PEC1_V2

Julian Sanchez Bautista

5/1/2020

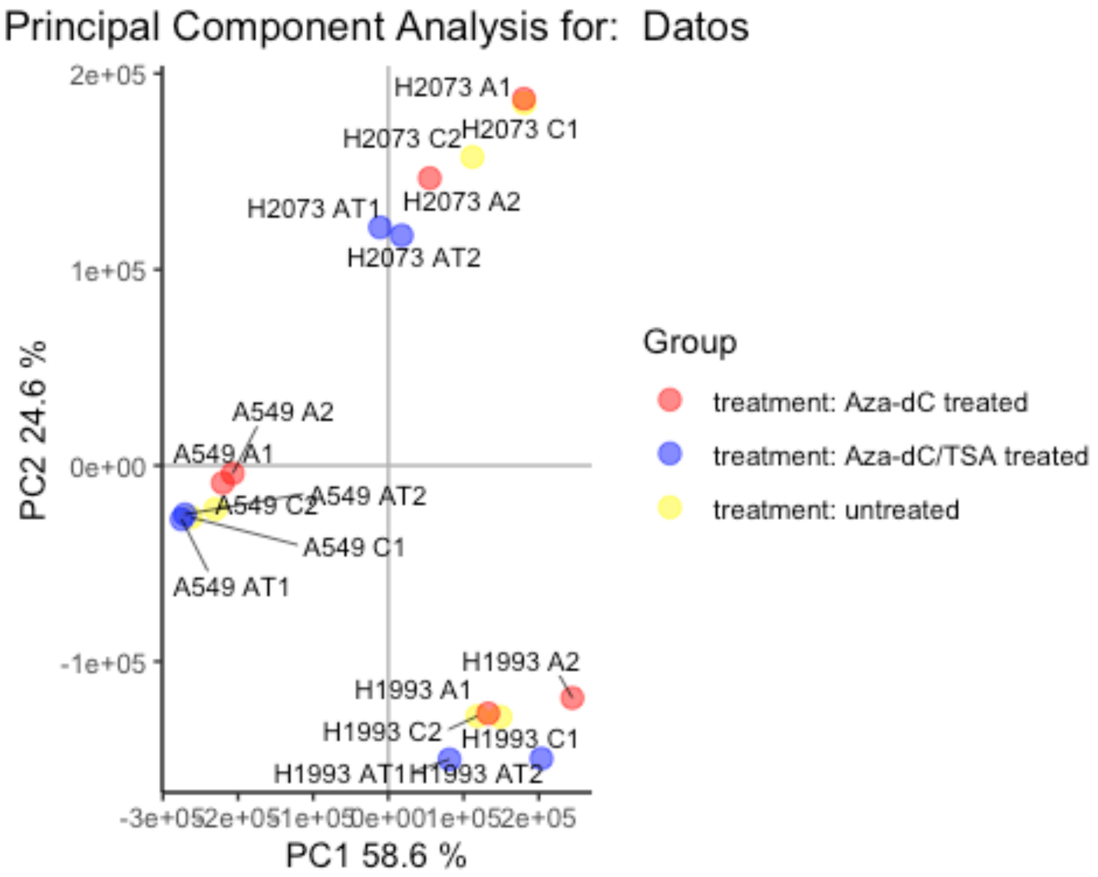
```
library(GEOquery)
```

```
## Loading required package: Biobase

## Loading required package: BiocGenerics

## Loading required package: parallel

##
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB

## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs

## The following objects are masked from 'package:base':
##
##     Filter, Find, Map, Position, Reduce, anyDuplicated, append,
##     as.data.frame, basename, cbind, colnames, dirname, do.call,
##     duplicated, eval, evalq, get, grep, grepl, intersect, is.unsorted,
##     lapply, mapply, match, mget, order, paste, pmax, pmax.int, pmin,
##     pmin.int, rank, rbind, rownames, sapply, setdiff, sort, table,
##     tapply, union, unique, unsplit, which, which.max, which.min

## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.

