

Drp1_coloc_analysis_PMP.R

kelse

2022-06-01

```
# Extract Pearson's R values from ImageJ Output
# for DRP1-PMP70 colocalization experiments using patient-derived
# fibroblasts immunostained for DRP1, TOM20, and DAPI
# performed by Ollie of UK team
# Two replicates per sample: C1, C2, P1, P2, P3, and P4
# Collected in same experiment/data set
# C1: pediatric control
# C2: adult control
# P1: Drp1 G401S variant
# P2: Drp1 G363D variant
# P3: Drp1 L230dup variant
# P4: Drp1 R710G variant

# load libraries
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.0 --

## v ggplot2 3.3.3      v purrr  0.3.4
## v tibble  3.1.0      v dplyr  1.0.5
## v tidyr   1.1.3      v stringr 1.4.0
## v readr   1.4.0      v forcats 0.5.1

## Warning: package 'stringr' was built under R version 4.0.5

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

library(reshape2)

## Warning: package 'reshape2' was built under R version 4.0.5

##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
## smiths
```

```
library(stringr)
library(formattable)
```

```
## Warning: package 'formattable' was built under R version 4.0.5
```

```
library(RColorBrewer)
```

```
#Import tr1 data
```

```
C1 <- tibble(read.delim("#01coloc2Log_allz.txt"))
C2 <- tibble(read.delim("#05coloc2Log_allz.txt"))
P1 <- tibble(read.delim("#09coloc2Log_allz.txt"))
P2 <- tibble(read.delim("#13coloc2Log_allz.txt"))
P3 <- tibble(read.delim("#17coloc2Log_allz.txt"))
P4 <- tibble(read.delim("#21coloc2Log_allz.txt"))
```

```
#Import tr2 data
```

```
C1_tr2 <- tibble(read.delim("#45coloc2Log_allZ.txt"))
C2_tr2 <- tibble(read.delim("#41coloc2Log_allZ.txt"))
P1_tr2 <- tibble(read.delim("#37coloc2Log_allz.txt"))
P2_tr2 <- tibble(read.delim("#33coloc2Log_allZ.txt"))
P3_tr2 <- tibble(read.delim("#29coloc2Log_allZ.txt"))
P4_tr2 <- tibble(read.delim("#25coloc2Log_allz.txt"))
```

```
# Change column names to aid in data extraction
```

```
colnames(C1) = c("X1")
colnames(C2) = c("X1")
colnames(P1) = c("X1")
colnames(P2) = c("X1")
colnames(P3) = c("X1")
colnames(P4) = c("X1")
```

```
colnames(C1_tr2) = c("X1")
```

```
colnames(C2_tr2) = c("X1")
```

```
colnames(P1_tr2) = c("X1")
```

```
colnames(P2_tr2) = c("X1")
```

```
colnames(P3_tr2) = c("X1")
```

```
colnames(P4_tr2) = c("X1")
```

```
#Extract values from C1
```

```
# Need to loop code for all samples together to reduce repetitiveness
```

```
# but this will work for now
```

```
C1_filtered <- C1 %>%
  filter(!str_detect(X1, "^Warning!|^RESULTS|^!!!"))
```

```
C1_filtered <- separate(C1_filtered, col=X1, into=c('variable', 'value'), sep=',')
```

```
C1_names <- C1_filtered %>%
```

```
filter(., variable == "Coloc_Job_Name")
```

```
colnames(C1_names) = c("variable", "Name")
```

```
C1_Pearson <- C1_filtered %>%
```

```
filter(., variable == "Pearson's R value (no threshold)")
```

```

colnames(C1_Pearson) = c("variable", "Pearson_no_threshold")

C1_merge <- bind_cols(C1_names, C1_Pearson)

## New names:
## * variable -> variable...1
## * variable -> variable...3

C1_merge_corr = C1_merge[,!grepl("^variable",names(C1_merge))]

C1_merge_corr <- C1_merge_corr %>%
  mutate(sample = "C1") %>%
  mutate(tr = 1)

write_csv(C1_merge_corr, "Pearson_values_C1.csv")

#Extract values from C2
C2_filtered <- C2 %>%
  filter(!str_detect(X1, "^Warning!|^RESULTS|^!!!"))

C2_filtered <- separate(C2_filtered, col=X1, into=c('variable', 'value'), sep=',')

C2_names <- C2_filtered %>%
  filter(., variable == "Coloc_Job_Name")
colnames(C2_names) = c("variable", "Name")

C2_Pearson <- C2_filtered %>%
  filter(., variable == "Pearson's R value (no threshold)")
colnames(C2_Pearson) = c("variable", "Pearson_no_threshold")

C2_merge <- bind_cols(C2_names, C2_Pearson)

## New names:
## * variable -> variable...1
## * variable -> variable...3

C2_merge_corr = C2_merge[,!grepl("^variable",names(C2_merge))]

C2_merge_corr <- C2_merge_corr %>%
  mutate(sample = "C2") %>%
  mutate(tr = 1)

write_csv(C2_merge_corr, "Pearson_values_C2.csv")

#Extract values from P1

P1_filtered <- P1 %>%
  filter(!str_detect(X1, "^Warning!|^RESULTS|^!!!"))

P1_filtered <- separate(P1_filtered, col=X1, into=c('variable', 'value'), sep=',')

P1_names <- P1_filtered %>%

