Heat Map with values

M. Clara Hernández Cañás

02/10/2023

Contents

Heat Map with values

```
library(ggplot2)
library(PupillometryR)
library(ggpubr)
library(ComplexHeatmap)
library(DESeq2)
library(reshape2)
library(gghalves)
library(tidyr)
library(tibble)
library(tidyr)
library(dplyr)
library(biomaRt)

theme_set(theme_bw() +
          theme(axis.text = element_text(size = 10, colour="black"),
                axis.title = element_text(size=12, colour="black"),
                axis.ticks=element_line(color="black"),
                axis.ticks.length=unit(.15, "cm"),
                panel.border=element_rect(color="black", fill = NA),
                panel.background = element_blank(),
                plot.background = element_blank(),
                legend.text = element_text(size=10),
                legend.position = "bottom"))

load("C:/Users/Clara/Dropbox (CEF - Frankfurt)/AK MMM Results/Clara/Colaborations/SR6_camila/HeLa_SR6/for_hea
```

Comparison - WT = "Hypoxia WT vs Normoxia WT" - OE = "Hypoxia OE vs Normoxia OE" - Hyp = "Hypoxia OE vs Hypoxia WT" - Nor = "Normoxia OE vs Normoxia WT"

First I subset the data set for the regulated genes in WT. Then I create the row annotations on how are the genes regulated in the different conditions and asiged colors.

```
resobject <- as.data.frame(subset(res_all, (res_all$regulated_WT != "Not")))


rowanno = rowAnnotation(Hyp = resobject$regulated_de_Hyp,
                        Nor = resobject$regulated_de_Nor,
                        OE = resobject$regulated_de_OE,
                        WT = resobject$regulated_de_WT,
                        col=list(Hyp=c("Up"="#db6575","Down"="#656bdb","Not"="grey"),
                                 Nor=c("Up"="#db6575","Down"="#656bdb","Not"="grey"),
                                 OE=c("Up"="#db6575","Down"="#656bdb","Not"="grey"),
                                 WT =c("Up"="#db6575","Down"="#656bdb","Not"="grey")
                        ))
```

Here I chose the ones that are up regulated and define the function to get the matrix.

```
hypoxia_upreg <- rownames(res_all)[which(res_all$regulated_WT != "Not")] #coordinates

head(hypoxia_upreg)
## [1] "ENSG00000001497.18" "ENSG00000001630.17" "ENSG00000002549.13"
## [4] "ENSG00000002746.15" "ENSG00000003096.14" "ENSG00000003400.15"
```

Heat Map with values

I do a funcion to get the matrix out of the regulated genes in hypoxia

```
get_matrix <- function(gene_list) {
  matrix <- assay(rld)[ match(gene_list, rownames(assay(rld))), ]
  matrix <- apply(matrix, 1, function (x) {
    (x - mean(x)) / sd(x)            # sat calculatio
  })
  matrix <- t(matrix)
  return(matrix)
}
```

