chp-clustering

Ar

2022-06-07

```
library("flowCore")
library("flowViz")
```

```
## Loading required package: lattice
```

```
fcsB = read.FCS("../data/Bendall_2011.fcs", truncate_max_range=FALSE)
slotNames(fcsB)
```

```
## [1] "exprs"       "parameters"  "description"
```

```
dim(Biobase::exprs(fcsB))
```

```
## [1] 91392    41
```

```
markersB = readr::read_csv("../data/Bendall_2011_markers.csv")
```

```
## Rows: 10 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): isotope, marker
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
mt = match(markersB$isotope, colnames(fcsB))
stopifnot(!any(is.na(mt)))
colnames(fcsB)[mt] = markersB$marker
```

```
markersB$marker
```

```
##  [1] "CD4"    "CD8"    "CD20"   "CD33"   "CD56"   "DNA191" "DNA192" "CD3"
##  [9] "CD45RA" "CD3all"
```

```
markersB
```

```
## # A tibble: 10 x 2
##    isotope             marker
##    <chr>               <chr>
##  1 Nd(144.912)-Dual    CD4
##  2 Nd(145.913)-Dual    CD8
##  3 Sm(146.914)-Dual    CD20
##  4 Gd(157.924)-Dual    CD33
##  5 Er(169.935)-Dual    CD56
##  6 Ir(190.960)-Dual    DNA191
##  7 Ir(192.962)-Dual    DNA192
##  8 Cd(109.903)-Dual    CD3
##  9 La(138.906)-Dual    CD45RA
```

```
## 10 Cd(110,111,112,114) CD3all
```

```
match(c(1,2),c(1,1,1,2,2,2,1,1,1,2,2))
```

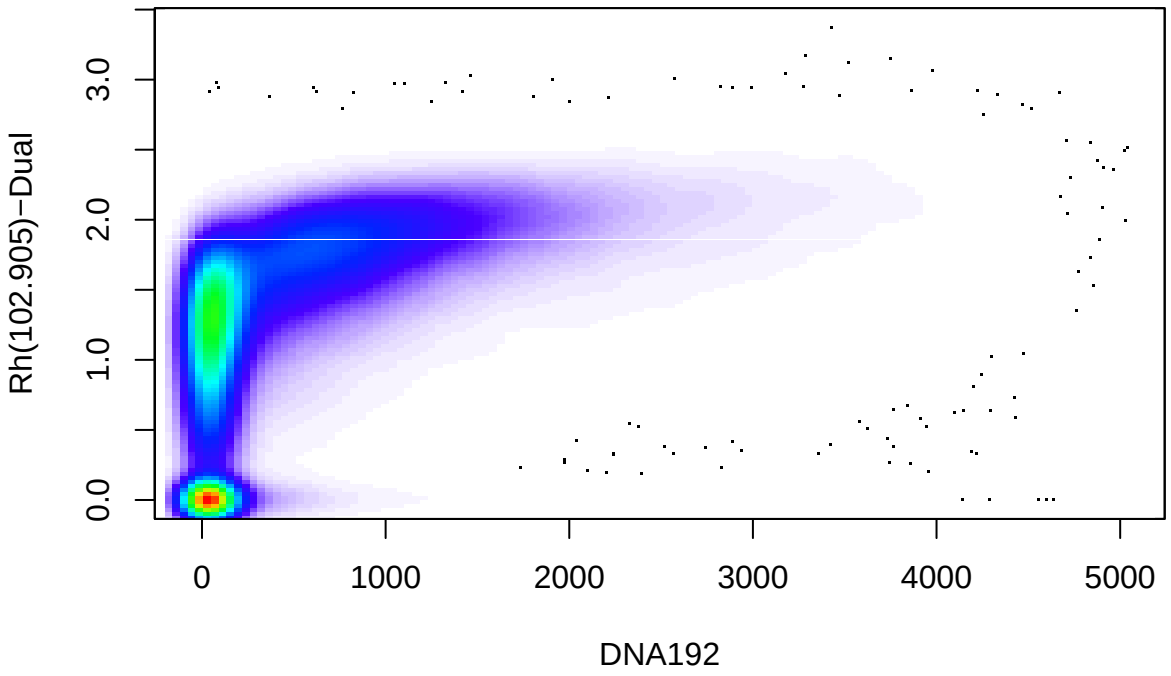
```
## [1] 1 4
```

```
?flowPlot
```

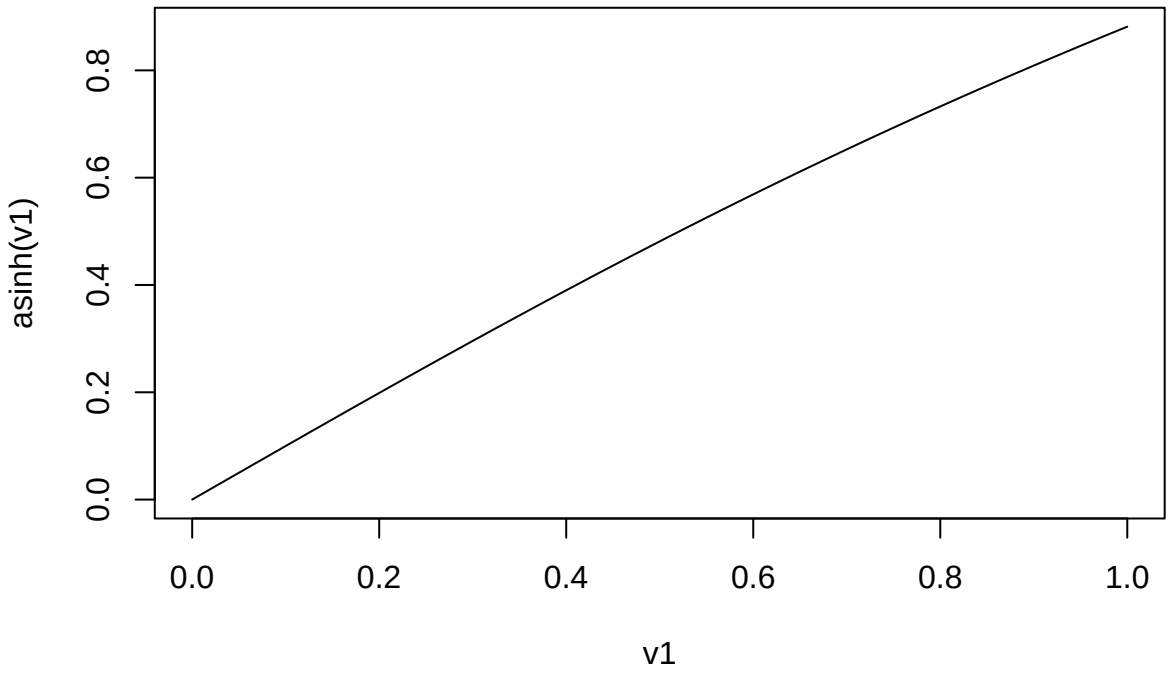
```
colnames(fcsB)[4]
```

```
## [1] "DNA192"
```

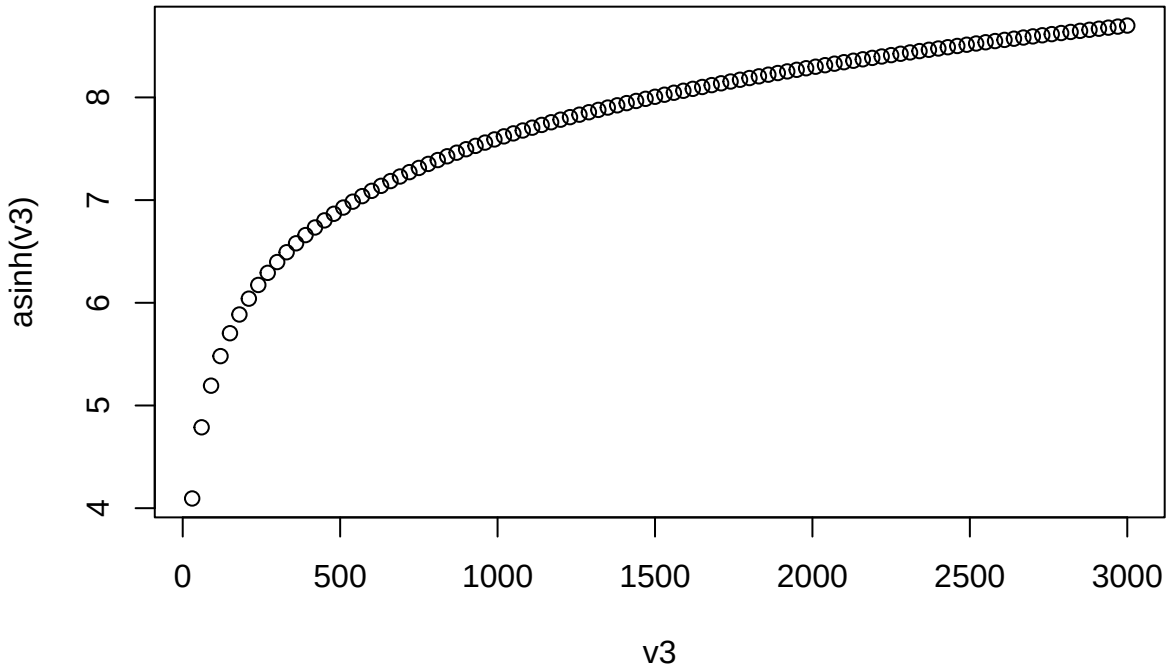
```
flowPlot(fcsB, plotParameters=colnames(fcsB)[4:7], logy=TRUE)
```




```
v1 = seq(0, 1, length.out = 100)
plot(v1, asinh(v1), type = 'l')
```




```
v3 = seq(30, 3000, length = 100)
plot(v3, asinh(v3))
```



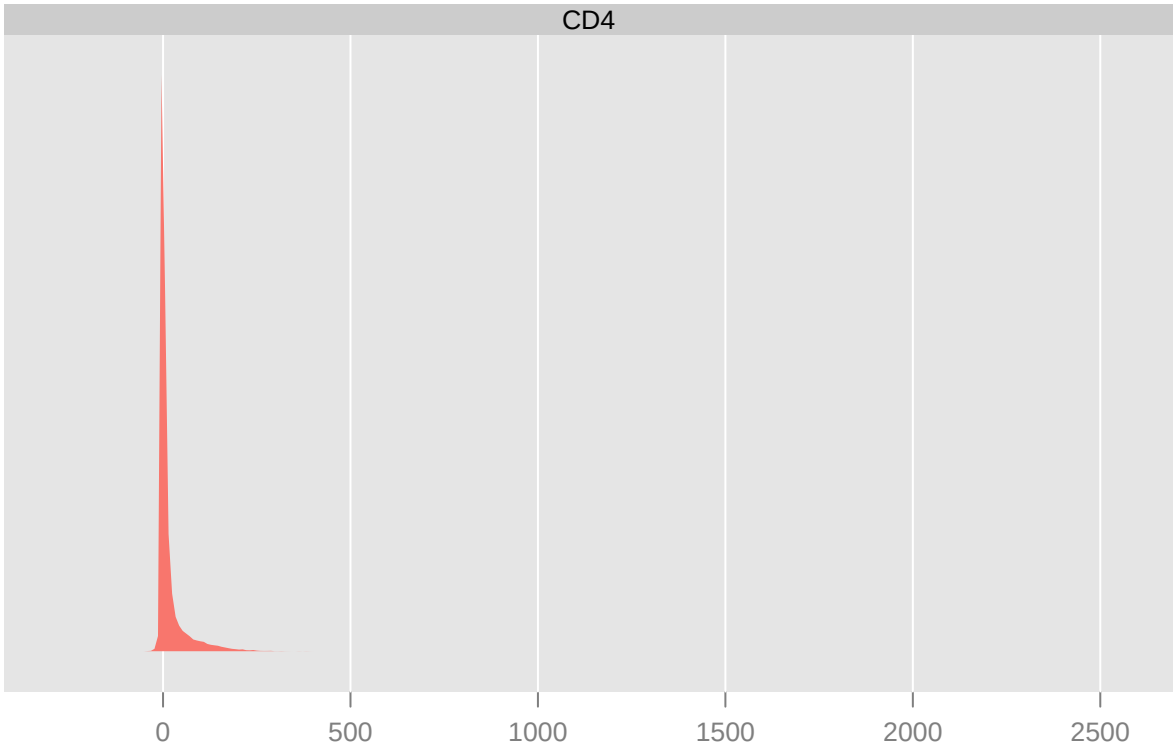

```
asinhtrsf = arcsinhTransform(a=0.1, b=1)
fcsBT = transform(fcsB,
                  transformList(colnames(fcsB)[-c(1,2,41)], asinhtrsf))
```

```
colnames(fcsB)
```

