Escola Tècnica Superior d'Enginyeria ETSE-UV

Máster Universitario en Bioinformática

Trabajo de Fin de Máster

MetaFun: Herramienta web para la integración y caracterización funcional de estúdios ómicos con técnicas de metaanálisis

autor:
Pablo Malmierca Merlo

tutores:
Francisco García García
Marta Hidalgo García
Guillermo Ayala Gallego

Septiembre, 2019

VNIVERSITAT DE VALÈNCIA

Escola Tècnica Superior d'Enginyeria ETSE-UV

MÁSTER UNIVERSITARIO EN BIOINFORMÁTICA

TRABAJO DE FIN DE MÁSTER

METAFUN: HERRAMIENTA WEB PARA LA INTEGRACIÓN Y CARACTERIZACIÓN FUNCIONAL DE ESTÚDIOS ÓMICOS CON TÉCNICAS DE METAANÁLISIS

AUTOR:
PABLO MALMIERCA MERLO

TUTORES:
FRANCISCO GARCÍA GARCÍA
MARTA HIDALGO GARCÍA
GUILLERMO AYALA GALLEGO

TRIBUNAL:

PRESIDENTE/A:

VOCAL 1:

VOCAL 2:

FECHA DE DEFENSA:

CALIFICACIÓN:

Índice

Índice de figuras

RESUMEN

Hoy en día la comunidad investigadora está generando un gran volumen de datos clínicos y biológicos procedentes de sus experimentos. Muchas de las preguntas de investigación podrían ser resueltas con el uso de herramientas apropiadas para el análisis de estos datos. Estas preguntas pueden estar relacionadas con procesos biológicos habituales en un estudio transcriptómico en el ámbito farmacológico o con rutas de señalización sobrerrepresentadas en alguna enfermedad, entre otras.

Las técnicas de metaanálisis centradas en la funcionalidad permiten combinar la información de diferentes estudios, con un diseño experimental similar, para responder al tipo de preguntas mencionadas anteriormente.

En este trabajo se ha desarrollado *MetaFun*, una herramienta web cuyo objetivo es proporcionar un recurso bioinformático a la comunidad científica que permita una mejor comprensión de los resultados obtenidos en la integración de diversos estudios, con técnicas de metaanálisis.

Palabras clave: Metaanálisis, Web, Transcriptómica, Enriquecimiento funcional.

ABSTRACT

Nowadays, the research community is generating a great volume of clinical and biological data from their experiments. There are a lot of questions that could be answered with the use of the right tools for the analysis of this data. This questions could be related to common biological processes in a particular transcriptomic study in pharmacology or overrepresented pathways in a disease, among others.

The meta-analysis techniques by function allow the combination of information of several studies, with a similar experimental design, in order to answer these questions above.

In this work, we introduce *MetaFun*, a web tool whose objective is to allow, to every single interested researcher in the community, to use meta-analysis techniques in their research aiming to a better comprehension of their results.

Keywords: Meta-analysis, Web, Transcriptomics, Functional enrichment.

1.
INTRODUCCIÓN

El inmenso avance tanto tecnológico como científico, así como el abaratamiento de los costes de los experimentos relacionados con las técnicas ómicas, han conseguido que la comunidad investigadora genere una inmensa cantidad de datos.

Con tantos datos disponibles de manera pública en los distintos repositorios como *GEO*, *ArrayExpress o TCGA,* cabe plantearse la necesidad de encontrar formas de combinar los diferentes estudios en pos de una mejor comprensión de los resultados obtenidos por cada grupo investigador. El metaanálisis es una técnica estadística basada en la combinación de resultados de diversos estudios seleccionados en función de distintos criterios de inclusión. Algunos ejemplos de estos criterios, en el ámbito de la bioinformática pueden ser, por ejemplo, la cantidad de muestras o la calidad de los metadatos adjuntos a los datos del estudio. Cuando los estudios han sido seleccionados y analizados individualmente, se procede a realizar una integración de la variable de interés entre los distintos estudios.

El metaanálisis, por tanto, permite comparar las variables de interés una vez que están en la misma escala, consiguiendo que sea posible tener una visión global de todos los estudios seleccionados.

Con el fin de aprovechar todo el conocimiento posible, *MetaFun* lleva a cabo un metaanálisis funcional. Este tipo de metaanálisis añade un enfoque distinto, en lugar de centrarse en el gen como unidad de análisis se utiliza la función que lleva a cabo,

lo que aporta puntos de vista más robustos, dado que la combinación de genes que regulan la misma función son comparados a la vez.

MetaFun es una aplicación web pensada no solo para realizar metaanálisis con una aproximación funcional, sino para llevar este tipo de estudios a todos los investigadores interesados independientemente de su formación bioinformática. Además está desarrollada con la idea de que sea sencilla y guíe al usuario de tal forma que tan solo con subir sus datos de expresión normalizados y sus diseños experimentales obtengan unos resultados claros e interactivos. Se cuenta también con la posibilidad de descargar sus resultados tras el procesamiento de los datos.

Este proyecto está pensado como un proyecto a largo plazo, al que se irán añadiendo nuevas funcionalidades, las cuales se comentarán al final de esta memoria, por lo que la escalabilidad del código, el diseño del modelo de dominio así como las diversas herramientas desarrolladas para dar soporte a la aplicación web, han sido clave en el desarrollo del ecosistema en torno al cual *MetaFun* se ejecuta.

Dado que los investigadores sin formación bioinformática son el usuario objetivo, desde el primer momento quedó claro que sería necesario minimizar los detalles técnicos y/o estadísticos que pedir al usuario y que debía haber explicaciones lo más sencillas posible de las opciones que influyan en el resultado final del metaanálisis.

Hoy en día la alta disponibilidad de datos de toda clase hace que se necesite una infraestructura que apoye la reutilización de los mismos (Wilkinson et al., 2016) [1]. Con este fin aparecen los principios *FAIR*, este acrónimo hace referencia a un conjunto de buenas prácticas para la publicación de datos científicos. Estos principios estipulan que los datos generados han de ser fáciles de encontrar *(Findable)*. Han de ser accesibles, tanto los datos, como los metadatos *(Accesible)*. También según estos principios han de seguir también los estándares de la comunidad para el tipo de dato, es decir, han de ser interoperables (*Interoperable*). Por último los datos han de ser reutilizables, lo que quiere decir que han de tener una licencia clara y accesible, para que la comunidad pueda hacer uso de ellos en otros estudios (*Reusable*).

MetaFun aprovecha y fomenta estos principios apoyándose en la reutilización de los estudios para generar nuevo conocimiento relevante.

El desarrollo de herramientas web como *Babelomics* (Al-Shahrour, Minguez, Vaquerizas, Conde & Dopazo, 2005) [2], *Hipathia* (Hidalgo et al., 2016) [3] o *MetaFun* acercan los métodos estadísticos y bioinformáticos a todos los investigadores que toda-

vía no tienen contacto directo con la bioinformática, bien sea por falta de financiación en sus centros de investigación, por falta de tiempo o por desconocimiento de lo poderosas que pueden ser estas herramientas. Por tanto, uno de los objetivos principales del desarrollo de *MetaFun* es la divulgación de los análisis bioinformáticos, en concreto del metaanálisis funcional (García-García, F., 2016) [4] entre los investigadores tanto clínicos como experimentales.

La orientación del desarrollo hacia un público no familiarizado con este medio, hace que la facilidad de uso y las explicaciones sencillas de las opciones que ofrece la herramienta, hayan sido un imperativo desde el comienzo del desarrollo.

2.
OBJETIVOS

El objetivo principal de este trabajo consiste en el desarrollo de una herramienta web accesible y dirigida al metaanálisis de datos ómicos desde el punto de vista funcional, que permita la generación de conocimiento relevante a partir de una selección de estudios biomédicos, tanto existentes, como en curso.