## Setting options('download.file.method.GEOquery'='auto')

## Setting options('GEOquery.inmemory.gpl'=FALSE)
```

```
gset <- getGEO("GSE32496", GSEMatrix =TRUE, getGPL=FALSE)
```

```
## Found 1 file(s)
## GSE32496_series_matrix.txt.gz

## Parsed with column specification:
## cols(
##   ID_REF = col_character(),
##   GSM804399 = col_double(),
##   GSM804400 = col_double(),
##   GSM804401 = col_double(),
##   GSM804402 = col_double(),
##   GSM804403 = col_double(),
##   GSM804404 = col_double(),
##   GSM804405 = col_double(),
##   GSM804406 = col_double(),
##   GSM804407 = col_double(),
##   GSM804408 = col_double(),
##   GSM804409 = col_double(),
##   GSM804410 = col_double(),
##   GSM804411 = col_double(),
##   GSM804412 = col_double(),
##   GSM804413 = col_double(),
##   GSM804414 = col_double(),
##   GSM804415 = col_double(),
##   GSM804416 = col_double()
## )
```

```
if (length(gset) > 1) idx <- grep("GPL570", attr(gset, "names")) else idx <-
1
gset <- gset[[idx]]

dev.new(width=4+dim(gset)[[2]]/5, height=6)
par(mar=c(2+round(max(nchar(sampleNames(gset)))/2),4,2,1))
title <- paste ("GSE32496", '/', annotation(gset), " selected samples", sep
='')
boxplot(exprs(gset), boxwex=0.7, notch=T, main=title, outline=FALSE, las=2)

library(arrayQualityMetrics)

library(ggplot2)
library(ggrepel)

plotPCA3 <- function (datos, labels, factor, title, scale,colores, size =
1.5, glineas = 0.25) {
  data <- prcomp(t(datos),scale=scale)
  dataDf <- data.frame(data$x)
  Group <- factor
  loads <- round(data$sdev^2/sum(data$sdev^2)*100,1)
  p1 <- ggplot(dataDf,aes(x=PC1, y=PC2)) +
    theme_classic() +
    geom_hline(yintercept = 0, color = "gray70") +
    geom_vline(xintercept = 0, color = "gray70") +
    geom_point(aes(color = Group), alpha = 0.55, size = 3) +
    coord_cartesian(xlim = c(min(data$x[,1])-5,max(data$x[,1])+5)) +
    scale_fill_discrete(name = "Group")
  p1 + geom_text_repel(aes(y = PC2 + 0.25, label = labels),segment.size =
0.25, size = size) +
    labs(x = c(paste("PC1",loads[1],"%")),y=c(paste("PC2",loads[2],"%"))) +
    ggtitle(paste("Principal Component Analysis for: ",title,sep=" "))+
