Pathway Analysis - Neutrophils
Input data: DEGs

Elena Eyre Sánchez, PhD

2025-01-04

Contents

1 Introducción y Objetivo

En este script realizo un pathways análisis con los DEGs obtenidos de los Venn Diagrams, y preparo el resultado en archivos gem para extraer un gráfico posteriormente en Cytoscape (https://cran.r-project.org/web/packages/gprofiler2/vignettes/gprofiler2.html).

2 Paquetes y datos

3 Datos

Cargo inicialmente los listados de Neutrófilos obtenidos en jVenn.

```
setwd("~/Desktop/ELENA_UOC/TFM")
jVenn_Neutrophiles_vs_GSE22155_02 <- read_csv("DEGs/Venn_diagrams_cell_type_treatments_vs_untreated/jVe
head(jVenn_Neutrophiles_vs_GSE22155_02) # sección de los resultados
```

```
## # A tibble: 6 x 3
##   GSE50509  GSE22155_02 `GSE50509|GSE22155_02`
##   <chr>     <chr>       <chr>
## 1 CEMIP     MME         ADH1B
## 2 CRABP1    PSG9        ADH1A
## 3 CNTNAP3B  ACSL1       CEACAM1
## 4 ASB16     SLC29A2     AKR1C3
## 5 LINC01091 HK2         CIDEC
## 6 CCL11     CACHD1      PLIN1
```

```
jVenn_Neutrophiles_vs_GSE22155_47 <- read_csv("DEGs/Venn_diagrams_cell_type_treatments_vs_untreated/jVe
jVenn_Neutrophiles_vs_GSE91061 <- read_csv("DEGs/Venn_diagrams_cell_type_treatments_vs_untreated/jVenn_
jVenn_Neutrophiles_vs_GSE35640 <- read_csv("DEGs/Venn_diagrams_cell_type_treatments_vs_untreated/jVenn_
jVenn_Neutrophiles_vs_GSE61992 <- read_csv("DEGs/Venn_diagrams_cell_type_treatments_vs_untreated/jVenn_
setwd("~/Desktop/ELENA_UOC/TFM/GOBPs") # Directorio de trabajo
jVenn_Neutrophiles <- read_delim("~/Desktop/ELENA_UOC/TFM/DEGs/Venn_diagrams_cell_type_treatments_compa
    delim = ",", escape_double = FALSE, trim_ws = TRUE)
```

4 Uncovered

En esta sección contemplo la posibilidad que los DEGs únicos en el dataset sin tratar se traten de mecanismos que se escapan al tratamiento pero que igualmente se ven afectados por la enfermedad.

Primero realizo el pathway multianálisis:

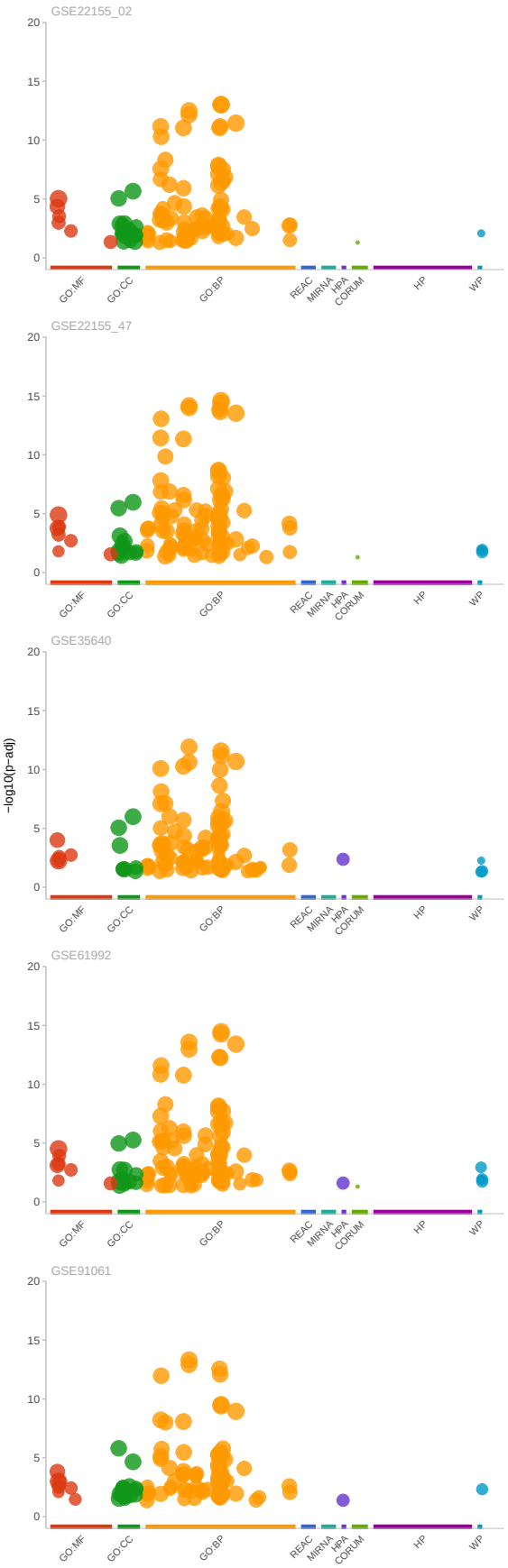
En esta sección contemplo la posibilidad que los DEGs únicos en el dataset sin tratar se traten de mecanismos que se escapan al tratamiento pero que igualmente se ven afectados por la enfermedad.

Primero realizo el pathway multianálisis:

```
multi_gostres <- gost(query = list("GSE22155_02" = unique(jVenn_Neutrophiles_vs_GSE22155_02$GSE50509[wh
                                   "GSE22155_47" = jVenn_Neutrophiles_vs_GSE22155_47$GSE50509[which(is.na(jVe
                                   "GSE35640" = jVenn_Neutrophiles_vs_GSE35640$GSE50509[which(is.na(jVenn_Neu
                                   "GSE91061" = jVenn_Neutrophiles_vs_GSE91061$GSE50509[which(is.na(jVenn_Neu
                                   "GSE61992" = jVenn_Neutrophiles_vs_GSE61992$GSE50509[which(is.na(jVenn_Neu
                                   ),
                      evcodes = TRUE, multi_query = FALSE,
                      sources = c("GO", "REAC", "MIRNA", "CORUM", "HP", "HPA", "WP"))
head(multi_gostres$result[,1:7])
```

```
##         query significant      p_value term_size query_size intersection_size
## 1 GSE22155_02        TRUE 4.999187e-02         2         58                2
## 2 GSE22155_02        TRUE 9.493110e-14     12287        350              278
## 3 GSE22155_02        TRUE 9.885614e-14     11738        350              270
## 4 GSE22155_02        TRUE 3.118515e-13      7669        350              203
## 5 GSE22155_02        TRUE 7.244126e-13      6453        350              180
## 6 GSE22155_02        TRUE 3.584711e-12     12680        350              280
##    precision
## 1 0.03448276
## 2 0.79428571
## 3 0.77142857
## 4 0.58000000
## 5 0.51428571
## 6 0.80000000
```

```
p <- gostplot(multi_gostres, capped = FALSE, interactive = F)
p
```