Para llevar a cabo este objetivo, se han definido los siguientes objetivos específicos:

1. Diseño de una interfaz amigable, sencilla y clara.

2. Fluidez en la navegación minimizando los tiempos de carga tanto de los resultados como de los datos almacenados.

3. Representación interactiva de los resultados.

4. Resumen de la información resultante del metaanálisis de tal manera que se pueda hacer una exploración de los resultados previa a un análisis más profundo.

5. Descarga de todos los resultados obtenidos de forma sencilla para poder llevar a cabo un análisis en detalle de los mismos.

6. Gestión confidencial de datos tanto de usuario, como de los estudios y resultados.

7. Comunicación segura entre cliente y servidor.

8. Diseño de una arquitectura para la solución web robusta y escalable.

9. Desarrollo de un pipeline que permita la realización de metaanálisis enfocado a la caracterización funcional de estudios ómicos.

3.
CONCEPTOS TEÓRICOS

En este apartado se proporciona una explicación de los conceptos más importantes respecto a los análisis que se realizan.

3.1. ANÁLISIS EXPLORATORIO

Durante el proceso de investigación en cualquier campo, la recogida de datos es uno de los puntos más importantes, esto hace que un repaso y procesado inicial de los datos recolectados sea vital. John W. Tukey definió este tipo de tratamiento de los datos en 1962 como *"procedimientos para analizar datos, técnicas para interpretar los resultados de estos procedimientos, formas de planificar la recogida de datos para hacer más sencillo su análisis, más preciso o más consistente, así como toda la maquinaria y resultados estadísticos (matemáticos) que se aplican para analizar los datos"* (Tukey, J.W., 1962) [5].

Para la representación gráfica de este tipo de análisis se suelen utilizar diagramas de caja, diagramas de puntos, histogramas y diagramas de Pareto entre otros.

En *MetaFun* se han elegido para el estudio de la distribución de los datos los diagramas de cajas y para el estudio de agrupación, los Análisis de Componentes Principales (PCA) y *Clustering* de las muestras para cada estudio a analizar.

3.1.1. Diagramas de cajas o *BOXPLOTS*

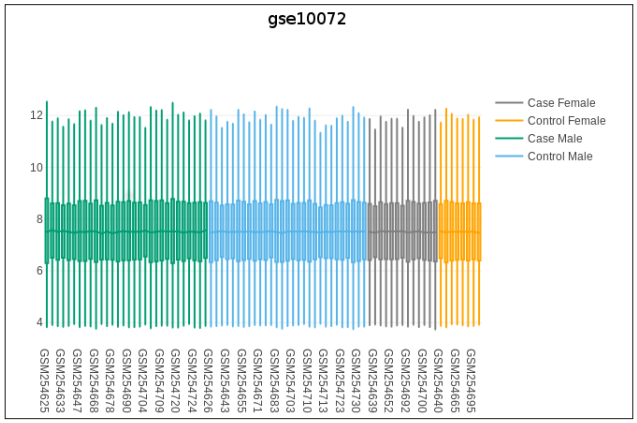

Figura 1. *Diagrama de cajas.*

Los *Boxplots* o diagramas de cajas permiten ver la distribución de los datos así como su simetría en función de sus cuartiles y sus valores atípicos. Cada una de las cajas estructura la información de tal manera que, en los extremos inferior y superior de cada una se representan el primer y tercer cuartil respectivamente, mientras que dentro de la caja, la línea que la cruza representa la mediana o segundo cuartil. Además también están representados los valores máximo y mínimo en las líneas del diagrama, en los denominados bigotes. En la Figura 1 se pueden observar los diagramas de cajas de las muestras del estudio GSE10072, coloreados en función de sexo y condición.

En *MetaFun* los boxplots se le muestran al usuario en el apartado *Exploratory Analysis* para que pueda comprobar la variabilidad de los distintos estudios a primera vista.

3.1.2. Análisis de componentes principales o *PCA*

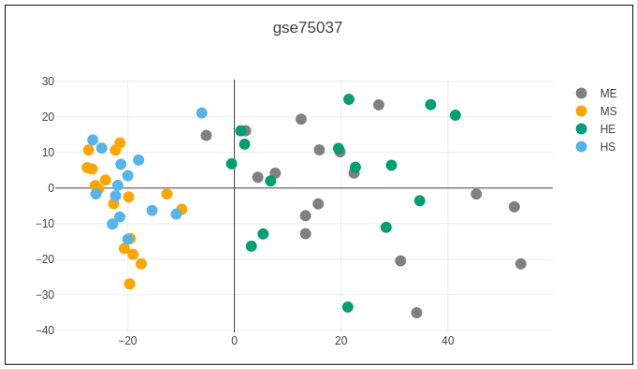

Figura 2. *Diagrama de puntos (PCA).*

El análisis de componentes principales permite resumir datos de alta dimensión realizando una combinación lineal de los valores originales. En este trabajo se realiza este análisis basándonos en las muestras, para ser capaces de dilucidar si la representación de las dos primeras componentes principales tienen la capacidad de dividir correctamente las muestras en función de las covariables. En la Figura 2 se puede ver el diagrama de puntos representando las dos primeras componentes principales para la separación de muestras por sexo y condición del estudio GSE75037.

En *MetaFun* este análisis permite al usuario comprobar la agrupación de las muestras en un diagrama de puntos donde cada eje (X e Y) representa una componente principal diferente. Además cada grupo experimental está representado de un color.

3.1.3. Análisis de grupos o análisis cluster

La idea principal tras este tipo de análisis es la agrupación de las variables, en nuestro caso las muestras de cada estudio, para detectar patrones comunes de expresión. En *MetaFun* se utilizan como complemento al análisis de componentes principales y así ayudar a los investigadores en la interpretación de la exploración y evaluación de los datos. Esto ayuda a decidir si los datos son lo suficientemente buenos como para que el resultado final sea relevante, y que en caso de no serlo, se pueda interpretar cuál es el problema por el que el metaanálisis funcional no es significativo. En el caso en el que los datos sean correctos servirán para aportar las evidencias de que los resultados son adecuados. En la Figura 3 se ve la separación por sexo y condición mediante técnicas de agrupación de las muestras del estudio GSE75037.

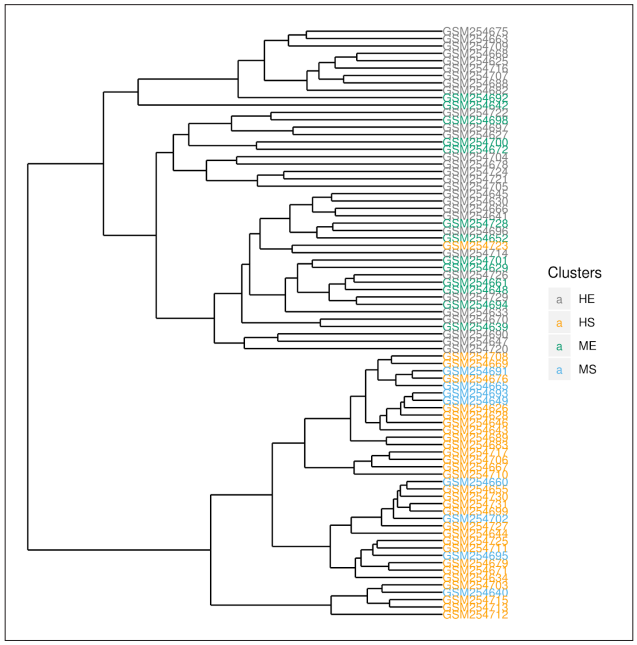

Figura 3. *Diagrama del análisis cluster.*

De la misma forma que los diagramas de cajas y el análisis de componentes principales, el análisis clúster se utiliza para mostrar al usuario otro tipo de representación de las agrupaciones de las muestras de cada estudio para que pueda comparar los resultados del PCA con los de este análisis. Esto es importante ya que si solo existiese una representación de la agrupación de muestras las decisiones como la eliminación de muestras concretas podrían estar sesgadas por el método de separación, de esta manera realizando dos tipos de análisis que permiten la agrupación de las muestras el usuario puede decidir si hay alguna muestra en algún estudio que debería ser eliminada.

