GO Enrichment

```
## Libraries
library(tidyr)
library(goseq)
library(GO.db)
library(yaml)
library(rmarkdown)

## Read in YAML guide
### Set Working Directory
rstudioapi::getActiveDocumentContext
```

```
## function ()
## {
##     context <- callFun("getActiveDocumentContext")
##     context$selection <- as.document_selection(context$selection)
##     structure(context, class = "document_context")
## }
## <environment: namespace:rstudioapi>
```

```
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

## Read in sample names from yaml
yamls <- yaml.load_file("de.yml")
sample1 <- yamls$sample1
sample2 <- yamls$sample2

sample1
```

```
## [1] "wtbmbr"
```

```
sample2
```

```
## [1] "wtbother"
```

```
## Render
render("GO_basedOnSkeletonGO.Rmd", "pdf_document", output_file = paste(sample1,"_",sample2,"_","GO.pdf"
```

Setting up the DE table for GO analysis

File Input

Input the output from DE analysis. This is made for a list that includes only the significant genes.

```
sigOnly <- read.table(paste("../../../../requisiteData/data_06Sept2017/", sample1,"_",sample2,"_DE_sig.t
sigOnly$logFC <- as.numeric(as.character(sigOnly$logFC))
```

```
## Warning: NAs introduced by coercion
```

```
colnames(sigOnly)[1] <- "itag"
```

Subset

First I need to subset the list to up or down regulated, then add a new colum that specififfys 1. This column is need to for merging.

```
upITAG <- subset(sigOnly, logFC > 0, select = c(itag))
upITAG$up <- 1

downITAG <- subset(sigOnly, logFC < 0, select = c(itag))
downITAG$down <- 1

allITAG <- subset(sigOnly, select = c(itag))
allITAG$all <- 1
```

Merge I - with normalized ITAG length gene list

read in guide.

```
geneLength <- read.csv("../../../../requisiteData/normalized_genes_length.csv")

## remove trailing numbers in ITAG
geneLength$itag <- gsub("^(.*)[.].*", "\\1", geneLength$itag)
geneLength$itag <- gsub("^(.*)[.].*", "\\1", geneLength$itag)

#isolate just the gene list
genes <- subset(geneLength, select = c(itag))
```

First merge each table to geneLength

```
upITAGmerge <- merge(genes, upITAG, by = "itag", all = TRUE)
downITAGmerge <- merge(genes, downITAG, by = "itag", all = TRUE)
allITAGmerge <- merge(genes, allITAG, by = "itag", all = TRUE)
```

Merge II - Merge them all together.

```
matrixGOupdown <- merge(upITAGmerge, downITAGmerge, by = "itag", all = TRUE)
matrixGOupdownall <- merge(matrixGOupdown, allITAG, by = "itag", all = TRUE)
matrixGO <- merge(matrixGOupdownall, geneLength, by = "itag", all = TRUE)
```

Clean Up

```
matrixGO[is.na(matrixGO)] <- 0
head(matrixGO)
```

```
##             itag up down all length
## 1     92.5170068  0    0   1      0
## 2     indirectly  0    0   1      0
## 3 Solyc00g005040  0    0   0    357
## 4 Solyc00g005050  0    0   0    588
## 5 Solyc00g005060  0    0   0    273
## 6 Solyc00g005070  0    0   0     81
```

```
## This is if you want to write out the table of the GO matrix.
# write.table(matrixGO, "mydata.txt", sep="\t", quote= FALSE)
```

