Heatmap of gene expr of bound TFs in pb

Amy Huang

02 August, 2023

Contents

Read in data

```
mouse_human_genes <- readRDS(here("scRNAseq", "data", "mouse_human_genes.rds"))
motifs_in_enh <- read_tsv(
  here(
    "ATACseq",
    "data",
    "processed_data_tables",
    "02_footprinting.TFBS_in_enh.tsv"
  )
)
motifs_bound_in_enh <- motifs_in_enh %>%
  filter(KW_CN_bound == 1 | KW_S_bound == 1)

genes_binding_in_enh <- motifs_bound_in_enh$TFBS_name %>%
  str_sub(end = -10) %>%
  unique()
mouse_genes <- genes_binding_in_enh %>% str_subset(".*[a-z].*")
human_genes <- genes_binding_in_enh %>% str_subset("^.*[a-z].*$", negate = TRUE)
mouse_genes_binding_in_enh <- mouse_human_genes %>%
  filter(HGNC.symbol %in% human_genes | MGI.symbol %in% mouse_genes) %>%
  .$MGI.symbol
```

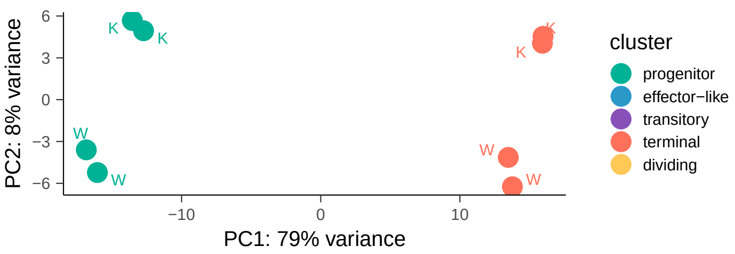
Run DESeq2 to compare clusters

```r
norm_counts_df <-
  read_tsv(
    here(
      "scRNAseq",
      "data",
      "processed_data_tables",
      "05_pseudobulk.norm_counts.tsv"
    )
  )
counts_df <- read_tsv(here(
  "scRNAseq",
  "data",
  "processed_data_tables",
  "05_pseudobulk.counts.tsv"
))
```

```r
norm_count_selected <- norm_counts_df[c("gene", str_subset(colnames(norm_counts_df), "_[W|K]_progenitor
  filter(gene %in% mouse_genes_binding_in_enh)
counts_selected <- counts_df[c("gene", str_subset(colnames(counts_df), "_[W|K]_progenitor|_[W|K]_termina
```

```r
metadata <- tibble(sample = colnames(counts_selected)[-1]) %>%
  mutate(genotype = str_sub(str_extract(sample, "_._"), start = 2, end = -2)) %>%
  mutate(cluster = str_sub(str_extract(sample, "_._.*$"), start = 4)) %>%
  column_to_rownames("sample")

deseq_obj <- DESeqDataSetFromMatrix(
  countData = column_to_rownames(counts_selected, "gene"),
  colData = metadata,
  design = ~cluster + genotype
)
deseq_obj <- DESeq(deseq_obj)
rlog_obj <- rlog(deseq_obj)
```

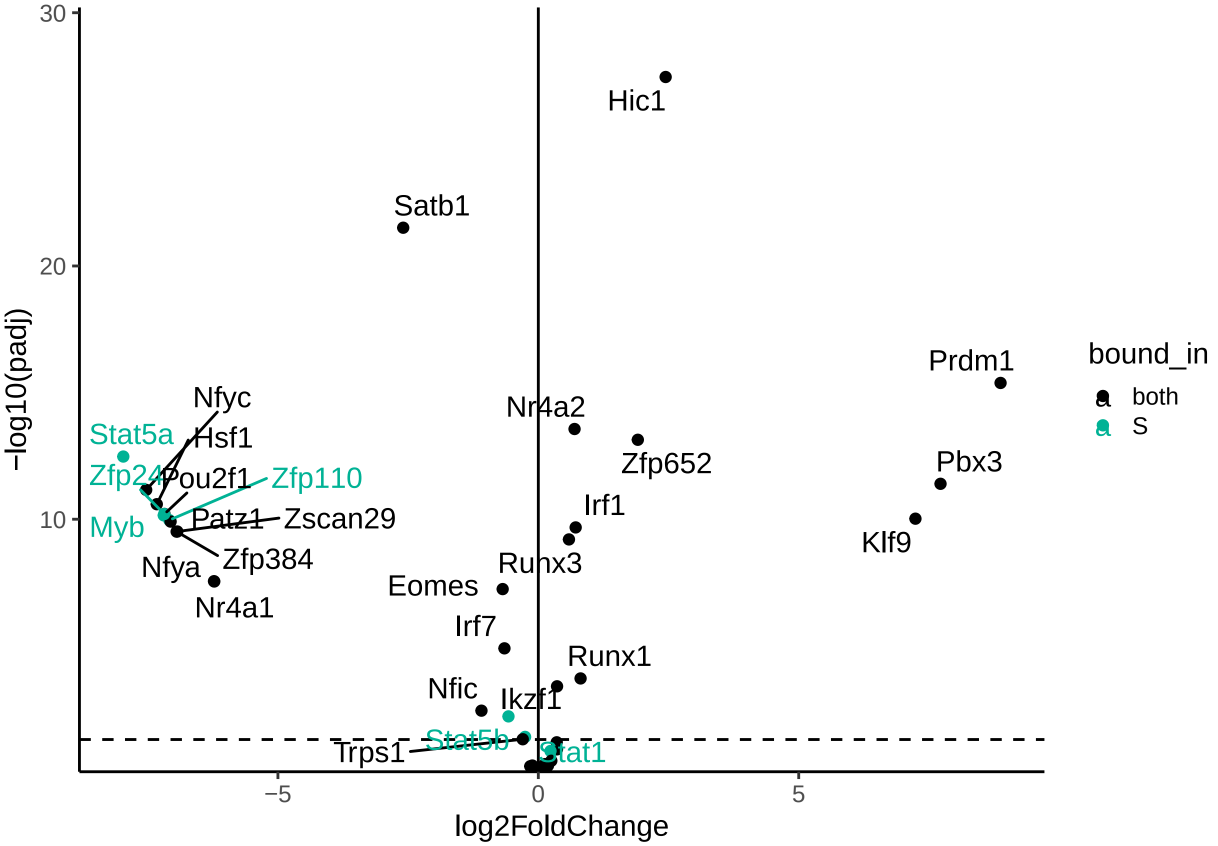
```r
plotPCA(rlog_obj, intgroup = "cluster", ntop=6415) +
  scale_color_manual(values = palette_cluster) +
  labs(color = "cluster") +
  theme(axis.text = element_text(size = 6),
        axis.title = element_text(size = 8),
        legend.text = element_text(size = 6),
        legend.title = element_text(size = 8),
        line = element_line(size = 0.2)) +
  geom_text_repel(aes(label = str_extract(name, ".")),
                  size = 6*GGPLOT_TEXT_SCALE_FACTOR)
```



