Integración de datos ómicos

Lourdes Martínez Martínez
Máster en Bioinformática y Bioestadística
Trabajo Fin de Máster

Nombre del consultor:
Ricardo Gonzalo Sanz

02/01/2018

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Integración de datos ómicos*
Nombre del autor:	*Lourdes Martínez Martínez*
Nombre del consultor/a:	*Ricardo Gonzalo Sanz*
Fecha de entrega (mm/aaaa):	02/01/2018
Titulación:	*Máster en Bioinformática y Bioestadística*
Área del Trabajo Final:	*Trabajo Fin de Máster*
Idioma del trabajo:	*Español*
Palabras clave	*Integración, datos ómicos, PLS*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.*

En los últimos años, se ha aumentado la generación de datos procedentes de las tecnologías ómicas, con fines biomédicos e industriales. El desarrollo de herramientas que permiten su análisis simultáneo es crucial para la obtención de información que no se puede conseguir con el estudio de una sola ómica aislada.

El presente Trabajo Fin de Máster se centra en el estudio de datos ómicos de manera integrada. Para ello, se ha trabajado con dos grupos de datos diferentes, procedentes de estudios de transcriptómica y metabolómica. Se ha procedido a un análisis individual de cada conjunto de datos, mediante lenguaje de programación R, concretamente a través de paquetes de Bioconductor.

Asimismo, la integración de datos se ha efectuado mediante estrategias de análisis multivariante, utilizando este mismo lenguaje de programación. Se ha desarrollado a través de dos aproximaciones diferentes: mínimos cuadrados parciales y análisis de correlación canónica, con el objetivo de compararlos y elegir el método más adecuado para la integración de los datos del estudio.

Tras el análisis, se ha seleccionado la aproximación mediante mínimos cuadrados parciales como la manera más adecuada de integrar los datos procedentes de RNA-seq y los de cuantificación de la concentración metabólica, ya que son técnicas que generan un gran número de variables.

De este modo, la implementación y aprendizaje de estas metodologías para estudiar las relaciones entre genes y metabolitos, ha supuesto una mejora en el conocimiento de las bases moleculares del organismo de estudio.

Abstract (in English, 250 words or less):

In recent years, the data generation from omic technologies has increased, specially, with biomedical and industrial purposes. The development of tools that allow their simultaneous analysis is essential to obtain information that could not be gathered with the study of a single isolated omic field.

The present Master's Project focuses on the study of omics data in an integrated manner. For that purpose, we have worked with different data groups, collected from transcriptomics and metabolomics studies. We have carried out an individual analysis of each data set, using the R programming language, specifically through Bioconductor packages.

Moreover, the data integration has been achieved by using multivariate analysis strategies, running this same programming language. It has been developed through two different approaches: partial least squares and canonical-correlation analysis in order to compare and choose the most appropriate method for the integration of the study data.

After the analysis, the approach was selected by using partial least squares as they are the most accurate way to integrate the data from the RNA-sequencing (RNA-Seq) and the quantification of metabolite concentration, which are techniques that generate a large number of variables.

Thus, the implementation and learning of these methodologies for the study of the relationships between genes and metabolites has improved the knowledge of the molecular bases of the studied organism.

ÍNDICE

Lista de figuras

Lista de tablas

1. Introducción

1.1. Contexto y Justificación del Trabajo

1.1.1. Descripción general

El Trabajo Fin de Máster (TFM) se centra en el estudio de datos ómicos de manera integrada, con el fin de ampliar el conocimiento y ayudar a identificar las relaciones biológicas que sólo son evidentes a través de ciertos análisis holísticos integrando datos procedentes de ómicas diferentes.

Se trabajó con dos dataset de ómicas diferentes, como son la transcriptómica y la metabolómica, de una misma muestra para poder integrarlos y estudiar sus relaciones con el fin de comprender mejor las bases moleculares de un organismo.

1.1.2 Justificación del TFM

En los últimos años, se ha aumentado la generación de datos ómicos (transcriptómica, metabolómica, proteómica…) con fines biomédicos e industriales. El desarrollo de herramientas que permiten su análisis hace que el estudio simultáneo de estos datos procedentes de distintas técnicas ómicas crezca exponencialmente, con el propósito de encontrar respuestas a las preguntas planteadas en el origen de los experimentos, que no se pueden dilucidar con el estudio de una sola ómica aislada.

La elección de este TFM basado en la integración de datos ómicos permitirá el aprendizaje de esta metodología, en auge hoy día debido a su gran aplicabilidad tanto en biomedicina como en biotecnología, como se puede observar en la generación de cantidades masivas de datos procedentes de estudios experimentales; con el fin de poder extrapolar este conocimiento a diversos estudios. Además este trabajo ayudará a consolidar conocimientos adquiridos en el Máster, así como ampliar las competencias adquiridas durante estos estudios.

Debido a la creciente demanda de conocimiento de los sistemas biológicos existentes en la actualidad, se propone en este TFM obtener una comprensión más compleja de dichos sistemas mediante los métodos de integración de datos, en concreto, para su aplicación en diversos campos, como la medicina y la industria. Por este motivo, y dada la complejidad del genoma de los organismos y su regulación, la integración de datos multi-ómicos prevée ser una estrategia acorde con la obtención de información biológicamente significativa para dilucidar los procesos biológicos que ocurren dentro de los organismos, con el fin desarrollar diferentes estrategias desde la medicina personalizada a cada paciente, por ejemplo, para identificar variantes biológicas (biomarcadores), caracterizar sistemas bioquímicos complejos y para estudiar procesos fisiopatológicos, hasta la producción masiva de un compuesto de interés biotecnológico en industria.

1.2. Objetivos

1.2.1. Objetivos generales

Objetivo 1: Analizar los datos de transcriptómica.
Objetivo 2: Analizar los datos de metabolómica.
Objetivo 3: Integración de los datos de transcriptómica con los de metabolómica.

1.2.2. Objetivos específicos

Objetivo 1: Analizar los datos de transcriptómica.

1.1. Selección de datos de expresión para implementar en R.

1.2. Analizar y realizar las transformaciones necesarias sobre los datos de transcriptómica para poder integrarlos posteriormente.

Objetivo 2: Analizar los datos de metabolómica.

2.1. Selección de datos de metabolómica para implementar en R.

2.2. Analizar y realizar las transformaciones necesarias sobre los datos para poder integrarlos posteriormente.

Objetivo 3: Integración de datos ómicos.

3.1. Integración de los datos de transcriptómica con los de metabolómica.

3.2. Integración de los datos mediante una segunda aproximación.

3.3. Comparación de los resultados de ambas aproximaciones.

3.4. Realización de un informe mediante R-Markdown.

1.3. Enfoque y método seguido

El TFM estará orientado por unos objetivos principales que servirán de base para la realización del proyecto, desde la recopilación de los datos, su análisis y tratamiento computacional, hasta la obtención de resultados.

En primer lugar se realizó una búsqueda bibliográfica para conocer los diferentes métodos de tratamiento de datos ómicos hoy día, así como su integración, seleccionando el más adecuado en nuestro estudio en base a los datos seleccionados, por lo que este proceso se hará en paralelo con la búsqueda de dichos datos procedentes de las diferentes ómicas, de la misma muestra.

Se seleccionó el lenguaje de programación a utilizar en el diseño, que en este caso será R, en concreto, utilizando Rstudio como entorno de trabajo. Los paquetes y librerías necesarios serán implementados según el procedimiento elegido.

Por último, se procedió a la realización de un informe de trabajo mediante RMarkdown, que permitió una visualización dinámica y reproducible de los resultados.

1.4. Planificación del Trabajo

1.4.1. Tareas

Los objetivos descritos anteriormente se dividieron a su vez en diversas tareas:

Objetivo 1.1: Selección de los datos de expresión para implementar en R.

- **Búsqueda bibliográfica** de información sobre las técnicas de integración de datos ómicos. Redacción de una breve introducción de esta temática.

- **Selección** del tipo de ómicas que se van a integrar.

- **Obtención de los datos de transcriptómica**, ya sea una base de datos, resultados experimentales de laboratorio…

Objetivo 1.2: Análisis y realización de las transformaciones necesarias sobre los datos de transcriptómica para poder integrarlos posteriormente.

- **Implementación de los datos de transcriptómica** en R, para lo que se realizará un estudio previo del formato.

- **Preprocesamiento** de los datos (control de calidad y transformación de los datos brutos, filtrado, normalización …)

- **Análisis de expresión diferencial** y visualización.

Objetivo 2.1: Selección de los datos de metabolómica para implementar en R.

- **Obtención de los datos de metabolómica**, ya sea una base de datos, resultados experimentales de laboratorio… Estos datos deben ser obtenidos de la misma muestra que los datos de transcriptómica del Objetivo 1.1.

Objetivo 2.2: Análisis y realización de las transformaciones necesarias sobre los datos para poder integrarlos posteriormente.

- **Implementación de los datos de metabolómica** en R, para lo que se realizará un estudio previo del formato.

- **Preprocesamiento** de los datos (transformación de los datos brutos, filtrado, normalización…)

- **Análisis estadístico** y visualización.

Objetivo 3.1: Integración de los datos de transcriptómica con los de metabolómica.

- Obtención del **método estadístico** más adecuado para las muestras del trabajo.

- Selección de los **paquetes** de R con los que se va a realizar la integración.

- Desarrollo de la **integración** de datos ómicos.

Esta tarea constituye el eje central del TFM.

Objetivo 3.2: Integración de los datos mediante una **segunda aproximación**.

- Seleccionar un nuevo método para realizar la aproximación de la integración, procedente de la bibliografía del Objetivo 1.1.

- Realizar nueva integración basada en la aproximación seleccionada.

Objetivo 3.3: Comparación de los resultados de ambas aproximaciones.

- **Comparación** de los resultados de ambos estudios.

- Definición de qué aproximación es la más adecuada, **discusión** de los resultados.

Objetivo 3.4: Realización de un informe mediante R-Markdown.

1.4.2. Calendario

Tabla 1. Calendario de Tareas donde se reflejan las diferentes fechas.

	Fecha Inicio	Fecha Fin
PEC 0. Definición de los contenidos del trabajo	20/09/17	02/10/17
PEC 1. Plan de trabajo	03/10/17	16/10/17
PEC 2. Desarrollo del trabajo Fase I	17/10/17	20/11/17
Búsqueda bibliográfica	17/10/17	09/11/17
Selección de ómicas	17/10/17	18/10/17
Obtención de los datos de transcriptómica	19/10/17	20/10/17
Implementación de los datos de transcriptómica	20/10/17	22/10/17
Preprocesamiento de datos	22/10/17	25/10/17
Análisis de expresión diferencial y visualización	26/10/17	30/11/17
Obtención de los datos de proteómica	02/11/17	03/11/17
Implementación de los datos de proteómica	04/11/17	06/11/17
Preprocesamiento de datos	07/11/17	11/11/17
Análisis estadístico	14/11/17	20/11/17
PEC 3. Desarrollo del trabajo Fase II	21/11/17	18/12/17
Obtención del método estadístico	21/11/17	23/11/17
Selección de paquetes de R	24/11/17	25/11/17
Desarrollo de la integración	26/11/17	05/12/17
Selección de la nueva aproximación	07/12/17	09/12/17
Realización de la integración	10/12/17	16/12/17
Comparación de los resultados y discusión	17/12/17	18/12/17
PEC 4. Redacción de la memoria	19/12/17	02/01/17
PEC 5a. Elaboración de la presentación	03/01/17	10/01/17
PEC 5b. Defensa	11/01/17	22/01/17

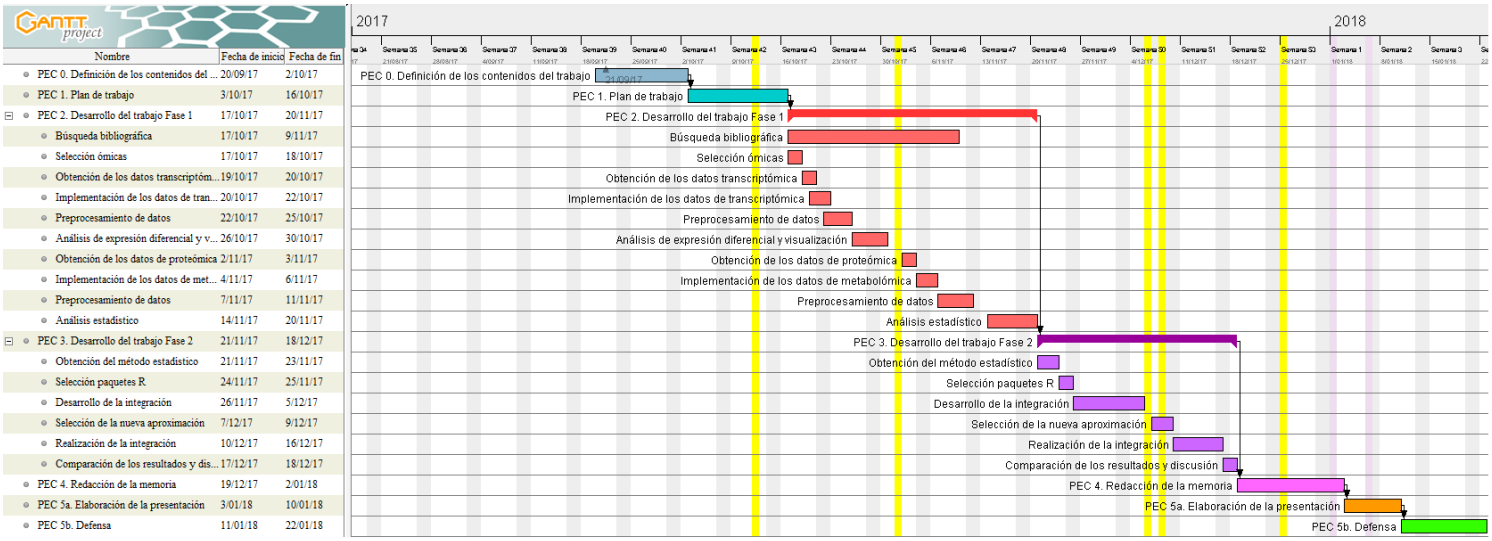


Figura 0. Diagrama de Gantt correspondiente a la planificación de las tareas del proyecto.

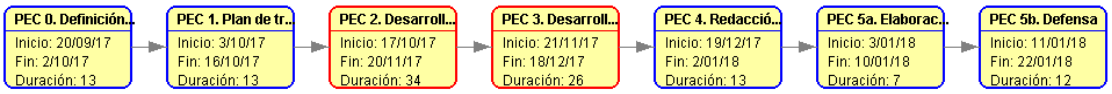


Figura 1. Diagrama de PERT correspondiente a la planificación de las tareas del proyecto.

1.4.3. Hitos

Los hitos de los que consta este trabajo se dividen en siete puntos clave a escala temporal, cada uno correspondiente a la entrega de una Prueba de Evaluación Continua (PEC):

PEC0: Definición de los contenidos del trabajo (2 de Octubre)

PEC1: Plan de trabajo (16 de Octubre)

PEC2: Desarrollo del trabajo Fase 1 (20 de Noviembre)

PEC3: Desarrollo del trabajo Fase 2 (18 de Diciembre)

PEC4: Redacción de la memoria (2 de Enero)

PEC5a: Elaboración de la presentación (10 de Enero)

PEC5b: Defensa (22 de Enero)

1.4.4. Análisis de riesgos

1) **Riesgos asociados a datos del el proyecto.** La selección de los datos siempre es un proceso costoso, ya que existe una gran cantidad de datos ómicos de los que se pueden disponer, sin embargo, son pocas las muestras sobre las que se hayan hecho las ómicas que se quieran integrar. No suelen ser fácil de encontrar y requieren un tiempo extra, ya que según los datos disponibles habrá que utilizar una herramienta u otra para su análisis. Existe la posibilidad de que en fases posteriores se detectaran problemas con los datos seleccionados que dificultaran su procesamiento dentro del plan previsto, por lo que la selección y ajuste de los datos debe ser precisa y adecuada.

2) **Riesgos asociados a la búsqueda de información:** El proyecto demanda una exhaustiva búsqueda bibliográfica de los métodos utilizados hoy en día para el análisis de datos ómicos y su integración. Debido a la dificultad y extensión de estos métodos, es posible que la búsqueda de información y familiarización práctica con las herramientas a utilizar sea más extensa de lo propuesto en el plan de trabajo.

3) **Riesgos asociados al tiempo y el alcance del proyecto:** El alcance que se ha previsto para el proyecto respecto al tiempo podría no adecuarse, dependiendo de la velocidad y calidad con la que se efectúen cada uno de los hitos del plan de trabajo.

1.5. Resultados esperados

Una vez finalizado el TFM se habrán generado los siguientes documentos:

- **Plan de trabajo**.
- **Informe** del trabajo en el que se detallará el proceso de integración de los datos (entregados en las PEC 2 y PEC3).
- **Memoria** en la que se describirá en profundidad el trabajo realizado, tendrá una extensión máxima de 90 páginas, y mínima de 50-60 páginas (incluyendo figuras y código).
- **Presentación virtual** en la que se realizará un resumen del trabajo realizado.
- **Autoevaluación** del proyecto.

1.6. Estructuración del proyecto

La estructura de la memoria será plasmada según el modelo especificado en la plantilla.

1. Introducción

1.1 Contexto y justificación del Trabajo

1.2 Objetivos del Trabajo

1.3 Enfoque y método seguido

1.4 Planificación del Trabajo

1.5 Breve sumario de productos obtenidos

1.6 Breve descripción de los otros capítulos de la memoria

2. Resto de capítulos

2.1 Introducción a la integración de datos ómicos

2.2 Selección y descripción de los datos para el estudio

2.2 Análisis de los datos

2.3 Desarrollo del trabajo de integración

2.4 Comparativa de las integraciones

3. Conclusiones: Descripción de las conclusiones del trabajo, así como la metodología, objetivos y futuros proyectos relacionados con el tema.