GO enrichment

The is the input of the GOslim categories. There are only two columns 1. itag and 2. go

```
pat <- matrixGO
head(pat)
```

```
##             itag up down all length
## 1     92.5170068  0    0   1      0
## 2     indirectly  0    0   1      0
## 3 Solyc00g005040  0    0   0    357
## 4 Solyc00g005050  0    0   0    588
## 5 Solyc00g005060  0    0   0    273
## 6 Solyc00g005070  0    0   0     81
```

```
## New GO table
cate <- read.table("../../../../requisiteData/ITAG3.2_protein_go.tsv")
colnames(cate) <- c("itag", "go")

summary(cate$itag)
```

```
## Solyc01g111990.3.1 Solyc02g079630.2.1 Solyc02g071260.3.1
##                  9                  9                  8
## Solyc03g083440.3.1 Solyc03g097290.3.1 Solyc10g044670.2.1
##                  8                  8                  8
## Solyc11g065920.2.1 Solyc11g071610.2.1 Solyc11g071620.3.1
##                  8                  8                  8
## Solyc12g008890.2.1 Solyc01g009235.1.1 Solyc01g059870.4.1
##                  8                  7                  7
## Solyc01g080460.3.1 Solyc01g088170.4.1 Solyc01g112290.3.1
##                  7                  7                  7
## Solyc04g014210.3.1 Solyc04g016430.3.1 Solyc04g076620.3.1
##                  7                  7                  7
## Solyc04g080820.2.1 Solyc05g053410.3.1 Solyc06g019170.3.1
##                  7                  7                  7
## Solyc07g008880.3.1 Solyc08g043170.3.1 Solyc09g011930.3.1
##                  7                  7                  7
## Solyc09g015240.1.1 Solyc10g017990.2.1 Solyc11g010310.2.1
##                  7                  7                  7
## Solyc11g040180.2.1 Solyc11g068830.2.1 Solyc11g071580.2.1
##                  7                  7                  7
## Solyc11g071600.2.1 Solyc11g072140.2.1 Solyc12g008900.2.1
##                  7                  7                  7
## Solyc01g080810.3.1 Solyc01g088200.3.1 Solyc01g090710.3.1
##                  6                  6                  6
## Solyc01g102410.3.1 Solyc01g103960.3.1 Solyc01g109540.3.1
##                  6                  6                  6
## Solyc02g063490.3.1 Solyc02g067930.3.1 Solyc02g068490.3.1
##                  6                  6                  6
## Solyc02g093300.3.1 Solyc03g118640.3.1 Solyc04g054890.3.1
##                  6                  6                  6
```

```
## Solyc05g009220.3.1 Solyc05g014720.3.1 Solyc07g017990.3.1
##                  6                  6                  6
## Solyc07g045480.3.1 Solyc07g062650.3.1 Solyc07g063770.3.1
##                  6                  6                  6
## Solyc07g064810.3.1 Solyc08g007420.3.1 Solyc08g061920.2.1
##                  6                  6                  6
## Solyc08g061930.3.1 Solyc08g078390.3.1 Solyc08g078400.3.1
##                  6                  6                  6
## Solyc08g078850.3.1 Solyc09g014710.3.1 Solyc09g014720.2.1
##                  6                  6                  6
## Solyc09g014730.3.1 Solyc09g014740.3.1 Solyc09g074990.3.1
##                  6                  6                  6
## Solyc09g090140.3.1 Solyc10g006710.3.1 Solyc10g079470.3.1
##                  6                  6                  6
## Solyc10g079870.2.1 Solyc11g005630.1.1 Solyc11g012140.2.1
##                  6                  6                  6
## Solyc11g013810.2.1 Solyc11g065930.2.1 Solyc12g007170.2.1
##                  6                  6                  6
## Solyc12g014180.2.1 Solyc12g019110.2.1 Solyc12g056940.2.1
##                  6                  6                  6
## Solyc00g026860.1.1 Solyc00g042130.2.1 Solyc00g055960.1.1
##                  5                  5                  5
## Solyc01g006190.3.1 Solyc01g006520.3.1 Solyc01g008330.3.1
##                  5                  5                  5
## Solyc01g073730.3.1 Solyc01g074010.3.1 Solyc01g088150.3.1
##                  5                  5                  5
## Solyc01g088160.3.1 Solyc01g088230.3.1 Solyc01g088310.3.1
##                  5                  5                  5
## Solyc01g091480.3.1 Solyc01g094500.3.1 Solyc01g094835.1.1
##                  5                  5                  5
## Solyc01g096020.3.1 Solyc01g096900.3.1 Solyc01g099620.3.1
##                  5                  5                  5
## Solyc01g102370.3.1 Solyc01g106480.3.1 Solyc01g106770.3.1
##                  5                  5                  5
## Solyc02g022930.3.1 Solyc02g038740.3.1 Solyc02g062430.3.1
##                  5                  5                  5
##            (Other)
##              31698
```

```
## remove the trailing num in itag id
cate$itag <- gsub("^(.*)[.].*", "\\1", cate$itag)
cate$itag <- gsub("^(.*)[.].*", "\\1", cate$itag)

cate <- separate(data = cate, col = go, into = c("go1", "go2", "g04", "go5", "go6", "go7", "go8", "go9"
```

```
## Warning: Too few values at 32311 locations: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
## 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...
```

```
cate <- gather(cate, itag, go1:go9, factor_key = TRUE)
colnames(cate)[3] <- "go"

## First remove rows with NA in go
cate <- cate[complete.cases(cate), ]

## Now every go term and itag pair is represented only once, so we can get rid of itag.1 column
```

```
cate <- cate[,-2]
head(cate)
```

```
##             itag         go
## 1 Solyc00g005000 GO:0004190
## 2 Solyc00g005285 GO:0008168
## 3 Solyc00g005285 GO:0008171
## 4 Solyc00g005440 GO:0003723
## 5 Solyc00g005460 GO:0003723
## 6 Solyc00g005840 GO:0045454
```

