Pathway Analysis - B cells
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 las células B obtenidos en jVenn.

```
setwd("~/Desktop/ELENA_UOC/TFM")
jVenn_Bcells_vs_GSE22155_02 <- read_csv("DEGs/Venn_diagrams_cell_type_treatments_vs_untreated/jVenn_Bce
head(jVenn_Bcells_vs_GSE22155_02) # sección de los resultados
```

```
## # A tibble: 6 x 3
##   GSE50509  GSE22155_02 `GSE50509|GSE22155_02`
##   <chr>     <chr>       <chr>
## 1 MXRA7     MS4A1       TLN1
## 2 CDK20     CARD11      CD79A
## 3 HTN1      NIBAN3      NDRG3
## 4 SCN1B     CR2         CD19
## 5 HPN       FCER2       DCTN3
## 6 FGF14-AS2 BANK1       VPREB3
```

```
jVenn_Bcells_vs_GSE22155_47 <- read_csv("DEGs/Venn_diagrams_cell_type_treatments_vs_untreated/jVenn_Bcel
jVenn_Bcells_vs_GSE61992 <- read_csv("DEGs/Venn_diagrams_cell_type_treatments_vs_untreated/jVenn_Bcells_
jVenn_Bcells_vs_GSE91061 <- read_csv("DEGs/Venn_diagrams_cell_type_treatments_vs_untreated/jVenn_Bcells_
jVenn_Bcells_vs_GSE35640 <- read_csv("DEGs/Venn_diagrams_cell_type_treatments_vs_untreated/jVenn_Bcells_
setwd("~/Desktop/ELENA_UOC/TFM/GOBPs") # Directorio de trabajo
jVenn_Bcells <- read_delim("~/Desktop/ELENA_UOC/TFM/DEGs/Venn_diagrams_cell_type_treatments_comparison/V
    delim = ",", escape_double = FALSE, trim_ws = TRUE)
```

4 Uncovered

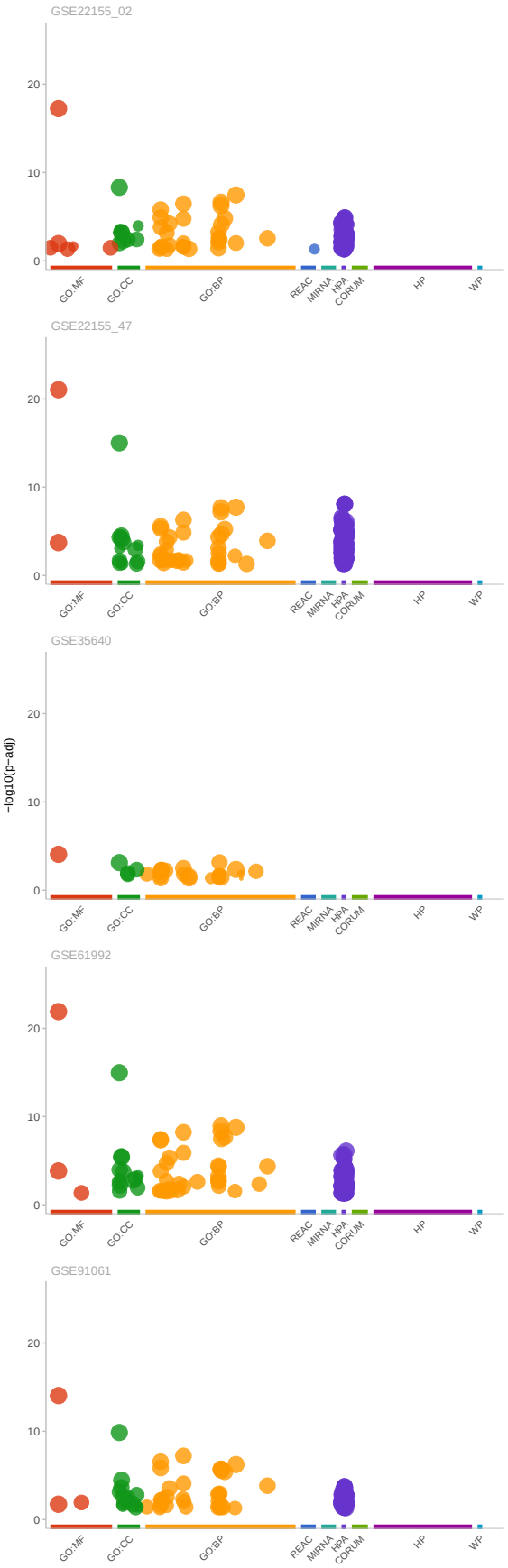
En esta sección contemplo la posibilidad que los DEGs únicos en el dataset con muestras sin tratar se traten de mecanismos que se han sido tratados en la cohorte con inmunotarapia.

Primero realizo el pathway multianálisis:

```
multi_gostres <- gost(query = list("GSE22155_02" = unique(jVenn_Bcells_vs_GSE22155_02$GSE50509[which(is
                                   "GSE22155_47" = jVenn_Bcells_vs_GSE22155_47$GSE50509[which(is.na(jVenn_Bcel
                                   "GSE35640" = jVenn_Bcells_vs_GSE35640$GSE50509[which(is.na(jVenn_Bcells_vs_
                                   "GSE91061" = jVenn_Bcells_vs_GSE91061$GSE50509[which(is.na(jVenn_Bcells_vs_
                                   "GSE61992" = jVenn_Bcells_vs_GSE61992$GSE50509[which(is.na(jVenn_Bcells_vs_
                                   ),
                      evcodes = TRUE, multi_query = FALSE,
                      sources = c("GO", "REAC", "MIRNA", "CORUM", "HP", "HPA", "WP"))
head(multi_gostres$result[,1:7])
```

```
##         query significant      p_value term_size query_size intersection_size
## 1 GSE22155_02        TRUE 3.524736e-08     12680        635              462
## 2 GSE22155_02        TRUE 2.316282e-07     11738        635              432
## 3 GSE22155_02        TRUE 3.462587e-07      6447        635              269
## 4 GSE22155_02        TRUE 5.757019e-07     12287        635              446
## 5 GSE22155_02        TRUE 1.620241e-06      6537        635              269
## 6 GSE22155_02        TRUE 1.259451e-05      5959        635              246
##   precision
## 1 0.7275591
## 2 0.6803150
## 3 0.4236220
## 4 0.7023622
## 5 0.4236220
## 6 0.3874016
```