    theme(plot.title = element_text(hjust = 0.5)) +
    scale_color_manual(values=colores)
}

plotPCA3(exprs(gset), labels = gset$title, factor =
gset$characteristics_ch1.1,
        title="Datos", scale = FALSE, size = 3,
        colores = c("red", "blue", "yellow"))
```



```
class(gset)

## [1] "ExpressionSet"
## attr(,"package")
## [1] "Biobase"

library(limma)
```

```
##
## Attaching package: 'limma'
```

```
## The following object is masked from 'package:BiocGenerics':
##
##     plotMA
```

```
designMat<- model.matrix(~0+gset$characteristics_ch1.1, pData(gset))
colnames(designMat) <- c("Aza.dC", "Aza.dC.TSA", "untreated")
print(designMat)
```

```
##           Aza.dC Aza.dC.TSA untreated
## GSM804399      0          0         1
## GSM804400      0          0         1
## GSM804401      1          0         0
## GSM804402      1          0         0
## GSM804403      0          1         0
## GSM804404      0          1         0
## GSM804405      0          0         1
## GSM804406      0          0         1
## GSM804407      1          0         0
## GSM804408      1          0         0
## GSM804409      0          1         0
## GSM804410      0          1         0
## GSM804411      0          0         1
## GSM804412      0          0         1
## GSM804413      1          0         0
## GSM804414      1          0         0
## GSM804415      0          1         0
## GSM804416      0          1         0
## attr(,"assign")
## [1] 1 1 1
## attr(,"contrasts")
## attr(,"contrasts")$`gset$characteristics_ch1.1`
## [1] "contr.treatment"
```

```
cont.matrix <- makeContrasts (Aza.dCvsAza.dc.TSA = Aza.dC-Aza.dC.TSA,
                              Aza.dCvsuntreated = Aza.dC-untreated,
                              Aza.dC.TSAvsuntreated = Aza.dC.TSA-untreated,
                              INT = (Aza.dC-Aza.dC.TSA)-(Aza.dC-untreated)-
(Aza.dC.TSA-untreated),
                              levels=designMat)
print(cont.matrix)
```

```
##             Contrasts
## Levels       Aza.dCvsAza.dc.TSA Aza.dCvsuntreated Aza.dC.TSAvsuntreated
INT