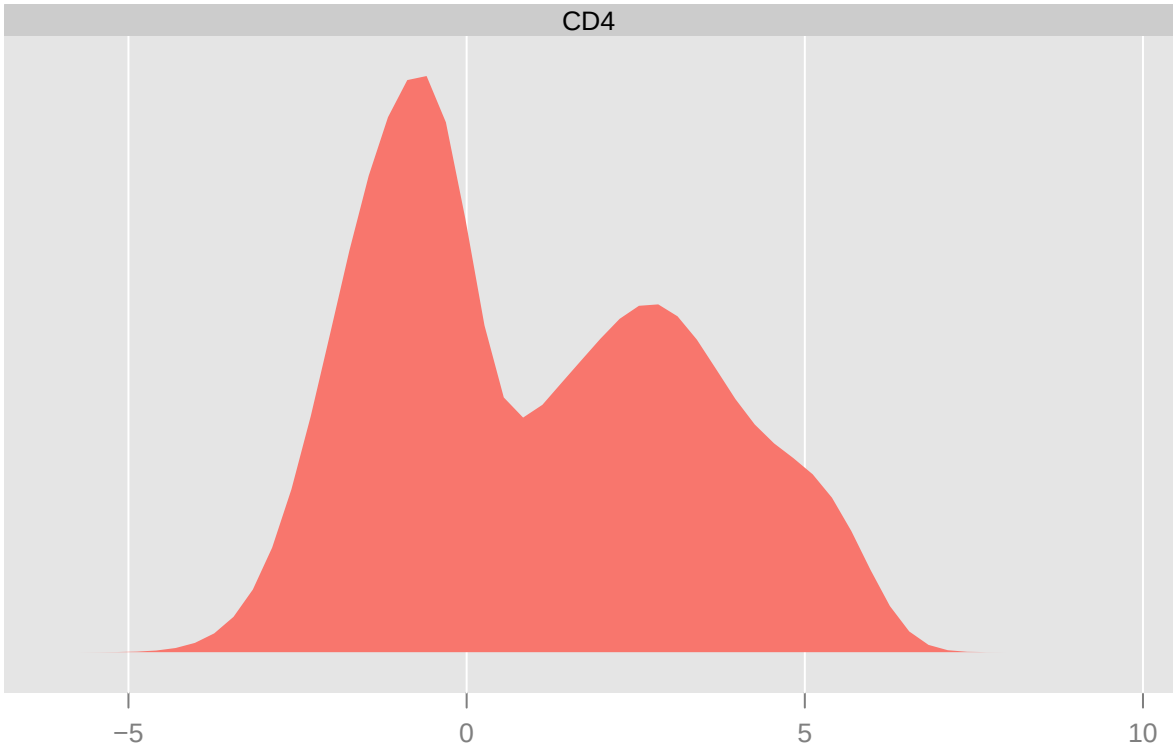
```
##  [1] "Time"                "Cell_length"         "DNA191"
##  [4] "DNA192"              "Rh(102.905)-Dual"    "In(114.903)-Dual"
##  [7] "CD3"                 "Cd(110.904)-Dual"    "Cd(111.902)-Dual"
```

```
## [10] "Cd(113.903)-Dual"    "CD45RA"              "Pr(140.907)-Dual"
## [13] "Nd(141.907)-Dual"    "Nd(143.910)-Dual"    "CD4"
## [16] "CD8"                 "Nd(147.916)-Dual"    "Nd(149.920)-Dual"
## [19] "CD20"                "Sm(151.919)-Dual"    "Sm(153.922)-Dual"
## [22] "Eu(150.919)-Dual"    "Eu(152.921)-Dual"    "Gd(155.922)-Dual"
## [25] "CD33"                "Gd(159.927)-Dual"    "Tb(158.925)-Dual"
## [28] "Dy(163.929)-Dual"    "Ho(164.930)-Dual"    "Er(165.930)-Dual"
## [31] "Er(166.932)-Dual"    "Er(167.932)-Dual"    "CD56"
## [34] "Tm(168.934)-Dual"    "Yb(170.936)-Dual"    "Yb(171.936)-Dual"
## [37] "Yb(173.938)-Dual"    "Yb(175.942)-Dual"    "Lu(174.940)-Dual"
## [40] "CD3all"              "absoluteEventNumber"
```

```
densityplot(~`CD4`, fcsB)
```




```
densityplot(~`CD4`, fcsBT)
```




```
?kmeansFilter
```

```
?filter
```

```
## Help on topic 'filter' was found in the following packages:
##
##   Package               Library
##   flowCore              /home/arun/R/x86_64-pc-linux-gnu-library/4.2
##   stats                 /usr/local/lib/R/library
##
##
## Using the first match ...
```

```
kf = flowCore::kmeansFilter("CD3all"=c("Pop1", "Pop2"), filterId = "myKmFilter")
fres = flowCore::filter(fcsBT, kf)

summary(fres)
```

```
## Pop1: 33434 of 91392 events (36.58%)
## Pop2: 57958 of 91392 events (63.42%)
```

```
fcsBT1 = flowCore::split(fcsBT, fres, population = "Pop1")
fcsBT2 = flowCore::split(fcsBT, fres, population = "Pop2")
```

```
?flowPeaks
```

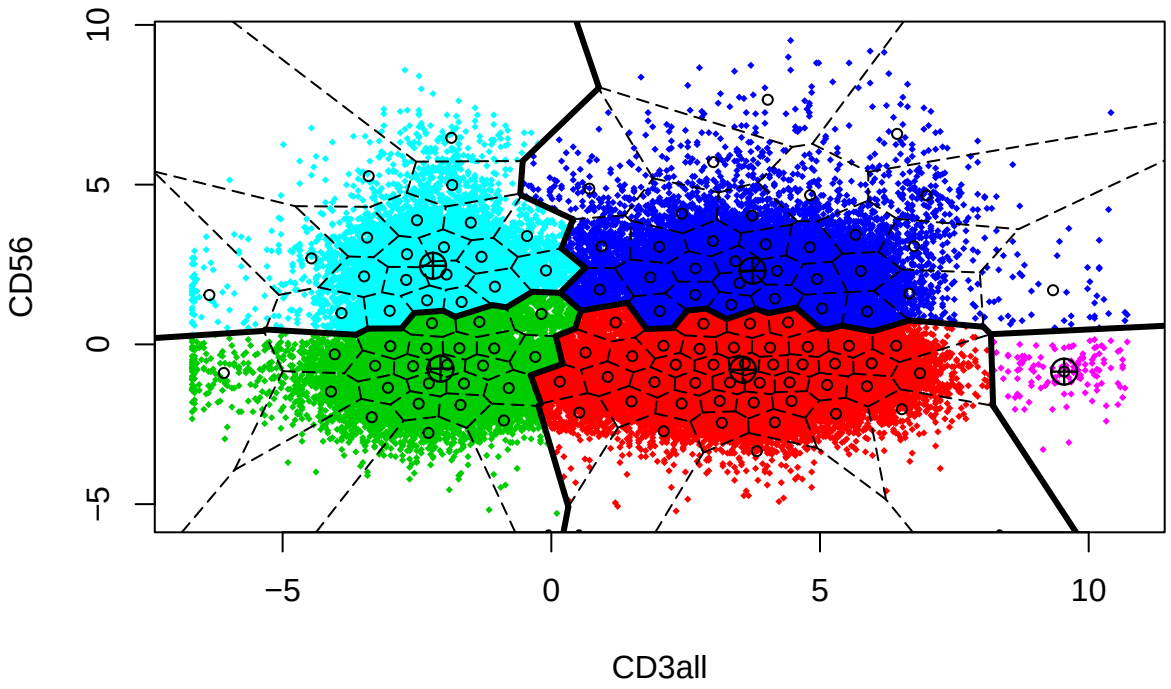
```
## No documentation for 'flowPeaks' in specified packages and libraries:
## you could try '??flowPeaks'
```

```
library("flowPeaks")
fp = flowPeaks(Biobase::exprs(fcsBT)[, c("CD3all", "CD56")])
```

```
##
## Starting the flow Peaks analysis...
```

```
##
##     Task A: compute kmeans...
##         step 0, set the intial seeds, tot.wss=14636.9
##         step 1, do the rough EM, tot.wss=10278.9 at 0.265059 sec
##         step 2, do the fine transfer of Hartigan-Wong Algorithm
##                 tot.wss=10221.1 at 0.524185 sec
##         ...finished summarization at 0.529 sec
##
##     Task B: find peaks...
## finished at 0.603 sec
```

```
plot(fp)
```



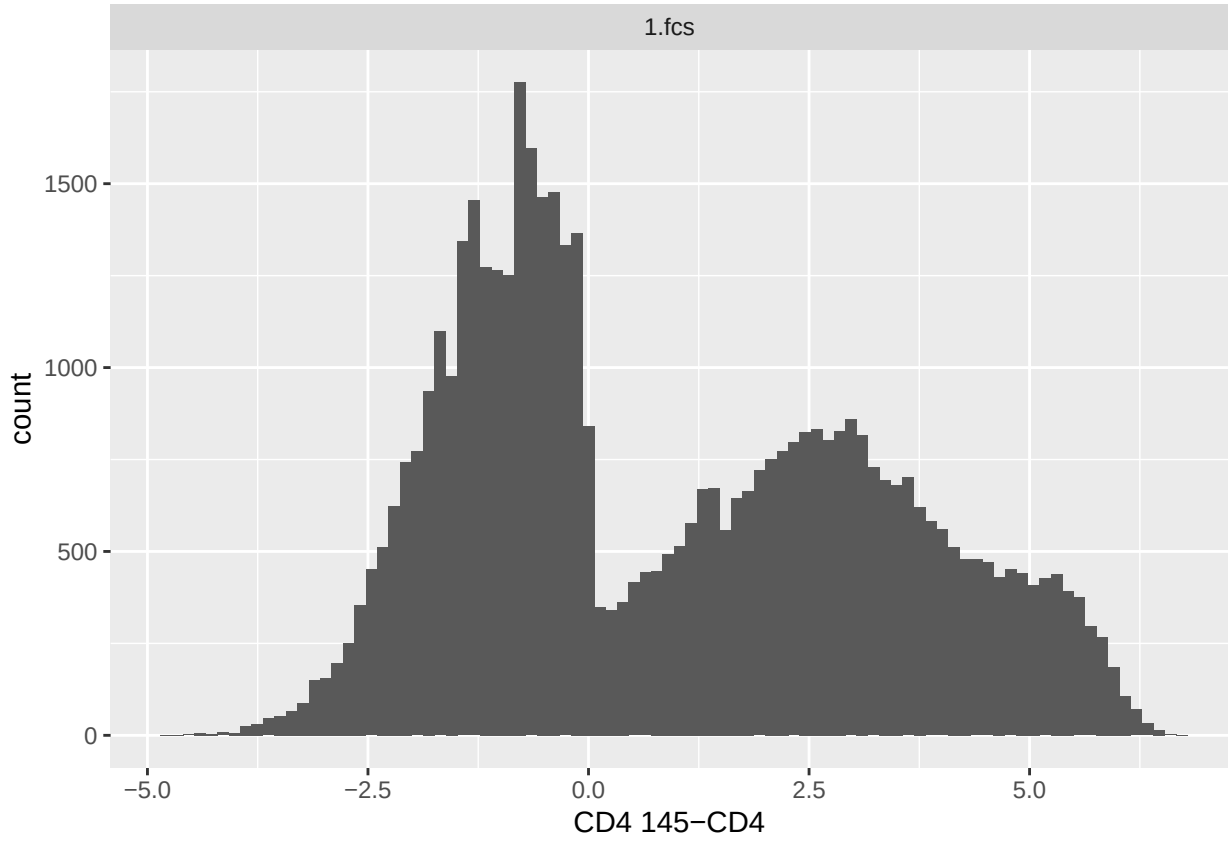

```
library("ggcyto")
```

```
## Loading required package: ggplot2
## Loading required package: ncdfFlow
## Loading required package: RcppArmadillo
## Loading required package: BH
## Loading required package: flowWorkspace
## As part of improvements to flowWorkspace, some behavior of
## GatingSet objects has changed. For details, please read the section
## titled "The cytoframe and cytoset classes" in the package vignette:
##
##   vignette("flowWorkspace-Introduction", "flowWorkspace")
```

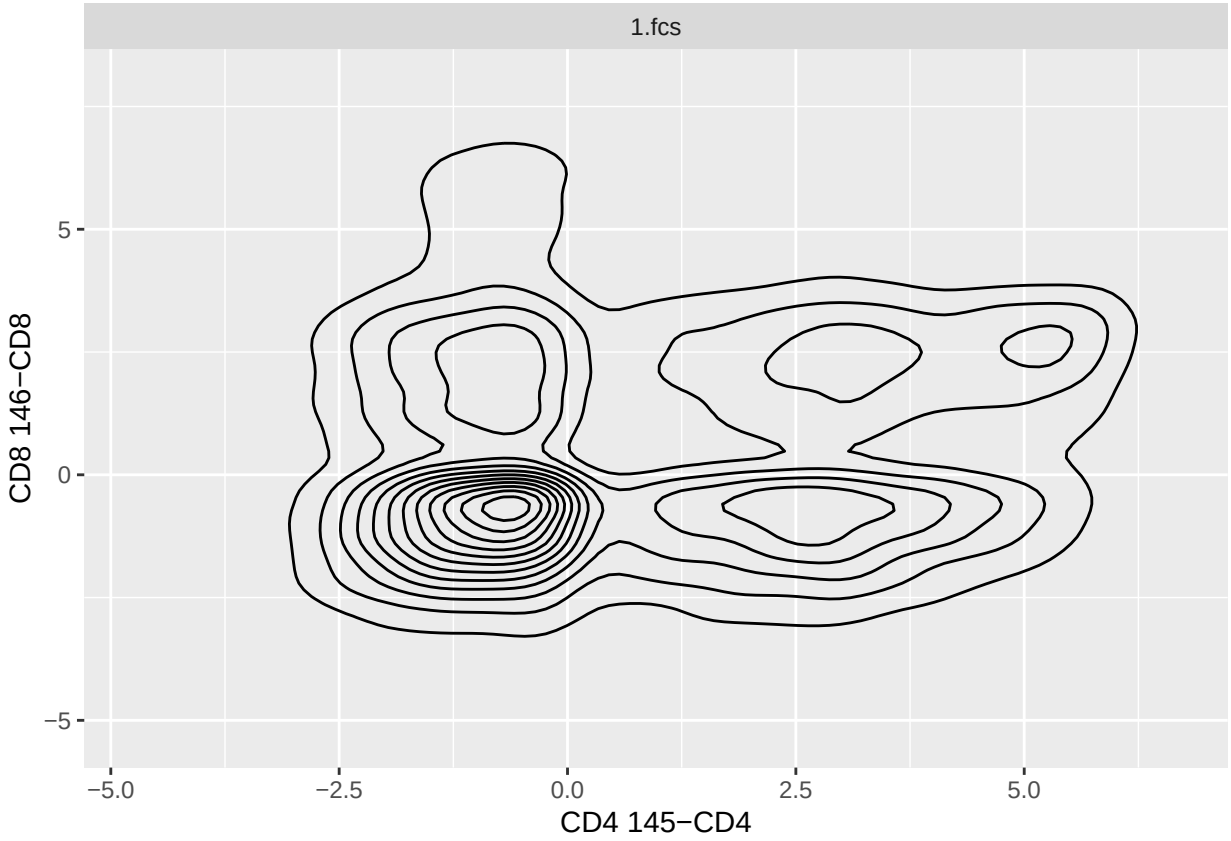
```
library("labeling")
ggcd4cd8 = ggcyto(fcsB, aes(x=CD4, y=CD8))
ggcd4 = ggcyto(fcsB, aes(x=CD4))
ggcd8 = ggcyto(fcsB, aes(x=CD8))
```

```
p1 = ggcd4 + geom_histogram(bins=60)
p1b = ggcd8 + geom_histogram(bins=60)
asinhT = arcsinhTransform(a=0, b=1)
trans1 = transformList(colnames(fcsB)[-c(1,2,41)], asinhT)
fcsBT = transform(fcsB, trans1)
p1t = ggcyto(fcsBT, aes(x=CD4))+geom_histogram(bins=90)
p2t = ggcyto(fcsBT, aes(x=CD4, y=CD8))+geom_density2d(colour="black")
p3t = ggcyto(fcsBT, aes(x=CD45RA, y=CD20))+geom_density2d(colour="black")
```

