Downstream Functional Analysis of Perseus-Processed Proteomics Data

Andrés Méndez Hurtado

Data Import and LFQ Matrix Preparation

The protein quantification data generated with Perseus was first imported and processed to extract the LFQ (Label-Free Quantification) matrix required for downstream analyses. Column names were adjusted to comply with the input format required by the AMICA platform.

```
# Load the dataset exported from Perseus
datos <- read.table(file = "perseus.txt", header = TRUE, sep = "\t")

# Remove default column names to assign headers manually
colnames(datos) <- NULL

# Extract actual column names from the first row
nombresCol <- datos[1,]

# Remove the first row containing the original headers and retain only data
datos_bien <- datos[-1,]

# Select columns corresponding to LFQ intensities and standardize their names
LFQ_names <- nombresCol[45:60]
LFQ_names_bien <- paste("LFQ intensity", LFQ_names)
nombresCol[45:60] <- LFQ_names_bien

# Apply corrected column names to the dataset
colnames(datos_bien) <- nombresCol

# Convert appropriate columns to numeric values, replacing commas with periods
# if necessary

cols_a_modificar <- setdiff(7:ncol(datos_bien), c(10, 14))
datos_bien[, cols_a_modificar] <- lapply(datos_bien[, cols_a_modificar], function(x) {
  if (is.character(x) || is.factor(x)) {
    as.numeric(gsub(",", ".", as.character(x)))  # Replace commas and convert to numeric
  } else {
    x
  }
})
```

LFQ Intensity Normalization

LFQ intensities corresponding to day 1 samples were normalized relative to the average of the remaining days to correct for batch-specific shifts. Additionally, an anti-log2 transformation was applied to revert values to

linear scale, as required by the AMICA tool.

```
# Define columns corresponding to day 1 samples
dia1_cols <- c("LFQ intensity Control-1", "LFQ intensity Cwp2-1",
               "LFQ intensity Emp24-1", "LFQ intensity Gas1-1")

# Identify remaining LFQ columns (other time points)
otros_dias <- setdiff(colnames(datos_bien)[45:60], dia1_cols)

# Compute row-wise mean intensities for day 1 and for other days
media_dia1 <- rowMeans(datos_bien[, dia1_cols])
media_rest <- rowMeans(datos_bien[, otros_dias])

# Calculate shift required to center day 1 values with respect to the rest
desplazamiento <- media_dia1 - media_rest

# Apply normalization by subtracting the shift from day 1 columns
for (col in dia1_cols) {
  datos_bien[[col]] <- datos_bien[[col]] - desplazamiento
}

# Perform anti-log2 transformation to obtain linear intensity values (required by AMICA)
datos_bien[, 45:60] <- lapply(datos_bien[, 45:60], function(x) { 2^x })

# Duplicate LFQ intensity columns with new names ("Intensity") for AMICA compatibility
lfq_cols <- grep("^LFQ intensity ", colnames(datos_bien), value = TRUE)
for (col in lfq_cols) {
  new_col <- sub("^LFQ intensity", "Intensity", col)
  datos_bien[[new_col]] <- datos_bien[[col]]
}

# Export the normalized data matrix to a TSV file
write.table(datos_bien, file = "perseus.tsv", sep = "\t", quote = FALSE,
            row.names = FALSE)
```

Identification of Enriched and Exclusive Interactors

```
# Select significantly enriched interactors (log2FC  1 and FDR < 0.05)

cwp2_enriched <- datos_bien[datos_bien$`Significant (FDR=0.05; S0=0.1)` == "+" &
                            datos_bien$`Difference (log2)` >= 1, "Gene names"]
gas1_enriched <- datos_bien[datos_bien$`Significant (FDR=0.05; S0=0.1).2` == "+" &
                            datos_bien$`Difference (log2).2` >= 1, "Gene names"]
emp24_enriched <- datos_bien[datos_bien$`Significant (FDR=0.05; S0=0.1).1` == "+" &
                             datos_bien$`Difference (log2).1` >= 1, "Gene names"]

# Define a helper function to split protein complexes into individual protein
# names (required for STRING analysis)

separar_proteinas <- function(lista) {
  proteinas_separadas <- unlist(strsplit(lista, ";"))  # Split by semicolon and flatten
  return(proteinas_separadas)
```

```
}

# Apply the function to each list of enriched proteins

cwp2_enriched <- separar_proteinas(cwp2_enriched)
gas1_enriched <- separar_proteinas(gas1_enriched)
emp24_enriched <- separar_proteinas(emp24_enriched)

# Identify exclusive interactors for each protein by removing those shared with
# the other two groups

cwp2_exclusivas <- setdiff(cwp2_enriched, union(gas1_enriched, emp24_enriched))
gas1_exclusivas <- setdiff(gas1_enriched, union(cwp2_enriched, emp24_enriched))
emp24_exclusivas <- setdiff(emp24_enriched, union(cwp2_enriched, gas1_enriched))

# Export enriched and exclusive interactors to text files

write(cwp2_enriched, file = "cwp2_TOTALES.txt")
write(gas1_enriched, file = "gas1_TOTALES.txt")
write(emp24_enriched, file = "emp24_TOTALES.txt")

write(cwp2_exclusivas, file = "cwp2_exclusivas.txt")
write(gas1_exclusivas, file = "gas1_exclusivas.txt")
write(emp24_exclusivas, file = "emp24_exclusivas.txt")

# Display the number of unique enriched interactors for each bait

length(unique(cwp2_enriched))
```

```
## [1] 555
```

```
length(unique(gas1_enriched))
```

```
## [1] 373
```

```
length(unique(emp24_enriched))
```

```
## [1] 510
```

