Checking Executables

Robert M Flight

2018-01-22 12:14:26

Contents

Purpose 1
R Version 1
Executable Version 2
Comparison 3
Missing GO terms 4
Different Genes Measured 4

```
knitr::opts_chunk$set(echo = TRUE)
root_loc <- rprojroot::find_root("DESCRIPTION")

tmp_loc <- tempdir()
Sys.setenv(file_loc = root_loc)
Sys.setenv(exec_loc = file.path(root_loc, "inst/executables"))
Sys.setenv(test_loc = file.path(root_loc, "inst/extdata/test_data"))
Sys.setenv(results_loc = tmp_loc)

test_loc <- file.path(root_loc, "inst/extdata/test_data")

Sys.chmod(dir(file.path(root_loc, "inst", "executables"), pattern = "*.R", full.names = TRUE), "0750")
library(categoryCompare2)
library(tools)
```

Purpose

Verify that the executables give the same results as running `categoryCompare` itself.

R Version

We will use our R programming to read in the data and generate the annotations.

```
get_feature_lists <- function(file_list){
  file_not_universe <- unlist(file_list[!(names(file_list) %in% "universe")])

  condition_names <- basename(file_not_universe)
  condition_names <- gsub(paste0(".", file_ext(condition_names[1])), "", condition_names)

  file_data <- lapply(file_not_universe, function(x){
    readLines(x)
  })
  names(file_data) <- condition_names

  if (is.null(file_list$universe)) {
    file_data$universe <- unique(unlist(file_data))
  } else {
```

```
    file_data$universe <- readLines(file_list$universe)
  }

  file_data
}

file_list <- list(file1 = file.path(test_loc, "10_symbol.txt"), universe = file.path(test_loc, "universe
feature_list <- get_feature_lists(file_list)
feature_universe <- feature_list$universe
feature_list$universe <- NULL
```

```
annotation_obj <- get_db_annotation("org.Hs.eg.db", feature_type = "SYMBOL", annotation_type = "CC")
```

```
##
## Attaching package: 'S4Vectors'
```

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

```
##
```

```
gene_enrichments <- lapply(feature_list, function(in_genes){
  hypergeometric_feature_enrichment(
    new("hypergeom_features", significant = in_genes,
        universe = feature_universe, annotation = annotation_obj),
    p_adjust = "BH"
  )
})

combined_enrichments <- combine_enrichments(gene_enrichments)

p_cutoff_column <- "padjust"
p_cutoff_value <- 0.01
p_cutoff_direction <- "<="

count_cutoff_column <- "counts"
count_cutoff_value <- 2
count_cutoff_direction <- ">="

count_call_info <- list(fun = count_cutoff_direction, var_1 = count_cutoff_column, var_2 = count_cutoff
p_call_info <- list(fun = p_cutoff_direction, var_1 = p_cutoff_column, var_2 = p_cutoff_value)

significant_calls <- list(counts = count_call_info, pvalues = p_call_info)

combined_significant <- combined_significant_calls(combined_enrichments, significant_calls)

results_table <- generate_table(combined_significant)
```

Executable Version

```
$exec_loc/feature_files_2_json.R --json="$results_loc/features.json" \
  --file1="$test_loc/10_symbol.txt" \
  --universe="$test_loc/universe_symbol.txt"
```

```
$exec_loc/create_annotations.R --orgdb="org.Hs.eg.db" \
  --feature-type="SYMBOL" \
  --annotation-type="CC" \
  --json="$results_loc/annotations.json"
```

```
##
## Attaching package: 'S4Vectors'
##
## The following object is masked from 'package:base':
##
##     expand.grid
```

```
$exec_loc/categoryCompare2.R --features="$results_loc/features.json" \
  --annotations="$results_loc/annotations.json" \
  --p-cutoff=0.01 \
  --count-cutoff=2 \
  --output-directory="$results_loc"
```

Comparison

```
exec_results <- read.table(file.path(tmp_loc, "full_table.txt"), sep = "\t", header = TRUE, stringsAsFa
```

```
both_results <- dplyr::full_join(results_table, exec_results)
```

```
## Joining, by = c("name", "description", "sig_data.sig", "meas_data.meas")
```

```
p_diff <- data.frame(diff = both_results$`10_symbol.p` - both_results$X10_symbol.p)
max(p_diff$diff)
```

```
## [1] 3.523657e-19
```

```
library(ggplot2)
sum(is.na(both_results$`10_symbol.p`))
```

```
## [1] 0
```

```
sum(is.na(both_results$X10_symbol.p))
```