4. Glosario: Definición de los términos y acrónimos más relevantes utilizados dentro de la Memoria.

5. Bibliografía: Lista numerada de las referencias bibliográficas utilizadas dentro de la memoria (libros, artículos de revistas, páginas web…).

6. Anexos: Listado de apartados que son demasiado extensos para incluir dentro de la memoria.

2. Integración de datos ómicos

2. 1. ¿Qué es la integración de datos ómicos?

En los últimos años, se ha aumentado la generación de datos procedentes de las tecnologías ómicas, (transcriptómica, metabolómica, proteómica, fluxómica…) con fines biomédicos e industriales. Su análisis independiente, a través de métodos estadísticos univariantes, proporciona una información valiosa para el conocimiento del organismo y su aplicación en diversas áreas biológicas [1].

Sin embargo, este análisis individualizado pierde cierta información que podría ser crucial. De este modo, el desarrollo de herramientas que permiten el análisis de los diferentes datos ómicos de manera conjunta, mediante análisis multivariantes, proporcionan una imagen más cercana a la realidad de lo que ocurre en el sistema biológico.

Esto hace que este tipo de estudio esté creciendo exponencialmente, con el propósito de encontrar respuestas a las preguntas planteadas en el origen de los experimentos, que no se pueden dilucidar con el estudio de una sola ómica aislada.

2.2. Diferentes tipos de integración

Existen diversos procedimientos en los que afrontar el proceso de integración de datos ómicos, de tres maneras diferentes descritas por Cavill y colaboradores [2]: integración conceptual, basada en el modelo e integración estadística.

En primer lugar, la integración conceptual se basa en el análisis individual de cada ómica y, una vez realizado, se ponen de manifiesto las conclusiones, buscando relaciones entre ellas. La limitación de este proceso es que pueden omitir cierta información que necesitaría el análisis de dos ómicas conjuntamente.

Por otro lado, la integración basada en el modelo es un ideal que todavía no se ha alcanzado, debido a la falta de datos para la construcción de los modelos y a su complejidad, ya que se trata de aproximaciones validadas con datos experimentales.

De este modo, el método de integración más utilizado es la integración estadística, donde se buscan asociaciones estadísticas entre los elementos de los diferentes conjuntos de datos. Esta aproximación se puede dividir en diferentes subgrupos: integración basada en la correlación, en la concatenación, análisis multivariante y por último, métodos basados en el conocimiento biológico, que son descritos a continuación:

Integración basada en la correlación

Se centra en encontrar vínculos correlativos entre elementos de un conjunto de datos y del otro, de diversas maneras: correlación de Pearson y Spearman, prueba gamma de Goodman, modelos lineales robustos y correlaciones parciales. La correlación a menudo falla debido a escalas diferentes de los datos. Los análisis basados en correlación son útiles para la integración de datos ómicos cuando hay una falta de conocimiento bioquímico, por lo que son ampliamente utilizados para la integración de datos multiómicos. Estos enfoques pueden proporcionar una visión limitada en casos de sistemas altamente multicolineales, por lo que se utilizan los modelos gráficos gaussianos, la correlación parcial y las redes bayesianas, ya que tienen la capacidad de desacoplarse directamente de asociaciones de variables indirectas [2].

Integración basada en concatenación

Se basa en concatenar las tablas de datos producidas por cada tecnología ómica en una única tabla de datos, para posteriormente realizar un análisis integrado. Al obtener los datos de tecnologías muy diferentes, se introducirá un sesgo que favorece el conjunto de datos más grande, con diferentes varianzas [3]. Por todo ello, la obtención resultados entre la integración de dos ómicas, como puede ser la la metabolómica y la transcriptómica, no es sencilla.

Integración multivariante

Son métodos más complejos. Trata de utilizar un conjunto de datos para predecir aspectos de otro conjunto de datos o para encontrar las asociaciones de "covarianza" entre los dos conjuntos de datos [4]. Los conjuntos de datos se usan de forma no concatenada, manteniendo los conjuntos en bloques o dimensiones separadas dentro del modelo. Las dos técnicas multivariantes más comunes son PCA y PLS. Se

utilizan para conjuntos de datos con altos niveles de colinealidad, como es el caso con los datos ómicos, donde muchos genes o metabolitos tendrán perfiles similares [5].

Técnicas integración basada en vías

Son métodos que intentan integrar los datos utilizando el conocimiento biológico existente a través de vías metabólicas predefinidas en bases de datos, representando conexiones complejas entre diversos tipos de componentes (genes, proteínas, metabolitos…). Asignan, de forma automatizada, los metabolitos y transcritos medidos, a redes y encuentran aquellos donde hay evidencia estadística de un cambio significativo en su comportamiento entre dos condiciones, o una correlación entre el comportamiento de la vía y un punto de interés fenotípico.

Tienen la limitación de que en el caso de conocimiento insuficiente de los genes, proteínas y/o interacciones de metabolitos, a menudo se extienden a través de relaciones empíricas o correlaciones entre medidas. En definitiva, cada conjunto de datos ómicos se analiza por separado y se construye un fundamento biológico coherente que explique los fenómenos observados en los perfiles moleculares individuales [6].

2.3. Datos del estudio

Transcriptómica

El transcriptoma es el conjunto completo de transcritos en una célula, y su cantidad, para una etapa del desarrollo o condición fisiológica determinada. El desarrollo de la secuenciación de próxima generación (NGS) de alto rendimiento ha revolucionado la transcriptómica al permitir el análisis de ARN a través de la secuenciación del ADN complementario (ADNc) [7]. De esta forma, se producen un gran número de secuencias de nucleótidos cortas que se alinean con un genoma de referencia. Una de estas técnicas es el RNA-seq, que usa las alineaciones para inferir niveles de expresión génica. La mayoría de los experimentos de RNA-seq toman una muestra de RNA purificado, lo cortan, lo convierten en cDNA y lo secuencian en una plataforma de alto rendimiento, llamadas plataformas de segunda generación, como Illumina, DNBS, SOLiD o Roche 454, Ion Torrent [8]. Los sistemas se secuenciación de tercera generación suponen un ahorro a nivel de librería, puede realizarse sin la necesidad de la creación de estas librerías de amplificación costosas que consumen mucho tiempo, algunos ejemplos son SMRT, Heliscope, Electron microscopy, Nanopore… [9].

Las muestras para transcriptómica se obtuvieron a partir de tres cultivos independientes de cada una de las tres condiciones a analizar, por lo que se tuvieron tres réplicas biológicas para cada condición. Al tener tres condiciones diferentes (denominadas “low”, “high” y “other”), cada individuo representaría una muestra, haciendo un total de nueve muestras, que fueron sometidas a un control de calidad de manera experimental.

Posteriormente, se realizó un análisis de control de calidad de los datos brutos, las lecturas generadas se mapearon frente a la última versión del genoma del organismo y se eliminaron aquellas de baja calidad. Por último, llevó a cabo el ensamblaje, identificación y cuantificación de los transcritos mediante inferencia bayesiana. Estos estudios no se recogen en el presente trabajo ya que hemos partido de los transcritos cuantificados, denominados “counts”, obtenidos mediante SOLiD™ v4, de un total de 3319 genes.

Metabolómica

El metaboloma representa la naturaleza y abundancia de cada metabolito en un sistema biológico. Proporciona información acerca de las variaciones moleculares que surgen como resultado de las diversas perturbaciones ambientales en el organismo. Sin embargo, debido a la naturaleza de los compuestos, la detección de un metaboloma completo con una sola técnica analítica es prácticamente imposible, por lo que los metabolitos se pueden detectar por varios métodos: espectrometría de masas, espectroscopia de resonancia magnética nuclear, espectrometría de absorción en el infrarrojo y espectroscopia de rayos ultravioleta [10].

Del mismo modo que se obtuvieron las muestras para el análisis de RNA-seq, los datos de metabolómica se obtuvieron a partir de tres cultivos independientes de cada una de las condiciones para analizar, por lo que tuvimos tres réplicas biológicas para cada condición, haciendo un total de nueve muestras.

En el estudio, los datos de metabolómica se obtuvieron mediante la separación cromatográfica, que se realizó mediante cromatografía líquida de alta resolución (HPLC) o por cromatografía de gases (GC), dependiendo del tipo de metabolito a analizar, estando ambos acoplados a espectrometría de masas (MS). La problemática que presentan los datos de metabolómica es la posible cuantificación errónea de ciertos

metabolitos intracelulares. Con objeto de evitar estos los errores y poder cuantificar correctamente la concentración intracelular de metabolitos, se utilizó el denominado método diferencial.

Además, con el objetivo de eliminar las cuantificaciones de baja calidad debidas a respuestas no lineales, que surgen por la supresión de iones en el electrospray y a efectos de la matriz, que contiene otros compuestos orgánicos no medidos en la metodología seguida, se utilizó un patrón de control interno, utilizando isotopólogos.

En la Figura 2 se puede observar el pipeline del presente Trabajo Fin de Máster, tanto del análisis de cada ómica individual como su integración.

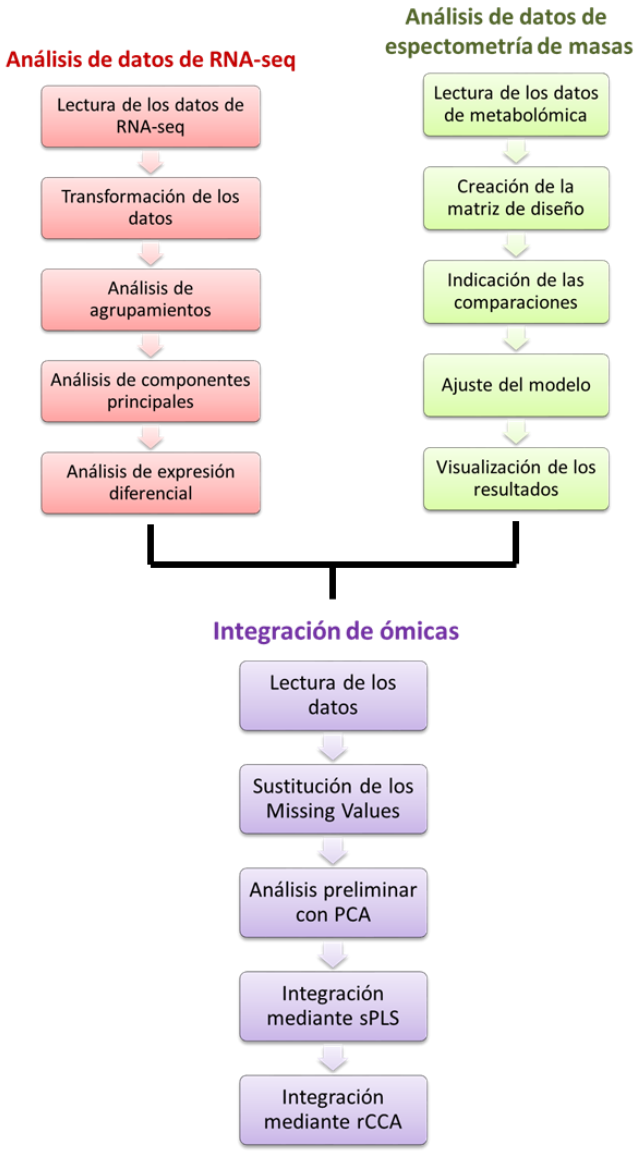


Figura 2. Pipeline del Trabajo Fin de Máster.

3. Integración de datos

3.1 Análisis de datos de RNA-seq

Dependiendo del sofware utilizado será necesario realizar la normalización antes de analizar los datos, o puede realizarla internamente el programa.

A continuación, el siguiente paso fue ajustar los datos a un Modelo Lineal Generalizado (MLGs), ya que son los que mejores se ajustan a los datos de conteo. Siguen la estructura de los modelos lineales clásicos, que tratan de explicar la variable objetivo como una combinación lineal de una función de un conjunto de variables explicativas [11].

Para ello, se ha usado el modelo de regresión Binomial Negativa. Tiene la ventaja de que resuelve el problema de la "sobredispersión" que surge frecuentemente al aplicar un modelo de regresión de Poisson.

Realizamos un contraste que se reduce a una comparación de medias, ya que estudiamos diferencias entre grupos independientes, por lo que se utilizó el estadístico de Wald. Asimismo, se ha trabajado con el logaritmo en base dos del fold-change, debido a la variabilidad de los datos de conteo.

Debido a que existen numerosos contrastes a realizar (uno por gen o metabolito, existiendo un gran número de ellos), es necesario evaluar métodos de comparaciones múltiples, con el objetivo de eliminar falsos positivos. Para ello, se ha utilizado la FDR (False Discovery Rate) [12], que se define como la proporción esperada de hipótesis nulas que son verdaderas entre las que son declaradas como significativas. Trata de controlar la tasa de error tipo uno (rechaza la hipótesis nula cuando es cierta).

En nuestro caso, se ha utilizado el paquete DESeq2 de Bioconductor [13] para el análisis de los datos bajo test Binomial Negativo, ya que este paquete posee un material estadístico potente. Su método de normalización es el denominado RPKM (Reads Per Kilobase of transcript and Million mapped reads). Así, la normalización previa en este paquete no es necesaria. Existen dos funciones que realizan estos cálculos directamente y se expresan luego en los resultados, lo mismo ocurre con el análisis de dispersión.

3.1.1. Lectura de los datos

Para poder implementar los datos de transcriptómica en R ha sido necesario crear una tabla de Excel en la que se sitúen los diferentes genes para analizar en las filas (Gene_ID), y en las columnas las diferentes muestras del estudio (LowA, Low B…), como se puede observar en la Tabla 2. Los números que presenta cada gen en cada muestra son los "counts", se presentan como una tabla que informa, para cada muestra, el número de fragmentos de secuencia que se han asignado a cada gen.

De este modo, se han obtenido los diferentes data.frame del estudio y se han comparado, de manera que coincidan los nombres de las columnas del archivo "Quantification" con el de las filas del "metadata", para poder realizar los análisis posteriores.

Tabla 2. Representación de los counts de cada muestra.

Gene_ID	LowA	LowB	LowC	HighA	HighB	HighC	OtherA	OtherB	OtherC
1 Gene_0001	6100	7446	6006	17293	21735	17293	21735	19466	20405
2 Gene_0002	19301	16710	17157	28759	22728	17157	34650	26410	26410
3 Gene_0003	8190	5087	10083	12411	15270	10429	15270	14652	14003
4 Gene_0004	16225	12689	14272	17998	18733	15526	17998	24752	21855
5 Gene_0005	247	145	223	425	473	350	562	500	617
6 Gene_0006	2149	2149	1927	3618	4219	4723	3741	6135	6135
7 Gene_0007	6769	5171	4704	11984	13365	15692	18125	20637	18125

Además se ha creado un objeto "metadata" en excel, con las condiciones del estudio ("low", "high" y "other").

Tabla 3. Representación de las diferentes condiciones del estudio.

	Name	Condition
1	LowA	low
2	LowB	low
3	LowC	low
4	HighA	high
5	HighB	high
6	HighC	high

De este modo, se han obtenido los diferentes data.frame del estudio y se han comparado, de manera que coincidan los nombres de las columnas del archivo "Quantification" con el de las filas del "metadata", para poder realizar los análisis posteriores.

3.1.2. Análisis mediante DeSeq2

El paquete DESeq2 se basa en la utilización del ajuste de la distribución binomial negativa a partir de los datos de la matriz de cuentas, como se ha visto anteriormente, para realizar el test de significancia de Wald. Dicho test se basa en que la estimación reducida del logaritmo del Fold change (LFC) se divide por su error estándar, lo que da como resultado una estadística z, que se compara con una distribución normal estándar [13]. Los p-valores obtenidos se ajustan para pruebas múltiples usando el procedimiento de Benjamini y Hochberg [12].

En primer lugar, se ha creado el objeto de clase DESeqDataSet, representado mediante el objeto "dds", para almacenar los conteos de lectura (mycounts) y las condiciones del estudio (metadata) que se han utilizado durante el análisis estadístico.

Para ello, se ha usado la función *DESeqDataSetFromMatrix*, ya se ha partido de una matriz de conteo. Además, se ha obtenido la información sobre las muestras (metadata) como un data.frame, y la fórmula de diseño (design=~Condition).

Posteriormente, la función DESeq coge los datos, estimando los factores de tamaño y de dispersión, para cada gen y la adaptación de un modelo lineal generalizado [13].

```
dds <- DESeqDataSetFromMatrix(countData=mycounts,
                              colData=metadata,
                              design=~Condition,
                              tidy=TRUE)

dds1 <- DESeq(dds)
```

3.1.3. Transformación de los datos para su visualización

Aunque en el paquete DESeq2 se opera con conteos, utilizando distribuciones discretas, para la realización de otros análisis como pueden ser la visualización o el

agrupamiento, es útil trabajar con versiones transformadas de los datos de conteo. Esto es debido a ciertos métodos estadísticos para el análisis de datos multidimensionales, funcionan mejor para los datos que generalmente tienen el mismo rango de varianza a diferentes rangos de los valores promedio (homocedasticidad), sin embargo, para los counts obtenidos mediante de ARN-seq, la varianza aumenta con la media [14].

Debido a esto, se ha realizado una transformación logarítmica regularizada (rlog). Para los genes con recuentos altos, el rlog dará un resultado similar a la transformación log2 ordinaria de los recuentos normalizados. Para los genes con recuentos más bajos, los valores se reducen a los promedios de los genes en todas las muestras. Los datos transformados por rlog se vuelven aproximadamente homoscedásticos [13].

