

P vs NP Z-score

HBG

11/5/2018

Comparisons of Z-scores in tumor (P) and normal (NP) cells

```
### this block perform z-score calculations
library("microbenchmark")
library("matrixStats")

## the GO terms for biological processes (BP)
GOcategory.file <- read.csv("../Data/human.bp.term.csv",header=TRUE, stringsAsFactors=F)
bp.go.cat <- GOcategory.file$GO.term
bp.dim <- length(bp.go.cat)
conn.dim <- 20
hspin <- matrix(as.numeric(unlist(read.table("human.rnf43.NP.bp.txt", header=F, sep=","))), nrow=bp.dim,
obs <- c(hspin)

perm <- c()
for (i in 1:100) {
  name <- paste("ms02.human", "/", "rnf43.NP", "/", "ms02.", i, ".bp.matrix.csv", sep="")
  mat <- matrix(as.numeric(unlist(read.table(name, header=F, sep=","))), nrow=bp.dim, ncol=conn.dim)
  perm <- rbind(perm, c(mat))
}

mean <- colMeans(perm)
std <- colSds(perm)

zscore <- round((obs - mean)/std, 3)

z.np <- matrix(zscore, nrow=bp.dim, ncol=conn.dim)

library('gplots')

##
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
##
##      lowess

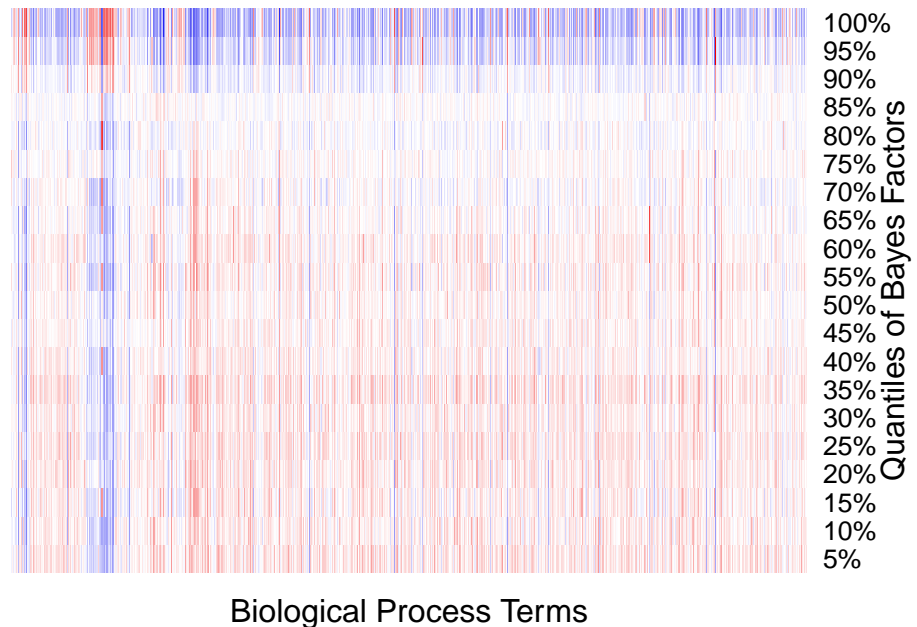
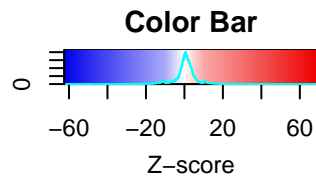
z.np <- t(z.np)

rownames(z.np) <- c("5%", "10%", "15%", "20%", "25%", "30%", "35%", "40%", "45%", "50%",
"55%", "60%", "65%", "70%", "75%", "80%", "85%", "90%", "95%", "100%")

colnames(z.np) <- bp.go.cat

colors = c(seq(min(z.np), -10.1, length=100), seq(-9.9, 9.9, length=100), seq(10.1, max(z.np), length=100))
my_palette <- colorRampPalette(c("blue2", "white", "red2"))(n = 299)
```

```
#png(filename = "16.PvsNP.z.NP.hm.png",width=6, height=5.5, res=1200, unit="in")
heatmap.2(z.np, col=my_palette, breaks=colors, dendrogram='none', Rowv=F, Colv=F,
  trace='none', offsetRow = 0, offsetCol = 0,
  xlab="Biological Process Terms", ylab="Quantiles of Bayes Factors",
  margins = c(2,3.5), key.title = "Color Bar", key.xlab="Z-score", key.ylab=NA,
  revC = T,
  labCol = NA, #labRow =,
  #srtCol=45, adjCol=c(1,0),
  #lmat=rbind(c(0,3,4), c(2,1,0)), lwid=c(1.5,4,2),
  scale="none", symbreaks=T, symm=F, symkey = F)
```



```
#dev.off()
```

```
#correlations to quantile 20 (95%-100%) genes
```