I get the matrix

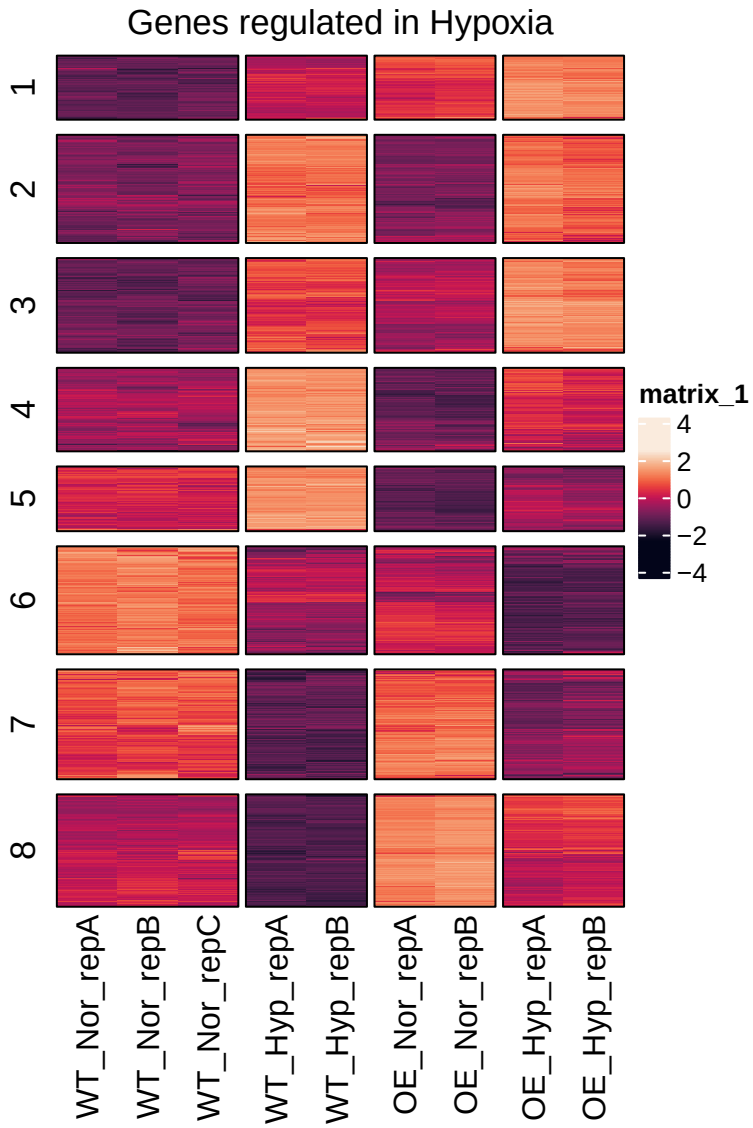
```
matrix_up <- get_matrix(hypoxia_upreg)

head(matrix_up)
##                     OE_Hyp_repA OE_Hyp_repB OE_Nor_repA OE_Nor_repB WT_Hyp_repA
## ENSG00000001497.18  -0.3629060  -0.4313907   1.3578232  1.60706701  -1.1726380
## ENSG00000001630.17  -1.0478698  -0.7969603  -0.5046729 -1.41126574   1.2513368
## ENSG00000002549.13  -0.8513787  -0.6957409   1.2069406  1.36214628  -1.1290492
## ENSG00000002746.15  -1.2568544  -1.2920227   0.5933546  0.33085611  -0.9874157
## ENSG00000003096.14  -1.4972599  -1.3246663   0.2430698  0.12179809  -0.6102622
## ENSG00000003400.15  -1.5069281  -1.4651909  -0.3196268 -0.03806555   0.1678802
##                      WT_Hyp_repB WT_Nor_repA WT_Nor_repB WT_Nor_repC
## ENSG00000001497.18 -1.350987464  0.07867548   0.2139265  0.06043002
## ENSG00000001630.17  1.397159171  0.12715512   0.4684432  0.51667444
## ENSG00000002549.13 -1.252065033  0.35630068   0.5594371  0.44340918
## ENSG00000002746.15 -0.443479918  1.06799895   1.0251612  0.96240184
## ENSG00000003096.14 -0.174645474  1.01721955   1.0197739  1.20497252
## ENSG00000003400.15 -0.008552511  1.11459554   0.8101655  1.24572260

col.order <- c("WT_Nor_repA","WT_Nor_repB","WT_Nor_repC", "WT_Hyp_repA", "WT_Hyp_repB","OE_Nor_repA", "OE_Nor

matrix_up <- matrix_up[,col.order]

head(matrix_up)
##                     WT_Nor_repA WT_Nor_repB WT_Nor_repC WT_Hyp_repA  WT_Hyp_repB
## ENSG00000001497.18  0.07867548   0.2139265  0.06043002  -1.1726380 -1.350987464
## ENSG00000001630.17  0.12715512   0.4684432  0.51667444   1.2513368  1.397159171
## ENSG00000002549.13  0.35630068   0.5594371  0.44340918  -1.1290492 -1.252065033
## ENSG00000002746.15  1.06799895   1.0251612  0.96240184  -0.9874157 -0.443479918
## ENSG00000003096.14  1.01721955   1.0197739  1.20497252  -0.6102622 -0.174645474
## ENSG00000003400.15  1.11459554   0.8101655  1.24572260   0.1678802 -0.008552511
##                     OE_Nor_repA OE_Nor_repB OE_Hyp_repA OE_Hyp_repB
## ENSG00000001497.18   1.3578232  1.60706701  -0.3629060  -0.4313907
## ENSG00000001630.17  -0.5046729 -1.41126574  -1.0478698  -0.7969603
## ENSG00000002549.13   1.2069406  1.36214628  -0.8513787  -0.6957409
## ENSG00000002746.15   0.5933546  0.33085611  -1.2568544  -1.2920227
## ENSG00000003096.14   0.2430698  0.12179809  -1.4972599  -1.3246663
## ENSG00000003400.15  -0.3196268 -0.03806555  -1.5069281  -1.4651909
```