Una vez que los datos han sido transformados, se puede ver el efecto de esta transformación mediante un gráfico de dispersion, y así comparar entre los datos obtenidos mediante el log2 y el rlog. Por ejemplo, seleccionando las dos primeras muestras (LowA y LowB), se ha podido observar (Figura 3) que la distribución cambia sobre todo en la parte inferior izquierda del gráfico, correspondiente a los datos de reads con menor conteo, por lo que en el rlog se ven más comprimidos.

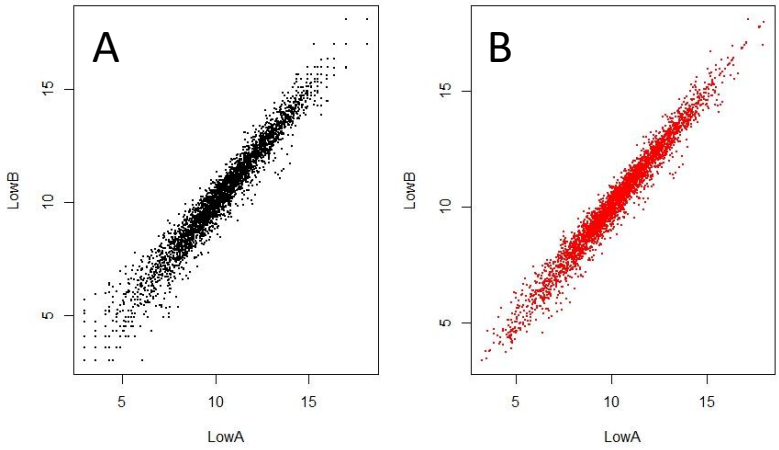


Figura 3. Efecto de la transformación rlog en las muestras LowA y LowB. A) Muestras sin transformar B) Muestras a las que se les ha realizado la transformación regularizada.

3.1.4. Análisis de agrupamientos

De forma complementaria al estudio de los datos, se ha podido realizar un análisis que indique si existe alguna relación entre las muestras, es decir, si hay alguna similitudes entre ellas, mediante un análisis de distancias, usando la función *dist* [13]. El resultado del análisis se puede observar en la Figura 4, indicando el color rojo (valores próximos a 0) una mayor similitud, y el color rosado, una menor:

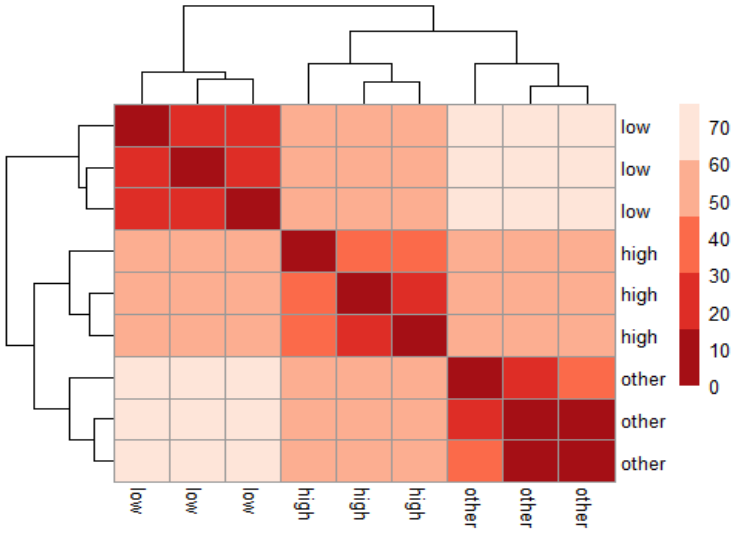


Figura 4. Heatmap de las distancias muestra a muestra utilizando los valores transformados.

De este modo, se ha podido observar que el gráfico tiene tres grupos diferenciados, las muestras se agrupan en condiciones de low, high y other. Por otro lado, se ha visto muy poca similitud entre las muestras pertenecientes a low y other, por lo que no se realizará una comparativa posterior entre estas muestras.

3.1.5. Análisis de componentes principales

De manera que complementaria al análisis de agrupamientos, se ha analizado también la similitud entres muestras mediante el análisis de componentes principales (PCA), mediante la función *plotPCA* del paquete DESeq2.

En esta gráfica, las muestras se proyectan de manera que se extienden en dos direcciones que explican la mayoría de las diferencias entre ellas. El eje x es la

dirección que más separa los puntos de datos, el eje y es una dirección ortogonal a la primera, que es la segunda que más separa los datos. El porcentaje de la varianza total que está contenida en la dirección se imprime en la etiqueta del eje [13].

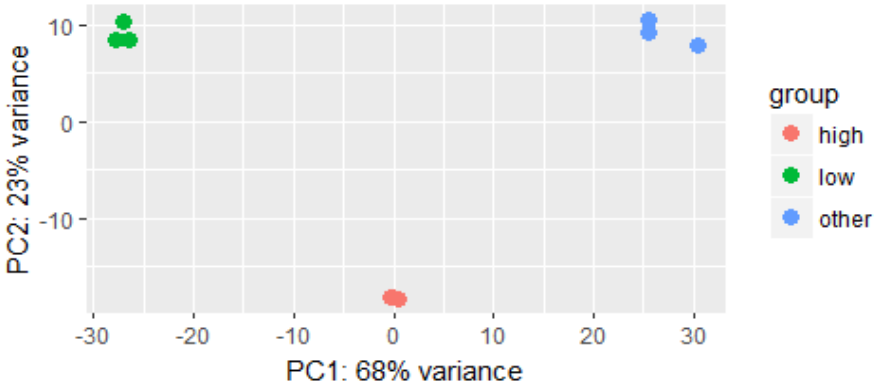


Figura 5. PCA utilizando los valores transformados. En rojo se observan las muestras pertenecientes a "high", en verde a "low" y en azul a "other".

Así, se ha corroborado que existen diferentes agrupamientos, correspondientes a low, high y other, las condiciones del estudio.

3.1.6. Análisis de expresión diferencial

Con el objetivo de visualizar los resultados, se ha utilizado la función *results* de DESeq2, a la que se le ha añadido un contraste (contrast=c("Condition","high","low"), por ejemplo) para seleccionar las muestras que queremos comparar. Así, se realizaron dos comparativas: Low_vs_high y High_vs_other, ya que no interesa un estudio de la comparativa Low_vs_other a nivel experimental, además hemos visto que estas dos últimas muestras tienen muy poca similitud (Apartado 4).

Mediante esta función, se calcula el log2FoldChange, que es la estimación del efecto. Indica cuánto cambia la expresión del gen de una muestra respecto a la otra. Además, DESeq2 realiza para cada gen una prueba de hipótesis para ver si existe una variabilidad experimental, para ello calcula el p-valor.

Posterior a la visualización de los resultados, se han ajustado los datos al estudio, estableciendo un ajuste del p-valor (padj) menor a 0.05. Este p-valor se obtiene mediante el método de Benjamini-Hochberg, indicando el false discovery rate (FDR) [12].

```
#Low_vs_High
res1 <- results(dds1, contrast=c("Condition","high","low"), alpha = 0.05)
summary(res1)


out of 3319 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)     : 839, 25%
LFC < 0 (down)   : 788, 24%
outliers [1]     : 2, 0.06%
low counts [2]   : 0, 0%
(mean count < 7)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results

resadj1 = subset(res1, padj < 0.05)
summary(resadj1)


out of 1627 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)     : 839, 52%
LFC < 0 (down)   : 788, 48%
outliers [1]     : 0, 0%
low counts [2]   : 0, 0%
(mean count < 7)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results

sum(res1$padj <= 0.05, na.rm=TRUE)
```

```
[1] 1627
```

Los resultados de la comparativa entre los datos de RNA-seq de la condición “high” y “low”, con un p-valor ajustado de 0.05 (padj), mostraron un total de 1627 genes diferencialmente expresados, frente a los 3319 genes del estudio (res1). Es estos 1627, 839 estaban sobreexpresados (un 52% para los diferencialmente expresados, un 25% de los genes totales), teniendo sólo dos valores outliers.

```
#High_vs_other
res2 <- results(dds1, contrast=c("Condition","high","other"), alpha = 0.05)
summary(res2)


out of 3319 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)     : 630, 19%
LFC < 0 (down)   : 567, 17%
outliers [1]     : 2, 0.06%
low counts [2]   : 0, 0%
(mean count < 7)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results

resadj2 = subset(res2, padj < 0.05)
summary(resadj2)


out of 1197 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)     : 630, 53%
LFC < 0 (down)   : 567, 47%
outliers [1]     : 0, 0%
low counts [2]   : 0, 0%
(mean count < 7)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results

sum(res2$padj <= 0.05, na.rm=TRUE)

[1] 1197
```

De esta forma, para la comparativa entre high y other, se obtuvieron, con los mismos valores de p-valor ajustado, un total de 1197 genes diferencialmente expresados, lo que supuso un 53% de genes sobreexpresados (630 genes) o un 19% del total de genes del organismo.

3.1.7. Obtención de los genes up y down regulados

Se han seleccionado aquellos genes que presentaban un valor de LFC mayor y menor a la unidad.

```
genes_up1<-as.data.frame(subset(top_genes1, log2FoldChange >= 1))
dim(genes_up1)

[1] 471   6

genes_down1<-as.data.frame(subset(top_genes1, log2FoldChange <= -1))
dim(genes_down1)

[1] 463   6

genes_up2<-as.data.frame(subset(top_genes2, log2FoldChange >= 1))
dim(genes_up2)

[1] 341   6

genes_down2<-as.data.frame(subset(top_genes2, log2FoldChange <= -1))
dim(genes_down2)

[1] 272   6
```

Como resultado, se han obtenido un total de 471 genes upregulados y 463 downregulados en la primera comparación (Low_vs_high), por otro lado, 341 genes upregulados y 272 downregulados en la segunda comparación (High_vs_other).

3.1.8. Gráfico de diferencia de medias

Una gráfica de diferencia de medias permite visualizar la distribución de los coeficientes estimados en el modelo, como pueden ser las comparaciones entre genes

[13]. En el eje y se sitúa el valor del LFC, mientras que en el eje x se sitúan las medias de los counts normalizados.

Se han representado los genes diferencialmente expresados en color rojo junto con el resto de genes analizados en cada una de las comparaciones (res1 y res2).

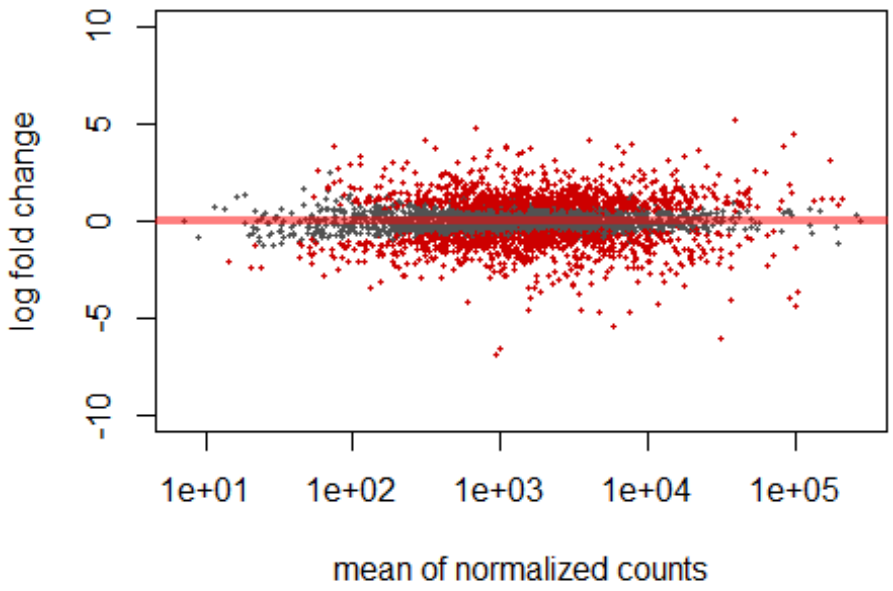


Figura 6. Gráfica de diferencia de medias. Comparativa de Low_vs_High.

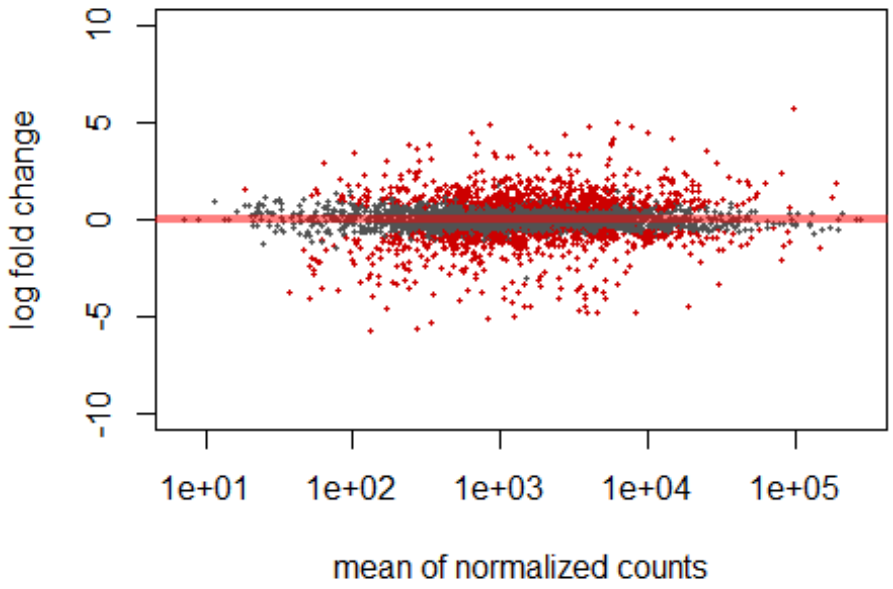


Figura 7. Gráfica de diferencia de medias. Comparativa de High_vs_Other.

3.1.9. Gráficos

Se ha obtenido un Volcano plot que permitiera representar en el mismo gráfico aquellos genes que están por encima y por debajo de los umbrales de fold change (2 y -2) y de FDR (0.05) que hemos indicado anteriormente, para cada una de las comparaciones.

En el eje de abcisas se reprensentan las diferencias de expresión de los genes mediante el LFC, mientras que en el de ordenadas se encuentra el –log del p-valor (Figura 8 y 9).

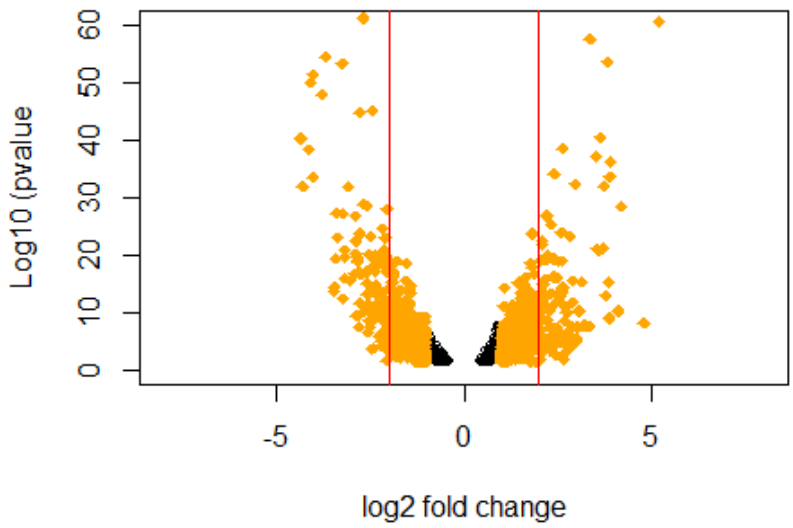


Figura 8. Volcano plot de la compatativa Low_vs_high.

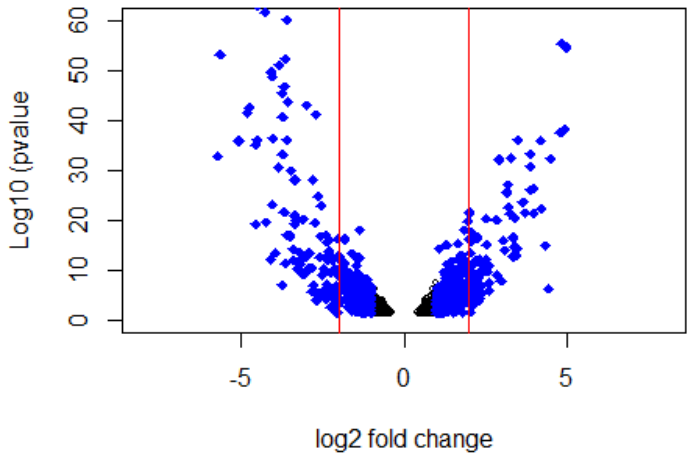


Figura 8. Volcano plot de la compatativa High_vs_other.

Por último, se ha diseñado un heatmap para observar los genes diferencialmente expresados, y así conseguir una información muy visual que puede ayudar a extraer el sentido biológico junto a los datos obtenidos anteriormente. Representa la desviación en una muestra especifica con respecto de la media de las muestras.

Se pueden seleccionar, por ejemplo, los genes más diferentes respecto a la media global, y así "reducir" el heatmap, según el interés de nuestro estudio.