GSE22155_02
GSE22155_47
GSE35640
GSE61992
GSE91061
−log10(p-adj)
20
15
10
5
0
GO:MF
GO:CC
GO:BP
REAC
MIRNA
HPA
CORUM
HP
WP

Estos resultados son los que utilizo para generar los archivos gem que archivo para usar más tarde en Cytoscape.

```
# format to GEM
gem <- multi_gostres$result[,c("query", "term_id", "term_name", "p_value", "intersection_size")]
colnames(gem) <- c("query", "GO.ID", "Description", "p.Val", "Genes")
gem$FDR <- gem$p.Val
gem$Phenotype = "+1"

gem %>% group_by(query) %>%
  group_walk(~
    write.table(data.frame(.x[,c("GO.ID", "Description", "p.Val", "FDR", "Phenotype", "Genes")]),
                file = paste0("./Venn_diagrams_cell_type_treatments_vs_untreated/gProfiler_Neutrophils_
                sep = "\t", quote = F, row.names = F))
```

Proporciones de los pathways para cada estudio:

```
df <- as.data.frame(multi_gostres$result[,1:13])
table(df$query)
```

```
##
## GSE22155_02 GSE22155_47    GSE35640    GSE61992    GSE91061
##         111         124         112         123          98
```

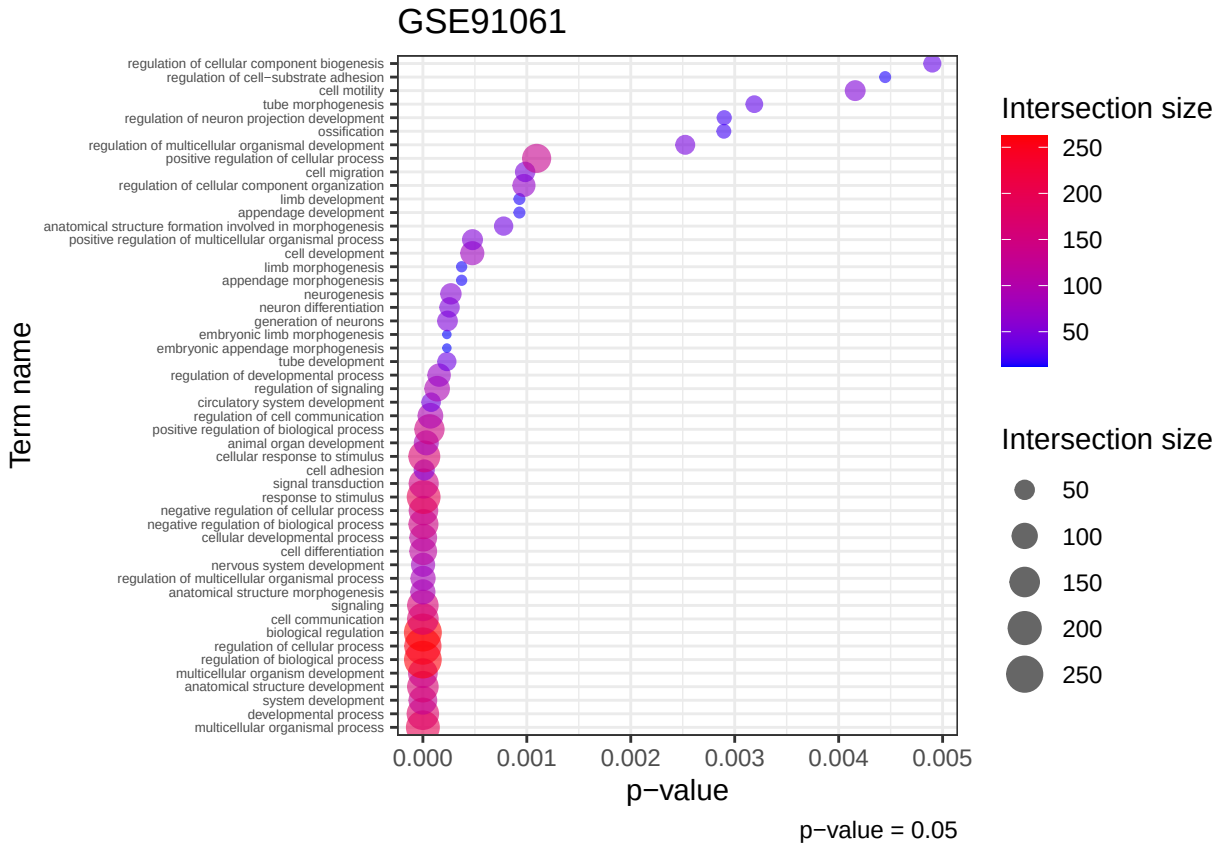
```
prop.table(table(df$query))*100
```

```
##
## GSE22155_02 GSE22155_47    GSE35640    GSE61992    GSE91061
##    19.54225    21.83099    19.71831    21.65493    17.25352
```

```
#table(df$term_name)
#prop.table(table(df$term_name))*100
df2 <- as.data.frame(table(df$term_name, df$query))
#length(unique(df2$Var1))
write.table(df, file = "./Venn_diagrams_cell_type_treatments_vs_untreated/Neutrophils_GOBPs.txt", sep =
write.table(df2, file = "./Venn_diagrams_cell_type_treatments_vs_untreated/Neutrophils_GOBPs_freq.txt",
#rm(jVenn_Neutrophiles_vs_GSE22155_02, jVenn_Neutrophiles_vs_GSE22155_47, jVenn_Neutrophiles_vs_GSE3564
```

Barplot of the top GO-BPs:

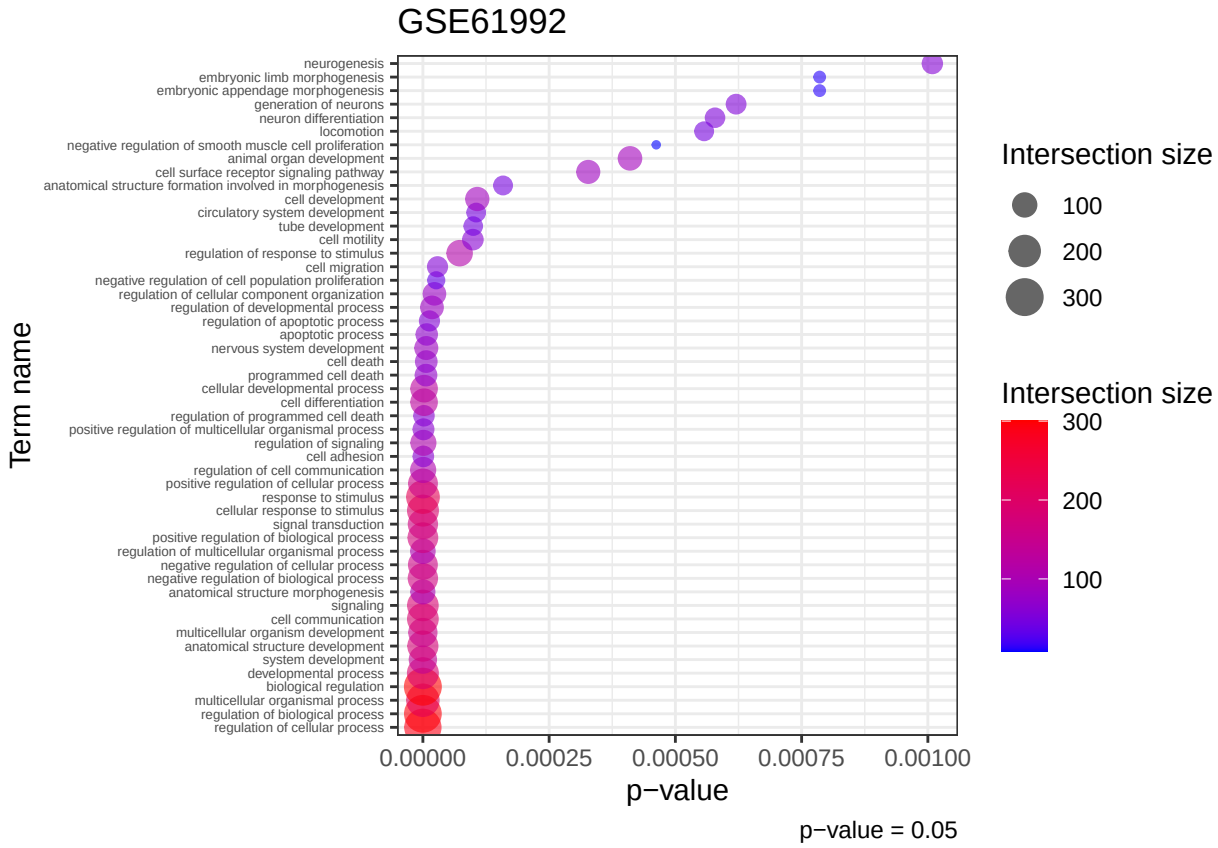
```
plot_gobps <- function(study, n = 50){
df2 <- df[df$source == "GO:BP",]
df2_GSE91061 <- df2[df2$query == study,]
df2_GSE91061 <- df2_GSE91061[1:n,]
ggplot(df2_GSE91061, aes(p_value, reorder(term_name,p_value))) + ggtitle(study) +
  geom_point(aes(p_value, reorder(term_name,p_value), colour = intersection_size, size = intersection_si
  theme_bw() + theme(axis.text.y.left = element_text(size = 5)) +
  labs(x= "p-value", y="Term name", colour = "Intersection size", size = "Intersection size", caption =
  scale_color_gradient(low="blue", high="red")
}

plot_gobps(study = "GSE91061")
```




```
print("\n")
```

```
## [1] "\n"
```

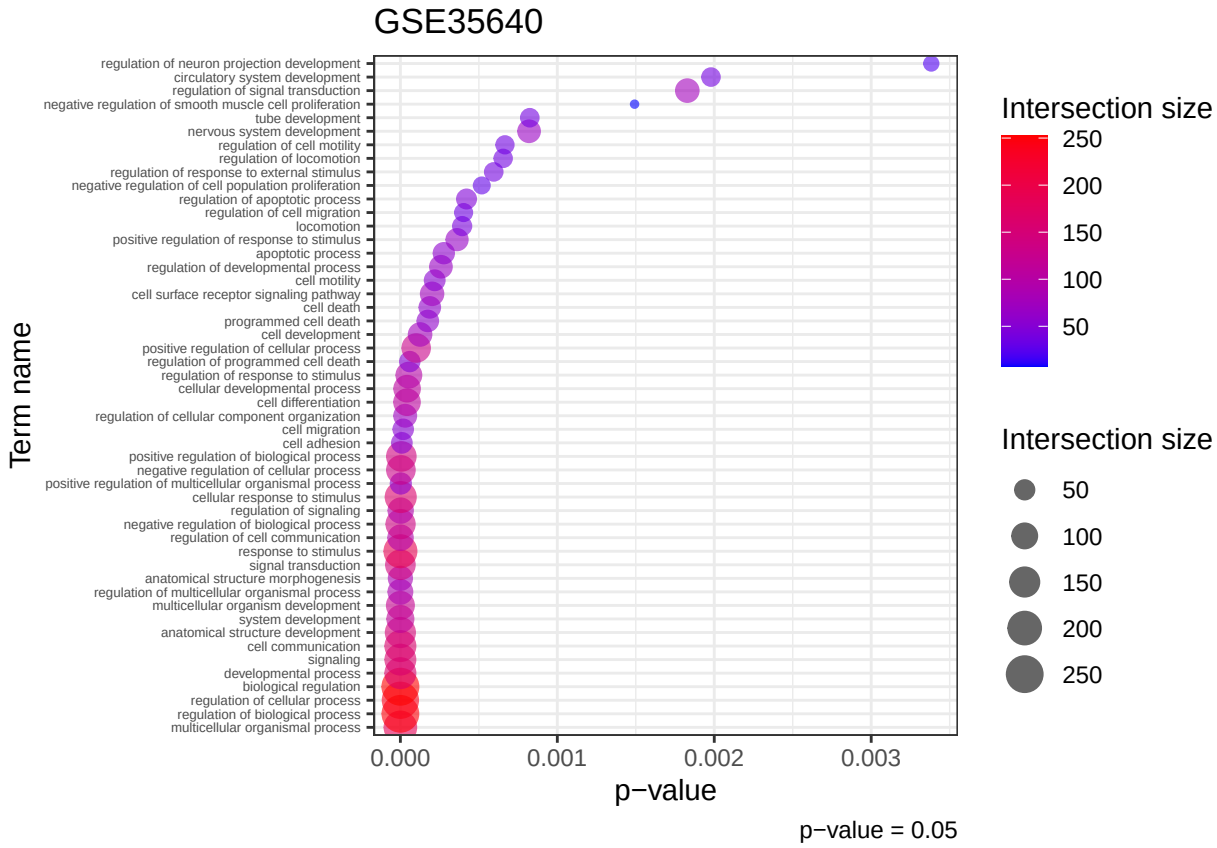
```
plot_gobps(study = "GSE61992")
```




```
print("\n")
```

```
## [1] "\n"
```