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


GSE22155_02
GSE22155_47
GSE35640
GSE61992
GSE91061
−log10(p−adj)
20
10
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_Bcells_Only_
                sep = "\t", quote = F, row.names = F))
```

Proporciones de los pathways para cada estudio:

```
df <- as.data.frame(multi_gostres$result[,1:13])
df3 <- table(df$query)
table(df$query)
```

```
##
## GSE22155_02 GSE22155_47    GSE35640    GSE61992    GSE91061
##         126         191          27         118          89
```

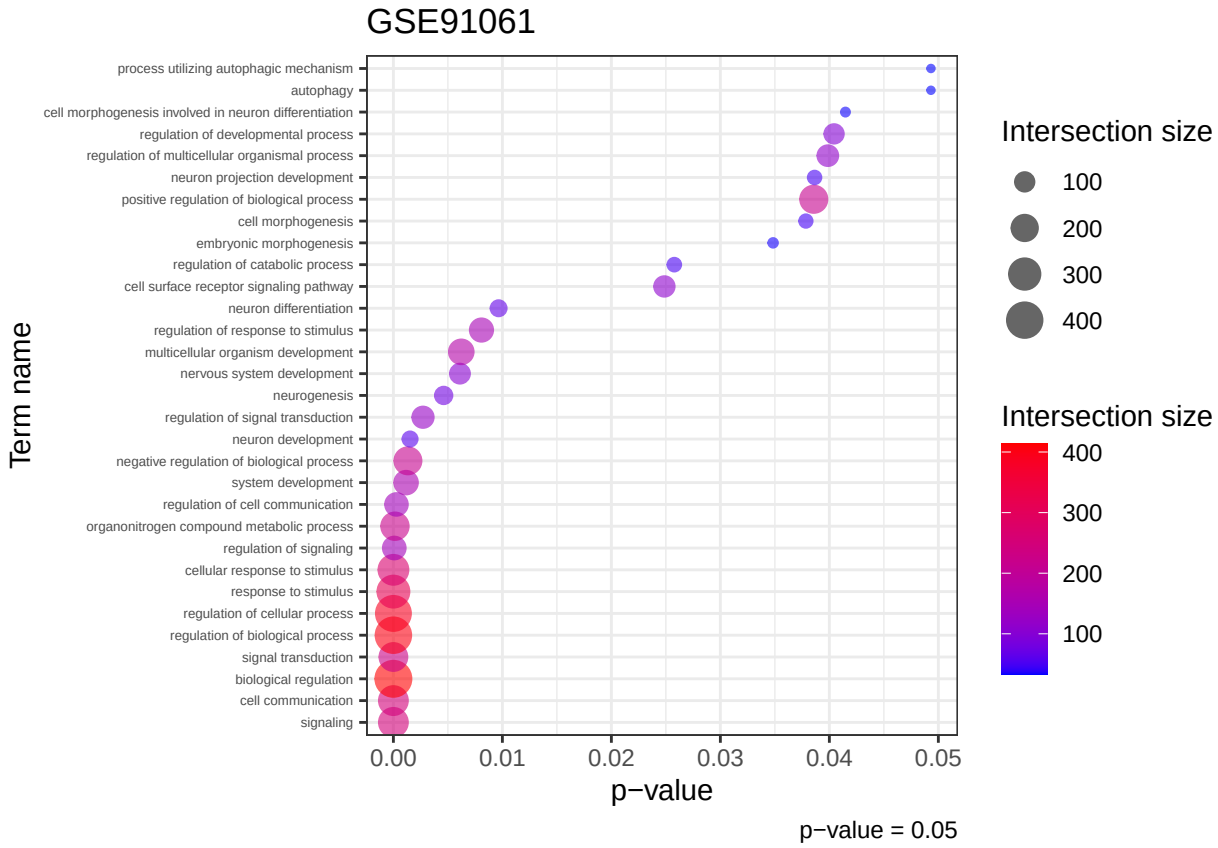
```
prop.table(table(df$query))*100
```

```
##
## GSE22155_02 GSE22155_47    GSE35640    GSE61992    GSE91061
##   22.867514   34.664247    4.900181   21.415608   16.152450
```