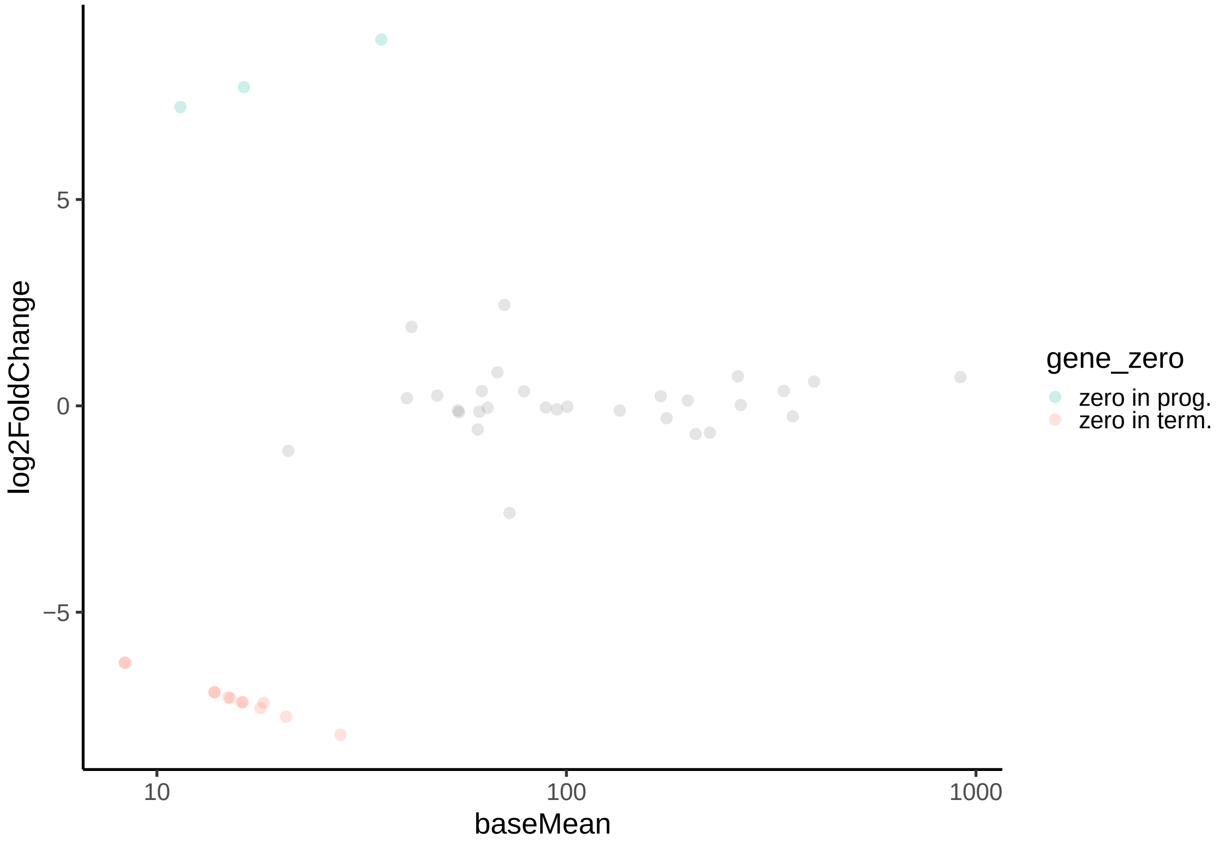
Plot bound motifs

```
h_mtx <- norm_count_selected %>%
  column_to_rownames("gene") %>%
  as.matrix() %>%
  t()

col_ann_df <-
  motifs_bound_in_enh %>%
  mutate(gene = str_sub(TFBS_name, end = -10)) %>%
  filter(gene %in% c(mouse_human_genes$MGI.symbol, mouse_human_genes$HGNC.symbol)) %>%
  left_join(mouse_human_genes, by = c("gene" = "HGNC.symbol")) %>%
  mutate(gene = ifelse(is.na(MGI.symbol), gene, MGI.symbol)) %>%
  group_by(gene) %>%
  summarise(mean_S = mean(KW_S_score, na.rm = TRUE),
            mean_CN = mean(KW_CN_score, na.rm = TRUE),
            n_bound_in_S = sum(KW_S_bound == 1),
            n_bound_in_CN = sum(KW_CN_bound == 1)) %>%
  mutate(bound_in = case_when(n_bound_in_CN == 0 ~ "S",
                              n_bound_in_S == 0 ~ "CN",
                              TRUE ~ "both")) %>%
  column_to_rownames("gene") %>%
  .[colnames(h_mtx), ]
```