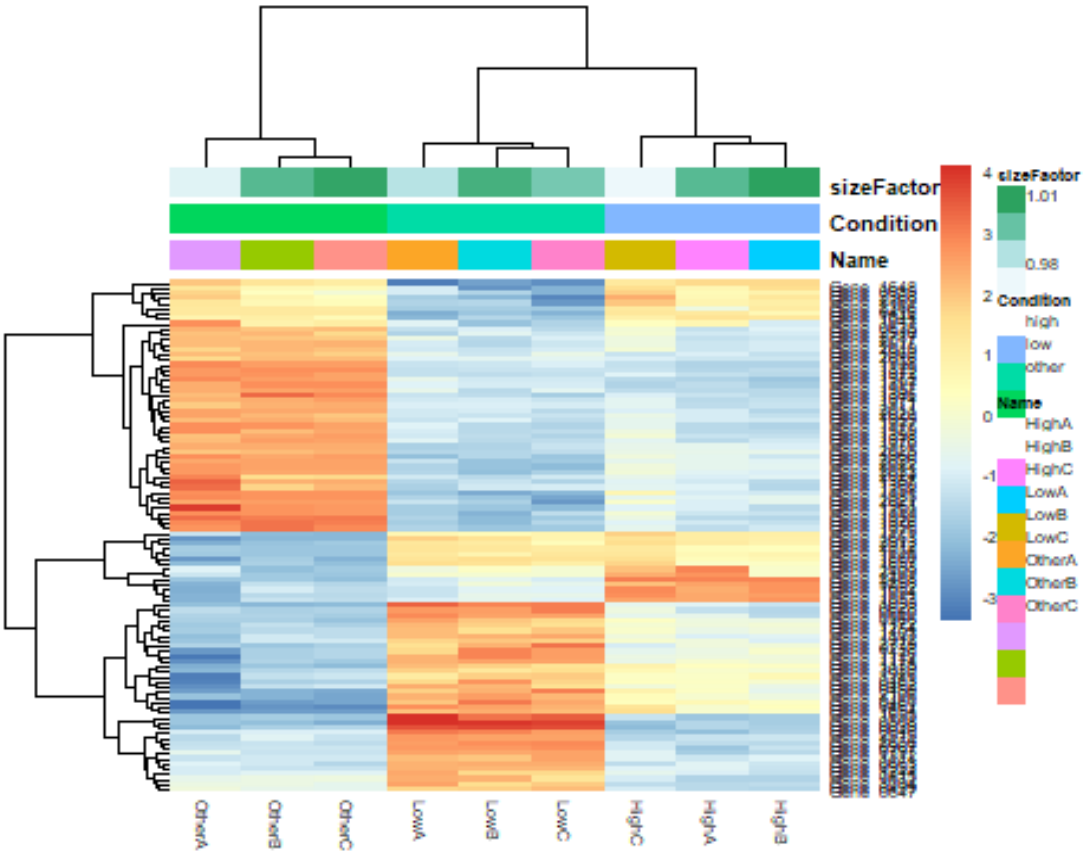


Figura 9. Heatmap de los valores transformados por rlog en las muestras.

De este modo, se puede observar qué genes están más sobreexpresados en las diferentes condiciones, agrupadas de manera jerárquica en el dendograma.

3.2 Análisis de datos de espectometría de masas

Se aplicó una aproximación basada en la utilización del modelo lineal, combinado con una estimación mejorada de la varianza. Para ello se utilizó el paquete *limma* de Bioconductor [15 y 16]. Limma se utiliza para el análisis de datos de expresión génica derivados de microarrays o RNA-Seq. Utiliza modelos lineales para evaluar la expresión diferencial en experimentos multifactoriales. Limma tiene la capacidad de analizar comparaciones entre muchos datos simultáneamente.

Se seleccionó este paquete ya que una de sus principales características es que el análisis es estable incluso para experimentos con un pequeño número de muestras. Además, la valoración de la diferente concentración de metabolitos entre las diferentes condiciones utilizando un método análogo al de microarrays, se consideró el procedimiento más adecuado, ya que se trataron datos contínuos, no discretos como podían ser los contajes del RNA-seq.

En este apartado, se habla de metabolitos "diferencialmente expresados", ya que el estudio se ha realizado de manera análoga a un análisis de microarrays. En realidad, lo que se ha medido han sido las diferencias entre las concentraciones de los metabolitos, entre una condición y otra.

3.2.1. Lectura de los datos

Con el objetivo de implementar los datos de metabolómica en R ha sido necesario crear una tabla en formato csv, en la que se sitúen los diferentes metabolitos (met1, met2…) para analizar en columnas, y en las filas las diferentes muestras del estudio (LowA, Low B…), como se puede visualizar en la tabla 4.

Tabla 4. Representación de las cuantificaciones de cada metabolito en las muestras.

	met1	met2	met3	met4	met5	met6	met7	met8
lowA	0.04773804	NA	0.05300833	0.3469441	1.08034038	0.044684062	1.58571267	0.3324614
lowB	NA	NA	0.04929081	0.1909762	0.95953905	0.035121297	1.33651786	0.4147550
lowC	NA	NA	0.04308331	0.2689602	0.79586700	0.050478091	1.78367293	0.2009711
highA	0.36830155	0.2634513	0.22238626	0.2523529	0.04854462	0.004643399	0.04524675	0.7641381
highB	0.24805141	0.2181564	0.19422515	0.3407767	0.06612695	0.006777754	0.08983785	0.3402145
highC	0.37328811	0.1710920	0.11141302	0.3626701	0.04661904	0.004735598	0.06575766	0.4089359
otherA	0.30447224	0.3914564	NA	NA	0.93402438	0.029519986	NA	NA
otherB	0.29014568	0.3831253	NA	NA	0.91053967	0.029434493	NA	NA
otherC	NA	NA	NA	NA	NA	NA	NA	NA

3.2.2. Creación de la matrix de diseño

En primer lugar, se debe proceder a la creación de una matriz de diseño. es una matriz de indicadores que especifica qué muestras de ARN se aplican en cada conjunto [17]. Se trató como un modelo de tres niveles: comparación entre High, Low y Other. Así había tres grupos, con tres muestras para cada grupo (Tabla 5).

```
group=c("Low","Low","Low","High","High","High", "Other", "Other",
        "Other")
design <- model.matrix( ~ 0 + group)
colnames(design)<-c("High","Low", "Other")
rownames(design)<-c("LowA","LowB","LowC","HighA","HighB","HighC",
                    "OtherA", "OtherB", "OtherC")
```

Tabla 5. Matriz de diseño

	High	Low	Other
LowA	0	1	0
LowB	0	1	0
LowC	0	1	0
HighA	1	0	0
HighB	1	0	0
HighC	1	0	0
OtherA	0	0	1
OtherB	0	0	1
OtherC	0	0	1

3.2.3. Indicación de las comparaciones

Dado un modelo lineal definido a través de una matriz de diseño, pueden formularse las preguntas de interés como contrastes, es decir, comparaciones entre los parámetros del modelo. Para poder realizar las comparaciones entre grupos, se ha procedido a la creación de una matriz de contrastes, mediante la función *makeContrasts* del paquete limma.

Se realizaron dos comparaciones, el grupo high_vs_low y high_vs_other. La comparación entre grupos se describe con un 1 y un -1 en las filas de los grupos a comparar y ceros en las restantes (Tabla 6).

```
cont.matrix <- makeContrasts(low_vs_high = High - Low, high_vs_other =
                                Other - High, levels = colnames(design))
```

Tabla 6. Matriz de contrastes.

	Contrasts	
Levels	low_vs_high	high_vs_other
High	1	-1
Low	-1	0
Other	0	1

3.2.4. Ajuste del modelo

A continuación, se realizó el ajuste del modelo mediante diferentes funciones del paquete "limma" de Bioconductor. En primer lugar, se procedió a la estimación del modelo, mediante la función *lmFit*, realizando luego la estimación de los contrastes con *contrast.fit*. Por último, se utilizó una función eBayes con el fin de obtener una mejor estimación de errores [18 y 19].

```
fit<-lmFit(data, design)
fit.main<-contrasts.fit(fit, cont.matrix)
fit.mainB<-eBayes(fit.main)
```

3.2.5. Visualización de los resultados

Con el objetivo de visualizar los resultados obtenidos, se ha utilizado la función *decideTest*, que permite extraer los resultados de qué metabolitos se encontraron diferencialmente "expresados" de manera significativa para cada contraste [20 y 21], con un p-valor de 0.05 [17].

Una vez guardados los resultados, se ha creado una topTable para cada comparación. Así como en el análisis de los datos de RNA-seq, también se controló el porcentaje de falsos positivos utilizando el metodo de Benjamini y Hochberg, mediante la FDR.

El resultado final fueron dos topTables diferentes, una para cada comparación, las cuales presentan una lista de metabolitos ordenados de mayor a menor diferencia entre sus concentraciones.

```
results <- decideTests(fit.mainB)


##    low_vs_high high_vs_other
## -1          19             7
## 0            1            10
## 1            6             9

topTab.low_vs_high <- topTable(fit.mainB, number=nrow(fit.mainB),
                               coef="low_vs_high", adjust="fdr")

topTab.high_vs_other <- topTable(fit.mainB, number=nrow(fit.mainB),
                                 coef="high_vs_other", adjust="fdr")
```

3.2.6. Gráficos

Se ha ejecutado un volcano-plot, que permite ordenar los metabolitos lo largo del modelo, en dos dimensiones, la biológica, representada por el "fold change" y la estadística representada por el logaritmo negativo del p–valor (Figura 10).

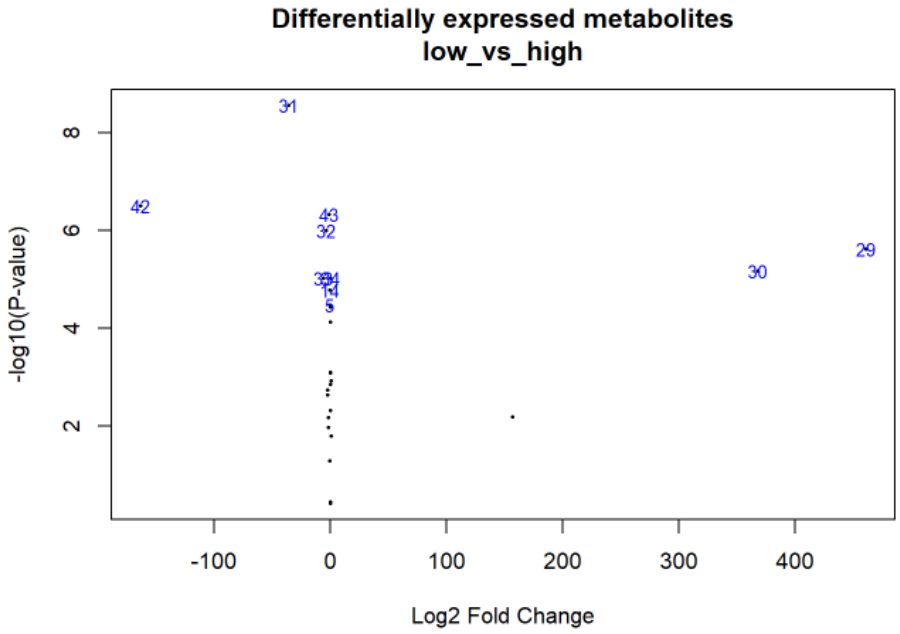


Figura 10. Representación mediante Volcano plot de los metabolitos diferencialmente expresados.

Otro método de visualización que se realizó fue el Diagrama de Venn, que se utiliza para ver que metabolitos varían conjuntamente o no en las dos comparaciones (Figura 11).

Mediante el diagrama se puede observar que las mayores diferencias entre comparaciones se encuentra en low_vs_high. De este modo, estas dos condiciones suponen un mayor contraste en la concentración metabólica del organismo de estudio, mucho mayor que la diferencias que presentarían los metabolitos ante una condición de high en comparación con other. Se podría dilucidar que la condición "low" es crucial para el desarollo molecular del organismo, por lo que el experimentador podría estudiarlo en profundidad.

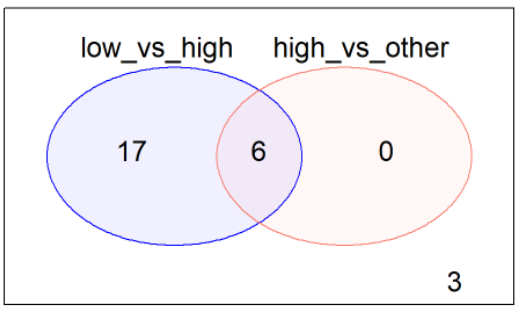


Figura 11. Representación mediante Diagrama de Venn de los metabolitos diferencialmente expresados.

3.3. Integración de datos ómicos

El análisis de datos biológicos procedentes de ómicas, es una tarea compleja, ya que se parte de una cantidad limitada de muestras con numerosas características, lo que da lugar a desajustes en los datos y multicolinealidad [22]. Existen numerosas maneras de abordar estos problemas, mediante aproximaciones multivariantes, siendo las seleccionadas para este estudio la regresión de mínimos cuadrados parciales y el análisis de correlación canónica.

La regresión de míminos cuadrados parciales (sPLS) permite modelar respuestas múltiples, al mismo tiempo que trata la multicolinealidad [23]. La integración mediante sPLS se basa en la relación entre dos matrices de datos X e Y. Su principal ventaja es que no está limitado a variables no correlacionadas, por lo que es muy eficiente cuando $p + q >> n$ [23].

La otra aproximación es el análisis de correlación canónica regularizado, o rCCA. Este tipo de integración rCCA modela una relación bidireccional para descubrir la correlación entre los dos conjuntos de datos, como el CCA. Este CCA busca combinaciones lineales de las variables (canónicas) para reducir las dimensiones de los conjuntos de datos, pero intentando maximizar la correlación entre las dos variables [24]. En el presente estudio, el número de variables es mucho mayor que el número de muestras ($p + q >> n$), por lo que se deberán elegir parámetros de regularización $\lambda 1$ y $\lambda 2$.

Con el objetivo de poder realizar estas aproximaciones en el lenguaje de programación R, se ha utilizado el paquete “mixOmics” de Bioconductor [25]. Permite la exploración, extracción, integración y visualización de grandes conjuntos de datos.

3.3.1. Lectura de los datos

En primer lugar, se redujo la lista de genes de RNA-seq, de 3319 a 395 genes, los cuales estaban diferencialmente expresados en el análisis de los datos de transcriptómica anteriormente realizado (Apartado 3.1).

Asimismo, esos 395 fueron seleccionados en base a su función biológica, estando estrechamente relacionados con el metabolismo del organismo de estudio, como son por ejemplo la glucólisis, ciclo de Krebs, ruta de las pentosas fosfato… Por este motivo se preveía que las relaciones de esos genes con los metabolitos, fueran remarcables.

La finalidad de esta reducción ha sido una mejor integración de los datos, debido a un menor número de variables implicadas, lo que permite acelerar el proceso, reduciendo el tiempo computacional, elevado para matrices con un gran número de genes, así como una mejor visualización de los gráficos obtenidos.

Para poder implementar los datos de transcriptómica en R ha sido necesario crear una tabla en formato xlsx en la que se sitúen los diferentes genes para analizar en las filas (Gene_ID), y en las columnas las diferentes muestras del estudio (LowA, Low B…), con la misma distribución que presentaba el archivo "Quantification.xlsx", siendo esta tabla de counts denominada "Metabolicgenes.xlsx".

Se cargaron los datos de transcriptómica y metabolómica en R, de la misma manera que se hicieron para los análisis individuales anteriores. Con los datos de transcriptómica, se creó el objeto de clase DESeqDataSet, para almacenar los conteos de lectura (mycounts) y las condiciones del estudio (metadata) tal y como se describió anteriormente en el análisis estadístico. Posteriormente se realizó la transformación logarítmica regularizada (rlog), y se creó una nueva matriz que conteía estos datos transformados (X). Para el posterior estudio, fue necesaria la matriz transpuesta.

3.3.2. Sustitución de los Missing Values

El paquete mixOmics permite trabajar con datos que contienen missing values. Una opción cuando se encuentran valores perdidos en los datos, es la eliminación de aquellas variables que contienen datos faltantes. Debido al número de missing values encontrados, y con el fin de evitar una reducción de la matriz de los metabolitos, no se

procedió a esta eliminación, sino que se sustituyeron los valores perdidos en el conjunto de datos, usando la función *nipals*, perteneciente al paquete mixOmics.

El algoritmo NIPALS (Nonlinear estimation by Iterative Partial Least Square) fue descrito por Wold en 1966 [26]. Fundamentalmente realiza una descomposición de la matriz de datos, mediante regresión. Su principal ventaja es que se puede realizar el análisis de componente con datos faltantes y obtener sus estimaciones a partir de la matriz de datos reconstituida. Es el método más comúnmente utilizado para calcular los componentes principales de un conjunto de datos [27].

3.3.3. Análisis preliminar con PCA

En primer lugar, se ha realizado un PCA de cada grupo de datos. El PCA se ejecuta para para explorar un solo tipo de datos ómicos e identificar las fuentes de variación más importantes, como se realizó en los análisis anteriores.

El primer componente principal explica la mayor variabilidad que se produce en los datos, y cada PC siguiente explica la siguiente mayor variabilidad. Se conservan los componentes principales que explican la varianza mayor. Esta es la razón por la que elegir el número de dimensiones o componentes (ncomp) es vital en el estudio [28].