```
#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/Bcell_GOBPs.txt", sep = "\t",
write.table(df2, file = "./Venn_diagrams_cell_type_treatments_vs_untreated/Bcell_GOBPs_freq.txt", sep =
#rm(jVenn_Bcells_vs_GSE22155_02, jVenn_Bcells_vs_GSE22155_47, jVenn_Bcells_vs_GSE35640, jVenn_Bcells_vs
```

Barplot of the top GO-BPs:

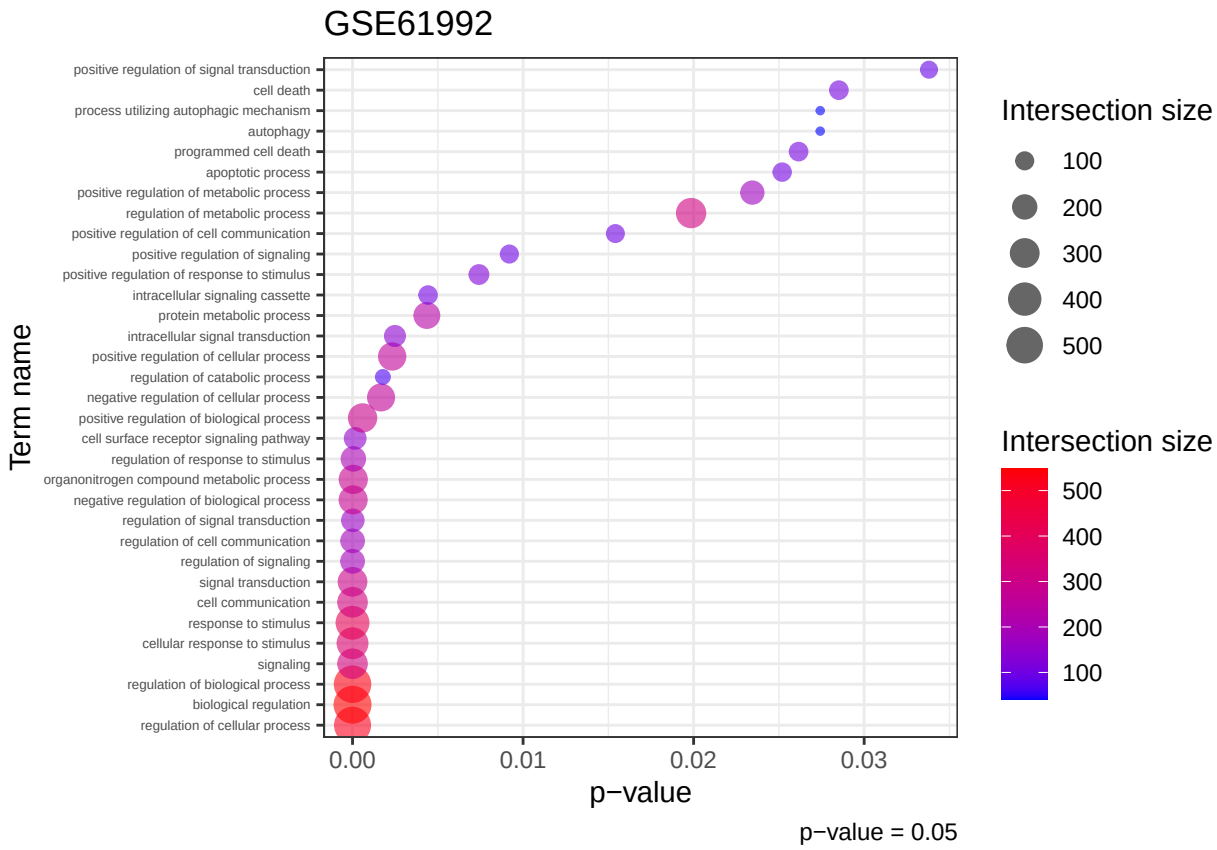
```
df2 <- df[df$source == "GO:BP",]
plot_gobps <- function(study){
df2_GSE91061 <- df2[df2$query == study,]
ggplot(df2_GSE91061, aes(p_value, reorder(term_name,p_value))) + ggtitle(study) +
  geom_point(aes(p_value, reorder(term_name,p_value), colour = intersection_size, size = intersection_si