Heat map Plot

Heat Map with values

```
        set.seed(123)

        clustermap <- Heatmap(matrix_up,
                              cluster_columns=F,
                              column_title = "Genes regulated in Hypoxia",
                              row_dend_reorder = T,
                              #right_annotation=rowanno,
                              show_row_names = FALSE,
                              use_raster=F, cluster_rows = TRUE,
                              col = viridis::rocket(1000),
                              show_row_dend=F,
                              column_split=c("A","A", "A", "B","B","C","C","D","D"),
                              border=T, row_gap=unit(2,"mm"),
                              column_gap=unit(1,"mm"),
                              row_km=8)
        clustermap <- draw(clustermap)
```

Heat Map with values



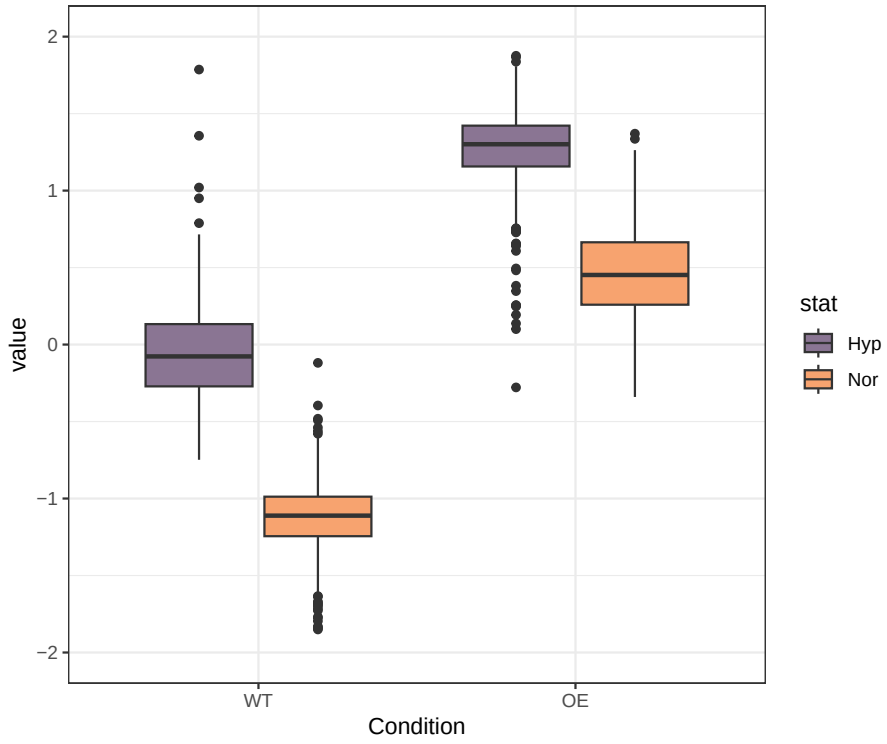

```
# Annotation for percentages of H3R2me_pro and H3R2me_enh marks per cluster
set.seed(123)
clusterindices <- row_order(clustermap) #indices form matrix
```