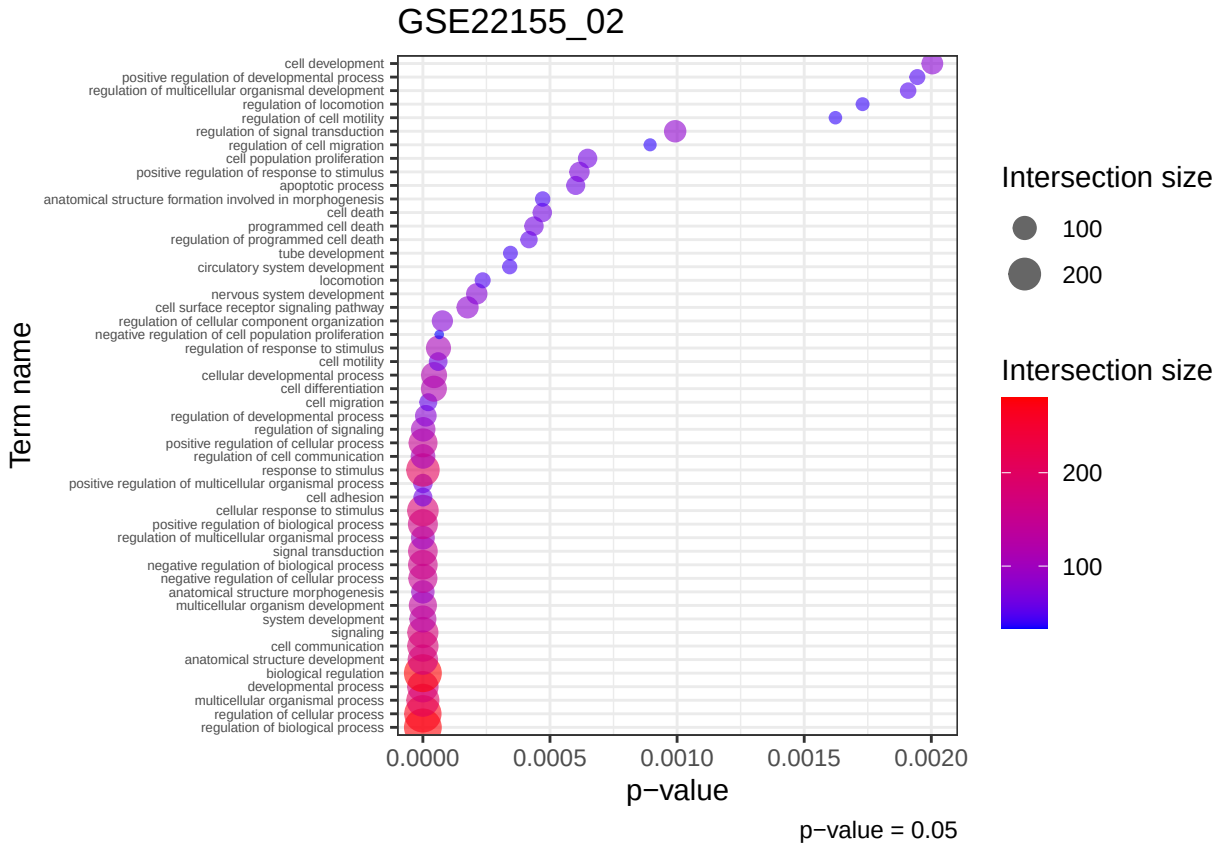
```
plot_gobps(study = "GSE35640")
```




```
print("\n")
```

```
## [1] "\n"
```

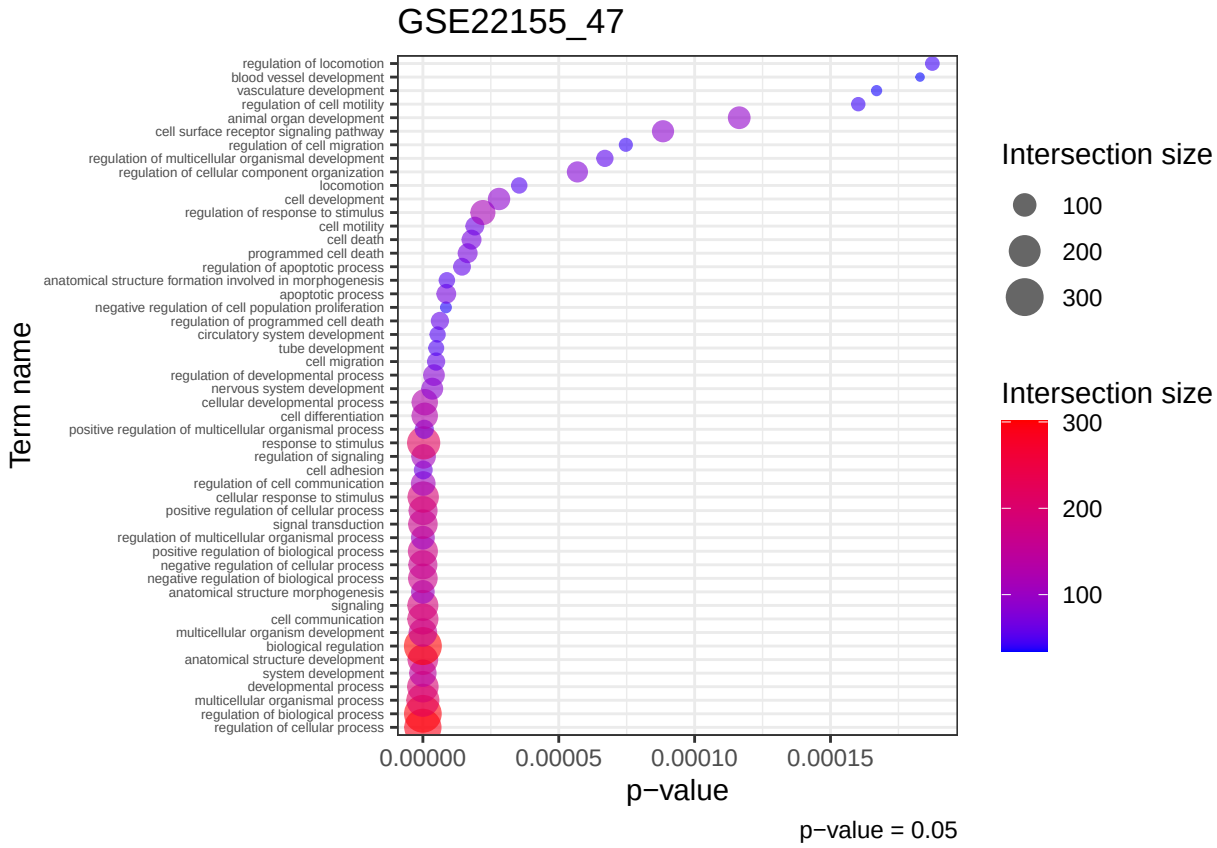
```
plot_gobps(study = "GSE22155_02")
```




```
print("\n")
```

```
## [1] "\n"
```

```
plot_gobps(study = "GSE22155_47")
```