  theme_bw() + 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 = "GSE91061")
```




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

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

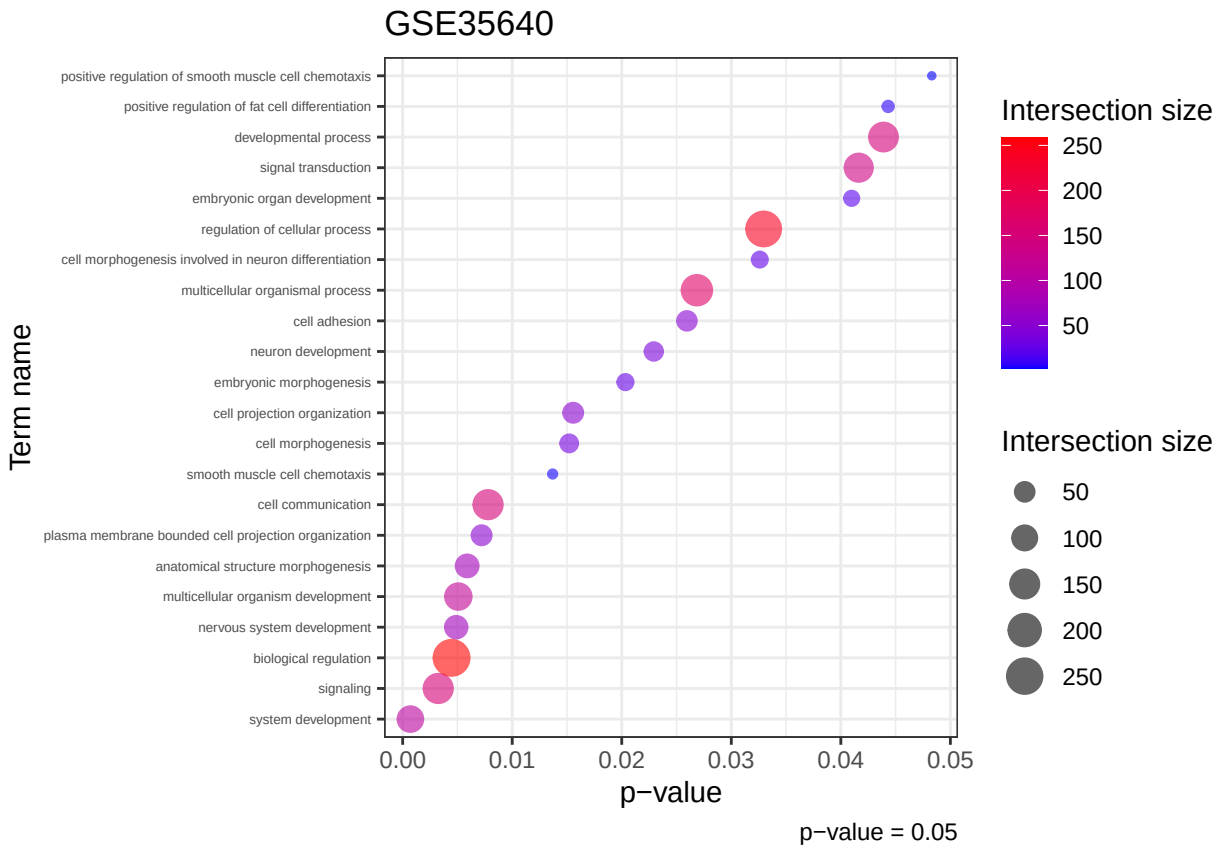
```
plot_gobps(study = "GSE61992")
```




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

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

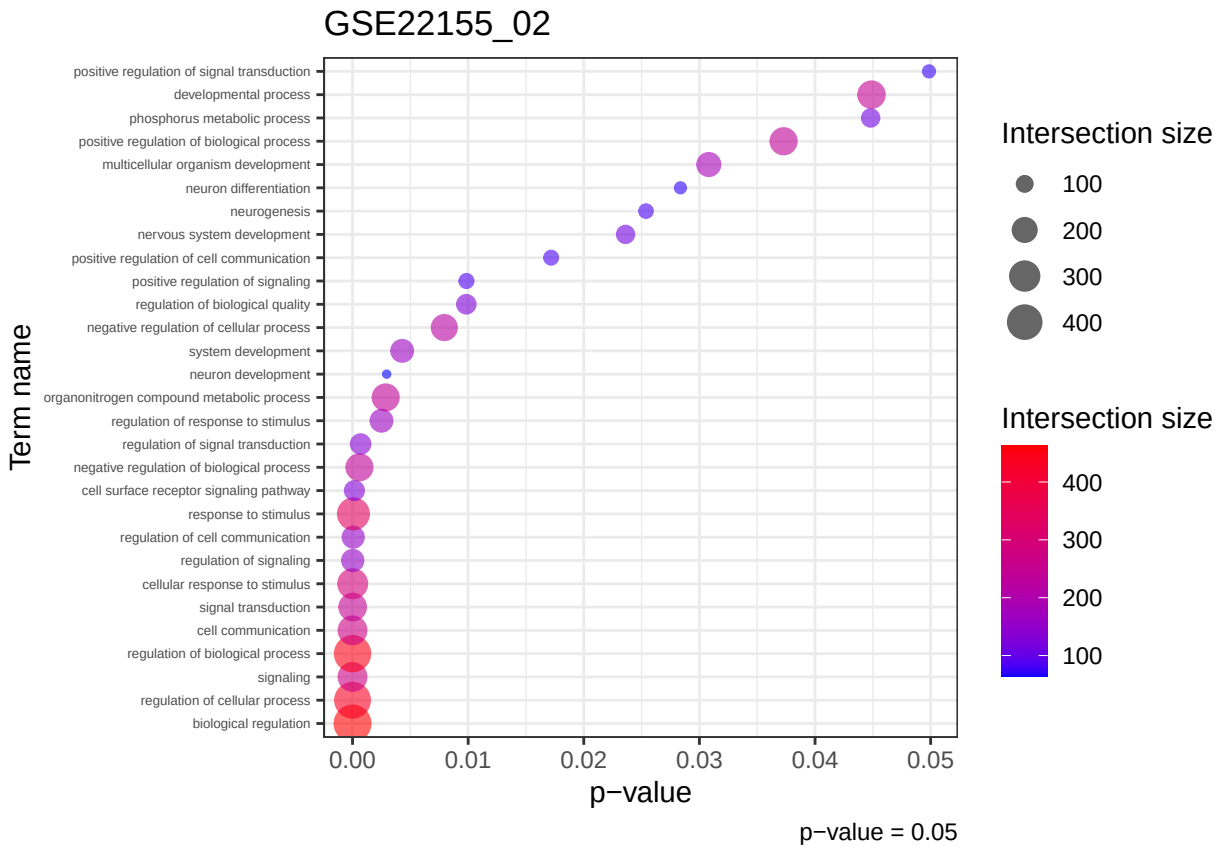
```
plot_gobps(study = "GSE35640")
```




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

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

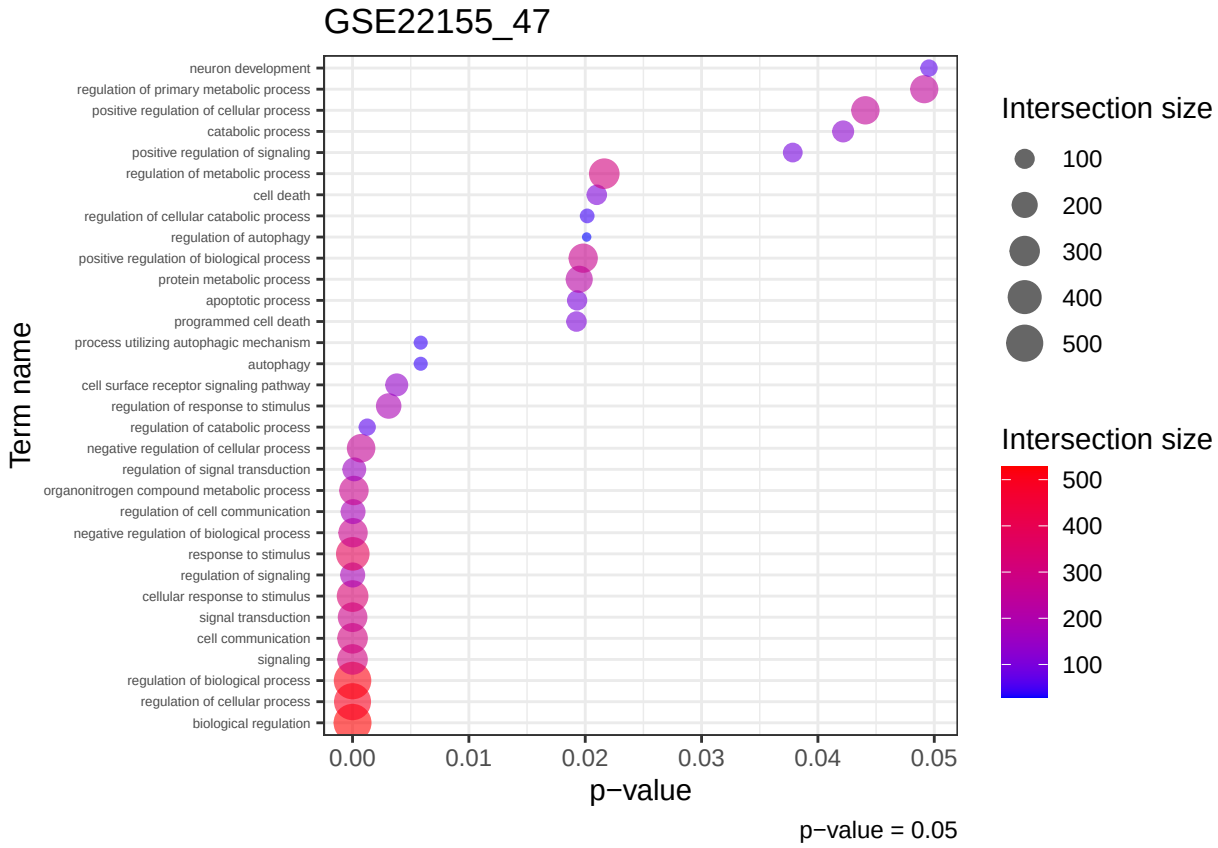
```
plot_gobps(study = "GSE22155_02")
```