```

```

    filter(., variable == "Coloc_Job_Name")
colnames(P1_names) = c("variable", "Name")

P1_Pearson <- P1_filtered %>%
  filter(., variable == "Pearson's R value (no threshold)")
colnames(P1_Pearson) = c("variable", "Pearson_no_threshold")

P1_merge <- bind_cols(P1_names, P1_Pearson)

## New names:
## * variable -> variable...1
## * variable -> variable...3

P1_merge_corr = P1_merge[,!grepl("^variable",names(P1_merge))]

P1_merge_corr <- P1_merge_corr %>%
  mutate(sample = "P1") %>%
  mutate(tr = 1)

write_csv(P1_merge_corr, "Pearson_values_P1.csv")

#Extract values from P2

P2_filtered <- P2 %>%
  filter(!str_detect(X1, "^Warning!|^RESULTS|^!!!"))

P2_filtered <- separate(P2_filtered, col=X1, into=c('variable', 'value'), sep=',')

P2_names <- P2_filtered %>%
  filter(., variable == "Coloc_Job_Name")
colnames(P2_names) = c("variable", "Name")

P2_Pearson <- P2_filtered %>%
  filter(., variable == "Pearson's R value (no threshold)")
colnames(P2_Pearson) = c("variable", "Pearson_no_threshold")

P2_merge <- bind_cols(P2_names, P2_Pearson)

## New names:
## * variable -> variable...1
## * variable -> variable...3

P2_merge_corr = P2_merge[,!grepl("^variable",names(P2_merge))]

P2_merge_corr <- P2_merge_corr %>%
  mutate(sample = "P2") %>%
  mutate(tr = 1)

write_csv(P2_merge_corr, "Pearson_values_P2.csv")

#Extract values from P3

```

```

P3_filtered <- P3 %>%
  filter(!str_detect(X1, "^Warning!|^RESULTS|^!!!"))

P3_filtered <- separate(P3_filtered, col=X1, into=c('variable', 'value'), sep=',')

P3_names <- P3_filtered %>%
  filter(., variable == "Coloc_Job_Name")
colnames(P3_names) = c("variable", "Name")

P3_Pearson <- P3_filtered %>%
  filter(., variable == "Pearson's R value (no threshold)")
colnames(P3_Pearson) = c("variable", "Pearson_no_threshold")

P3_merge <- bind_cols(P3_names, P3_Pearson)

```

```

## New names:
## * variable -> variable...1
## * variable -> variable...3

