

# Glioma Cancer Differential Gene Analysis

Chigozie Nkwocha

2024-10-18

## Overview

In this notebook, we will perform differential gene expression (DGE) analysis on our already preprocessed glioma cancer datasets. At the preprocessing stage, we performed the winsorised outlier handling method to clip values below 5th and above 95th percentiles to values at the 5th & 95th percentiles (90% winsorisation). We also performed batch effect correction to remove any variability that comes from different experimental conditions since our datasets come from different sources. So here, we will perform DGE analysis on our dataset. DGE analysis will be based on the following

- a. Find differentially expressed genes in non-tumour and tumour conditions
- b. Finding differential genes at tumour grades: G1-G2, G2-G3, G3-G4. We want to find genes whose differences are statistically significant at each tumour stage. Because our dataset has cancers from different glioma cell types, we will focus on the astrocytes which include Pilocytic astrocytoma (G1), Low-grade astrocytoma (G2), anaplastic astrocytoma (G3) and Glioblastoma multiforme (G4). This will help us determine prognostic biomarkers.
- c. Finding differential genes at each cancer cell type of glioma cancer: Astrocytes, ependymomas, oligodendrogiomas/oligodendrocytes and mixed gliomas (oligoastrocytes and astrocytes). We want to determine genes that differentiate each cell-type glioma cancers.

## Loading libraries and datasets

```
suppressMessages(library(tidyverse))

## Warning: package 'ggplot2' was built under R version 4.3.3

suppressMessages(library(limma))
suppressMessages(library(tidyheatmaps))

## Warning: package 'tidyheatmaps' was built under R version 4.3.3

library(enrichR)

## Warning: package 'enrichR' was built under R version 4.3.3

## Welcome to enrichR
## Checking connection ...
```

```

## Enrichr ... Connection is Live!
## FlyEnrichr ... Connection is Live!
## WormEnrichr ... Connection is Live!
## YeastEnrichr ... Connection is Live!
## FishEnrichr ... Connection is Live!
## OxEnrichr ... Connection is Live!

suppressMessages(library(clusterProfiler))

## Warning: package 'clusterProfiler' was built under R version 4.3.3

expr_data <- read.csv('glioma_cancer_exprs.csv', row.names = 1)
metadata <- read.csv('glioma_cancer_metadata.csv')

head(expr_data[, 1:5])

##      GSM3242216 GSM3242217 GSM3242218 GSM3242219 GSM3242220
## A1BG     5.515270   5.395054   5.456711   5.418592   5.632935
## A1CF     5.424709   5.297825   5.125438   5.295408   5.254164
## A2M      10.855696  10.855696  12.175815  12.665022  11.359839
## A2ML1    5.239263   5.016883   4.994302   5.035702   5.271587
## A4GALT   5.007046   4.822156   6.029816   5.481442   5.154528
## AAAS     6.482492   6.800803   7.206842   7.392528   7.439305

head(metadata)

##   sample_id tumor_class tumor_grade   gse_id tumor_type
## 1 GSM3242216 glioblastoma          G4 GSE116520 malignant
## 2 GSM3242217 glioblastoma          G4 GSE116520 malignant
## 3 GSM3242218 glioblastoma          G4 GSE116520 malignant
## 4 GSM3242219 glioblastoma          G4 GSE116520 malignant
## 5 GSM3242220 glioblastoma          G4 GSE116520 malignant
## 6 GSM3242221 glioblastoma          G4 GSE116520 malignant

# Adding new features to our metadata
metadata <- metadata |>
  mutate(across(~sample_id, ~(x) factor(x)),
         is_tumour = factor(ifelse(tumor_grade == 'Normal', 'Normal', 'Tumor'),
                           labels = c('Normal', 'Tumor')),
         cell_type = str_replace(tumor_class, 'glioblastoma|astrocytoma|oligoastrocytoma', 'astrocyte'),
         cell_type = str_replace(cell_type, 'olig.+glioma$', 'oligodendrocyte') |> as.factor())

table(metadata$is_tumour)

##
## Normal  Tumor
##      95    726

```

```

table(metadata$cell_type)

##
##          astrocyte      ependymoma high grade glioma      mixed gliomas
##          498                  118                 32                  15
##          normal    oligodendrocyte
##          95                  63

plot_volcano <- function(model.fit, lfc_cutoff=2, p.value=0.05,
                           coef.pos=NULL, title='Volcano plot'){
  res <- topTable(model.fit, n=Inf, coef = coef.pos)
  res <- res |>
    mutate(status = ifelse(logFC > lfc_cutoff & adj.P.Val < p.value, 'Up',
                           ifelse(logFC < -lfc_cutoff & adj.P.Val < p.value, 'Down',
                                  'NS')) |> factor(levels=c('NS','Down','Up'))) |>
    mutate(genes = rownames(res))

  fig <- res |>
    ggplot(aes(logFC, -log10(adj.P.Val), color=status)) +
    geom_point(alpha=0.7) +
    geom_vline(xintercept = c(-lfc_cutoff, lfc_cutoff), linetype='dashed') +
    geom_hline(yintercept = -log10(p.value), linetype='dashed') +
    theme_minimal() +
    theme(legend.position = 'top',
          legend.key = element_blank(),
          panel.grid = element_blank(),
          legend.title = element_text(face='bold', size=10),
          plot.title = element_text(face='bold')) +
    labs(title=title, x='Log2 Fold change', y='Log 10 pvalue',
         color='Regulation') +
    scale_color_manual(values=c('dimgray', 'forestgreen', 'firebrick'))

  top_up <- res |> filter(status == 'Up') |> slice_min(adj.P.Val, n=5)
  top_down <- res |> filter(status == 'Down') |> slice_min(adj.P.Val, n=5)

  fig +
    ggrepel::geom_text_repel(data = top_up, mapping=aes(logFC, -log10(adj.P.Val),
                                                       label=genes),
                             size=3, color='black', fontface='bold',
                             arrow = arrow(length = unit(0.02, "npc")), box.padding = 1) +
    ggrepel::geom_text_repel(data=top_down, mapping=aes(logFC, -log10(adj.P.Val),
                                                       label=genes),
                             size=3, color='black', fontface='bold',
                             arrow = arrow(length = unit(0.02, "npc")), box.padding = 1)
}

}

```

## Differential Gene Expression Analysis

- a. Non-tumour vs Tumour

```

design <- model.matrix(~0 + metadata$is_tumour)
colnames(design) <- c('Normal', 'Tumour')

# instantiate a contrast matrix
const.matrix <- makeContrasts(NormalvsCancer = Normal-Tumour,
                               levels = colnames(design))

model_fit <- lmFit(expr_data, design)
model_fit <- contrasts.fit(model_fit, const.matrix)

ebfit <- eBayes(model_fit, robust=T)

```

**ne**

```

# number of differential genes at |lfc| > 2
summary(decideTests(ebfit, lfc=2))

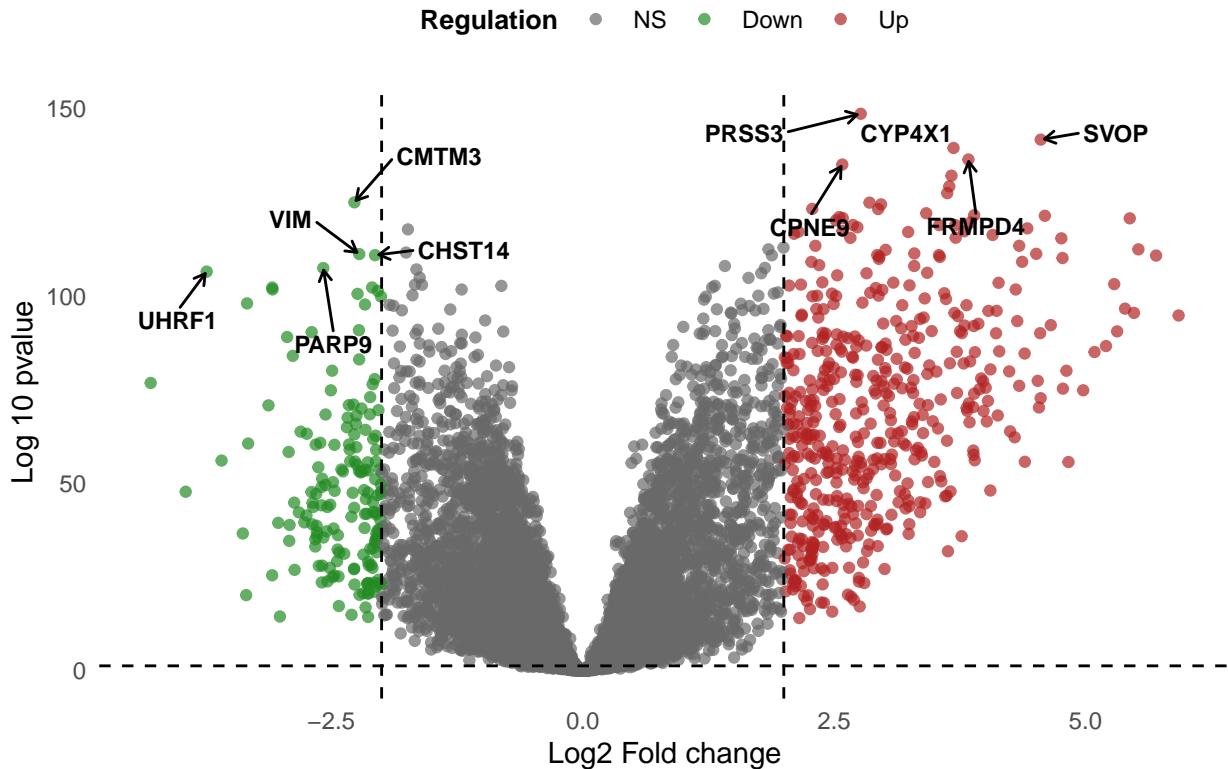
##          NormalvsCancer
## Down           166
## NotSig        12048
## Up            485

# differentially expressed genes by LFC of 2
normal_vs_tumour_DED <- topTable(ebfit, sort.by = 'P', n=Inf, lfc=2, p.value=0.05)

plot_volcano(ebfit, title='Normal vs Tumour tissues')

