Analyze CRISPRi growth competition data for growth at high CO2 gas feeds, calculate adjusted p values between two conditions: compare to high-light data from Miao et al.

Ute Hoffmann (Science For Life Laboratory (KTH), Stockholm, Sweden)

november 04, 2024

Contents

1 Aim of the analysis

Calculation of Wilcoxon rank sum test between sgRNA fitness values between 4% CO2 data and 30% CO2 data.

2 Analysis

In a first step, the results given by the Nextflow pipeline are loaded.

```r
load("../results_controlsgRNAs/fitness/result.Rdata")
DESeq_result_table <- subset(DESeq_result_table, DESeq_result_table$condition=="CO2_30percent")
# adjust data set to be easily compatible with other data set
DESeq_result_table$group <- NULL
DESeq_result_table$reference_group <- NULL
DESeq_result_table$baseMean <- NULL
load("../input/CRISPRi_library_2022.Rdata")
HL_dataset <- subset(CRISPRi_library_2022, CRISPRi_library_2022$condition=="HC, HL")[,c(names(DESeq_res
HL_dataset$condition <- "HCHL"
DESeq_result_table <- bind_rows(DESeq_result_table, HL_dataset)
df_red_wide <- pivot_wider(unique(DESeq_result_table[,c("sgRNA_target", "condition", "wmean_fitness", "

df_red_wide$impact_score <- (df_red_wide$wmean_fitness_CO2_30percent - df_red_wide$wmean_fitness_HCHL)/

mapping_gene_locus <- read_tsv("../input/2023-02-24_mapping_trivial_names.tsv", show_col_types=FALSE)
names(mapping_gene_locus) <- c("sgRNA_target", "locus")
df_red_wide <- df_red_wide %>% left_join(mapping_gene_locus)

annotation <- read_tsv("../input/annotation_locusTags_stand13012021.csv", show_col_types = FALSE)
annotation_2 <- annotation[,c(1,2,3)]
names(annotation_2) <- c("locus", "Gene name","Product")
df_red_wide <- df_red_wide %>% left_join(annotation_2)
```

```r
write_tsv(df_red_wide[,c(1,7,8,9,2,3,4,5,6)], file="../R_results_controlssgRNAs/result_table_comparison_

plot_fitness_fitness <- function(df_input, y_axis, y_axis_label, x_axis, x_axis_label, filename_save){
  df_input$diff <- "NO"
  df_input$diff[(df_input[[x_axis]] - df_input[[y_axis]] > 2.5) | (df_input[[x_axis]] - df_input[[y_axi
  # prepare labels for plot
  df_input$delabel <- NA
  df_input$delabel[df_input$diff !="NO"] <- df_input$sgRNA_target[df_input$diff != "NO"]
  mycolors <- c("darkblue",  "#d3d3d3b2")
  names(mycolors) <- c("YES", "NO")
  p <- ggplot(data=df_input, aes(x=eval(parse(text=x_axis)), y=eval(parse(text=y_axis)), label=delabel,
    theme_light() + labs(y=y_axis_label, x=x_axis_label) + theme(legend.position = "none") + geom_abline
  ggsave(filename = filename_save, plot=p, width=12, height=12, units="cm")
return(p)
}

p <- plot_fitness_fitness(df_red_wide, "wmean_fitness_CO2_30percent", y_axis_label="Weighted mean fitnes
ggsave(filename="../R_results_controlssgRNAs/wfitnes_plot_30percent_HCHL_large.pdf", width=24, height=24
```

Session info

```
##  [9] fastmap_1.2.0      fansi_1.0.6       scales_1.3.0      textshaping_0.4.0
## [13] cli_3.6.3          rlang_1.1.4       crayon_1.5.3      bit64_4.5.2
## [17] munsell_0.5.1      withr_3.0.1       yaml_2.3.10       tools_4.4.1
## [21] parallel_4.4.1     tzdb_0.4.0        colorspace_2.1-1  vctrs_0.6.5
## [25] R6_2.5.1           lifecycle_1.0.4   bit_4.5.0         vroom_1.6.5
## [29] ragg_1.3.3         pkgconfig_2.0.3   pillar_1.9.0      gtable_0.3.5
## [33] glue_1.8.0         Rcpp_1.0.13       systemfonts_1.1.0 xfun_0.48
## [37] tidyselect_1.2.1   rstudioapi_0.17.0 farver_2.1.2      htmltools_0.5.8.1
## [41] labeling_0.4.3     rmarkdown_2.28    compiler_4.4.1
```