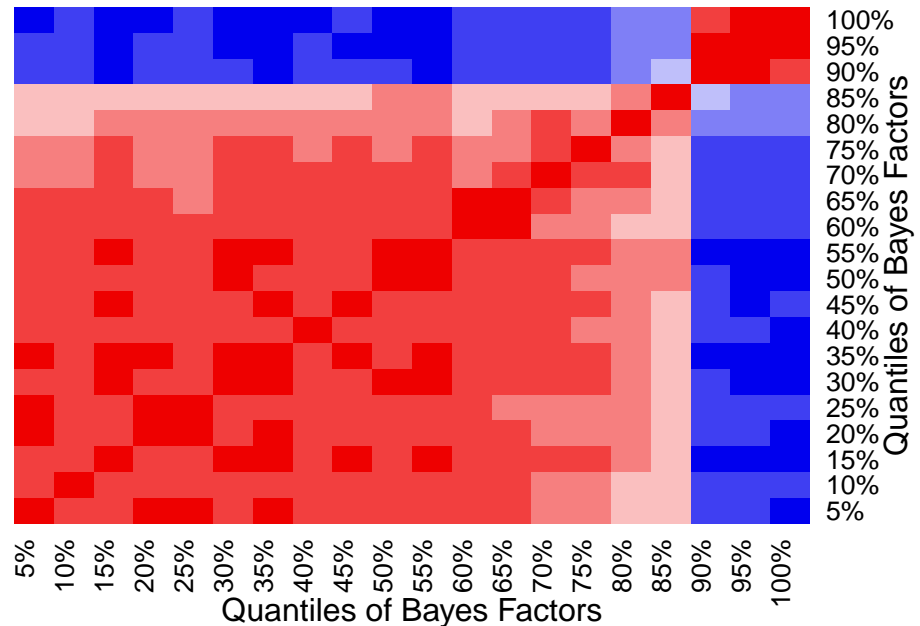
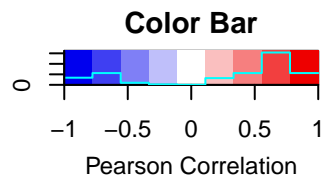
```
y.val <- c()
for (i in 1:20) {
  y.val2 <- c()
  for (j in 1:20) {
    cor.val <- cor.test(z.np[j,], z.np[i,])
    y.val2 <- rbind(y.val2, cor.val$estimate)
  }
  y.val <- cbind(y.val, y.val2)
}
```

```
rownames(y.val) <- c("5%", "10%", "15%", "20%", "25%", "30%", "35%", "40%", "45%", "50%",
  "55%", "60%", "65%", "70%", "75%", "80%", "85%", "90%", "95%", "100%")
colnames(y.val) <- c("5%", "10%", "15%", "20%", "25%", "30%", "35%", "40%", "45%", "50%",
```

```

        "55%", "60%", "65%", "70%", "75%", "80%", "85%", "90%", "95%", "100%")
my_palette2 <- colorRampPalette(c("blue2", "white", "red2"))(9)
#png(filename = "16.PvsNP.z.NP.cor.matrix.png", width=6, height=5.5, res=1200, unit="in")
heatmap.2(y.val, col=my_palette2, dendrogram='none', Rowv=F, Colv=F,
  trace='none', offsetRow = 0, offsetCol = 0,
  xlab="Quantiles of Bayes Factors", ylab="Quantiles of Bayes Factors",
  margins = c(3.5,3.5), key.title = "Color Bar", key.xlab="Pearson Correlation", key.ylab=NA,
  revC = T,
  #labCol = NA, #labRow =,
  #srtCol=45, adjCol=c(1,0),
  #lmat=rbind(c(0,3,4), c(2,1,0)), lwid=c(1.5,4,2),
  scale="none", symbreaks=T, symm=F, symkey = F)

