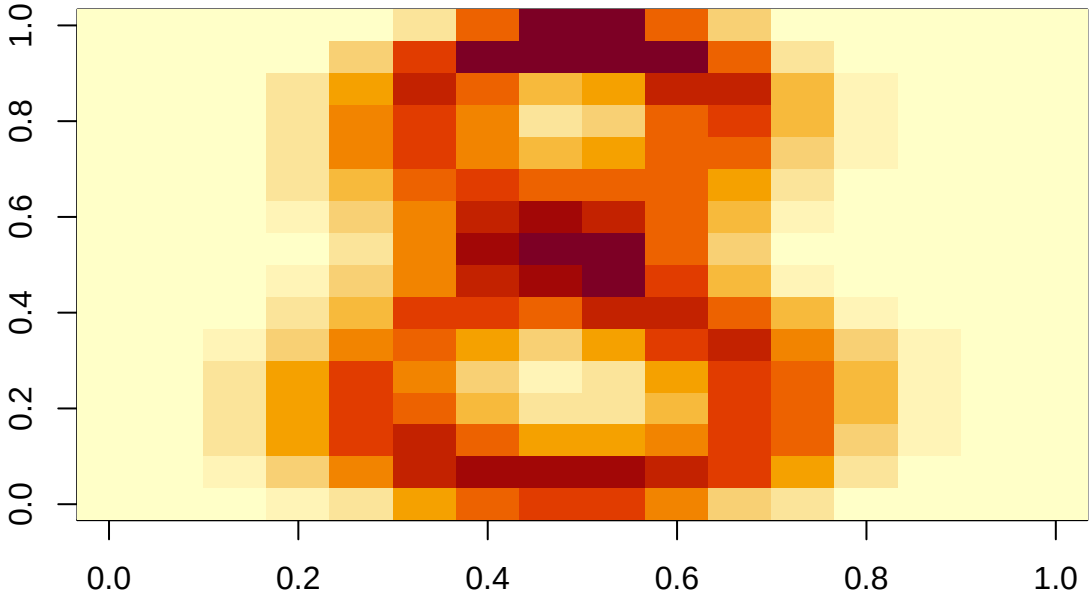
```r
library(MBCbook)
library(HDclassif)
data("usps358")
X = usps358[,-1]
Y = usps358$cls

out1 = hddc(X,K=3)
```

```
##       model K threshold         BIC
## 1 AKJBKQKDK 3       0.2 -613,317.19
##
## SELECTED: model  AKJBKQKDK  with  3  clusters.
## Selection Criterion: BIC.
```

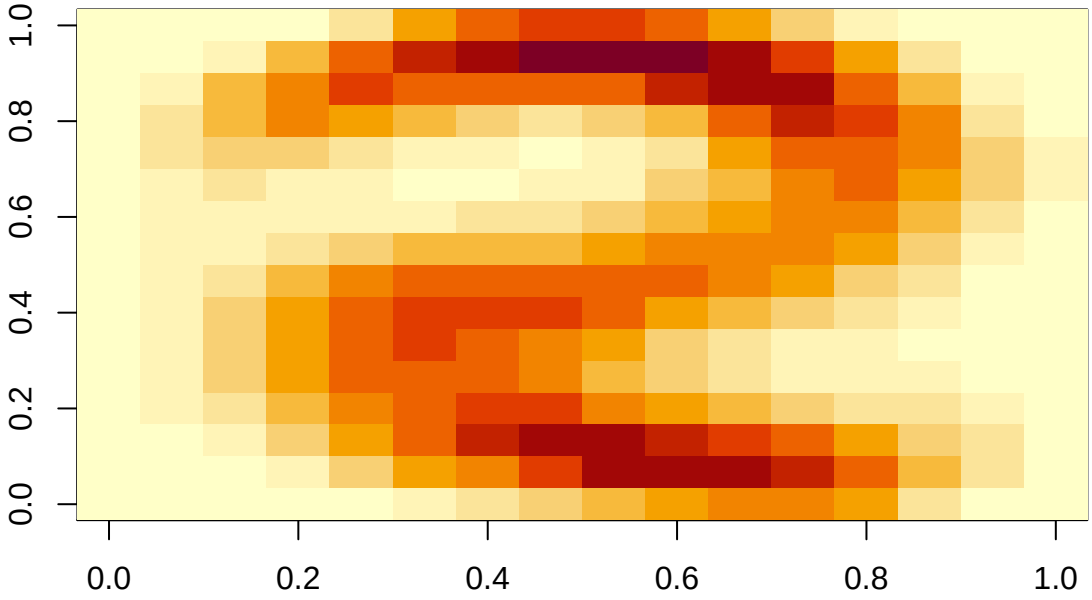
```r
table(out1$cl,Y)
```

```
##    Y
##       1   2   3
##   1  19   2 492
##   2  43 549  26
##   3 596   5  24
```