Subseting for GO analysis

Specify the column you are interested in `pat$all` refers to all the DE gene regardless if they are up or down regulated. If you want to specify down regulated, specify `pat$down`. I am going to put this into a loop, where each time the loop goes thought it will perform GO enrichment on all three types of lists of significant genes and them write them to a table.

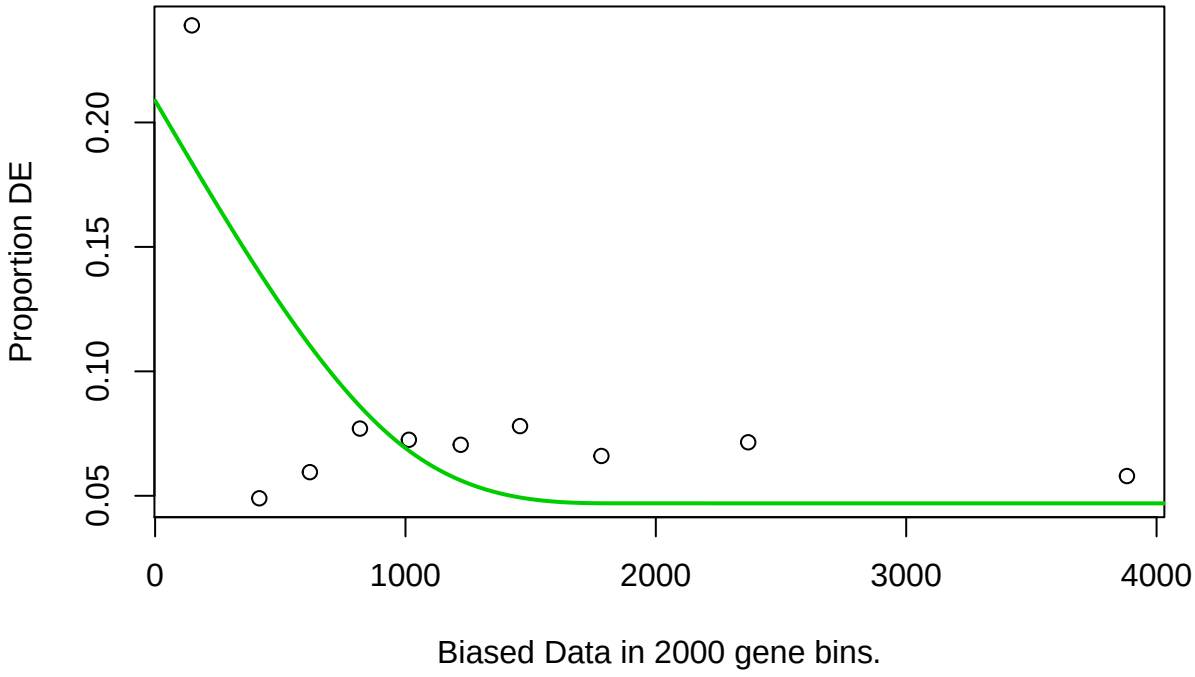
```
sigType <- c("up", "down", "all")

for (type in sigType) {
  genes = as.integer(pat[,type])
  names(genes) = pat$itag
  table(genes)
  length(genes)

  pwf = nullp(genes, bias.data = pat$length)

  GO.wall = goseq(pwf, gene2cat = cate)
  head(GO.wall)

#This is going to correct for multiple testing.  You can specify the p-value cut-off of GO categories y

  enriched.GO = GO.wall$category[p.adjust(GO.wall$over_represented_pvalue, method = "BH") < 0.05]

  enriched.GO

  my.GO <- as.character(enriched.GO)
  my.GO.table <- Term(my.GO)
  my.GO.table
  t <- as.matrix(my.GO.table)

  print(type) #this is for the knitr document
  print(t) #this is for the knitr document

  write.table(t, file = paste(sample1,"_",sample2,"DE1_sigonly_",type,"_GO.txt", sep = ""))
write.table(GO.wall, file = paste(sample1,"_",sample2,"DE1_sigValues_",type,"_GO.txt", sep = ""))
  }
```

```
## Using manually entered categories.
```