Here I get the indices used for creating the heat map and get it to a data.frame.

```
df = matrix_up[clusterindices$`1`,] %>%
  as.data.frame() %>%
  tibble::rownames_to_column("ID") %>%
  pivot_longer(-ID) %>%
  mutate(cond = ifelse(grepl("WT", name), "WT", "OE")) %>%
  mutate(stat = ifelse(grepl("Nor", name), "Nor", "Hyp")) %>%
  mutate(cond = factor(cond, levels = c( "WT", "OE")))
```

Heat Map with values

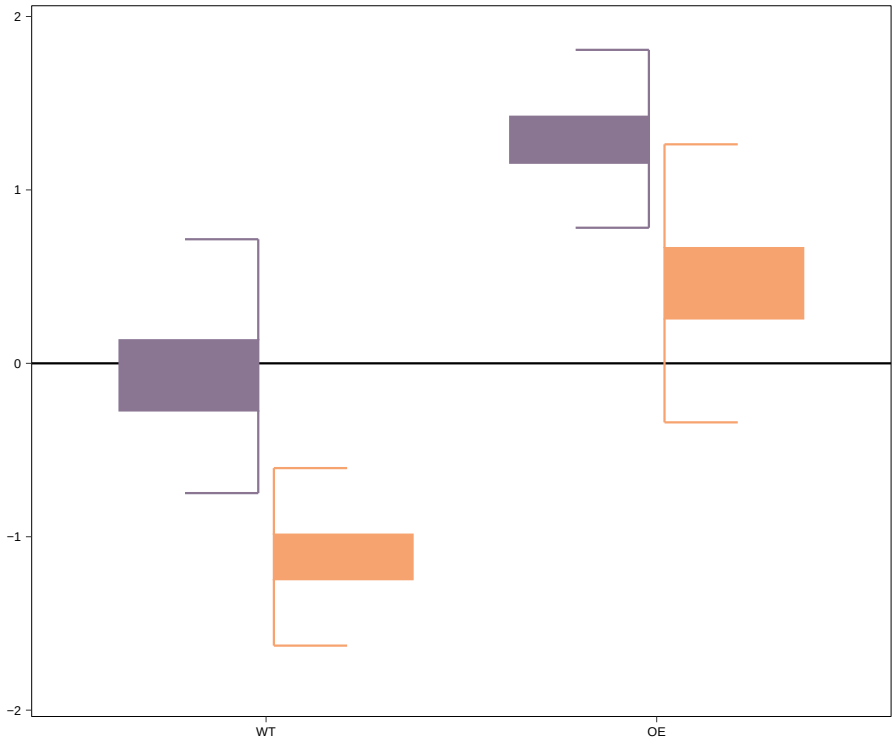
```
ggplot(df, aes(x = cond, y = value, fill = stat)) +
  geom_boxplot() +
  ylim(-2,2) +
  scale_fill_manual(values = c(Hyp = "#8A7593", Nor = "#F7A36F")) +
  theme(legend.position = "none") +
  labs(x = "Condition", y = "value") +
  theme_bw()
```



To make the heat map easier to read we use GGhalf for the box-plots with the indices

```
ggplot(df, aes(x = cond, y = value, fill = stat, color = stat)) +
  geom_hline(yintercept = 0) +
  geom_half_boxplot(data = subset(df, stat == "Hyp"), aes(x = cond, y = value, fill = stat, color = stat), nu
  geom_half_boxplot(data = subset(df, stat == "Nor"), aes(x = cond, y = value, fill = stat, color = stat), nu
  scale_fill_manual(values = c(Hyp = "#8A7593", Nor = "#F7A36F")) +
  scale_color_manual(values = c(Hyp = "#8A7593", Nor = "#F7A36F")) +
  theme_pub() +
  theme(legend.position = "none") +
  labs(x = "Condition", y = "value") +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        panel.grid = element_blank())
```