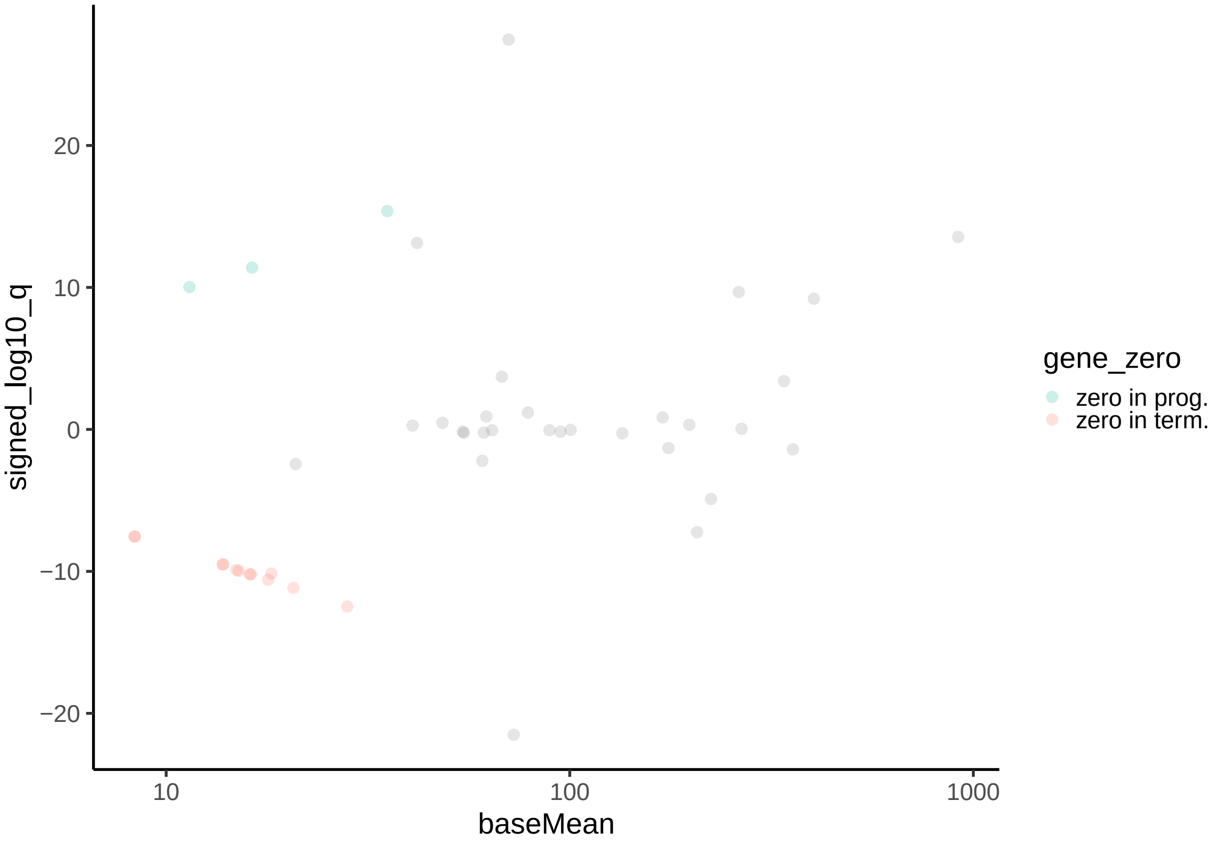
```
volcano_df <- results(deseq_obj, name = "cluster_terminal_vs_progenitor") %>%
  as_tibble(rownames = "gene") %>%
  filter(gene %in% mouse_genes_binding_in_enh) %>%
  drop_na(log2FoldChange) %>% # 2 genes with NA log2FoldChange
  mutate(is_signif = padj < 0.05) %>%
  left_join(as_tibble(col_ann_df, rownames = "gene"),
            by = "gene")
volcano_df %>%
  ggplot() +
  aes(log2FoldChange, -log10(padj)) +
  geom_vline(xintercept = 0) +
  geom_hline(yintercept = -log10(0.05), linetype = "dashed") +
  geom_point(aes(color = bound_in)) +
  geom_text_repel(aes(label = gene, color = bound_in),
                  data = ~..1 %>% filter(is_signif),
                  max.overlaps = Inf) +
  scale_y_continuous(expand = c(0, 0), limits = ~c(..1[1], ..1[2]*1.1)) +
  scale_color_manual(values = c("both" = "black", "S" = palette_cluster_ids[["progenitor exhausted"]]))
```



```r
gene_zeros_in_prog <- counts(deseq_obj, normalized = TRUE) %>%
  as_tibble(rownames = "gene") %>%
  filter(K1_K_progenitor == 0 &
           K2_K_progenitor == 0 &
           W1_W_progenitor == 0 &
           W2_W_progenitor == 0) %>%
  .$gene
gene_zeros_in_term <- counts(deseq_obj, normalized = TRUE) %>%
  as_tibble(rownames = "gene") %>%
  filter(K1_K_terminal == 0 &
           K2_K_terminal == 0 &
           W1_W_terminal == 0 &
           W2_W_terminal == 0) %>%
  .$gene
volcano_df %>%
  filter(baseMean != 0) %>%
  mutate(gene_zero = case_when(
    gene %in% gene_zeros_in_prog ~ "zero in prog.",
    gene %in% gene_zeros_in_term ~ "zero in term."
  )) %>%
  ggplot() +
  aes(baseMean, log2FoldChange, color = gene_zero) +
  geom_point(alpha = 0.2) +
  scale_x_log10() +
  scale_color_manual(values = c("zero in prog." = palette_cluster[["progenitor"]],
                                "zero in term." = palette_cluster[["terminal"]]))
```

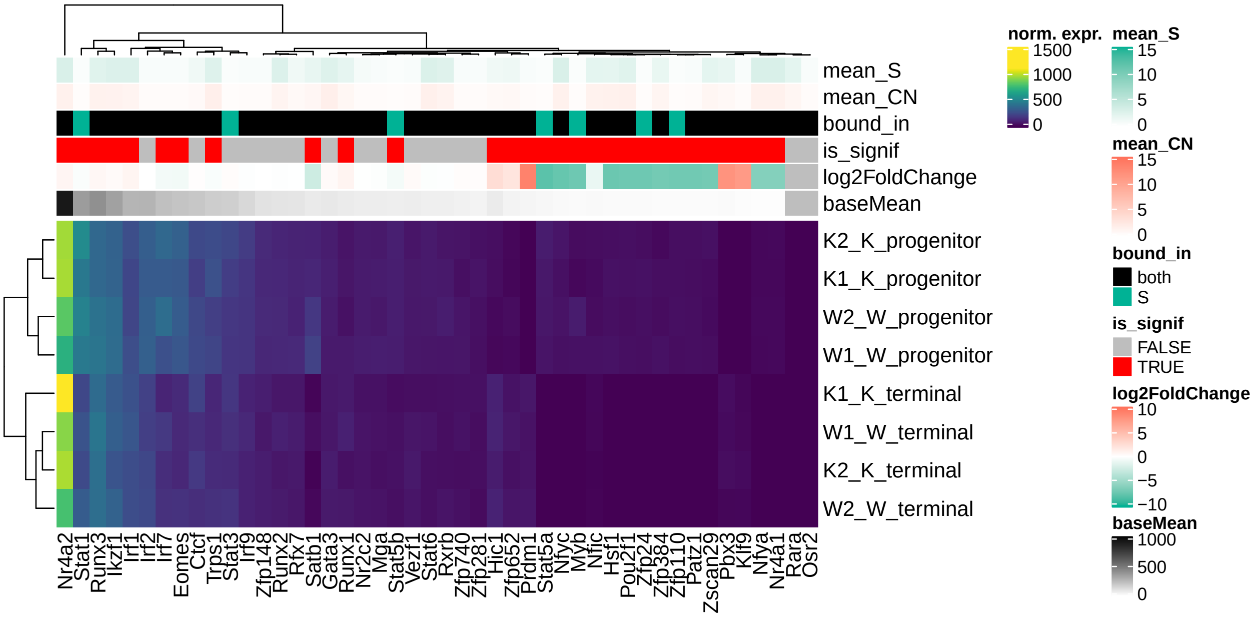


```
volcano_df %>%
  filter(baseMean != 0) %>%
  mutate(signed_log10_q = ifelse(log2FoldChange > 0,
                                 -log10(padj),
                                 log10(padj))) %>%
  mutate(gene_zero = case_when(
    gene %in% gene_zeros_in_prog ~ "zero in prog.",
    gene %in% gene_zeros_in_term ~ "zero in term."
  )) %>%
  ggplot() +
  aes(baseMean, signed_log10_q, color = gene_zero) +
  geom_point(alpha = 0.2) +
  scale_x_log10() +
  scale_color_manual(values = c("zero in prog." = palette_cluster[["progenitor"]],
                                "zero in term." = palette_cluster[["terminal"]]))
```