5 Exclusive

En esta sección analizo los DEGs exclusivos para cada estudio, y realizo un multi-análisis para cada estudio. Los resultados los guardo como archivos gem específicos para usar en Cytoscape.

Primero realizo el pathway multianálisis:

```
multi_gostres <- gost(query = list("GSE61992" = jVenn_Neutrophiles$GSE61992[which(is.na(jVenn_Neutrophil
                                   "GSE91061" = jVenn_Neutrophiles$GSE91061[which(is.na(jVenn_Neutrophiles$GSE
                                   ),
                      evcodes = TRUE, multi_query = FALSE,
                      sources = c("GO", "REAC", "MIRNA", "CORUM", "HP", "HPA", "WP"))
head(multi_gostres$result[,1:7])
```

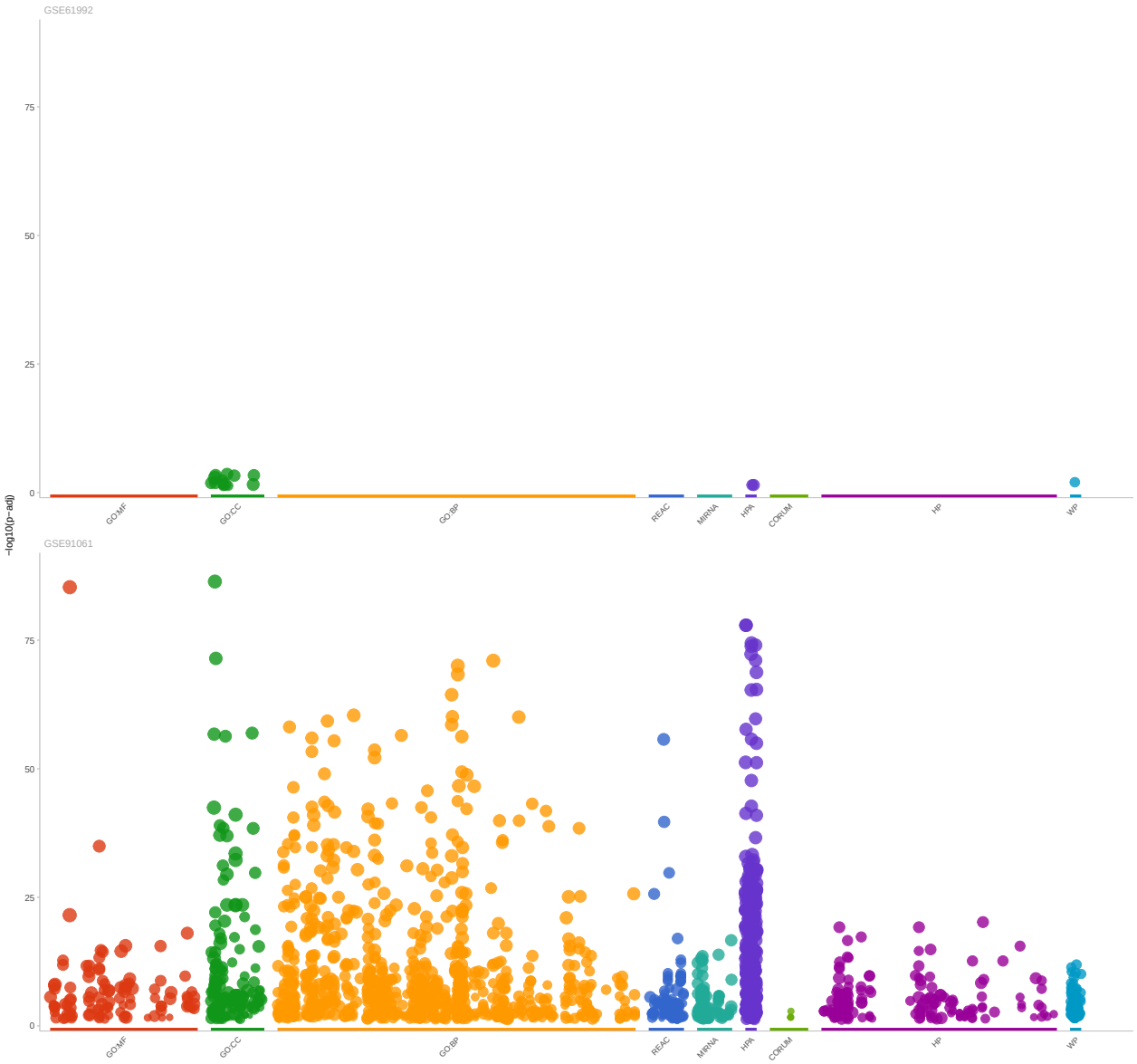
```
##      query significant      p_value term_size query_size intersection_size
## 1 GSE61992        TRUE 0.0002506850      1521         57               16
## 2 GSE61992        TRUE 0.0003981464      1194         57               14
## 3 GSE61992        TRUE 0.0004224308      1200         57               14
## 4 GSE61992        TRUE 0.0004986246      1217         57               14
## 5 GSE61992        TRUE 0.0014433451     12345         57               48
## 6 GSE61992        TRUE 0.0041492770       312         57                7
##   precision
## 1 0.2807018
## 2 0.2456140
## 3 0.2456140
## 4 0.2456140
## 5 0.8421053
```

```
## 6 0.1228070
```