Heat Map with values



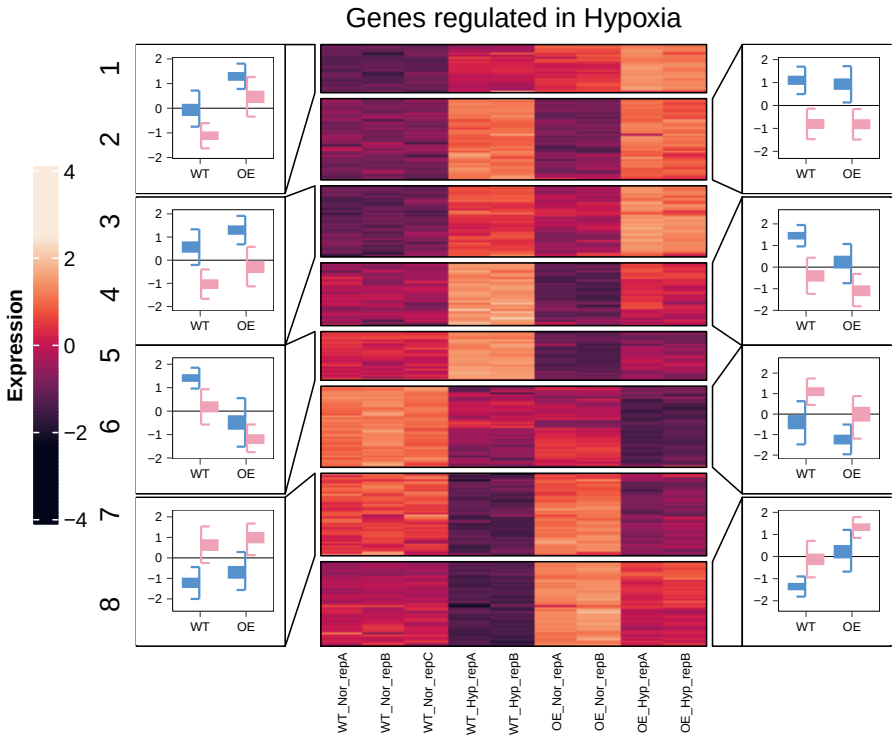
Heat map we add the gghalf box plots to the heat map

```
set.seed(123)
m = matrix_up

s1 = clusterindices[names(clusterindices) %in% c("1","3", "5", "7")]
s2 = clusterindices[names(clusterindices) %in% c("2","4", "6", "8")]

pFun2 <- function(index){
  grid.rect()

  df = m[index,] %>%
    as.data.frame() %>%
    tibble::rownames_to_column("ID") %>%
    tidyr::pivot_longer(-ID) %>%
    mutate(cond = ifelse(grepl("WT", name), "WT", "OE")) %>%
    mutate(stat = ifelse(grepl("Nor", name), "Nor", "Hyp")) %>%
    mutate(cond = factor(cond, levels = c( "WT", "OE")))


  p = ggplot(df, aes(x = cond, y = value, fill = stat, color = stat)) +
    geom_hline(yintercept = 0, linewidth= 0.2) +
    geom_half_boxplot(data = subset(df, stat == "Hyp"),
                      aes(x = cond, y = value, fill = stat, color = stat),
                      nudge = 0.02, outlier.alpha = 0, side = "l") +
    geom_half_boxplot(data = subset(df, stat == "Nor"),
                      aes(x = cond, y = value, fill = stat, color = stat),
```