```

## Normal vs Tumour tissues



b. By cell type

What genes are significantly expressed as you move from one grade to another? We will use the astrocyte cell type and compare tumour progression

```
astrocytes <- which(metadata$cell_type == 'astrocyte')

design <- model.matrix(~ 0 + factor(metadata[astrocytes, 'tumor_grade']))
colnames(design) <- str_extract(colnames(design), 'G[0-9]$')

# instantiate a contrast matrix
const.matrix <- makeContrasts(G1vsG2 = G1-G2,
                                G2vsG3 = G2-G3,
                                G3vsG4 = G3-G4,
                                levels = design)

astrocyte_df <- expr_data[, astrocytes]

model_fit <- lmFit(astrocyte_df, design)
model_fit <- contrasts.fit(model_fit, const.matrix)

ebfit <- eBayes(model_fit, robust=T)
```

ne

```
# number of differential genes at |lfc| > 2
summary(decideTests(ebfit, lfc=1.5))

##          G1vsG2  G2vsG3  G3vsG4
## Down      249     10     12
## NotSig   12076   12679  12683
## Up       374     10      4

# differentially expressed genes by LFC of 2

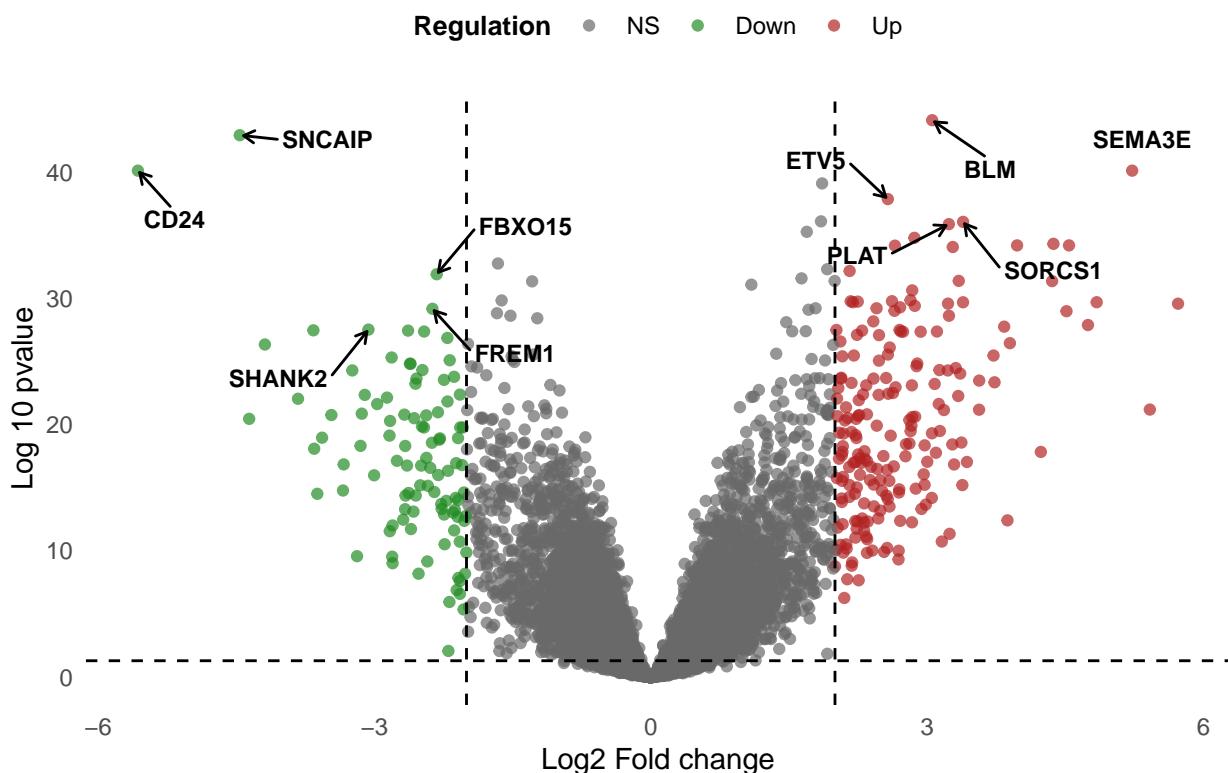
G1_vs_G2 <- topTable(ebfit, sort.by = 'P', n=Inf, lfc=1.5,
                      p.value=0.05, coef='G1vsG2')

G2_vs_G3 <- topTable(ebfit, sort.by = 'P', n=Inf, lfc=1.5,
                      p.value=0.05, coef='G2vsG3')

G3_vs_G4 <- topTable(ebfit, sort.by = 'P', n=Inf, lfc=1.5,
                      p.value=0.05, coef='G3vsG4')

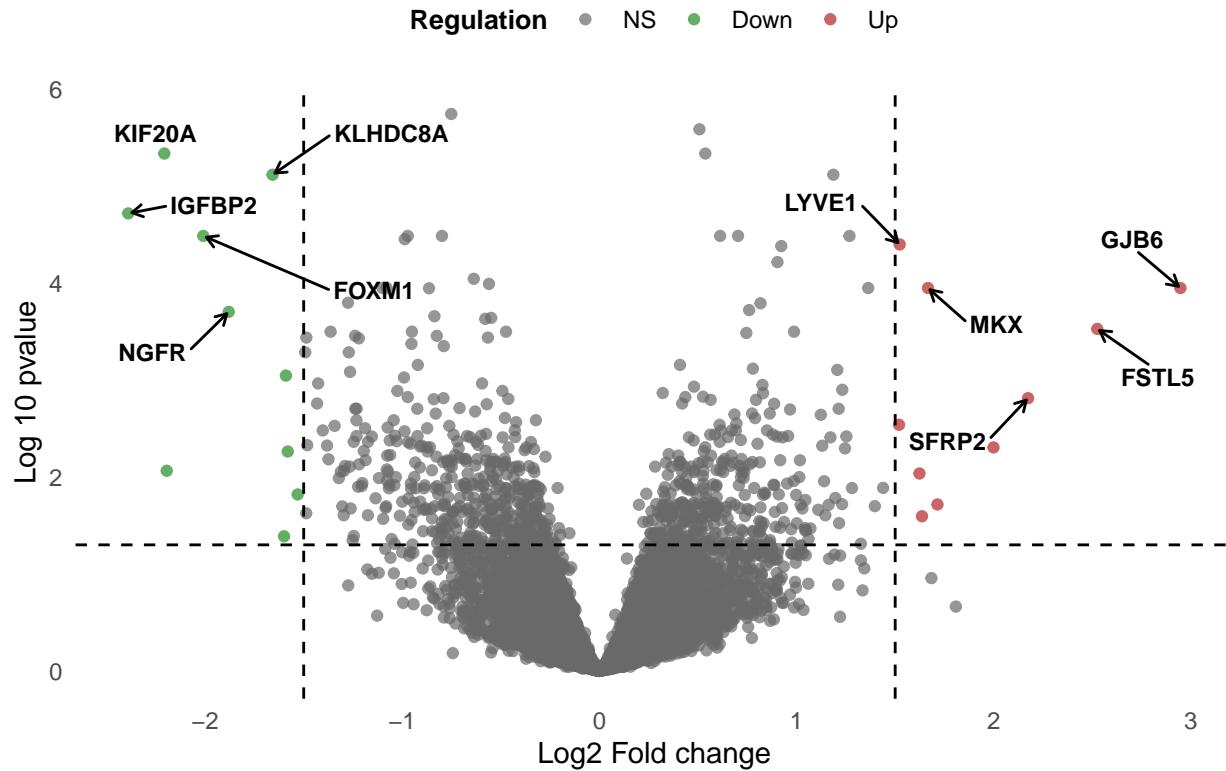
plot_volcano(ebfit, title='Differential Genes in Grade1 vs Grade2', coef='G1vsG2')
```

## Differential Genes in Grade1 vs Grade2



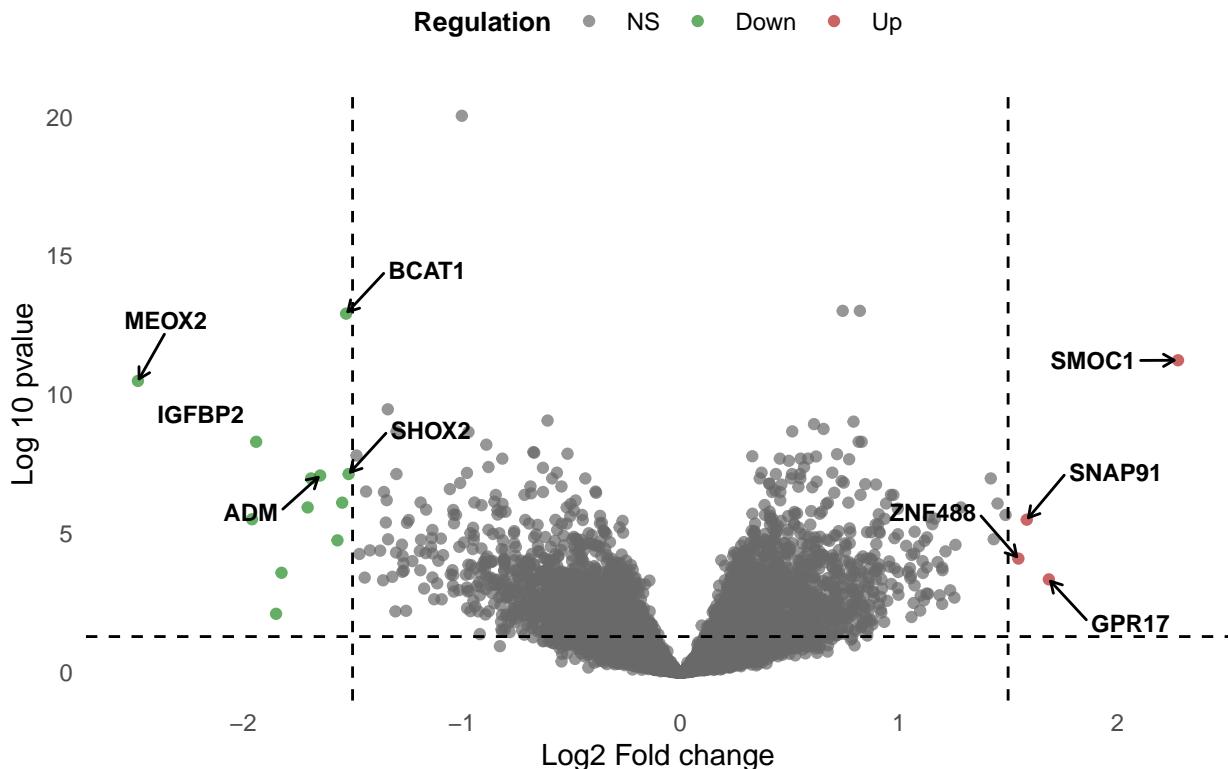
```
plot_volcano(ebfit, title='Differential Genes in Grade2 vs Grade3',
             coef='G2vsG3', lfc_cutoff = 1.5)
```

### Differential Genes in Grade2 vs Grade3



```
plot_volcano(ebfit, title='Differential Genes in Grade3 vs Grade4',
             coef='G3vsG4', lfc_cutoff = 1.5)
```

## Differential Genes in Grade3 vs Grade4



```
# getting differential gene in all grades

sig.genes <- list(G1_vs_G2, G2_vs_G3, G3_vs_G4) |>
  map(.f = function(x) filter(x, adj.P.Val < 0.05, abs(logFC) > 1.5) |>
    rownames_to_column(var='gene') |>
    pull(gene))

genes <- rownames(expr_data)
similar.genes = genes
for (i in 1:length(sig.genes)) similar.genes <- intersect(similar.genes, sig.genes[[i]])

print(similar.genes)

## [1] "CHI3L1"
```