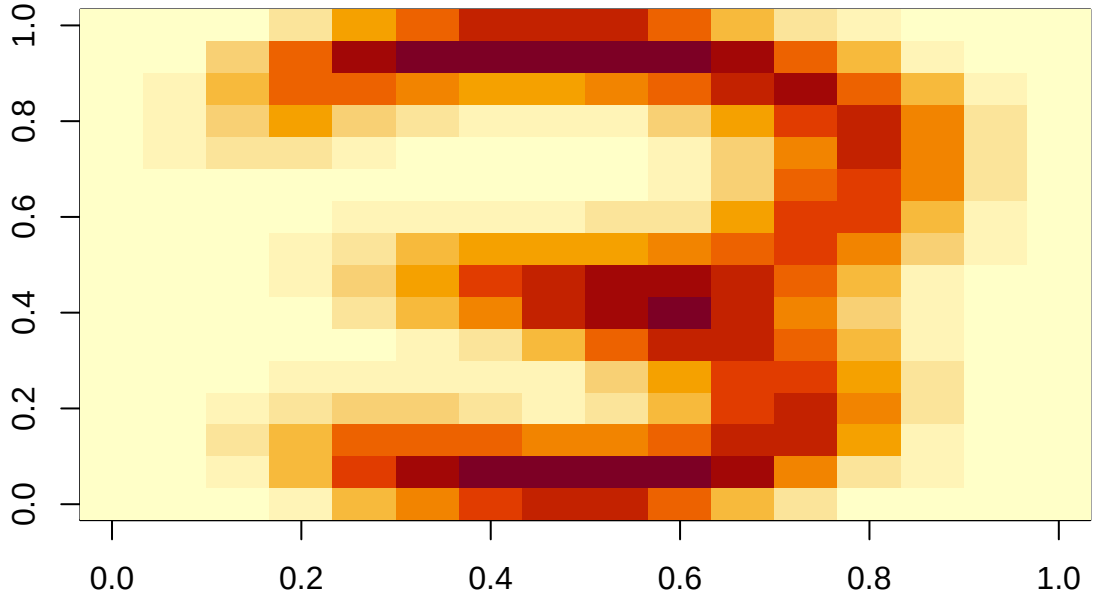
```r
image(matrix(out1$mu[1,],ncol=16))
```




```
image(matrix(out1$mu[2,],ncol=16))
```




```
image(matrix(out1$mu[3,],ncol=16))
```



Fisher EM

```
#install.packages('FisherEM')
library(FisherEM)
```

```
## Warning: package 'FisherEM' was built under R version 3.6.1
```

```
## Loading required package: parallel
```

```
## Loading required package: elasticnet
```