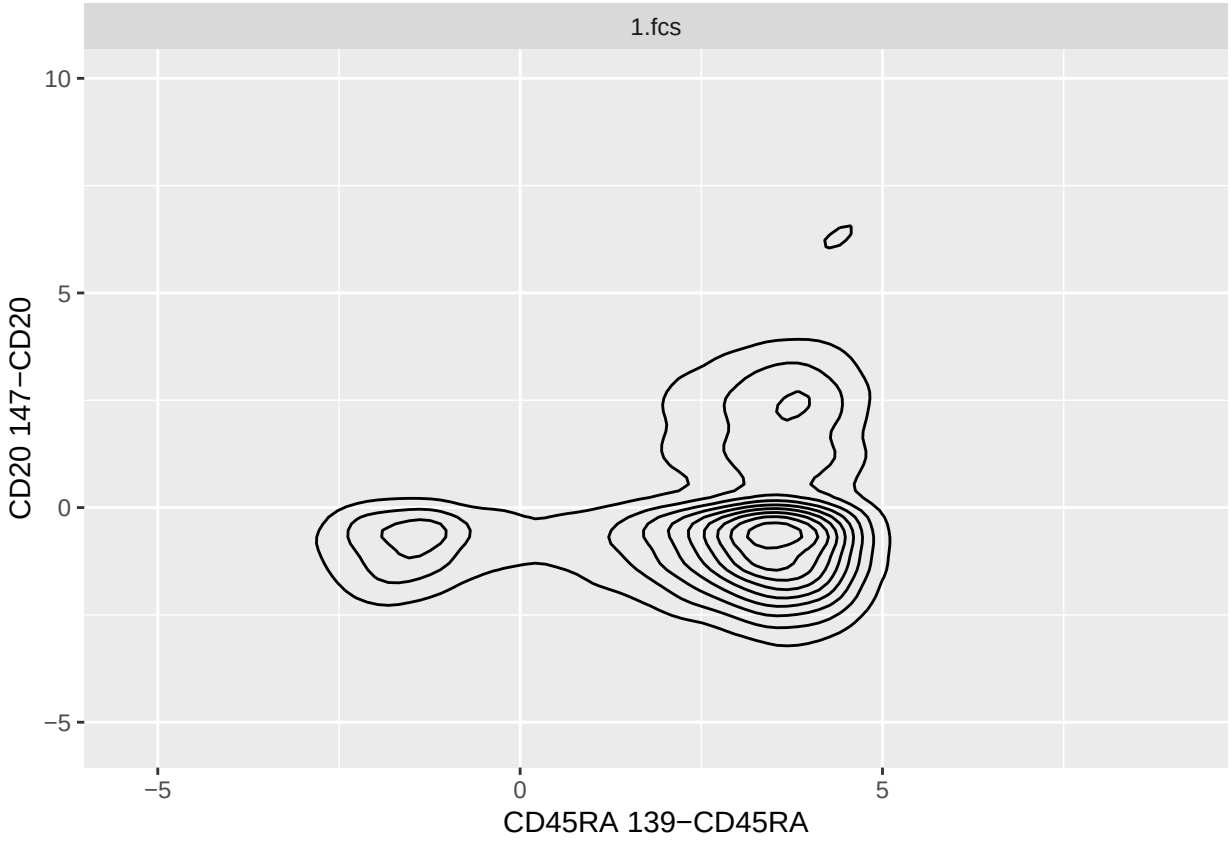
```
p1t
```




```
p2t
```




```
p3t
```




```
?dbscan
```

```
## No documentation for 'dbscan' in specified packages and libraries:
## you could try '??dbscan'
```

```
library("dbscan")

mc5 = Biobase::exprs(fcsBT)[,c(15,16,19,33, 40)]
res5 = dbscan::dbscan(mc5, eps=0.95, minPts=100)
mc5df = data.frame(mc5, cluster=as.factor(res5$cluster))
table(mc5df$cluster)
```

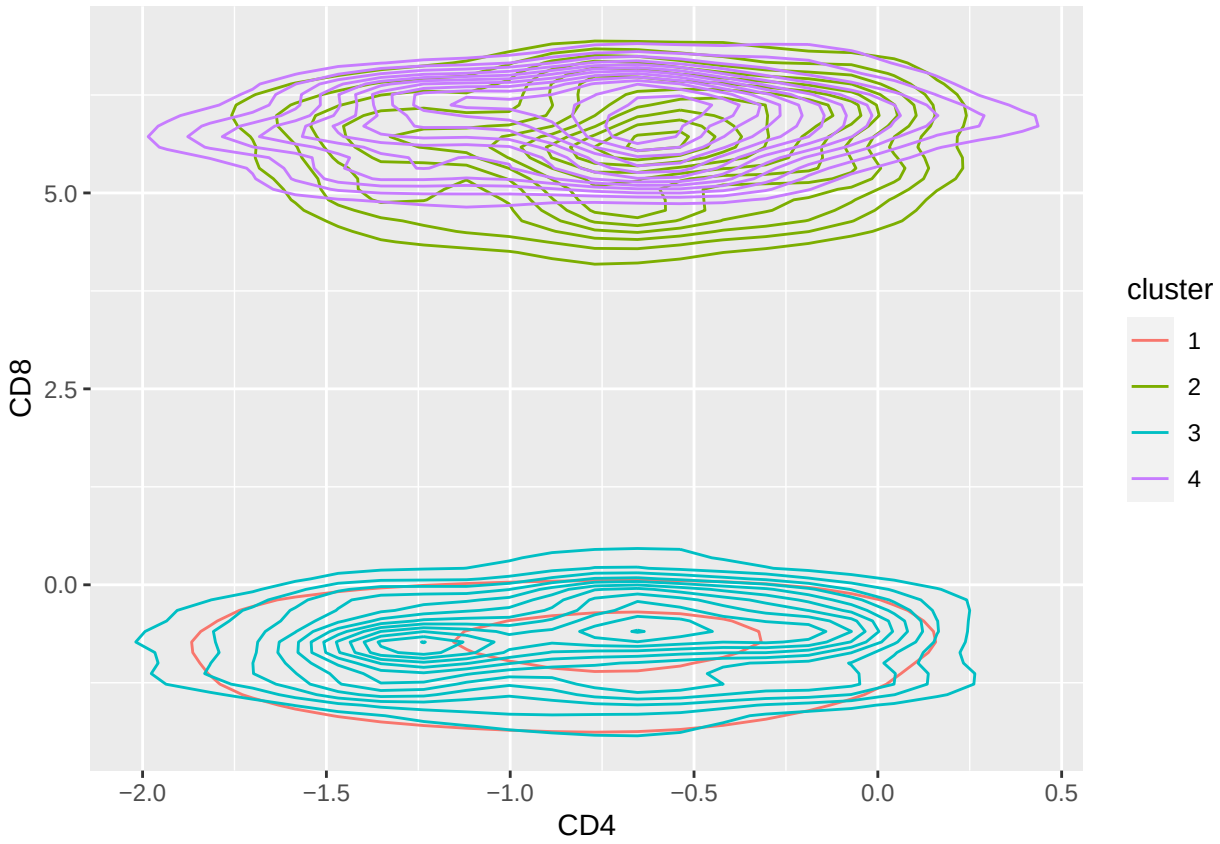
```
##
##     0     1     2     3     4
## 58988 31454   330   460   160
```

```
ggplot(mc5df, aes(x=CD4, y=CD8, col=cluster))+geom_density2d()
```

```
## Warning: stat_contour(): Zero contours were generated
```

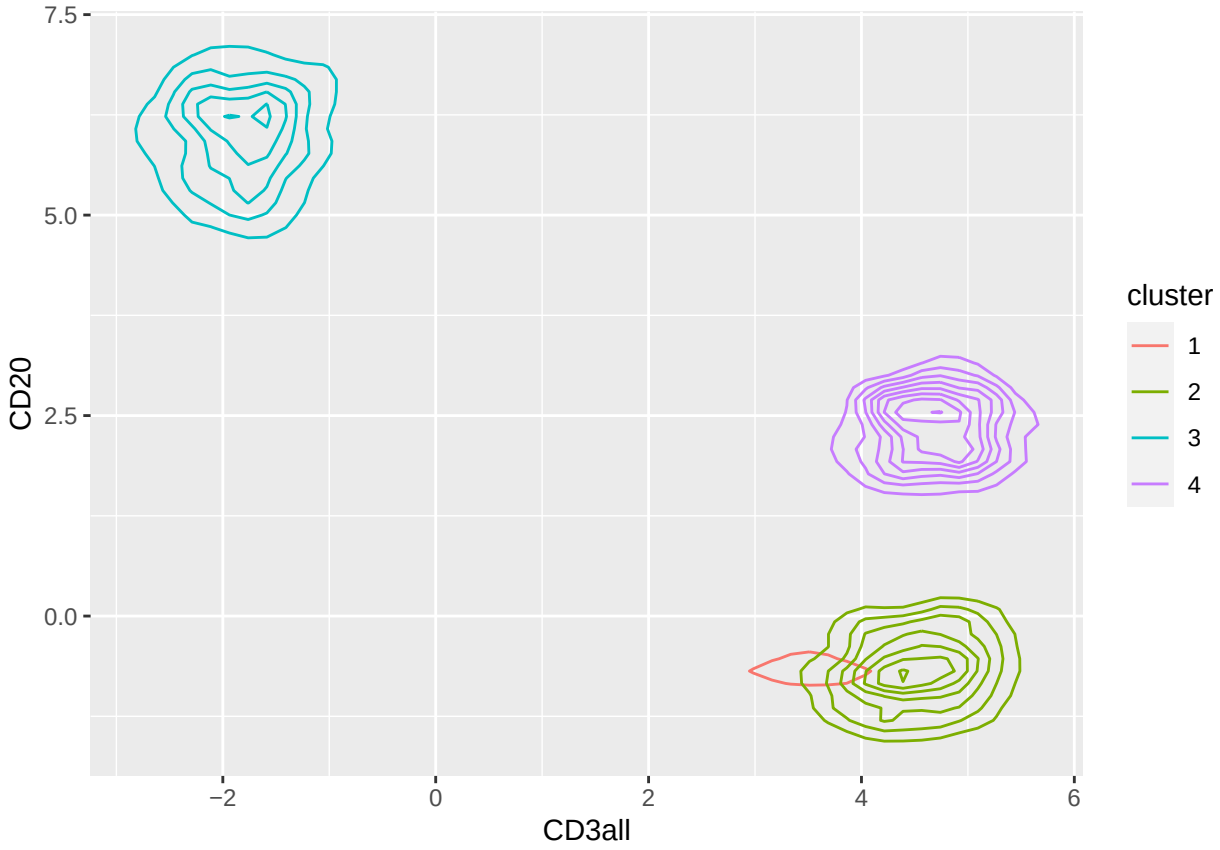
```
## Warning in min(x): no non-missing arguments to min; returning Inf
```

```
## Warning in max(x): no non-missing arguments to max; returning -Inf
```




```
ggplot(mc5df, aes(x=CD3all,  y=CD20, col=cluster))+geom_density2d()
```

```
## Warning: stat_contour(): Zero contours were generated
## Warning in min(x): no non-missing arguments to min; returning Inf
## Warning in max(x): no non-missing arguments to max; returning -Inf
```




```
load("../data/Morder.RData")
```

```
dim(Morder)
```

```
## [1]  30 156
```

```
length(Morder[2,])
```

```
## [1] 156
```

```
sqrt(sum((Morder[1, ] - Morder[2, ])^2))
```

```
## [1] 5.593667
```

```
as.matrix(dist(Morder))[2,1]
```

```
## [1] 5.593667
```

```
mut = read.csv("../data/HIVmutations.csv")
mut[1:3, 10:16]
```

```
##   p32I p33F p34Q p35G p43T p46I p46L
## 1    0    1    0    0    0    0    0
## 2    0    1    0    0    0    1    0
## 3    0    1    0    0    0    0    0
```

```
library("vegan")
```