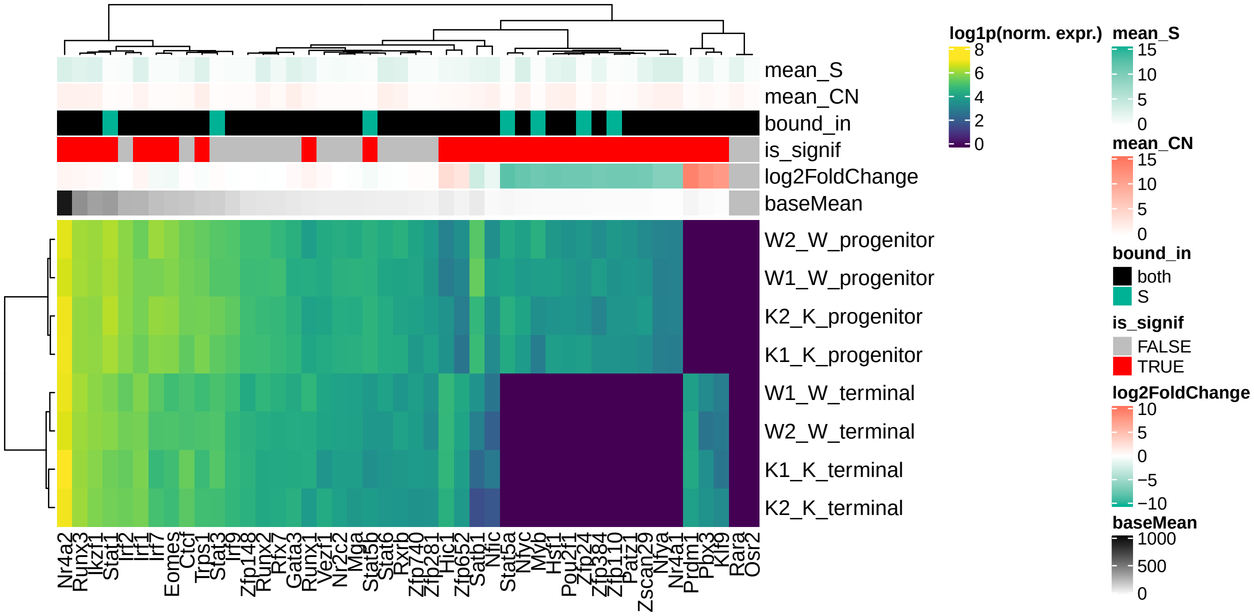


```
col_ann_df <- col_ann_df %>%
  dplyr::select(-n_bound_in_S, -n_bound_in_CN) %>%
  as_tibble(rownames = "gene") %>%
  left_join(dplyr::select(volcano_df, gene, is_signif, log2FoldChange, baseMean),
            by = "gene") %>%
  column_to_rownames("gene")

color_list <- list(
  mean_S = colorRamp2(
    breaks = c(0, 15),
    colors = c("white", palette_cluster_ids[["progenitor exhausted"]])
  ),
  mean_CN = colorRamp2(
    breaks = c(0, 15),
    colors = c("white", palette_cluster_ids[["terminally exhausted"]])
  ),
  bound_in = c("both" = "black",
               "S" = palette_cluster_ids[["progenitor exhausted"]]),
  is_signif = c("TRUE" = "red", "FALSE" = "grey"),
  log2FoldChange = colorRamp2(
    breaks = c(-10, 0, 10),
    colors = c(palette_cluster_ids[["progenitor exhausted"]],
               "white",
               palette_cluster_ids[["terminally exhausted"]])
  ),
  baseMean = colorRamp2(
    breaks = c(0, 1000),
    colors = c("white", "black")
  )
)

col_ann <- HeatmapAnnotation(
  df = col_ann_df,
  col = color_list
)
```

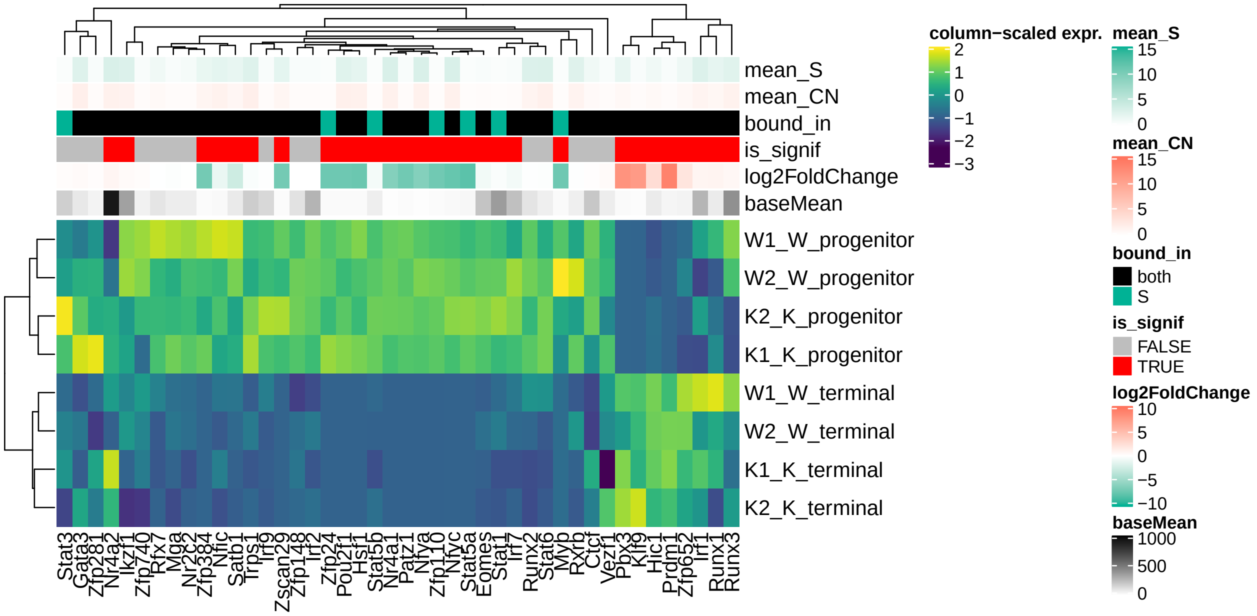
```
Heatmap(
  h_mtx,
  name = "norm. expr.",
  col = viridis(100),
  top_annotation = col_ann,
  column_dend_reorder = TRUE
)
```



```
Heatmap(
  log(h_mtx + 1),
  name = "log1p(norm. expr.)",
  col = viridis(100),
  top_annotation = col_ann,
  column_dend_reorder = TRUE
)
```



```
scaled_h_mtx <- h_mtx %>%
  t() %>%
  {
    .[apply(., 1, sum) != 0, ]
  } %>%
  t() %>%
  scale()
Heatmap(
  scaled_h_mtx,
  name = "column-scaled expr.",
  col = viridis(100),
  top_annotation = HeatmapAnnotation(df = col_ann_df[colnames(scaled_h_mtx), ],
                                     col = color_list),
  column_dend_reorder = TRUE
)
```



What about unbound motifs?

```
motifs_in_enh_w_gene <- motifs_in_enh %>%
  mutate(gene = str_sub(TFBS_name, end = -10)) %>%
  filter(gene %in% c(mouse_human_genes$MGI.symbol, mouse_human_genes$HGNC.symbol)) %>%
  left_join(mouse_human_genes, by = c("gene" = "HGNC.symbol")) %>%
  mutate(gene = ifelse(is.na(MGI.symbol), gene, MGI.symbol)) %>%
  select(-MGI.symbol)
```

