Identifying over-represented GO terms

In this notebook we will identify the over-represented GO terms in the proteins previously identified as changing RNA binding. The steps in this analysis are:

1. Model the relationship between differential RNA binding and an expected bias factor
2. Identify all the over-represented GO terms
3. Filter by FDR and degree of over-representation
4. Plot
5. Compare GO terms over-represented when using `limma` or `lm`

For this analysis, we will need to source the GO.R script which contains some utility functions for dealing with GO terms.

```
source("./GO.R")
library(tidyverse)
library(goseq)
library(Hmisc)
library(limma)
library(biobroom)

# set up standardised plotting scheme
theme_set(theme_bw(base_size = 20) +
            theme(panel.grid.major=element_blank(),
                  panel.grid.minor=element_blank(),
                  aspect.ratio=1))

cbPalette <- c("#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7", "#999999")
```

Below, we load the result for differential RNA binding testing from the previous notebooks

```
rna_binding_fit <- readRDS("../results/limma_rna_binding_fit.rds")
rna_binding_fit_treat <- treat(rna_binding_fit,lfc=log2(1.5))

limma_rna_binding_changes <- topTreat(rna_binding_fit_treat, coef = "conditionG1:typeOOPS", number = In
                                      p.value=0.01, adjust.method="BH", confint=0.95)

limma_rna_binding_changes <- limma_rna_binding_changes[limma_rna_binding_changes$logFC>0,]

compare_methods <- readRDS("../results/compare_methods_rna_binding_results.rds")
```

To perform the differential RNA binding testing, we will use the R package `goseq`. This was orignally designed to account for length bias in RNA-Seq count-based differential expression testing but can be applied to any GO over-representation scenario where one expects there is a bias due to increased power to detect changes for some features. The basic idea is that the bias factor must have a monotonic relationship with the probability of a feature presenting as significantly changed. `goseq` estimates this relationship with a spline fit (we will see this later). `goseq` then takes this relationship into account when performing the GO over-representation testing. For more details about `goseq`, see here: https://genomebiology.biomedcentral.com/articles/10.1186/gb-2010-11-2-r14

In our case, we have reason to suspect that proteins with higher intensity will be more likely to be identified as having significantly altered RNA binding since the variance will be smaller so we will have more power to detect differences.

The `limma` output contains a "AveExpr" value for each protein which represents the mean intensity. This is precisely the bias factor we are after so we can use this directly.

The first step is to make a probability weight function (PWF) linking the bias factor with the probabilty of significant difference using the `nullp` function. By default, the function expects a genome name and it will go and fetch the bias data. However, we can also just provide this data ourselves. Below, we obtain the PWF and `goseq` outputs plots to show how closely these fit the observed relationship.

```
limma_sig_bool <- compare_methods$Row.names %in% rownames(limma_rna_binding_changes)
names(limma_sig_bool) <- compare_methods$Row.names
print(table(limma_sig_bool))
```

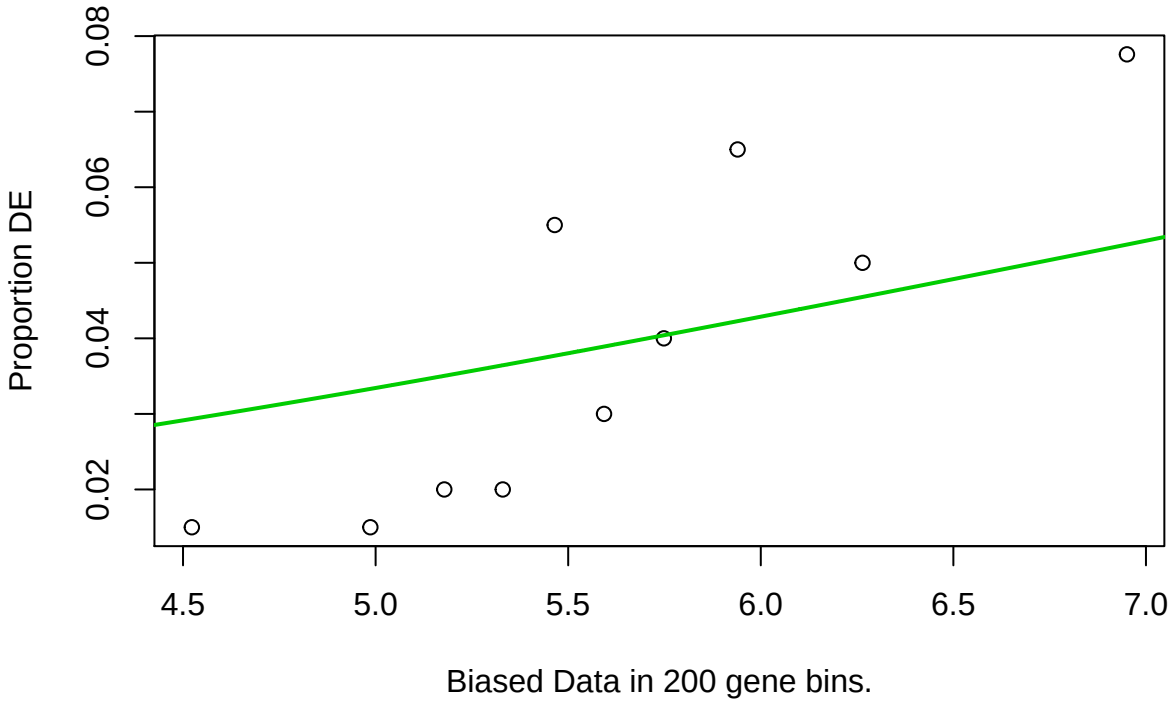
```
## limma_sig_bool
## FALSE  TRUE
##  1845    71
```

```
lm_sig_bool <- (compare_methods$lm_BH < 0.01 & compare_methods$logFC>0)
names(lm_sig_bool) <- compare_methods$Row.names
print(table(lm_sig_bool))
```

```
## lm_sig_bool
## FALSE  TRUE
##  1567   349
```