##   Aza.dC                      1                 1                     0
0
##   Aza.dC.TSA                 -1                 0                     1  -
2
```

```
##   untreated                   0                -1                    -1
2
```

```
library(limma)
fit<-lmFit(gset, designMat)
fit.main<-contrasts.fit(fit, cont.matrix)
fit.main<-eBayes(fit.main)
class(fit.main)
```

```
## [1] "MArrayLM"
## attr(,"package")
## [1] "limma"
```

```
topTab_Aza.dCvsAza.dc.TSA <- topTable (fit.main, number=nrow(fit.main),
coef="Aza.dCvsAza.dc.TSA", adjust="fdr")
head(topTab_Aza.dCvsAza.dc.TSA)
```

```
##                  logFC   AveExpr          t      P.Value    adj.P.Val
B
## 214432_at   -169.93835  93.55024 -10.412559 1.942811e-08 0.0006123219 -
4.010739
## 241908_at   -227.48463 368.15936 -10.305503 2.239861e-08 0.0006123219 -
4.012451
## 204229_at   -393.01054 208.54777  -9.619467 5.728878e-08 0.0010440880 -
4.024547
## 223242_s_at -298.89571 835.96450  -9.322801 8.731283e-08 0.0011934572 -
4.030448
## 1557948_at   279.42891 461.06174   8.624017 2.449209e-07 0.0026782099 -
4.046254
## 209663_s_at  -92.36867  54.28896  -8.270905 4.215444e-07 0.0038413237 -
4.055415
```

```
topTab_Aza.dCvsuntreated <- topTable (fit.main, number=nrow(fit.main),
coef="Aza.dCvsuntreated", adjust="fdr")
head(topTab_Aza.dCvsuntreated)
```

```
##                logFC   AveExpr         t      P.Value  adj.P.Val         B
## 225332_at -427.20447 897.13925 -8.398644 3.457635e-07 0.01890462 -4.577982
## 224580_at -117.81529 279.52284 -6.264333 1.250435e-05 0.34183763 -4.580324
## 217530_at  -37.78214  22.91832 -5.502283 5.224823e-05 0.66210164 -4.581582
## 205808_at -276.68211 730.72300 -5.487543 5.375457e-05 0.66210164 -4.581610
## 223944_at  104.61549 102.89354  5.425973 6.054885e-05 0.66210164 -4.581725
## 205966_at   42.17870  61.97030  5.234334 8.797069e-05 0.70624959 -4.582097
```

```
topTab_Aza.dC.TSAvsuntreated <- topTable (fit.main, number=nrow(fit.main),
coef="Aza.dC.TSAvsuntreated", adjust="fdr")
head(topTab_Aza.dC.TSAvsuntreated)
```

```
##                 logFC   AveExpr         t      P.Value    adj.P.Val
B