3.2. DATOS ÓMICOS Y TECNOLOGÍAS DE ALTO RENDIMIENTO

Llamamos datos ómicos a la información que podemos recabar de un conjunto de elementos similares en un nivel biológico completo (Transcriptómica, Proteómica...).

En *MetaFun* la tecnología que nos proporciona la información es la transcriptómica. Esta técnica nos proporciona información sobre el conjunto de transcritos en una célula y nos permiten conocer qué genes se están expresando en mayor o menor medida.

Existen principalmente dos técnicas para obtener datos transcriptómicos. Una son los llamados microarrays, unos dispositivos basados en la síntesis de cadenas de *DNA* a cuyas bases se le han añadido componentes, como la proteína verde fluorescente *GFP,* que nos permiten realizar mediciones como, por ejemplo, la cantidad de fluorescencia que emiten, y que se fijan a una superficie sólida para que puedan ser analizadas a través de técnicas de imagen. La otra técnica se denomina *RNA-Seq,* y se utiliza para estudiar el transcriptoma de una muestra biológica en un momento concreto. Se basa en la secuenciación masiva, por lo que la forma de trabajar con los datos resultantes está ligada a las lecturas que se realizan. Estas se mapean contra el genoma de referencia del organismo estudiado, obteniendo así el número de veces que ha alineado correctamente el transcrito de un gen.

En cada caso los datos han de ser tratados y analizados de forma distinta, ya que, la primera técnica nos proporciona dato continuo, la cantidad de fluorescencia detectada, mientras que la segunda nos da como resultado datos discretos, el número de lecturas por transcrito de interés "contados" en la célula.

Con estos datos se pueden llevar a cabo distintos análisis, que se explican a continuación, pero es vital que estos datos sean públicos y puedan ser reutilizados por la comunidad, no solo para poder replicar los resultados sino también para llegar a nuevas conclusiones mediante el diseño de experimentos *in silico*. Para que esto sea posible es también necesario aportar información sobre los propios datos, los llamados metadatos. Un ejemplo claro de la necesidad de estos metadatos es la dificultad de realizar estudios comparativos por sexo, ya que, en muchos de los estudios disponibles en repositorios públicos no se incluye este tipo de información.

3.2.1. Expresión diferencial

La expresión diferencial es el análisis más frecuente cuando hablamos de transcriptómica, y es el primer paso en el análisis realizado por *MetaFun.* En este tipo de análisis lo que intentamos encontrar son diferencias significativas en la expresión de los genes de interés entre los distintos grupos experimentales. Con los resultados del análisis de expresión diferencial, por tanto, conoceremos qué genes son los que están infraexpresados o sobreexpresados y en qué proporción.

En *MetaFun* la expresión diferencial se realiza haciendo uso del paquete de R *Limma* (Ritchie et al., 2015) [6], mediante la función *DiffExp* de *MetaFunR. Limma*

es un paquete que nos permite analizar, haciendo uso de modelos lineales, los datos obtenidos a partir tanto de microarray como de *RNA-seq*, desde el preprocesado de los datos hasta el propio análisis comparativo múltiple entre los distintos *RNAs* objetivo.

3.2.2. Caracterización funcional

Una vez que se obtienen los genes resultantes de la expresión diferencial, puede ser interesante conocer la función o funciones que realizan en la célula. También puede ocurrir que aquellos que se encuentran diferencialmente expresados sean escasos o nulos, esto hace que haya que ampliar el espectro de búsqueda con distintas técnicas. La idea general detrás de la caracterización funcional es la asociación de cada gen, transcrito, proteína o producto con, por ejemplo, un proceso biológico, una función molecular específica o una ruta de señalización o metabólica. En *MetaFun*, la caracterización funcional se puede realizar bien buscando el proceso biológico detrás de los genes mediante *Gene Set Enrichment Analysis (GSEA)* o bien buscando las rutas de señalización en las que intervienen mediante *Hipathia*.

- Gene Set Enrichment Analysis (GSEA)

  GSEA nos permite realizar un análisis computacional de grupos de genes enriquecidos funcionalmente. El resultado de este tipo de enriquecimiento funcional indica cuál es la función biológica, proceso molecular o componente celular detrás de un conjunto de genes con un patrón común de expresión (Montaner & Dopazo, 2010) [7].

- Hipathia

  Hipathia es un método para la computación de las señales de transducción a lo largo de los *pathways* de señalización a partir de datos transcriptómicos. El método está basado en un algoritmo iterativo el cual es capaz de computar la intensidad de la señal que atraviesa los nodos de una red teniendo en cuenta el nivel de expresión de cada gen y la intensidad de la señal que le llega. También provee un nuevo enfoque al análisis funcional permitiendo el cómputo de las señales que llegan a las funciones anotadas en cada *pathway* (Hidalgo et al., 2016) [3].

3.3. METAANÁLISIS

El metaanálisis puede ser definido como *"una revisión sistemática de la literatura existente acerca de un tema haciendo uso de métodos estadísticos donde la meta es sumar y contrastar los descubrimientos de varios estudios relacionados"* (Glass GV, 1976) [8].

Para llevar a cabo un metaanálisis el primer paso es realizar una exhaustiva búsqueda de estudios referentes a nuestro tema de interés, esta parte es crucial, puesto que una errónea selección de los criterios de inclusión puede sesgar los resultados. Si, por ejemplo, se realiza un estudio de diferencias de sexo en una enfermedad, y los estudios seleccionados están desbalanceados, es decir, incluyen por ejemplo un mayor número de hombres que de mujeres entre sus muestras, los resultados se verán influenciados por la comparación entre hombres caso y hombres control en cuyo caso las diferencias de sexo no serían apreciables.

Una vez que se han elegido correctamente los estudios, el siguiente paso es el cálculo de la magnitud del efecto de la variable de interés y de su variabilidad ya que en función de esta última tendremos que elegir entre realizar el metaanálisis siguiendo un modelo de efectos fijos o un modelo de efectos aleatorios.

- Efectos Fijos

  Cuando utilizamos un modelo de efecto fijo para un metaanálisis, estamos asumiendo que todas las diferencias que se encuentran en nuestros datos son debidas al error de muestreo. En los metaanálisis con datos biológicos, las diferencias entre las técnicas y la variabilidad de los propios datos obtenidos por las mismas hace que las diferencias puedan no estar relacionadas solamente con el error de muestreo.

- Efectos Aleatorios

  Cuando asumimos que las diferencias en los datos de nuestros estudios a metaanalizar varían entre ellos, es decir, que cada uno de los estudios tiene un error de muestreo distinto, aplicaremos un modelo de efectos aleatorios. Este es el caso más común en los datos biológicos, y por tanto es el recomendado en *MetaFun*.

3.3.1. Consideraciones acerca del Metaanálisis

El metaanálisis es capaz de detectar diferencias significativas que, a partir de uno solo de los conjuntos de datos seleccionados no sería posible. Sin embargo, hay que tener en cuenta que los resultados del metaanálisis están fuertemente ligados a la calidad de los datos de entrada. Además, el sesgo de publicación puede dificultar la búsqueda de los estudios relevantes, puesto que lo resultados negativos tienden a no ser publicados, y por tanto esto puede ser un handicap muy importante.