```

```

P3_merge_corr = P3_merge[,!grepl("^variable",names(P3_merge))]

P3_merge_corr <- P3_merge_corr %>%
  mutate(sample = "P3") %>%
  mutate(tr = 1)

write_csv(P3_merge_corr, "Pearson_values_P3.csv")

#Extract values from P4

P4_filtered <- P4 %>%
  filter(!str_detect(X1, "^Warning!|^RESULTS|^!!!"))

P4_filtered <- separate(P4_filtered, col=X1, into=c('variable', 'value'), sep=',')

P4_names <- P4_filtered %>%
  filter(., variable == "Coloc_Job_Name")
colnames(P4_names) = c("variable", "Name")

P4_Pearson <- P4_filtered %>%
  filter(., variable == "Pearson's R value (no threshold)")
colnames(P4_Pearson) = c("variable", "Pearson_no_threshold")

P4_merge <- bind_cols(P4_names, P4_Pearson)

```

```

## New names:
## * variable -> variable...1
## * variable -> variable...3

```

```

P4_merge_corr = P4_merge[,!grepl("^variable",names(P4_merge))]

P4_merge_corr <- P4_merge_corr %>%
  mutate(sample = "P4") %>%

```

```

mutate(tr = 1)

write_csv(P4_merge_corr, "Pearson_values_P4.csv")

#Extract values from C1 - set2 (tr2)

C1_tr2_filtered <- C1_tr2 %>%
  filter(!str_detect(X1, "^Warning!|^RESULTS|^!!!"))

C1_tr2_filtered <- separate(C1_tr2_filtered, col=X1, into=c('variable', 'value'), sep=',')

C1_tr2_names <- C1_tr2_filtered %>%
  filter(., variable == "Coloc_Job_Name")
colnames(C1_tr2_names) = c("variable", "Name")

C1_tr2_Pearson <- C1_tr2_filtered %>%
  filter(., variable == "Pearson's R value (no threshold)")
colnames(C1_tr2_Pearson) = c("variable", "Pearson_no_threshold")

C1_tr2_merge <- bind_cols(C1_tr2_names, C1_tr2_Pearson)

## New names:
## * variable -> variable...1
## * variable -> variable...3

C1_tr2_merge_corr = C1_tr2_merge[,!grepl("^variable",names(C1_tr2_merge))]

C1_tr2_merge_corr <- C1_tr2_merge_corr %>%
  mutate(sample = "C1") %>%
  mutate(tr = 2)

write_csv(C1_tr2_merge_corr, "Pearson_values_C1_tr2.csv")

#Extract values from C2

C2_tr2_filtered <- C2_tr2 %>%
  filter(!str_detect(X1, "^Warning!|^RESULTS|^!!!"))

C2_tr2_filtered <- separate(C2_tr2_filtered, col=X1, into=c('variable', 'value'), sep=',')

C2_tr2_names <- C2_tr2_filtered %>%
  filter(., variable == "Coloc_Job_Name")
colnames(C2_tr2_names) = c("variable", "Name")

C2_tr2_Pearson <- C2_tr2_filtered %>%
  filter(., variable == "Pearson's R value (no threshold)")
colnames(C2_tr2_Pearson) = c("variable", "Pearson_no_threshold")

C2_tr2_merge <- bind_cols(C2_tr2_names, C2_tr2_Pearson)

## New names:
## * variable -> variable...1
## * variable -> variable...3