## 223242_s_at  392.0735 835.96450 12.229094 2.036631e-09 0.0001113528 -
3.987277
```

```
## 241908_at    239.2212 368.15936 10.837192 1.116969e-08 0.0002132936 -
4.004356
## 204229_at    441.2780 208.54777 10.800879 1.170335e-08 0.0002132936 -
4.004877
## 214432_at    165.3870  93.55024 10.133689 2.821073e-08 0.0003856054 -
4.015294
## 204230_s_at  401.0797 225.42877  9.296857 9.063177e-08 0.0009910584 -
4.030986
## 226882_x_at -500.2685 754.55690 -9.068445 1.262791e-07 0.0011507179 -
4.035874
```

```
topTab_INT  <- topTable (fit.main, number=nrow(fit.main), coef="INT",
adjust="fdr")
head(topTab_INT)
```

```
##                 logFC   AveExpr          t      P.Value    adj.P.Val
B
## 223242_s_at -784.1471 835.96450 -12.229094 2.036631e-09 0.0001113528 -
4.437612
## 241908_at   -478.4424 368.15936 -10.837192 1.116969e-08 0.0002132936 -
4.441912
## 204229_at   -882.5561 208.54777 -10.800879 1.170335e-08 0.0002132936 -
4.442044
## 214432_at   -330.7741  93.55024 -10.133689 2.821073e-08 0.0003856054 -
4.444670
## 204230_s_at -802.1594 225.42877  -9.296857 9.063177e-08 0.0009910584 -
4.448631
## 226882_x_at 1000.5370 754.55690   9.068445 1.262791e-07 0.0011507179 -
4.449866
```

```
annotatedTopTable <- function(topTab, anotPackage)
 {
   topTab <- cbind(PROBEID=rownames(topTab), topTab)
   myProbes <- rownames(topTab)
   thePackage <- eval(parse(text = anotPackage))
   geneAnots <- select(thePackage, myProbes, c("SYMBOL", "ENTREZID",
"GENENAME"))
   annotatedTopTab<- merge(x=geneAnots, y=topTab, by.x="PROBEID",
by.y="PROBEID")
 return(annotatedTopTab)
 }
```

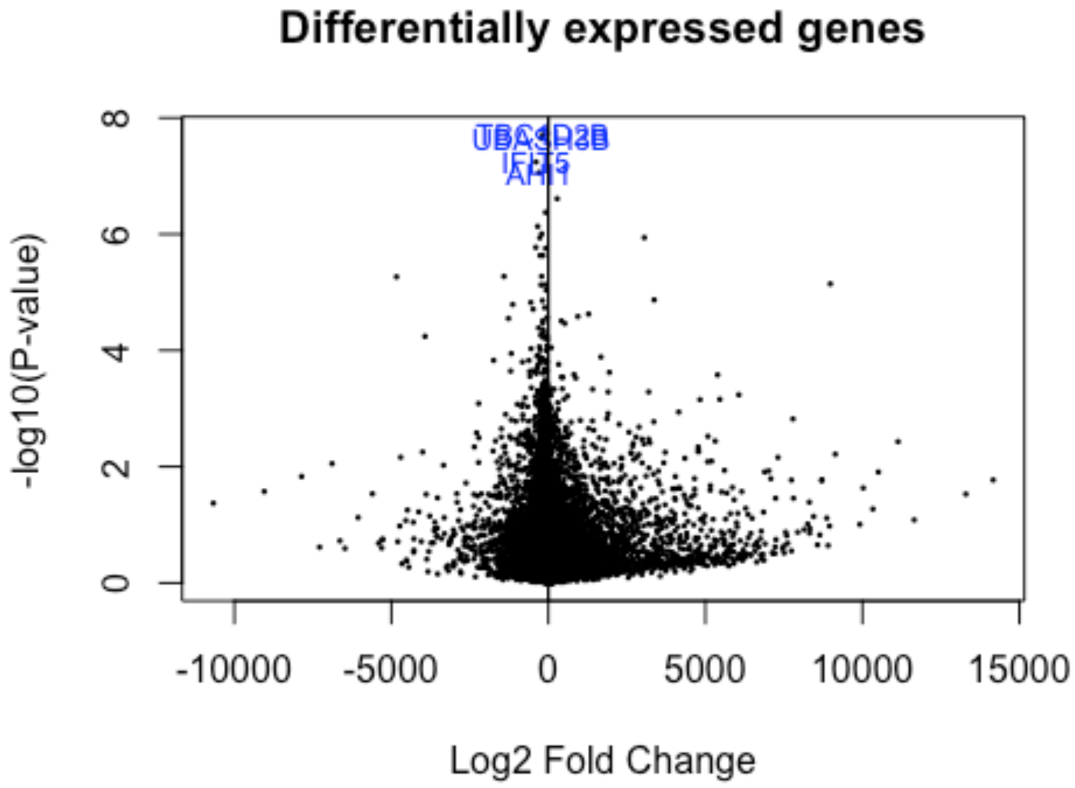
```
require(hgu133plus2.db)
```

```
## Loading required package: hgu133plus2.db
```

```
## Loading required package: AnnotationDbi
```

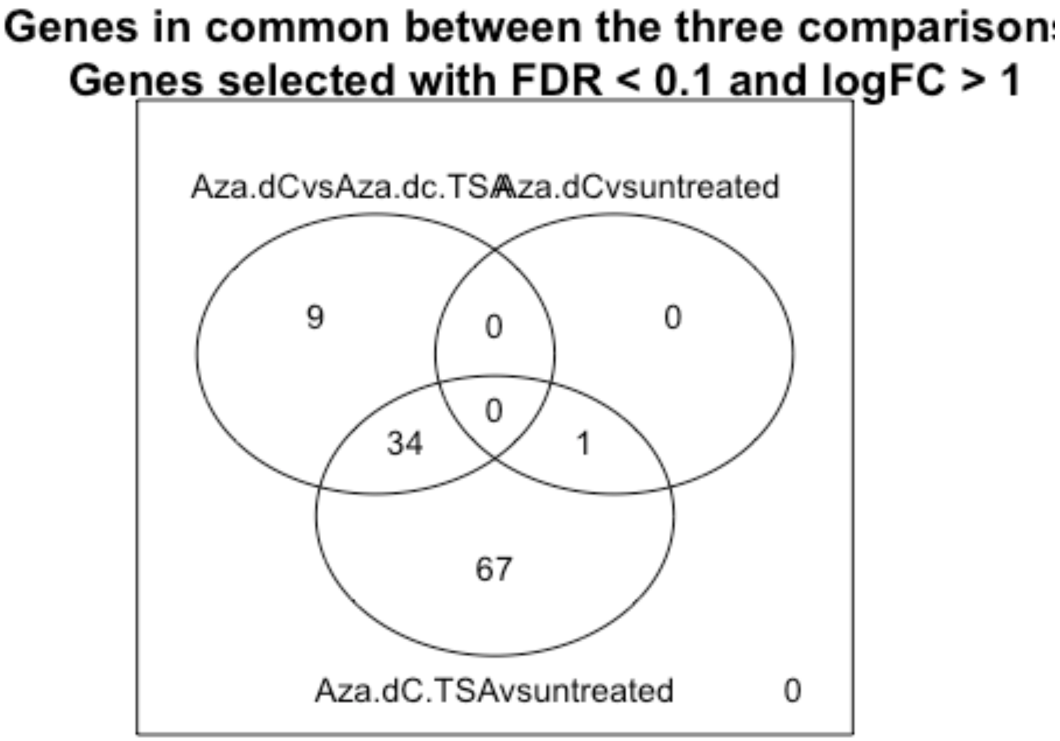
```
## Loading required package: stats4
```

```
## Loading required package: IRanges
## Loading required package: S4Vectors

##
## Attaching package: 'S4Vectors'

## The following object is masked from 'package:base':
##
##     expand.grid

## Loading required package: org.Hs.eg.db

##

##

topAnnotated_topTab_Aza.dCvsAza.dc.TSA <-
