PEC1_Marc_Rios_Cadenas

Marc Rios Cadenas

2025-04-02

Tabla de contenidos

Abstract 1
Objetivos 2
Métodos 2
Selección del *Dataset* 2
Creación del objeto *SummarizedExperiment* 2
Análisis Exploratorio 4
Resultados 6
Discusión 9
Conclusión 10
Referencias 10
Anexo 11

Abstract

Este estudio presenta un análisis exploratorio de datos de metabolómica, centrado en la caracterización de perfiles metabólicos en muestras de orina de pacientes con cáncer gástrico (GC), enfermedades gástricas benignas (BN) y personas sanas (HE). La metodología incluyó el preprocesamiento y normalización de datos para garantizar su calidad, seguido de pruebas estadísticas para evaluar diferencias metabólicas entre los grupos. Se aplicaron pruebas de normalidad, la prueba de Mann-Whitney U para comparar metabolitos y visualización mediante mapas de calor (*pheatmap*). Los resultados revelan diferencias metabólicas sutiles entre los grupos, lo que sugiere que la variabilidad metabólica es limitada y que podrían ser necesarias metodologías complementarias para detectar diferencias más marcadas. Estos hallazgos refuerzan la necesidad de validaciones experimentales y estudios adicionales para evaluar el potencial diagnóstico de estos perfiles metabólicos.

Objetivos

- Explorar las diferencias en los perfiles metabólicos de muestras de orina entre pacientes con cáncer gástrico, enfermedades gástricas benignas y controles sanos.
- Evaluar la distribución y normalidad de los datos para seleccionar pruebas estadísticas adecuadas.
- Aplicar técnicas de análisis estadístico y visualización para identificar diferencias significativas en la expresión de metabolitos entre los grupos.
- Discutir la relevancia de los hallazgos en el contexto de la detección temprana del cáncer gástrico y sus posibles implicaciones clínicas.

Métodos

La metodología seguida en este trabajo incluye tres etapas principales: la selección del *dataset*, la creación del objeto *SummarizedExperiment* para estructurar los datos y el análisis exploratorio para identificar patrones y diferencias entre grupos. El código completo utilizado en la metodología se encuentra disponible en el Anexo.

Selección del *Dataset*

El dataset seleccionado proviene del repositorio de GitHub de *metaboData* y corresponde a un estudio sobre el cáncer gástrico (GC). Fue originalmente publicado por Chan et al. (2016) [1] en el *British Journal of Cancer* y está disponible en el repositorio *Metabolomics Workbench* [2], lo que lo hace una fuente confiable.

Este conjunto de datos analiza perfiles metabólicos en muestras de orina de tres grupos: 43 pacientes con GC, 40 con enfermedades gástricas benignas (BN) y 40 personas sanas (HE). La información fue obtenida mediante espectroscopia de resonancia magnética nuclear (^{1}H-NMR), identificando 77 metabolitos. Se halló que tres metabolitos específicos (2-hydroxyisobutyrate, 3-indoxylsulfate y alanina) pueden discriminar entre GC y personas sanas con alta precisión (AUC = 0.95 en la curva ROC), sugiriendo su posible utilidad en la detección temprana del cáncer gástrico.

El *dataset* contiene datos procesados y anotados, organizados en una matriz con abundancias de metabolitos y metadatos clínicos (GC, BN, HE). Sin embargo, no incluye los espectros de RMN originales, lo que limita la verificación del procesamiento. A pesar de esto, su estructura es adecuada para análisis exploratorios en R con *SummarizedExperiment*, permitiendo la aplicación de métodos estadísticos como PCA y clustering.

Creación del objeto *SummarizedExperiment*

SummarizedExperiment y *ExpressionSet* son estructuras de datos de *Bioconductor* utilizadas para gestionar datos ómicos, pero presentan diferencias clave.