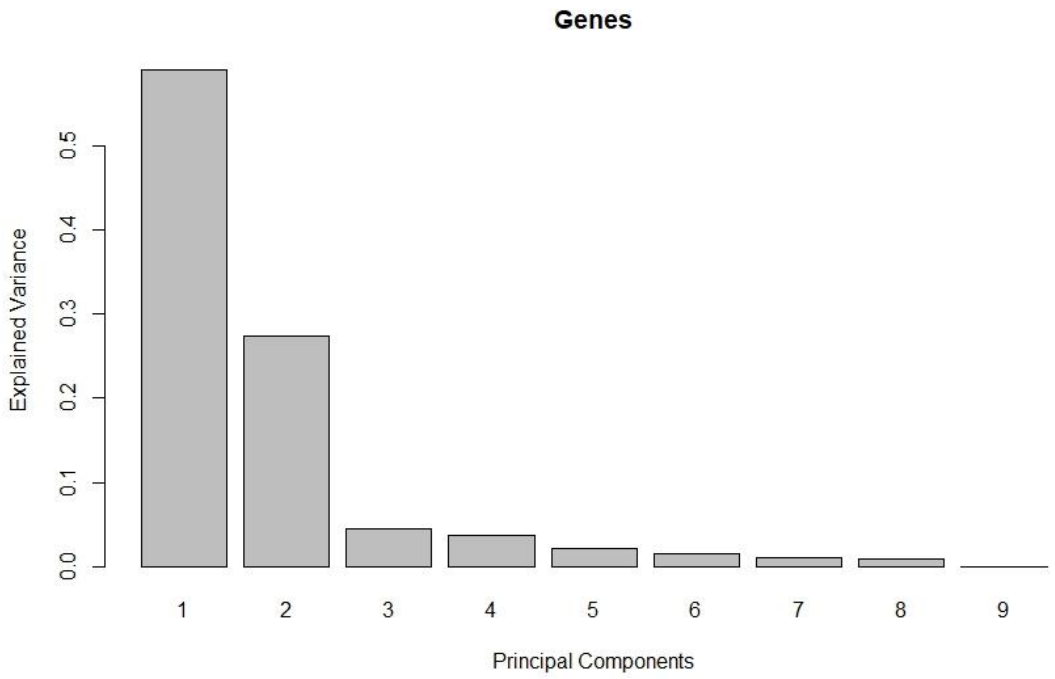


Figura 12. PCA de la expresión de los genes.

```
PC1      PC2      PC3      PC4      PC5      PC6
0.5895325 0.8625862 0.9070840 0.9434765 0.9649776 0.9796605
```

El resultado numérico de PCA muestra que el 86% de la varianza total se explica con 2 componentes. Las barras del gráfico de barras de la Figura 13 muestran la varianza explicada por componente.

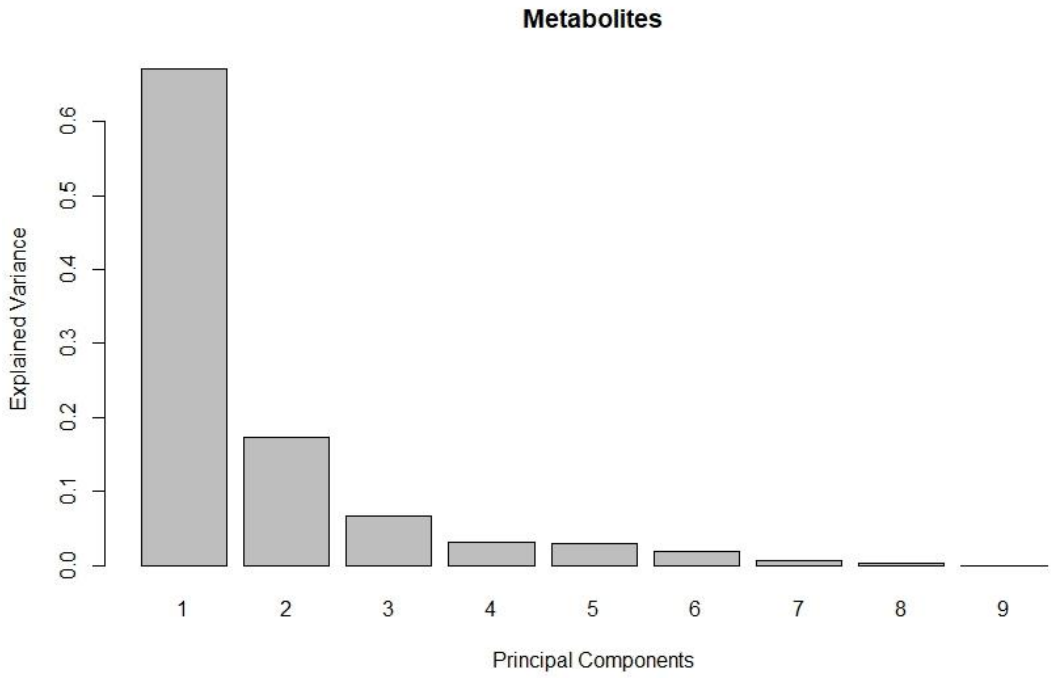


Figura 13. PCA de la cuantificación de los metabolitos.

```
PC1      PC2      PC3      PC4      PC5      PC6
0.7027590 0.8527009 0.9126315 0.9553382 0.9748902 0.9884801
```

El resultado numérico de PCA muestra que el 85% de la varianza total se explica con 2 componentes principales.

De este modo, se dedujo que el número de componentes principales que se debían seleccionar sería ncom=2.

Por último, se realizó un gráfico, de manera que las muestras se representaran como puntos colocados según su proyección en el subespacio más pequeño abarcado

por los componentes. Este tipo de gráfico (plotIndiv) (Figura 14 y 15) permite visualizar las similitudes diferencias entre muestras.

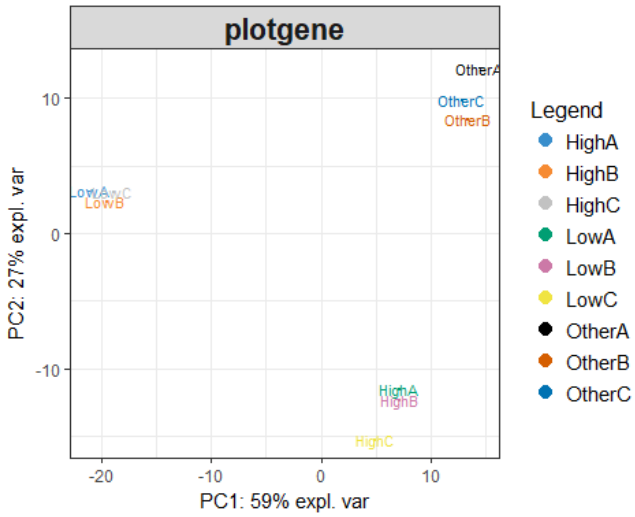


Figura 14. PlotIndiv del PCA de los genes. Cada muestra está representada en un color en la leyenda.

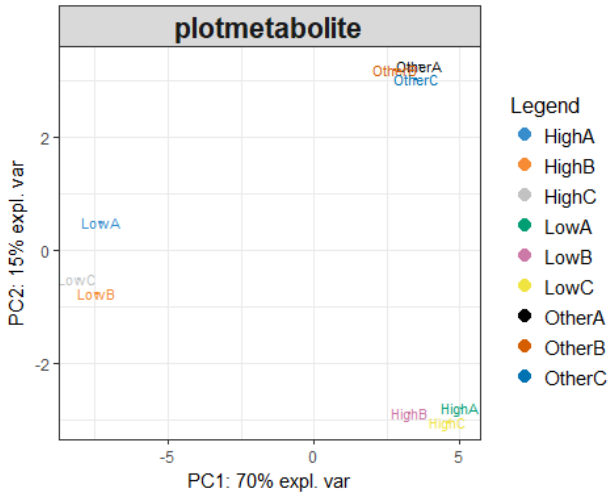


Figura 15. PlotIndiv del PCA de los metabolitos. Cada muestra está representada en un color en la leyenda.

Estos gráficos indican que las muestras están agrupas según la condición: low, high y other.

3.3.4 Integración mediante sPLS

sPLS combina la integración y la selección de variables, para maximizar la covarianza entre dos conjuntos de datos e identificar variables latentes. El análisis de componentes principales (Apartado 3.3.3), indica que el número de componentes a seleccionar es ncomp=2. Sin embargo, se debe realizar un estudio más profundo mediante la función *perf* de sPLS, para determinar el verdadero número de componentes de la aproximación.

Se ha seleccionado un número de componentes mayor, ncomp=5, de manera que después se produzca la restricción de los componentes. Además, el modo sPLS utilizado fue el determinado por el modelo de regresión (mode = "regression"), donde las variables X explicarán las variables Y [29]. El modo de regresión se puede aplicar cuando hay un conocimiento a priori sobre qué grupo de variables implica el otro grupo, en lugar de usar el modelo canónico. Se utiliza con frecuencia en estudios que implican datos ómicos, ya que el dogma de la biología molecular implica tres niveles funcionales principales (transcriptómica, proteómica y metabolómica), relacionados entre sí, por lo que fue el seleccionado en el presente trabajo.

La función *spls* contiene por tanto, ambos dataset de datos (X para genes e nipals.Y para los metabolitos), el número de componentes, y el tipo de aproximación. Además, esta función selecciona por defecto todas las variables presentes en el modelo, si se desea seleccionar las variables, es necesario indicarlo con un vector del tipo keepX y keepY. Mediante la función *perf* se procede a la validación cruzada basada en 10-fold, del modelo, con un número elevado de repeticiones (nrepeat = 100).

```
spls0 <- spls(X, nipals.Y, ncomp =5, mode = "regression")
tune.spls0 <- perf(spls0, validation = "Mfold", folds = 9, progressBar
= FALSE, nrepeat = 100)
```

Un componente PLS debe incluirse en el modelo si su valor es mayor o igual a 0.0975 [30], por lo que analizamos el Q2 gráficamente (Figura 16), este parámetro se utiliza para valorar la potencia predictiva del modelo PLS, al realizar cálculos de validación cruzada.

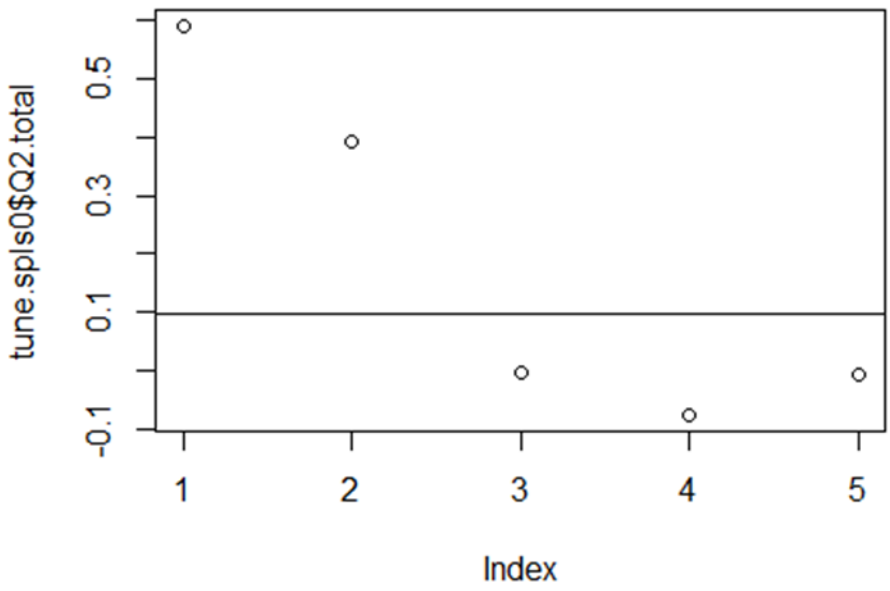


Figura 16. Gráfica de la validación del modelo spls mediante el parámetro Q2, con ncomp=5.

De este modo, el número de componentes seleccionados mediante spls ha sido ncomp=2, ya que son mayores a la línea de 0.0975, por lo que se diseña la nueva aproximación con este número de componentes:

```
spls <- spls(X, nipals.Y, ncomp =2, mode = "regression")
tune.spls <- perf(spls, validation = "Mfold", folds = 9, progressBar =
FALSE, nrepeat = 100)
```

Posteriormente, se representaron las muestras proyectadas en los componentes de sPLS utilizando la función plotIndiv ().

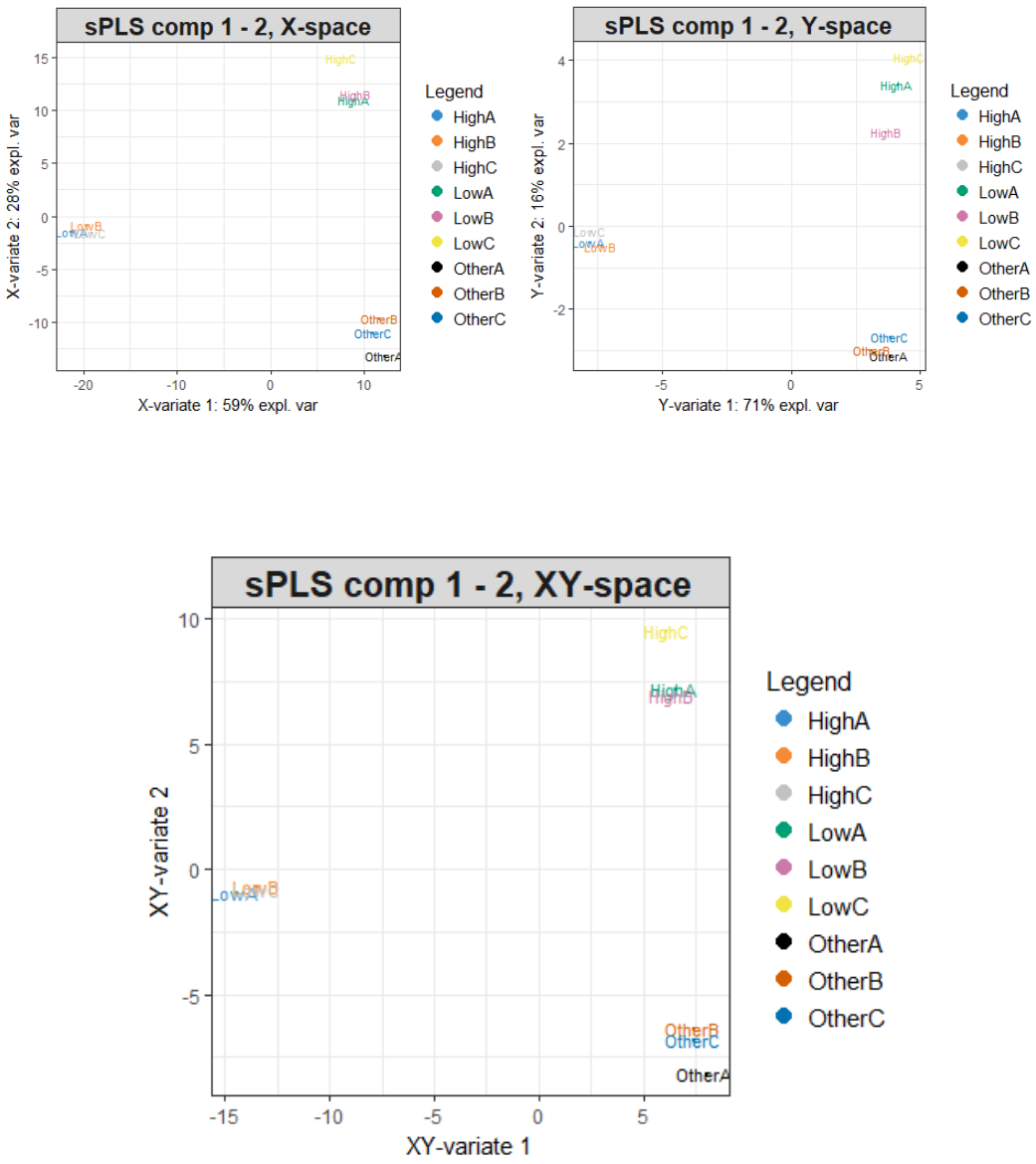


Figura 17. Diferentes gráficos plotIndiv de las muestras proyectadas en los distintos subespacios.

Los diagramas individuales muestran tres subespacios diferentes abarcados por la variable X, la variable Y y el subespacio medio en el que se promedian las coordenadas de los dos primeros subespacios (XY). Las proyecciones de los gráficos son muy parecidas (Figura 17).

Además, se ha realizado un diagrama de flechas (Figura 18), en el que cada flecha corresponde a una muestra. El inicio de la flecha indica la ubicación de la muestra en X, y la punta la ubicación de la muestra en Y. Las flechas cortas indican si ambos conjuntos de datos están muy de acuerdo y las flechas largas un desacuerdo entre

los dos conjuntos de datos. El diagrama presenta así, un acuerdo entre los dos conjuntos de datos, que podría ser menor si las dimensiones de ambos fuesen más parecidas.

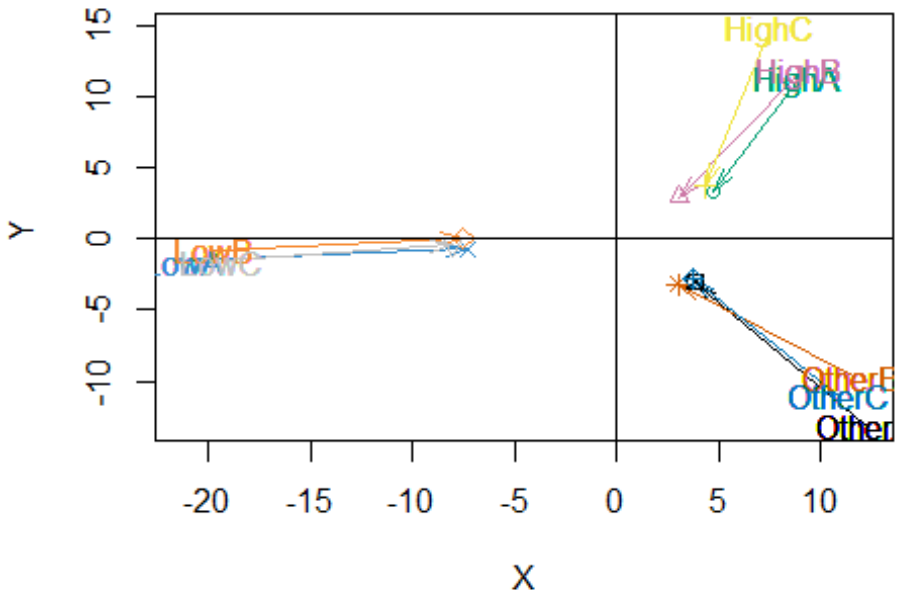


Figura 18. PlotArrow de las diferentes proyecciones. X (determinadas con el nombre de la muestra) e Y (correspondiente a los iconos situados en el extremo de la flecha).