The CHI3L1 gene is found in all cancer grades for astrocytoma. This could mean that it could act as a prognostic biomarker to detect cancer progression for astrocytoma.

c. Cell types

```
cell_types <- which(!metadata$cell_type %in% c('normal', 'high grade glioma'))
```

```

design <- model.matrix(~ 0 + factor(metadata[cell_types, 'cell_type']))
colnames(design) <- c('AC', 'EC', 'MGC', 'OGC')

# AC -> Astrocytoma, EC -> Ependymoma, MG -> mixed glioma, OGC -> oligodendrocytoma/oligodendrogloma

# instantiate a contrast matrix
const.matrix <- makeContrasts(ACvsEC = AC-EC,
                               ACvsMGC = AC-MGC,
                               ACvsOGC = AC-OGC,
                               ECvsMGC = EC-MGC,
                               ECvsOGC = EC-OGC,
                               MGCvsOGC = MGC-OGC,
                               levels = design)

model_fit <- lmFit(expr_data[, cell_types], design)
model_fit <- contrasts.fit(model_fit, const.matrix)

ebfit <- eBayes(model_fit, robust=F)

```

## ne

```

# number of differential genes at |lfc| > 2
summary(decideTests(ebfit, lfc=1.5))

##          ACvsEC  ACvsMGC  ACvsOGC  ECvsMGC  ECvsOGC  MGCvsOGC
## Down      391      34      37      82      214       0
## NotSig   11964    12594   12594   12430   12236   12699
## Up       344       71      68     187      249       0

```

- No significant genes between mixed gliomas and oligodendroglomas

```

# differentially expressed genes by LFC of 2

AC_vs_EC <- topTable(ebfit, sort.by = 'P', n=Inf, lfc=1.5,
                      p.value=0.05, coef='ACvsEC')

AC_vs_MGC <- topTable(ebfit, sort.by = 'P', n=Inf, lfc=1.5,
                       p.value=0.05, coef='ACvsMGC')

AC_vs_OGC <- topTable(ebfit, sort.by = 'P', n=Inf, lfc=1.5,
                       p.value=0.05, coef='ACvsOGC')

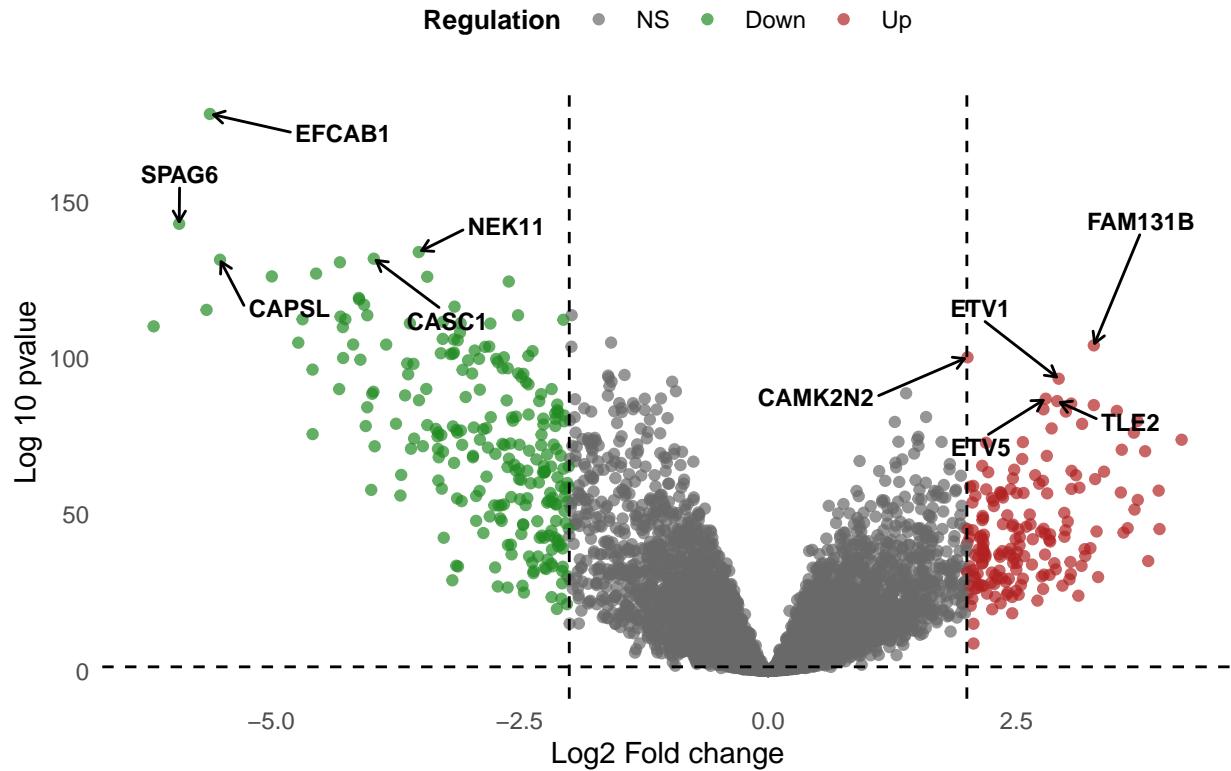
EC_vs_MGC <- topTable(ebfit, sort.by = 'P', n=Inf, lfc=1.5,
                       p.value=0.05, coef='ECvsMGC')

EC_vs_OGC <- topTable(ebfit, sort.by = 'P', n=Inf, lfc=1.5,
                       p.value=0.05, coef='ECvsOGC')

```

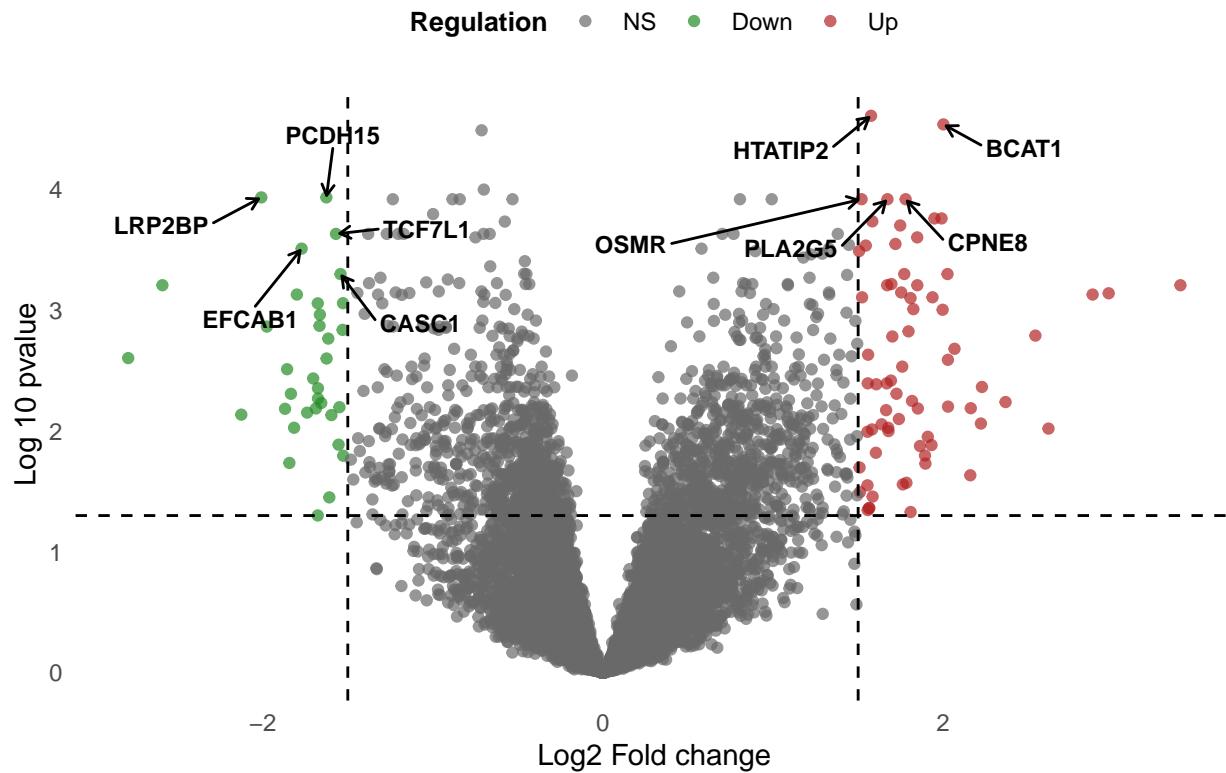
```
plot_volcano(ebfit, title='Differential Genes in Astrocytoma vs Ependymoma',
             coef=1, lfc_cutoff = 2)
```

## Differential Genes in Astrocytoma vs Ependymoma



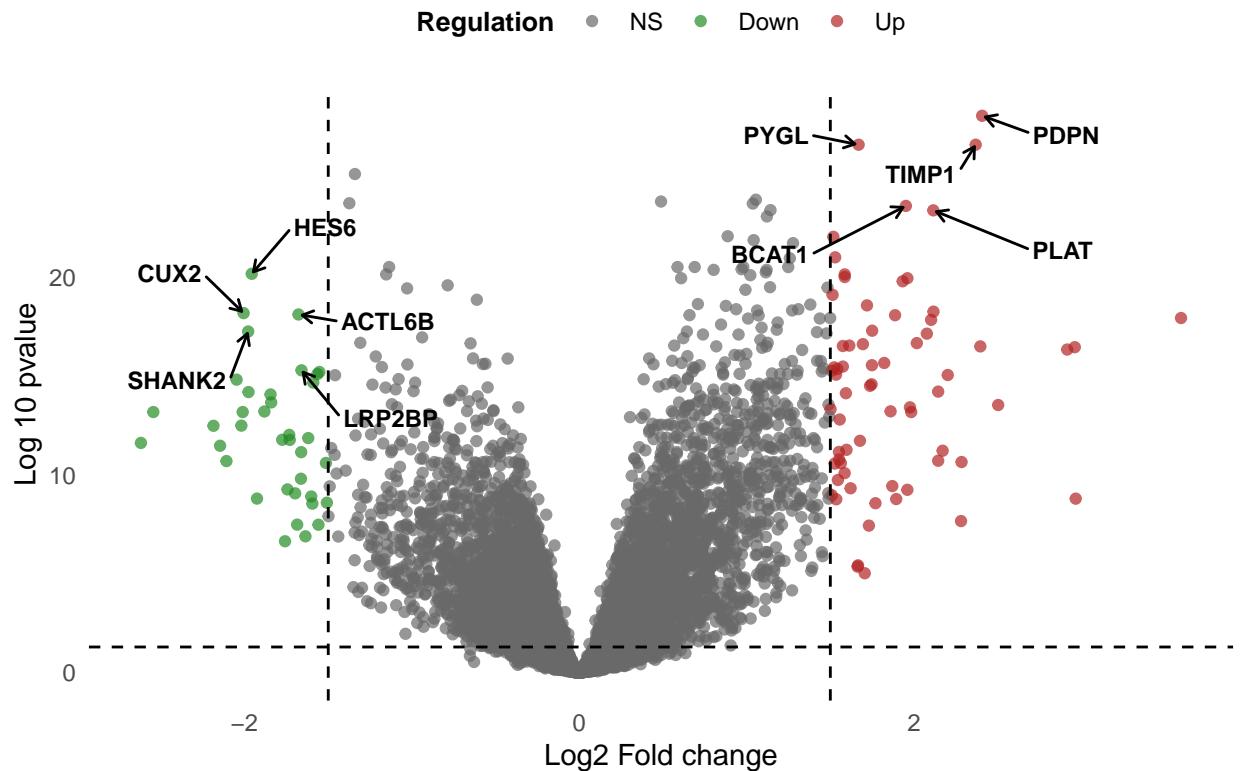
```
plot_volcano(ebfit, title='Differential Genes in Astrocytoma vs Mixed Gliomas',
             coef=2, lfc_cutoff = 1.5)
```