ExpressionSet, del paquete *Biobase*, ha sido el estándar en estudios de expresión génica, especialmente en microarrays, mientras que *SummarizedExperiment*, del paquete homónimo, es más flexible y se usa en transcriptómica, metabolómica y epigenómica. A nivel estructural, *ExpressionSet* almacena una única matriz de datos (*exprs()*), mientras que *SummarizedExperiment* permite múltiples matrices (*assays()*), facilitando el manejo de datos multi-ómicos. Además, la gestión de metadatos en *SummarizedExperiment* es más versátil, utilizando *colData()* y *rowData()* en lugar de *pData()* y *fData()*, lo que lo hace más adecuado para análisis modernos.

En este análisis se ha creado un objeto *SummarizedExperiment* a partir del *dataset* de metabolómica mediante la integración de tres componentes principales: la matriz de expresión de metabolitos, la información asociada a cada muestra y la anotación de los metabolitos.

En primer lugar, cargamos los datos desde un archivo *.xlsx*, dividiéndolos en dos estructuras: una tabla con las intensidades de los metabolitos en cada muestra (*data_sheet*) y otra con la información sobre los picos de metabolitos (*peak_sheet*). A partir de estos datos, extraemos la matriz de expresión asegurando que los metabolitos se organizaran en filas y las muestras en columnas. Lo hacemos seleccionando las columnas de intensidades de metabolitos mediante la función *select()* y luego transponiendo la matriz utilizando la función *t()* para ajustar la disposición de los datos.

Paralelamente, generamos un *dataframe* con la información de las muestras, que incluía su tipo (*QC o Sample*) y el lote experimental en el que fueron procesadas. Para asegurar que los nombres de las filas coincidieran con las muestras, eliminamos la columna de identificación redundante.

Asimismo, se prepara una tabla con las anotaciones de los metabolitos, asociando el nombre del metabolito correspondiente a cada variable de la matriz de expresión. Esto lo hacemos mediante la función *rename()* para cambiar el nombre de las columnas y la función *select()* para elegir los campos relevantes en la hoja de datos de los picos de metabolitos.

Una vez organizados estos elementos, se crea el objeto *se* mediante la función *SummarizedExperiment*(), integrando los datos de expresión, los metadatos de las muestras y la anotación de los metabolitos. La matriz de expresión se almacena en el *slot assays*, los datos de las muestras en *colData* y la información de los metabolitos en *rowData*.

Este objeto nos permite estructurar los datos de manera coherente para facilitar su análisis. Finalmente, se guarda en formato *".Rda*" para su uso posterior en análisis de metabolómica, asegurando la conservación de la estructura y la reproducibilidad del estudio.

Análisis Exploratorio

Distribución de intensidad de los Metabolitos por Tipo de Muestra

El análisis exploratorio de los datos empieza con la evaluación de la distribución de intensidades de los metabolitos según el tipo de muestra (QC o Sample). Para ello, reaorganizamos los datos en un formato largo (*long format*) utilizando la función *pivot_longer()* de la librería *tidyr*, permitiendo transformar las columnas de metabolitos en una única variable categórica (*Metabolite*), con sus respectivas intensidades en una columna separada (*Intensity*):

```
df_long <- data_sheet %>%
  select(Sample_Type, starts_with("M")) %>%
  pivot_longer(cols = starts_with("M"), names_to = "Metabolite",
values_to = "Intensity")
```

Esta transformación nos facilita la visualización de la distribución de las intensidades mediante *boxplots*, donde se compara la variabilidad de los metabolitos entre los grupos de muestras. Para mejorar la representación de los datos y evitar distorsiones debidas a valores extremos, se aplica una escala logarítmica en el eje *y* (*scale_y_log10()*) y se omiten los valores atípicos en los gráficos (*outlier.shape = NA*):

```
ggplot(df_long, aes(x = Sample_Type, y = Intensity, fill = Sample_Type))
+
  geom_boxplot(outlier.shape = NA) +
  scale_y_log10() +
  theme_minimal() +
  labs(title = "Distribución de Intensidad de Metabolitos por Tipo de
Muestra",
       x = "Tipo de Muestra", y = "Intensidad (log10)")
```

Adicionalmente, se realiza una verificación de la presencia de valores no finitos en la matriz de expresión del objeto *SummarizedExperiment*. Cuantificamos los valores NA, Inf y NaN para evaluar la calidad del *dataset* antes de su análisis estadístico.