Las representaciones gráficas de los resultados del metaanálisis implementadas en *MetaFun,* de cara al usuario son las siguientes:

3.3.2. Gráfico de bosque o *FOREST PLOT*

Los gráficos de bosque son uno de los métodos de presentación de resultados en los metaanálisis. Estos representan en el eje vertical los estudios que han sido parte del metaanálisis. En el eje horizontal tenemos el valor de la variable de interés junto con su intervalo de confianza, además el punto que representa este valor varía de tamaño en función del peso que haya tenido en el metaanálisis. En la Figura 4 se puede ver un gráfico de bosque para el término GO:0006335 en un metaanálisis con los estudios GSE10072, GSE75037 y GSE19188.

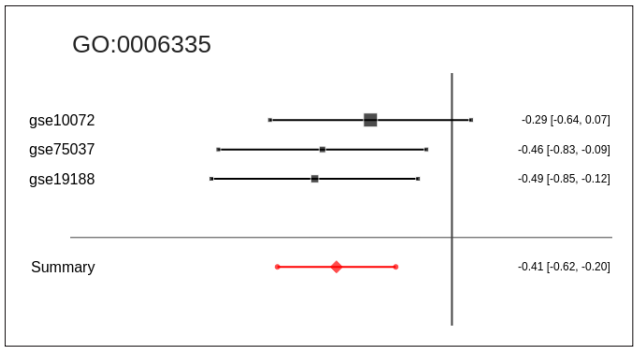

Figura 4. *Gráfico de Bosque.*

En el caso de *MetaFun* la variable de interés es el *Log Odds Ratio* del término representado obtenido a partir de de la caracterización funcional seleccionada por el usuario.

3.3.3. Gráfico de embudo o *FUNNEL PLOT*

Los gráficos de embudo también son muy útiles en la representación de resultados provenientes de metaanálisis. Estos enfrentan el valor cuantitativo de la variable de interés con su error estándar, por tanto, los estudios que una vez representados aparecen alejados horizontalmente del centro, son estudios poco precisos. En el eje vertical, cuanto más cerca del 0 se encuentren su error estándar será menor. En la Figura 5 se observa un ejemplo de gráfico de embudo para el término GO:0006335 en un metaanálisis de tres estudios.

En *MetaFun,* la variable de interés representada en el eje X, es el *Log Odds Ratio* del término funcional representado, obtenido a partir de la caracterización funcional seleccionada por el usuario.

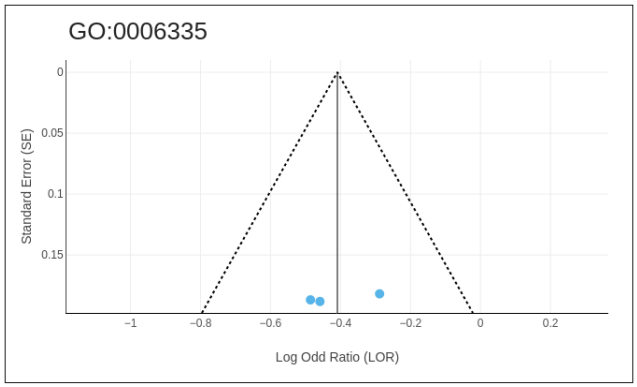

Figura 5. *Diagrama de Embudo.*

4.
METODOLOGÍA Y HERRAMIENTAS

En este apartado se explican las distintas técnicas metodológicas que se han utilizado durante el desarrollo de este proyecto tanto para la elaboración de la aplicación web como de las demás partes necesarias para su correcto funcionamiento.

4.1. ASPECTOS RELEVANTES DEL DESARROLLO

A continuación se comentarán los aspectos técnicos relevantes para el desarrollo de *MetaFun*.

4.1.1. Ciclo de vida

Se llama ciclo de vida a la evolución que tiene un *software* desde su diseño hasta que deja de mantenerse. En una aplicación web este ciclo de vida podría separarse en Planificación, Desarrollo, Pruebas, Lanzamiento, Monitorización y Mantenimiento, tras el cual, es necesario volver al primer punto, pero esta vez para planificar el arreglo de los fallos detectados durante las fases de Monitorización y/o Mantenimiento.

En este apartado se explica la fase de Planificación que puede subdividirse en dos fases más, la Fase de Análisis y la Fase de Diseño.

4.1.1.1. *Fase de análisis*

Una de las partes más importantes cuando se realiza un proyecto de desarrollo de software es la recolección de objetivos y requisitos. Por tanto una pobre recolección de estos puede conllevar un alargamiento de plazos en las distintas fases, también una mala captura de requisitos es un problema que puede repercutir de una manera sustancial a lo largo del ciclo de vida en sus posteriores iteraciones.

Entre los objetivos básicos que surgen de un primer análisis del dominio del problema se encuentran:

– *Gestión de usuarios:* El sistema tendrá que ser capaz de permitir el registro de los usuarios así como permitir la modificación de manera correcta y ágil de la información personal. Todo el tratamiento de los datos ha de ser confidencial.

– *Gestión de datos:* El sistema tendrá que ser capaz de permitir a los usuarios el correcto tratamiento tanto de sus datasets como de los datasets públicos que utilicen para llevar a cabo sus análisis.

– *Almacenamiento de los resultados:* El sistema tendrá que ser capaz de almacenar y permitir explorar los resultados de los análisis que el usuario realice.

Los requisitos recogidos se pueden separar en:

- Requisitos de información: Son aquellos referidos a la recogida de los distintos tipos de datos.
- Requisitos de restricción: Son aquellos referidos, por ejemplo, a las limitaciones del sistema en función del tipo de usuario o tipo de datos.
- Requisitos no funcionales: Son aquellos que no hacen referencia a funcionalidades del software sino a especificaciones abstractas o de interfaz.
- Requisitos funcionales: Son aquellos referentes a las funcionalidades necesarias de un software.

Requisitos de información:

– *Usuario:* La información recogida de cada usuario registrado será tan solo su dirección de correo electrónico, para enviar avisos de finalización de análisis, cambios de contraseña y demás funcionamientos habituales.

– *Trabajo:* La información de cada análisis recogida es la referente a las opciones del propio análisis, los datasets que se utilizan, así como sus archivos de diseño experimental asociados.

– *Datos:* La información recogida de los datos que se utilizará para cada análisis sólo deben ser accesibles por el usuario que los aportó.

Requisitos de restricción:

– *Resultados confidenciales:* El sistema deberá garantizar la confidencialidad tanto de los análisis como de los datasets que se suban a la plataforma.

Requisitos no funcionales:

– *Facilidad de uso:* la experiencia de uso de la aplicación final deberá ser sencilla, de tal manera que presente la información a la que se trata de acceder de manera ágil, así como de introducir y gestionar los datos de forma sencilla.

– *Fiabilidad:* El sistema ha de ser fiable principalmente con respecto a los datos que gestionará.

– *Adaptabilidad:* El sistema deberá adaptarse a las resoluciones más comunes para garantizar la usabilidad.

Requisitos funcionales:

– *Identificación de usuario:* El sistema deberá ser capaz de identificar el usuario para la correcta visualización de los resultados y datos que se poseen en el servidor.

– *Gestión de archivos:* El sistema deberá ser capaz de permitir la subida de archivos así como la eliminación de los mismos.

– *Lanzamiento de los análisis:* El sistema deberá ser capaz de permitir lanzar el metaanálisis funcional a partir de los datos que seleccione el usuario.

– *Visualización de resultados:* El sistema deberá ser capaz de garantizar una cómoda y correcta correcta visualización de los resultados.