## Differential Genes in Astrocytoma vs Mixed Gliomas



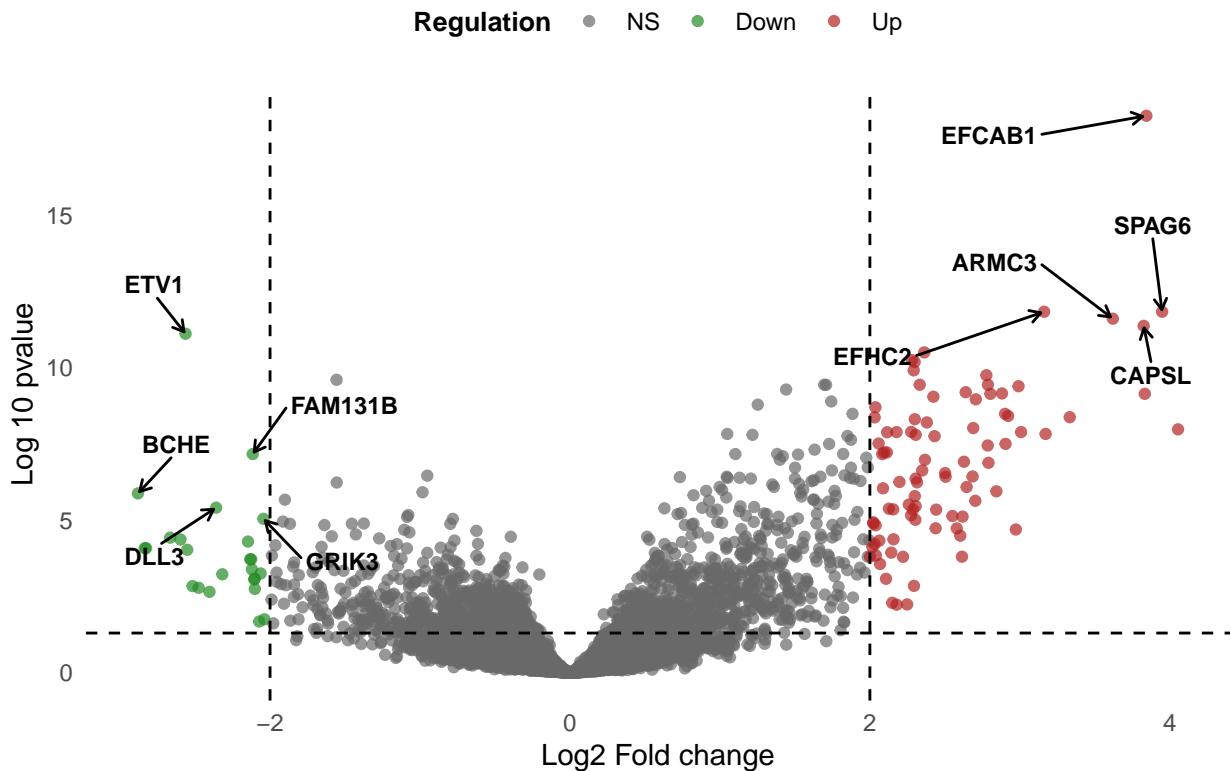
```
plot_volcano(ebfit, title='Differential Genes in Astrocytoma vs Oligodendrogloma',  
            coef=3, lfc_cutoff = 1.5)
```

## Differential Genes in Astrocytoma vs Oligodendrogloma



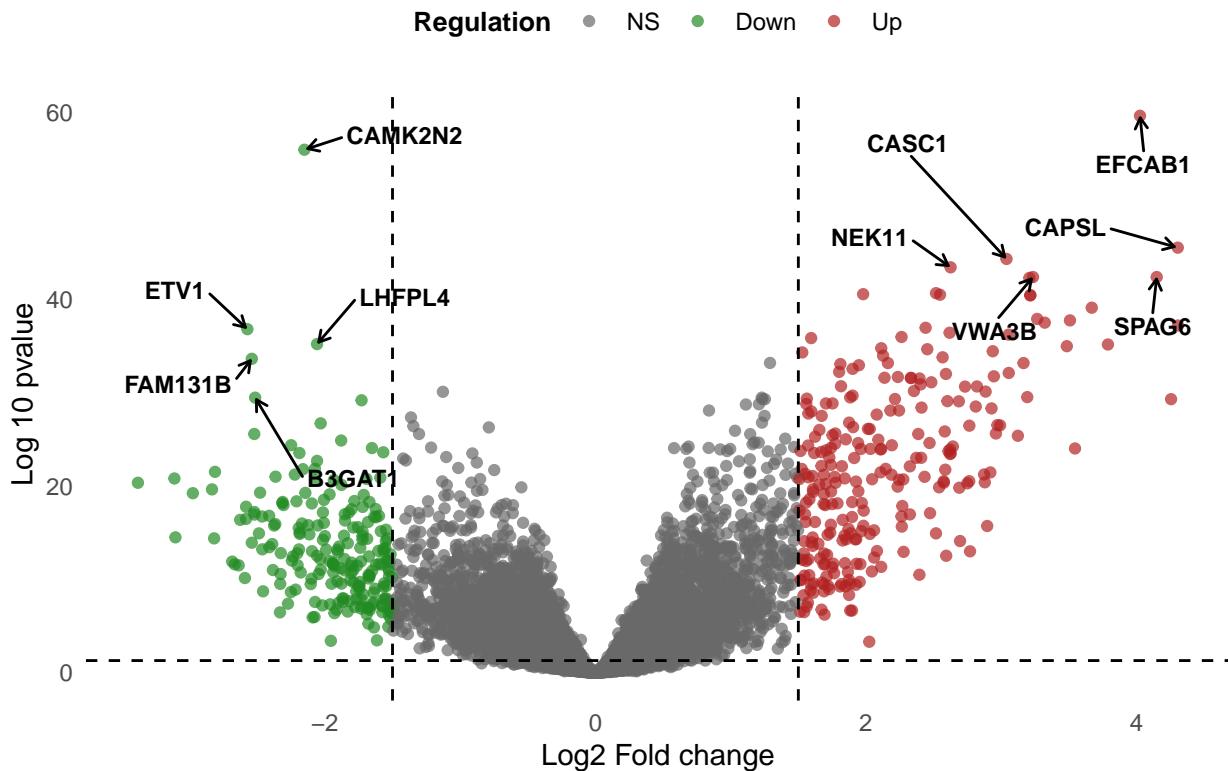
```
plot_volcano(ebfit, title='Differential Genes in Ependymoma vs Mixed glioma',  
            coef=4, lfc_cutoff = 2)
```

## Differential Genes in Ependymoma vs Mixed glioma



```
plot_volcano(ebfit, title='Differential Genes in Ependymoma vs Oligodendrogloma',
             coef=5, lfc_cutoff = 1.5)
```

## Differential Genes in Ependymoma vs Oligodendrogloma



```
# getting differential gene in all grades

sig.genes <- list(AC_vs_EC, AC_vs_MGC, AC_vs_OGC,
                  EC_vs_OGC, EC_vs_MGC) |>
  map(.f = function(x) filter(x, adj.P.Val < 0.05, abs(logFC) > 1.5) |>
    rownames_to_column(var='gene') |>
    pull(gene))

similar.genes = genes
for (i in 1:length(sig.genes)) similar.genes <- intersect(similar.genes, sig.genes[[i]])

print(similar.genes)

## [1] "C9orf24"  "CITED1"   "CNR1"     "CRLF1"    "DACH1"    "EFCAB1"   "GABRG1"
## [8] "LRP2BP"   "MKX"      "PPP1R1B"   "RELN"     "RSPH1"    "SPAG6"    "SRPX"
## [15] "VIPR2"
```