annotatedTopTable(topTab_Aza.dCvsAza.dc.TSA,
                                          anotPackage="hgu133plus2.db")

## 'select()' returned 1:many mapping between keys and columns

topAnnotated_topTab_Aza.dCvsuntreated <-
annotatedTopTable(topTab_Aza.dCvsuntreated,
 anotPackage="hgu133plus2.db")

## 'select()' returned 1:many mapping between keys and columns

topAnnotated_topTab_Aza.dC.TSAvsuntreated <-
annotatedTopTable(topTab_Aza.dC.TSAvsuntreated,
 anotPackage="hgu133plus2.db")

## 'select()' returned 1:many mapping between keys and columns

topAnnotated_topTab_INT <- annotatedTopTable(topTab_INT,
 anotPackage="hgu133plus2.db")

## 'select()' returned 1:many mapping between keys and columns

library(hgu133plus2.db)
geneSymbols <- select(hgu133plus2.db, rownames(fit.main), c("SYMBOL"))

## 'select()' returned 1:many mapping between keys and columns

SYMBOLS<- geneSymbols$SYMBOL
volcanoplot(fit.main, coef=1, highlight=4, names=SYMBOLS,
            main=paste("Differentially expressed genes"))
abline(v=c(-1,1))
```



##Multiples comparaciones

```
library(limma)
res<-decideTests(fit.main, method="separate", adjust.method="fdr",
p.value=0.1, lfc=1)

sum.res.rows<-apply(abs(res),1,sum)
res.selected<-res[sum.res.rows!=0,]
print(summary(res))

##        Aza.dCvsAza.dc.TSA Aza.dCvsuntreated Aza.dC.TSAvsuntreated   INT
## Down                   35                 1                    29    73
## NotSig              54632             54674                 54573 54573
## Up                      8                 0                    73    29

vennDiagram (res.selected[,1:3], cex=0.9)
title("Genes in common between the three comparisons\n Genes selected with
FDR < 0.1 and logFC > 1")
```



Analisis de signifacion biologica

```
listOfTables <- list(Aza.dCvsAza.dc.TSA = topTab_Aza.dCvsAza.dc.TSA,
                     Aza.dCvsuntreated  = topTab_Aza.dCvsuntreated,
                     Aza.dC.TSAvsuntreated = topTab_Aza.dC.TSAvsuntreated,
                     INT = topTab_INT)
listOfSelected <- list()
for (i in 1:length(listOfTables)){