– *Descarga de resultados:* El sistema deberá permitir la descarga de un resumen suficiente de los resultados generados del metaanálisis.

4.1.1.2. *Fase de diseño*

Durante la fase de diseño se intenta concretar una solución que permita realizar los requisitos captados durante la fase de análisis, tras investigar diferentes alternativas de diseño la resultante fue la que se explica a continuación.

4.1.1.2.1. Diagrama de Clases/Componentes de la aplicación Web

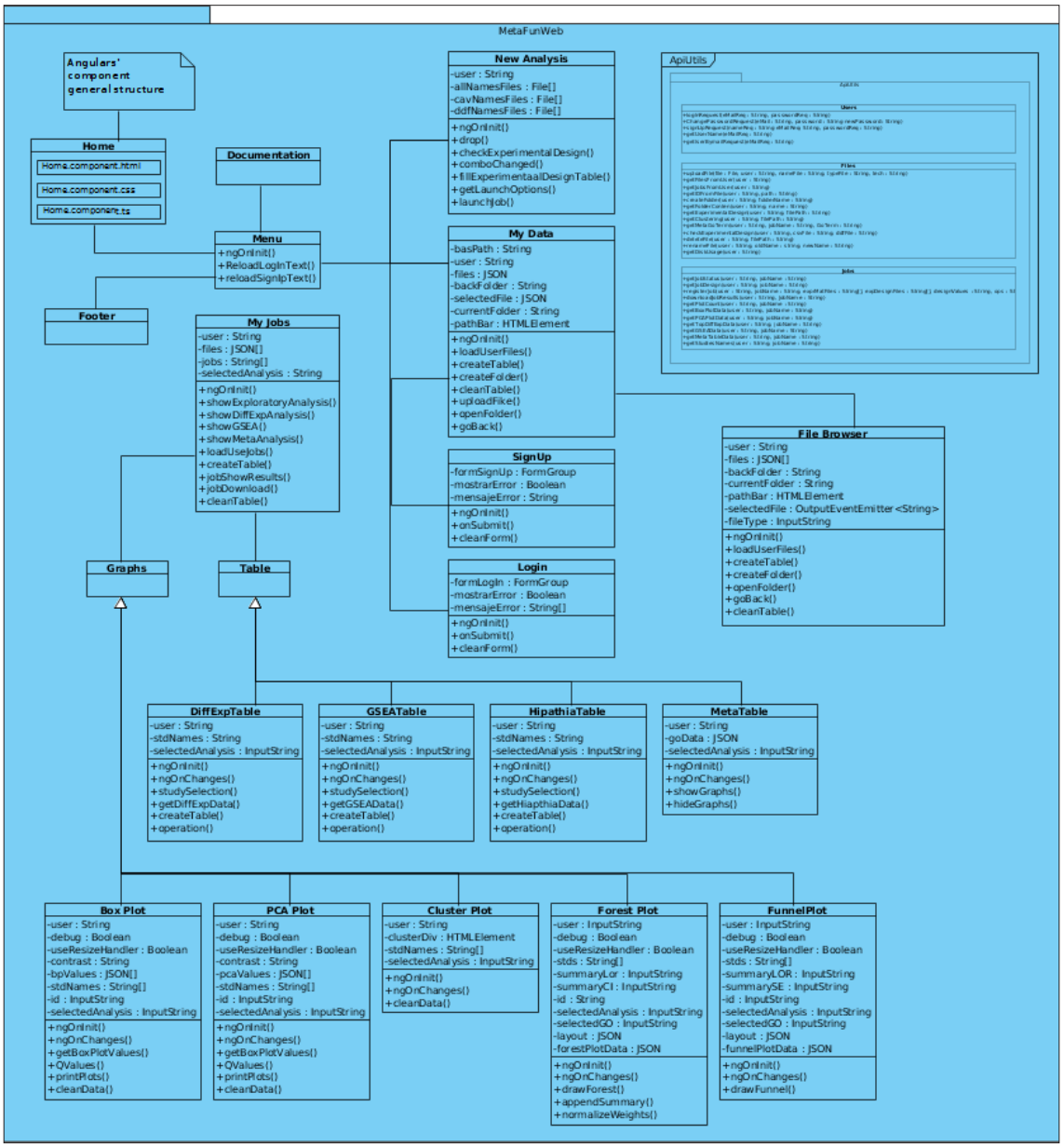

Figura 6. *Diagrama de Clases/Componentes.*

La Figura 6 muestra el diseño del diagrama de clases de la solución web. Dado que el marco de trabajo elegido para el desarrollo es *Angular*, cada una de nuestras clases/componentes del modelo está subdividida en cuatro archivos fuente:

– *.component.html: Contiene la vista del componente.

– *.component.css: Contiene la hoja de estilos del componente.

– *.component.spec.ts: Contiene los descriptores del componente

– *.component.ts: Contiene la lógica de negocio del componente.

A continuación se explicarán tanto la función como los detalles más relevantes del desarrollo de cada componente de forma individual.

Home:

Es la página de inicio de la aplicación web, en ella se puede leer la descripción de la misma así como acceder a los distintos apartados a través del siguiente componente.

Menu:

Es el componente encargado de representar y gestionar la barra de acciones de la aplicación web, redirigiendo al usuario a la funcionalidad que desee utilizar.

Footer:

Es el componente encargado de representar la barra inferior con los logos y las disposiciones legales y oportunas.

Documentation:

Es el componente encargado de entregar la información del manual de uso al usuario. En ella se especifican los formatos necesarios para el correcto funcionamiento de la herramienta.

SignUp:

Es el componente encargado del registro de un usuario nuevo.

LogIn:

Es el componente encargado de validar las credenciales de acceso de un usuario existente.

File Browser:

Este componente se encarga de la recreación de un navegador de archivos convencional como los que se pueden ver en cualquier sistema operativo, le permite al usuario navegar por sus datos como si estuviese trabajando en el de su ordenador. Está instanciado en el componente *My Data*, este hecho, nos permite desacoplar el desarrollo de la funcionalidad haciendo que cualquier reimplementación, rediseño o corrección pueda ser llevada a cabo sin intervenir en la estructura de la web.

My Data:

Es el componente encargado de permitir al usuario la gestión de los datos que ha subido a la página web. Este componente contiene, además de las funciones necesarias para las operaciones de subida, las de renombre y eliminación de los datos, la cual es una instancia del componente *File Browser*.

New Analysis:

Es el componente encargado de obtener los datos oportunos para el correcto lanzamiento del análisis. Es el componente que contiene los primeros detalles relevantes en su desarrollo.

La Vista está dividida en dos bloques claros, por un lado los archivos que tiene el usuario almacenados, y por otro las configuraciones necesarias para el funcionamiento del metaanálisis.

Al usuario se le piden a lo largo de tres apartados distintos valores para las opciones de lanzamiento del análisis.

En el primero de los apartado se obtiene información sobre:

– Si quiere que el metaanálisis siga un modelo de efectos fijos o de efectos aleatorios.

– Si desea la caracterización funcional de sus resultados basada en *GSEA* o en *Hipathia.*

– El organismo de referencia del que se han obtenido los datos, puesto que la caracterización funcional dependerá de ello.

El segundo apartado de datos le pide al usuario que elija tanto los archivos que quiere que sean analizados, como sus correspondientes archivos con su diseño experimental. Estos archivos deben contener un listado con los nombres de las muestras en una columna y el grupo experimental al que pertenecen en la otra.