Asimismo, las variables seleccionadas por el sPLS se pueden proyectar en un círculo de correlación. Estas proyecciones están dentro de un círculo de radio 1 centrado en el origen llamado círculo de correlación. Las variables correlacionadas se proyectan en la misma dirección desde el origen. Cuanto mayor es la distancia desde el origen, más fuerte es la asociación. Además se ha trazado otra circunferencia de radio 0.5 para revelar la estructura de correlación de las variables.

En el gráfico se observan numerosos metabolitos correlacionados a genes, ya que se encuentran cercanos al círculo exterior de radio 1. El investigador debe seleccionar un menor número de variables para visualizar aquellas correlaciones específicas de manera supervisada, según el interés generado tras la visualización "masiva" de los datos.

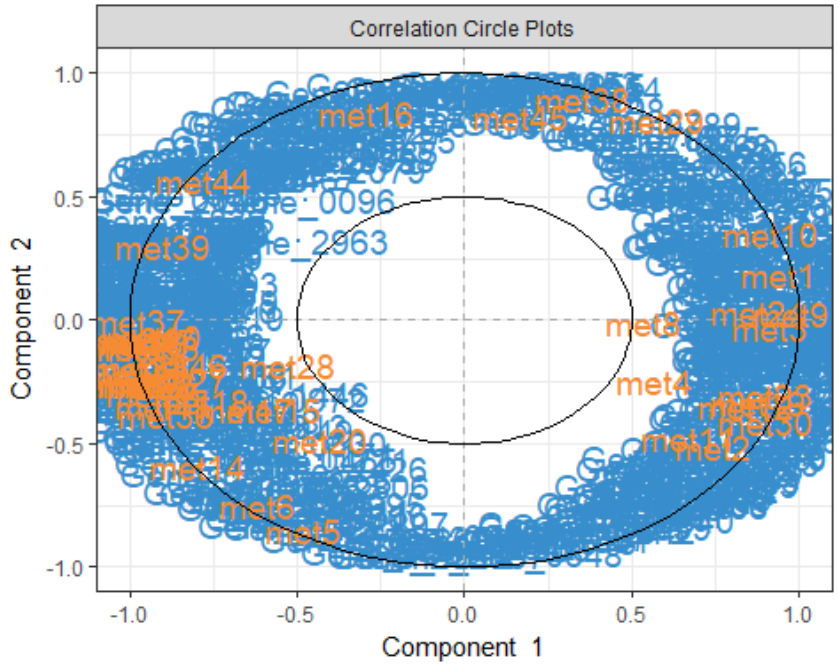


Figura 19. Gráfico de círculo de correlación entre los genes y metabolitos de las muestras mediante sPLS.

Mediante un gráfico de redes de correlación, se pueden observar estas mismas relaciones entre conjuntos de datos. Su principal ventaja es que se visualizan relaciones tanto directas (por ejemplo una mayor expresión de un gen supone una mayor concentración de su metabolito asociado), como indirectas (una mayor expresión de un gen puede suponer la no producción de cierto metabolito, o viceversa).

El gráfico de redes de correlación indica una serie de metabolitos (rosa) asociados otro determinado número de genes (blanco). La correlaciones directas se marcaron con líneas rojas, mientras la indirectas, con líneas verdes (Figura 20). Para una mejor visualización de las redes, se puede exportar este gráfico a la herramienta Cytoscape [31].

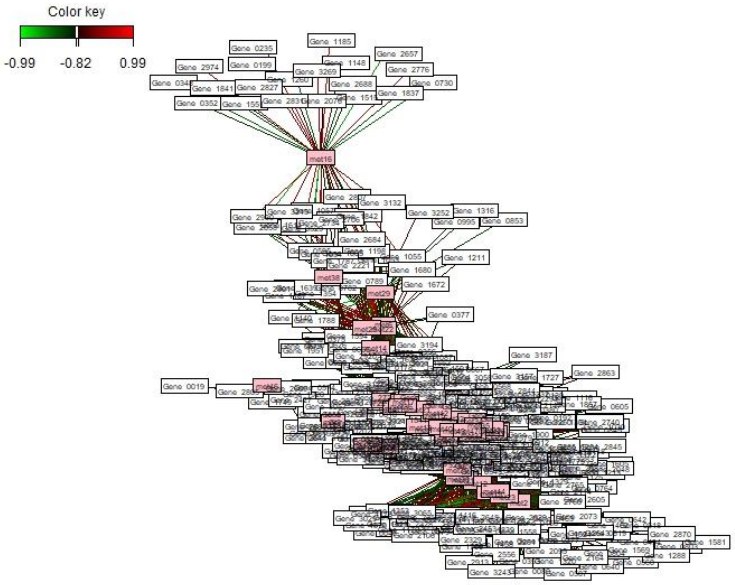


Figura 20. Gráfico de redes de correlación entre los genes (blanco) y metabolitos (rosa) de las muestras mediante sPLS. Las líneas rojas representarán una correlación positiva o directa, y las verdes negtiva o inversa.

Por último, mediante un heatmap similar al utilizado en el análisis de RNA-seq, se pueden agrupar las correlaciones entre datos, para observar aquellos grupos divididos jerárquicamente que estén más relacionados (rojo) o menos (azul) (Figura 21). De este tipo de gráficos, así como de los anteriores, se puede sacar numero información biológica interesante para el investigador.

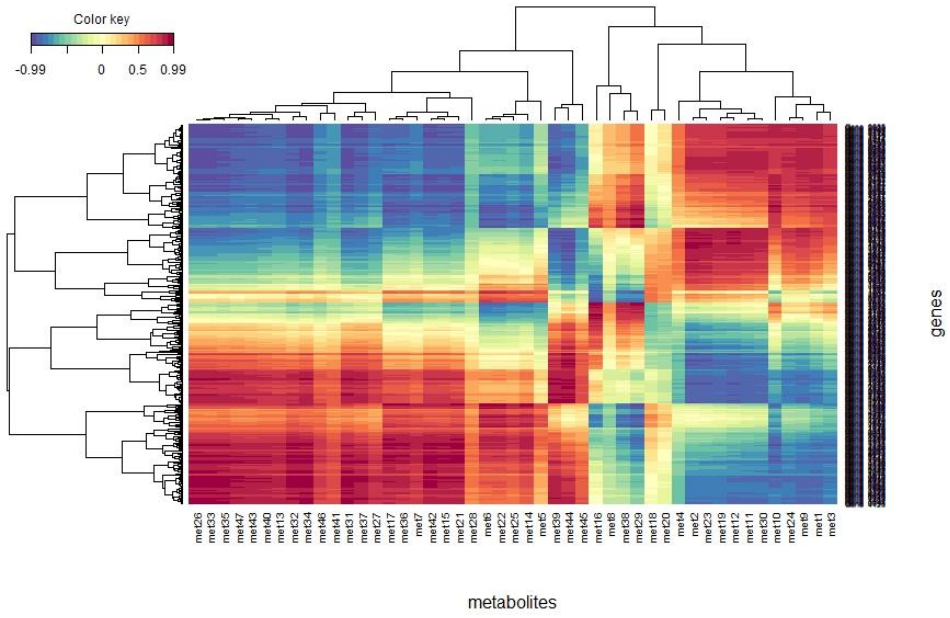


Figura 21. Heatmap de las correlaciones entre genes (eje Y) y metabolitos (eje X) mediante sPLS.

3.3.5. Integración mediante rCCA

El CCA es un enfoque no supervisado que se centra en maximizar la correlación entre los dos conjuntos de datos X e Y [32]. Consiste en la regularización de las matrices de covarianzas de X e Y al agregar un múltiplo (λ) de la identidad de la matriz [24], que es:

$$Cov(X) + \lambda 1I \text{ and } Cov(Y) + \lambda 2I.$$

Cuando ambas lambda son 0, es un modelo clásico de CCA (n> p + q). En el presente estudio, al ser las matrices mucho mayores al número de muestras, no es posible que estos valores sean nulos, por lo que se procedió a una estimación de estos parámetros mediante el método de validación cruzada. Como el número de muestras n es pequeño, se ha utilizado loo-cv.

Los datos fueron implementados ya en Rstudio en el apartado 3.3.4 (Integración mediante sPLS), por lo que no ha sido necesario realizar este proceso de nuevo.

En primer lugar, se procedió a visualizar la relación entre los datos X y nipals.Y, mediante una matriz de correlación. Estas correlaciones no fueron muy buenas (color azul) (Figura 22), ya que tenemos un número alto de variables en la muestra, y este método es ampliamente utilizado cuando n> p + q, es decir, es número de muestras es mayor que el número de variables, por lo que no sería la aproximación más adecuada.

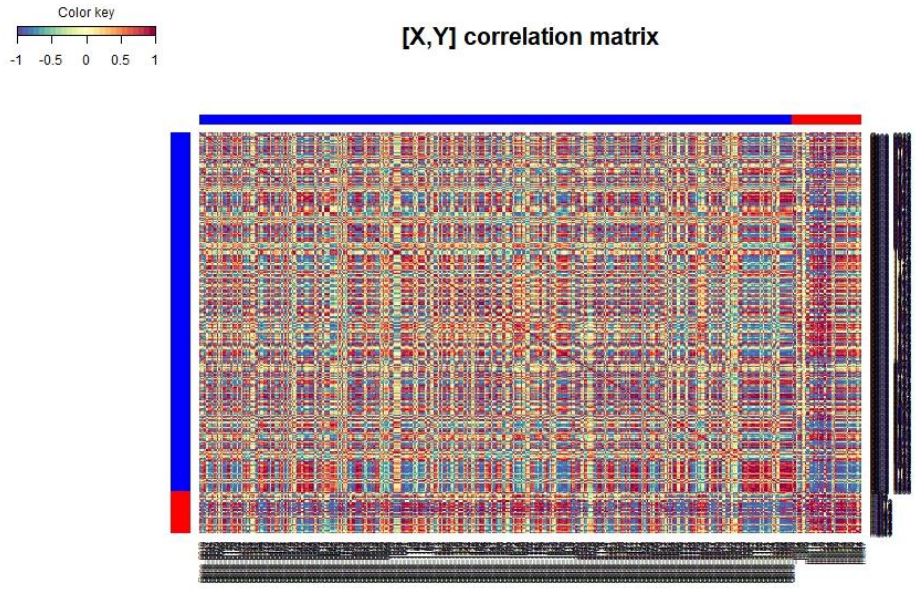


Figura 22. Matriz de correlación entre las matrices X e Y.

En primer lugar, se ajustaron los parámetros de regularización λ1 y λ2. Se utilizaron el procedimiento de validación cruzada (CV). Para ello, hay que definir los valores "grid", que son vectores numéricos que definen los valores de lambda1 y lambda2, donde se buscan estos parámetros, siempre con valores entre 0 y 1. Posteriormente se genera una matriz de celdas, seleccionando el tipo de validación cruzada (loo) y las matrices de los datos [25].

```
grid1 <- seq(0.0001, 1, length = 51)
grid2 <- seq(0.0001, 1, length = 51)
cv <- tune.rcc(X, nipals.Y, grid1 = grid1, grid2 = grid2, validation =
"loo")
```

Mediante la función *tune.rcc*, se calculan los puntajes de validación cruzada en una matriz bidimensional para determinar los valores óptimos de regularización (Figura 23). Este cálculo tiene un elevado coste computacional, mayor cuantas más opciones de celda se indique en el comando donde se obtienen los valors grid (length = 51).

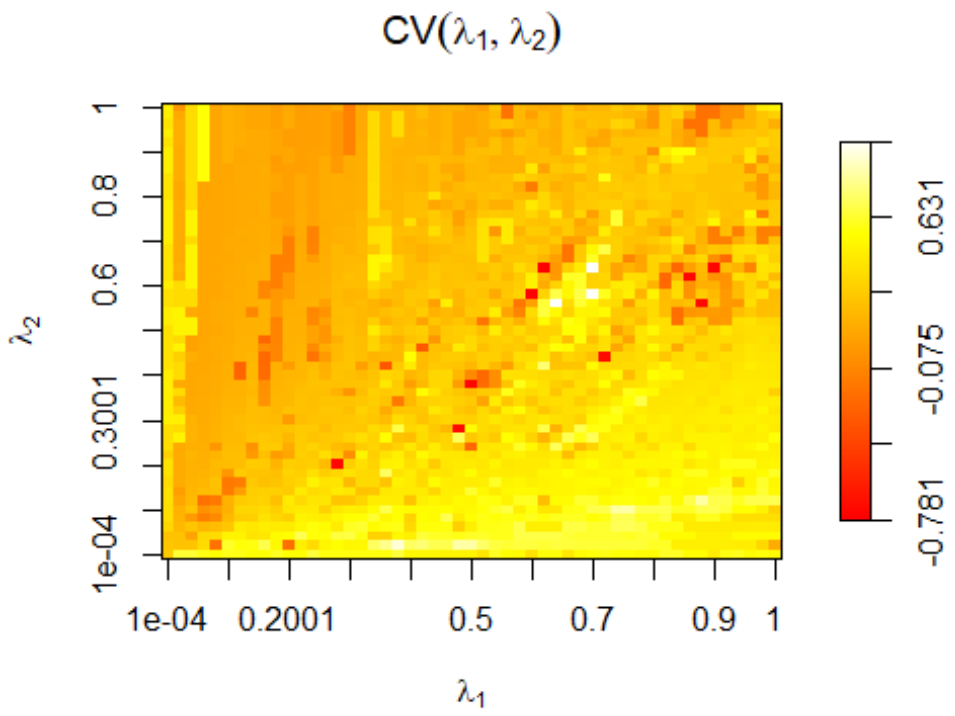


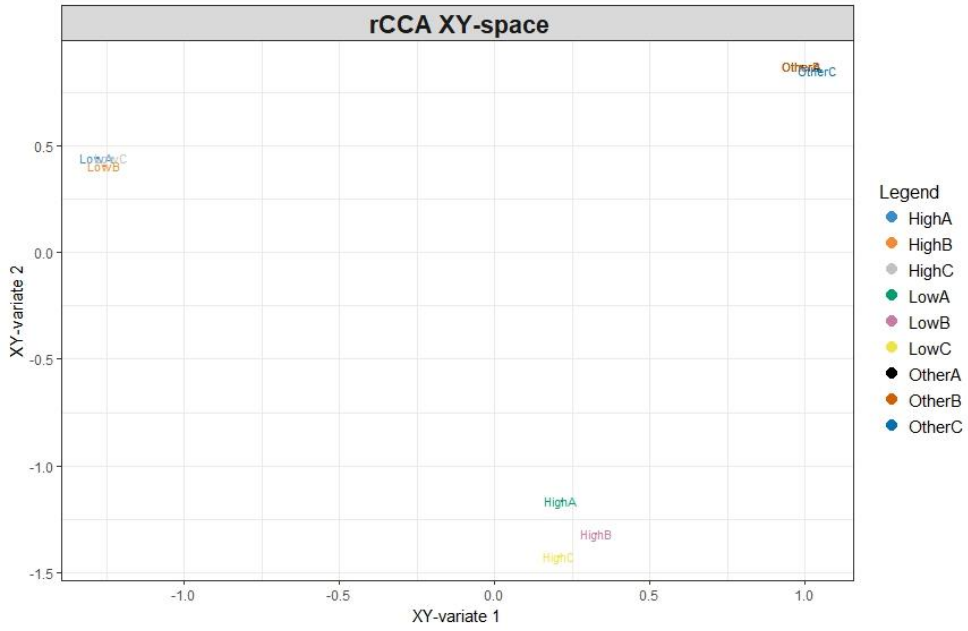
Figura 23. Gráfico de cuadrícula donde se indican los diferentes valores de lambda1 y lambda 2.

Así, se obtuvieron los siguientes valores de lambda, que posteriormente fueron sustituidos en la función *rcc*, la cual es la indicada para realizar la aproximación mediante rCCA [25]:

```
##   lambda1 =  0.70003
##   lambda2 =  0.580042
##  CV-score =  0.9805094

rcc <- rcc(X,nipals.Y, ncomp = 2,  lambda1 = 0.70003, lambda2 =
0.580042)
```

Una vez realizada la aproximación, se procede a representar las gráficas de los diagramas individuales, donde se muestran los diferentes subespacios de las proyecciones (Figura 24).



