ELT2_repression_microscopy

Rtpw

8/4/2022

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## Warning: package 'tidyr' was built under R version 4.1.2
```

```
## Warning: package 'readr' was built under R version 4.1.2
```

```
## Warning: package 'dplyr' was built under R version 4.1.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
input_df <- read_csv("../01_input/ELT2_Repression_Microscopy_All_Data.csv") %>% mutate(gene = fct_relev
                                                                                       RNAi = fct_relevel(RNAi, c
  separate(Label, sep = ":", into = c("Label", "type", "channel"))
```

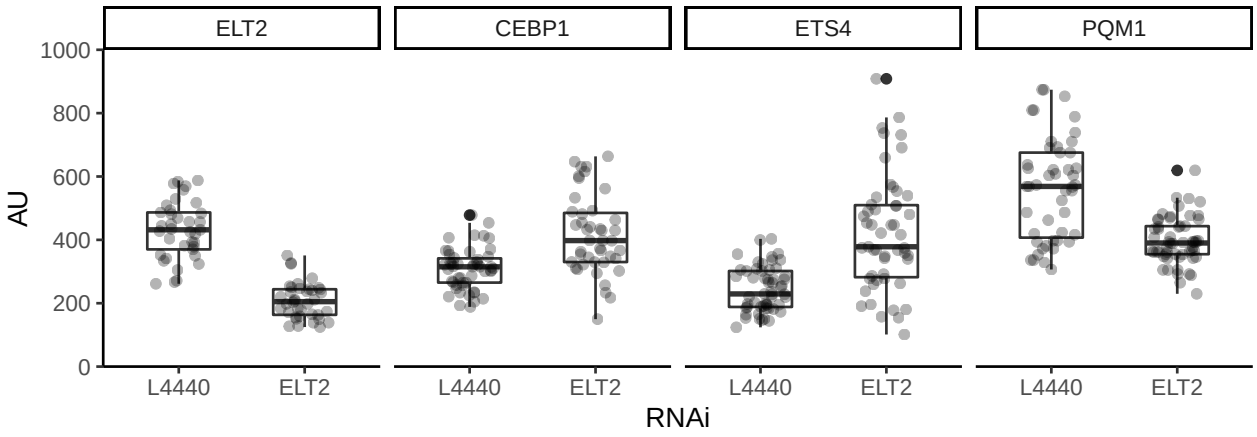
```
## Rows: 1886 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (4): Label, gene, type, RNAi
## dbl (8): val, Area, Mean, Min, Max, IntDen, RawIntDen, rep
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
analysis_df <- input_df %>%
  group_by(Label, type) %>%
  summarise(mean_background = mean(Mean)) %>%
  filter(type == "background") %>%
  select(-type) %>%
  right_join(input_df, by = "Label") %>%
  filter(type == "intestine") %>%
  mutate(AU = (IntDen - (mean_background*Area))/Area)
```

```
## `summarise()` has grouped output by 'Label'. You can override using the
## `.groups` argument.
```

```
gfp_plot <- analysis_df %>%
  # filter(!(gene == "ETS4" & rep == 4)) %>%
  ggplot(aes(x = RNAi, y = AU)) +
```

```
  # geom_violin() +
  geom_boxplot(width = 0.5) +
  geom_jitter(alpha = 0.3, width = 0.2, size = 2, aes(stroke = 0)) +
  # facet_grid(rep~gene) +
  facet_grid(. ~ gene) +
  # facet_grid(gene~.) + #, scales = "free") +
  scale_y_continuous(limits = c(0,1000), breaks = seq(0,1000, by = 200), expand = c(0,0))+
  # expand_limits(y = 0) +
  theme_classic()
  # theme(
  #   plot.title = element_text(
  #     face = "bold",
  #     size = rel(1.2),
  #     hjust = 0.5
  #   ),
  #   # text = element_text(family = "Arial"),
  #   # axis.title = element_text(
  #   #   face = "bold",
  #   #   size = rel(1),
  #   #   color = "black"
  #   # ),
  #   # axis.title.y = element_text(angle = 90, vjust = 2),
  #   # axis.title.x = element_text(vjust = -0.2),
  #   axis.text = element_text(colour = "black"),
  #   axis.line = element_line(colour = "black"),
  #   axis.ticks = element_line(),
  #   panel.background = element_rect(fill = "white",
  #           colour = NA),
  #   panel.grid.major = element_line(colour = "grey"),
  #   panel.grid.minor = element_line(colour = "grey"),
  #   panel.grid.major.x = element_blank(),
  #   strip.background = element_blank(),
  #   strip.text = element_text(face = "bold"),
  #   panel.border = element_rect(fill = NA,
  #           colour = "black"),
  #
  # )
gfp_plot
```



Save the plot

```
# ggsave(gfp_plot, filename = "../03_output/elt2_regulated_TFs_GFP_reporter_plot.pdf", width = 7, heigh
```

```
t.test(x = analysis_df %>% filter(gene == "ELT2", RNAi == "L4440") %>% pull(AU),
       y = analysis_df %>% filter(gene == "ELT2", RNAi == "ELT2") %>% pull(AU))$p.value
```

```
## [1] 1.210741e-18
```

```
t.test(x = analysis_df %>% filter(gene == "CEBP1", RNAi == "L4440") %>% pull(AU),
       y = analysis_df %>% filter(gene == "CEBP1", RNAi == "ELT2") %>% pull(AU))$p.value
```

```
## [1] 3.319637e-06
```

```
t.test(x = analysis_df %>% filter(gene == "ETS4", RNAi == "L4440") %>% pull(AU),
       y = analysis_df %>% filter(gene == "ETS4", RNAi == "ELT2") %>% pull(AU))$p.value
```

```
## [1] 1.331861e-08
```

```
t.test(x = analysis_df %>% filter(gene == "PQM1", RNAi == "L4440") %>% pull(AU),
       y = analysis_df %>% filter(gene == "PQM1", RNAi == "ELT2") %>% pull(AU))$p.value
```

```
## [1] 1.455525e-08
```