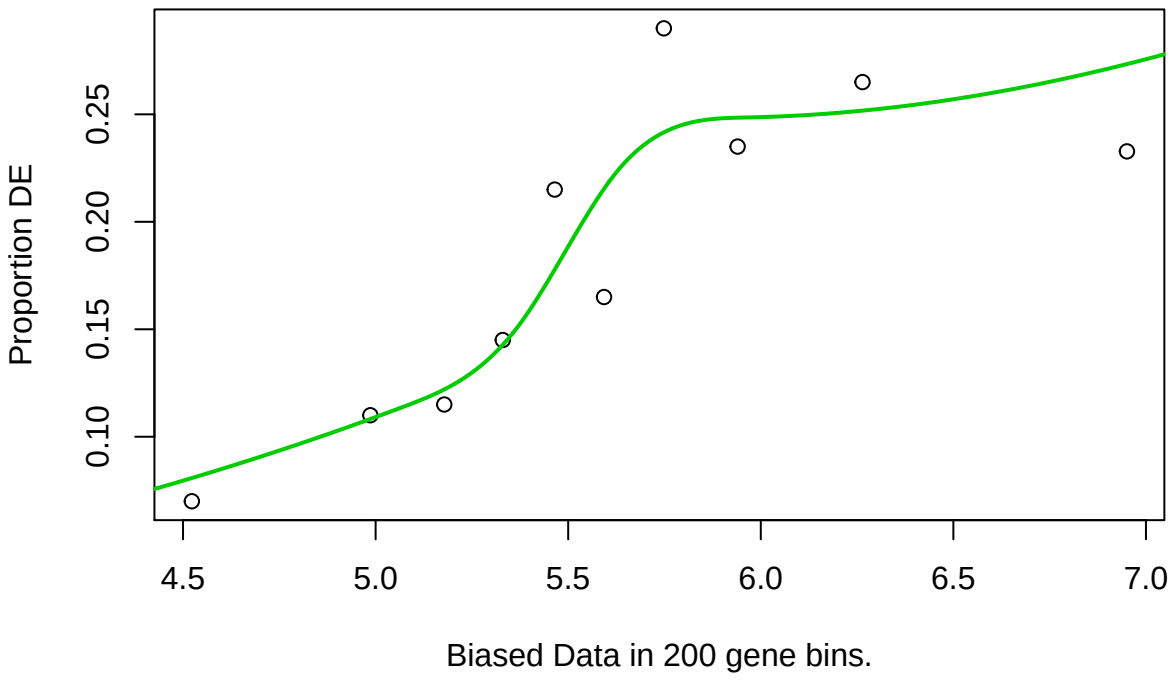
```
bias <- compare_methods$AveExpr

limma_pwf <- nullp(limma_sig_bool, bias.data=bias, plot.fit=TRUE)
```

```
## Warning in pcls(G): initial point very close to some inequality constraints
```




```
lm_pwf <- nullp(lm_sig_bool, bias.data=bias, plot.fit=TRUE)
```

```
## Warning in pcls(G): initial point very close to some inequality constraints
```



Note that the fit for the `limma` results isn't great. For some reason, `goseq` sometimes doesn't obtain a very good fit but at least it's going to account for the relationship to some extent. If the fit is really terrible, it's possible to perform the spline fitting manually and provide this to `goseq` but we won't go into that here.

Now, we need those GO terms from the previous notebook

```
sapiens.go.full <- readRDS("../results/h_sapiens_go_full.rds")
head(sapiens.go.full)
```

```
##   UNIPROTKB      GO.ID                              TERM ONTOLOGY
## 1    P09874 GO:0000002 mitochondrial genome maintenance       BP
## 2    Q9UJZ1 GO:0000002 mitochondrial genome maintenance       BP
## 3    Q9Y243 GO:0000002 mitochondrial genome maintenance       BP
## 4    Q02078 GO:0000002 mitochondrial genome maintenance       BP
## 5    Q9BUK6 GO:0000002 mitochondrial genome maintenance       BP
## 6    Q96RR1 GO:0000002 mitochondrial genome maintenance       BP
```

And we're ready to run `goseq`. By default, `goseq` expects a `genome` and will obtain the GO terms automatically. However, if we provide our own to the `gene2cat` argument, it will use this instead. This also means we could use `goseq` to interrogate over-represented KEGG terms etc by simply providing them at this point.

```
limma_over_rep_go <- goseq(limma_pwf, gene2cat=sapiens.go.full)
```

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

```
## For 14 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
```

```
lm_over_rep_go <- goseq(lm_pwf, gene2cat=sapiens.go.full)
```

```
## Using manually entered categories.

## For 14 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
```

We need to adjust the p-values to account for multiple testing ()

```
limma_over_rep_go$BH <- p.adjust(limma_over_rep_go$over_represented_pvalue, method="BH")
lm_over_rep_go$BH <- p.adjust(lm_over_rep_go$over_represented_pvalue, method="BH")
```

Questions:

- How many GO terms did we test for over-representation in the ‘limma‘ results?
- How many would we expect to have a p-value < 0.05 by chance?
- How many do?
- What can you infer from this?

We can filter the output to only include the GO terms which are over-represented at 1% FDR

```
sig_terms <- limma_over_rep_go %>% filter(BH<0.01)
print(dim(sig_terms))
```

```
## [1] 95  8
```

```
print(head(sig_terms, 10))
```

```
##       category over_represented_pvalue under_represented_pvalue numDEInCat
## 1  GO:0044281            6.262462e-17                         1         42
## 2  GO:0055114            1.732575e-15                         1         23
## 3  GO:0044429            1.226345e-13                         1         28
## 4  GO:0019752            3.117391e-13                         1         31
## 5  GO:0043436            3.577823e-13                         1         31
## 6  GO:0006082            4.120215e-13                         1         31
## 7  GO:0006091            1.889344e-12                         1         21
## 8  GO:0005739            2.958563e-12                         1         33
## 9  GO:0017144            4.091103e-12                         1         24
## 10 GO:0044282            1.959778e-11                         1         17
##    numInCat                                        term ontology
## 1       310                small molecule metabolic process       BP
## 2        83                     oxidation-reduction process       BP
## 3       159                               mitochondrial part       CC
## 4       203              carboxylic acid metabolic process       BP
## 5       204                      oxoacid metabolic process       BP
## 6       205                  organic acid metabolic process       BP
## 7        92 generation of precursor metabolites and energy       BP
## 8       252                                    mitochondrion       CC
## 9       128                          drug metabolic process       BP
## 10       62                small molecule catabolic process       BP
##                BH
## 1  7.413503e-13
## 2  1.025511e-11
## 3  4.839157e-10
## 4  8.129183e-10
## 5  8.129183e-10
## 6  8.129183e-10
## 7  3.195151e-09
## 8  4.377934e-09
## 9  5.381164e-09
## 10 2.319985e-08
```

We would also like to know the effect size, e.g how over-represented are the terms. We could use the `numDEInCat` and `numInCat` columns and calculate the over-representation as (`numInCat`/Number of proteins observed) / (`numDEInCat`/ Number of proteins with change in RNA binding). However, this would not take account of the bias we known exists.

The function below estimates the effect size, taking into account the bias (using the PWF)

```
# ------------------------------------------------------------------------------------------------
# Function  : 'addAdjustedOverRep' A crude function to add an adjusted estimate of the over-representat
# Input
#           : obj = A data frame with the results from goseq. As generated by GetEnrichedGO
#           : pwf = a PWF from goseq
#           : gene2cat = A dataframe mapping features to categories
# Output  : The input obj + a column with estimated adjusted over-representation for each term ($adj_ov
# ------------------------------------------------------------------------------------------------

addAdjustedOverRep <- function(obj, pwf, gene2cat){
  len_fore <- sum(pwf$DEgenes)
  len_back <- length(pwf$DEgenes)
```

```
  obj$adj_over_rep <- apply(obj[,c("numDEInCat", "numInCat", "category")], MARGIN=1, function(x){
    term_features <- gene2cat[gene2cat[["GO.ID"]]==x[["category"]], "UNIPROTKB"]
    term_weight <- mean(pwf[rownames(pwf) %in% term_features, "pwf"])
    non_term_weight <- mean(pwf[!rownames(pwf) %in% term_features, "pwf"])
    as.numeric(x[['numDEInCat']])/as.numeric(x[['numInCat']]) / (term_weight/non_term_weight) / (len_fo

  return(obj)
}
```