Identification of shared and partially shared interactors among Gas1, Cwp2, and Emp24

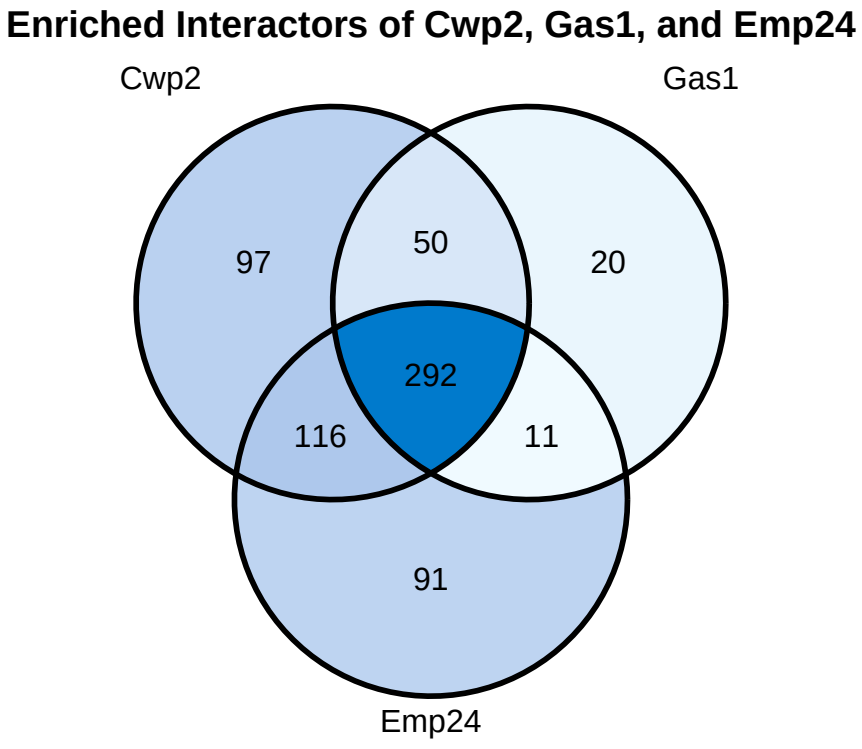
```
# 1. Interactors shared between Gas1 and Emp24, but not present in Cwp2

# Overlap between Gas1 and Emp24
gas1_emp24 <- intersect(gas1_enriched, emp24_enriched)

# Remove those also found in Cwp2
gas1_emp24_only <- setdiff(gas1_emp24, cwp2_enriched)
```

```r
# Combine Gas1-exclusive with shared-only-with-Emp24
gas1_total_31 <- union(gas1_exclusivas, gas1_emp24_only)

# Export shared-only-with-Emp24 interactors
write(gas1_emp24_only, file = "gas1andemp24.txt")


# 2. Interactors shared between Cwp2 and Emp24, but not present in Gas1

# Overlap between Cwp2 and Emp24
cwp2_emp24 <- intersect(cwp2_enriched, emp24_enriched)

# Remove those also found in Gas1
cwp2_emp24_only <- setdiff(cwp2_emp24, gas1_enriched)

# Combine Cwp2-exclusive with shared-only-with-Emp24
cwp2_total_213 <- union(cwp2_exclusivas, cwp2_emp24_only)

# Export shared-only-with-Emp24 interactors
write(cwp2_emp24_only, file = "cwp2andemp24.txt")


# 3. Interactors shared between Gas1 and Cwp2, but not present in Emp24

# Overlap between Gas1 and Cwp2
gas1_cwp2 <- intersect(gas1_enriched, cwp2_enriched)

# Remove those also found in Emp24
gas1_cwp2_only <- setdiff(gas1_cwp2, emp24_enriched)

# Export shared-only-with-Cwp2 interactors
write(gas1_cwp2_only, file = "gas1andcwp2.txt")


# 4. Interactors shared across all three bait proteins (Gas1, Cwp2, Emp24)

# Find common interactors across all three
shared_all_three <- Reduce(intersect, list(gas1_enriched, cwp2_enriched, emp24_enriched))

# Export interactors common to all three
write(shared_all_three, file = "gas1_cwp2_emp24_comunes.txt")
```

