

Zadanie 10 - Enrichment analysis (wykres słupkowy)

Ksenia Kvitko

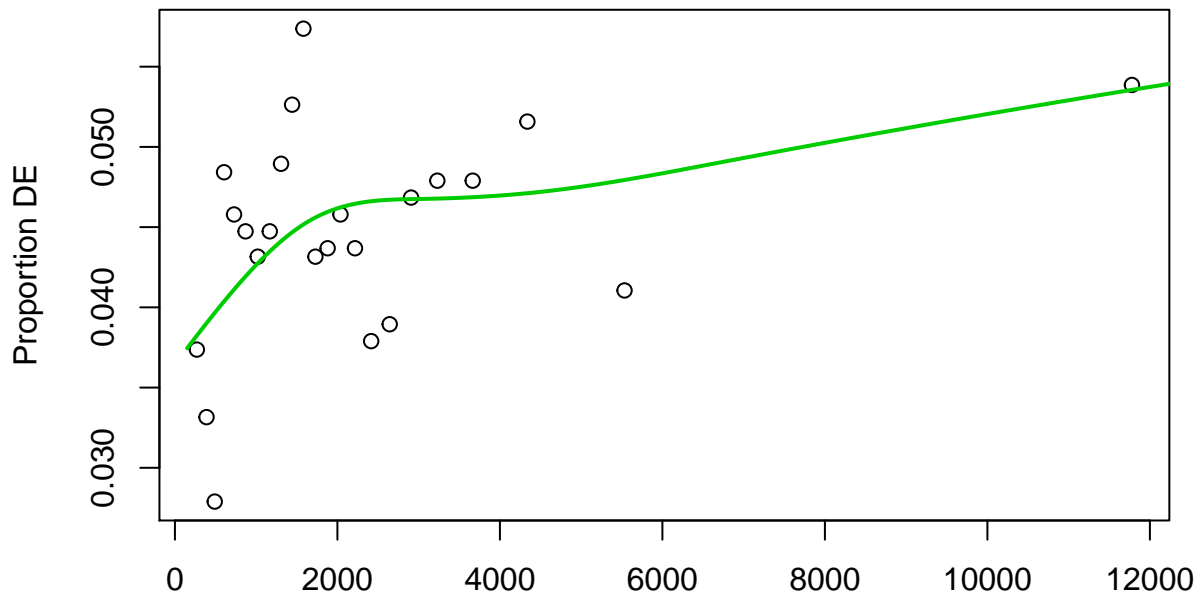
22 06 2020

1. Ładowanie niezbędnych bibliotek

```
# biblioteka ggplot2 do wykresów
library(ggplot2)
# biblioteki GO
library(goseq)
library(topGO)
# biblioteki operacji na danych
library(tidyr)
library(stringr)
```

2. Analiza GO

```
# --- KOD Z ĆWICZEŃ --- START
dane <- read.csv("/Dydaktyka/counts.txt", skip = 1, sep = "\t")
gene.bad.id <- dane$Geneid
geneID2GO <- readMappings(file = '/Dydaktyka/geneid2go.map2')
r <- read.csv('/Dydaktyka/DE_wyniki.csv', row.names = 1)
gene.bad.id <- dane$Geneid
popraw.nazwy <- function(gen.id){
  dobry.gen.id <- strsplit(as.character(gen.id), split = ".", fixed = T)[[1]][1]
  return(dobry.gen.id)
}
r1 <- as.data.frame(r)
de.gene <- row.names(r1)
de.gene <- sapply(de.gene, popraw.nazwy)
assigned.gene <- sapply(gene.bad.id, popraw.nazwy)
gene.vector <- as.integer(assigned.gene%in%de.gene)
names(gene.vector) <- assigned.gene
length.data <- dane$Length
pwf <- nullp(gene.vector, "unnOrg", "unnGen", length.data)
```