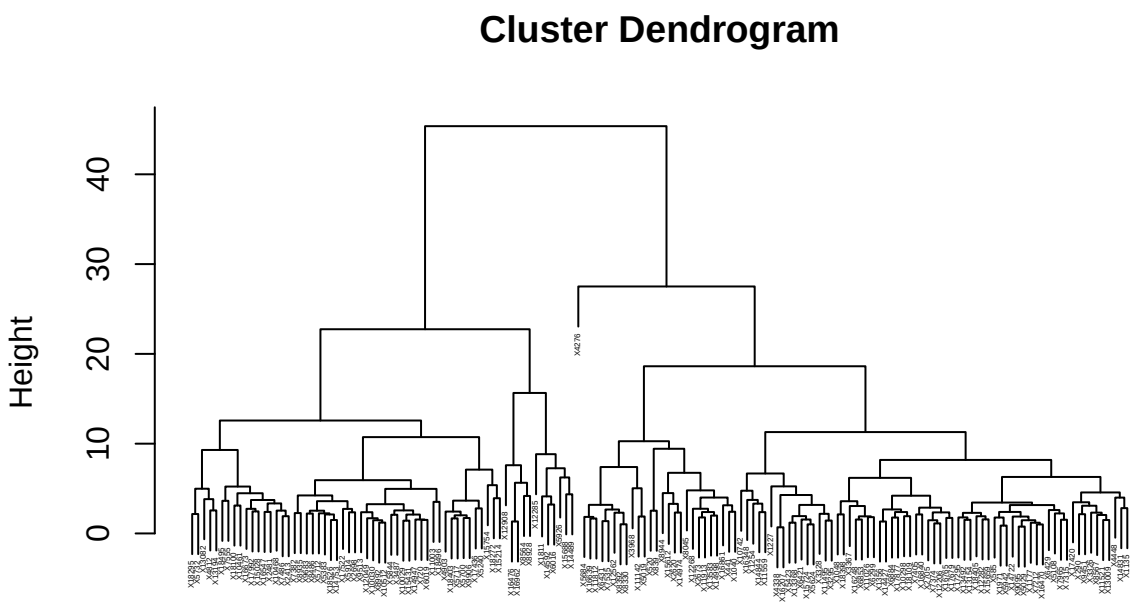
```
## Loading required package: permute
```

```
## This is vegan 2.6-2
```

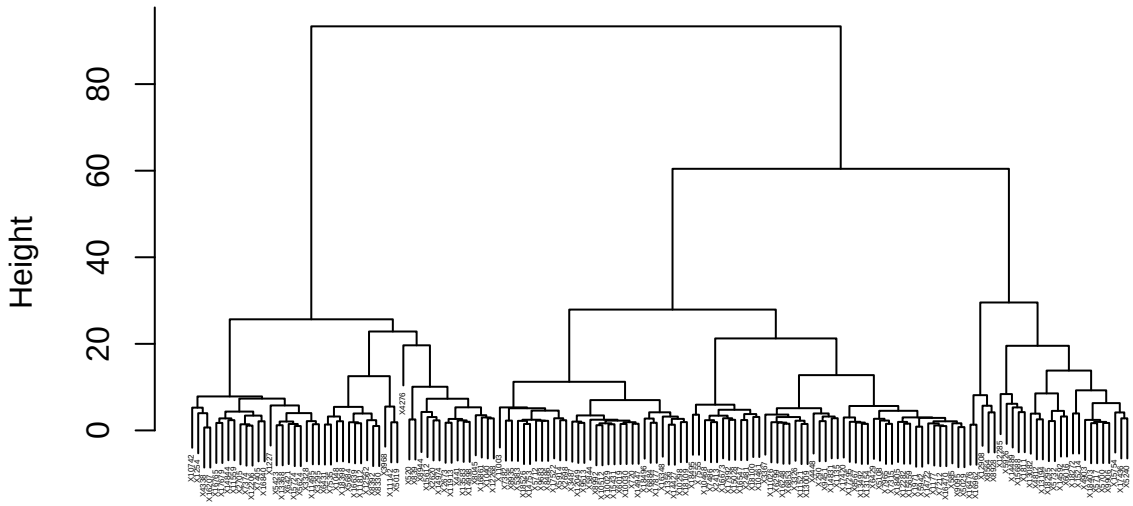
```
as.matrix(vegdist(mut[,10:16]))[1:3,1:3]
```

```
##           1         2         3
## 1 0.0000000 0.3333333 0.0000000
## 2 0.3333333 0.0000000 0.3333333
## 3 0.0000000 0.3333333 0.0000000
```

```
D = dist(t(Morder))
gene_clust = hclust(d = D)
plot(gene_clust, cex=0.25)
```

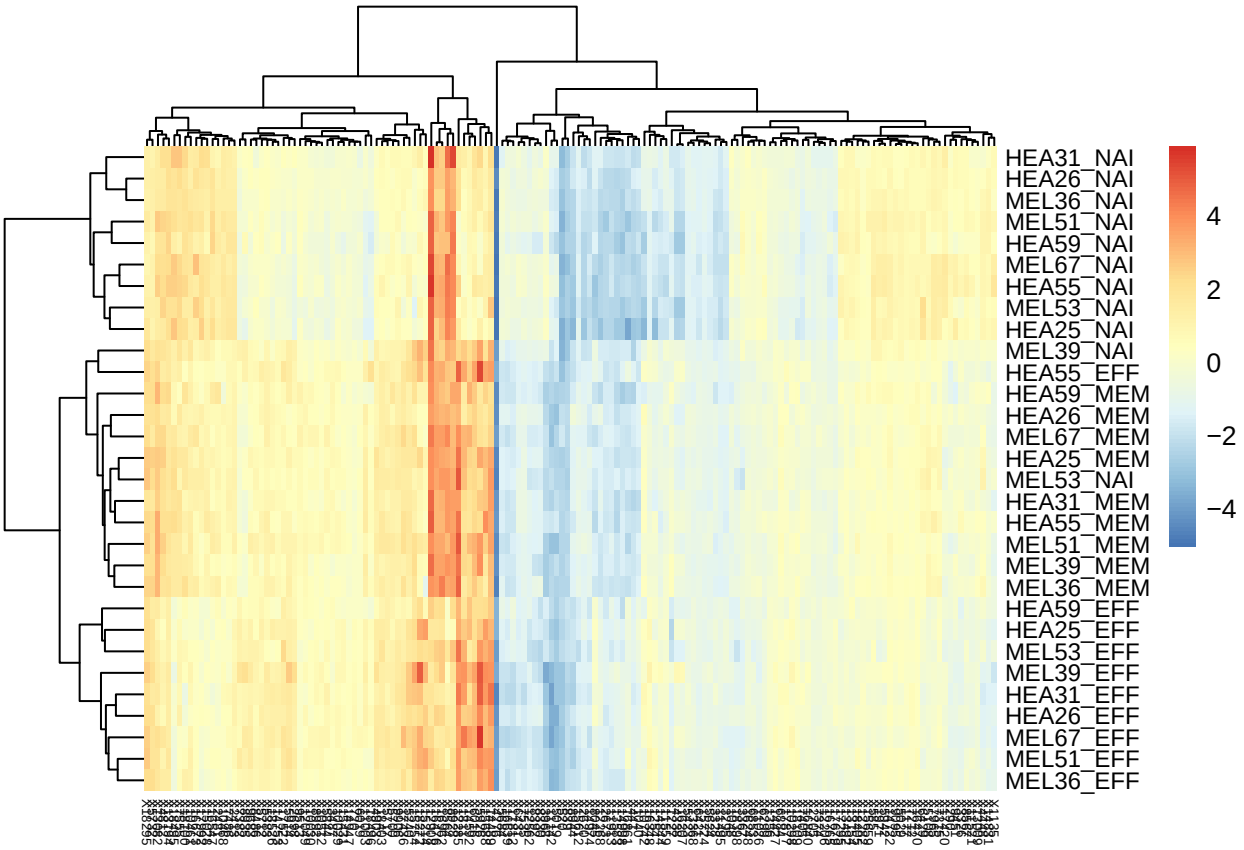



```
gene_clust = hclust(d = D, method="ward.D2")
plot(gene_clust, cex=0.25)
```



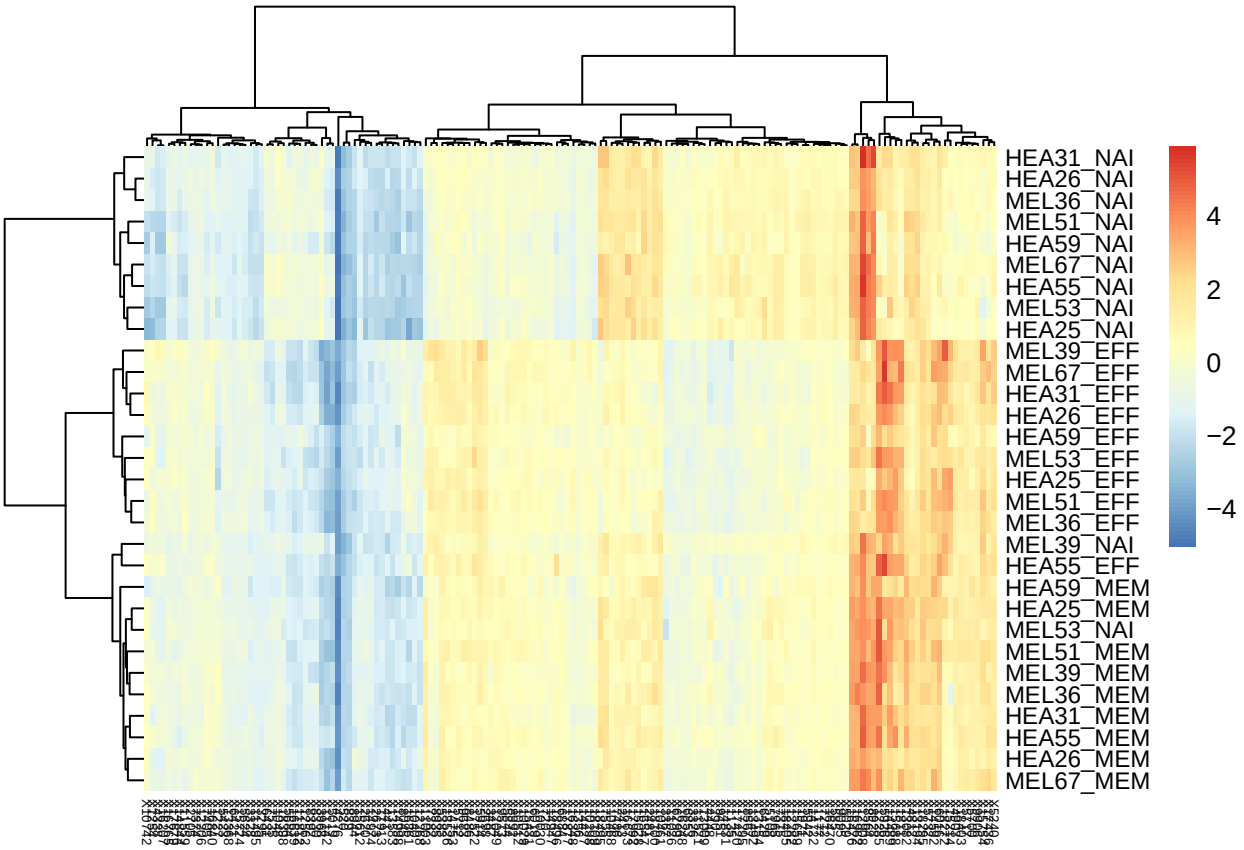

```
#BiocManager::install("pheatmap")
```

```
library("pheatmap")
pheatmap(Morder, fontsize_col = 5, fontsize_row = 9)
```




```
?pheatmap
```

```
pheatmap(Morder, fontsize_col = 5, fontsize_row = 9, clustering_method="ward.D2")
```




```
library('tidyverse')
```

```
## Warning in system("timedatectl", intern = TRUE): running command 'timedatectl'
## had status 1
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
## v purrr   0.3.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x ggcyto::%+%()   masks ggplot2::%+%()
## x dplyr::filter() masks ncdfFlow::filter(), flowCore::filter(), stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
cc_data=read.csv("../SCR3_new_data_SP.csv",fileEncoding="UTF-8-BOM")
d = cc_data
```

```
library("flowCore")
library("flowViz")
fcsB = read.FCS("../data/Bendall_2011.fcs", truncate_max_range=FALSE)
slotNames(fcsB)
```

```
## [1] "exprs"       "parameters"  "description"
```

```
dim(Biobase::exprs(fcsB))
```

```
## [1] 91392    41
```

```
markersB = readr::read_csv("../data/Bendall_2011_markers.csv")
```

```
## Rows: 10 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): isotope, marker
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
mt = match(markersB$isotope, colnames(fcsB))
stopifnot(!any(is.na(mt)))
colnames(fcsB)[mt] = markersB$marker
```

```
markersB
```

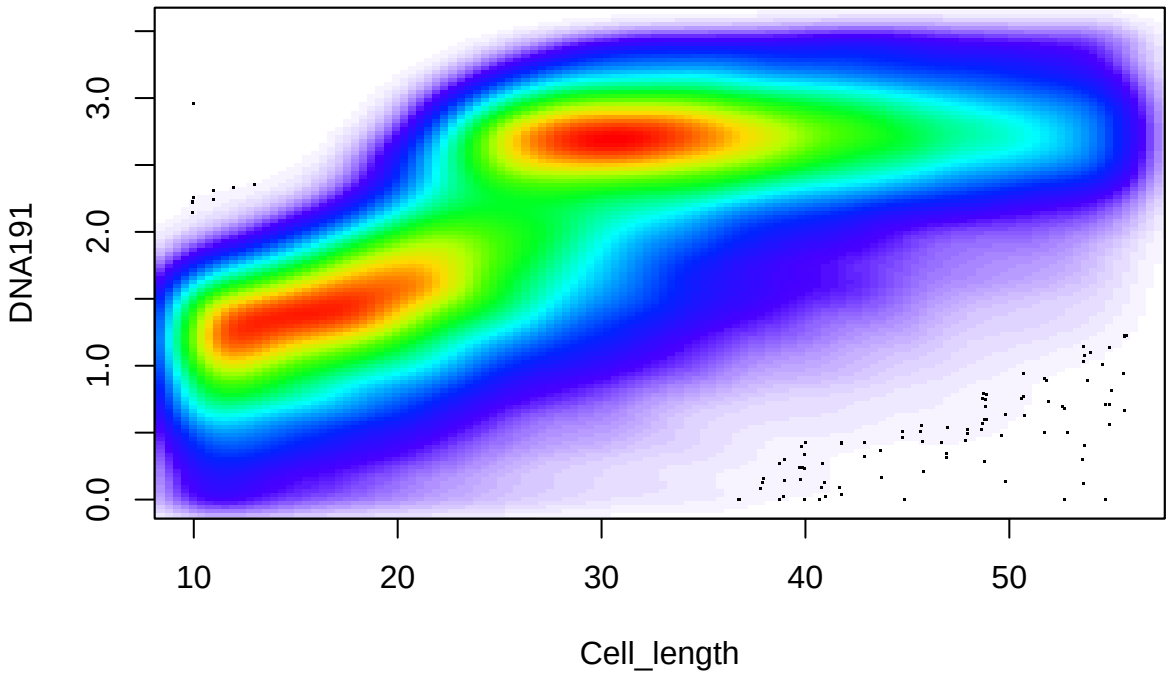
```
## # A tibble: 10 x 2
##    isotope             marker
##    <chr>               <chr>
##  1 Nd(144.912)-Dual    CD4
##  2 Nd(145.913)-Dual    CD8
##  3 Sm(146.914)-Dual    CD20
##  4 Gd(157.924)-Dual    CD33
##  5 Er(169.935)-Dual    CD56
##  6 Ir(190.960)-Dual    DNA191
##  7 Ir(192.962)-Dual    DNA192
##  8 Cd(109.903)-Dual    CD3
##  9 La(138.906)-Dual    CD45RA
## 10 Cd(110,111,112,114) CD3all
```

```
match(c(1,2),c(1,1,1,2,2,2,1,1,1,2,2))
```