```
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("GSE91061" = jVenn_Bcells$GSE91061[which(is.na(jVenn_Bcells$GSE91061)
                                   "GSE61992" = jVenn_Bcells$GSE61992[which(is.na(jVenn_Bcells$GSE61992) == F
                                   ),
                      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 1.578067e-03      3855        122               45
## 2 GSE61992        TRUE 2.577132e-11     12345        123              107
## 3 GSE61992        TRUE 7.833757e-07      4777        123               56
## 4 GSE61992        TRUE 1.267838e-04      5487        123               56
## 5 GSE61992        TRUE 1.862430e-04      2055        123               30
## 6 GSE61992        TRUE 2.349428e-04      1521        123               25
##   precision
## 1 0.3688525
## 2 0.8699187
## 3 0.4552846
## 4 0.4552846
## 5 0.2439024
```

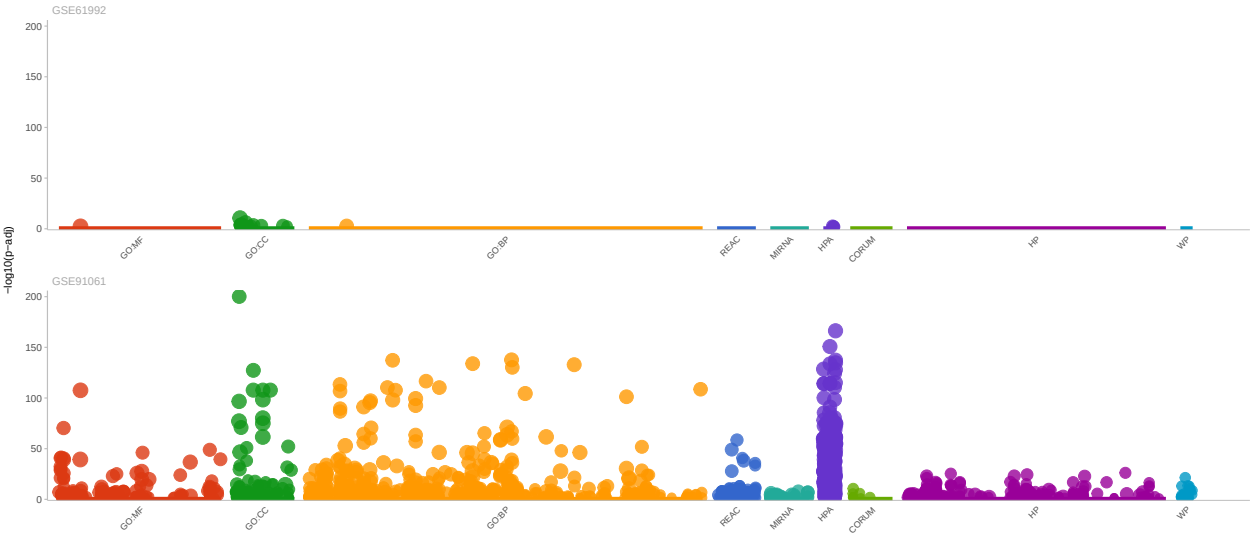
```
## 6 0.2032520
```

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_Bcells_Only_geneset_", unique(.y$query
                sep = "\t", quote = F, row.names = F))
```