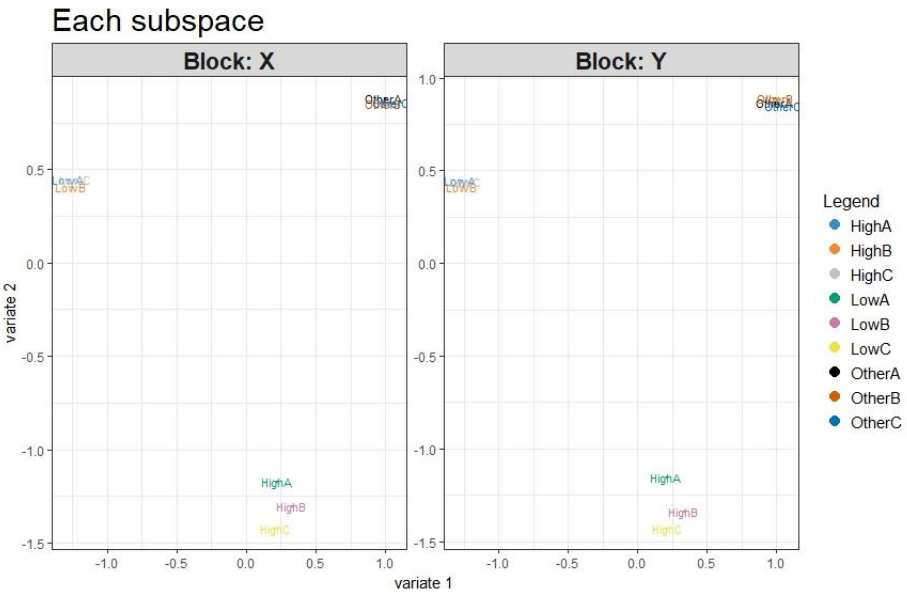


Figura 24. Gráfico de cuadrícula donde se indican los diferentes valores de lambdaa y lambda 2, mediante rCCA.

En el gráfico de correlación circular (Figura 25) se observan numerosos metabolitos correlacionados a genes, ya que se encuentran cercanos al círculo exterior de radio 1. Algunos tienen una correlación parecida a la diseñada mediante sPLS (por ej: met4, met8…), pero otros metabolitos difieren, por lo que ambas aproximaciones darán resultados diferentes que el investigador debe evaluar.

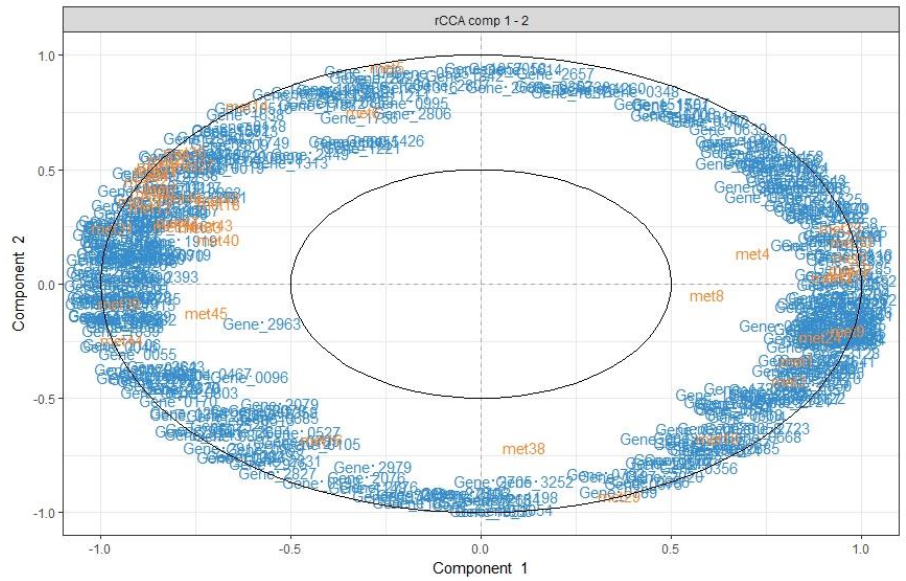


Figura 25. Gráfico de círculo de correlación entre los genes y metabolitos de las muestras, mediante rCCA.

El gráfico de redes de correlación indica una serie de metabolitos (rosa) asociados otro determinado número de genes (blanco). La correlaciones directas se marcaron con líneas rojas, mientras la indirectas, con líneas verdes (Figura 24). Existen numerosas similitudes respecto a ciertas correlaciones, como por ejemplo el met16, lo que indicaría que la aproximación se acerca a la aproximación realizada mediante sPLS.

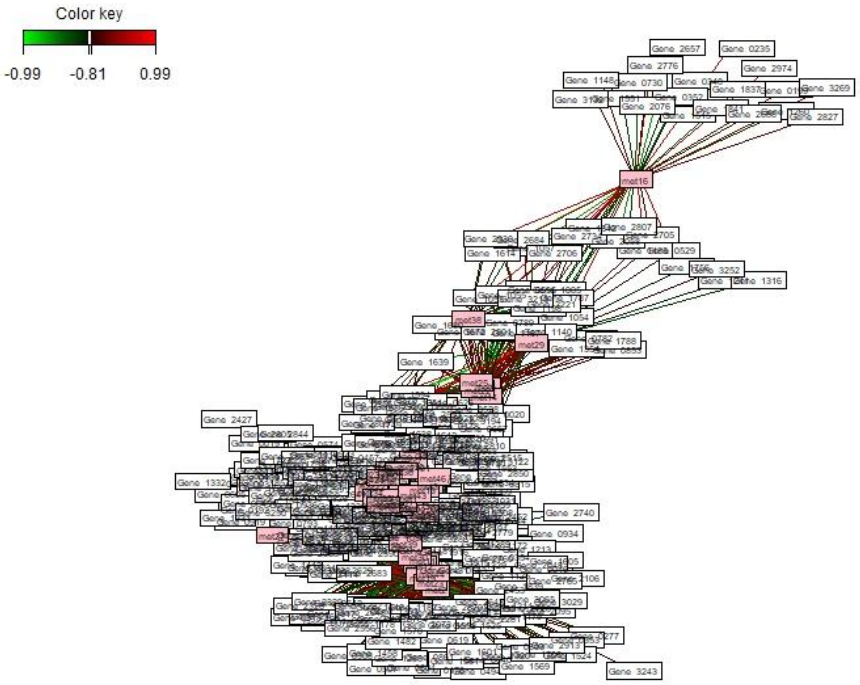


Figura 26. Gráfico de redes de correlación entre los genes (blanco) y metabolitos (rosa) de las muestras mediante rCCA. Las líneas rojas representarán una correlación positiva o directa, y las verdes negtiva o inversa.

Por último, mediante un heatmap agruparon las correlaciones entre datos, para observar aquellos grupos divididos jerárquicamente que estén más relacionados (rojo) o menos (azul) (Figura 27). Es necesaria una visualización más clara de este tipo de mapas, seleccionando un menor número de genes de interés, para sacar conclusiones bológicas de estos datos.

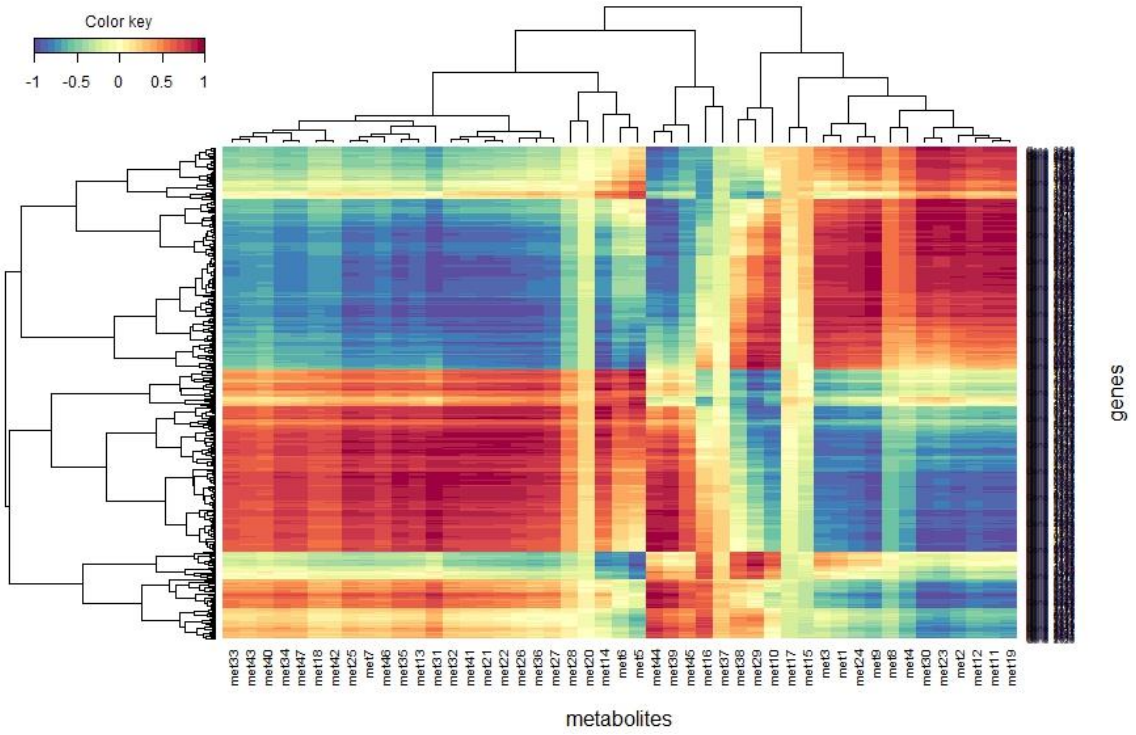


Figura 27. Heatmap de las correlaciones entre genes (eje Y) y metabolitos (eje X) mediante rCCA.

4. Conclusiones

4.1. Conclusiones generales

1. R es un lenguaje de programación ampliamente utilizado para el estudio de ómicas, tanto individuales como para su integración.

2. En la actualidad, existen numerosas maneras de integrar datos ómicos, mediante diversos enfoques. Es necesario un estudio exahustivo de cada uno de ellos para su utilización según los datos a analizar.

3. El desarrollo de informes mediante R-Markdown suponen una excelente visualización de los resultados obtenidos, con el fin de facilitar la interpretabilidad y corrección de los procesos por el tutor.

4. La estrategia desarrollada ha permitido aprender diversas metodologías de integración de datos, así como profundizar en los conocimientos estadísticos de estas aproximaciones, y a un uso más exhaustivo del lenguaje de programación R, elemento básico en el análisis de datos procedentes de estas nuevas tecnologías.

5. Tanto la aproximación mediante sPLS, como la obtenida por rCCA, consiguen ciertas similitudes en la integración de los datos. A pesar de que sPLS es una estrategia más fiable para la integración de ómicas con un elevado número de variables, sería bueno comparar ambas técnicas y estudiar aquellos puntos en común entre ambas aproximaciones.

4.2. Objetivos del trabajo

Los objetivos descritos en el trabajo fueron los siguientes, desglosados a continuación:

Objetivo 1: Analizar los datos de transcriptómica.
Objetivo 2: Analizar los datos de proteómica.
Objetivo 3: Integración de los datos de transcriptómica con los de proteómica.

Objetivo 1: Analizar los datos de transcriptómica.

1.1. Selección de datos de expresión para implementar en R.

1.2. Analizar y realizar las transformaciones necesarias sobre los datos de transcriptómica para poder integrarlos posteriormente.

Este proceso fue decisivo a la hora de seleccionar los datos del estudio. En concreto, se eligieron datos de RNA-seq anónimos de un grupo de investigación, tras un proceso de extensa revisión bibliográfica.

El hecho de que no fueran datos anteriormente publicados dificultó su implementación en R y su posterior análisis, para el que se siguió un procedimiento de R, basado en el uso del paquete *DESeq2* de Bioconductor.

Objetivo 2: Analizar los datos de metabolómica.

2.1. Selección de datos de metabolómica para implementar en R.

2.2. Analizar y realizar las transformaciones necesarias sobre los datos para poder integrarlos posteriormente.

El Objetivo 2 fue más difícil de cumplir y consumió más tiempo del esperado. Inicialmente, el conjunto de datos que se iba a analizar procedían de un estudio de proteómica, procedente de las mismas muestras que los datos de transcriptómica. Sin embargo, debido a dificultades en la obtención de estos datos, se procedió a su cambio por unos de metabolómica dirigida de las mismas muestras, obtenida mediante espectometría de masas, y se realizaron los análisis, mediante el paquete *limma* de Bioconductor.

Estos cambios en los datos del estudio no supusieron ningún impedimento a la hora de realizar el estudio, ya que seguimos teniendo datos de dos ómicas diferentes de las mismas muestras.

Aunque el proceso de adaptación de los datos fue difícil, los dos primeros objetivos han sido desarrollados a tiempo, cumpliéndose su entrega satisfactoriamente.

Objetivo 3: Integración de datos ómicos.

3.1. Integración de los datos de transcriptómica con los de metabolómica.

3.2. Integración de los datos mediante una segunda aproximación.

3.3. Comparación de los resultados de ambas aproximaciones.

3.4. Realización de un informe mediante R-Markdown.

En primer lugar se tuvo que realizar una adaptación adecuada de los datos del estudio, diferente al realizado en el análisis individual de las ómicas.

El Objetivo fue complicado de cumplir, ya que se utilizaron aproximaciones y métodos nuevos, no vistos en asignaturas cursadas en el Máster. Esto complicó el proceso de obtención de resultados, pero fue a su vez motivador, al tener que resolver ciertas incógnitas para superar los retos que iban surgiendo. De este modo, se consumió más tiempo del esperado, para alcanzar un análisis estadístico adecuado.

Además, se generaron unos ficheros HTML (ver Anexo), de los diferentes análisis realizados, obtenidos desde Rmd, que facilitaron la observación, evaluación y comentarios de los análisis realizados. Además, se adjuntan los ficheros Rmd tanto del análisis de las ómicas individuales (TFM_Analisis omicas), como de la integración (TFM_integracion), sin aquellas gráficas de elevado coste computacional, de manera que estos comandos solo estarían disponibles en el Anexo.

4.3. Análisis de la planificación y metodología

Los hitos propuestos y visualizados en el Diagrama de Gantt (Figura 2) fueron cumplidos en su totalidad, por lo que en principio no fue necesaria una desviación en el cronograma.

Los cambios que se tuvieron que realizar en los datos del estudio no supusieron ningún impedimento a la hora de realizalo, ya que seguían siendo datos de dos ómicas diferentes de las mismas muestras. El principal problema que presentaron los datos fue su procedencia, ya que son resultados experimentales de un grupo de investigación, debían ser anónimos y a veces no se dispuso de la información de manera directa, lo que dificultaba el ajuste del análisis al tiempo.

Sin embargo, trabajar con este tipo de datos ha ofrecido una visión más ajustada de la realidad, sin realizar el análisis con "datos modelo", ya implementados o ampliamente utilizados en otros estudios, debido a que no siempre se dispone de una información inmediata y de unos datos de calidad elevada cuando hay que realizar los análisis. Esto ha permitido el desarrollo de otro tipo de competencias transversales y motivación adicional.

Asimismo, la búsqueda biliográfica ha sido constante y determinante para la realización del trabajo. Ya existían conocimientos previos de análisis de datos de RNA-seq, gracias a la asignatura de Análisis de Datos Ómicos del Máster. Sin embargo, no se disponía de conocimiento previo para la integración de estos datos, por lo que la búsqueda de material y estudio de la bibliografía ha sido amplio y exhaustivo.

Por último, el trabajo a distancia ha sido adecuado, aunque difícil de adaptar al principio. El proceso ha tenido tanto una gran carga autónoma por parte del alumno, como una buena comunicación periódica con el profesor, cuyos tiempos de respuesta han sido siempre muy eficientes, algo gratamente positivo en un trabajo que se debe realizar a distancia. Todo ello junto a las aclaraciones de dudas e indicaciones pertinentes, ha sido clave para la obtención de este TFM.

4.4. Trabajo futuro

Además de realizar una integración de datos de transcriptómica y metabolómica, sería interesante poder obtener unos datos de proteómica para implementarlos en el estudio, realizando otra integración de datos de transcriptómica y proteómica.

Se realizaron distintas aproximaciones que permitieron desarrollar un método de integración correcto, esta información, junto al aprendizaje obtenido, podría permitir siendo el diseño posterior de otro tipo de integraciones, mejorar las aproximaciones realizadas, utilizar otro tipo de ómicas, incluso intentar realizar una integración de tres ómicas a la vez mediante un nuevo aprendizaje de otros paquetes de R, ya que se han obtenido las destrezas necesarias para poder alcanzar este tipo de objetivos, mediante el continuo estudio de la metodología y el aprendizaje.

5. Glosario

cDNA	ADN complementario
Cov	Covarianza
CV	Validación cruzada
FDR	False Discovery Rate
GC	Cromatografía de gases
HPLC	Cromatografía líquida de alta eficacia
LFC	Logaritmo del Fold Change
MLGs	Modelo Lineal Generalizado
MS	Espectometría de masas
Ncomp	Número de componentes
PCA	Análisis de Componentes Principales
rCCA	Análisis de Regresión Canónica
RNA-seq	Secuenciación de ARN
RPKM	Lecturas por kilobase de transcrito y millones de lecturas
SOLiD	Secuenciación por ligadura y detección de oligonucleótidos
sPLS	Regresión de Mínimos Cuadrados Parciales
TFM	Trabajo fin de Máster
NGS	Secuenciación de próxima generación
DNBS	Secuenciación de ADN nanoball

6. Bibliografía

1. **Wanichthanarak.K., Fahrmann. J., Grapov. D.** Genomic, Proteomic, and Metabolomic Data Integration Strategies. Biomark Insights. **2015**. 10: 1-6.

2. **Cavill. R., Jennen. D., Kleinjans. J., Briedé**. J.J. Transcriptomic and metabolomic data integration. Brief Bioinform. **2016**; 17(5):891-901.

3. **Spicker, J., Brunak, S., Frederiksen K Toft, H.** Integration of Clinical Chemistry, Expression, and Metabolite Data Leads to Better Toxicological Class Separation *Toxicological Sciences.* **2008**; 444-454.

4. **Krumsiek, J., Bartel, J., Theis, F**. Computational approaches for systems metabolomics. *Current Opinion in Biotechnology***. 2016**; 198-206.