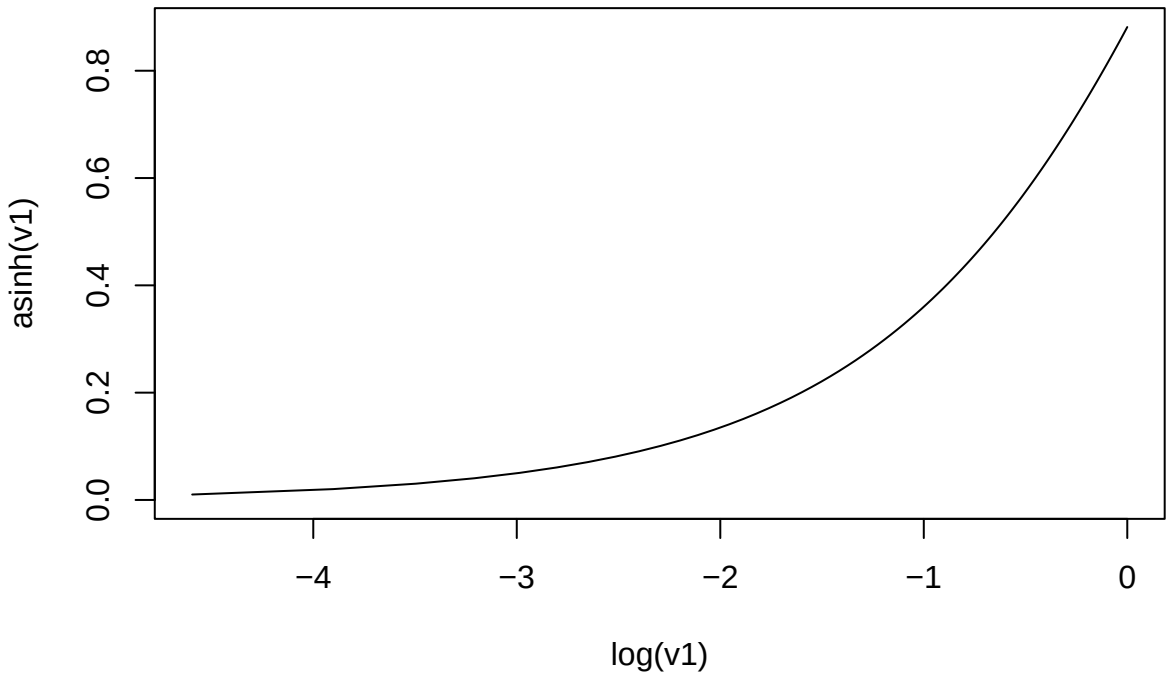
```
## [1] 1 4
```

```
?flowPlot
```

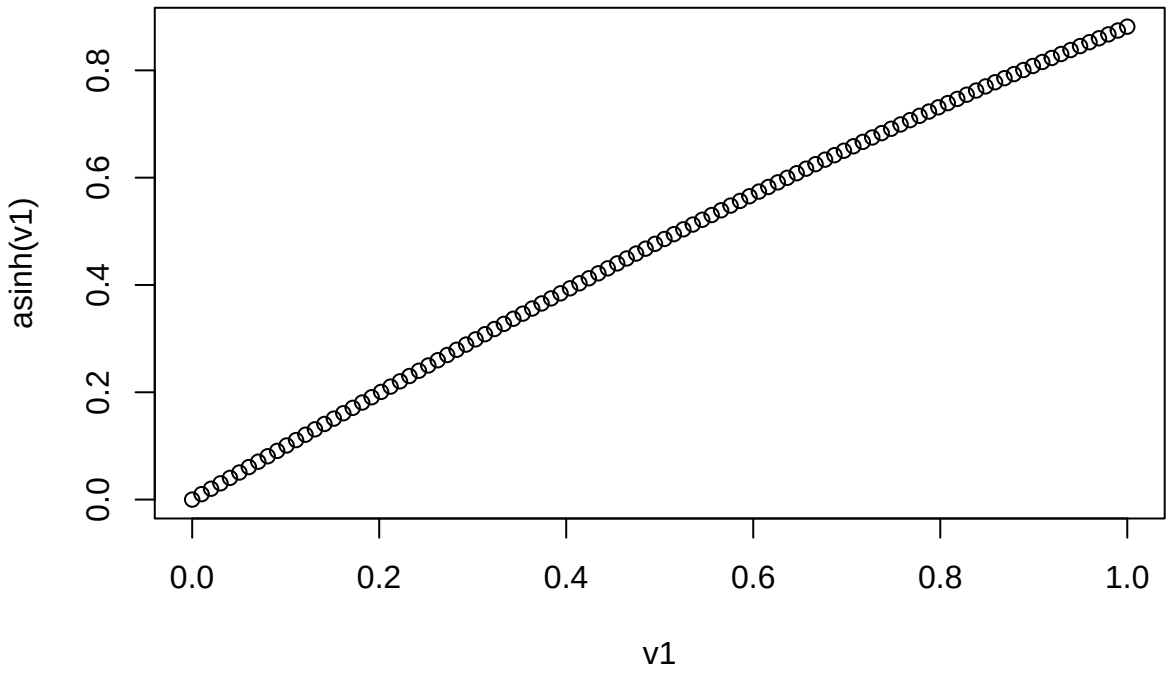
```
flowPlot(fcsB, plotParameters=colnames(fcsB)[2:3], logy=TRUE)
```



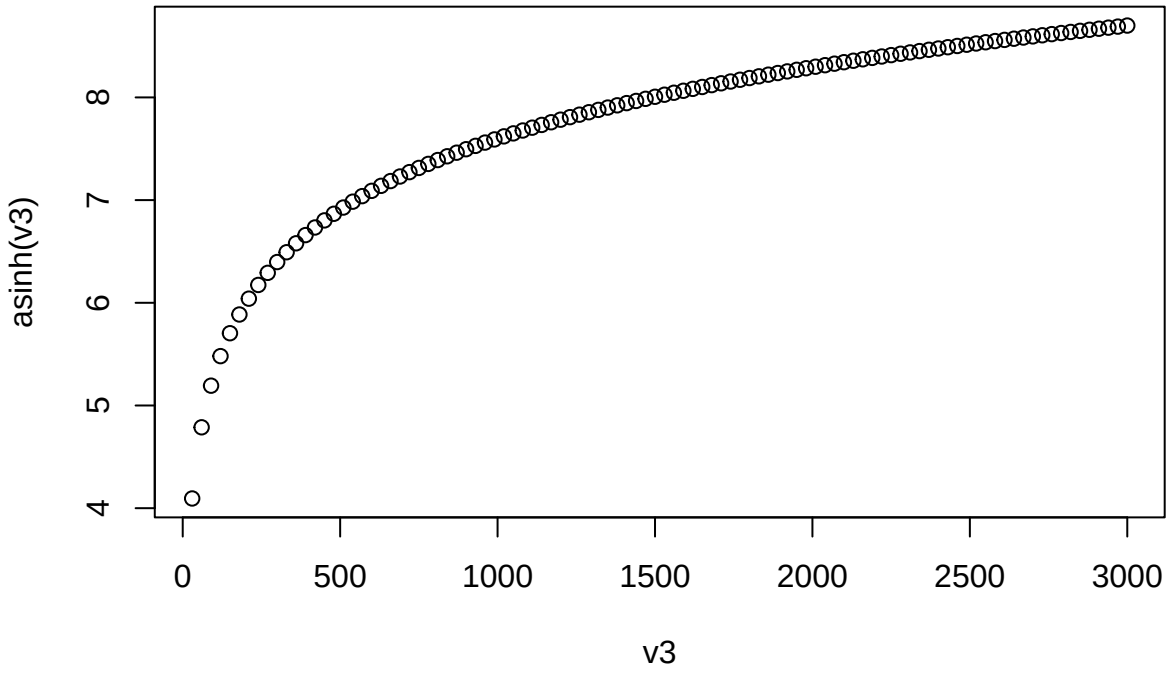

```
v1 = seq(0, 1, length.out = 100)
plot(log(v1), asinh(v1), type = 'l')
```




```
plot(v1, asinh(v1))
```




```
v3 = seq(30, 3000, length = 100)
plot(v3, asinh(v3))
```



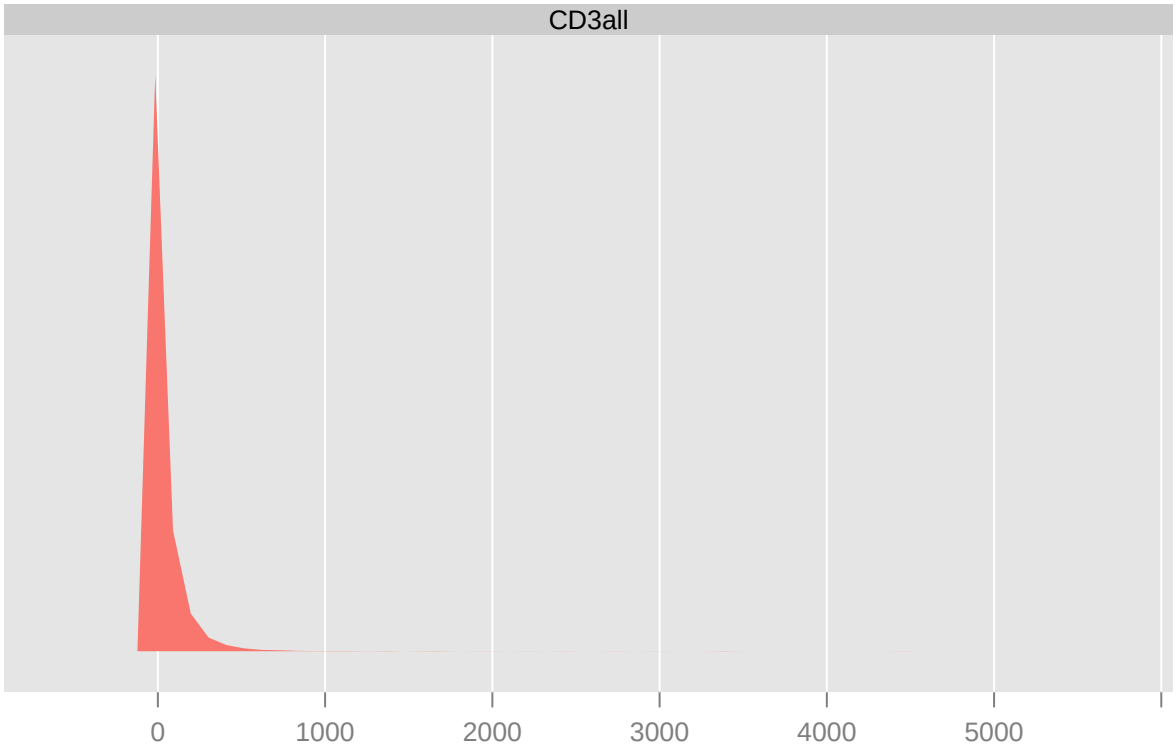

```
asinhtrsf = arcsinhTransform(a=0.1, b=1)
fcsBT = transform(fcsB,
                  transformList(colnames(fcsB)[-c(1,2,41)], asinhtrsf))
```

```
colnames(fcsB)
```