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



Proporciones de los pathways para cada estudio:

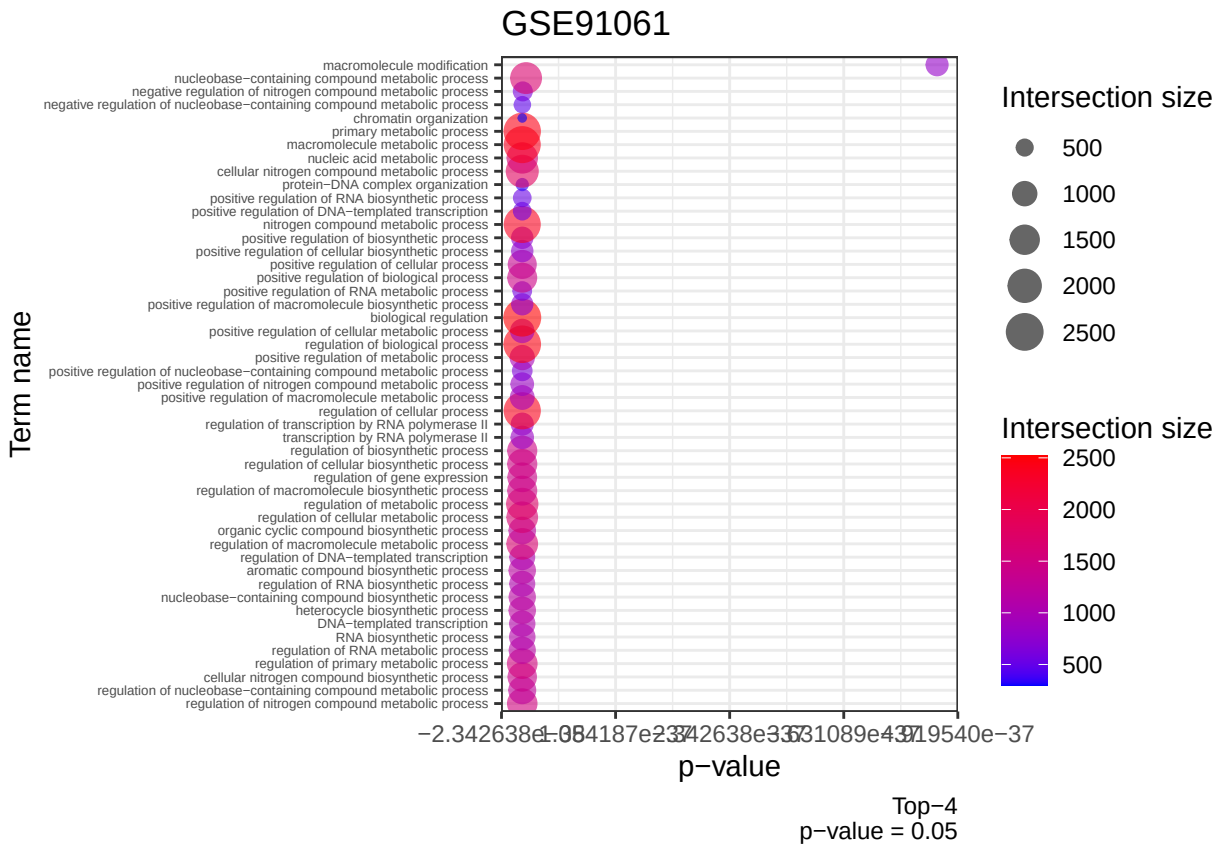
```
df <- as.data.frame(multi_gostres$result[,1:13])
table(df$query)
```

```
##
## GSE61992 GSE91061
##       17     1571
```

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

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

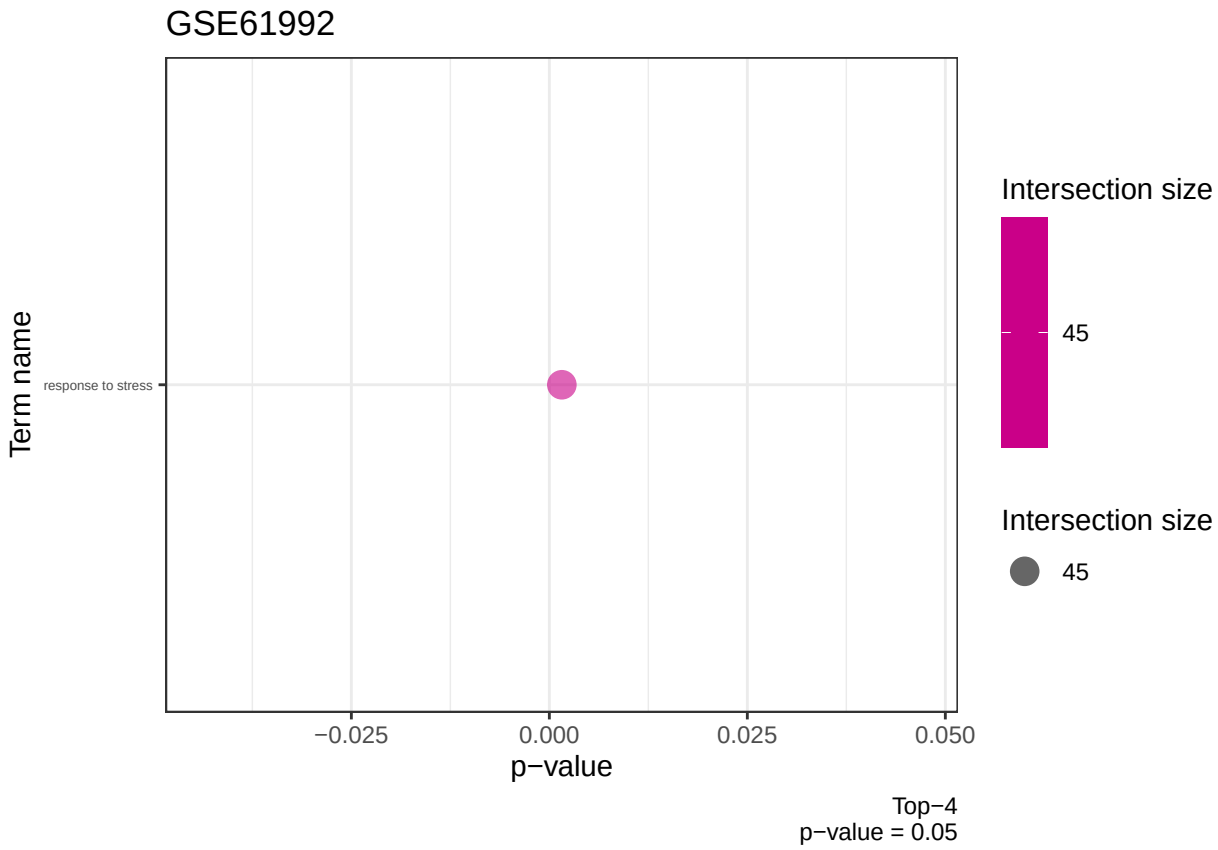
```
df2 <- df2[1:50,]
plot_gobps(study = "GSE91061")
```