```



```
#dev.off()
```

```
### this block perform z-score calculations
```

```
library("microbenchmark")
```

```
library("matrixStats")
```

```
## the GO terms for biological processes (BP)
```

```
GOcategory.file <- read.csv("../Data/human.bp.term.csv", header=TRUE, stringsAsFactors=F)
```

```
bp.go.cat <- GOcategory.file$GO.term
```

```
bp.dim <- length(bp.go.cat)
```

```
conn.dim <- 20
```

```
hspin <- matrix(as.numeric(unlist(read.table("human.rnf43.all.bp.txt", header=F, sep=","))), nrow=bp.dim,
```

```
obs <- c(hspin)
```

```

perm <- c()
for (i in 1:100) {
  name <- paste("ms02.human", "/", "rnf43.heatmap", "/", "ms02.", i, ".bp.matrix.csv", sep="")
  mat <- matrix(as.numeric(unlist(read.table(name, header=F, sep=","))), nrow=bp.dim, ncol=conn.dim)
  perm <- rbind(perm, c(mat))
}

mean <- colMeans(perm)
std <- colSds(perm)

zscore <- round((obs - mean)/std, 3)

z.p <- matrix(zscore, nrow=bp.dim, ncol=conn.dim)

library('gplots')
z.p <- t(z.p)

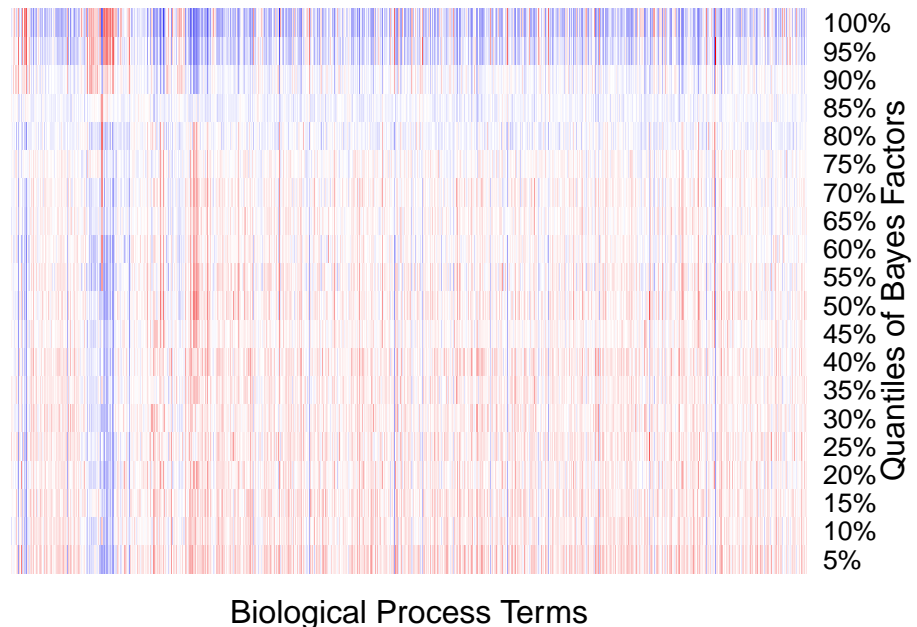
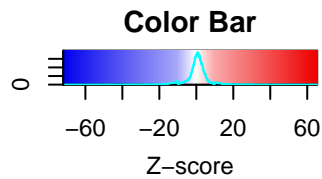
rownames(z.p) <- c("5%", "10%", "15%", "20%", "25%", "30%", "35%", "40%", "45%", "50%",
  "55%", "60%", "65%", "70%", "75%", "80%", "85%", "90%", "95%", "100%")

colnames(z.p) <- bp.go.cat

colors = c(seq(min(z.p), -10.1, length=100), seq(-9.9, 9.9, length=100), seq(10.1, max(z.p), length=100))
my_palette <- colorRampPalette(c("blue2", "white", "red2"))(n = 299)

#png(filename = "16.PvsNP.z.P.hm.png", width=6, height=5.5, res=1200, unit="in")
heatmap.2(z.p, col=my_palette, breaks=colors, dendrogram='none', Rowv=F, Colv=F,
  trace='none', offsetRow = 0, offsetCol = 0,
  xlab="Biological Process Terms", ylab="Quantiles of Bayes Factors",
  margins = c(2,3.5), key.title = "Color Bar", key.xlab="Z-score", key.ylab=NA,
  revC = T,
  labCol = NA, #labRow =,
  #srtCol=45, adjCol=c(1,0),
  #lmat=rbind(c(0,3,4), c(2,1,0)), lwid=c(1.5,4,2),
  scale="none", symbreaks=T, symm=F, symkey = F)