Now we can filter by the estimated over-representation too

```
limma_over_rep_go %>% filter(BH<0.01)  %>%
  addAdjustedOverRep(lm_pwf, sapiens.go.full) %>%
  filter(adj_over_rep>2) %>%
  head(10)
```

```
##      category over_represented_pvalue under_represented_pvalue numDEInCat
## 1  GO:0009060            1.451592e-09                1.0000000         10
## 2  GO:0009062            3.582883e-07                1.0000000          7
## 3  GO:0072329            1.504581e-06                0.9999999          7
## 4  GO:0006635            1.649774e-06                1.0000000          6
## 5  GO:0003988            2.179142e-06                1.0000000          4
## 6  GO:0019395            5.061231e-06                0.9999998          6
## 7  GO:0034440            5.061231e-06                0.9999998          6
## 8  GO:0016408            1.008649e-05                0.9999999          4
## 9  GO:1990542            2.493916e-05                0.9999991          5
## 10 GO:0016508            5.505364e-05                1.0000000          3
##    numInCat                                    term ontology           BH
## 1        22                     aerobic respiration       BP 1.227425e-06
## 2        15            fatty acid catabolic process       BP 1.146329e-04
## 3        18   monocarboxylic acid catabolic process       BP 3.710673e-04
## 4        12              fatty acid beta-oxidation       BP 3.906004e-04
## 5         4   acetyl-CoA C-acyltransferase activity       MF 4.867298e-04
## 6        14                    fatty acid oxidation       BP 9.822107e-04
## 7        14                         lipid oxidation       BP 9.822107e-04
## 8         5             C-acyltransferase activity       MF 1.705769e-03
## 9        11   mitochondrial transmembrane transport       BP 3.600363e-03
## 10        3 long-chain-enoyl-CoA hydratase activity       MF 7.083967e-03
##    adj_over_rep
## 1      2.334663
## 2      2.497658
## 3      2.157995
## 4      2.626936
## 5      4.491990
## 6      2.236750
## 7      2.236750
## 8      3.696617
## 9      2.527376
## 10     4.933226
```

```
lm_over_rep_go %>% filter(BH<0.01)  %>%
  addAdjustedOverRep(lm_pwf, sapiens.go.full) %>%
```

```
  filter(adj_over_rep>2) %>%
  head(10)
```

```
##      category over_represented_pvalue under_represented_pvalue numDEInCat
## 1  GO:0044281            2.813530e-29                        1        135
## 2  GO:0043436            3.024232e-25                        1        100
## 3  GO:0006082            5.360784e-25                        1        100
## 4  GO:0019752            9.529217e-25                        1         99
## 5  GO:0055114            5.730568e-21                        1         54
## 6  GO:0006520            2.267964e-16                        1         55
## 7  GO:0044429            3.302703e-13                        1         67
## 8  GO:0017144            2.590984e-12                        1         57
## 9  GO:0004812            4.015289e-12                        1         21
## 10 GO:0016875            4.015289e-12                        1         21
##    numInCat                                          term ontology
## 1       310             small molecule metabolic process       BP
## 2       204                    oxoacid metabolic process       BP
## 3       205                  organic acid metabolic process       BP
## 4       203               carboxylic acid metabolic process       BP
## 5        83                    oxidation-reduction process       BP
## 6        97           cellular amino acid metabolic process       BP
## 7       159                              mitochondrial part       CC
## 8       128                          drug metabolic process       BP
## 9        24                  aminoacyl-tRNA ligase activity       MF
## 10       24 ligase activity, forming carbon-oxygen bonds       MF
##                BH adj_over_rep
## 1  3.330656e-25     2.304025
## 2  1.790043e-21     2.546123
## 3  2.115365e-21     2.530950
## 4  2.820172e-21     2.533344
## 5  1.130641e-17     3.380498
## 6  2.684815e-13     2.783865
## 7  2.606493e-10     2.268486
## 8  1.704004e-09     2.366569
## 9  2.359568e-09     4.103975
## 10 2.359568e-09     4.103975
```

Note that many of the above GO terms are essentially redundant and are simply the same set of proteins with each of the GO terms up the heirachy being over-represented. We can perform a simplification by taking each over-represented GO term, considering all its offspring and removing it if any of the offspring terms are more significantly over-represented.

Below, I've wrapped this all up into a function so that we can pass a data.frame of significantly over-represented GO terms and it will remove the redundant terms.

```
remove_redundant_GO_terms <- function(go_df){
  all_observed_go <- unique(go_df$category) # identify all the GO terms
  all_observed_go <- all_observed_go[!is.na(all_observed_go)] # Remove any NAs

  # Get the ontologies for the GO terms
  ontologies <- AnnotationDbi::select(GO.db, all_observed_go, columns = c('ONTOLOGY'), keytype='GOID')
  ontologies <- setNames(ontologies$ONTOLOGY, ontologies$GOID)
```

```
  # Get the mappings from GO term to parent GO terms using functions in GO.R
  go2Offspring <- getAllMappings(all_observed_go, ontologies, verbose=FALSE, direction="offspring")
  go2Ancesters <- getAllMappings(all_observed_go, ontologies, verbose=FALSE, direction="ancester")

  # start with all observed GO terms being retained
  retained <- all_observed_go

  # We want to keep track of the GO IDs we have processed
  processed <- NULL

  # If any GO term has no detected offspring or ancesters, mark them as already processed
  # This will also mean they are always retained
  no_anc_off <- setdiff(all_observed_go, union(names(go2Ancesters), names(go2Offspring)))
  if(length(no_anc_off)>0){
    cat(sprintf("No offspring or ancesters could be found for these terms: %s", no_anc_off))
    processed <- no_anc_off
  }

  # When all observed go terms are in processed, stop while loop
  while(length(setdiff(all_observed_go, processed))!=0){
    go_id <- setdiff(all_observed_go, processed)[1]

    # go_tree = the go term plus all ancesters and offspring also observed
    go_tree <- union(go2Ancesters[[go_id]], go2Offspring[[go_id]]) %>%
      intersect(all_observed_go) %>% c(go_id)

    top_go <- go_df %>%
      filter(category %in% go_tree) %>% # subset to the terms in go_tree
      arrange(over_represented_pvalue) %>% # order by p-value (ascending by default)
      head(1) %>% # keep the top row
      pull(category) # pull out the category

    # We want to remove all the terms in the tree except top_go
    terms_to_remove <- setdiff(go_tree, top_go)

    processed <- union(processed, go_tree) # all terms in the tree are now considered "processed"

    retained <- setdiff(retained, terms_to_remove) # remove the unwanted terms from retained
  }

  go_df <- go_df %>% filter(category %in% retained) # subset to the retained terms

  return(go_df)
}
```

Task: Modify the `remove_redundant_GO_terms` function to keep an arbitrary number of top GO terms within each sub tree of GO terms

```
limma_over_rep_go %>% filter(BH<0.01)  %>%
  addAdjustedOverRep(limma_pwf, sapiens.go.full) %>%
  filter(adj_over_rep>3) %>%
  remove_redundant_GO_terms() %>%
```

```
  head(10)
```