```

```

C2_tr2_merge_corr = C2_tr2_merge[,!grepl("^variable",names(C2_tr2_merge))]

C2_tr2_merge_corr <- C2_tr2_merge_corr %>%
  mutate(sample = "C2") %>%
  mutate(tr = 2)

write_csv(C2_tr2_merge_corr, "Pearson_values_C2tr2_.csv")

#Extract values from P1

P1_tr2_filtered <- P1_tr2 %>%
  filter(!str_detect(X1, "^Warning!|^RESULTS|^!!!"))

P1_tr2_filtered <- separate(P1_tr2_filtered, col=X1, into=c('variable', 'value'), sep=',')

P1_tr2_names <- P1_tr2_filtered %>%
  filter(., variable == "Coloc_Job_Name")
colnames(P1_tr2_names) = c("variable", "Name")

P1_tr2_Pearson <- P1_tr2_filtered %>%
  filter(., variable == "Pearson's R value (no threshold)")
colnames(P1_tr2_Pearson) = c("variable", "Pearson_no_threshold")

P1_tr2_merge <- bind_cols(P1_tr2_names, P1_tr2_Pearson)

## New names:
## * variable -> variable...1
## * variable -> variable...3

P1_tr2_merge_corr = P1_tr2_merge[,!grepl("^variable",names(P1_tr2_merge))]

P1_tr2_merge_corr <- P1_tr2_merge_corr %>%
  mutate(sample = "P1") %>%
  mutate(tr = 2)

write_csv(P1_tr2_merge_corr, "Pearson_values_P1_tr2.csv")

#Extract values from P2

P2_tr2_filtered <- P2_tr2 %>%
  filter(!str_detect(X1, "^Warning!|^RESULTS|^!!!"))

P2_tr2_filtered <- separate(P2_tr2_filtered, col=X1, into=c('variable', 'value'), sep=',')

P2_tr2_names <- P2_tr2_filtered %>%
  filter(., variable == "Coloc_Job_Name")
colnames(P2_tr2_names) = c("variable", "Name")

P2_tr2_Pearson <- P2_tr2_filtered %>%
  filter(., variable == "Pearson's R value (no threshold)")
colnames(P2_tr2_Pearson) = c("variable", "Pearson_no_threshold")

P2_tr2_merge <- bind_cols(P2_tr2_names, P2_tr2_Pearson)

```

```

## New names:
## * variable -> variable...1
## * variable -> variable...3

P2_tr2_merge_corr = P2_tr2_merge[,!grepl("^variable",names(P2_tr2_merge))]

P2_tr2_merge_corr <- P2_tr2_merge_corr %>%
  mutate(sample = "P2") %>%
  mutate(tr = 2)

write_csv(P2_tr2_merge_corr, "Pearson_values_P2_tr2.csv")

#Extract values from P3

P3_tr2_filtered <- P3_tr2 %>%
  filter(!str_detect(X1, "^Warning!!^RESULTS|^!!!"))

P3_tr2_filtered <- separate(P3_tr2_filtered, col=X1, into=c('variable', 'value'), sep=',')

P3_tr2_names <- P3_tr2_filtered %>%
  filter(., variable == "Coloc_Job_Name")
colnames(P3_tr2_names) = c("variable", "Name")

P3_tr2_Pearson <- P3_tr2_filtered %>%
  filter(., variable == "Pearson's R value (no threshold)")
colnames(P3_tr2_Pearson) = c("variable", "Pearson_no_threshold")

P3_tr2_merge <- bind_cols(P3_tr2_names, P3_tr2_Pearson)

## New names:
## * variable -> variable...1
## * variable -> variable...3

P3_tr2_merge_corr = P3_tr2_merge[,!grepl("^variable",names(P3_tr2_merge))]

P3_tr2_merge_corr <- P3_tr2_merge_corr %>%
  mutate(sample = "P3") %>%
  mutate(tr = 2)

write_csv(P3_tr2_merge_corr, "Pearson_values_P3_tr2.csv")

#Extract values from P4

P4_tr2_filtered <- P4_tr2 %>%
  filter(!str_detect(X1, "^Warning!!^RESULTS|^!!!"))

P4_tr2_filtered <- separate(P4_tr2_filtered, col=X1, into=c('variable', 'value'), sep=',')

P4_tr2_names <- P4_tr2_filtered %>%
  filter(., variable == "Coloc_Job_Name")
colnames(P4_tr2_names) = c("variable", "Name")

P4_tr2_Pearson <- P4_tr2_filtered %>%