Estos resultados son los que utilizo para generar los archivos gem que archivo para usar más tarde en Cytoscape.

```
# format to GEM
gem <- multi_gostres$result[,c("query", "term_id", "term_name", "p_value", "intersection_size")]
colnames(gem) <- c("query", "GO.ID", "Description", "p.Val", "Genes")
gem$FDR <- gem$p.Val
gem$Phenotype = "+1"

gem %>% group_by(query) %>%
  group_walk(~
    write.table(data.frame(.x[,c("GO.ID", "Description", "p.Val", "FDR", "Phenotype", "Genes")]),
                file = paste0("./Treatment_comparisons/gProfiler_Neutrophils_Only_geneset_", unique(.y$
                sep = "\t", quote = F, row.names = F))
```

```
p <- gostplot(multi_gostres, capped = FALSE, interactive = F)
p
```



Proporciones de los pathways para cada estudio:

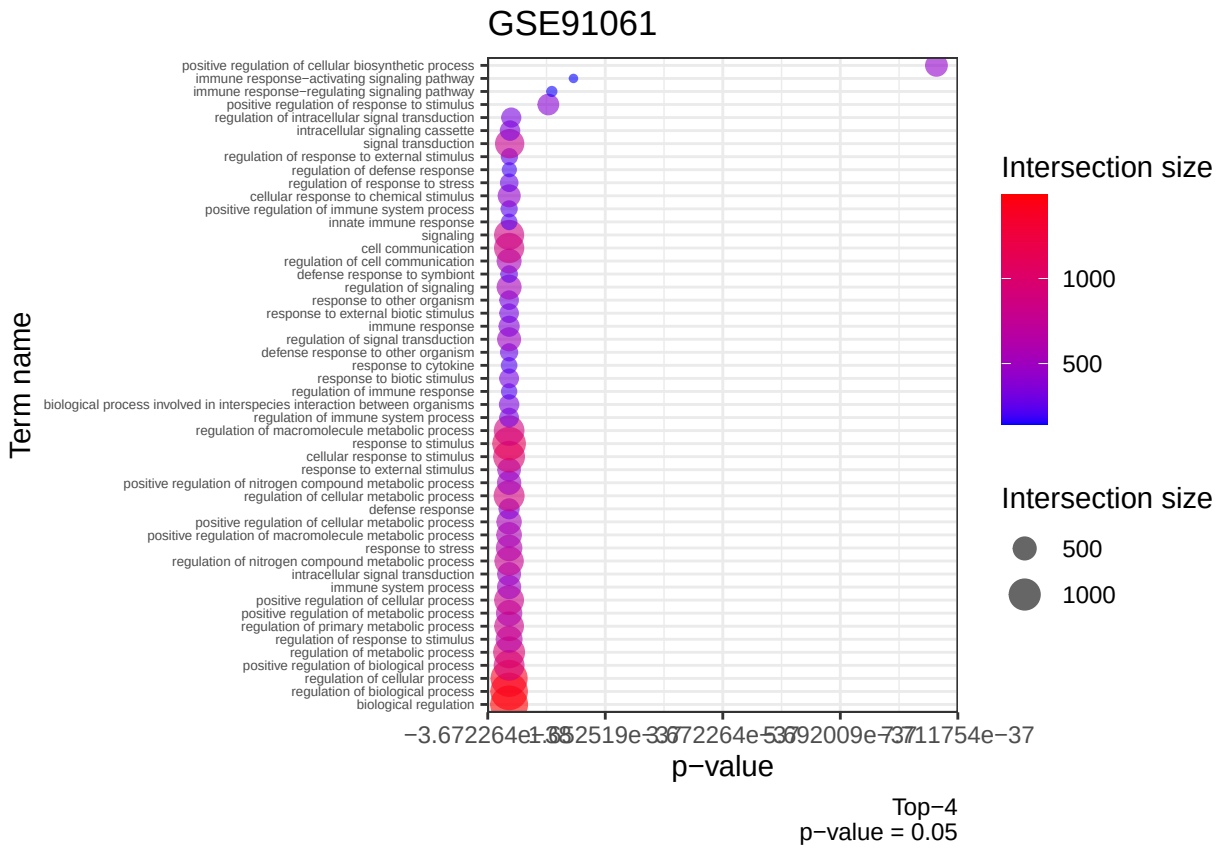
```
df <- as.data.frame(multi_gostres$result[,1:13])
table(df$query)
```

```
##
## GSE61992 GSE91061
##       16     1814
```

```
write.table(gem, file = "./Treatment_comparisons/gProfiler_Neutrophils_Only_genesets.txt", sep = "\t",qu
```

```
plot_gobps <- function(study){
df2_GSE91061 <- df2[df2$query == study,]
ggplot(df2_GSE91061, aes(p_value, reorder(term_name,p_value))) + ggtitle(study) +
  geom_point(aes(p_value, reorder(term_name,p_value), colour = intersection_size, size = intersection_si
  theme_bw() + theme(axis.text.y.left = element_text(size = 5)) +
  labs(x= "p-value", y="Term name", colour = "Intersection size", size = "Intersection size", caption =
  scale_color_gradient(low="blue", high="red")
```

```
}
df2 <- df[df$source == "GO:BP",]
df2 <- df2[1:50,]
plot_gobps(study = "GSE91061")
```




```
print("\n")
```

```
## [1] "\n"
```

```
df2 <- df[df$source == "GO:BP",]
plot_gobps(study = "GSE61992")
```

