

# Script used for analyzing and graphing flow data

## Contents

### 0.0.0.0.1 This script will go through an excel file containing all the DVs of interest and:

1. Conduct t-tests between the experimental conditions vs medium control for all the DVs
2. Output these results into an excel file
3. Plot all the data with statistical significance indicated asterisks

```
# Built on R version 4.2.2
# library
packages <- c('readxl', 'Rmisc', 'ggplot2', 'lsmeans', 'patchwork', 'dplyr', 'tidyverse',
              'ggpval', 'openxlsx', 'knitr')
invisible(lapply(packages, library, character.only=TRUE, quietly=TRUE))
```

```
# read in data
DATA <- read_excel("analysis_flow.xlsx", sheet = "d2_aav", col_names=TRUE)
kable(head(DATA))
```

### 0.0.0.0.2 The input file looks like this:

Condition	DonorGroup	Effector memory:	Effector memory:	Effector memory:	Effector memory:
		CD45RO+ / CD62L- CD25+ CD25 mean	CD45RO+ / CD62L- % parent	CD45RO+ / CD62L- CD25+ % parent	CD45RO+ / CD62L- % gp
medium	DonorA	6439	76.8	9.59	55.2
medium	DonorA	6614	71.1	12.30	37.1
cond_	DonorA	6596	71.7	11.10	35.2
cond_	DonorA	6414	75.9	7.14	33.6
cond_	DonorA	7620	77.2	9.52	41.8
cond_	DonorA	5961	78.1	6.51	31.2

```
# name output file
file_name <- "d2_aav.xlsx"

# make variable that holds all DVs which will be analyzed
numeric_cols <- sapply(DATA, is.numeric)
dep_vars <- names(DATA)[numeric_cols]
exclude_vars <- c("file", "Donor", "Condition", "Group")
DVs <- dep_vars[!dep_vars %in% exclude_vars]
```

```

# make list to store plots
plot_list <- list()
# make list to store t-tests
ttest_list <- list()

# iterate over each DV to make plots & ttests
# just display the first 4 plots
for (DV in head(DVs, 12)) {
  # Calculate mean and SEM for each Condition
  DATA_Summary <- DATA %>%
    group_by(Condition) %>%
    summarize(
      mean_value = mean(!is.na(DV), na.rm = TRUE),
      sem_value = sd(!is.na(DV), na.rm = TRUE) / sqrt(sum(!is.na(DV))),
      # ymax will be used for * positioning
      ymax = max(!is.na(DV), na.rm = TRUE))

  # get max value for DV, regardless of condition (this will be used for y axis height)
  max_value <- max(DATA[[DV]], na.rm = TRUE)

  # do t-test per DV
  ttest <- pairwise.t.test(DATA[[DV]], DATA$Condition, p.adj = "none")
  pval_df <- as.data.frame(ttest$p.value)

  # add the words rows/columns to the rows/columns so that the long df is made properly
  rownames(pval_df) <- paste0(rownames(pval_df), '_row')
  colnames(pval_df) <- paste0(colnames(pval_df), '_column')
  pval_df_long <- pval_df %>%
    rownames_to_column(var = "row") %>%
    gather(key = "column", value = "pval", -row) %>%
    mutate(grp1 = sub("_column", "", column),
           grp2 = sub("_row", "", row)) %>%
    select(grp1, grp2, pval)
  # remove rows with NAs & save to ttest list
  pval_df_long_fortable <- pval_df_long %>%
    filter(!is.na(pval))
  ttest_list[[DV]] <- pval_df_long_fortable

  # remove NAs and just keep comparisons to condition M (this is used for graphing)
  pval_df_long_filtered <- pval_df_long %>%
    filter(!is.na(pval), grp2 %in% c("Medium"))

  # add significance info to Data summary
  DATA_Summary <- left_join(DATA_Summary, pval_df_long_filtered,
                             by = c("Condition" = "grp1"))
  # Rename the columns for clarity
  DATA_Summary <- DATA_Summary %>%
    rename(Condition2 = grp2, pvalue = pval)
  # add column to datasummary to indicate significance
  DATA_Summary <- DATA_Summary %>%
    mutate(significance = case_when(
      pvalue < 0.001 ~ "***",
      pvalue < 0.01 ~ "**",