There are 169 motifs found in the enhancer that have an potentially associated murine gene.

```
motifs_in_enh_w_gene$gene %>% unique() %>% length()
```

```
## [1] 169
```

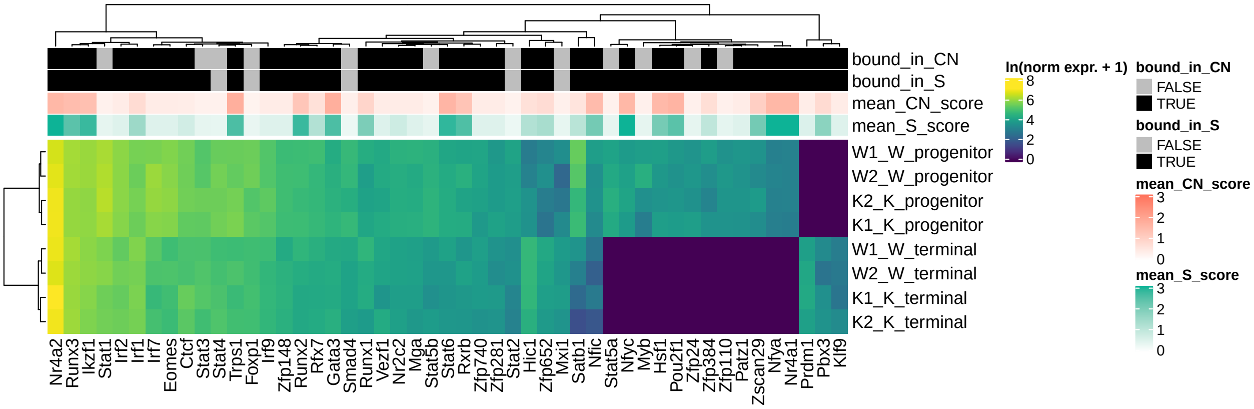
There are 49 which are found in our normalized pseudobulk counts matrix.

```
select_cols <-
  c("gene",
    colnames(norm_counts_df) %>%
      str_subset("progenitor|terminal") %>%
      str_subset("K|W"))
norm_counts_mtx_filtered <- norm_counts_df %>%
  .[, select_cols] %>%
  filter(gene %in% motifs_in_enh_w_gene$gene) %>%
  column_to_rownames("gene") %>%
  as.matrix() %>%
  {
    .[apply(., 1, sum) != 0, ] # filter out columns which are all zeros
  }
nrow(norm_counts_mtx_filtered)
```

```
## [1] 49
```

```
col_ann_df <- motifs_in_enh_w_gene %>%
  filter(gene %in% rownames(norm_counts_mtx_filtered)) %>%
  group_by(gene) %>%
  summarise(bound_in_CN = sum(KW_CN_bound) > 0,
            bound_in_S = sum(KW_S_bound) > 0,
            mean_CN_score = mean(KW_CN_score),
            mean_S_score = mean(KW_S_score)) %>%
  column_to_rownames("gene")
col_ann <- HeatmapAnnotation(
  df = col_ann_df,
  col = list(
    bound_in_CN = c("TRUE" = "black", "FALSE" = "grey"),
    bound_in_S = c("TRUE" = "black", "FALSE" = "grey"),
    mean_CN_score = colorRamp2(
      breaks = c(0, 3),
      colors = c("white", palette_cluster_ids[["terminally exhausted"]])
    ),
    mean_S_score = colorRamp2(
      breaks = c(0, 3),
      colors = c("white", palette_cluster_ids[["progenitor exhausted"]])
    )
  )
)
```

```
norm_counts_mtx_filtered %>%
  t() %>%
  {log(. + 1)} %>%
  .[, rownames(col_ann_df)] %>%
  Heatmap(name = "ln(norm expr. + 1)",
          col = viridis(100),
          top_annotation = col_ann)
```



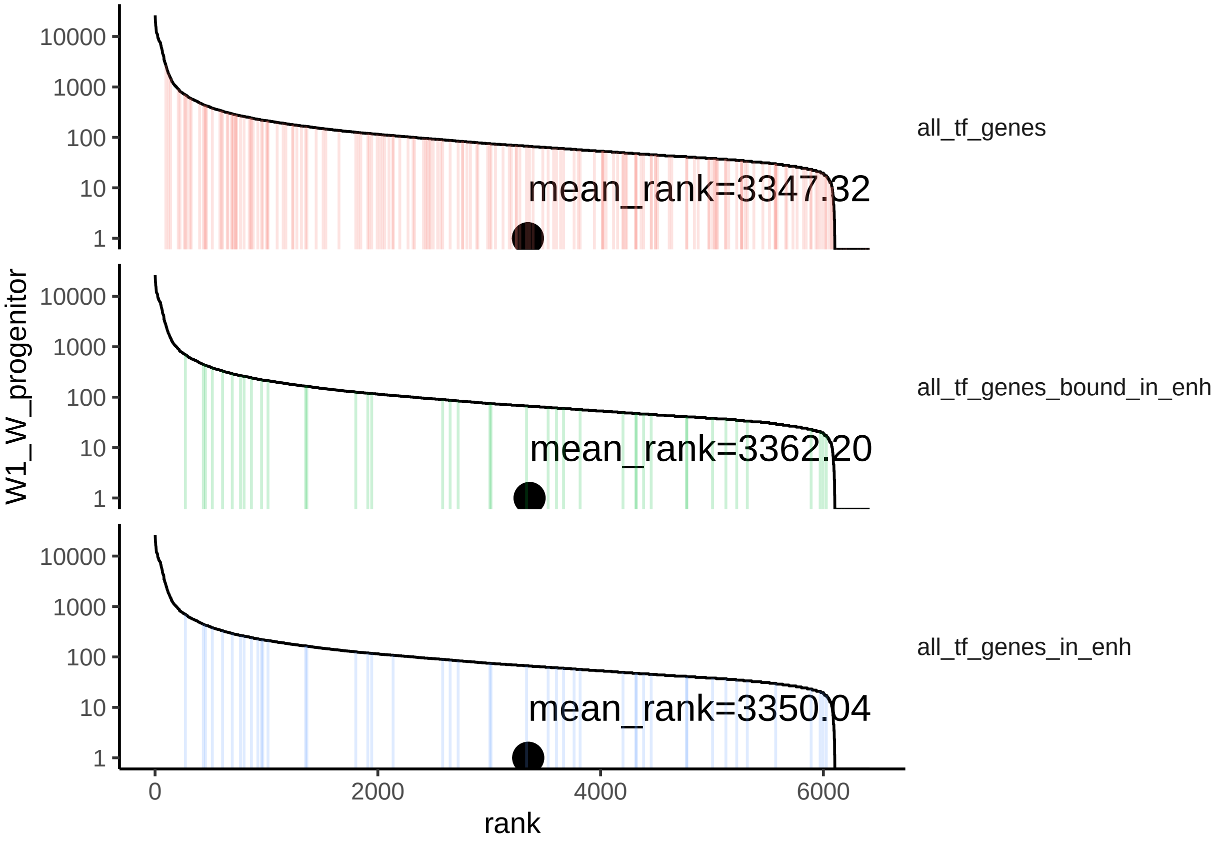
Expression of bound, unbound, and unrelated motifs

```
jaspar_path <- here("ATACseq", "data", "footprinting", "JASPAR2022_CORE_non-redundant_pfms.meme")
test_df <- read_tsv(jaspar_path, col_names = FALSE)
all_tf_genes_df <- test_df %>%
  filter(X1 %>% str_detect("MOTIF")) %>%
  mutate(gene = str_sub(X1, start = 25)) %>%
  select(-X1) %>%
  left_join(mouse_human_genes, by = c("gene" = "HGNC.symbol")) %>%
  mutate(in_mouse = gene %in% mouse_human_genes$MGI.symbol) %>%
  mutate(mouse_gene = case_when(
    in_mouse ~ gene, # we found the mouse gene and it matched with a human gene
    !is.na(MGI.symbol) ~ MGI.symbol, # we found an equivalent human gene
  )) %>%
  drop_na(mouse_gene) %>%
  select(mouse_gene, in_mouse) %>%
  dplyr::rename(gene = mouse_gene)
```