Venn Diagram of Enriched Interactors

```r
# Load required packages
library(ggVennDiagram)
library(ggplot2)

# Define a named list of enriched interactors for each bait protein
listas <- list(
  Cwp2 = cwp2_enriched,
```

```r
  Gas1 = gas1_enriched,
  Emp24 = emp24_enriched
)

# Generate Venn diagram showing the overlap of enriched proteins among the three baits
ggVennDiagram(listas, label_alpha = 0, label = "count", set_label_size = 4) +

  # Apply a color gradient for the overlapping areas
  scale_fill_gradient(low = "#f0faff", high = "#007acc") +

  # Remove background and axes for a cleaner appearance
  theme_void() +

  # Customize theme elements: title, margins, background
  theme(
    legend.position = "none",
    plot.title = element_text(size = 14, face = "bold", hjust = 0.5),
    plot.margin = margin(20, 20, 20, 20),
    plot.background = element_rect(fill = "white", color = NA),
    panel.background = element_rect(fill = "white", color = NA)
  ) +

  # Add a title to the plot
  labs(title = "Enriched Interactors of Cwp2, Gas1, and Emp24")
```



Comparison with SGD Physical Interactome Data

```
# Load physical interaction datasets from Saccharomyces Genome Database (SGD)
CWP2_SGD <- read.delim("CWP2_physical_interactions.txt",
                       header = TRUE, sep = "\t", quote = "")
CWP2_SGD_PROTS <- unique(CWP2_SGD$Interactor.1)

GAS1_SGD <- read.delim("GAS1_physical_interactions.txt",
                       header = TRUE, sep = "\t", quote = "")
GAS1_SGD_PROTS <- unique(GAS1_SGD$Interactor.1)

EMP24_SGD <- read.delim("EMP24_physical_interactions.txt",
                        header = TRUE, sep = "\t", quote = "")
EMP24_SGD_PROTS <- unique(EMP24_SGD$Interactor.1)

# Display number of unique interactors from SGD for each bait
length(CWP2_SGD_PROTS)
```

```
## [1] 25
```

```
length(GAS1_SGD_PROTS)
```