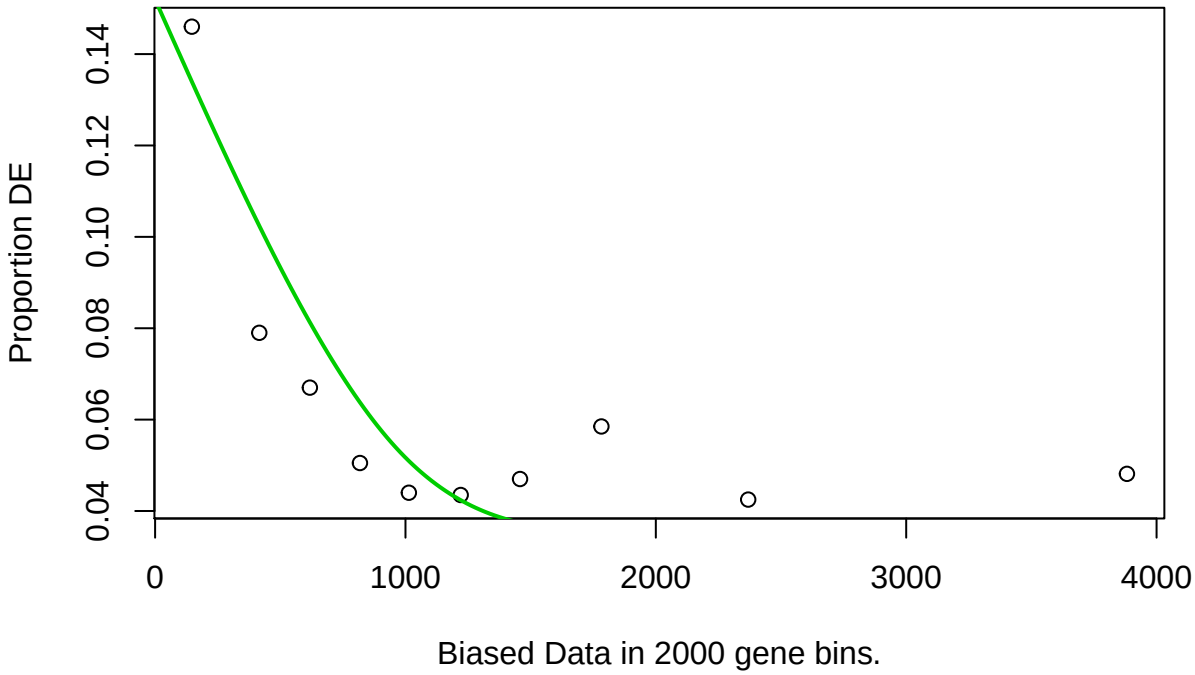
```
## For 7222 genes, we could not find any categories. These genes will be excluded.
```

```
## To force their use, please run with use_genes_without_cat=TRUE (see documentation).
```

```
## This was the default behavior for version 1.15.1 and earlier.
## Calculating the p-values...
## 'select()' returned 1:1 mapping between keys and columns
```