Prueba de normalidad

Para evaluar la normalidad de la distribución de intensidades de los metabolitos en los distintos tipos de muestra (*QC* y *Sample*), realizamos una prueba de *Shapiro-Wilk*. Dado que el tamaño de los datos completos era potencialmente grande, se selecciona una muestra aleatoria de intensidades dentro de cada grupo, asegurando que el número de valores seleccionados no excediera 5000 o el total disponible en el grupo. Para garantizar la reproducibilidad del muestreo, se fija una semilla aleatoria con *set.seed(123)*.

Una vez extraídas las muestras representativas, se aplica la prueba de *Shapiro-Wilk* para evaluar la hipótesis de normalidad en la distribución de las intensidades dentro de cada grupo.

Para complementar esta evaluación, se generan gráficos de *QQ-plots* que permiten analizar visualmente el ajuste de la distribución de intensidades con respecto a la normal teórica. Se utiliza *stat_qq()* y *stat_qq_line()* en *ggplot2* para cada grupo de muestra, confirmando visualmente la desviación de la normalidad observada en la prueba estadística.

Prueba de Mann-Whitney U (para datos no normales)

La prueba de suma de rangos de *Wilcoxon* (*wilcox.test*) se aplica debido a que, en el análisis previo de normalidad, vemos que las muestras no siguen una distribución normal. Dado que las pruebas paramétricas requieren que los datos sean normales, se decide realizar esta prueba no paramétrica para comparar las intensidades (*Intensity*) entre los dos grupos definidos por la variable *Sample_Type*.

Para visualizar estos resultados, se genera un *boxplot* mediante *ggplot*, que muestra la distribución de las intensidades en ambos grupos, y se incluyen puntos dispersos *geom_jitter* para resaltar la variabilidad interna.

Por último, se calcula el tamaño del efecto utilizando el estadístico *r* de Wilcoxon, con la fórmula *abs(qnorm(0.00285) / sqrt(nrow(df_long)))*.

Pheatmap

En el último análisis, se genera un mapa de calor (*heatmap*) para visualizar los patrones de expresión de los metabolitos. La función *pheatmap* se utiliza para generar este gráfico a partir de la matriz de expresión, que contiene los datos de los metabolitos. La matriz se transpone utilizando *t(expr_matrix)* para que las filas representen las muestras y las columnas los metabolitos. Además, se añaden anotaciones de las columnas a partir de los datos de las muestras (*colData(se)*) para contextualizar mejor la información. El gráfico incluye agrupamiento tanto de filas como de columnas, lo que permite identificar relaciones entre los metabolitos y las muestras de manera más clara.

Antes de generar el *pheatmap*, se hace un tratamiento de los valores faltantes y no finitos en los datos. Para esto, se utiliza una función que reemplaza los valores *NA* y *Inf*

o *NaN* por la mediana de cada fila. Esto se realizó mediante la función *apply*, que asegura que cada fila de la matriz se procesa adecuadamente. Posteriormente, la matriz resultante se convierte nuevamente a formato numérico con *as.matrix(expr_matrix)* y se verifica que su tipo de dato sea numérico con *mode(expr_matrix) <- "numeric".*

Finalmente, se realizan pruebas para asegurar que no queden valores faltantes o no finitos en la matriz de expresión después del tratamiento. Se visualiza el número de valores *NA* y *Inf* o *NaN* restantes, que deberían ser cero si la limpieza ha sido buena. Esto garantiza que los datos estén listos para ser visualizados sin problemas de datos incompletos o inválidos.