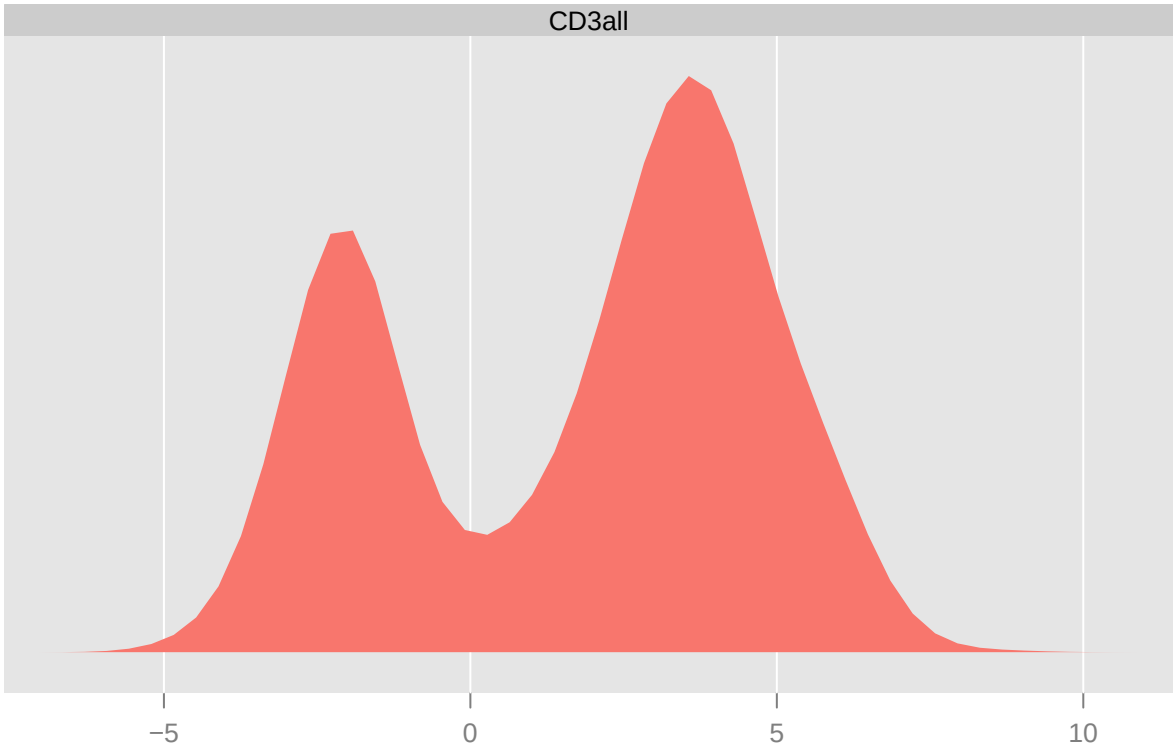
```
##  [1] "Time"                "Cell_length"         "DNA191"
##  [4] "DNA192"              "Rh(102.905)-Dual"    "In(114.903)-Dual"
##  [7] "CD3"                 "Cd(110.904)-Dual"    "Cd(111.902)-Dual"
```

```
## [10] "Cd(113.903)-Dual"    "CD45RA"              "Pr(140.907)-Dual"
## [13] "Nd(141.907)-Dual"    "Nd(143.910)-Dual"    "CD4"
## [16] "CD8"                 "Nd(147.916)-Dual"    "Nd(149.920)-Dual"
## [19] "CD20"                "Sm(151.919)-Dual"    "Sm(153.922)-Dual"
## [22] "Eu(150.919)-Dual"    "Eu(152.921)-Dual"    "Gd(155.922)-Dual"
## [25] "CD33"                "Gd(159.927)-Dual"    "Tb(158.925)-Dual"
## [28] "Dy(163.929)-Dual"    "Ho(164.930)-Dual"    "Er(165.930)-Dual"
## [31] "Er(166.932)-Dual"    "Er(167.932)-Dual"    "CD56"
## [34] "Tm(168.934)-Dual"    "Yb(170.936)-Dual"    "Yb(171.936)-Dual"
## [37] "Yb(173.938)-Dual"    "Yb(175.942)-Dual"    "Lu(174.940)-Dual"
## [40] "CD3all"              "absoluteEventNumber"
```

```
densityplot(~`CD3all`, fcsB)
```




```
densityplot(~`CD3all`, fcsBT)
```




```
?kmeansFilter
```

```
kf = kmeansFilter("CD3all"=c("Pop1", "Pop2"), filterId = "myKmFilter")
fres = flowCore::filter(fcsBT, kf)
summary(fres)
```

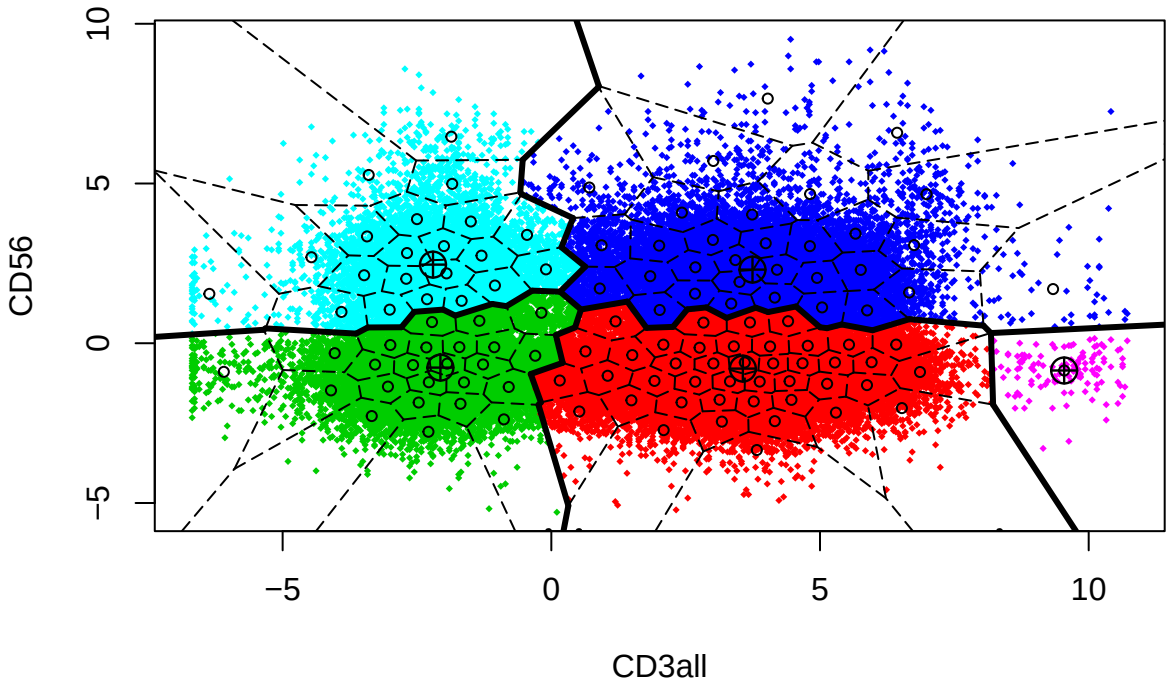
```
## Pop1: 33434 of 91392 events (36.58%)
## Pop2: 57958 of 91392 events (63.42%)
```

```
fcsBT1 = flowCore::split(fcsBT, fres, population = "Pop1")
fcsBT2 = flowCore::split(fcsBT, fres, population = "Pop2")
```

```
library("flowPeaks")
fp = flowPeaks(Biobase::exprs(fcsBT)[, c("CD3all", "CD56")])
```

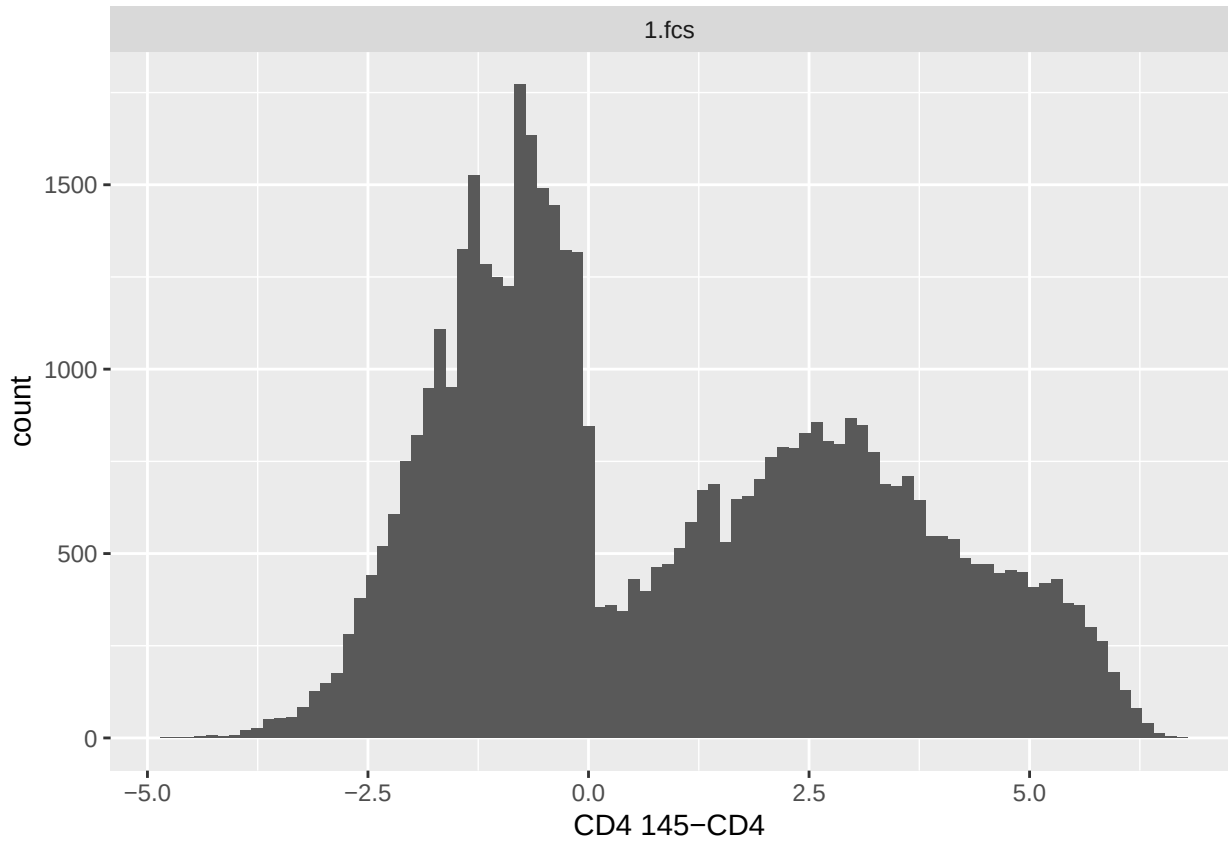
```
##
## Starting the flow Peaks analysis...
##
##     Task A: compute kmeans...
##         step 0, set the intial seeds, tot.wss=14636.9
##         step 1, do the rough EM, tot.wss=10278.9 at 0.247494 sec
##         step 2, do the fine transfer of Hartigan-Wong Algorithm
##                  tot.wss=10221.1 at 0.512915 sec
##         ...finished summarization at 0.515 sec
##
##     Task B: find peaks...
## finished at 0.588 sec
```

```
plot(fp)
```

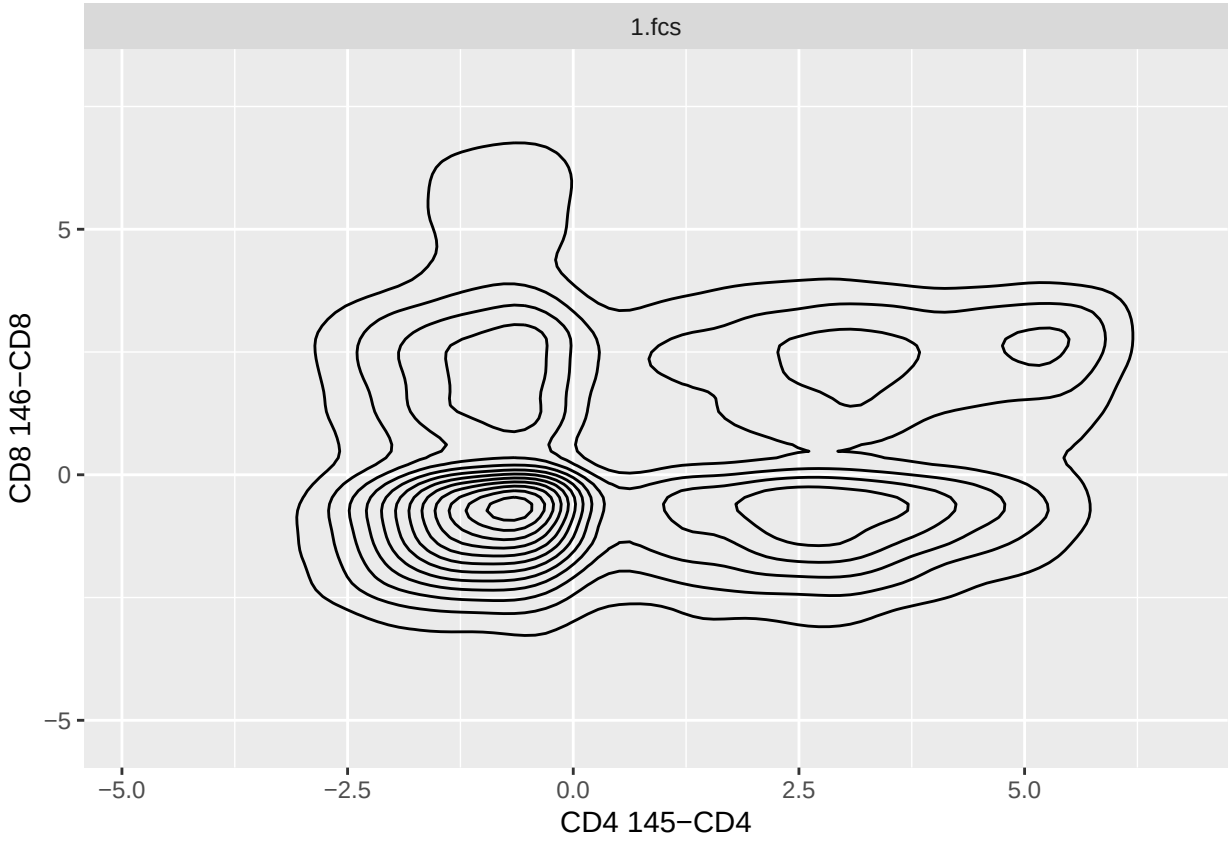



```
library("ggcyto")
library("labeling")
ggcd4cd8 = ggcyto(fcsB, aes(x=CD4, y=CD8))
ggcd4 = ggcyto(fcsB, aes(x=CD4))
ggcd8 = ggcyto(fcsB, aes(x=CD8))
p1 = ggcd4 + geom_histogram(bins=60)
p1b = ggcd8 + geom_histogram(bins=60)
asinhT = arcsinhTransform(a=0, b=1)
trans1 = transformList(colnames(fcsB)[-c(1,2,41)], asinhT)
fcsBT = transform(fcsB, trans1)
p1t = ggcyto(fcsBT, aes(x=CD4))+geom_histogram(bins=90)
p2t = ggcyto(fcsBT, aes(x=CD4, y=CD8))+geom_density2d(colour="black")
p3t = ggcyto(fcsBT, aes(x=CD45RA, y=CD20))+geom_density2d(colour="black")
```