6 Máximo solapamiento

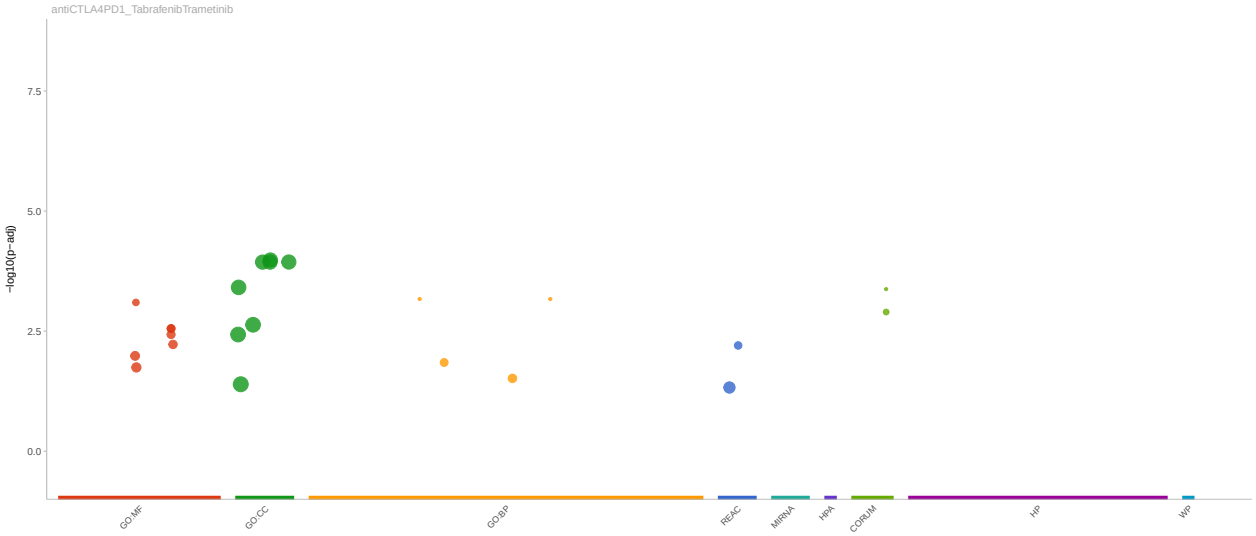
En esta sección analizo los DEGs que más se comparten entre los estudios para observar los mecanismos que pudieran haber en común entre los tratamientos.

Primero realizo el pathway multianálisis:

```
multi_gostres <- gost(query = list("antiCTLA4PD1_TabrafenibTrametinib" = jVenn_Neutrophiles$`GSE91061|GS
                                   ),
                      evcodes = TRUE, multi_query = FALSE,
                      sources = c("GO", "REAC", "MIRNA", "CORUM", "HP", "HPA", "WP"))
head(multi_gostres$result[,1:7])
```

```
##                              query significant      p_value term_size
## 1 antiCTLA4PD1_TabrafenibTrametinib        TRUE 0.0004219436         2
## 2 antiCTLA4PD1_TabrafenibTrametinib        TRUE 0.0012645829         3
## 3 antiCTLA4PD1_TabrafenibTrametinib        TRUE 0.0006783004         2
## 4 antiCTLA4PD1_TabrafenibTrametinib        TRUE 0.0006783004         2
## 5 antiCTLA4PD1_TabrafenibTrametinib        TRUE 0.0142149806         7
## 6 antiCTLA4PD1_TabrafenibTrametinib        TRUE 0.0304230305        10
##   query_size intersection_size precision
## 1          7                 2 0.2857143
## 2          7                 2 0.2857143
## 3         15                 2 0.1333333
## 4         15                 2 0.1333333
## 5         15                 2 0.1333333
## 6         15                 2 0.1333333
```

```
p <- gostplot(multi_gostres, capped = FALSE, interactive = F)
p
```



Estos resultados son los que utilizo para generar los archivos gem que archivo para usar más tarde en Cytoscape.

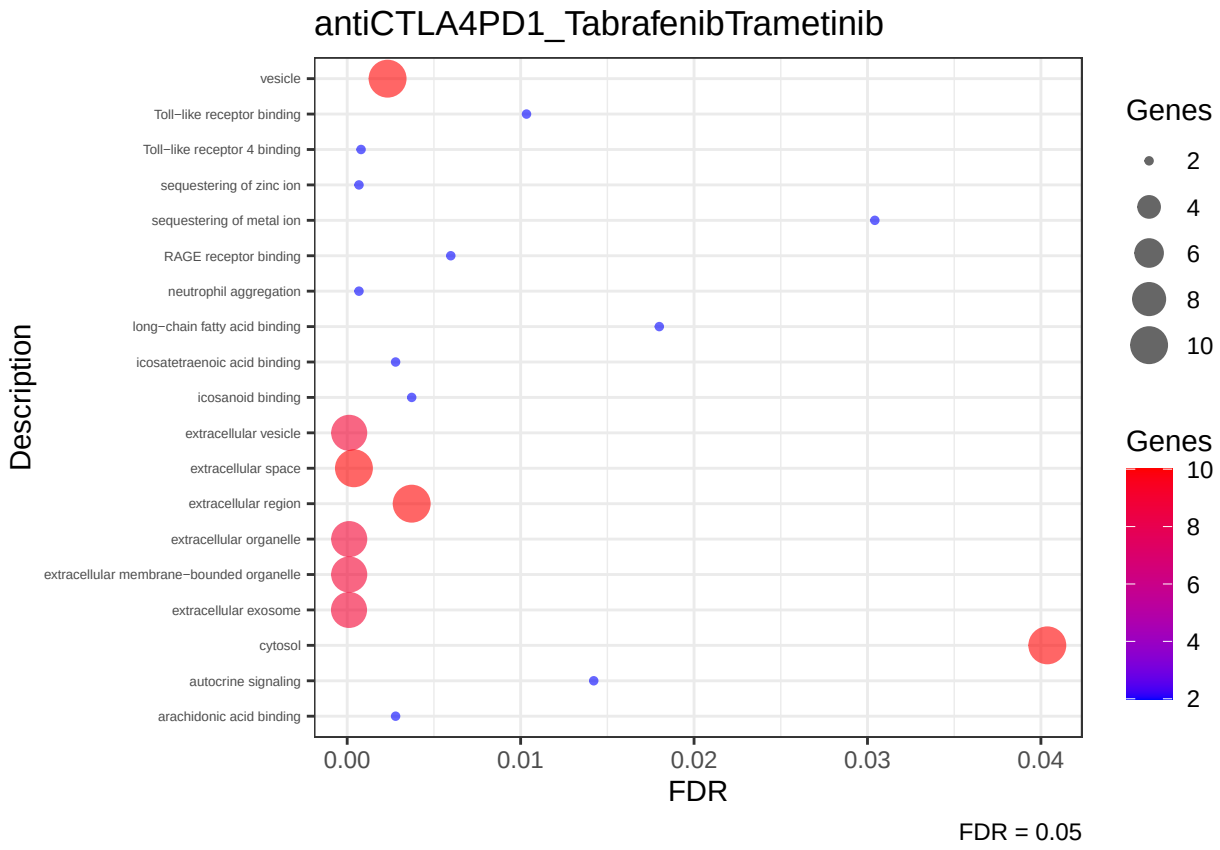
```
# format to GEM
gem <- multi_gostres$result[,c("query", "term_id", "term_name", "p_value", "intersection_size")]
colnames(gem) <- c("query", "GO.ID", "Description", "p.Val", "Genes")
gem$FDR <- gem$p.Val
gem$Phenotype = "+1"

gem %>% group_by(query) %>%
  group_walk(~
                   write.table(data.frame(.x[,c("GO.ID", "Description", "p.Val", "FDR", "Phenotype", "G
                file = paste0("./Treatment_comparisons/gProfiler_Neutrophils_maxOverlap_geneset_", uniq
                sep = "\t", quote = F, row.names = F))
```