Resultados

El objeto *SummarizedExperiment* creado a partir del dataset de metabolómica presenta una estructura adecuada para su análisis. Contiene 129 metabolitos como filas y 140 muestras como columnas, lo que indica que los datos de expresión han sido correctamente organizados. Además, la información adicional está bien incorporada: los metadatos de las muestras incluyen las variables *Sample_Type* y *Batch*, mientras que los metadatos de los metabolitos contienen la anotación correspondiente. Para verificar la integridad del objeto, se han comparado los nombres de filas y columnas entre la matriz de expresión y sus respectivos metadatos, confirmando que coinciden. Este objeto se encuentra listo para realizar análisis exploratorios como visualización de patrones, agrupaciones o identificación de biomarcadores, lo que permitirá evaluar las diferencias metabolómicas entre los distintos grupos de muestras.

El análisis exploratorio del *dataset* de metabolómica permitió examinar la distribución de la intensidad de los metabolitos en función del tipo de muestra.

Este *boxplot* (Fig. 1) generado con `ggplot2` nos muestra la distribución de las intensidades de los metabolitos en función del tipo de muestra (*QC vs. Sample*). Aunque ambas distribuciones presentan un comportamiento similar, se observa una leve diferencia en la dispersión y la mediana de los valores. Esto podría indicar cierta variabilidad entre los grupos, pero sin un análisis estadístico más detallado, no se puede concluir si estas diferencias son significativas.

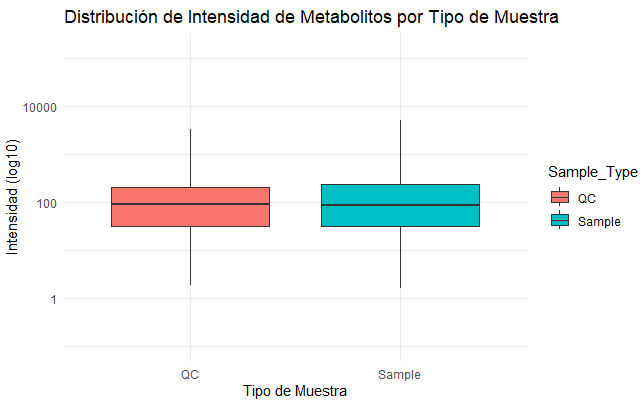

Figura 1. Distribución de la intensidad de metabolitos (log10) en muestras de calidad (QC) y experimentales (Sample).

Este análisis permite detectar las diferencias en la distribución de intensidades entre los grupos y evaluar la calidad de los datos, lo que es crucial para garantizar la validez de los resultados en los siguientes pasos del estudio. Además, se han encontrado un total de 915 valores no finitos, lo que sugiere la necesidad de estrategias de imputación o filtrado para evitar sesgos en los análisis posteriores.

Posteriormente, se realiza una prueba de normalidad para evaluar la distribución de los datos, lo que determinó que la mayoría de los metabolitos no seguían una distribución normal, lo que motivó el uso de métodos no paramétricos en los análisis posteriores (Fig. 2).

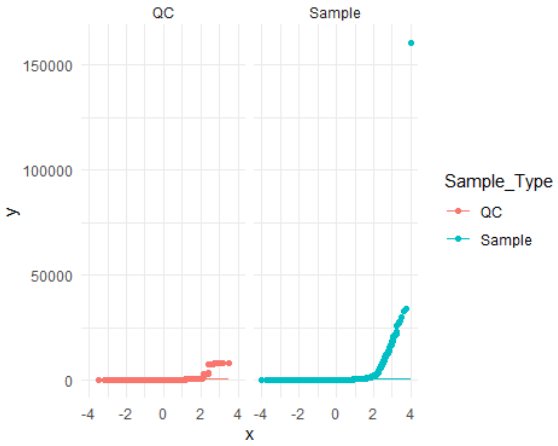

Figura 2. Prueba de Shapiro-Wilk para evaluar la normalidad en muestras de calidad (QC) y experimentales (Sample), evidenciando una fuerte asimetría en las muestras experimentales.

Los resultados de la prueba de *Shapiro-Wilk* nos indican que los datos no siguen una distribución normal ya que como los valores de W son muy bajos, esto sugiere una fuerte desviación de la normalidad. Además, los *p-values* (< 2.2e-16) son extremadamente pequeños y nos indica que la hipótesis nula de normalidad se rechaza con alta confianza.