En la interfaz gráfica los archivos que serán analizados tienen que organizarse de tal manera que queden en el mismo orden tanto en la columna *Expression Files* como en la columna *Experimental Design*. Esto es muy importante ya que en el orden en el que se coloquen es en el orden en el que se analizarán durante el *pipeline* y por tanto, en caso de que los datos de los nombres de las muestras de los estudios no coincidan con el nombre de las muestras en el diseño experimental no se le permitirá al usuario elegir el contraste que desear realizar en el análisis y por tanto no podrá lanzar el metaanálisis.

Si el usuario ha colocado correctamente los archivos con la matriz de expresión y su diseño experimental, se pide que elija el contraste así como que nos informe de que etiquetas corresponden a cada una de las partes del propio contraste. Por ejemplo si el usuario desea llevar a cabo un metaanálisis con diferencias de género y en el archivo con el diseño correspondiente a una de las matrices de expresión tiene identificadas a cada una de las muestras con cuatro etiquetas posibles referentes a la enfermedad, *Adenocarcinoma_Mujer*, *Adenocarcinoma_Hombre*, *Control_Mujer* y *Control_Hombre*, en su selección deberá indicar para cada estudio cuál es *Case_Female, Cas_Male, Control_Female, Control_Male*, ya que puede tener distintos nombres en cada diseño.

Una vez que todos los datos están correctos, el botón que permite lanzar el análisis se habilitará y lanzará una petición *REST* para la ejecución del mismo. Más adelante se explicará la *API RESTful* con detalle.

My Jobs:

Es el componente en el que el usuario puede gestionar los análisis que tiene activos, los que ya finalizaron y ver los resultados de cada uno de ellos. Este componente, además, tiene embebidos varios cada uno perteneciente a un tipo de análisis de los que se realizan, todos ellos explicados en detalle a continuación y que podrían dividirse en dos tipos de componente distintos, componentes de tipo Gráfico (*BoxPlot, PCAPlot, ClusterPlot, ForestPlot y FunnelPlot*) y componente de tipo Tabla (*DiffExpTable, GSEATable, HipathiaTable y MetaTable*), con la excepción del encargado de representar el metaanálisis que posee ambos tipos de componente.

En la visualización del análisis exploratorio intervienen:

BoxPlot:

Este componente se encarga de la creación de los gráficos de cajas. Lleva a cabo una petición *REST* por estudio para obtener los datos mínimos necesarios para la representación de los mismos, recuperando, por tanto la información del primer y tercer cuartil, la mediana así como los datos correspondientes a los bigotes del gráfico.

PCAPlot:

Este componente se encarga de la creación de los gráficos de puntos que representan las dos primeras componentes principales del análisis de cada uno de los estudios por separado. Obtiene los datos a representar a través de una petición *REST* por estudio.

ClusterPlot:

Es el componente encargado de la representación del análisis cluster y pide al servidor a través de otra petición *REST* la imagen generada durante el análisis y la coloca en la web.

En la visualización de los resultados del análisis de expresión diferencial interviene:

DiffExpTable:

Este componente se encarga de la generación de una tabla que incluye los cincuenta genes con mayor expresión diferencial en cada uno de los estudios y enlaza cada uno de ellos a través de su *ENTREZID*, con su descripción en *NCBI*.

En la visualización de los resultados del enriquecimiento funcional intervienen:

GSEATable:

Es el componente encargado de la realización de la tabla que muestra los resultados del análisis de enriquecimiento funcional basado en *GSEA* para cada estudio y enlaza cada término GO con su descripción en la web de la *Gene Ontology*.

HipathiaTable:

Es el componente encargado de la realización de la tabla que muestra los resultados del análisis de enriquecimiento funcional basado en Hipathia para cada estudio y enlaza cada ruta con su descripción en *KEGG*.

En la representación de los resultados del metaanálisis intervienen:

ForestPlot:

Este componente se encarga de la representación del diagrama de bosque en el que se representan los valores del *Log Odds Ratio* y la variabilidad de cada uno de los términos GO que resultan significativos en el metaanálisis por estudios y su resumen, así como el peso que tienen en el resultado del análisis cada uno de ellos. Los datos necesarios para su representación se obtienen mediante una petición *REST* a través de *MetaApi*.

FunnelPlot:

Este componente se encarga de la representación del diagrama de embudo perteneciente a cada término GO significativo resultante del metaanálisis. En este tipo de gráficos se enfrenta el *Log Odds Ratio* de cada estudio frente a su *Standard Error* así como una medida de variabilidad aceptable. Los datos necesarios para su representación se obtienen mediante una petición *REST* a través de *MetaApi.*

MetaTable:

Este componente representa la tabla que contiene los resultados del metaanálisis, enlaza cada término GO con su descripción en la *Gene Ontology* además es el componente encargado de mostrar los gráficos de bosque y embudo a petición del usuario cuando hace click en el botón de información de cada fila.

Además de los componentes explicados se ha desarrollado una clase encargada de la realización de las peticiones *REST*, *ApiUtils*, que está representada en el diagrama en la parte superior derecha (Figura 6). La relación de uso entre los componentes que la utilizan y la misma no ha sido representada para una mayor legibilidad del mismo, ya que, la mayoría de ellos la utilizan. *ApiUtils* funciona de manera similar a un *Data Access Object (DAO)*, los componentes que la utilizan son: *My Jobs,* los componentes agregados *Graphs* y *Tables*, *New Analysis, MyData, SignUp y LogIn*.

4.1.1.2.2. Diseño de la base de datos

La elección de un diseño no relacional, implementado con *MongoDB* [9], permite una mayor eficiencia en las peticiones. Dado que la aplicación final tendrá un gran número de interacciones a través de *MetaApi,* la *API* desarrollada para este proyecto, la elección de utilizar *MySQL* habría hecho que la solución final tuviese un peor rendimiento y una escalabilidad inferior. Por otro lado como se puede ver en la representación gráfica de la base de datos, la tablas, aunque tienen alguna relación entre ellas, son muy independientes.

El diseño de la base de datos por tanto se basa en tres colecciones, *Files, Jobs* y *Users* como se muestra en el modelo Entidad-Relación (Figura 7).

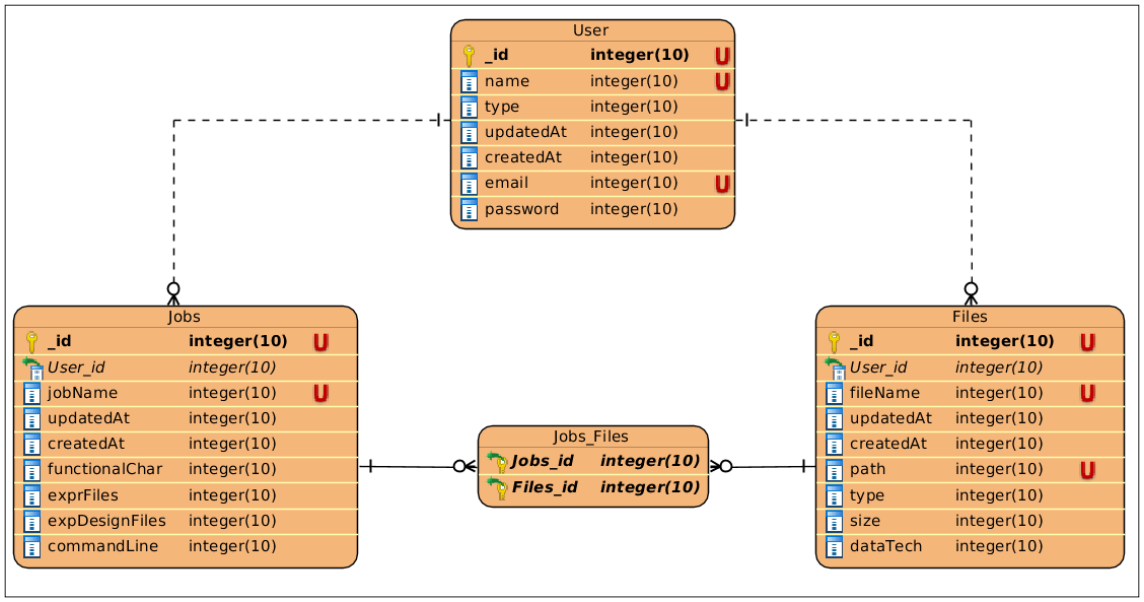

Figura 7. *Diagrana del Modelo Entidad-Relación.*

En *Files* se almacenan las particularidades de los archivos que el usuario ha subido a la plataforma. Es necesario saber si un archivo es una matriz de expresión, un diseño experimental o una anotación funcional. Es importante saberlo dado que se tratan y presentan en la aplicación de manera distinta.