Al mirar los procesos por cada comparación salieron demasiados para poder visualizarlos bien, por lo que decidí restringir a los procesos GO-BPs únicamente y guardarlos en pdf.

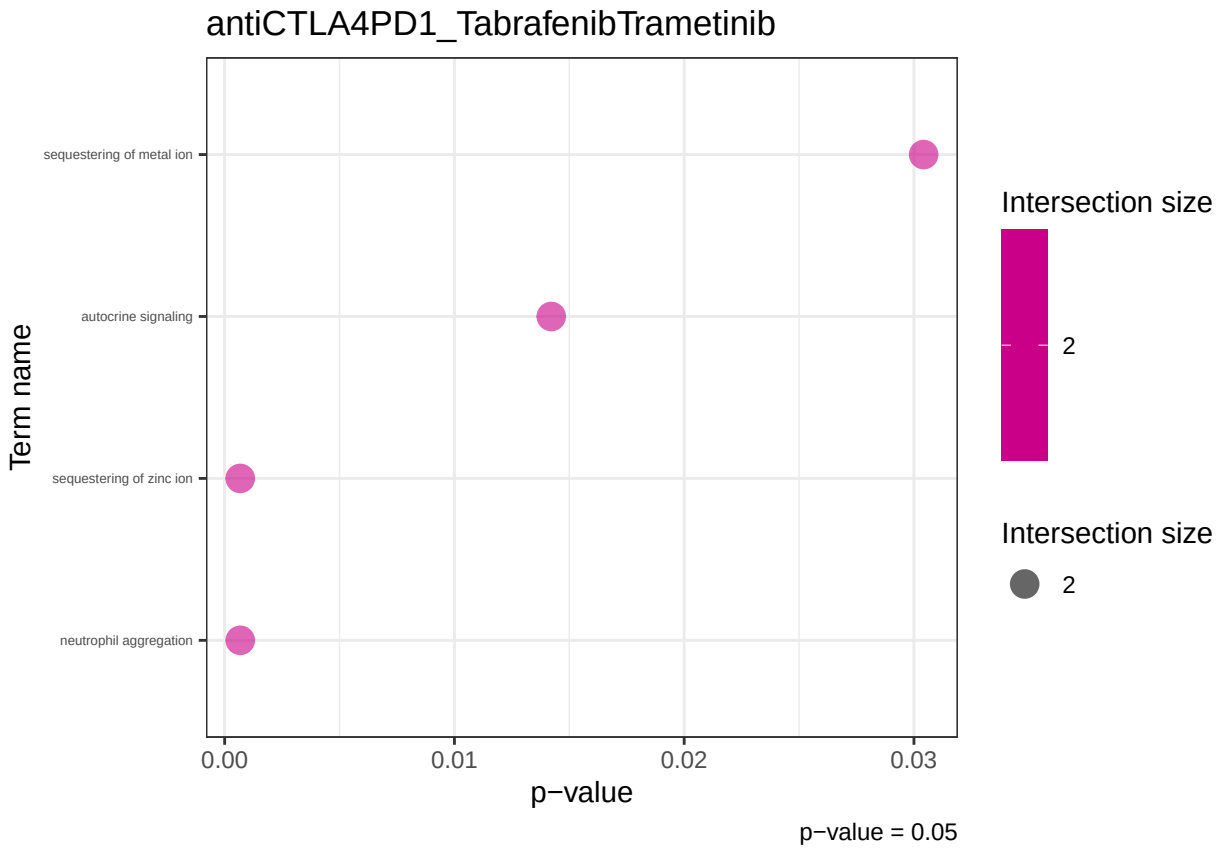
```
unique(gem$query)
```

```
## [1] "antiCTLA4PD1_TabrafenibTrametinib"
```

```
gem2 <- gem[grep("GO",gem$GO.ID),]
plot_gobps <- function(query, cutoff = 0.05) {
gem2 <- gem2[gem2$FDR <= cutoff,]
gem2_1 <- gem2[gem2$query == query, ]
ggplot(gem2_1, aes(FDR, Description)) + ggtitle(query) +
  geom_point(aes(FDR, Description, colour = Genes, size = Genes), alpha = 0.6)+
  theme_bw() + theme(axis.text.y.left = element_text(size = 5)) +
  labs(caption = paste0("FDR = ",cutoff))+
  scale_color_gradient(low="blue", high="red")
}

plot_gobps("antiCTLA4PD1_TabrafenibTrametinib")
```




```
df <- as.data.frame(multi_gostres$result[,1:13])
df2 <- df[df$source == "GO:BP",]
plot_gobps <- function(study){
df2_GSE91061 <- df2[df2$query == study,]
ggplot(df2_GSE91061, aes(p_value, reorder(term_name,p_value))) + ggtitle(study) +
  geom_point(aes(p_value, reorder(term_name,p_value), colour = intersection_size, size = intersection_si
  theme_bw() + theme(axis.text.y.left = element_text(size = 5)) +
  labs(x= "p-value", y="Term name", colour = "Intersection size", size = "Intersection size", caption =
  scale_color_gradient(low="blue", high="red")
}
plot_gobps(study = "antiCTLA4PD1_TabrafenibTrametinib")
```


antiCTLA4PD1_TabrafenibTrametinib
sequestering of metal ion
autocrine signaling
sequestering of zinc ion
neutrophil aggregation
Term name
0.00
0.01
0.02
0.03
p−value
p−value = 0.05
Intersection size
2
Intersection size
2