Después de ver que los datos no siguen una distribución normal, se aplica la prueba de *Mann-Whitney U (Wilcoxon rank-sum)* para identificar diferencias significativas entre los grupos de muestras. La comparación entre los grupos QC y *Sample* muestra un *p-value* = 0.00285, indicando una diferencia estadísticamente significativa en las distribuciones (Fig. 3). Sin embargo, el tamaño del efecto es muy pequeño (r = 0.0206), lo que sugiere que, aunque existen diferencias, su impacto biológico real es mínimo. Por lo tanto, vemos que el tamaño del efecto (*r de Wilcoxon*) es muy pequeño, es decir, que, aunque la diferencia entre los grupos es estadísticamente significativa, puede que estas no sean biológicamente relevantes.

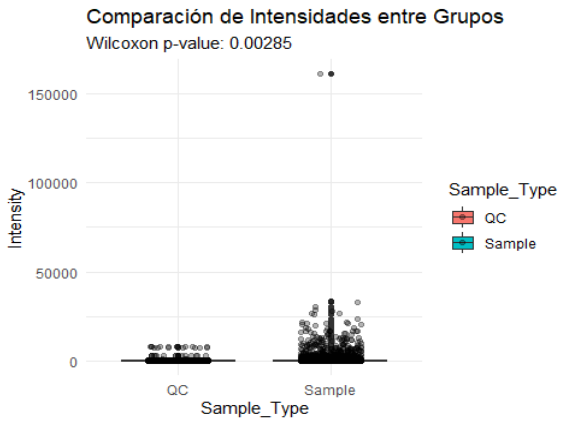

Figura 3. Comparación de intensidades entre muestras de calidad (QC) y experimentales (Sample) mediante la prueba de Wilcoxon, indicando una diferencia significativa entre los grupos (p = 0.00285).

Por último, para visualizar patrones globales en la expresión metabólica, se genera el *pheatmap* (Fig. 4), lo que permite identificar ciertos patrones de agrupación en los metabolitos. Sin embargo, en el mapa de calor generado con *pheatmap*, se observa una gran homogeneidad en la intensidad de los metabolitos entre las muestras, lo que nos sugiere que los valores están altamente uniformes, posiblemente debido a un rango de intensidades muy amplio que enmascara variaciones más sutiles. Además, la agrupación jerárquica de muestras no parece mostrar una clara separación entre grupos, lo que podría indicar que las diferencias metabólicas entre las condiciones analizadas no son marcadamente distinguibles a simple vista en este formato.

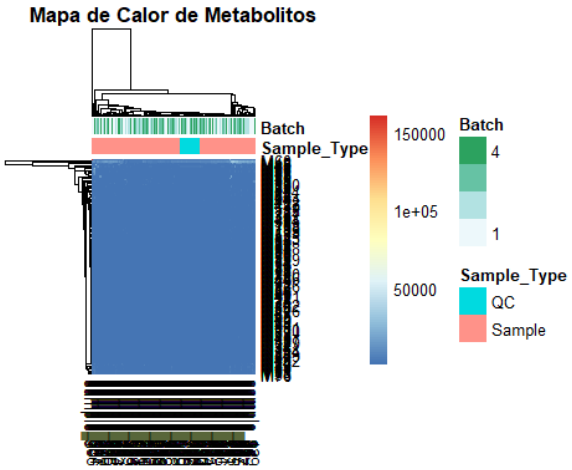

Figura 4. Pheatmap de metabolitos, agrupados por Batch y tipo de muestra, con una escala de intensidad de señal

En conjunto, estos análisis sugieren que, aunque hay diferencias estadísticamente significativas en la expresión de algunos metabolitos, estas son sutiles. Para comprender mejor su impacto biológico y su posible aplicación en la detección del cáncer gástrico, serían necesarios análisis adicionales, como modelos de clasificación o identificación de metabolitos clave con mayor relevancia diagnóstica.