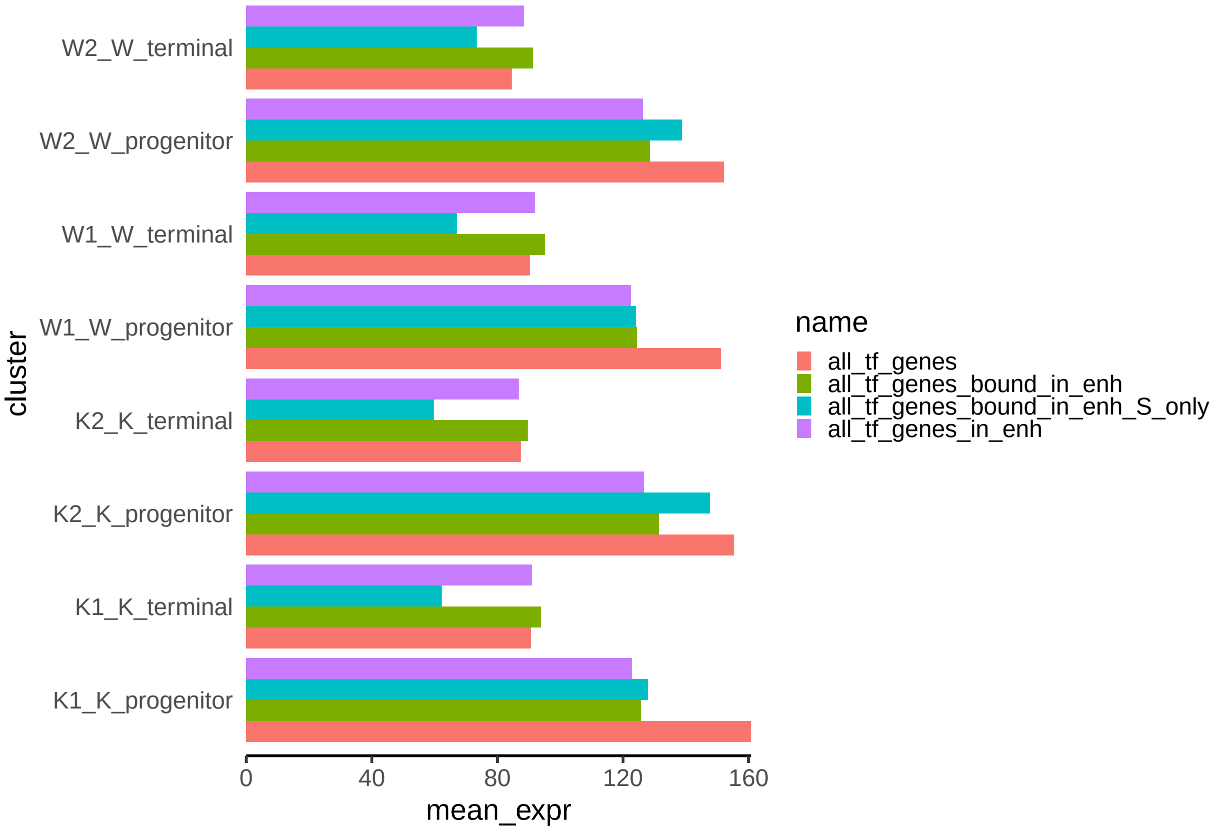
```
motifs_in_enh_w_gene_grouped <- motifs_in_enh_w_gene %>%
  group_by(gene) %>%
  summarise(KW_CN_score_mean = mean(KW_CN_score),
            KW_S_score_mean = mean(KW_S_score),
            KW_CN_bound = as.integer(sum(KW_CN_bound) > 0),
            KW_S_bound = as.integer(sum(KW_S_bound) > 0))
```

```
all_genes <- all_tf_genes_df$gene
all_genes_in_enh <- motifs_in_enh_w_gene_grouped$gene
all_genes_bound_in_enh <- motifs_in_enh_w_gene_grouped %>%
  filter(KW_CN_bound == 1 | KW_S_bound == 1) %>%
  .$gene
all_genes_bound_in_end_both <- motifs_in_enh_w_gene_grouped %>%
  filter(KW_CN_bound == 1 & KW_S_bound == 1) %>%
  .$gene
all_genes_bound_in_enh_S_only <- motifs_in_enh_w_gene_grouped %>%
  filter(KW_S_bound == 1 & KW_CN_bound == 0) %>%
  .$gene
```

```
norm_counts_df %>%
  select(gene, W1_W_progenitor) %>%
  arrange(desc(W1_W_progenitor)) %>%
  mutate(rank = row_number()) %>%
  mutate(
    all_tf_genes = gene %in% all_genes,
    all_tf_genes_in_enh = gene %in% all_genes_in_enh,
    all_tf_genes_bound_in_enh = gene %in% all_genes_bound_in_enh
  ) %>%
  pivot_longer(cols = str_subset(colnames(.), "all_tf_genes")) %>%
  ggplot() +
  aes(rank, W1_W_progenitor) +
  geom_line() +
  geom_point(
    aes(x = mean_rank, y = 1),
    data = ~ ..1 %>%
      filter(value) %>%
      group_by(name) %>%
      summarise(mean_rank = mean(rank)),
    size = 5
  ) +
  geom_text(
    aes(
      x = mean_rank,
      y = 10,
      label = sprintf("mean_rank=%s", format(mean_rank, digits = 6))
    ),
    data = ~ ..1 %>%
      filter(value) %>%
      group_by(name) %>%
      summarise(mean_rank = mean(rank)),
    size = 5,
    hjust = 0
  ) +
  geom_segment(data = ~ ..1 %>% filter(value),
               aes(xend = rank, yend = 0, color = name),
               alpha = 0.2) +
  facet_grid(name ~ .) +
  scale_y_log10() +
  theme(strip.text.y = element_text(angle = 0, hjust = 0),
        strip.background = element_blank(),
        legend.position = "none")
```