Heat Map with values

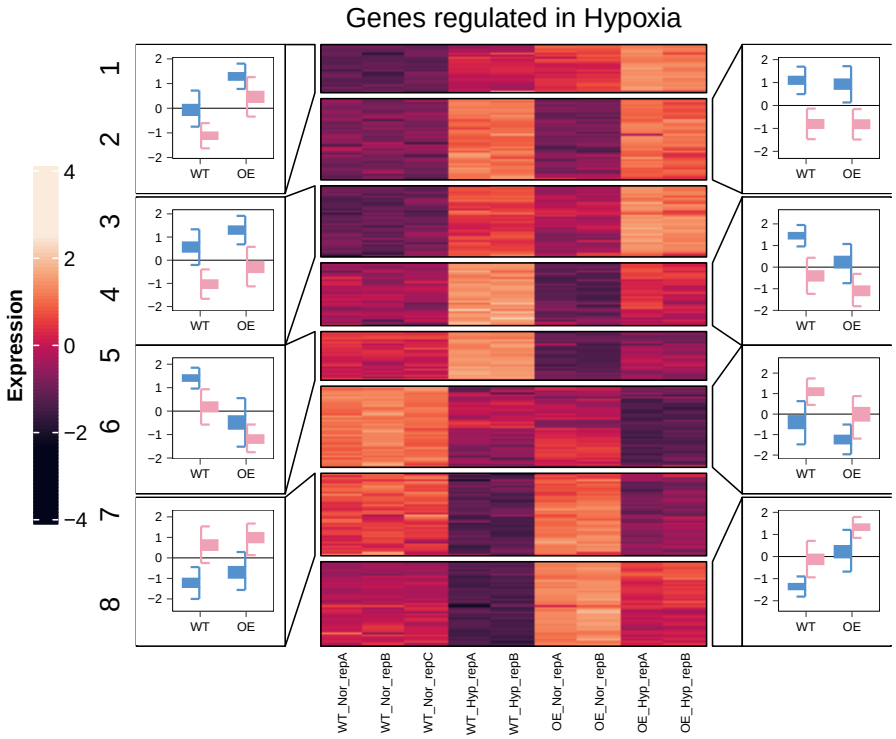
```
                        nudge = 0.02, outlier.alpha = 0, side = "r") +
    scale_fill_manual(values = c(Hyp = "#5592CD", Nor = "#EFA1B5")) +
    scale_color_manual(values = c(Hyp = "#5592CD", Nor = "#EFA1B5")) +
    theme_pub() +
    theme(legend.position = "none") +
    labs(x = "Condition", y = "value") +
    theme(axis.title.x = element_blank(),
          axis.title.y = element_blank(),
          aspect.ratio=1,
          panel.background = element_blank(),
          panel.grid = element_blank())

  gt = ggplot_gtable(ggplot_build(p))
  grid.draw(gt)
}

haZoomL = anno_zoom(align_to = s1, which = "row", panel_fun = pFun2, side = "left",
                    size = unit(2.5, "cm"), gap = unit(1, "cm"), width = unit(3, "cm"))
haZoomR = anno_zoom(align_to = s2, which = "row", panel_fun = pFun2, side = "right",
                    size = unit(2.5, "cm"), gap = unit(1, "cm"), width = unit(3, "cm"))


h <- Heatmap(m,
             heatmap_legend_param = list(
               legend_height = unit(6, "cm"),
               title_position = "leftcenter-rot",
               title = "Expression"
             ),
             cluster_columns=FALSE,
             column_title = "Genes regulated in Hypoxia",
             show_row_names = FALSE,
             use_raster=T,
             cluster_rows = TRUE,
             col = viridis::rocket(1000),
             show_row_dend=F,
             left_annotation = rowAnnotation(profile = haZoomL),
             right_annotation = rowAnnotation(profile = haZoomR),
             border=T,
             row_gap=unit(1,"mm"),
             row_km=8,
             row_dend_reorder = TRUE,
             column_names_gp = gpar(fontsize = 6),
             row_names_gp = gpar(fontsize = 6)

)


set.seed(123)

draw(h, heatmap_legend_side = "left")
```

Heat Map with values




```
# Getting more info

set.seed(123)
draw(h, heatmap_legend_side = "left")
```

Heat Map with values




```
clusterindices <- row_order(h) #indices form matrix
```

We get the genes (ensembl ID) of each cluster

```
all_clusters <- lapply(clusterindices, function(i){
  rownames(matrix_up)[i]
})
```