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

```
##      category over_represented_pvalue under_represented_pvalue numDEInCat
## 1  GO:0044281            6.262462e-17                1.0000000         42
## 2  GO:0055114            1.732575e-15                1.0000000         23
## 3  GO:0044429            1.226345e-13                1.0000000         28
## 4  GO:0006091            1.889344e-12                1.0000000         21
## 5  GO:0017144            4.091103e-12                1.0000000         24
## 6  GO:1901135            1.432374e-09                1.0000000         23
## 7  GO:0019637            6.714199e-09                1.0000000         23
## 8  GO:0016491            2.011993e-07                1.0000000         16
## 9  GO:0016746            3.232270e-07                1.0000000          9
## 10 GO:0006732            5.270707e-07                0.9999999         15
##    numInCat                                          term ontology
## 1       310              small molecule metabolic process       BP
## 2        83                   oxidation-reduction process       BP
## 3       159                             mitochondrial part       CC
## 4        92 generation of precursor metabolites and energy       BP
## 5       128                         drug metabolic process       BP
## 6       154      carbohydrate derivative metabolic process       BP
## 7       166              organophosphate metabolic process       BP
## 8        95                        oxidoreductase activity       MF
## 9        28 transferase activity, transferring acyl groups       MF
## 10       90                     coenzyme metabolic process       BP
##              BH adj_over_rep
## 1  7.413503e-13     3.628931
## 2  1.025511e-11     7.315526
## 3  4.839157e-10     4.757734
## 4  3.195151e-09     6.128840
## 5  5.381164e-09     4.980478
## 6  1.227425e-06     4.033058
## 7  4.675452e-06     3.740975
## 8  7.005286e-05     4.425321
## 9  1.093246e-04     8.693051
## 10 1.641964e-04     4.427250
```

```
lm_over_rep_go %>% filter(BH<0.01)  %>%
  addAdjustedOverRep(lm_pwf, sapiens.go.full) %>%
  filter(adj_over_rep>3) %>%
  remove_redundant_GO_terms() %>%
  head(10)
```

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

```
##      category over_represented_pvalue under_represented_pvalue numDEInCat
## 1  GO:0055114            5.730568e-21                1.0000000         54
## 2  GO:0004812            4.015289e-12                1.0000000         21
## 3  GO:0006418            1.430149e-10                1.0000000         21
## 4  GO:0006007            4.564981e-07                1.0000000         10
## 5  GO:0009062            5.816656e-06                0.9999996         11
```

```
## 6  GO:0005832            1.260381e-05                1.0000000          8
## 7  GO:0019319            1.377513e-05                0.9999984         13
## 8  GO:1901998            1.528521e-05                0.9999996          9
## 9  GO:0016903            1.708435e-05                0.9999989         10
## 10 GO:0022624            5.127074e-05                0.9999934         12
##    numInCat
## 1        83
## 2        24
## 3        26
## 4        10
## 5        15
## 6         8
## 7        20
## 8        10
## 9        13
## 10       19
##                                                                      term
## 1                                              oxidation-reduction process
## 2                                           aminoacyl-tRNA ligase activity
## 3                               tRNA aminoacylation for protein translation
## 4                                             glucose catabolic process
## 5                                            fatty acid catabolic process
## 6                                          chaperonin-containing T-complex
## 7                                              hexose biosynthetic process
## 8                                                        toxin transport
## 9  oxidoreductase activity, acting on the aldehyde or oxo group of donors
## 10                                           proteasome accessory complex
##    ontology           BH adj_over_rep
## 1        BP 1.130641e-17     3.380498
## 2        MF 2.359568e-09     4.103975
## 3        BP 4.232526e-08     3.683657
## 4        BP 6.004471e-05     4.154998
## 5        BP 5.738131e-04     3.924892
## 6        CC 1.138961e-03     3.934342
## 7        BP 1.216941e-03     3.222809
## 8        BP 1.311942e-03     3.753739
## 9        MF 1.424257e-03     3.761397
## 10       CC 3.865879e-03     3.096415
```

OK so now we're getting a more useful set of terms. The next thing we might want to do is some basic plot to show the result. Again, I've wrapped this up into a a function below so we can pass the GO dataframe straight through the above pipe and plot the results

```
plotTerms <- function(go_df,
                      horizontal=FALSE, # make plot hoizontal
                      plot_top=10,# plot the top n most significant GO terms
                      shorten_term=FALSE){ # shorten the term to max 30 char

  # re-order data frame by p-value
  if(horizontal){
    go_df <- go_df %>% arrange(ontology, over_represented_pvalue)
    go_df <- go_df %>% head(plot_top) # subset to top n most significant terms
  }
  else{
```

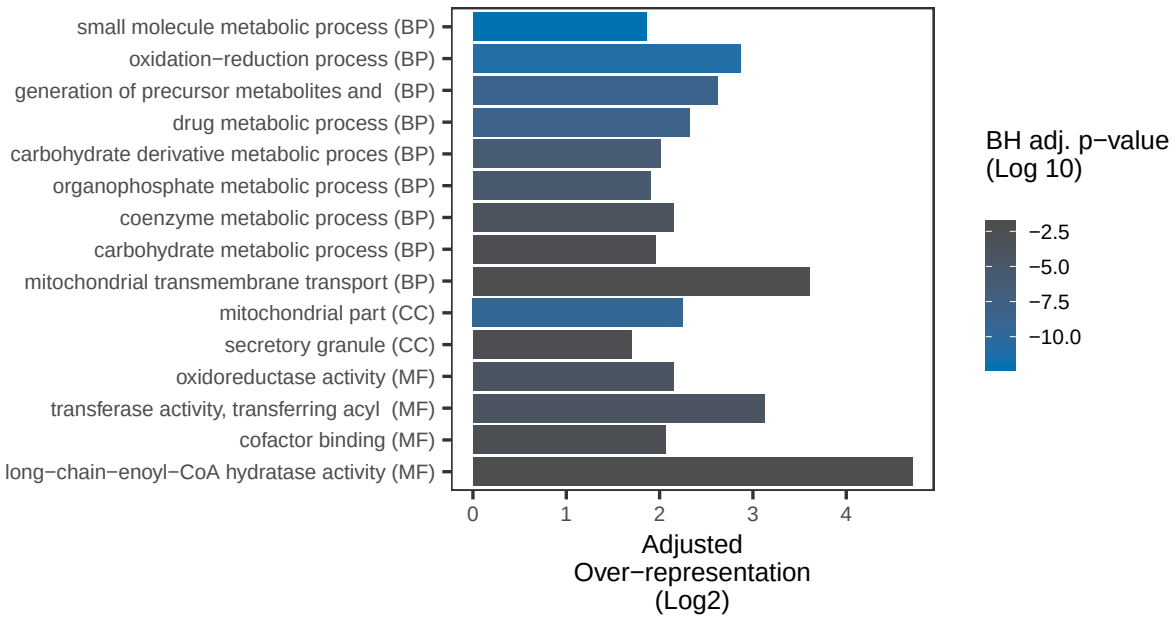
```
    go_df <- go_df %>% arrange(desc(ontology), desc(over_represented_pvalue))
    go_df <- go_df %>% tail(plot_top) # subset to top n most significant terms
  }

  print(go_df)

  if(shorten_term){
    go_df$term_for_plot <- substr(go_df$term, 1, 40) # cut at character 40
  }
  else{
    go_df$term_for_plot <- go_df$term
  }

  # add the ontology (BP, MF, CC) to the end of the term
  go_df$term_for_plot <- paste0(go_df$term_for_plot, " (", go_df$ontology, ")")

  # re-level factor make keep plotting order in order of dataframe (ontology, p-value)
  go_df$term_for_plot <- factor(go_df$term_for_plot, levels=rev(go_df$term_for_plot))

  p <- go_df %>%
    ggplot(aes(x=term_for_plot, y=log(adj_over_rep,2), fill=log(BH,10))) +
    geom_bar(stat="identity") + # When geom_bar is plotting a single data point, need to set stat="iden
    xlab("") +
    ylab("Adjusted\nOver-representation\n(Log2)") +
    scale_fill_continuous(name="BH adj. p-value\n(Log 10)\n", low=cbPalette[5], high="grey30", limits=c
    theme(text=element_text(size=15),
          plot.title=element_text(hjust=0.5))

  if(horizontal){
    p <- p + coord_flip() # Flip the coordinates
  }
  else{
    # If vertical bards, set the x-axis text at an angle so it fits better
    p <- p + theme(axis.text.x=element_text(size=12, angle=30, vjust=1, hjust=1))
  }

  return(list("p"=p, "data"=go_df))
}
```

```
limma_over_rep_go %>% filter(BH<0.01)  %>%
  addAdjustedOverRep(limma_pwf, sapiens.go.full) %>%
  filter(adj_over_rep>3) %>%
  remove_redundant_GO_terms() %>%
  plotTerms(horizontal=TRUE, shorten_term=TRUE, plot_top=20)
```