```



```

  filter(., variable == "Pearson's R value (no threshold)")
colnames(P4_tr2_Pearson) = c("variable", "Pearson_no_threshold")

P4_tr2_merge <- bind_cols(P4_tr2_names, P4_tr2_Pearson)

## New names:
## * variable -> variable...1
## * variable -> variable...3

P4_tr2_merge_corr = P4_tr2_merge[,!grepl("^variable",names(P4_tr2_merge))]

P4_tr2_merge_corr <- P4_tr2_merge_corr %>%
  mutate(sample = "P4") %>%
  mutate(tr = 2)

write_csv(P4_tr2_merge_corr, "Pearson_values_P4_tr2.csv")

#Merge all data sets

merge_all <- bind_rows(C1_merge_corr, C2_merge_corr, P1_merge_corr, P2_merge_corr,
                      P3_merge_corr, P4_merge_corr, C1_tr2_merge_corr, C2_tr2_merge_corr,
                      P1_tr2_merge_corr, P2_tr2_merge_corr, P3_tr2_merge_corr, P4_tr2_merge_corr)

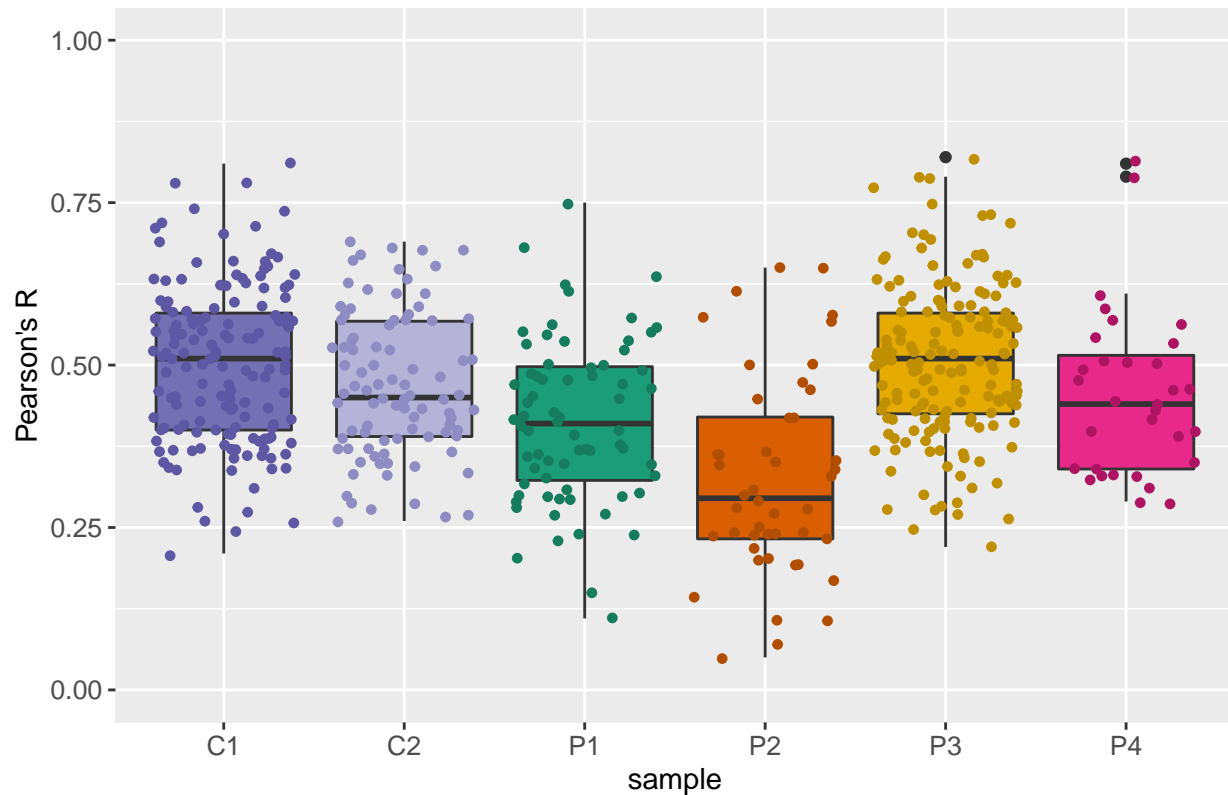
merge_all_no_outliers <- merge_all %>%
  filter(., as.numeric(Pearson_no_threshold) > 0.3)

write_csv(merge_all, "merge_all_Pearson_values.csv")
write_csv(merge_all_no_outliers, "merge_all_Pearson_values_no_outlier.csv")

merge_all %>%
  ggplot(aes(x = sample, y = as.numeric(Pearson_no_threshold), fill = sample)) +
  geom_boxplot() +
  geom_jitter(size = 2, shape = 20, aes(color = sample)) +
  scale_fill_manual(values = c("C1" = "#7570B3", "C2" = "#b5b3d7",
                              "P1" = "#1B9E77", "P2" = "#D95F02",
                              "P3" = "#E6AB02", "P4" = "#E7298A")) +
  scale_color_manual(values = c("C1" = "#5d57a4", "C2" = "#8f8cc3",
                              "P1" = "#157c5e", "P2" = "#b24e02",
                              "P3" = "#c08f02", "P4" = "#ae1462")) +
  scale_y_continuous(limits = c(0, 1.0), breaks = c(0, 0.25, 0.5, 0.75, 1.0)) +
  labs(title = "Drp1 vs. PMP colocalization", x = "sample",
       y = "Pearson's R") +
  theme(plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(size = 10),
        axis.text.y = element_text(size = 10),
        legend.position = "none")