## Functional Enrichment Analysis

For functional analysis we will perform Gene Ontology (GO) analysis and the Pathway analysis. The GO analysis will contain the biological processes of these differential genes, molecular function of their gene product and cell compartment localisation. For the pathway analysis, we will perform the KEGG and reactome pathways.

```

# databases
dbs <- enrichR::listEnrichrDbs()

# head(dbs)

# dbs$libraryName[grep('(^GO/KEGG/React/Wiki)', dbs$libraryName)]

db_GO <- c("GO_Biological_Process_2023", "GO_Cellular_Component_2023", "GO_Molecular_Function_2023")
db_pw <- c("Reactome_2022", "KEGG_2021_Human")

get_FEA <- function(genes.list, selected_database){
  res <- enrichR::enrichr(genes.list, selected_database)
  res <- res |>
    map(function(x) x |>
      dplyr::filter(Adjusted.P.value < 0.05) |>
      mutate(Term=str_remove_all(Term, '\\\\(GO:.+\\\\)$|R-HSA-\\\\d+$|'))
    )
  
  return(res)
}

# GO for normal vs tumour samples
GO_results_nt_up <- get_FEA(row.names(
  normal_vs_tumour_DED |>
    filter(logFC > 2)), selected_database = db_GO)

## Uploading data to Enrichr... Done.
##   Querying GO_Biological_Process_2023... Done.
##   Querying GO_Cellular_Component_2023... Done.
##   Querying GO_Molecular_Function_2023... Done.
## Parsing results... Done.

# GO for normal vs tumour samples
GO_results_nt_down <- get_FEA(row.names(
  normal_vs_tumour_DED |>
    filter(logFC < -2)), selected_database = db_GO)

## Uploading data to Enrichr... Done.
##   Querying GO_Biological_Process_2023... Done.
##   Querying GO_Cellular_Component_2023... Done.
##   Querying GO_Molecular_Function_2023... Done.
## Parsing results... Done.

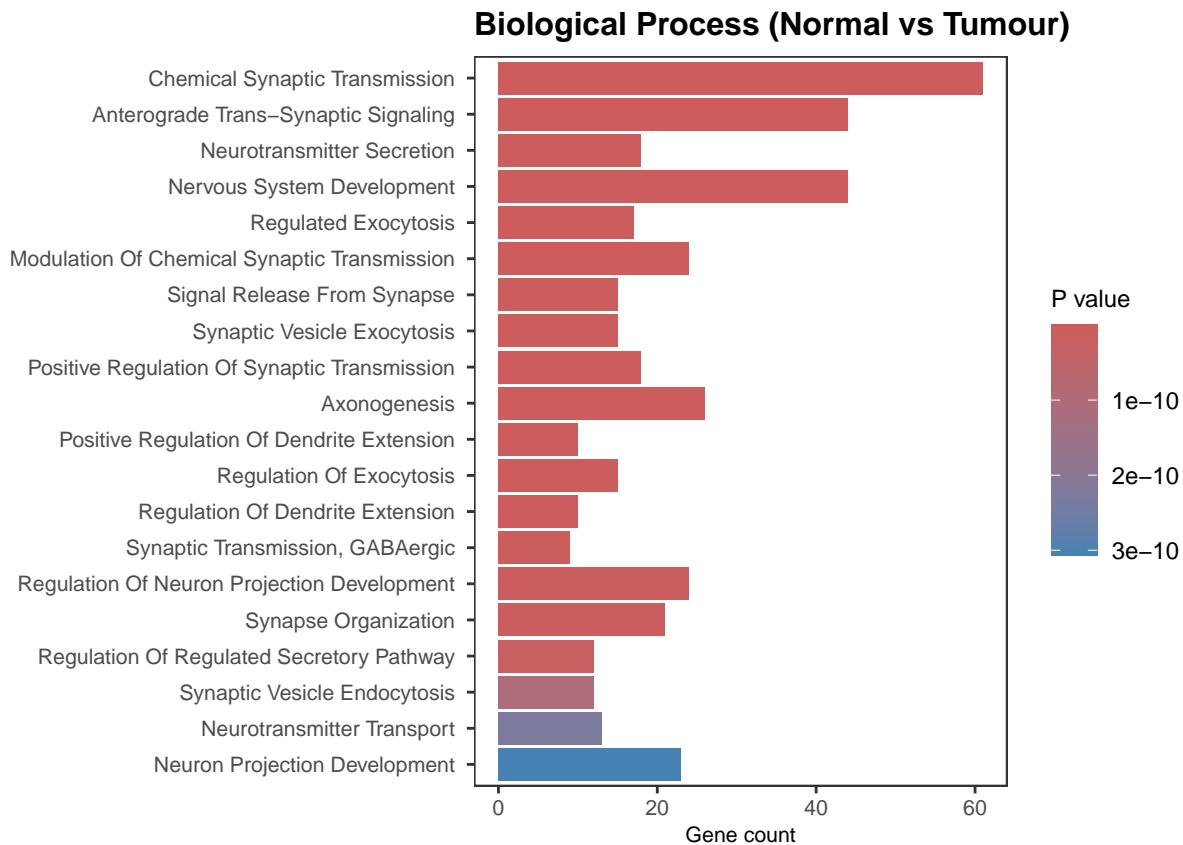
plotEnrich(GO_results_nt_up$GO_Biological_Process_2023, numChar = 70) +
  theme_bw() +
  theme(panel.grid = element_blank(),
        axis.text = element_text(size=8),
        axis.title.x = element_text(size=8),
        legend.title = element_text(size=9),
        plot.title=element_text(face='bold', size=12)) +
  scale_fill_gradient(low='indianred', high='steelblue',
                      guide = guide_colorbar(reverse = T)) +
  labs(title='Biological Process (Normal vs Tumour)', x='')


```

```

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

```



```

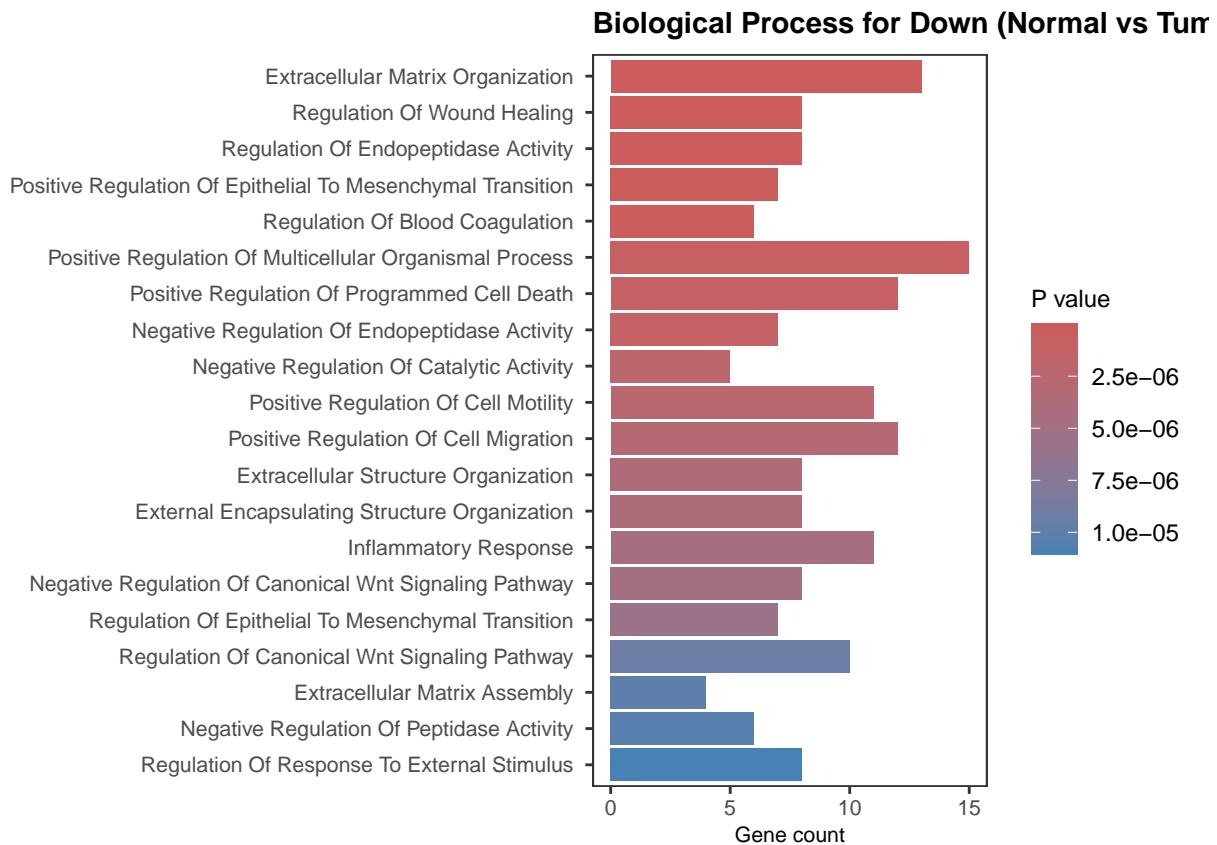
plotEnrich(GO_results_nt_down$GO_Biological_Process_2023, numChar = 70) +
  theme_bw() +
  theme(panel.grid = element_blank(),
        axis.text = element_text(size=8),
        axis.title.x = element_text(size=8),
        legend.title = element_text(size=9),
        plot.title=element_text(face='bold', size=11)) +
  scale_fill_gradient(low='indianred', high='steelblue',
                      guide = guide_colorbar(reverse = T)) +
  labs(title='Biological Process for Down (Normal vs Tumour)', x='')

```

```

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

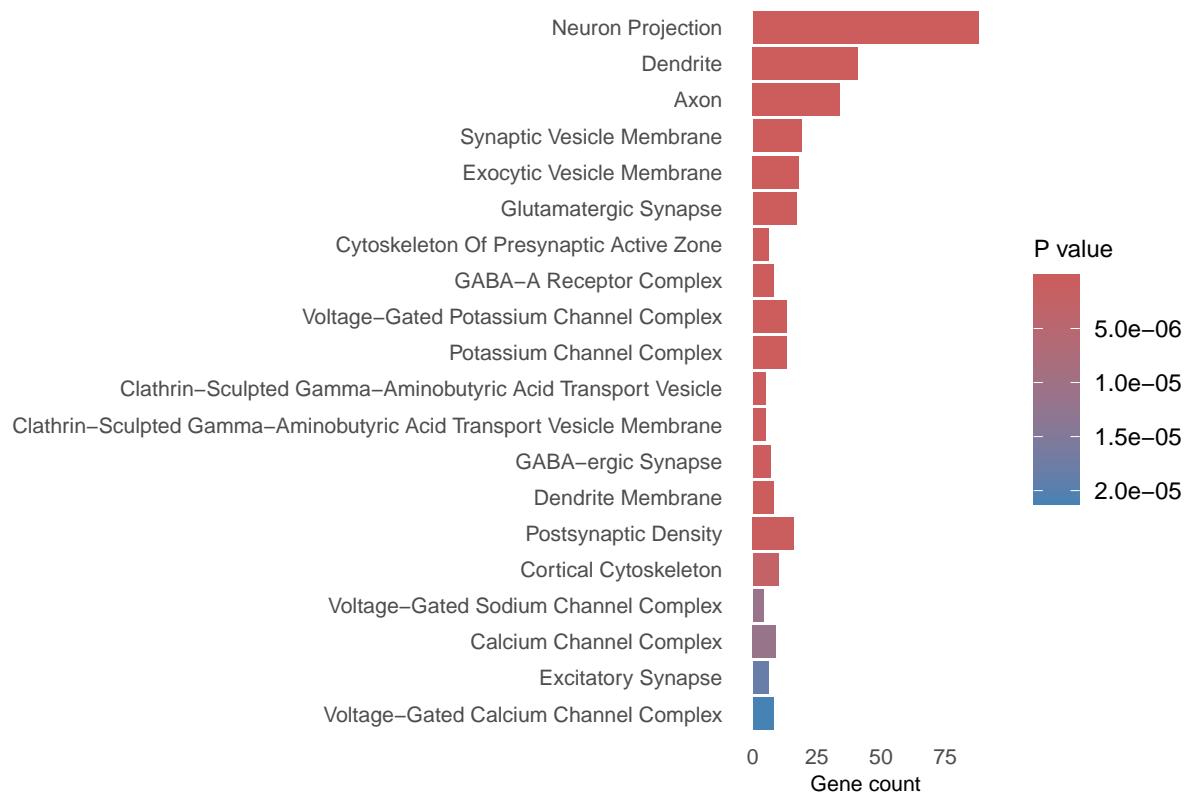
```



```
plotEnrich(GO_results_nt_up$GO_Cellular_Component_2023, numChar = 70) +
  theme_minimal() +
  theme(panel.grid=element_blank(),
        axis.text = element_text(size=8),
        axis.title.x = element_text(size=8),
        legend.title = element_text(size=9),
        plot.title=element_text(face='bold', size=12)) +
  scale_fill_gradient(low='indianred', high='steelblue',
                      guide = guide_colorbar(reverse = T)) +
  labs(title='Cellular Component (Normal vs Tumour)', x='')
```

```
## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.
```