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

```
##      category over_represented_pvalue under_represented_pvalue numDEInCat
## 1  GO:0044281           6.262462e-17                1.0000000         42
## 2  GO:0055114           1.732575e-15                1.0000000         23
## 3  GO:0006091           1.889344e-12                1.0000000         21
## 4  GO:0017144           4.091103e-12                1.0000000         24
## 5  GO:1901135           1.432374e-09                1.0000000         23
```

```
## 6  GO:0019637            6.714199e-09                1.0000000         23
## 7  GO:0006732            5.270707e-07                0.9999999         15
## 8  GO:0005975            1.559771e-05                0.9999972         13
## 9  GO:1990542            2.493916e-05                0.9999991          5
## 10 GO:0044429            1.226345e-13                1.0000000         28
## 11 GO:0030141            2.720569e-05                0.9999942         15
## 12 GO:0016491            2.011993e-07                1.0000000         16
## 13 GO:0016746            3.232270e-07                1.0000000          9
## 14 GO:0048037            7.163320e-06                0.9999988         13
## 15 GO:0016508            5.505364e-05                1.0000000          3
##    numInCat                                          term ontology
## 1       310              small molecule metabolic process       BP
## 2        83                   oxidation-reduction process       BP
## 3        92 generation of precursor metabolites and energy       BP
## 4       128                        drug metabolic process       BP
## 5       154     carbohydrate derivative metabolic process       BP
## 6       166             organophosphate metabolic process       BP
## 7        90                   coenzyme metabolic process       BP
## 8        89                carbohydrate metabolic process       BP
## 9        11         mitochondrial transmembrane transport       BP
## 10      159                            mitochondrial part       CC
## 11      117                             secretory granule       CC
## 12       95                       oxidoreductase activity       MF
## 13       28 transferase activity, transferring acyl groups       MF
## 14       85                                cofactor binding       MF
## 15        3       long-chain-enoyl-CoA hydratase activity       MF
##              BH adj_over_rep
## 1  7.413503e-13     3.628931
## 2  1.025511e-11     7.315526
## 3  3.195151e-09     6.128840
## 4  5.381164e-09     4.980478
## 5  1.227425e-06     4.033058
## 6  4.675452e-06     3.740975
## 7  1.641964e-04     4.427250
## 8  2.397996e-03     3.881342
## 9  3.600363e-03    12.207464
## 10 4.839157e-10     4.757734
## 11 3.836353e-03     3.249868
## 12 7.005286e-05     4.425321
## 13 1.093246e-04     8.693051
## 14 1.284839e-03     4.166883
## 15 7.083967e-03    26.089154

## $p
```




```
##
## $data
##      category over_represented_pvalue under_represented_pvalue numDEInCat
## 1  GO:0044281            6.262462e-17                1.0000000         42
## 2  GO:0055114            1.732575e-15                1.0000000         23
## 3  GO:0006091            1.889344e-12                1.0000000         21
## 4  GO:0017144            4.091103e-12                1.0000000         24
## 5  GO:1901135            1.432374e-09                1.0000000         23
## 6  GO:0019637            6.714199e-09                1.0000000         23
## 7  GO:0006732            5.270707e-07                0.9999999         15
## 8  GO:0005975            1.559771e-05                0.9999972         13
## 9  GO:1990542            2.493916e-05                0.9999991          5
## 10 GO:0044429            1.226345e-13                1.0000000         28
## 11 GO:0030141            2.720569e-05                0.9999942         15
## 12 GO:0016491            2.011993e-07                1.0000000         16
## 13 GO:0016746            3.232270e-07                1.0000000          9
## 14 GO:0048037            7.163320e-06                0.9999988         13
## 15 GO:0016508            5.505364e-05                1.0000000          3
##    numInCat                                          term ontology
## 1       310                 small molecule metabolic process       BP
## 2        83                      oxidation-reduction process       BP
## 3        92 generation of precursor metabolites and energy       BP
## 4       128                          drug metabolic process       BP
## 5       154       carbohydrate derivative metabolic process       BP
## 6       166                organophosphate metabolic process       BP
```

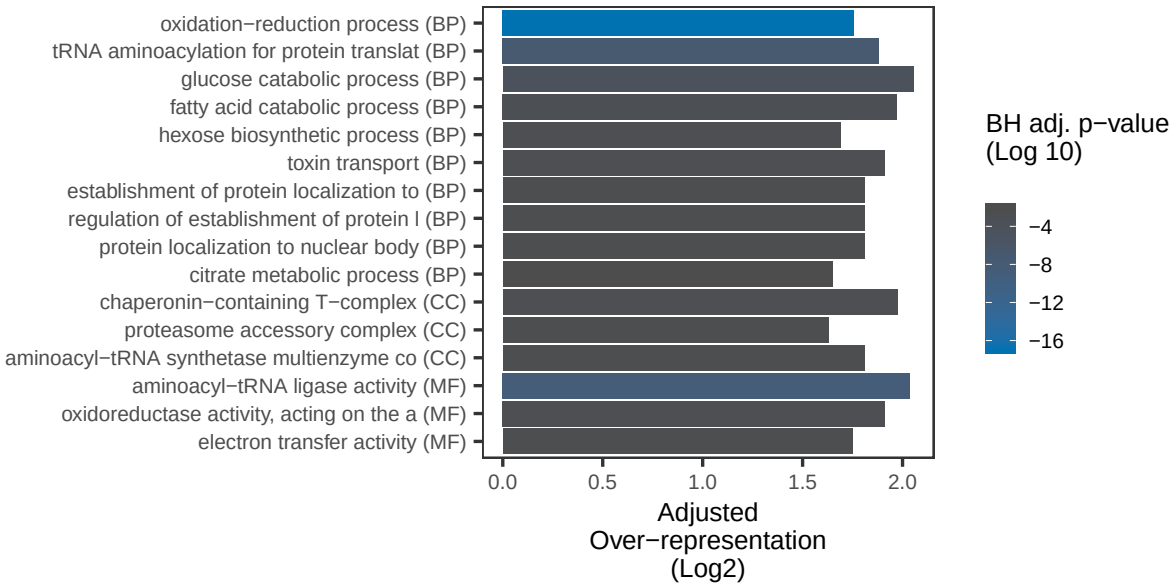
```
## 7        90                     coenzyme metabolic process       BP
## 8        89                 carbohydrate metabolic process       BP
## 9        11           mitochondrial transmembrane transport       BP
## 10      159                              mitochondrial part       CC
## 11      117                               secretory granule       CC
## 12       95                          oxidoreductase activity       MF
## 13       28 transferase activity, transferring acyl groups       MF
## 14       85                                cofactor binding       MF
## 15        3        long-chain-enoyl-CoA hydratase activity       MF
##              BH adj_over_rep                                    term_for_plot
## 1  7.413503e-13     3.628931         small molecule metabolic process (BP)
## 2  1.025511e-11     7.315526              oxidation-reduction process (BP)
## 3  3.195151e-09     6.128840 generation of precursor metabolites and  (BP)
## 4  5.381164e-09     4.980478                   drug metabolic process (BP)
## 5  1.227425e-06     4.033058 carbohydrate derivative metabolic proces (BP)
## 6  4.675452e-06     3.740975        organophosphate metabolic process (BP)
## 7  1.641964e-04     4.427250              coenzyme metabolic process (BP)
## 8  2.397996e-03     3.881342           carbohydrate metabolic process (BP)
## 9  3.600363e-03    12.207464    mitochondrial transmembrane transport (BP)
## 10 4.839157e-10     4.757734                       mitochondrial part (CC)
## 11 3.836353e-03     3.249868                        secretory granule (CC)
## 12 7.005286e-05     4.425321                  oxidoreductase activity (MF)
## 13 1.093246e-04     8.693051 transferase activity, transferring acyl  (MF)
## 14 1.284839e-03     4.166883                          cofactor binding (MF)
## 15 7.083967e-03    26.089154  long-chain-enoyl-CoA hydratase activity (MF)
```

```
lm_over_rep_go %>% filter(BH<0.01)  %>%
  addAdjustedOverRep(lm_pwf, sapiens.go.full) %>%
  filter(adj_over_rep>3) %>%
  remove_redundant_GO_terms() %>%
  plotTerms(horizontal=TRUE, shorten_term=TRUE, plot_top=20)
```