Discusión

El análisis de los datos de metabolómica nos proporciona información valiosa sobre los perfiles metabólicos en muestras de orina de pacientes con cáncer gástrico (GC), enfermedades gástricas benignas (BN) y personas sanas (HE). Sin embargo, es importante considerar ciertas limitaciones. En primer lugar, la calidad y disponibilidad de los datos pueden influir en la interpretación de los resultados, ya que factores como el ruido técnico y la variabilidad biológica pueden afectar la robustez de los hallazgos. Además, si bien los métodos de normalización y preprocesamiento ayudan a mitigar estos efectos, la elección de diferentes enfoques podría impactar la reproducibilidad de los resultados.

Otra limitación clave radica en la capacidad de extrapolar los resultados a otros sistemas biológicos o cohortes independientes. La validación experimental y el uso de conjuntos de datos externos serían fundamentales para confirmar la aplicabilidad de los patrones observados. Además, la interpretación biológica de los resultados requiere consideraciones adicionales, como la integración con información funcional y de redes metabólicas, lo que podría fortalecer la relevancia de los resultados.

A pesar de estas limitaciones, el estudio nos proporciona una base sólida para futuras investigaciones en metabolómica aplicada a la detección del cáncer gástrico. Los resultados nos sugieren que, si se sigue avanzando en el análisis de estos datos,

podríamos estudiar si hay metabolitos que puedan actuar como biomarcadores potenciales.

Conclusión

El presente estudio ha explorado los perfiles metabólicos de muestras de orina de pacientes con cáncer gástrico, enfermedades gástricas benignas y controles sanos mediante un enfoque exploratorio basado en pruebas estadísticas y técnicas de visualización. Aunque se han encontrado diferencias significativas en la intensidad de algunos metabolitos, el tamaño del efecto es pequeño, lo que sugiere que estas diferencias pueden no ser lo suficientemente marcadas como para ser utilizadas de forma directa en aplicaciones diagnósticas.

El análisis mediante *pheatmap* ha mostrado cierta estructura en los datos, pero no reveló una separación clara entre los grupos, lo que sugiere que factores adicionales pueden estar influyendo en la variabilidad observada. Para fortalecer la utilidad de estos resultados, es necesario aplicar modelos de clasificación más avanzados y realizar estudios de validación experimental para evaluar la relevancia biológica de los metabolitos identificados.

Por lo tanto, en este trabajo resaltamos la importancia de nuevos enfoques complementarios en estudios de metabolómica y destacamos la necesidad de futuras investigaciones para mejorar la detección del cáncer gástrico mediante análisis metabolómicos.

Referencias

[1] Chan, A. W., Mercier, P., Schiller, D., Bailey, R., Robbins, S., Eurich, D. T., Sawyer, M. B., Broadhurst, D. (2016). 1H-NMR urinary metabolomic profiling for diagnosis of gastric cancer. British Journal of Cancer, 114(1), 59-62. doi:10.1038/bjc.2015.414

[2] National Metabolomics Data Repository (NMDR), https://www.metabolomicsworkbench.org/