En *Jobs* se almacena la información referente a los trabajos que lanza cada usuario, con las opciones elegidas, la hora de comienzo y distinta información relevante.

En *Users* se almacena la información del usuario, nombre de usuario, contraseña encriptada, correo electrónico, fecha de registro, último acceso correcto, etc.

4.1.1.2.3. Diseño arquitectónico

Al tratarse de una aplicación web con relaciones complejas y que hace uso constante de los datos almacenados acerca de los resultados y datasets el patrón arquitectónico que mejor se adapta a este propósito es el Patrón Modelo-Vista-Controlador (MVC). Este patrón separa en tres capas diferenciadas claramente las distintas necesidades de la aplicación.

– Modelo: Es la abstracción que contiene el modelo de dominio de la aplicación.

– Vista: Es la capa encargada de la representación de los datos que observa el usuario, no tiene capacidad de modificación del modelo, pero sí lo observa.

– Controlador: Contiene la lógica de negocio de la aplicación y tiene capacidad de modificación del modelo.

La adaptación del patrón MVC para *MetaFun* puede verse, simplificada, en la Figura 8 Diagrama Arquitectónico de la Aplicación.

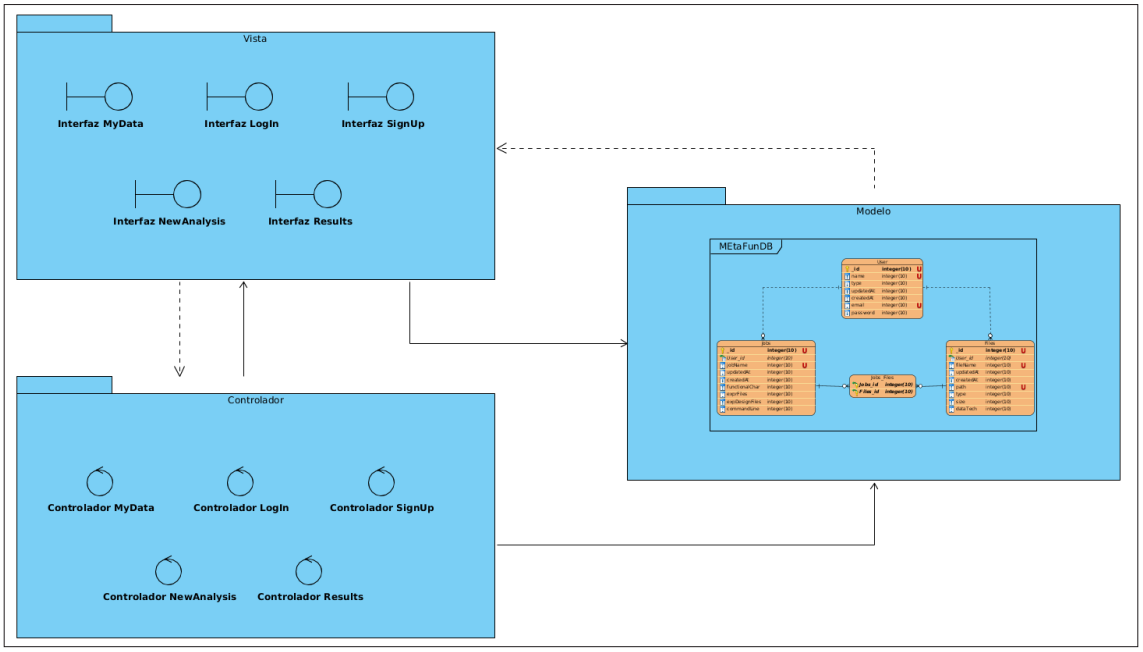

Figura 8. *Diagrama Arquitectónico de la Aplicación.*

4.2. HERRAMIENTAS Y LIBRERÍAS DESARROLLADAS

4.2.1. MetaApi

MetaApi es la *API RESTful* desarrollada para la comunicación de la aplicación web con el servidor que almacena los datos y ejecuta los análisis. Está desarrollada en Java y haciendo uso de la librería Jersey, esta solución permite que sea ampliable muy fácilmente.

MetaApi está dividida en tres clases que contienen los métodos referentes a los usuarios por un lado, los archivos por otro y por último los trabajos que lanzan los usuarios en la web. Esta división se puede observar en la Figura 9.

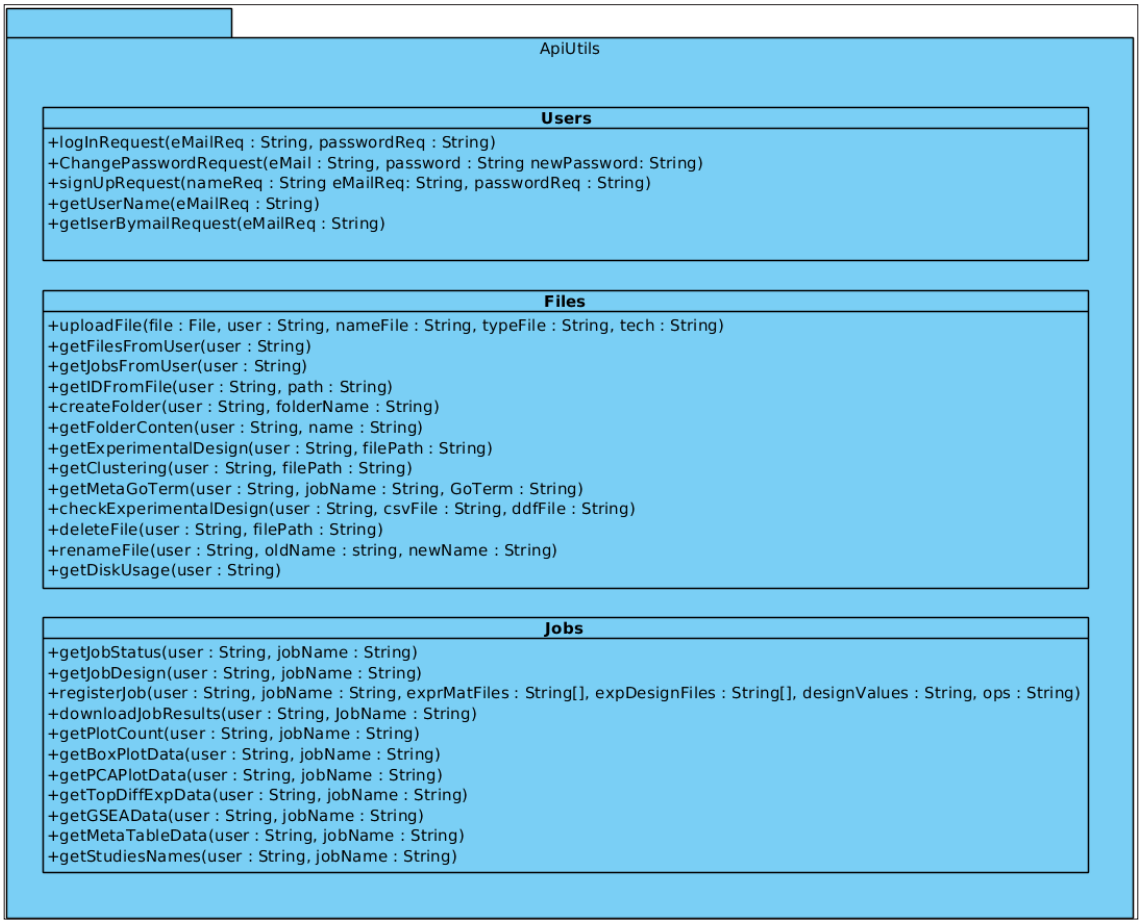

Figura 9. *Diagrama de Clases de MetaApi.*

Casi todos los componentes descritos en el apartado 4.1.3 tienen comunicación con el servidor a través de este desarrollo gracias a las anotaciones que nos permite añadir la librería *Jersey* para la correcta comunicación mediante REST.