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

```
##      category over_represented_pvalue under_represented_pvalue numDEInCat
## 1  GO:0055114            5.730568e-21                1.0000000         54
## 2  GO:0006418            1.430149e-10                1.0000000         21
## 3  GO:0006007            4.564981e-07                1.0000000         10
## 4  GO:0009062            5.816656e-06                0.9999996         11
## 5  GO:0019319            1.377513e-05                0.9999984         13
## 6  GO:1901998            1.528521e-05                0.9999996          9
## 7  GO:0070200            8.721730e-05                0.9999970          8
## 8  GO:0070202            8.721730e-05                0.9999970          8
## 9  GO:1903405            8.721730e-05                0.9999970          8
## 10 GO:0006101            1.111138e-04                0.9999852         11
## 11 GO:0005832            1.260381e-05                1.0000000          8
## 12 GO:0022624            5.127074e-05                0.9999934         12
## 13 GO:0017101            5.488376e-05                0.9999969          9
## 14 GO:0004812            4.015289e-12                1.0000000         21
## 15 GO:0016903            1.708435e-05                0.9999989         10
## 16 GO:0009055            6.210380e-05                0.9999923         11
##    numInCat
## 1        83
```

```
## 2        26
## 3        10
## 4        15
## 5        20
## 6        10
## 7         9
## 8         9
## 9         9
## 10       18
## 11        8
## 12       19
## 13       11
## 14       24
## 15       13
## 16       18
##                                                                    term
## 1                                          oxidation-reduction process
## 2                            tRNA aminoacylation for protein translation
## 3                                            glucose catabolic process
## 4                                          fatty acid catabolic process
## 5                                           hexose biosynthetic process
## 6                                                     toxin transport
## 7                       establishment of protein localization to telomere
## 8      regulation of establishment of protein localization to chromosome
## 9                                    protein localization to nuclear body
## 10                                           citrate metabolic process
## 11                                 chaperonin-containing T-complex
## 12                                   proteasome accessory complex
## 13                         aminoacyl-tRNA synthetase multienzyme complex
## 14                                    aminoacyl-tRNA ligase activity
## 15 oxidoreductase activity, acting on the aldehyde or oxo group of donors
## 16                                           electron transfer activity
##    ontology           BH adj_over_rep
## 1        BP 1.130641e-17     3.380498
## 2        BP 4.232526e-08     3.683657
## 3        BP 6.004471e-05     4.154998
## 4        BP 5.738131e-04     3.924892
## 5        BP 1.216941e-03     3.222809
## 6        BP 1.311942e-03     3.753739
## 7        BP 5.768036e-03     3.507372
## 8        BP 5.768036e-03     3.507372
## 9        BP 5.768036e-03     3.507372
## 10       BP 6.850860e-03     3.140298
## 11       CC 1.138961e-03     3.934342
## 12       CC 3.865879e-03     3.096415
## 13       CC 4.060712e-03     3.503343
## 14       MF 2.359568e-09     4.103975
## 15       MF 1.424257e-03     3.761397
## 16       MF 4.538178e-03     3.360560

## $p
```




```
##
## $data
##      category over_represented_pvalue under_represented_pvalue numDEInCat
## 1  GO:0055114            5.730568e-21                1.0000000         54
## 2  GO:0006418            1.430149e-10                1.0000000         21
## 3  GO:0006007            4.564981e-07                1.0000000         10
## 4  GO:0009062            5.816656e-06                0.9999996         11
## 5  GO:0019319            1.377513e-05                0.9999984         13
## 6  GO:1901998            1.528521e-05                0.9999996          9
## 7  GO:0070200            8.721730e-05                0.9999970          8
## 8  GO:0070202            8.721730e-05                0.9999970          8
## 9  GO:1903405            8.721730e-05                0.9999970          8
## 10 GO:0006101            1.111138e-04                0.9999852         11
## 11 GO:0005832            1.260381e-05                1.0000000          8
## 12 GO:0022624            5.127074e-05                0.9999934         12
## 13 GO:0017101            5.488376e-05                0.9999969          9
## 14 GO:0004812            4.015289e-12                1.0000000         21
## 15 GO:0016903            1.708435e-05                0.9999989         10
## 16 GO:0009055            6.210380e-05                0.9999923         11
##    numInCat
## 1        83
## 2        26
## 3        10
## 4        15
## 5        20
```

```
## 6        10
## 7         9
## 8         9
## 9         9
## 10       18
## 11        8
## 12       19
## 13       11
## 14       24
## 15       13
## 16       18
##                                                                      term
## 1                                          oxidation-reduction process
## 2                              tRNA aminoacylation for protein translation
## 3                                            glucose catabolic process
## 4                                         fatty acid catabolic process
## 5                                          hexose biosynthetic process
## 6                                                      toxin transport
## 7                          establishment of protein localization to telomere
## 8       regulation of establishment of protein localization to chromosome
## 9                                      protein localization to nuclear body
## 10                                           citrate metabolic process
## 11                                     chaperonin-containing T-complex
## 12                                        proteasome accessory complex
## 13                           aminoacyl-tRNA synthetase multienzyme complex
## 14                                       aminoacyl-tRNA ligase activity
## 15 oxidoreductase activity, acting on the aldehyde or oxo group of donors
## 16                                         electron transfer activity
##    ontology           BH adj_over_rep
## 1        BP 1.130641e-17     3.380498
## 2        BP 4.232526e-08     3.683657
## 3        BP 6.004471e-05     4.154998
## 4        BP 5.738131e-04     3.924892
## 5        BP 1.216941e-03     3.222809
## 6        BP 1.311942e-03     3.753739
## 7        BP 5.768036e-03     3.507372
## 8        BP 5.768036e-03     3.507372
## 9        BP 5.768036e-03     3.507372
## 10       BP 6.850860e-03     3.140298
## 11       CC 1.138961e-03     3.934342
## 12       CC 3.865879e-03     3.096415
## 13       CC 4.060712e-03     3.503343
## 14       MF 2.359568e-09     4.103975
## 15       MF 1.424257e-03     3.761397
## 16       MF 4.538178e-03     3.360560
##                                    term_for_plot
## 1              oxidation-reduction process (BP)
## 2  tRNA aminoacylation for protein translat (BP)
## 3                glucose catabolic process (BP)
## 4             fatty acid catabolic process (BP)
## 5              hexose biosynthetic process (BP)
## 6                          toxin transport (BP)
## 7  establishment of protein localization to (BP)
## 8  regulation of establishment of protein l (BP)
```

```
## 9      protein localization to nuclear body (BP)
## 10                citrate metabolic process (BP)
## 11          chaperonin-containing T-complex (CC)
## 12             proteasome accessory complex (CC)
## 13 aminoacyl-tRNA synthetase multienzyme co (CC)
## 14           aminoacyl-tRNA ligase activity (MF)
## 15 oxidoreductase activity, acting on the a (MF)
## 16               electron transfer activity (MF)
```