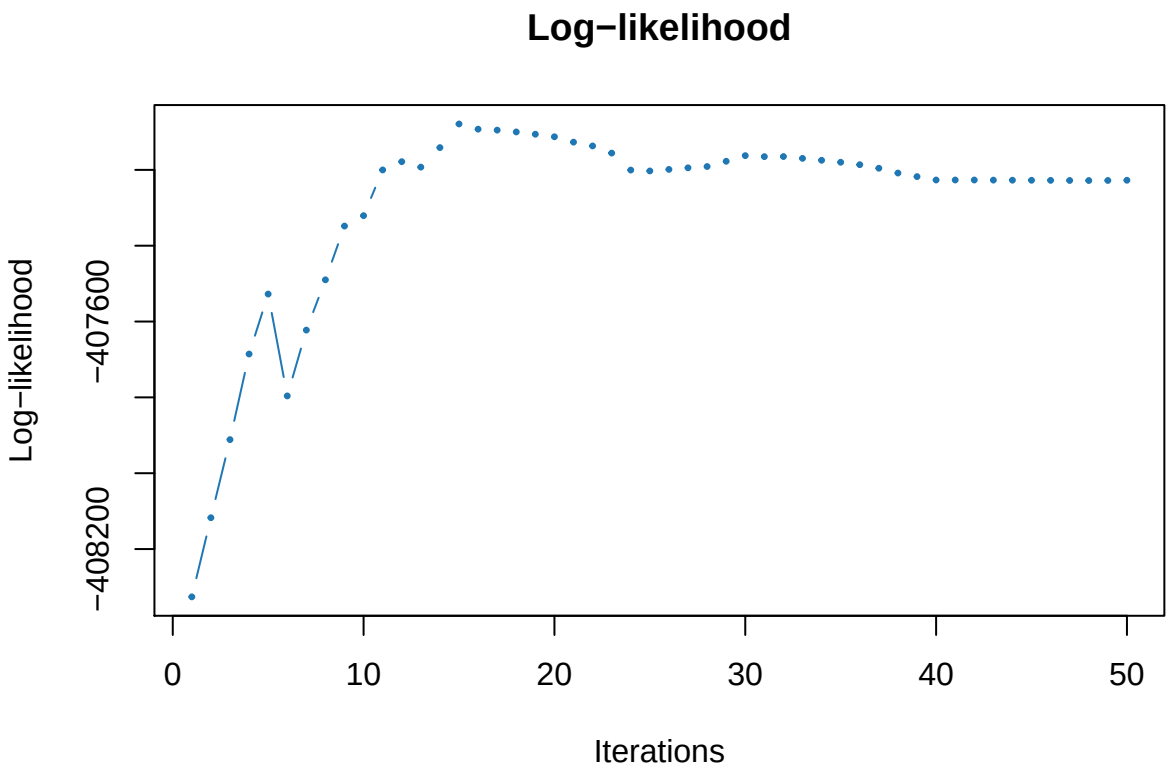
```
## Loading required package: lars
```