```

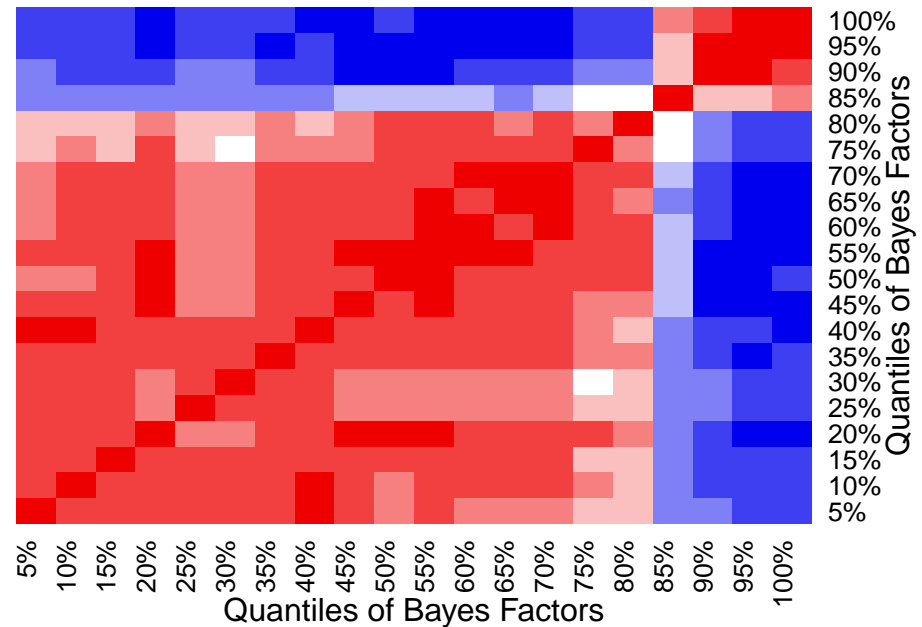
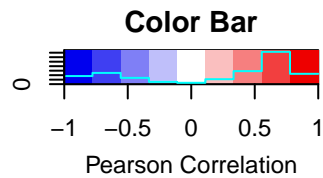


```
#dev.off()

#correlations to quantile 20 (95%-100%) genes
y.val <- c()
for (i in 1:20) {
  y.val2 <- c()
  for (j in 1:20) {
    cor.val <- cor.test(z.p[j,], z.p[i,])
    y.val2 <- rbind(y.val2, cor.val$estimate)
  }
  y.val <- cbind(y.val, y.val2)
}

rownames(y.val) <- c("5%", "10%", "15%", "20%", "25%", "30%", "35%", "40%", "45%", "50%",
  "55%", "60%", "65%", "70%", "75%", "80%", "85%", "90%", "95%", "100%")
colnames(y.val) <- c("5%", "10%", "15%", "20%", "25%", "30%", "35%", "40%", "45%", "50%",
  "55%", "60%", "65%", "70%", "75%", "80%", "85%", "90%", "95%", "100%")
my_palette2 <- colorRampPalette(c("blue2", "white", "red2"))(9)
#png(filename = "16.PusNP.z.P.cor.matrix.png", width=6, height=5.5, res=1200, unit="in")
heatmap.2(y.val, col=my_palette2, dendrogram='none', Rowv=F, Colv=F,
  trace='none', offsetRow = 0, offsetCol = 0,
  xlab="Quantiles of Bayes Factors", ylab="Quantiles of Bayes Factors",
  margins = c(3.5,3.5), key.title = "Color Bar", key.xlab="Pearson Correlation", key.ylab=NA,
  revC = T,
  #labCol = NA, #labRow = ,
  #srtCol=45, adjCol=c(1,0),
  #lmat=rbind(c(0,3,4), c(2,1,0)), lwid=c(1.5,4,2),
```

```
scale="none", symbreaks=T, symm=F, symkey = F)
```



```
#dev.off()
```

```
library(GO.db)
```

```
## Loading required package: AnnotationDbi
## Loading required package: stats4
## 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':
##
##   anyDuplicated, append, as.data.frame, basename, cbind,
```

```

##      colMeans, colnames, colSums, dirname, do.call, duplicated,
##      eval, evalq, Filter, Find, get, grep, grepl, intersect,
##      is.unsorted, lapply, lengths, Map, mapply, match, mget, order,
##      paste, pmax, pmax.int, pmin, pmin.int, Position, rank, rbind,
##      Reduce, rowMeans, rownames, rowSums, sapply, setdiff, sort,
##      table, tapply, union, unique, unsplit, which, which.max,
##      which.min

## Loading required package: Biobase

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

##
## Attaching package: 'Biobase'

## The following objects are masked from 'package:matrixStats':
##
##      anyMissing, rowMedians

## Loading required package: IRanges

## Loading required package: S4Vectors

##
## Attaching package: 'S4Vectors'

## The following object is masked from 'package:gplots':
##
##      space

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

##
p.names <- names(z.p[20, which(z.p[20,] > 3)])
np.names <- names(z.np[20, which(z.np[20,] > 3)])
#p.names
#np.names
'%nin%' <- Negate('%in%')
p.specific.terms <- p.names[which(p.names %nin% np.names)]
p.q20 <- c("GO.ID", "GO.Term", "Z.Tumor", "Z.Normal")
for (i in 1:length(p.specific.terms)) {
  id <- as.character(p.specific.terms[i])
  term <- Term(GOID(id))
  seri <- which(bp.go.cat %in% id)
  z.tumor <- z.p[20,seri]
  z.normal <- z.np[20,seri]
  p.q20 <- rbind(p.q20, c(id, term, z.tumor, z.normal))
}
p.q20

##              GO:0006890
## p.q20 "GO.ID"          "GO.Term"
##      "GO:0006890" "retrograde vesicle-mediated transport, Golgi to ER"