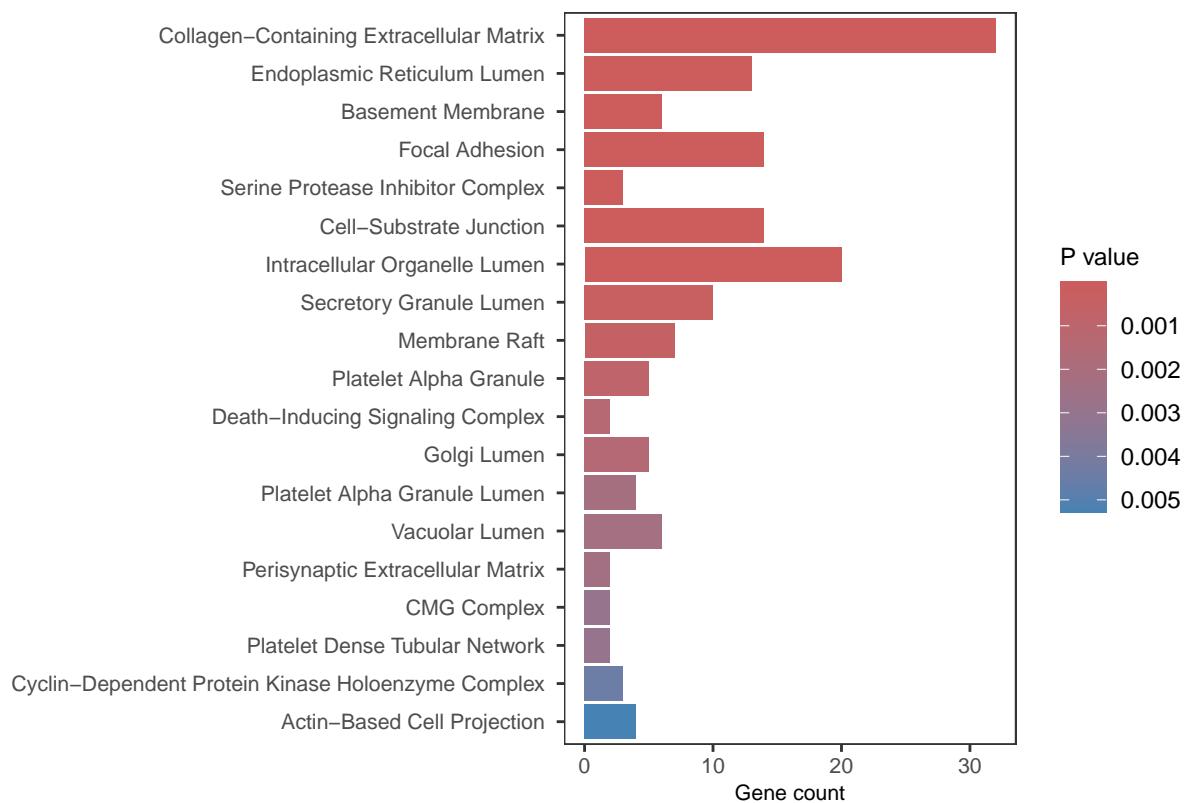
## Cellular Component (Normal vs Tumour)



```
plotEnrich(GO_results_nt_down$GO_Cellular_Component_2023, numChar = 70) +
  theme_bw() +
  theme(panel.grid = element_blank(),
        axis.text = element_text(size=8),
        axis.title.x = element_text(size=8),
        legend.title = element_text(size=9),
        plot.title=element_text(face='bold', size=11)) +
  scale_fill_gradient(low='indianred', high='steelblue',
                      guide = guide_colorbar(reverse = T)) +
  labs(title='Cellular Component for Down (Normal vs Tumour)', x='')
```

```
## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.
```

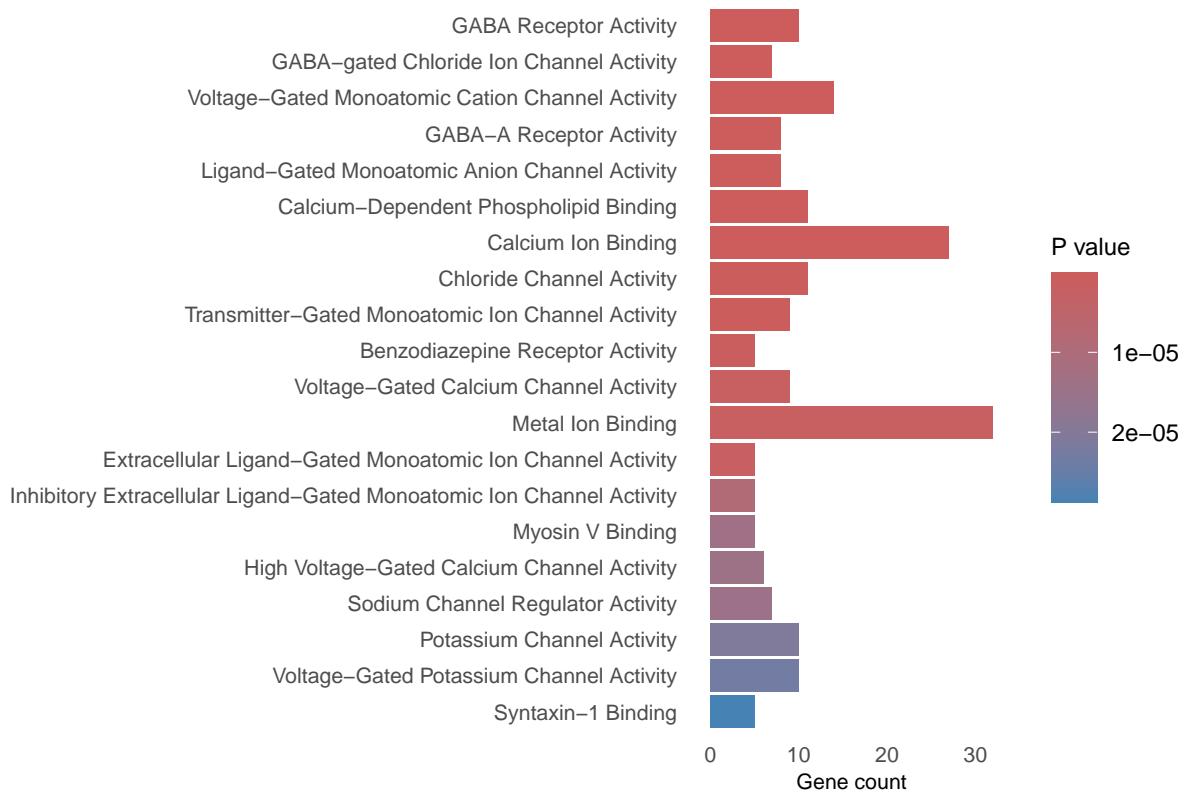
### Cellular Component for Down (Normal vs Tumour)



```
plotEnrich(GO_results_nt_up$GO_Molecular_Function_2023, numChar = 70) +
  theme_minimal() +
  theme(panel.grid=element_blank(),
        axis.text = element_text(size=8),
        axis.title.x = element_text(size=8),
        legend.title = element_text(size=9),
        plot.title=element_text(face='bold', size=12)) +
  scale_fill_gradient(low='indianred', high='steelblue',
                      guide = guide_colorbar(reverse = T)) +
  labs(title='Molecular Function (Normal vs Tumour)', x='')
```

```
## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.
```

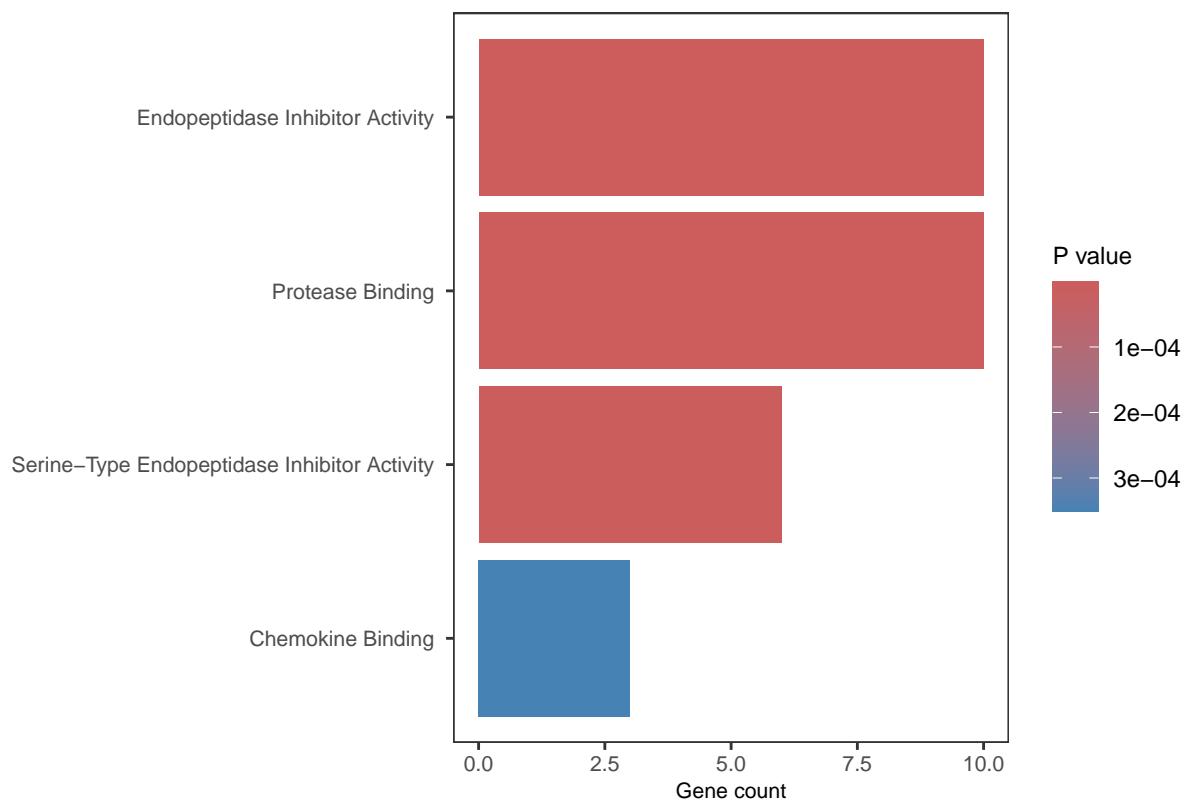
## Molecular Function (Normal vs Tu



```
plotEnrich(GO_results_nt_down$GO_Molecular_Function_2023,
           numChar = 70, showTerms = 4) +
  theme_bw() +
  theme(panel.grid=element_blank(),
        axis.text = element_text(size=8),
        axis.title.x = element_text(size=8),
        legend.title = element_text(size=9),
        plot.title=element_text(face='bold', size=12)) +
  scale_fill_gradient(low='indianred', high='steelblue',
                      guide = guide_colorbar(reverse = T)) +
  labs(title='Molecular Function Down (Normal vs Tumour)', x='')
```

```
## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.
```