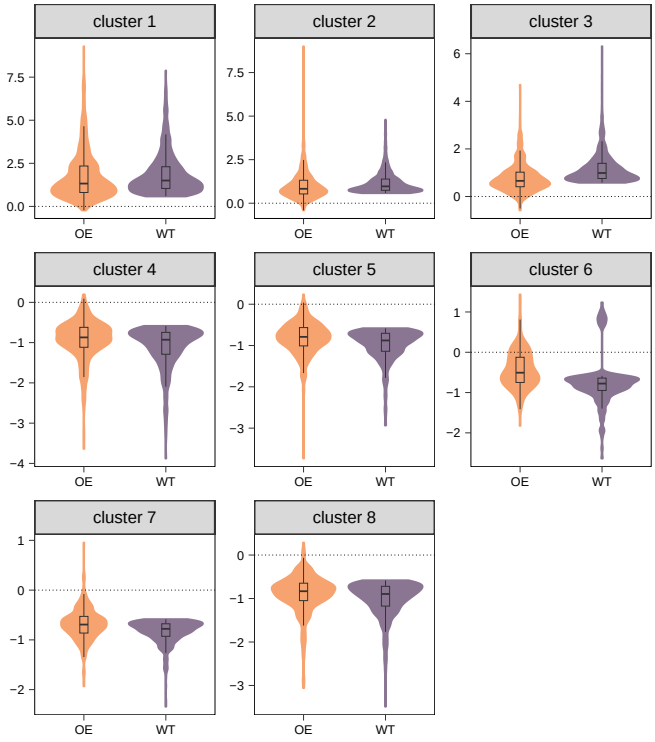
We extract the log2FC. And we do a ggplot friendly table

```
log2FC_all_clusters <- lapply(all_clusters, function(i){
  WT= res_all$log2FoldChange_WT[rownames(res_all)%in% i]
  OE= res_all$log2FoldChange_OE[rownames(res_all)%in% i]
  return(list(WT,OE))
})

log2FC_all_clusters_df <- (melt(log2FC_all_clusters))

log2FC_all_clusters_df <- mutate(log2FC_all_clusters_df, comparisson = factor(
  case_when(
    L2 == 1 ~ "WT",
    L2 == 2 ~ "OE",
    TRUE ~ "Not"))
  )

log2FC_all_clusters_df$cluster <- paste0("cluster ",
                                         log2FC_all_clusters_df$L1)
```

Heat Map with values

Plot

```
ggplot(log2FC_all_clusters_df,
       aes(y=value, x = comparisson, fill=comparisson))+
  geom_violin(data = log2FC_all_clusters_df,
                  aes(x = comparisson, y = value, fill = comparisson, color = comparisson),
                  nudge = 0.02, outlier.alpha = 0, side = "r") +
  scale_fill_manual(values = c(WT = "#8A7593", OE = "#F7A36F")) +
  scale_color_manual(values = c(WT = "#8A7593", OE = "#F7A36F")) +
  theme_pub() +
  scale_y_continuous(labels = scales::comma) +
  geom_boxplot(width=0.1, outlier.alpha = 0, size=0.2) +
  geom_hline(yintercept = 0, linetype = "dotted", size= 0.2) +
  theme(legend.position = "none") +
  labs(x = "Condition", y = "value") +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        aspect.ratio=1,
        panel.grid = element_blank()) +
  facet_wrap(~ cluster, scales = "free")
```



gene names

```
names_all_clusters <- lapply(all_clusters, function(i){
  WT= res_all$ensembl_gene_id[rownames(res_all)%in% i]
  OE= res_all$ensembl_gene_id[rownames(res_all)%in% i]
  return(list(WT,OE))
})
```

Heat Map with values

```
names_all_clusters_df <- (melt(names_all_clusters))

names_all_clusters_df <- mutate(names_all_clusters_df, comparisson = factor(
  case_when(
    L2 == 1 ~ "WT",
    L2 == 2 ~ "OE",
    TRUE ~ "Not"))
)

names_all_clusters_df$cluster <- paste0("cluster ",
                                        names_all_clusters_df$L1)
```