[3] Bioconductor, https://www.bioconductor.org/about/

[4] Repositorio GitHub: https://github.com/mriosc/Rios-Cadenas-Marc-PEC1

Anexo

Código para crear el objeto *SummarizedExperiment*

```
# Cargar librerias

library(SummarizedExperiment)
library(readxl)
library(dplyr)
library(ggplot2)
library(pheatmap)

# Ruta del archivo
file_path <- "Gastric_NMR.xlsx"

# Leer hojas del archivo Excel
data_sheet <- read_excel(file_path, sheet = "data")
peak_sheet <- read_excel(file_path, sheet = "peak")

# Mostrar primeras filas
head(data_sheet)

## # A tibble: 6 × 136
##     Idx `Day of Expt`       Sample_Type    QC Batch Order Sample_id
M1     M2
##   <dbl> <dttm>              <chr>       <dbl> <dbl> <dbl> <chr>
<dbl>  <dbl>
## 1     1 2014-12-08 00:00:00 QC              1     1     1 sample_1
90.1 4.92e2
## 2     2 2014-12-08 00:00:00 Sample          0     1     2 sample_2
43   5.26e2
## 3     3 2014-12-08 00:00:00 Sample          0     1     3 sample_3
214.  1.07e4
## 4     4 2014-12-08 00:00:00 Sample          0     1     4 sample_4
31.6 5.97e1
## 5     5 2014-12-08 00:00:00 Sample          0     1     5 sample_5
81.9 2.59e2
## 6     6 2014-12-08 00:00:00 Sample          0     1     6 sample_6
197.  1.28e2
## # ℹ 127 more variables: M3 <dbl>, M4 <dbl>, M5 <dbl>, M6 <dbl>, M7
<dbl>,
##   M8 <dbl>, M9 <dbl>, M10 <dbl>, M11 <dbl>, M12 <dbl>, M13 <dbl>,
M14 <dbl>,
##   M15 <dbl>, M16 <dbl>, M17 <dbl>, M18 <dbl>, M19 <dbl>, M20 <dbl>,
##   M21 <dbl>, M22 <dbl>, M23 <dbl>, M24 <dbl>, M25 <dbl>, M26 <dbl>,
##   M27 <dbl>, M28 <dbl>, M29 <dbl>, M30 <dbl>, M31 <dbl>, M32 <dbl>,
##   M33 <dbl>, M34 <dbl>, M35 <dbl>, M36 <dbl>, M37 <dbl>, M38 <dbl>,
##   M39 <dbl>, M40 <dbl>, M41 <dbl>, M42 <dbl>, M43 <dbl>, M44 <dbl>,
…

head(peak_sheet)
```

```
## # A tibble: 6 × 3
##     Idx Name  Label
##   <dbl> <chr> <chr>
## 1     1 M1    1_3-Dimethylurate
## 2     2 M2    1_6-Anhydro-β-D-glucose
## 3     3 M3    1_7-Dimethylxanthine
## 4     4 M4    1-Methylnicotinamide
## 5     5 M5    2-Aminoadipate
## 6     6 M6    2-Aminobutyrate
```

```
# Asegurar que la matriz de expresión tenga metabolitos en filas y
muestras en columnas
expr_data <- data_sheet %>%
  select(starts_with("M")) %>%
  as.matrix()

rownames(expr_data) <- data_sheet$Sample_id  # Etiquetar filas con IDs de
muestra
expr_data <- t(expr_data)  # Transponer para que metabolitos sean filas

# Asegurar que meta_samples tenga filas con nombres de muestra
meta_samples <- data_sheet %>%
  select(Sample_id, Sample_Type, Batch)

rownames(meta_samples) <- meta_samples$Sample_id  # Asignar Sample_id
como rownames
meta_samples$Sample_id <- NULL  # Eliminar columna redundante


# Asegurar que meta_metabolites tenga filas con nombres de metabolitos
meta_metabolites <- peak_sheet %>%
  rename(Metabolite = Label) %>%
  select(Name, Metabolite)

rownames(meta_metabolites) <- meta_metabolites$Name  # Asignar Name como
rownames
meta_metabolites$Name <- NULL  # Eliminar la columna redundante


######################################################################3


# Crear objeto SummarizedExperiment con dimensiones correctas
se <- SummarizedExperiment(
  assays = list(counts = expr_data),
  colData = meta_samples,
  rowData = meta_metabolites
)
```

```
# Mostrar estructura del objeto
se

## class: SummarizedExperiment
## dim: 129 140
## metadata(0):
## assays(1): counts
## rownames(129): M1 M2 ... M128 M129
## rowData names(1): Metabolite
## colnames(140): sample_1 sample_2 ... sample_139 sample_140
## colData names(2): Sample_Type Batch

# Guardar el objeto SummarizedExperiment en formato .Rda
save(se, file = "SummarizedExperiment_Gastric_NMR.Rda")
```