5. **Griffin, J., Bonney, S., Mann, C., Hebbachi, A., Gibbons, G., Nicholson, J., Shoulders, C., Scott, J.** An integrated reverse functional genomic and metabolic approach to understanding orotic acid-induced fatty liver; **2004**.

6. **Cavill, R., Kamburov, A., Ellis, J., Athersuch, T., Blagrove, M., Herwig, R., Ebbels, T., Keun, H**. Consensus-Phenotype Integration of Transcriptomic and Metabolomic Data Implies a Role for Metabolism in the Chemosensitivity of Tumour Cells. PLOS Computational Biology; **2011**.

7. **Wang, Z., Gerstein, M., Snyder, M.** RNA-Seq: a revolutionary tool for transcriptomics. *Nat Rev Genet.* **2009**; 10(1):57-63.

8. **Kukurba, K., Montgomery, S**. RNA Sequencing and Analysis. *Cold Spring Harb Protoc*. **2015**(11): 951-969.

9. **Kulski Jerzy**. Next Generation Sequencing - Advances, Applications and Challenges. **2016**.

10. **Zhang, A., Sun, H., Wang, P., Han, Y., Wang, X.** Modern analytical techniques in metabolomics analysis. *Analyst.* 2**012**;137(2):293-300.

11. **Hardin, J., Hilbe, J.** Generalized Linear Models and Extensions. *StataPress*. **2007**.

12. **Benjamini, Y., Hochberg, Y.** Controlling the false discovery rate: a practical and powerful approach to multiple testing . *J R Stat Soc Ser B Methodol*. **1995**; 57: 289-300.

13. **Love, M.I., Huber, W., Anders, S.** Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*. **2014;** 15:550.

14. **Ritchie, M.E., Phipson, B., Wu, D., Hu, Y., Law, C.W., Shi, W., Smyth, G.K.** Limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res*. **2015;** 20:43(7):

15. **Phipson, B., Lee, S., Majewski, I.J., Alexander, W.S., Smyth, G.K**. Robust hyperparameter estimation protects against hypervariable genes and improves power to detect differential expression. *Annals of Applied Statistics*. **2016**; 10(2), 946-963.

16**. Law, C.W., Chen, Y., Shi, W., Smyth, G.K.** Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biology*. **2014.**

17. **Ritchie, M.E., Phipson, B., Wu, D., Hu, Y., Law, C.W., Shi, W., Smyth, G.K.** Limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res*. **2015;** 20:43(7):47.

18. **Gordon, K,. Matthew, S., Ritchie, M., Thorne, N., Wettenhall, J., Shi, W., Hu, Y** Package ‘limma’**. 2018**.

19. **Gordon, K,. Matthew, S., Ritchie, M., Thorne, N., Wettenhall, J., Shi, W., Hu, Y.** Linear Models for Microarray and RNA-Seq Data User’s Guide**. 2017**.

20. **Dudoit, R., Yang. Y., Callow, M., Speed, T**. Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *Statistica Sinica*. **2002**; 111–39.

21. **Law C.W., Alhamdoosh, M., Su, S., Smyth G. K., Ritchie, M.E.** RNA-seq analysis is easy as 1-2-3 with limma, Glimma and edgeR. *F1000Res*. **2016**; 5: 1408.

22. **Wold, S., Sjöström, M., and Eriksson, L.** Pls-regression: a basic tool of chemometrics. Chemometrics and intelligent laboratory systems. **2001**; 58(2), 109–130.

23. **Palermo, G.,1 Piraino, P., Zucht H.** Performance of PLS regression coefficients in selecting variables for each response of a multivariate PLS for omics-type data. *Appl Bioinforma Chem*. **2009**; 2: 57–70.

24. **González, I., Déjean, S., Martin, P.G. and Baccini, A.** CCA: An R package to extend canonical correlation analysis. *Journal of Statistical Software*. **2008**; 23(12):1-

25. **Rohart, F., Gautier, B., Singh, A., Lê Cao, K.** MixOmics: An R package for 'omics feature selection and multiple data integration. *PLoS Comput Biol*. **2017**; 13(11): e1005752.

26. **Wold H.** Multivariate Analysis. *Academic Press*. 1966.

27. **Risvik, H.** Principal Component Analysis (PCA) & NIPALS algorithm. **2007**.

28. **Wold H.** Path models with latent variables: The NIPALS approach. In: Blalock H. M. et al. (editors*).* Quantitative Sociology: International perspectives on mathematical and statistical model building. *Academic Press*. 1975; 307-357.

29. **Lê Cao, K. A., Rossouw, D., Robert-Granié, C., Besse, P**. A sparse PLS for variable selection when integrating Omics data. Statistical Applications in Genetics and Molecular Biology. *Stat Appl Genet Mol Biol*. **2008**; 7(1):Article 35

30. **Tenenhaus, M.** La régression PLS: théorie et pratique. **1998**. *Editions Technip*.

31. **Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N., Schwikowski, B., Ideker, T.** Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res*. **2003**;13(11) :2498-504.

32. **Leurgans, S.E., Moyeed, R.A. and Silverman, B.W.** 1993. Canonical correlation analysis when the data are curves. Journal of the Royal Statistical Society. *Series B (Methodological).* **2003;** 725-740.

7. Anexos

Listado de páginas HTML con los resultados del presente trabajo.

Análisis de ómicas individuales:

file:///C:/Users/Lourdes/Documents/TFM_Analisis_omicas.html

Integración de datos ómicos (sin gráficos):

file:///C:/Users/Lourdes/Desktop/TFM_integracion.html

Listado de comandos utilizados en el presente trabajo.

INSTALACIÓN DE PAQUETES

```
#Instalación de los paquetes necesarios
source("http://Bioconductor.org/biocLite.R")
biocLite('DESeq2')
biocLite('limma')
biocLite('mixOmics')
#Carga de los paquetes utilizados en el estudio
library(readxl)
library('limma')
library('pcaMethods')
library("pheatmap")
library("RColorBrewer")
library ('DESeq2')
library ('mixOmics')
```

ANÁLISIS DE LAS ÓMICAS

Análisis de datos de RNA-seq

1) Lectura de datos

```
mycounts <- read_excel("C:/Users/Lourdes/Desktop/Quantification.xlsx")
metadata <- read_excel("C:/Users/Lourdes/Desktop/metadata.xlsx")
```

```
#Visualización
View(metadata)
View(mycounts)
#Creación de los dataframe
mycounts <- as.data.frame(mycounts)
metadata <- as.data.frame(metadata)
head(mycounts)
head(metadata)
#Hacer coincidir los nombres de las columnas
class(mycounts)
class(metadata)
names(mycounts)[-1]
metadata$Name
names(mycounts)[-1]==metadata$Name
```

2) Análisis mediante DeSeq2

```
dds <- DESeqDataSetFromMatrix(countData=mycounts,  colData=metadata,
design=~Condition,  tidy=TRUE)
dds1 <- DESeq(dds)
```

3) Transformación de los datos para su visualización

```
logData <- rlog(dds, blind=F)
head(assay(logData), 10)
DeseqMatrix <- estimateSizeFactors(dds)
par( mfrow = c( 1, 2) )
plot(log2(counts(DeseqMatrix, normalized=TRUE)[,1:2]+1),pch=19, cex=0.1)
plot(assay(logData)[,1:2],pch=19, col="red", cex=0.1)
```

4) Análisis de agrupamientos

```
distancias<- dist( t( assay(logData) ) )

distancias.matrix <- as.matrix(distancias)

rownames(distancias.matrix) <- paste(logData$Condition,sep="-" )

colnames(distancias.matrix) <- paste(logData$Condition,sep="-" )

colors <- rev(brewer.pal(5, "Reds"))

pheatmap(distancias.matrix,clustering_distance_rows=distancias,clustering_distance_
cols=distancias,col=colors)
```

5) Análisis de componentes principales

```
plotPCA(logData,  intgroup = 'Condition')
```

6) Análisis de expresión diferencial

```
#Low_vs_High

res1 <- results(dds1, contrast=c("Condition","high","low"), alpha = 0.05)

summary(res1)

dim(res1)

resadj1 = subset(res1, padj < 0.05)

summary(resadj1)

sum(res1$padj <= 0.05, na.rm=TRUE)

top_genes1 <- resadj1[order(resadj1$log2FoldChange),]

top_genes_DESeq2_1 <- rownames(top_genes1)[1:100]

#High_vs_other

res2 <- results(dds1, contrast=c("Condition","high","other"), alpha = 0.05)

summary(res2)

dim(res2)

resadj2 = subset(res2, padj < 0.05)

summary(resadj2)

sum(res2$padj <= 0.05, na.rm=TRUE)
```

```
top_genes2 <- resadj2[order(resadj2$log2FoldChange),]

top_genes_DESeq2_2 <- rownames(top_genes2)[1:100]
```

7) Obtención de los genes up y down regulados

```
genes_up1<-as.data.frame(subset(top_genes1, log2FoldChange >= 1))

dim(genes_up1)

genes_down1<-as.data.frame(subset(top_genes1, log2FoldChange <= -1))

dim(genes_down1)

genes_up2<-as.data.frame(subset(top_genes2, log2FoldChange >= 1))

dim(genes_up1)

genes_down2<-as.data.frame(subset(top_genes2, log2FoldChange <= -1))

dim(genes_down1)
```

8) Gráfico de diferencia de medias

```
plotMA(res1, ylim = c(-10, 10))

plotMA(res2, ylim = c(-10, 10))
```

9) Gráficos

```
#Volcanoplot 1

colnames(resadj1)

plot(resadj1$log2FoldChange,-log10(resadj1$padj), cex =0.5, xlim = c(-8, 8), ylim = c(0, 60), xlab = "log2 fold change", ylab = "Log10 (pvalue")

with(subset(resadj1, padj<0.05 & abs(log2FoldChange)>=1), points(log2FoldChange, -log10(padj), pch=18, col="orange"))

with(subset(resadj1, padj<0.05 & abs(log2FoldChange)<=-1), points(log2FoldChange, -log10(padj), pch=18, col="green"))

abline(v=2,col="orange")

abline(v=-2,col="green")

#Volcanoplot 2

colnames(resadj2)
```

```
[1] "baseMean"      "log2FoldChange" "lfcSE"         "stat"

[5] "pvalue"        "padj"

plot(resadj2$log2FoldChange,-log10(resadj2$padj), cex =0.5, xlim = c(-8, 8), ylim = c(0, 60), xlab = "log2 fold change", ylab = "Log10 (pvalue")

with(subset(resadj2, padj<0.05 & abs(log2FoldChange)>=1), points(log2FoldChange, -log10(padj), pch=18, col="blue"))

with(subset(resadj2, padj<0.05 & abs(log2FoldChange)<=-1), points(log2FoldChange, -log10(padj), pch=18, col="green"))

abline(v=2,col="orange")

abline(v=-2,col="green")

# Heatmap

top100VarGenes <- head(order(rowVars(assay(logData)), decreasing = TRUE), 100)

mat  <- assay(logData)[ top100VarGenes, ]

mat  <- mat - rowMeans(mat)

anno <- as.data.frame(colData(logData))

pheatmap(mat, annotation_col = anno, cex=0.7)
```

Análisis de datos de espectometría de masas

1) Lectura de datos

```
data1 <- read.csv2("C:/Users/Lourdes/Desktop/metabolites.csv", row.names=1)

data= t(data1)
```

2) Creación de la matrix de diseño

```
group=c("Low","Low","Low","High","High","High", "Other", "Other", "Other")

design <- model.matrix( ~ 0 + group)

colnames(design)<-c("High","Low", "Other")

rownames(design)<-c("LowA","LowB","LowC","HighA","HighB","HighC",

                    "OtherA", "OtherB", "OtherC")

print(design)
```

3) Indicación de las comparaciones

```
cont.matrix <- makeContrasts(low_vs_high = High - Low, high_vs_other =
                        Other - High, levels = colnames(design))
print(cont.matrix )
```

4) Ajuste del modelo

```
fit<-lmFit(data, design)
fit.main<-contrasts.fit(fit, cont.matrix)
fit.mainB<-eBayes(fit.main)
```

5) Visualización de los resultados

```
results <- decideTests(fit.mainB)
summary(results)
topTab.low_vs_high <- topTable(fit.mainB, number=nrow(fit.mainB),
coef="low_vs_high", adjust="fdr")
topTab.high_vs_other <- topTable(fit.mainB, number=nrow(fit.mainB),
                        coef="high_vs_other", adjust="fdr")
# Volcano plot
volcanoplot(fit.mainB, coef = 1, highlight = 10, names = fit.main$ID,
        main =paste('Differentially expressed metabolites',
                colnames(cont.matrix)[1], sep = '\n'))
# Diagrama de Venn
vennDiagram(results, circle.col=c("blue", "salmon"))
```

INTEGRACIÓN

1) Lectura de datos

```
mycounts<- read_excel("C:/Users/Lourdes/Desktop/Metabolicgenes.xlsx")
View(mycounts)
metadata <- read_excel("C:/Users/Lourdes/Desktop/metadata.xlsx")
```

```
View(metadata)

metabolomica <- read.csv2("C:/Users/Lourdes/Desktop/metabolites.csv",
row.names=1)

mycounts <- as.data.frame(mycounts)

metadata <- as.data.frame(metadata)

head(mycounts)

head(metadata)

dds <- DESeqDataSetFromMatrix(countData=mycounts, colData=metadata,

design=~Condition,  tidy=TRUE)

logData <- rlog(dds, blind=F)

X = assay(logData)

X = t(X)

Z <- metadata

Y <- metabolomica
```

2) Sustitución de los Missing Values

```
na.row <- sample(1:nrow(Y), replace = TRUE)

na.col <- sample(1:ncol(Y), replace = TRUE)

Y.na <- as.matrix(Y)

Y.na[cbind(na.row, na.col)] <- NA

sum(is.na(Y.na))

nipals.Y = nipals(Y.na, reconst = TRUE, ncomp = 9)$rec

id.na = is.na(Y.na)

nipals.Y[!id.na] = Y[!id.na]

nipals.Y[id.na]

Y[id.na]
```

3) Integración mediante sPLS

Análisis preliminar con PCA

```
#PCA genes

pca.gene <- pca(X, ncomp = 9, center = TRUE, scale = TRUE)

pca.gene

#PCA metabolitos

pca.metabolite <- pca(nipals.Y, ncomp = 9, center = TRUE, scale = TRUE)

pca.metabolite

#Gráficos

plot(pca.gene, main="Genes")

plot(pca.metabolite, main="Metabolites")

plotIndiv(pca.gene, comp = c(1, 2), group = Z[, 1], ind.names = Z[, 1], legend = TRUE,
title = 'plotgene')

plotIndiv(pca.metabolite, comp = c(1, 2), group = Z[, 1],ind.names = Z[, 1], legend =
TRUE, title = 'plotmetabolite')

#sPLS

spls0 <- spls(X, nipals.Y, ncomp =5, mode = "regression")

tune.spls0 <- perf(spls0, validation = "Mfold", folds = 9, progressBar = FALSE, nrepeat =
100)

plot(tune.spls0$Q2.total)

abline(h = 0.0975)

tune.spls0$Q2.total

spls <- spls(X, nipals.Y, ncomp =2, mode = "regression")

tune.spls <- perf(spls, validation = "Mfold", folds = 9, progressBar = FALSE, nrepeat =
100)

plot(tune.spls$Q2.total)

abline(h = 0.0975)

#Gráficos de las muestras

plotIndiv(spls, comp = 1:2, rep.space= 'X-variate', group = Z[, 1],

        ind.names = Z[, 1],

        legend = TRUE, title = 'sPLS comp 1 - 2, X-space')
```

```
plotIndiv(spls, comp = 1:2, rep.space= 'Y-variate', group = Z[, 1],
        ind.names = Z[, 1],
        legend = TRUE, title = 'sPLS comp 1 - 2, Y-space')


plotIndiv(spls, comp = 1:2, rep.space= 'XY-variate', group = Z[, 1],
        ind.names = Z[, 1],
        legend = TRUE, title = 'sPLS comp 1 - 2, XY-space')
plotArrow(spls, ind.names= Z[, 1], position.names='start',group= Z[, 1], abline = TRUE)
#Gráficos de variables
plotVar(spls, comp =1:2)
color.edge <- color.GreenRed(50)
network(spls, comp = 1:2, shape.node = c("rectangle", "rectangle"), color.node =
c("white", "pink"), color.edge = color.edge)
cim(spls, comp = 1:2, xlab = "metabolites", ylab = "genes", margins = c(7, 7))
```

4) Integración mediante rCCA

```
imgCor(X, nipals.Y)
#Método CV
grid1 <- seq(0.0001, 1, length = 51)
grid2 <- seq(0.0001, 1, length = 51)
cv <- tune.rcc(X, nipals.Y, grid1 = grid1, grid2 = grid2, validation = "loo")
rcc <- rcc(X,nipals.Y, ncomp = 2,  lambda1 = 0.70003, lambda2 = 0.580042)
#Gráficos de las muestras
plotIndiv(rcc, comp = 1:2, ind.names = Z[, 1],
        group = Z[, 1], rep.space = "XY-variate",
        legend = TRUE, title = ' rCCA XY-space')


plotIndiv(rcc, comp = 1:2, ind.names = Z[, 1],
        group = Z[, 1],
```

```
    legend = TRUE, title = 'Each subspace')
```

#Gráficos de variables

```
plotVar(rcc, comp = 1:2, cutoff = 0.5, var.names = c(TRUE, TRUE),cex = c(4, 4), title = '
rCCA comp 1 - 2')

network(rcc, comp = 1:2, shape.node = c("rectangle", "rectangle"), color.node =
c("white", "pink"), color.edge = color.edge)

cim(rcc, comp = 1:2, xlab = "metabolites", ylab = "genes",

  margins = c(5, 6))
```