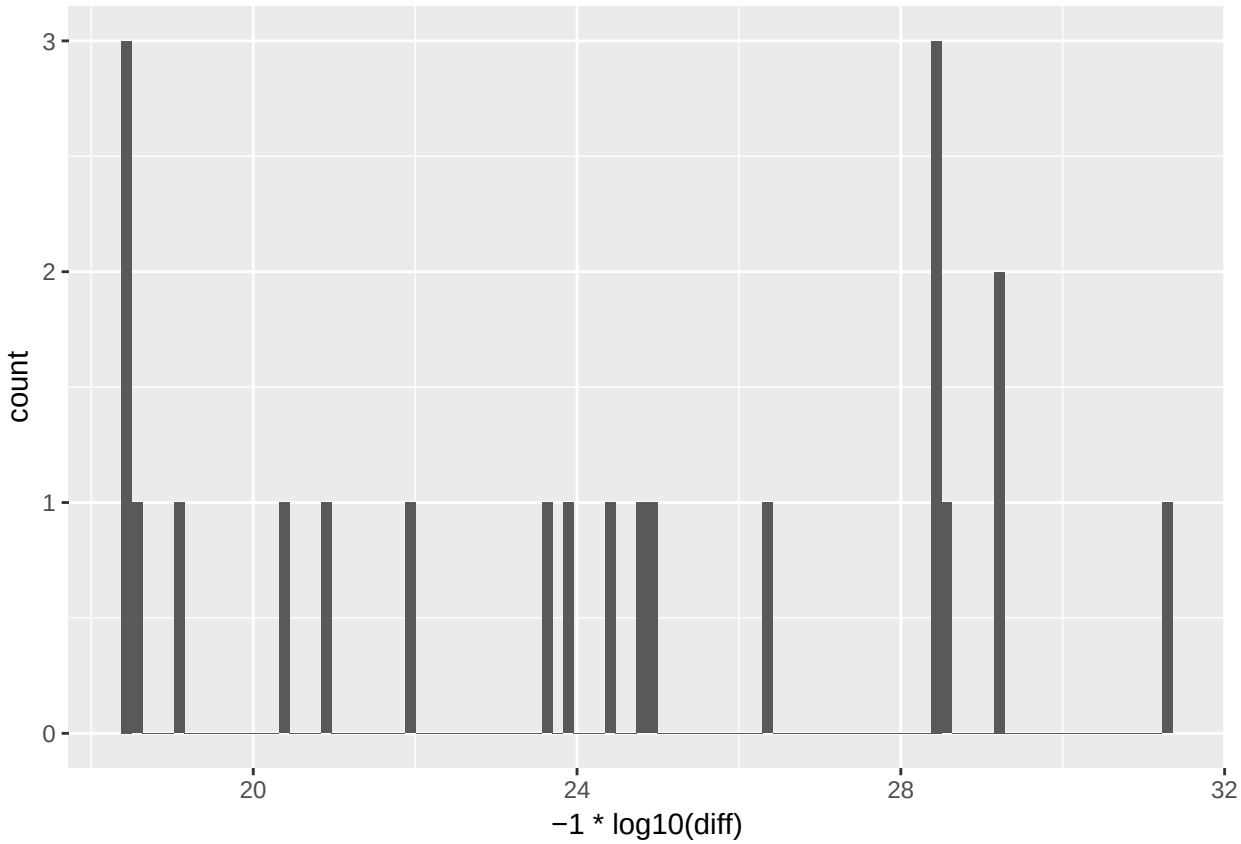
```
## [1] 0
```

```
ggplot(p_diff, aes(x = -1*log10(diff))) + geom_histogram(bins = 100)
```

```
## Warning in fun(x, ...): NaNs produced
```

```
## Warning in FUN(X[[i]], ...): NaNs produced
```

```
## Warning: Removed 27 rows containing non-finite values (stat_bin).
```



OK, so where there are both GO terms, the differences are on the order of machine precision, but there are 0 GO terms missing from the executable case. That is not good!

Missing GO terms

Lets read in the annotation object and see what GO terms are present there compared to the one we generated.

```
json_annotations <- json_2_annotation(file.path(tmp_loc, "annotations.json"))
```

```
all.equal(json_annotations, annotation_obj)
```

```
## [1] TRUE
```

Nope, supposedly have the exact same set of annotations.

Different Genes Measured

```
json_genes <- jsonlite::fromJSON(file.path(tmp_loc, "features.json"))

setdiff(json_genes$`10_symbol`, feature_list$`10_symbol`)
```

```
## character(0)
```

```
length(json_genes$`10_symbol`)
```

```
## [1] 666
```

```r
length(feature_list$`10_symbol`)
```

```
## [1] 666
```

```r
setdiff(json_genes$universe, feature_universe)
```

```
## character(0)
```

```r
length(json_genes$universe)
```

```
## [1] 8595
```

```r
length(feature_universe)
```

```
## [1] 8595
```