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

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

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



6 Máximo solapamiento

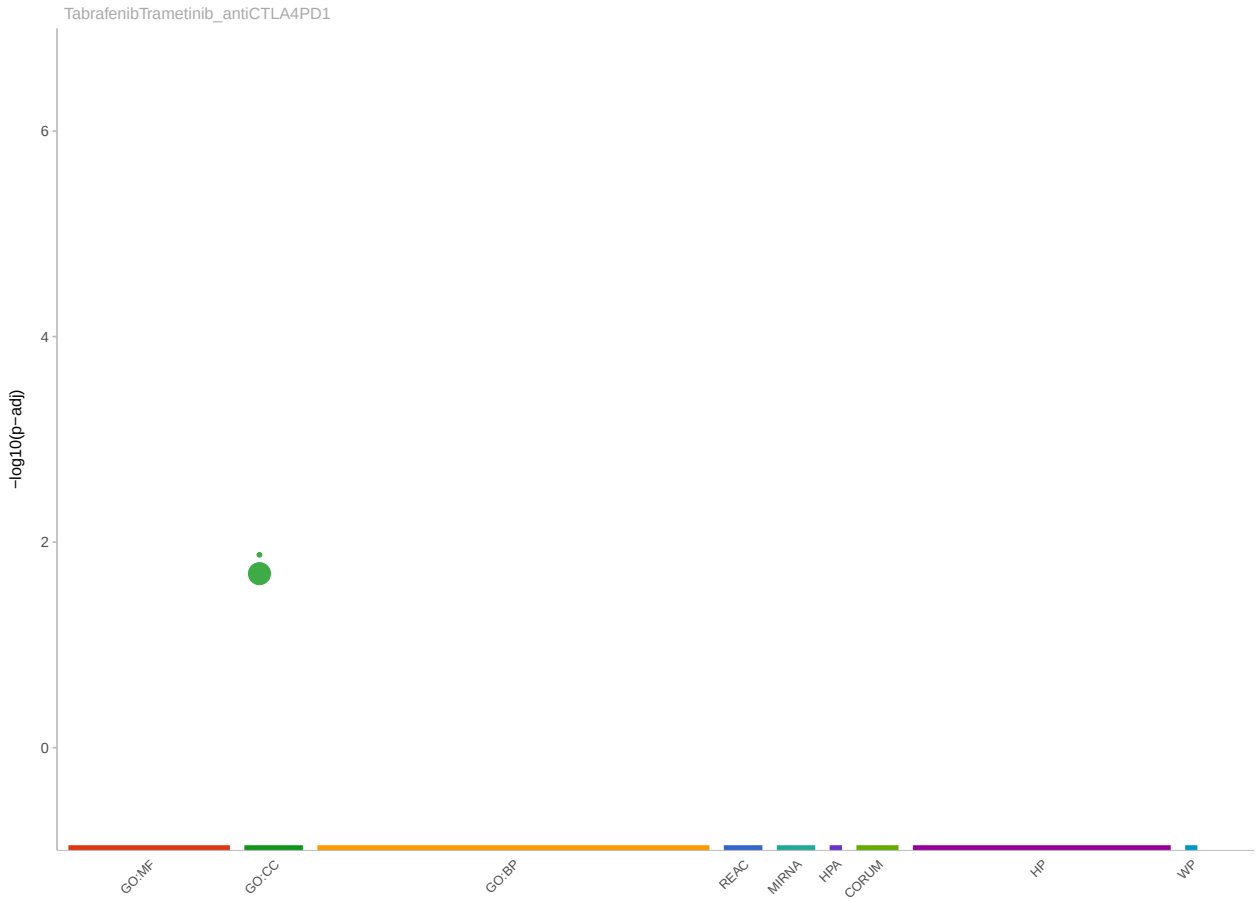
En esta sección analizo los DEGs que más se comparten entre los estudios para observar los mecanismos que pudieran haber en común entre los tratamientos.

Primero realizo el pathway multianálisis:

```
multi_gostres <- gost(query = list(
"TabrafenibTrametinib_antiCTLA4PD1" =
jVenn_Bcells$`GSE91061|GSE61992`[which(is.na(jVenn_Bcells$`GSE91061|GSE61992`)==F)]
                              ),
                      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
## 1 TabrafenibTrametinib_antiCTLA4PD1        TRUE 0.01328509         9         30
## 2 TabrafenibTrametinib_antiCTLA4PD1        TRUE 0.02026241        11         30
##   intersection_size  precision
## 1                 2 0.06666667
## 2                 2 0.06666667
```

```
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", "Genes")]),
                file = paste0("./Treatment_comparisons/gProfiler_Bcells_maxOverlap_geneset_", unique(.y$
                sep = "\t", quote = F, row.names = F))
```