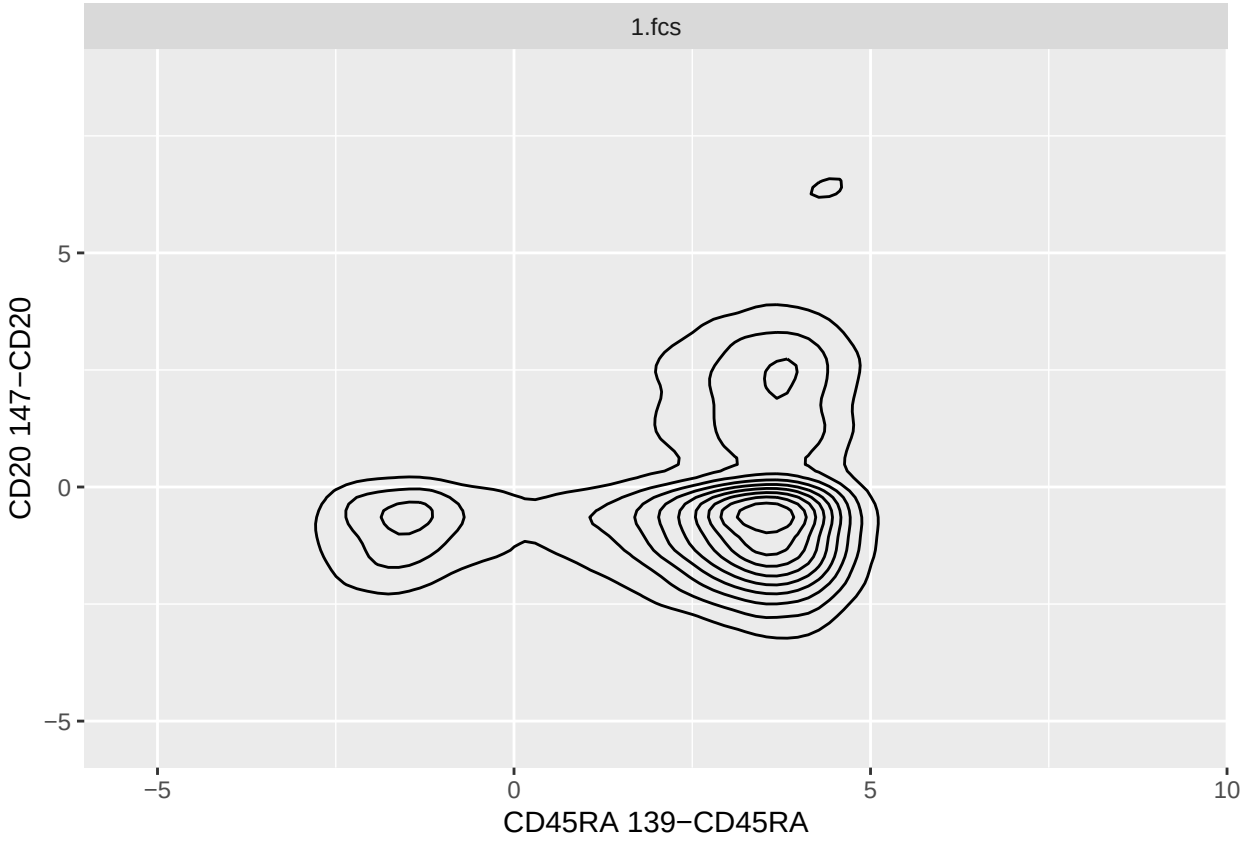
```
p1t
```




```
p2t
```




```
p3t
```

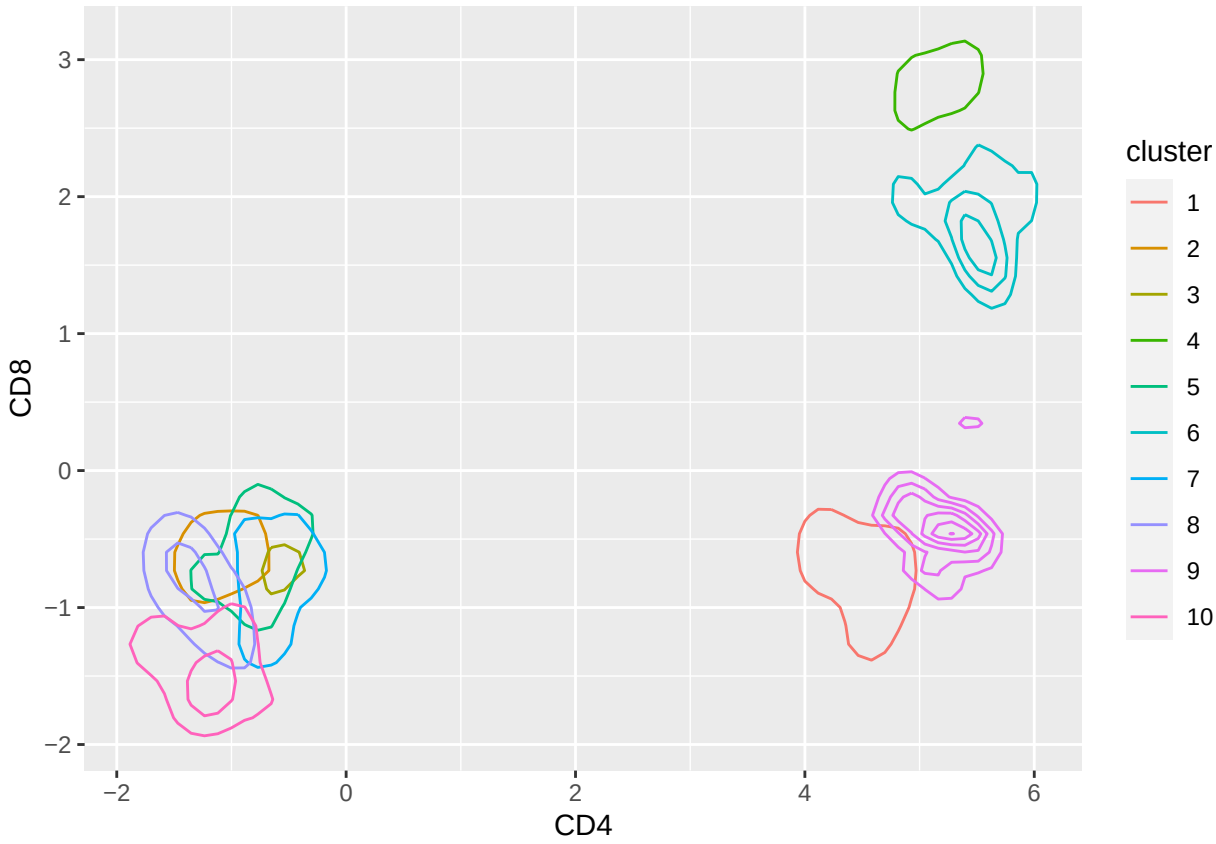



```
library("dbscan")
mc5 = Biobase::exprs(fcsBT)[,c(11,15,16,19,25, 33, 40)]
res5 = dbscan::dbscan(mc5, eps=0.95, minPts=20)
mc5df = data.frame(mc5, cluster=as.factor(res5$cluster))
table(mc5df$cluster)
```

```
##
##     0     1     2     3     4     5     6     7     8     9    10
## 90249    21   102   445   158   119    19   224    17    20    18
```

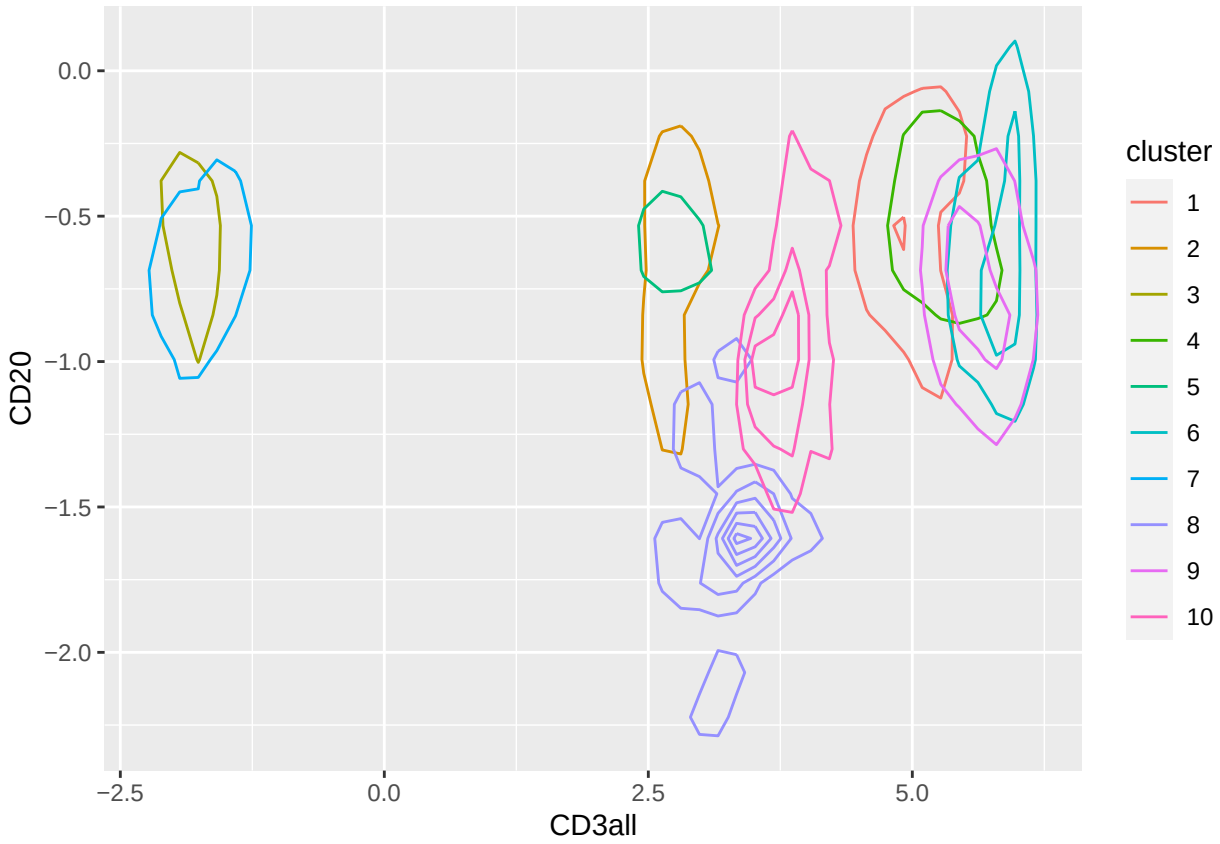
```
ggplot(mc5df, aes(x=CD4, y=CD8, col=cluster))+geom_density2d()
```

```
## Warning: stat_contour(): Zero contours were generated

## Warning in min(x): no non-missing arguments to min; returning Inf

## Warning in max(x): no non-missing arguments to max; returning -Inf
```




```
ggplot(mc5df, aes(x=CD3all,  y=CD20, col=cluster))+geom_density2d()
```

```
## Warning: stat_contour(): Zero contours were generated
## Warning in min(x): no non-missing arguments to min; returning Inf
## Warning in max(x): no non-missing arguments to max; returning -Inf
```




```
BiocManager::install("Morder")
```

```
## Bioconductor version 3.15 (BiocManager 1.30.18), R 4.2.0 (2022-04-22)
```

```
## Installing package(s) 'Morder'
```

```
## Warning: package 'Morder' is not available for Bioconductor version '3.15'
##
## A version of this package for your version of R might be available elsewhere,
## see the ideas at
## https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

```
## Installation paths not writeable, unable to update packages
##   path: /usr/local/lib/R/library
##   packages:
##     MASS, nlme
```

```
## Old packages: 'broom', 'bslib', 'callr', 'DBI', 'dbplyr', 'ensembldb',
##   'farver', 'flowViz', 'generics', 'ggcyto', 'gtools', 'htmltools', 'KEGGREST',
##   'latticeExtra', 'limma', 'locfit', 'MatrixGenerics', 'ncdfFlow', 'pillar',
##   'processx', 'Rcpp', 'RcppArmadillo', 'restfulr', 'RSQLite', 'rtracklayer',
##   'sass', 'shiny', 'stringi', 'tidyverse', 'tinytex'
```