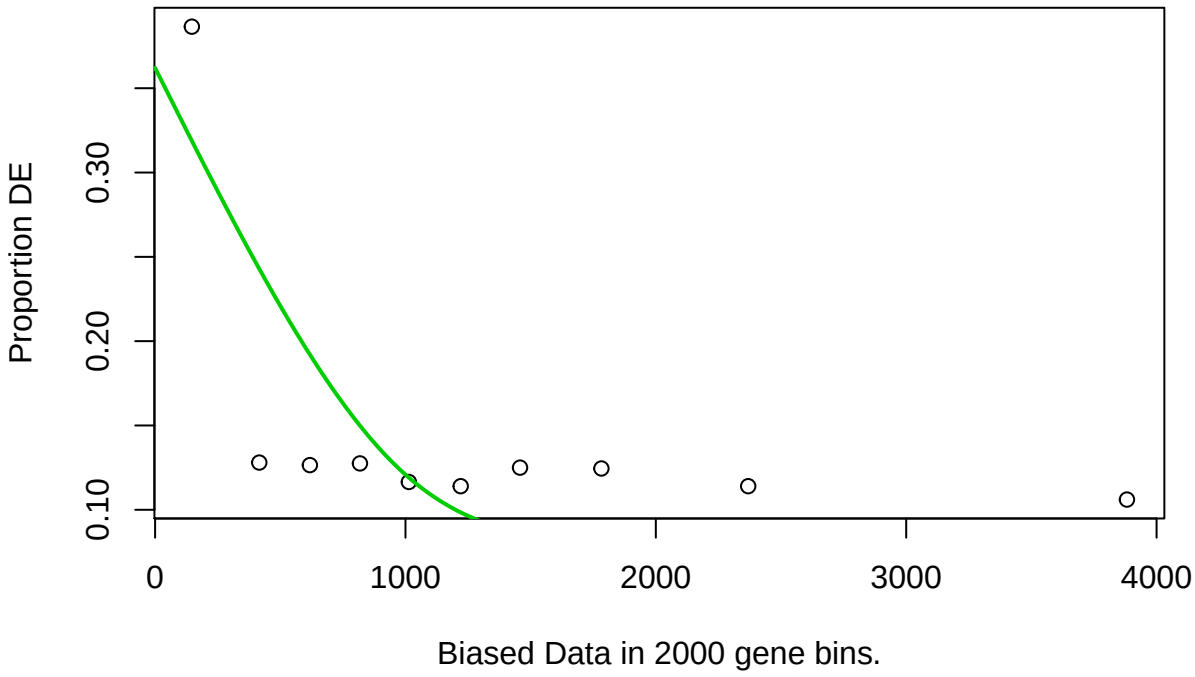



```
## [1] "up"
##            [,1]
## GO:0009765 "photosynthesis, light harvesting"
## GO:0016020 "membrane"
## GO:0005215 "transporter activity"
## GO:0055085 "transmembrane transport"
## GO:0046983 "protein dimerization activity"
## GO:0005975 "carbohydrate metabolic process"
## GO:0004553 "hydrolase activity, hydrolyzing O-glycosyl compounds"
## GO:0009664 "plant-type cell wall organization"
## GO:0004672 "protein kinase activity"
## GO:0006468 "protein phosphorylation"
## GO:0009538 "photosystem I reaction center"
## GO:0006810 "transport"
## GO:0009725 "response to hormone"
## GO:0000160 "phosphorelay signal transduction system"
## GO:0006006 "glucose metabolic process"
## GO:0000155 "phosphorelay sensor kinase activity"
## Using manually entered categories.
## For 7222 genes, we could not find any categories. These genes will be excluded.
## To force their use, please run with use_genes_without_cat=TRUE (see documentation).
## This was the default behavior for version 1.15.1 and earlier.
## Calculating the p-values...
## 'select()' returned 1:1 mapping between keys and columns
```




```
## [1] "down"
##            [,1]
## GO:0003735 "structural constituent of ribosome"
## GO:0006412 "translation"
## GO:0005840 "ribosome"
## GO:0000786 "nucleosome"
## GO:0046982 "protein heterodimerization activity"
## GO:0005622 "intracellular"
## GO:0006334 "nucleosome assembly"
## GO:0003677 "DNA binding"
## GO:0006457 "protein folding"
## GO:0005634 "nucleus"
## GO:0005852 "eukaryotic translation initiation factor 3 complex"
## GO:0003723 "RNA binding"
## GO:0042254 "ribosome biogenesis"
## GO:0003676 "nucleic acid binding"
## GO:0006414 "translational elongation"
## GO:0006364 "rRNA processing"
## GO:0003743 "translation initiation factor activity"
## GO:0008017 "microtubule binding"
## GO:0005874 "microtubule"
## GO:0003777 "microtubule motor activity"
## GO:0007018 "microtubule-based movement"
## GO:0030529 "intracellular ribonucleoprotein complex"
## GO:0006260 "DNA replication"
## GO:0004298 "threonine-type endopeptidase activity"
## GO:0005839 "proteasome core complex"
## GO:0051603 "proteolysis involved in cellular protein catabolic process"
## GO:0015935 "small ribosomal subunit"
## GO:0051082 "unfolded protein binding"
## GO:0005853 "eukaryotic translation elongation factor 1 complex"
## GO:0016747 "transferase activity, transferring acyl groups other than amino-acyl groups"
```

```
## GO:0042026 "protein refolding"
## Using manually entered categories.
## For 7222 genes, we could not find any categories. These genes will be excluded.
## To force their use, please run with use_genes_without_cat=TRUE (see documentation).
## This was the default behavior for version 1.15.1 and earlier.
## Calculating the p-values...
## 'select()' returned 1:1 mapping between keys and columns
```




```
## [1] "all"
##            [,1]
## GO:0000786 "nucleosome"
## GO:0003677 "DNA binding"
## GO:0009765 "photosynthesis, light harvesting"
## GO:0006006 "glucose metabolic process"
## GO:0005215 "transporter activity"
## GO:0006334 "nucleosome assembly"
```