```

Drp1 vs. PMP colocalization



```
ggsave("Pearson_no_outliers.pdf", height = 20, width = 30, dpi = 300, units = "cm")
ggsave("Pearson_no_outliers.png", height = 20, width = 30, dpi = 300, units = "cm")
```

```
anova_merge_all <- aov(Pearson_no_threshold ~ sample, data = merge_all)
summary(anova_merge_all)
```

```
##           Df Sum Sq Mean Sq F value Pr(>F)
## sample      5  1.540  0.30790   20.29 <2e-16 ***
## Residuals 546  8.286  0.01518
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
tukey_merge <- TukeyHSD(anova_merge_all)
print(tukey_merge)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = Pearson_no_threshold ~ sample, data = merge_all)
##
## $sample
##           diff            lwr            upr      p adj
## C2-C1 -0.030711553 -0.077627037  0.0162039315 0.4205454
## P1-C1 -0.086648061 -0.137590068 -0.0357060534 0.0000222
```

```
## P2-C1 -0.174716384 -0.234047584 -0.1153851834 0.0000000
## P3-C1 0.005833096 -0.033959288 0.0456254802 0.9983447
## P4-C1 -0.045246275 -0.113807646 0.0233150958 0.4110403
## P1-C2 -0.055936508 -0.112080996 0.0002079801 0.0514936
## P2-C2 -0.144004831 -0.207858623 -0.0801510390 0.0000000
## P3-C2 0.036544649 -0.009721421 0.0828107192 0.2129432
## P4-C2 -0.014534722 -0.087045252 0.0579758080 0.9927244
## P2-P1 -0.088068323 -0.154936344 -0.0212003023 0.0025171
## P3-P1 0.092481157 0.042136597 0.1428257166 0.0000032
## P4-P1 0.041401786 -0.033776685 0.1165802563 0.6154774
## P3-P2 0.180549480 0.121730453 0.2393685066 0.0000000
## P4-P2 0.129470109 0.048371985 0.2105682322 0.0000898
## P4-P3 -0.051079371 -0.119198004 0.0170392618 0.2660733
```

```
tukey_merge_all <- TukeyHSD(anova_merge_all)
summary(tukey_merge_all)
```

```
##           Length Class  Mode
## sample 60      -none- numeric
```

```
TK_results <- tukey_merge_all
TK_results<-as.data.frame(TK_results[1])
write_csv(TK_results, "post_hoc_tukey.csv")
```