### Molecular Function Down (Normal vs Tumour)



```
# Pathway for normal vs tumour samples
PW_results_nt_up <- get_FEA(row.names(
  normal_vs_tumour_DED |>
    filter(logFC > 2)), selected_database = db_pw)
```

```
## Uploading data to Enrichr... Done.
##   Querying Reactome_2022... Done.
##   Querying KEGG_2021_Human... Done.
## Parsing results... Done.
```

```
PW_results_nt_down <- get_FEA(row.names(
  normal_vs_tumour_DED |>
    filter(logFC < -2)), selected_database = db_pw)
```

```
## Uploading data to Enrichr... Done.
##   Querying Reactome_2022... Done.
##   Querying KEGG_2021_Human... Done.
## Parsing results... Done.
```

```
plotEnrich(PW_results_nt_up$KEGG_2021_Human,
           numChar = 70) +
  theme_bw() +
  theme(panel.grid=element_blank(),
        axis.text = element_text(size=8),
        axis.title.x = element_text(size=8),
```

```

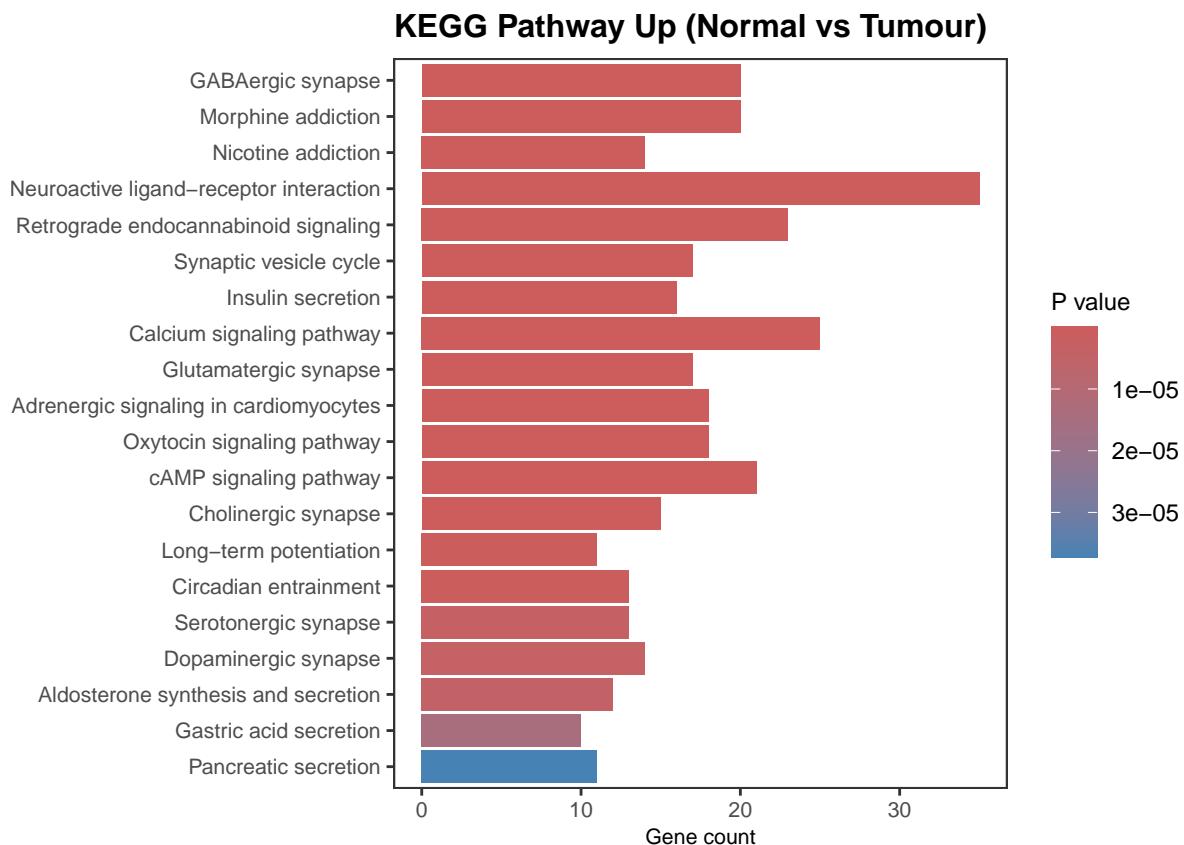
    legend.title = element_text(size=9),
    plot.title=element_text(face='bold', size=12)) +
  scale_fill_gradient(low='indianred', high='steelblue',
                      guide = guide_colorbar(reverse = T)) +
  labs(title='KEGG Pathway Up (Normal vs Tumour)', x=' ')

```

```

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

```



```

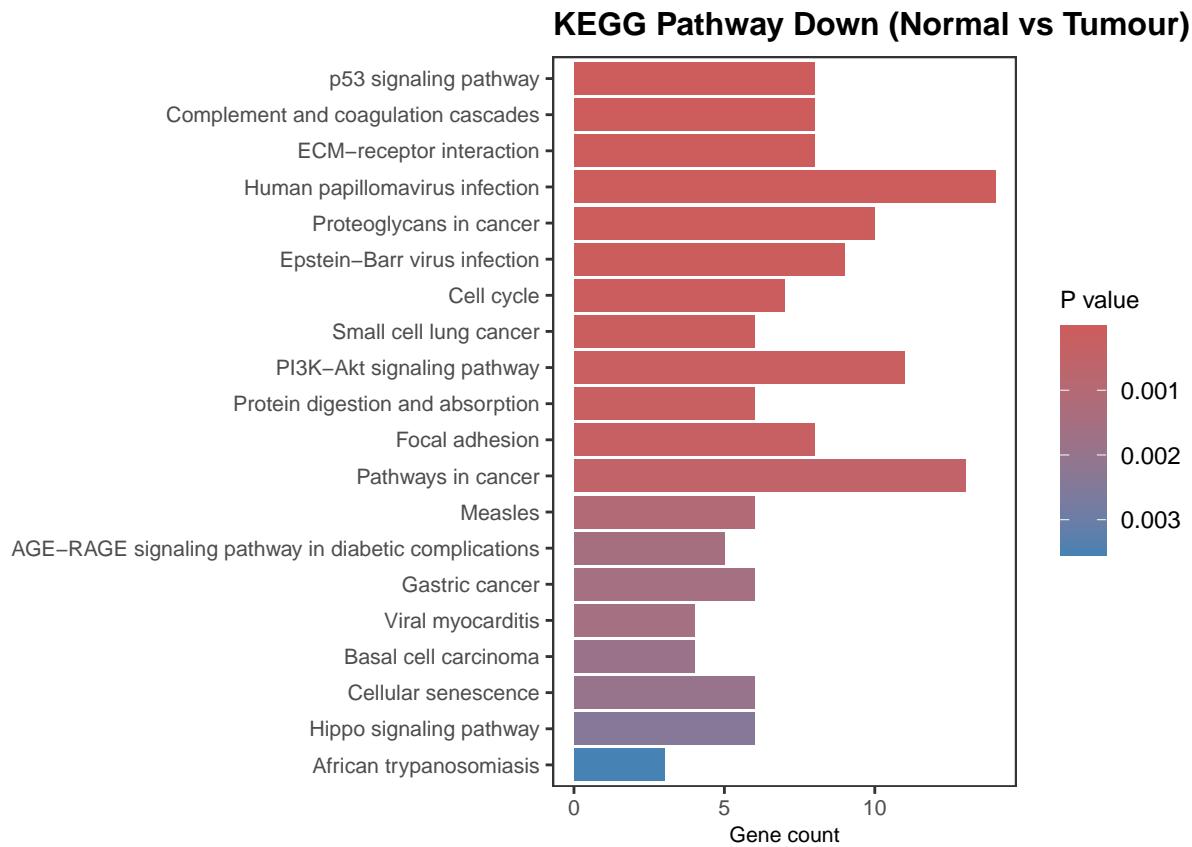
plotEnrich(PW_results_nt_down$KEGG_2021_Human,
            numChar = 70) +
  theme_bw() +
  theme(panel.grid=element_blank(),
        axis.text = element_text(size=8),
        axis.title.x = element_text(size=8),
        legend.title = element_text(size=9),
        plot.title=element_text(face='bold', size=12)) +
  scale_fill_gradient(low='indianred', high='steelblue',
                      guide = guide_colorbar(reverse = T)) +
  labs(title='KEGG Pathway Down (Normal vs Tumour)', x=' ')

```

```

## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.