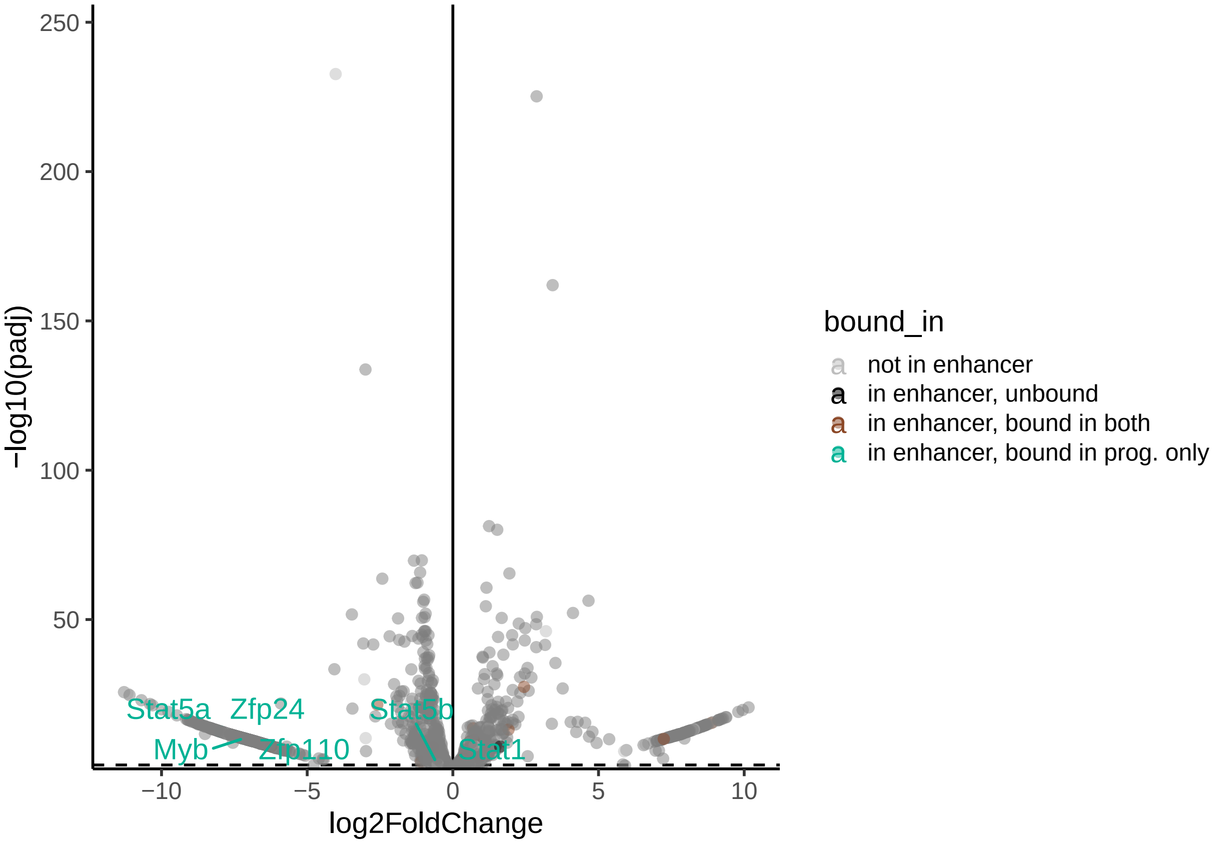


```
plot_df <- norm_counts_df %>%
  select(gene, str_subset(colnames(.), "[K|W]_progenitor|[K_W]_terminal")) %>%
  pivot_longer(cols = 2:ncol(.),
               names_to = "cluster",
               values_to = "expr") %>%
  group_by(cluster) %>%
  arrange(desc(expr)) %>%
  mutate(rank = row_number()) %>%
  mutate(
    all_tf_genes = gene %in% all_genes,
    all_tf_genes_in_enh = gene %in% all_genes_in_enh,
    all_tf_genes_bound_in_enh = gene %in% all_genes_bound_in_enh,
    all_tf_genes_bound_in_enh_S_only = gene %in% all_genes_bound_in_enh_S_only
  ) %>%
  pivot_longer(cols = str_subset(colnames(.), "all_tf_genes")) %>%
  filter(value) %>%
  group_by(name, cluster) %>%
  summarise(mean_rank = mean(rank),
            mean_expr = mean(expr))
plot_df %>%
  ggplot() +
  aes(cluster, mean_expr, fill = name) +
  geom_col(position = "dodge") +
  coord_flip() +
  theme(axis.line.y = element_blank(),
        axis.ticks.y = element_blank()) +
  scale_y_continuous(expand = c(0, 0))
```