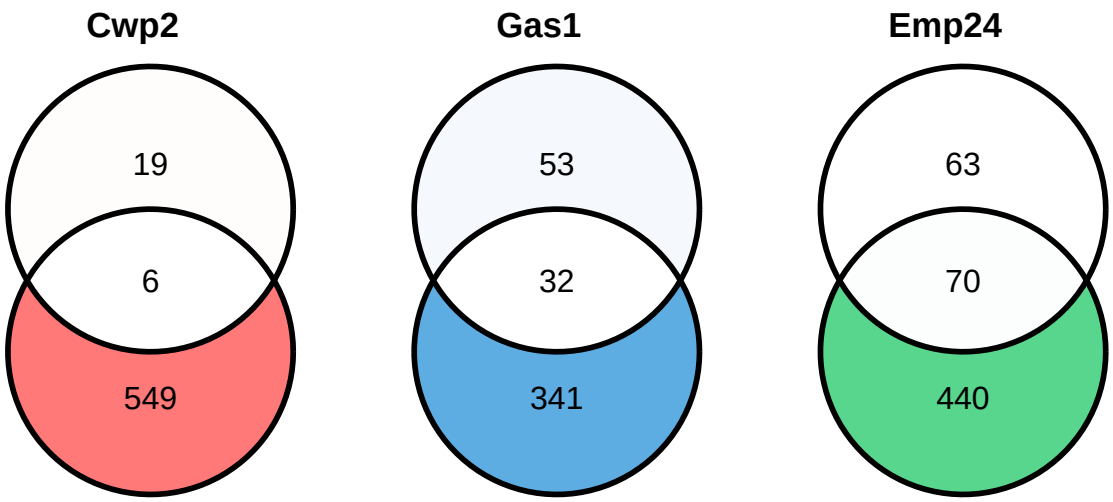
```
## [1] 85
```

```
length(EMP24_SGD_PROTS)
```

```
## [1] 133
```

```
# Load necessary libraries for plotting
library(ggplot2)
library(ggVennDiagram)
library(patchwork)

# Create Venn diagram: Cwp2 (experimental vs. SGD interactors)
p1 <- ggVennDiagram(
  list(" " = cwp2_enriched, " " = CWP2_SGD_PROTS),
  label_alpha = 0, label = "count", set_label_size = 4,
  set_name = NULL
) +
  scale_fill_gradient(low = "#ffffff", high = "#ff7979") +
  theme_void() +
  labs(subtitle = "Cwp2") +
  theme(
    plot.subtitle = element_text(size = 13, face = "bold", hjust = 0.5),
    plot.margin = margin(10, 10, 10, 10),
    legend.position = "none"
  )

# Venn diagram: Gas1 (experimental vs. SGD)
p2 <- ggVennDiagram(
```

```
  list(" " = gas1_enriched, " " = GAS1_SGD_PROTS),
  label_alpha = 0, label = "count", set_label_size = 4,
  set_name = NULL
) +
  scale_fill_gradient(low = "#ffffff", high = "#5dade2") +
  theme_void() +
  labs(subtitle = "Gas1") +
  theme(
    plot.subtitle = element_text(size = 13, face = "bold", hjust = 0.5),
    plot.margin = margin(10, 10, 10, 10),
    legend.position = "none"
  )

# Venn diagram: Emp24 (experimental vs. SGD)
p3 <- ggVennDiagram(
  list(" " = emp24_enriched, " " = EMP24_SGD_PROTS),
  label_alpha = 0, label = "count", set_label_size = 4,
  set_name = NULL
) +
  scale_fill_gradient(low = "#ffffff", high = "#58d68d") +
  theme_void() +
  labs(subtitle = "Emp24") +
  theme(
    plot.subtitle = element_text(size = 13, face = "bold", hjust = 0.5),
    plot.margin = margin(10, 10, 10, 10),
    legend.position = "none"
  )

# Combine all three Venn diagrams into a single figure
final_plot <- (p1 + p2 + p3) +
  plot_annotation(
    title = "Overlap between Experimental Enriched Interactors and SGD
    Physical Interactome",
    theme = theme(
      plot.title = element_text(size = 13, face = "bold", hjust = 0.5,
                                margin = margin(b = 18))
    )
  )

# Display and save the final plot
final_plot
```




```
ggsave("venn_comparison.png", plot = final_plot, width = 25, height = 5, dpi = 300)
```

Create Cytoscape-Compatible Table for ER-to-Golgi Transport Proteins Enriched in Cwp2

```
# Vector of Cytoscape-compatible IDs (shared name), in the same order as the gene list
shared_names <- c(
  "4932.YLR078C", "4932.YLR080W", "4932.YAR002C-A", "4932.YHR110W",
  "4932.YGL054C", "4932.YPL053C", "4932.YDR189W", "4932.YOR307C",
  "4932.YJL192C", "4932.YGL145W", "4932.YKR044W", "4932.YNL044W"
)

# Corresponding gene names
gene_names <- c(
  "BOS1", "EMP46", "ERP1", "ERP5", "ERV14",
  "KTR6", "SLY1", "SLY41", "SOP4", "TIP20",
  "UIP5", "YIP3"
)

# Match rows from the input data frame (datos_bien) based on gene names
filtered_data <- datos_bien[match(gene_names, datos_bien$`Gene names`), ]

# Create the output data frame with log2 fold changes and identifiers
```

```r
cytoscape_table <- data.frame(
  `shared name` = shared_names,
  Gene = gene_names,
  log2FC_CWP2 = filtered_data$`Difference (log2)`,
  log2FC_GAS1 = filtered_data$`Difference (log2).2`
)

# Calculate the ratio CWP2 / GAS1; handle Inf/NaN by capping to 1.25× max finite value
cytoscape_table$ratio_CWP2_GAS1 <-
  cytoscape_table$log2FC_CWP2 / cytoscape_table$log2FC_GAS1
max_ratio <- max(cytoscape_table$ratio_CWP2_GAS1[
  is.finite(cytoscape_table$ratio_CWP2_GAS1)], na.rm = TRUE)
cytoscape_table$ratio_CWP2_GAS1[
  !is.finite(cytoscape_table$ratio_CWP2_GAS1)] <- 1.25 * max_ratio

# Export the table to CSV for Cytoscape import
write.csv(cytoscape_table, file = "Cytoscape_ER_Golgi_CWP2_ratio.csv",
          row.names = FALSE, quote = FALSE)
```