```
library("MOrder")
```

```
load("../data/Morder.RData")
```

```
dim(Morder)
```

```
## [1]  30 156
```

```r
length(Morder[2,])
```

```
## [1] 156
```

```r
sqrt(sum((Morder[1, ] - Morder[2, ])^2))
```

```
## [1] 5.593667
```

```r
as.matrix(dist(Morder))[2,1]
```

```
## [1] 5.593667
```

```r
mut = read.csv("../data/HIVmutations.csv")
mut[1:3, 10:16]
```

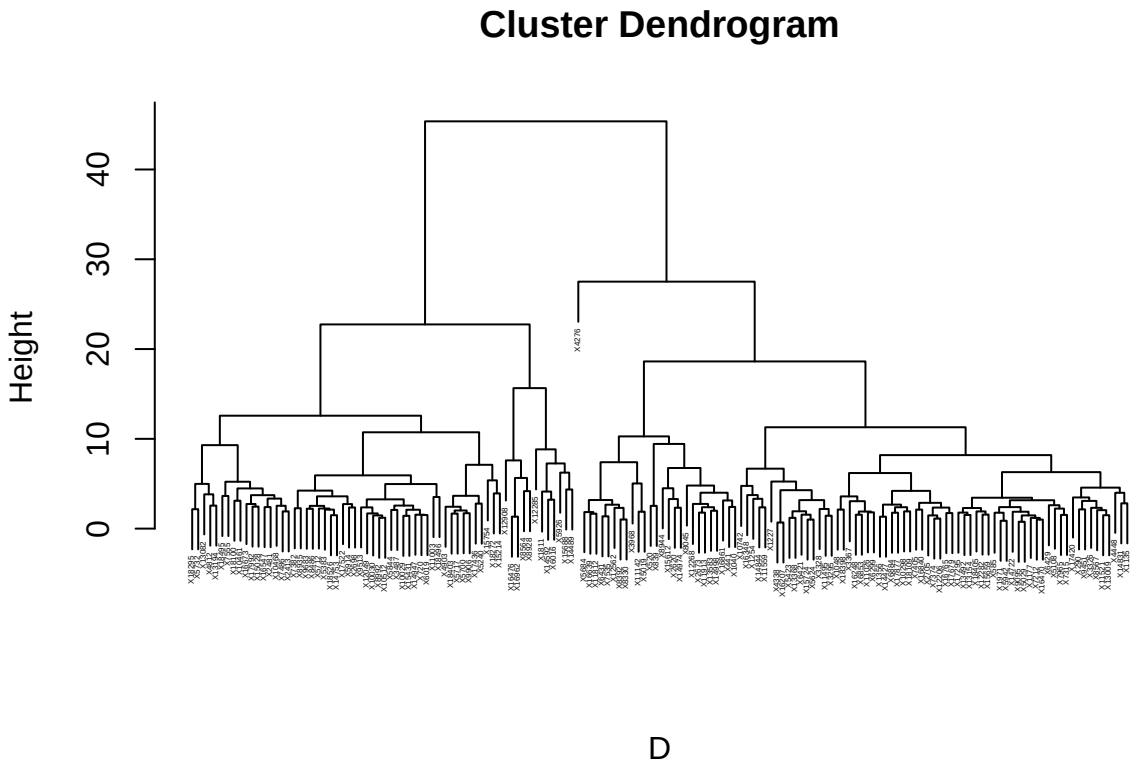
```
##   p32I p33F p34Q p35G p43T p46I p46L
## 1    0    1    0    0    0    0    0
## 2    0    1    0    0    0    1    0
## 3    0    1    0    0    0    0    0
```

```r
library("vegan")
```

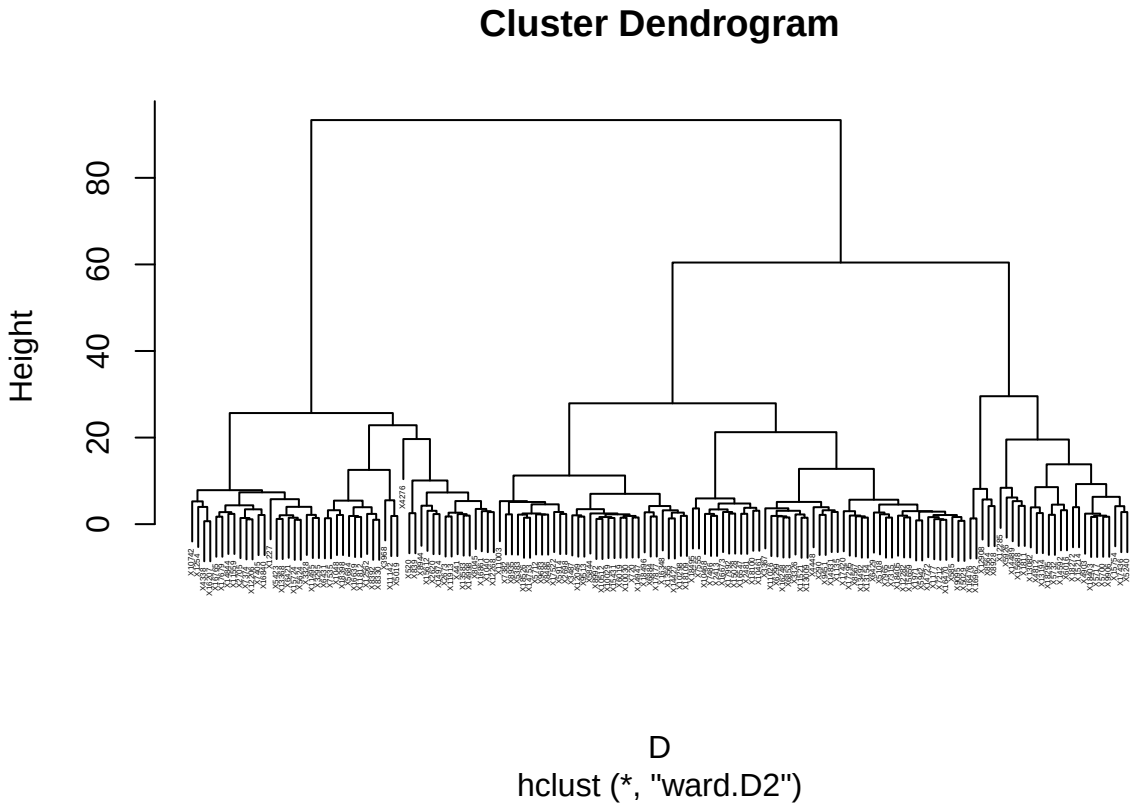
```r
as.matrix(vegdist(mut[,10:16]))[1:3,1:3]
```

```
##           1         2         3
## 1 0.0000000 0.3333333 0.0000000
## 2 0.3333333 0.0000000 0.3333333
## 3 0.0000000 0.3333333 0.0000000
```

```r
D = dist(t(Morder))
gene_clust = hclust(d = D)
plot(gene_clust, cex=0.25)
```

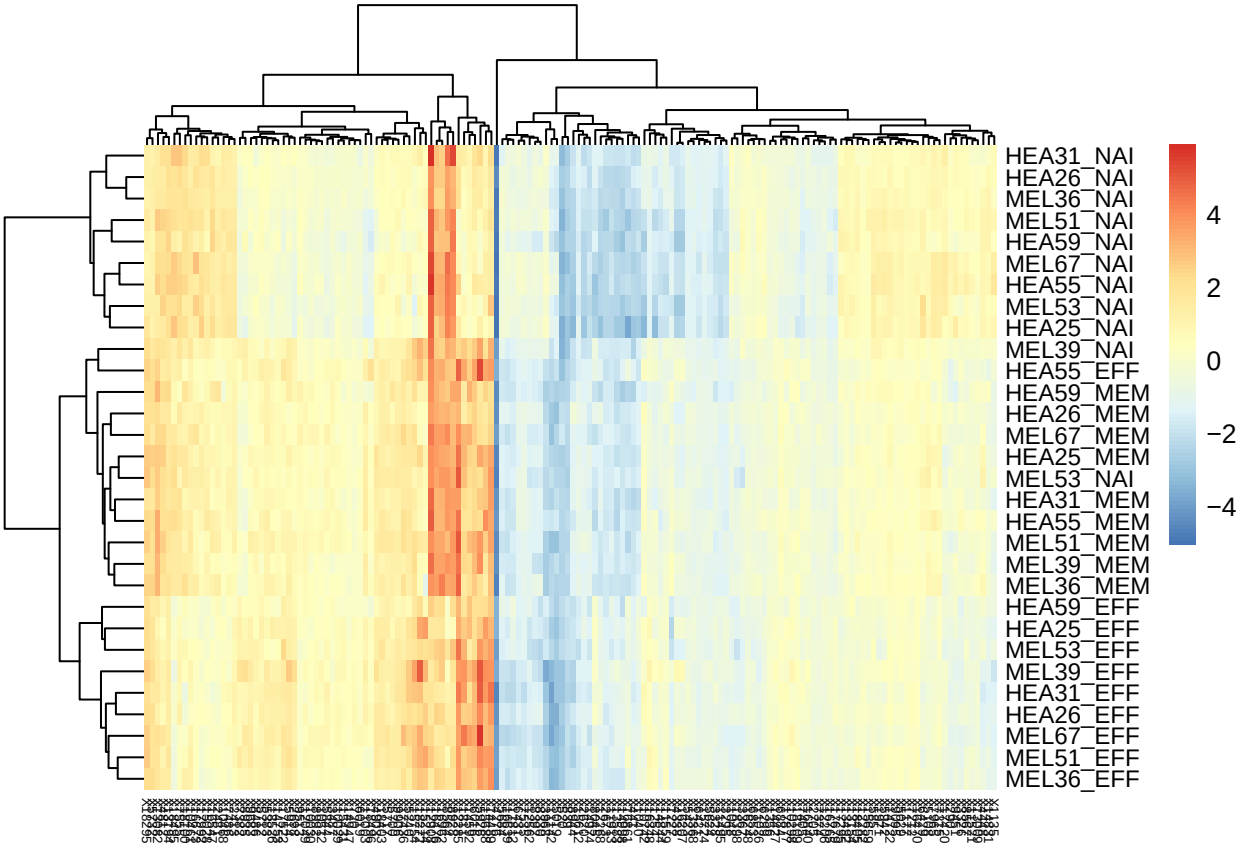



```
gene_clust = hclust(d = D, method="ward.D2")
plot(gene_clust, cex=0.25)
```



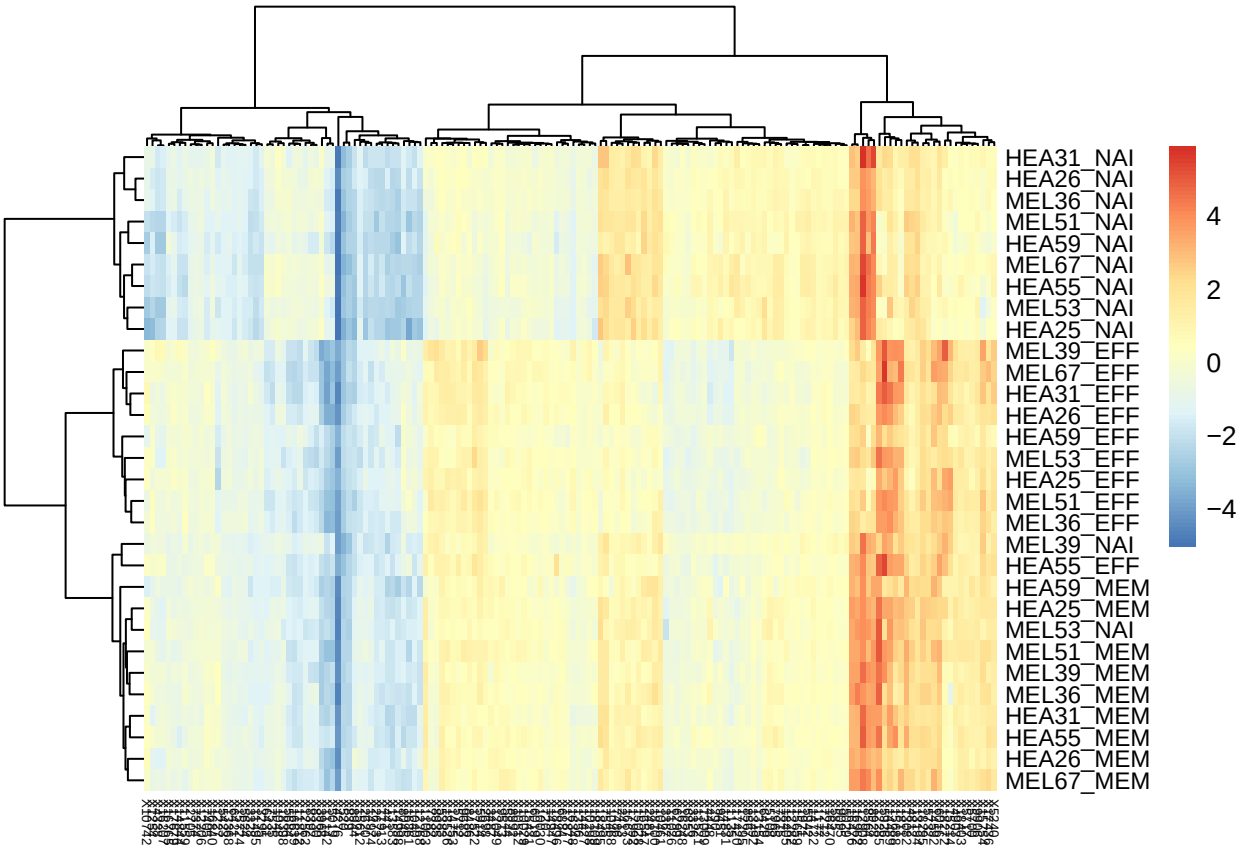
```
#BiocManager::install("pheatmap")
```

```
library("pheatmap")
pheatmap(Morder, fontsize_col = 5, fontsize_row = 9)
```




```
?pheatmap
```

```
pheatmap(Morder, fontsize_col = 5, fontsize_row = 9, clustering_method="ward.D2")
```




```r
library('tidyverse')
```

```r
cc_data=read.csv("../SCR3_new_data_SP.csv",fileEncoding="UTF-8-BOM")
d = cc_data
d1= filter(d,f2=="Medio_lateral")
d1$type=rep("MU",length(d1$x))
d1$type[d1$f1 %in% "WT"] = "WT"
e <- ggplot(data = d1, aes(x = f1, y = x,color=type))+
  stat_boxplot(geom ='errorbar',position = position_dodge(0.85), width=0.5)
```

```r
mx = c(0,0,0,1,1,1)
my = c(1,0,1,1,0,1)
mz = c(1,1,1,0,1,1)

mat = rbind(mx, my, mz)
dist(mat)
```

```
##          mx       my
## my 1.732051
## mz 2.000000 1.732051
```

d1= filter(d,f2==“Medio_lateral”) d1$type = rep(”MU”, length(d1x)) d1$type[d1f1 %in% “WT”] = “WT” e <- ggplot(data = d1, aes(x = f1, y = x,color=type))+ stat_boxplot(geom =‘errorbar’,position = position_dodge(0.85), width=0.5)

```r
mx = c(0,0,0,1,1,1)
```

```
my = c(1,0,1,1,0,1)
mz = c(1,1,1,0,1,1)

mat = rbind(mx, my, mz)
dist(mat)
```

```
##          mx       my
## my 1.732051
## mz 2.000000 1.732051
```