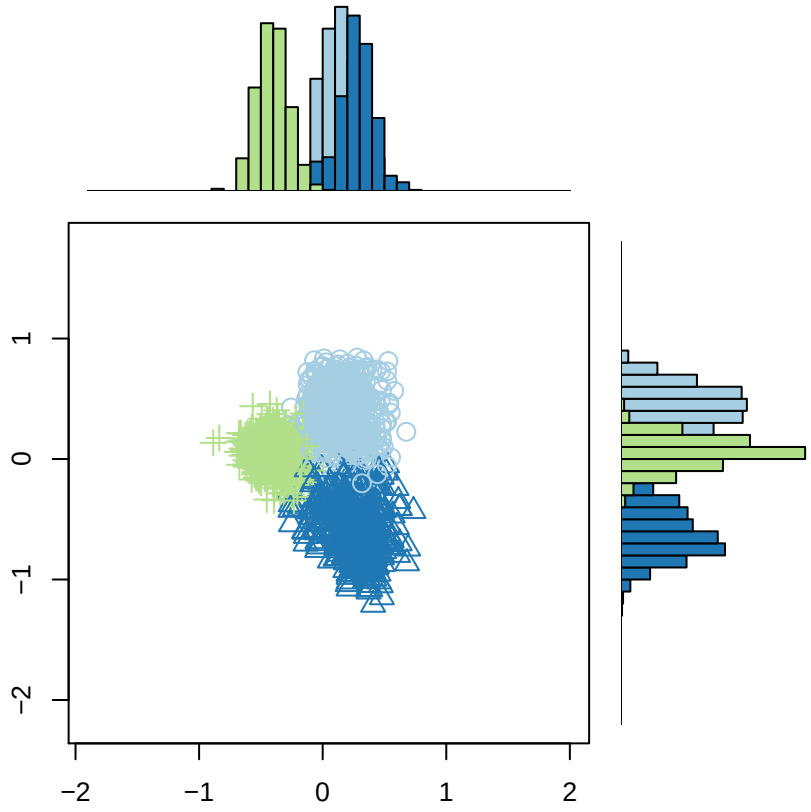
```
## Loaded lars 1.2
```

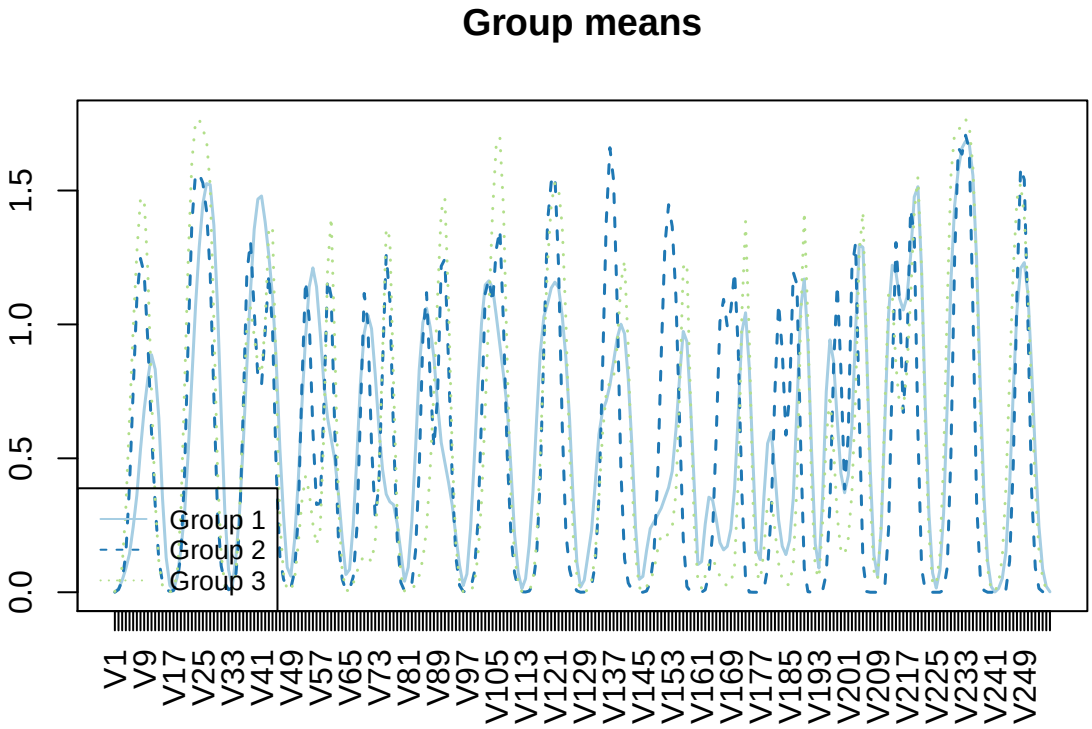
```
out=fem(X,K=3)
table(out$cl,Y)
```

```
##    Y
##       1   2   3
##   1  95 517  62
##   2  48  34 462
##   3 515   5  18
```

```
plot(out)
```


Log−likelihood
Log−likelihood
−407600
−408200
0
10
20
30
40
50
Iterations


−2
−1
0
1
2
−2
−1
0
1
2


Group means
0.0
0.5
1.0
1.5
Group 1
Group 2
Group 3
V1
V9
V17
V25
V33
V41
V49
V57
V65
V73
V81
V89
V97
V105
V113
V121
V129
V137
V145
V153
V161
V169
V177
V185
V193
V201
V209
V217
V225
V233
V241
V249