Código para el análisis exploratorio completo

Distribución de intensidad de los Metabolitos por Tipo de Muestra

```
library(tidyr)

df_long <- data_sheet %>%
  select(Sample_Type, starts_with("M")) %>%
  pivot_longer(cols = starts_with("M"), names_to = "Metabolite",
values_to = "Intensity")

# Boxplot de distribución de intensidad
ggplot(df_long, aes(x = Sample_Type, y = Intensity, fill = Sample_Type))
+
  geom_boxplot(outlier.shape = NA) +
  scale_y_log10() +
  theme_minimal() +
  labs(title = "Distribución de Intensidad de Metabolitos por Tipo de
Muestra",
       x = "Tipo de Muestra", y = "Intensidad (log10)")

# Verificación de valores no finitos
cat("Número de valores NA en la matriz de expresión:",
sum(is.na(assay(se))), "\n")

## Número de valores NA en la matriz de expresión: 915

cat("Número de valores Inf o NaN en la matriz de expresión:",
sum(!is.finite(assay(se))), "\n")

## Número de valores Inf o NaN en la matriz de expresión: 915
```

Prueba de normalidad

```r
set.seed(123)  # Fijar semilla para reproducibilidad

# Obtener el número total de observaciones en cada grupo
n_qc <- sum(df_long$Sample_Type == "QC")
n_sample <- sum(df_long$Sample_Type == "Sample")

# Seleccionar la cantidad mínima entre 5000 y el tamaño real
sample_qc <- sample(df_long$Intensity[df_long$Sample_Type == "QC"],
min(n_qc, 5000))
sample_sample <- sample(df_long$Intensity[df_long$Sample_Type ==
"Sample"], min(n_sample, 5000))

# Prueba de Shapiro-Wilk
shapiro.test(sample_qc)

##
##  Shapiro-Wilk normality test
##
## data:  sample_qc
## W = 0.21944, p-value < 2.2e-16

shapiro.test(sample_sample)

##
##  Shapiro-Wilk normality test
##
## data:  sample_sample
## W = 0.078549, p-value < 2.2e-16

ggplot(df_long, aes(sample = Intensity, color = Sample_Type)) +
  stat_qq() +
  stat_qq_line() +
  facet_wrap(~Sample_Type) +
  theme_minimal()
```

Prueba de Mann-Whitney U (para datos no normales)

```r
wilcox.test(Intensity ~ Sample_Type, data = df_long)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Intensity by Sample_Type
## W = 15252062, p-value = 0.00285
## alternative hypothesis: true location shift is not equal to 0

ggplot(df_long, aes(x = Sample_Type, y = Intensity, fill = Sample_Type))
+
  geom_boxplot() +
  geom_jitter(width = 0.2, alpha = 0.3) +
  labs(title = "Comparación de Intensidades entre Grupos",
```

```
       subtitle = paste("Wilcoxon p-value:", signif(0.00285, 3))) +
  theme_minimal()

r_wilcoxon <- abs(qnorm(0.00285) / sqrt(nrow(df_long)))
r_wilcoxon

## [1] 0.02057154
```

Pheatmap

```
# Tratamiento de valores faltantes
expr_matrix <- assay(se)
expr_matrix <- apply(expr_matrix, 1, function(x) {
  x[is.na(x)] <- median(x, na.rm = TRUE)
  x[!is.finite(x)] <- median(x, na.rm = TRUE)
  return(x)
})
expr_matrix <- as.matrix(expr_matrix)
mode(expr_matrix) <- "numeric"

# Confirmaciones
cat("Número de valores NA después de limpieza:", sum(is.na(expr_matrix)),
"\n")

## Número de valores NA después de limpieza: 0

cat("Número de valores Inf o NaN después de limpieza:",
sum(!is.finite(expr_matrix)), "\n")

## Número de valores Inf o NaN después de limpieza: 0

library(pheatmap)

pheatmap(
  t(expr_matrix),
  annotation_col = as.data.frame(colData(se)),  # Convertir colData a
data.frame
  main = "Mapa de Calor de Metabolitos",
  cluster_cols = TRUE,
  cluster_rows = TRUE
)
```