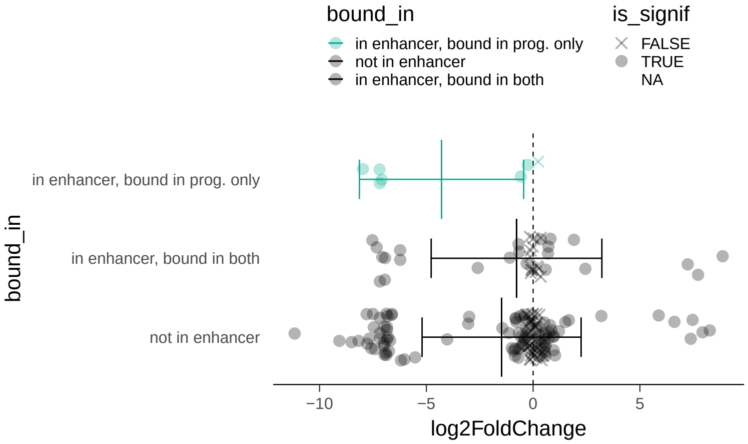


Volcano plot

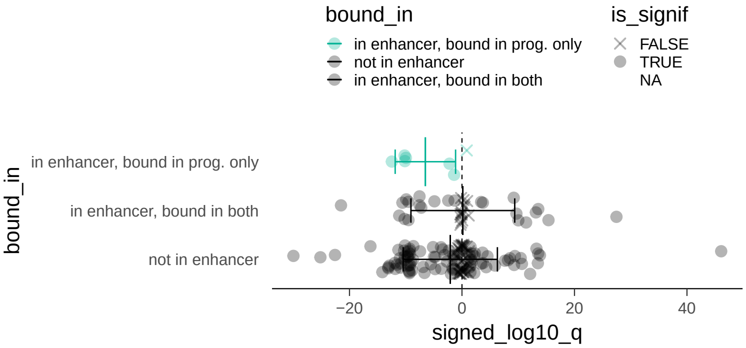
```
volcano_df <- results(deseq_obj, name = "cluster_terminal_vs_progenitor") %>%
  as_tibble(rownames = "gene") %>%
  mutate(is_signif = padj < 0.05) %>%
  mutate(bound_in = case_when(
    gene %in% all_genes_bound_in_enh_S_only ~ "in enhancer, bound in prog. only",
    gene %in% all_genes_bound_in_end_both ~ "in enhancer, bound in both",
    gene %in% all_genes_in_enh ~ "in enhancer, unbound",
    gene %in% all_genes ~ "not in enhancer",
  ))
volcano_df %>%
  ggplot() +
  aes(log2FoldChange,-log10(padj)) +
  geom_vline(xintercept = 0) +
  geom_hline(yintercept = -log10(0.05), linetype = "dashed") +
  geom_point(aes(color = bound_in),
             alpha = 0.5) +
  geom_text_repel(
    aes(label = gene, color = bound_in),
    data = ~ ..1 %>%
      filter(is_signif) %>%
      filter(bound_in == "in enhancer, bound in prog. only"),
    max.overlaps = Inf
  ) +
  scale_y_continuous(expand = c(0, 0), limits = ~ c(..1[1], ..1[2] * 1.1)) +
  scale_color_manual(
    values = c(
      "not in enhancer" = "grey",
      "in enhancer, unbound" = "black",
      "in enhancer, bound in both" = "sienna4",
      "in enhancer, bound in prog. only" = palette_cluster_ids[["progenitor exhausted"]]
    )
  )
```



```
volcano_df %>%
  mutate(bound_in = factor(bound_in, levels = c(
    "not in enhancer",
    "in enhancer, unbound",
    "in enhancer, bound in both",
    "in enhancer, bound in prog. only"
  ))) %>%
  filter(bound_in != "in enhancer, unbound") %>%
  drop_na(bound_in) %>%
  ggplot() +
  aes(bound_in, log2FoldChange) +
  geom_hline(yintercept = 0, linetype = "dashed", size = 0.2) +
  geom_errorbar(
    aes(
      y = log2FoldChange_mean,
      ymin = log2FoldChange_mean - log2FoldChange_sd,
      ymax = log2FoldChange_mean + log2FoldChange_sd,
      color = bound_in
    ),
    data = ~ ..1 %>%
      group_by(bound_in) %>%
      summarise_at(
        c("baseMean", "log2FoldChange", "padj"),
        .funs = c(mean = mean, sd = sd),
        na.rm = TRUE
      ),
    width = 0.5,
    size = 0.25
  ) +
  geom_errorbar(
    aes(y = log2FoldChange_mean,
        ymin = log2FoldChange_mean,
        ymax = log2FoldChange_mean,
      color = bound_in),
    data = ~ ..1 %>%
      group_by(bound_in) %>%
      summarise_at(
        c("baseMean", "log2FoldChange", "padj"),
        .funs = c(mean = mean, sd = sd),
        na.rm = TRUE
      ),
    width = 1,
    size = 0.25
  ) +
  geom_jitter(
    aes(
      # size = -log10(pvalue),
      shape = is_signif,
      color = bound_in
    ),
    width = 0.3,
    alpha = 0.3
  ) +
  coord_flip() +
  scale_shape_manual(values = c(4, 19)) +
  scale_color_manual(values = c("in enhancer, bound in prog. only" = palette_cluster[["progenitor"]],
                                "not in enhancer" = "black",
                                "in enhancer, bound in both" = "black"))+
  theme(legend.position = "top",
        legend.direction = "vertical",
        axis.text = element_text(size = 6),
        axis.title = element_text(size = 8),
        legend.text = element_text(size = 6),
        legend.title = element_text(size = 8),
        line = element_line(size = 0.2),
        legend.spacing = unit(0, "pt"),
        axis.line.y = element_blank(),
        axis.ticks.y = element_blank())
```



```
volcano_df %>%
  filter(gene != "Klf2") %>%
  mutate(signed_log10_q = ifelse(log2FoldChange > 0,
                                 -log10(padj),
                                 log10(padj))) %>%
  mutate(bound_in = factor(bound_in, levels = c(
    "not in enhancer",
    "in enhancer, unbound",
    "in enhancer, bound in both",
    "in enhancer, bound in prog. only"
  ))) %>%
  filter(bound_in != "in enhancer, unbound") %>%
  drop_na(bound_in) %>%
  ggplot() +
  aes(bound_in, signed_log10_q) +
  geom_hline(yintercept = 0, linetype = "dashed", size = 0.2) +
  geom_errorbar(
    aes(
      y = signed_log10_q_mean,
      ymin = signed_log10_q_mean - signed_log10_q_sd,
      ymax = signed_log10_q_mean + signed_log10_q_sd,
      color = bound_in
    ),
    data = ~ ..1 %>%
      group_by(bound_in) %>%
      summarise_at(
        c("baseMean", "log2FoldChange", "signed_log10_q"),
        .funs = c(mean = mean, sd = sd),
        na.rm = TRUE
      ),
    width = 0.5,
    size = 0.25
  ) +
  geom_errorbar(
    aes(y = signed_log10_q_mean,
        ymin = signed_log10_q_mean,
        ymax = signed_log10_q_mean,
      color = bound_in),
    data = ~ ..1 %>%
      group_by(bound_in) %>%
      summarise_at(
        c("baseMean", "log2FoldChange", "signed_log10_q"),
        .funs = c(mean = mean, sd = sd),
        na.rm = TRUE
      ),
    width = 1,
    size = 0.25
  ) +
  geom_jitter(
    aes(
      shape = is_signif,
      color = bound_in
    ),
    width = 0.3,
    alpha = 0.3
  ) +
  coord_flip() +
  scale_shape_manual(values = c(4, 19)) +
  scale_color_manual(values = c("in enhancer, bound in prog. only" = palette_cluster[["progenitor"]],
                               "not in enhancer" = "black",
                               "in enhancer, bound in both" = "black"))+
  theme(legend.position = "top",
        legend.direction = "vertical",
        axis.text = element_text(size = 6),
        axis.title = element_text(size = 8),
        legend.text = element_text(size = 6),
        legend.title = element_text(size = 8),
        line = element_line(size = 0.2),
        legend.spacing = unit(0, "pt"),
        axis.line.y = element_blank(),
        axis.ticks.y = element_blank())
```



There's one gene that's an outlier that we had to remove.

```
volcano_df %>%
  mutate(signed_log10_q = ifelse(log2FoldChange > 0,
                                 -log10(padj),
                                 log10(padj))) %>%
  arrange(signed_log10_q) %>%
  head(1)
```

```
## # A tibble: 1 x 10
##   gene  baseMean log2FoldChange lfcSE  stat    pvalue      padj is_signif
##   <chr>    <dbl>          <dbl> <dbl> <dbl>     <dbl>     <dbl> <lgl>
## 1 Klf2      762.          -4.02 0.122 -32.9 3.55e-237 2.21e-233 TRUE
## # ... with 2 more variables: bound_in <chr>, signed_log10_q <dbl>
```