Biased Data in 1900 gene bins.

```
G0.wall <- goseq(pwf,"unnOrg","unnGen", gene2cat = geneID2G0, test.cats = c("GO:CC", "GO:BP", "GO:MF"),
G0.wall.CC <- G0.wall[G0.wall$ontology == 'CC',]
enriched.GO <- G0.wall.CC$category[p.adjust(G0.wall.CC$over_represented_pvalue, method = "BH") < .05]
G0.wall1.CC <- G0.wall.CC[G0.wall.CC$category%in%enriched.GO,]
G0.wall1.CC$FDR <- p.adjust(G0.wall.CC$over_represented_pvalue,method = "BH")[p.adjust(G0.wall.CC$over_
G0.wall1.CC <- G0.wall1.CC[!is.na(G0.wall1.CC$FDR),]
G0.wall.MF <- G0.wall[G0.wall$ontology == 'MF',]
enriched.GO <- G0.wall.MF$category[p.adjust(G0.wall.MF$over_represented_pvalue,method = "BH")<.05]
G0.wall1.MF <- G0.wall.MF[G0.wall.MF$category%in%enriched.GO,]
G0.wall1.MF$FDR <- p.adjust(G0.wall.MF$over_represented_pvalue,method = "BH")[p.adjust(G0.wall.MF$over_
G0.wall1.MF <- G0.wall1.MF[!is.na(G0.wall1.MF$FDR),]
G0.wall.BP <- G0.wall[G0.wall$ontology == 'BP',]
enriched.GO <- G0.wall.BP$category[p.adjust(G0.wall.BP$over_represented_pvalue, method = "BH")<.05]
G0.wall1.BP <- G0.wall.BP[G0.wall.BP$category%in%enriched.GO,]
G0.wall1.BP$FDR <- p.adjust(G0.wall.BP$over_represented_pvalue,method="BH")[p.adjust(G0.wall.BP$over_rep
G0.wall1.BP <- G0.wall1.BP[!is.na(G0.wall1.BP$FDR),]
# --- KOD Z ĆWICZEŃ ---
```

2. Rozwiązanie zadania - stworzenie wykresu słupkowego ilustrującego wyniki Enrichment analysis

Opracowanie danych

```
polaczone <- rbind(G0.wall1.BP, G0.wall1.MF, G0.wall1.CC)

row.names(r) <- de.gene
r <- cbind(rownames(r), data.frame(r, row.names = NULL))
colnames(r) <- c('loc','logFC','logCPM','PValue','FDR')
geny <- data.frame(unlist(geneID2G0, use.names = TRUE))
genId <- cbind(rownames(geny), data.frame(geny, row.names = NULL))
colnames(genId) <- c('loc','GO')
```

```

genId$loc <- substr(genId$loc,1,11)
genyScalone <- merge(r,genId)
genyScalone <- na.omit(genyScalone)

polaczone$plus <- 0
polaczone$minus <- 0
for (m in 1:length(genyScalone$logFC)){
  if (genyScalone$logFC[m] > 0) {
    for (n in 1:length(polaczone$category)){
      if (polaczone$category[n] == genyScalone$GO[m]){
        polaczone$plus[n] <- polaczone$plus[n] + 1
      }
    }
  } else if (genyScalone$logFC[m] <= 0){
    for (n in 1:length(polaczone$category)){
      if (polaczone$category[n] == genyScalone$GO[m]){
        polaczone$minus[n] <- polaczone$minus[n] + 1
      }
    }
  }
}

plotData <- data.frame(polaczone$category, polaczone$ontology, polaczone$plus, polaczone$minus)
colnames(plotData) <- c("category", "ontology", "Up", "Down")
plotData$DGEs_Number <- 0
plotData <- rbind(plotData, plotData)
plotData$DGE <- ''
for (s in 1:(length(plotData$category)/2)){
  plotData[s,5] <- plotData[s,3]
  plotData[s,6] <- "Up"
  s <- s+length(plotData$category)/2
  plotData[s,5] <- plotData[s,4]
  plotData[s,6] <- "Down"
}

```

Wygenerowanie wykresu

```

ggplot(data = plotData, aes(x = category, y = DGEs_Number, fill = DGE)) +
  geom_bar(stat="identity", position=position_dodge()) +
  coord_flip() +
  theme_minimal() +
  scale_fill_manual("Regulation", values = c("Up" = "red", "Down" = "blue")) +
  ylab("DGEs Number") +
  xlab("Category") +
  ggtitle("DEGs Number of The Most Enriched GO Term edgeR") +
  geom_text(aes(label = DGEs_Number), position=position_dodge(width=0.1), size = 2) +
  facet_grid(ontology ~ ., scales = "free")

```