```

```

    pvalue < 0.05 ~ "*",
    TRUE ~ ""
  ))

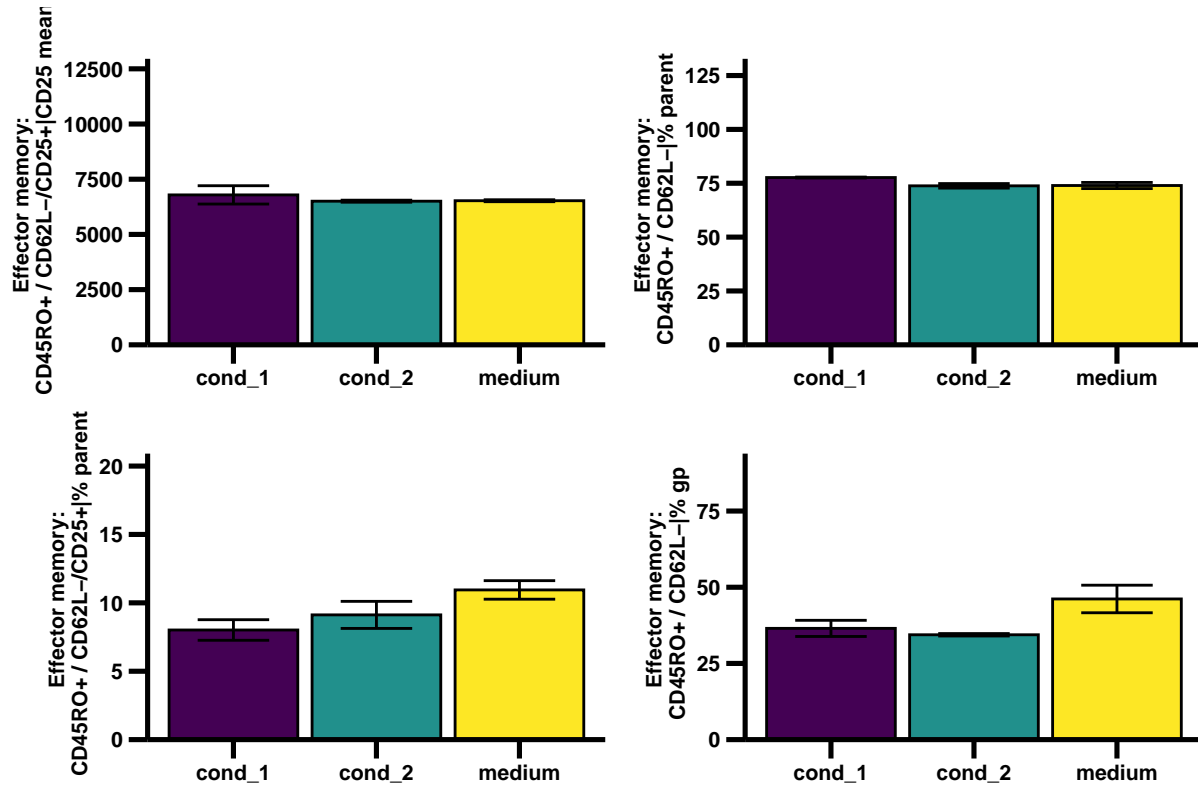
# create a bar graph for each DV with SEM bars
PLOT <- ggplot(DATA_Summary, aes(x = Condition, y = mean_value, fill = Condition)) +
  geom_bar(stat = "identity", position = position_dodge(), color = 'black', linewidth = 0.5) +
  # 3 colors
  scale_fill_manual(values = viridisLite::viridis(3)[c(1:3, 1)]) +
  geom_errorbar(aes(ymin = mean_value - sem_value/2, ymax = mean_value + sem_value/2),
    position = position_dodge(0.7), width = 0.5, lwd = 0.5) +
  geom_point(data = DATA, aes(x = Condition, y = !!sym(DV), shape = factor(Donor), fill = Condition),
    color = "black", size = 1, stroke = 1,
  ) +
  # set shapes for geom_point
  scale_shape_manual(values = c("1" = 21, "2" = 22)) + # 21 is for circle, 22 is for square
  geom_text(data = DATA_Summary, aes(x = Condition, y = ymax * 1.47, label = significance,
    fontface = "bold"), color = "black", size = 5,
    position = position_dodge(0.7), vjust = 0.8,
    angle = 270) +
  theme_classic() +
  # y axis label ; make a new line after the colon, and remove column name
  labs(y = gsub(":", ": \n", sub("\\.{3}\\d+", "", DV)), x = '') +
  theme(
    axis.title.y = element_text(face = 'bold', size = 8, color = 'black'),
    axis.text.x = element_text(size = 8, color = 'black', face = 'bold'),
    axis.text.y = element_text(size = 8, color = 'black', face = 'bold')) +
  # change axis line and tick thickness and size
  theme(axis.line = element_line(linewidth = 0.9),
    axis.ticks = element_line(colour = "black", linewidth = 0.9),
    axis.ticks.length = unit(0.25, "cm")) +
  # change legend attributes
  guides(fill = guide_legend(title = "Condition"),
    shape = guide_legend(title = "Donor")) +
  theme(legend.position = c(100, 100)) +
  theme(legend.title = element_text(size = 12)) +
  theme(legend.text = element_text(size = 12)) +
  # put legend title in the center
  theme(legend.title.align = 0.5) +
  # legend box
  theme(legend.background = element_rect(colour = 'black', fill = 'white',
    linetype = 'solid')) +
  # change size of graph
  theme(aspect.ratio = 1/1.6)

PLOT <- PLOT + # make bars touch x-axis, and specify breaks between axis lines
  scale_y_continuous(expand = c(0, 0)) +
  coord_cartesian(ylim = c(0, max_value * 1.7))

# Add the plot to list
plot_list[[length(plot_list) + 1]] <- PLOT
}

```

```
##### Combine the plots and display them
combined_plots <- patchwork::wrap_plots(plot_list, ncol = 2)
print(combined_plots)
```



```
##### save the t test output
combined_ttests <- bind_rows(ttest_list, .id = "Source")
# add significance column
combined_ttests <- combined_ttests %>%
  mutate(significance = case_when(
    pval < 0.001 ~ "***",
    pval < 0.01 ~ "**",
    pval < 0.05 ~ "*",
    TRUE ~ ""))
write.xlsx(combined_ttests, file = file_name, rowNames = FALSE)
#####
#####
#####
```

0.0.0.0.3 The excel output looks like this:

Source	grp1	grp2	pval	significance
Effector memory: CD45RO+ / CD62L-/CD25+ CD25 mean	cond_1	cond_2	0.7048929	

Source	grp1	grp2	pval	significance
Effector memory: CD45RO+ / CD62L-/CD25+ CD25 mean	cond_1	medium	0.7256739	
Effector memory: CD45RO+ / CD62L-/CD25+ CD25 mean	cond_2	medium	0.9769352	
Effector memory: CD45RO+ / CD62L- % parent	cond_1	cond_2	0.2781312	
Effector memory: CD45RO+ / CD62L- % parent	cond_1	medium	0.2936930	
Effector memory: CD45RO+ / CD62L- % parent	cond_2	medium	0.9621792	