Ok, so now we have a reasonable list of over-represented GO terms and some pretty plots to show off our results...

Task: Modify the plotTerms function so that it makes a separate plot for each ontology

One thing you may have noticed in the above is that the over-represented terms are slightly different for `lm` and `limma` above. For example, "aminoacyl-tRNA ligase activity" is the most significantly overrepresented MF GO term in the `lm` proteins but not present in the `limma` over-rep. GO terms.

Below, we take the GO terms over-rep with `lm` (1% FDR, >4-fold over-rep.) and inspect them with our `limma` GO over-rep analysis. Note that very few of the proteins annotated with "aminoacyl-tRNA ligase activity" are detected as having a significant change in RNA binding according to `limma` - 3/24 vs 21/24 for `lm`!!

```
lm_over_rep_go_sig <- lm_over_rep_go %>% filter(BH<0.01)  %>%
  addAdjustedOverRep(lm_pwf, sapiens.go.full) %>%
  filter(adj_over_rep>4) %>%
  remove_redundant_GO_terms() %>%
  pull(category)

limma_over_rep_go %>% filter(category %in% lm_over_rep_go_sig) %>%
  addAdjustedOverRep(lm_pwf, sapiens.go.full)  %>%
  head(10)
```

```
##     category over_represented_pvalue under_represented_pvalue numDEInCat
## 1 GO:0006007            0.0006827133                0.9999635          4
## 2 GO:0006735            0.0006827133                0.9999635          4
## 3 GO:0004812            0.0622546853                0.9879801          3
## 4 GO:0002161            1.0000000000                0.7888786          0
##   numInCat                             term ontology        BH adj_over_rep
## 1       10       glucose catabolic process       BP 0.0486865    1.6619992
## 2       10               NADH regeneration       BP 0.0486865    1.6619992
## 3       24  aminoacyl-tRNA ligase activity       MF 1.0000000    0.5862821
## 4        6 aminoacyl-tRNA editing activity       MF 1.0000000    0.0000000
```

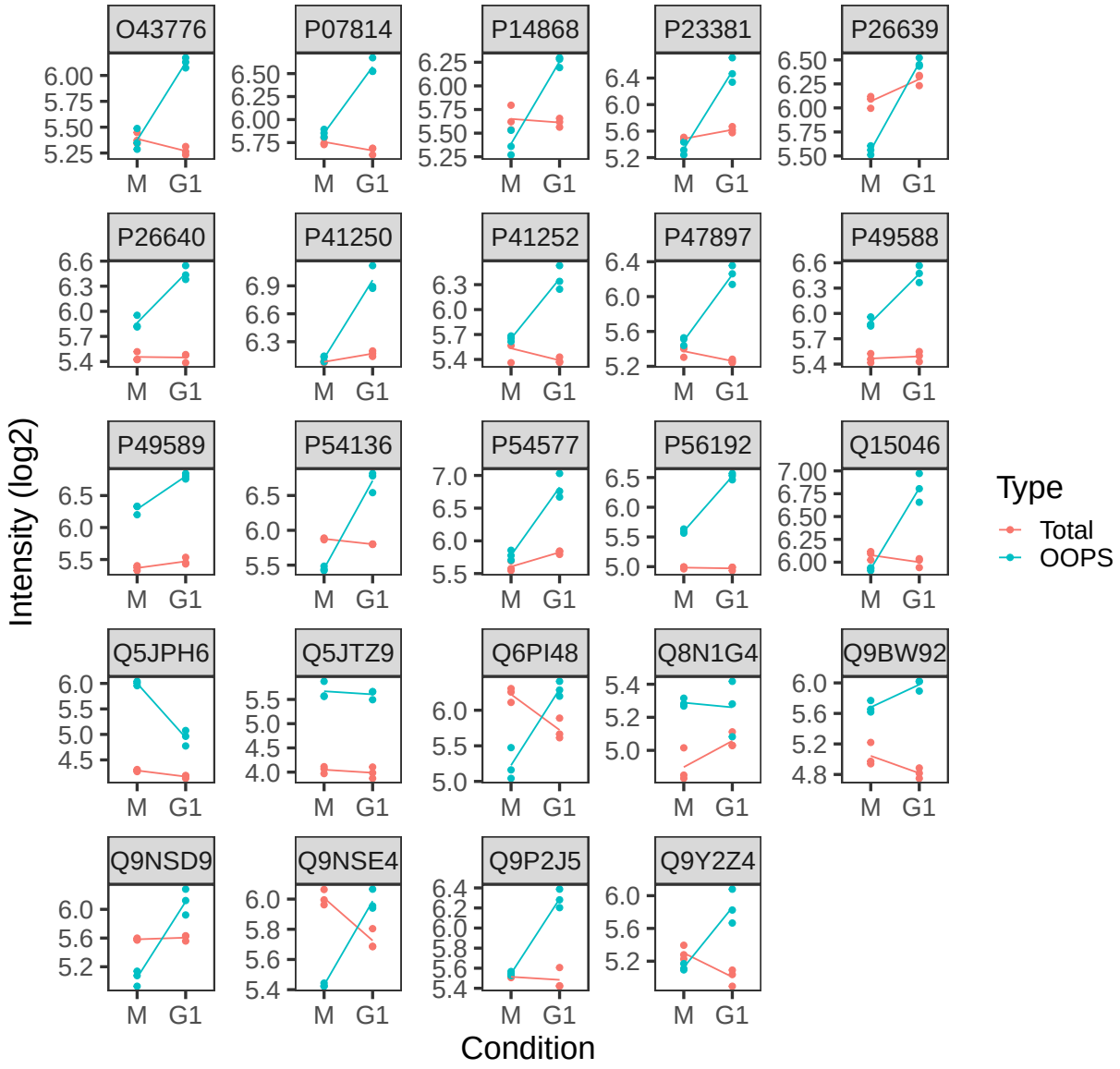
We want to take a look at the intensity values, so let's make that `plotIntensities()` function again

```
combined_intensities <- readRDS("../results/combined_intensities.rds")

plotIntensities <- function(obj){

  p <- tidy(obj, addPheno=TRUE) %>%
    ggplot(aes(Condition, value, colour=Type, group=Type)) +
    geom_point() +
```

```
    stat_summary(geom="line", fun.y=mean) +
    facet_wrap(~gene, scales='free') +
    ylab("Intensity (log2)")

  invisible(p)
}
```