```

```

##      "GO:0007030" "Golgi organization"
##      "GO:0008652" "cellular amino acid biosynthetic process"
##      "GO:0010043" "response to zinc ion"
##      "GO:0015991" "ATP hydrolysis coupled proton transport"
##      "GO:0016192" "vesicle-mediated transport"
##      "GO:0016241" "regulation of macroautophagy"
##      "GO:0032008" "positive regulation of TOR signaling"
##      "GO:0032781" "positive regulation of ATPase activity"
##      "GO:0033572" "transferrin transport"
##      "GO:0043312" "neutrophil degranulation"
##      "GO:0045454" "cell redox homeostasis"
##      "GO:0045540" "regulation of cholesterol biosynthetic process"
##      "GO:0045648" "positive regulation of erythrocyte differentiation"
##      "GO:0050821" "protein stabilization"
##
## p.q20 "Z.Tumor" "Z.Normal"
##      "5.891"      "1.758"
##      "3.117"      "-0.006"
##      "5.154"      "2.853"
##      "3.025"      "1.649"
##      "6.554"      "2.113"
##      "4.109"      "0.105"
##      "3.269"      "-1.876"
##      "3.171"      "-1.486"
##      "3.029"      "1.821"
##      "5.408"      "0.851"
##      "3.505"      "0.389"
##      "3.64"       "2.443"
##      "3.114"      "1.565"
##      "4.026"      "2.486"
##      "4.114"      "2.46"

```

```

write.table(p.q20, file="16.PvsNP.q20.Pspecific.csv", sep="\t", col.names = F, row.names = F, quote=F)

```