topTab <- listOfTables[[i]]
whichGenes<-topTab["adj.P.Val"]<0.15
selectedIDs <- rownames(topTab)[whichGenes]
EntrezIDs<- select(hgu133plus2.db, selectedIDs, c("ENTREZID"))
EntrezIDs <- EntrezIDs$ENTREZID
listOfSelected[[i]] <- EntrezIDs
names(listOfSelected)[i] <- names(listOfTables)[i]
}

## 'select()' returned 1:many mapping between keys and columns

## 'select()' returned 1:1 mapping between keys and columns

## 'select()' returned 1:many mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
```

```
sapply(listOfSelected, length)
```

```
##    Aza.dCvsAza.dc.TSA     Aza.dCvsuntreated Aza.dC.TSAvsuntreated
##                    66                     1                   213
##                   INT
##                   213
```

```
library(clusterProfiler)
```

```
##
```

```
## clusterProfiler v3.16.0  For help:
https://guangchuangyu.github.io/software/clusterProfiler
##
## If you use clusterProfiler in published research, please cite:
## Guangchuang Yu, Li-Gen Wang, Yanyan Han, Qing-Yu He. clusterProfiler: an R
package for comparing biological themes among gene clusters. OMICS: A Journal
of Integrative Biology. 2012, 16(5):284-287.
```

```
##
## Attaching package: 'clusterProfiler'
```

```
## The following object is masked from 'package:AnnotationDbi':
##
##     select
```

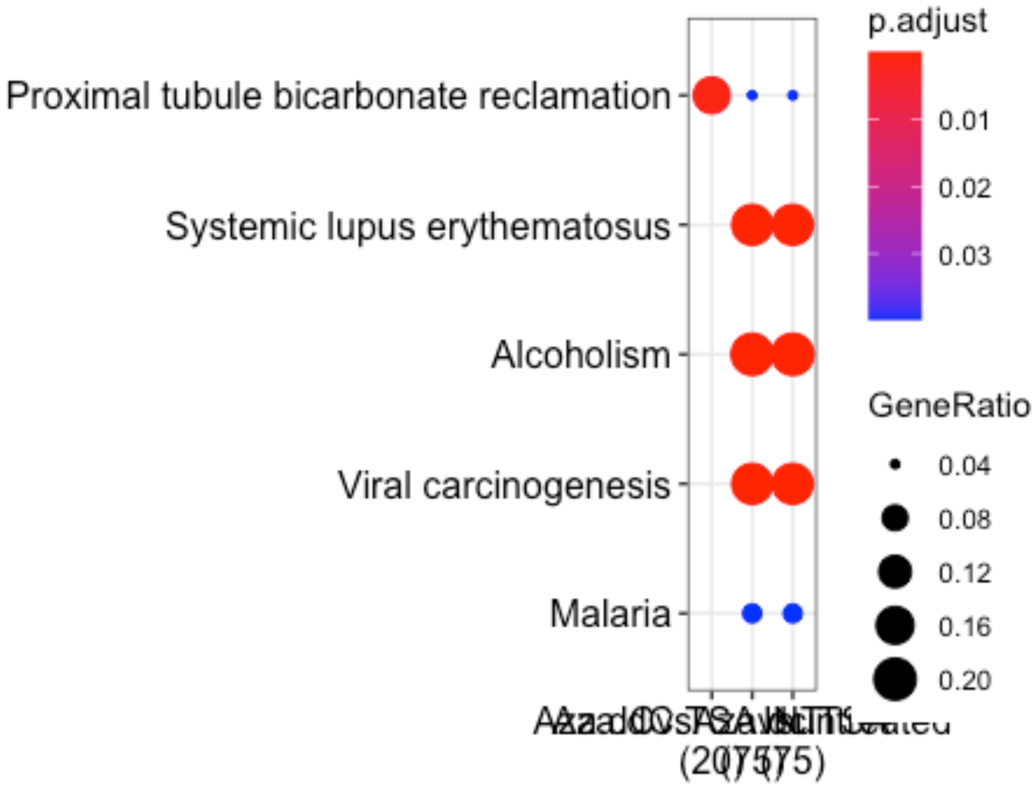
```
## The following object is masked from 'package:IRanges':
##
##     slice
```

```
## The following object is masked from 'package:S4Vectors':
##
##     rename
```

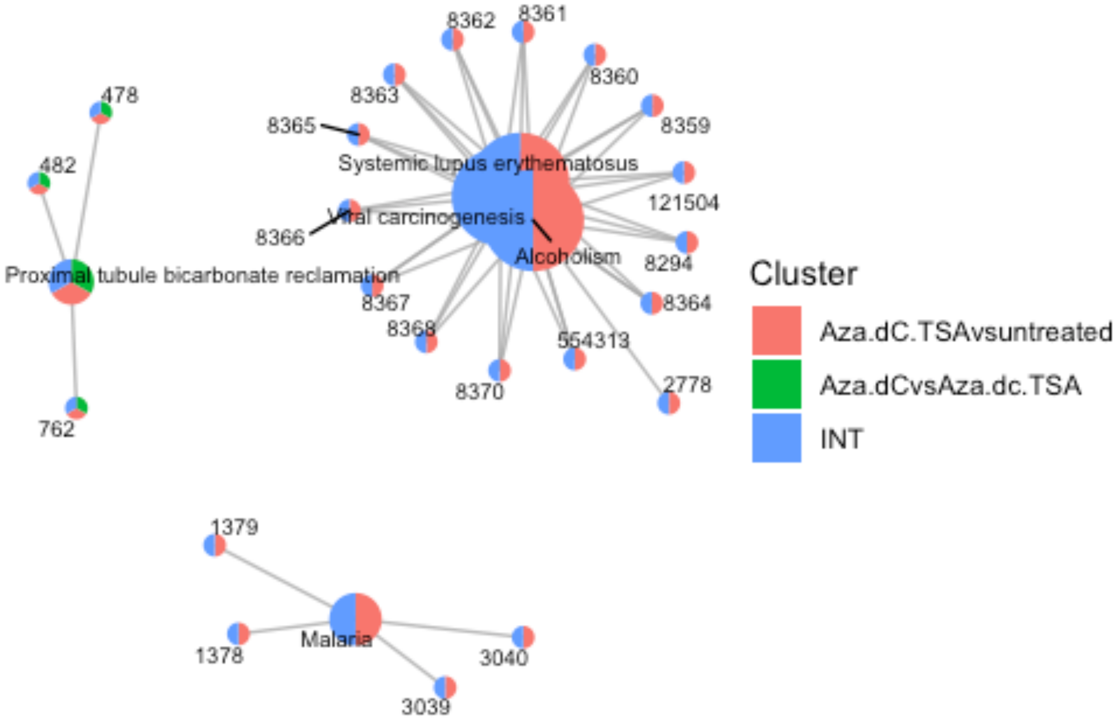
```
## The following object is masked from 'package:stats':
##
##     filter
```

```
ck <- compareCluster(geneCluster = listOfSelected, fun = "enrichKEGG")
```

```
dotplot(ck)
```



```
cnetplot(ck, categorySize = "geneNum", schowCategory = 15,
        vertex.label.cex = 0.75)
```


8362
8361
8360
8363
478
8359
8365
482
Systemic lupus erythematosus
121504
Viral carcinogenesis
8366
Alcoholism
8294
Proximal tubule bicarbonate reclamation
8367
8364
8368
554313
8370
2778
762
Cluster
Aza.dC.TSAvsuntreated
Aza.dCvsAza.dc.TSA
INT
1379
Malaria
1378
3040
3039