```



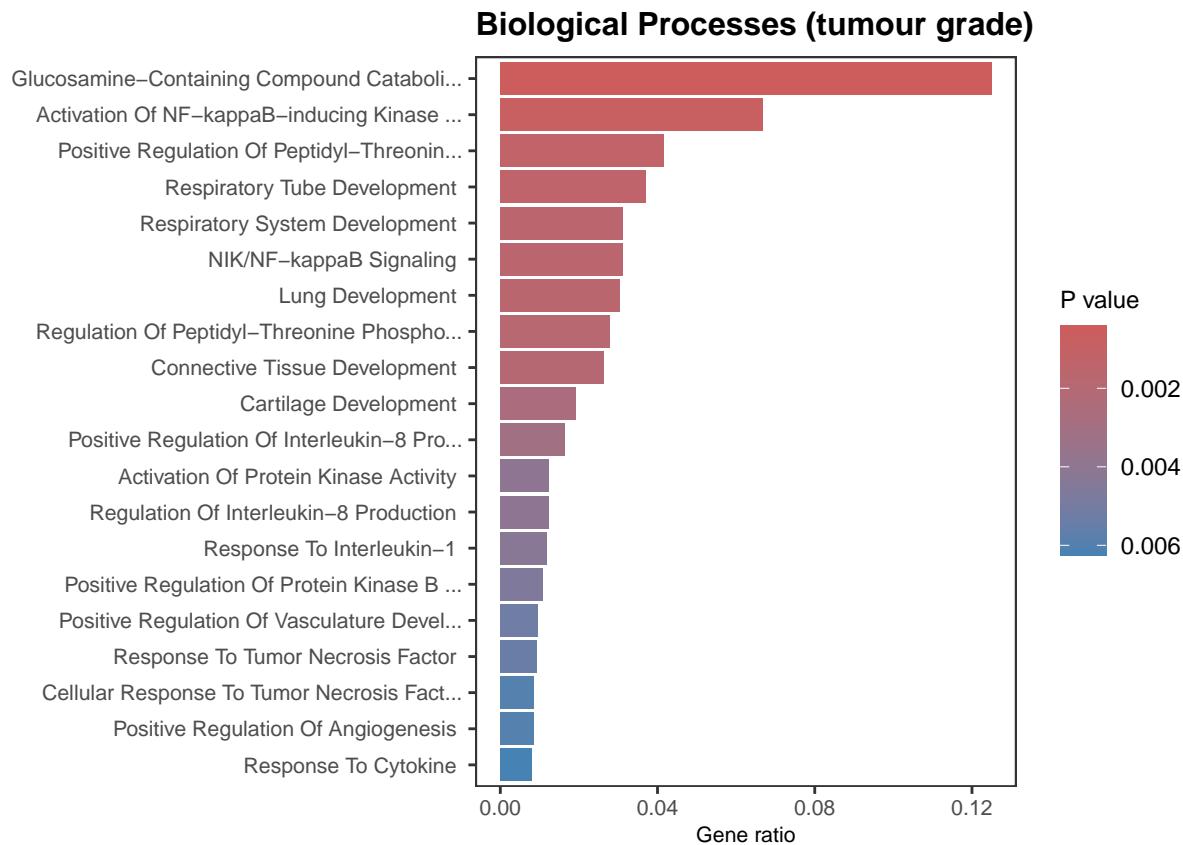
- Similar Genes for Astrocytoma tumour progression and cell-type tumours

```
res <- get_FEA("CHI3L1", db_GO)
```

```
## Uploading data to Enrichr... Done.
##   Querying GO_Biological_Process_2023... Done.
##   Querying GO_Cellular_Component_2023... Done.
##   Querying GO_Molecular_Function_2023... Done.
## Parsing results... Done.
```

```
plotEnrich(res$GO_Biological_Process_2023, y = 'Ratio') +
  theme_bw() +
  theme(panel.grid=element_blank(),
        axis.text = element_text(size=8),
        axis.title.x = element_text(size=8),
        legend.title = element_text(size=9),
        plot.title=element_text(face='bold', size=12)) +
  scale_fill_gradient(low='indianred', high='steelblue',
                      guide = guide_colorbar(reverse = T)) +
  labs(title='Biological Processes (tumour grade)', x='')
```

```
## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.
```



```
res$GO_Molecular_Function_2023
```

```
##                                     Term Overlap      P.value
## 1 Hydrolase Activity, Hydrolyzing O-glycosyl Compounds    1/29 0.00144997
##   Adjusted.P.value Old.P.value Old.Adjusted.P.value Odds.Ratio Combined.Score
## 1          0.00144997           0             0       19971        130534.7
##   Genes
## 1 CHI3L1
```

```
res$GO_Cellular_Component_2023
```

```
##                                     Term Overlap      P.value Adjusted.P.value Old.P.value
## 1 Specific Granule Lumen     1/61 0.003049951    0.009149854      0
## 2 Specific Granule         1/159 0.007949911    0.011924867      0
## 3 Secretory Granule Lumen  1/316 0.015799870    0.015799870      0
##   Old.Adjusted.P.value Odds.Ratio Combined.Score  Genes
## 1                  0     19939     115499.24 CHI3L1
## 2                  0     19841     95923.19 CHI3L1
## 3                  0     19684     81644.38 CHI3L1
```

```
get_FEA("CHI3L1", 'Reactome_2013')
```

```
## Uploading data to Enrichr... Done.
## Querying Reactome_2013... Done.
## Parsing results... Done.
```

```

## $Reactome_2013
## [1] Term          Overlap          P.value
## [4] Adjusted.P.value Old.P.value   Old.Adjusted.P.value
## [7] Odds.Ratio    Combined.Score Genes
## <0 rows> (or 0-length row.names)

similar.genes.celltype <- c("C9orf24", "CITED1", "CNR1", "CRLF1", "DACH1",
                           "EFCAB1", "GABRG1", "LRP2BP", "MKX", "PPP1R1B",
                           "RELN", "RSPH1", "SPAG6", "SRPX", "VIPR2")

# Pathway for normal vs tumour samples
GO_results <- get_FEA(similar.genes.celltype, selected_database = db_pw)

## Uploading data to Enrichr... Done.
## Querying Reactome_2022... Done.
## Querying KEGG_2021_Human... Done.
## Parsing results... Done.

PW_results <- get_FEA(similar.genes.celltype, selected_database = db_pw)

## Uploading data to Enrichr... Done.
## Querying Reactome_2022... Done.
## Querying KEGG_2021_Human... Done.
## Parsing results... Done.

GO_results

## $Reactome_2022
## [1] Term          Overlap          P.value
## [4] Adjusted.P.value Old.P.value   Old.Adjusted.P.value
## [7] Odds.Ratio    Combined.Score Genes
## <0 rows> (or 0-length row.names)
##
## $KEGG_2021_Human
##                               Term Overlap      P.value Adjusted.P.value
## 1 Neuroactive ligand-receptor interaction 3/341 0.001919667 0.03263434
## 2 Retrograde endocannabinoid signaling 2/148 0.005361340 0.04557139
## Old.P.value Old.Adjusted.P.value Odds.Ratio Combined.Score Genes
## 1          0                  0 14.53180 90.90521 VIPR2;CNR1;GABRG1
## 2          0                  0 20.90516 109.30351 CNR1;GABRG1

PW_results

## $Reactome_2022
## [1] Term          Overlap          P.value
## [4] Adjusted.P.value Old.P.value   Old.Adjusted.P.value
## [7] Odds.Ratio    Combined.Score Genes
## <0 rows> (or 0-length row.names)
##
## $KEGG_2021_Human

```

```

##                                     Term Overlap      P.value Adjusted.P.value
## 1 Neuroactive ligand-receptor interaction 3/341 0.001919667    0.03263434
## 2 Retrograde endocannabinoid signaling   2/148 0.005361340    0.04557139
## Old.P.value Old.Adjusted.P.value Odds.Ratio Combined.Score          Genes
## 1           0                 0     14.53180      90.90521 VIPR2;CNR1;GABRG1
## 2           0                 0     20.90516     109.30351 CNR1;GABRG1

```

## Clustering

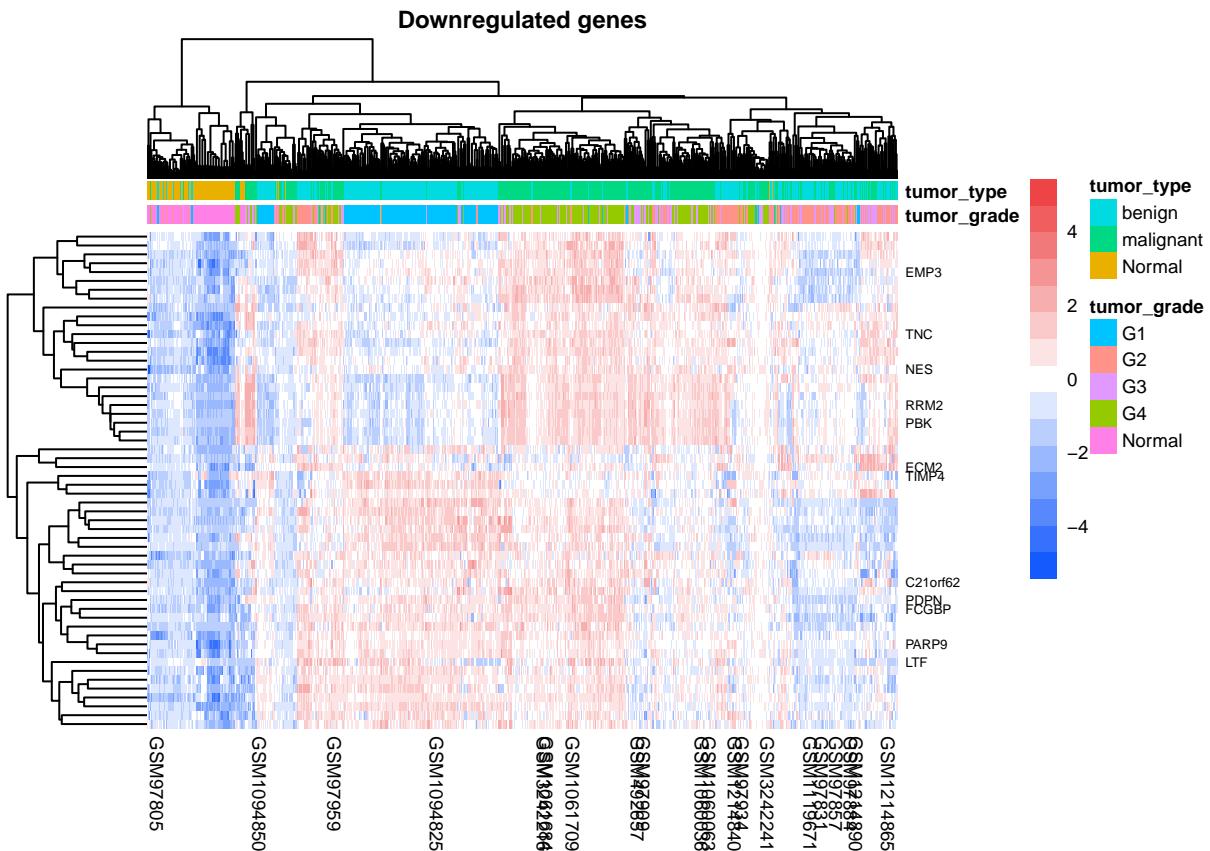
```

t <- expr_data[rownames(normal_vs_tumour_DED) |> filter(logFC < -2.5), ] |>
  rownames_to_column(var='Gene') |>
  pivot_longer(-Gene, names_to = 'samples') |>
  inner_join(metadata, by=c('samples' = 'sample_id')) |>
  arrange(tumor_grade)

select_genes <- (t |> distinct(Gene) |> pull(Gene))

tidyheatmaps::tidy_heatmap(t, rows = 'Gene', width = 8,
                           colors = c("#145afc", "#ffffff", "#ee4445"),
                           height = 20, columns='samples',
                           annotation_col = c('tumor_grade', 'tumor_type'),
                           values='value', show_colnames = T,
                           gaps_row ='tumor_type',
                           show_selected_col_labels = metadata$sample_id[seq(1,500,25)],
                           show_selected_row_labels = select_genes[seq(1, length(select_genes), 5)],
                           fontsize_row = 5, main = 'Downregulated genes',
                           scale='row', cluster_rows = TRUE, cluster_cols = T)

```



```
t <- expr_data[rownames(normal_vs_tumour_DED) |> filter(logFC > 2.), ] |>
  rownames_to_column(var='Gene') |>
  pivot_longer(-Gene, names_to = 'samples') |>
  inner_join(metadata, by=c('samples' = 'sample_id')) |>
  arrange(tumor_grade)
```

```
select_genes <- (t |> distinct(Gene) |> pull(Gene))

tidyheatmaps::tidy_heatmap(t, rows = 'Gene', width = 8,
                           colors = c("#145afc", "#ffffff", "#ee4445"),
                           height = 15, columns='samples',
                           annotation_col = c('tumor_grade', 'tumor_type'),
                           values='value', show_colnames = T,
                           show_selected_col_labels = metadata$sample_id[seq(1,500,50)],
                           show_selected_row_labels = select_genes[seq(1, length(select_genes), 20)],
                           fontsize_row = 5, main = 'Upregulated genes',
                           scale='row', cluster_rows = F, cluster_cols = F)
```

### Upregulated genes