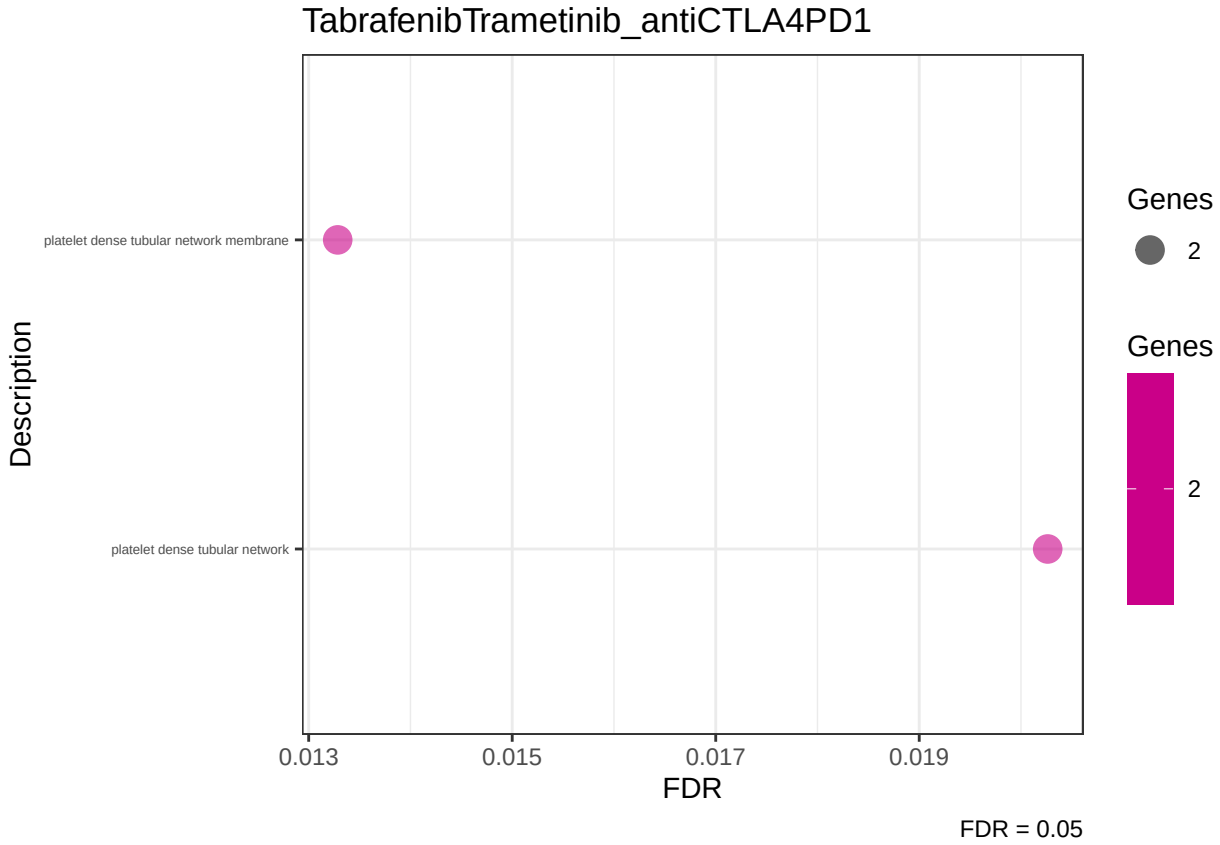
Al mirar los procesos por cada comparación salieron demasiados para poder visualizarlos bien, por lo que decidí restringir a los procesos GO-BPs únicamente y guardarlos en pdf.

```
#unique(gem$query)
gem2 <- gem[grep("GO",gem$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)) +
```

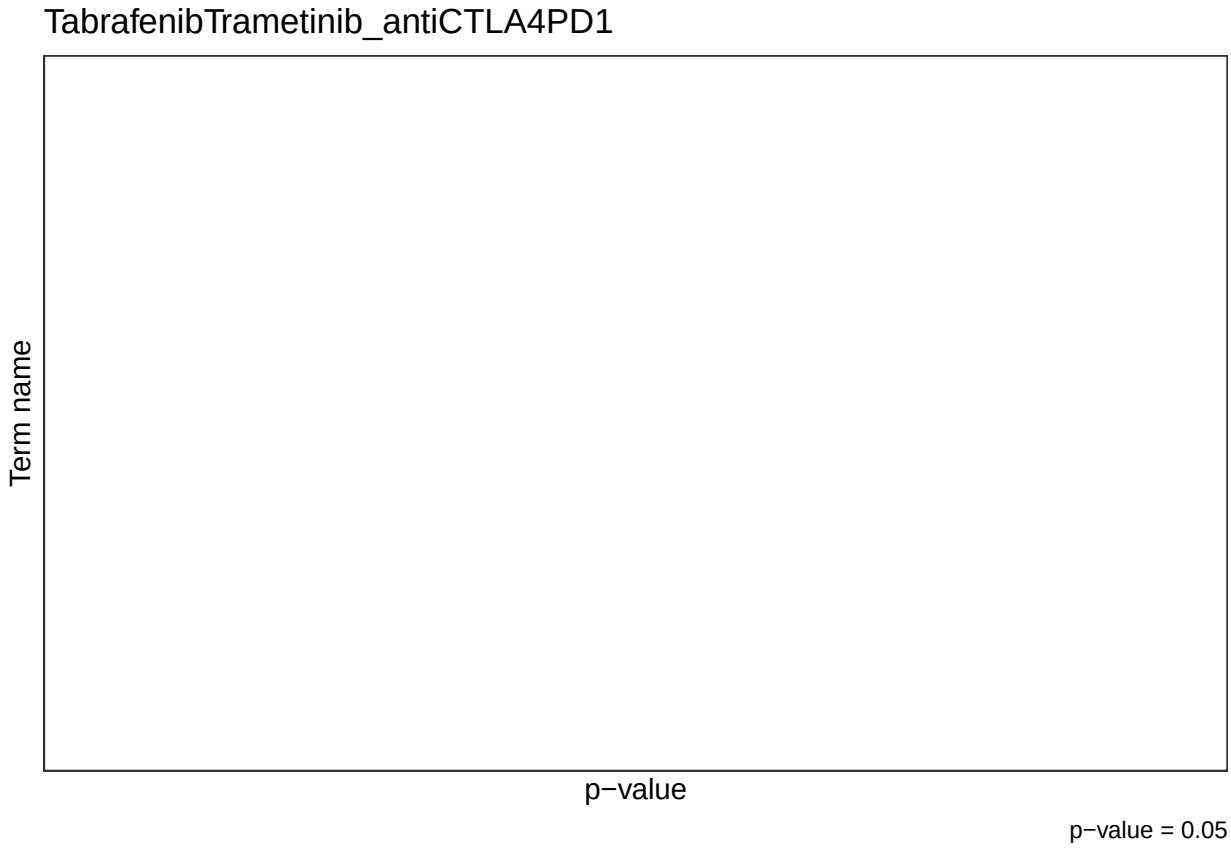
```
  labs(caption = paste0("FDR = ",cutoff))+
  scale_color_gradient(low="blue", high="red")
}
pdf(file = paste0("./Treatment_comparisons/gProfiler_Bcells_maxOverlap_geneset_TabrafenibTrametinib_anti
plot_gobps("TabrafenibTrametinib_antiCTLA4PD1")
dev.off()
```

```
## pdf
##   2
```

```
plot_gobps("TabrafenibTrametinib_antiCTLA4PD1")
```




```
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 = "TabrafenibTrametinib_antiCTLA4PD1")
```


TabrafenibTrametinib_antiCTLA4PD1
Term name
p−value
p−value = 0.05