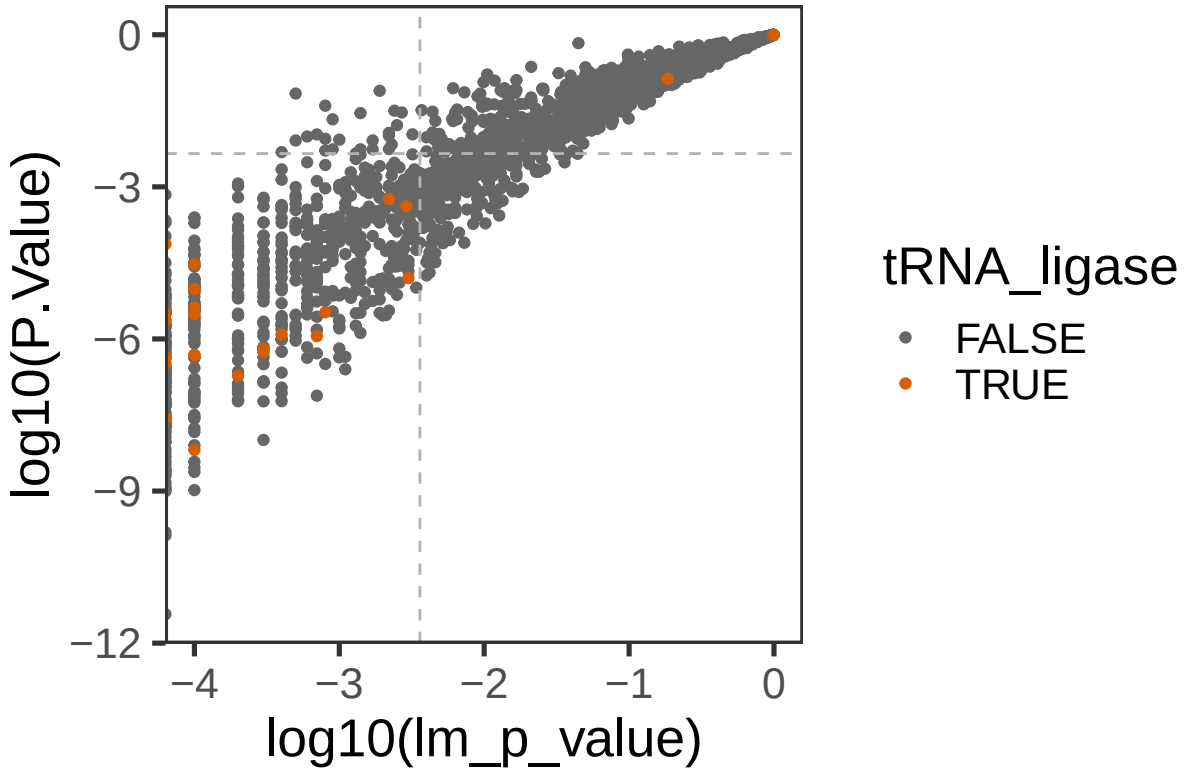
If we take a look at the intensity values for these 24 proteins, most do seem to have a clear increase in RNA binding (the two that don't are both mitochondrial). So, why does `limma` only detect 3/24 of these as increasing RNA binding...

```
tRNA_ligases <- sapiens.go.full %>% filter(GO.ID=="GO:0004812") %>% pull(UNIPROTKB)
combined_intensities[intersect(rownames(combined_intensities), tRNA_ligases),] %>% plotIntensities() %>%
```



Next, we compare the p-values again. Note that all but two tRNA-ligases (the Mt ones) are in the bottom left quadrant (e.g <1% with both `lm` and `limma`).

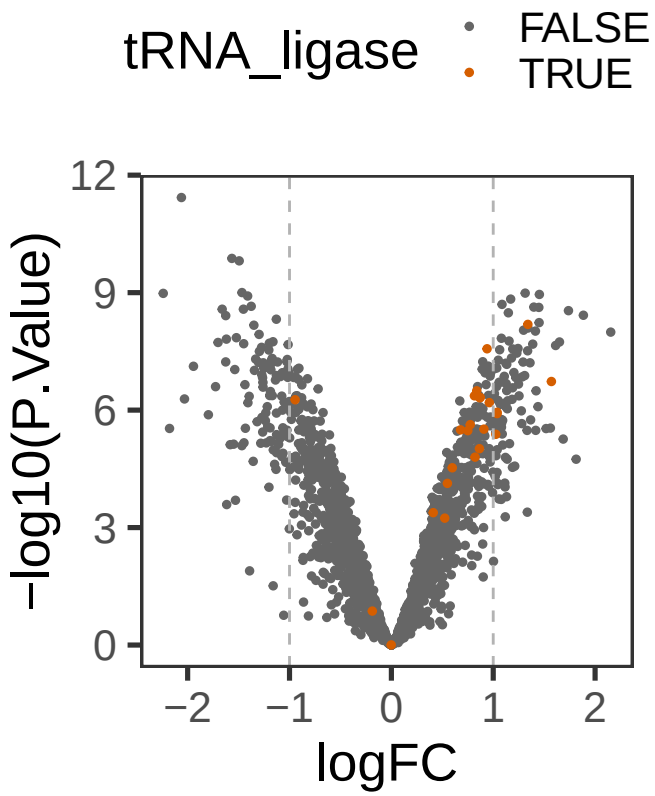
```
max_p_sig_lm <- compare_methods %>% filter(lm_BH<0.01) %>% pull(lm_p_value) %>% max()
max_p_sig_limma <- compare_methods %>% filter(adj.P.Val<0.01) %>% pull(P.Value) %>% max()

compare_methods %>%
  mutate(tRNA_ligase=Row.names %in% tRNA_ligases) %>%
  arrange(tRNA_ligase) %>%
  ggplot(aes(x=log10(lm_p_value), y=log10(P.Value), colour=tRNA_ligase)) +
  geom_point() +
  scale_colour_manual(values=c("grey40", cbPalette[6])) +
  geom_vline(xintercept=log10(max_p_sig_lm), linetype=2, colour="grey70") +
  geom_hline(yintercept=log10(max_p_sig_limma), linetype=2,colour="grey70")
```



Finally, we look at the estimated fold change. Note that most of the fold change estimates are relatively low (~1.5 - 2-fold). So these proteins aren't identified by `limma` because we've used a more conservative approach and also thresholded on the confidence interval for the estimated fold change. This demonstrates one of the downsides of applying such thresholds since these proteins were subsequently shown to have a consistent (if still slight) change in RNA binding in a Thymidine + Nocadazole experiment suggesting this is probably a real change in RNA binding. So, while a threshold on the log fold change is a sensible approach, be careful about what threshold you use!

```
compare_methods %>%
  mutate(tRNA_ligase=Row.names %in% tRNA_ligases) %>%
  arrange(tRNA_ligase) %>%
```

```
ggplot(aes(logFC, -log10(P.Value), colour=tRNA_ligase)) +
geom_point(size=1) +
scale_colour_manual(values=c("grey40", cbPalette[6])) +
geom_vline(xintercept=1, linetype=2, colour="grey70") +
geom_vline(xintercept=-1, linetype=2, colour="grey70") +
theme(legend.position="top", legend.direction=2)
```



And that's the end of the workshop!

Task: Repeat the above analysis but for proteins with a decrease in RNA binding with either 'lm' or 'limma'. How would you interpret this set of GO terms?