Las anotaciones ensenciales, son:

- *@Path*: Indica la ruta relativa sobre la que se añadira la URI para acceder a una clase o método en la petición.
- *@GET, @POST, @PUT, @DELETE, @HEAD*: Esta anotación hace referencia a la petición HTTP que satisface el método.
- *@Produces*: Indica el tipo *MIME* que devuelve el método.
- @Consumes: Indica el tipo *MIME* que necesita un método como parámetro.

Users:

Es la clase encargada de la resolución de las peticiones pertenecientes a los datos de los usuarios. De su funcionamiento cabe destacar que toda la información controvertida está encriptada haciendo uso del algoritmo irreversible *MD5* [10], el cual genera un código hexadecimal siempre de trenta y dos caracteres independientemente de la longitud de la palabra introducida. Los métodos desarrollados en esta clase se centran en la comunicación con la tabla *USERS* de la base de datos y de llevar a cabo las comprobaciones oportunas.

Files:

Se encarga de resolver las peticiones referentes a los archivos que han subido los usuarios, también se encarga de gestionar los archivos que generan los análisis en en el servidor. En esta clase cabe señalar el mecanismo de subida de archivos.

Cuando un usuario quieres subir un archivo al servidor, esta acción desencadena varias acciones. Por un lado la web obtiene el archivo gracias a los mecanismos propios de los *webkit*, una vez que esto concluye, el archivo se transforma a BASE64, y se envía a través de una petición de tipo *POST* desde *ApiUtils* hasta el servidor y cuando llega se vuelve a transformar desde BASE64 para almacenarlo en la ruta apropiada del sistema de archivos.

Para la descarga de archivos se hace uso del tipo *MIME APPLICATION_OCTET_STREAM* y por tanto se construye la respuesta a partir de la lectura de los archivos usando un array de tipo *byte*.

Cuando las peticiones que recibe esta clase son de tipo *GET* normalmente se necesita información de un archivo, esta información es tratada en la propia *API* para convertirla a formato *JSON* y devolvérsela a la aplicación web para su tratamiento de cara al usuario.

Jobs:

Se encarga de las peticiones relacionadas con la gestión de la ejecución de los análisis, creación, eliminación y lanzamiento de los mismos. Cuando un usuario lanza un nuevo análisis, la aplicación web hace una petición que contiene los parámetros necesarios para la ejecución del pipeline, y el trabajo comienza su ejecución en el servidor.

4.2.2. MetaFunR

MetaFunR es un derivado del paquete desarrollado por la Unidad de Bioinformática y Bioestadística del Centro de Investigación Príncipe Felipe para la realización de metaanálisis funcional adaptado a las necesidades de la aplicación web. Está desarrollado en R y se puede consultar en https://gitlab.com/metafundev/metafunr.

4.2.3. MetaPipeline

Es el pipeline que conduce el metaanálisis. Cuando el usuario lanza el metaanálisis, el pipeline *parsea* los argumentos que han sido seleccionados, carga los datos de los estudios que previamente el usuario había subido y sigue los siguientes pasos:

1. Realización del análisis exploratorio generando al final de proceso los datos pertenecientes a los gráficos BoxPlot, PCA y Clúster para cada estudio.

2. Realización del análisis de expresión diferencial de los estudios de manera individual.

3. Realización de la caracterización funcional elegida por el usuario, GSEA o Hipathia, según el organismo de referencia seleccionado, ya sea Humano, Ratón o Rata.

4. Realización del metaanálisis de los resultados obtenidos a partir de la caracterización funcional de los estudios y generación de los datos referentes a los diagramas de bosque y de embudo.

El código de *MetaPipeline* se puede encontrar en https://gitlab.com/metafundev/metafunpipeline/blob/master/metapipe.r.

Durante el transcurso del pipeline se generan dos tipos de archivos. Por un lado están los que almacenan la información de los resultados de cada uno de los pasos anteriores, por otro lado también se generan archivos que reducen los accesos a la base de datos. Por ejemplo los cálculos necesarios para la realización de los distintos diagramas se realizan durante el pipeline, ya que como está desarrollado en R nos permite llevar a cabo de manera muy sencilla operaciones sobre todos los elementos de las matrices de resultados a la vez. Esto se puede observar claramente durante el cálculo de los

intervalos de confianza de cada estudio utilizados en los diagramas de bosque, este intervalo de confianza se calcula en función del valor del *Log Odds Ratio,* del límite de significación utilizado y del error estándar de cada estudio para cada término GO que ha resultado significativo.

De manera similar ocurre con los diagramas de embudo, en los cuales el embudo se dibuja a partir una recta cuya ecuación es $y = \pm 1.96 * SE + LOR$ [11] y que puede ser resumida en solo tres puntos, siendo en el eje horizontal los valores inferior y superior del intervalo calculado, y el centro el valor del *LOR* resumen del término, y en el eje Y el menor valor en los extremos y 0 en el centro.

Sin embargo, cuando estamos tratando con diagramas como los de caja, que tienen muchos valores por cada muestra del estudio a representar, para evitar la ralentización de la web, toda esta información se resume en los elementos básicos del diagrama, es decir, primer y tercer cuartil, mediana y los valores de los bigotes.

4.3. HERRAMIENTAS Y LIBRERÍAS EXTERNAS

El desarrollo de este tipo de herramientas web hace necesario el uso de tecnologías de terceros, las más importantes se comentan en este apartado.

4.3.1. Angular

Angular es una plataforma y marco de trabajo o *Framework* para el desarrollo de aplicaciones web basadas en *HTML* y *TypeScript.*

Una aplicación desarrollada con este *framework* está basada en lo que denomina componentes, estos componentes definen las vistas, que son conjuntos de elementos entre los que *Angular* puede elegir y modificar de acuerdo a la lógica de negocio y los datos. Además los componentes utilizan servicios que proveen de funcionalidades específicas que no están relacionadas directamente con las vistas, por tanto, se puede decir, que un componente, no es más que una clase simple que implementa el patrón *Decorator* y que informa a *Angular* de cómo han de ser tratados en ejecución. Si son vistas, nos permitirán modificar el código *HTML* después de que haya sido renderizado para mostrarse. [12]

4.3.2. TypeScript

TypeScript [13] es un lenguaje de programación de código libre desarrollado por microsoft. Fundamentalmente es un superconjunto de JavaScript, al que le añade tipados estáticos, y orientación a objetos [10].

TypeScript ha sido utilizado como lenguaje para el desarrollo de las funcionalidades en *MetaFun.*

4.3.3. Java

Lenguaje multiplataforma por excelencia, *Java* [14] es un lenguaje compilado que se ejecuta sobre una máquina virtual, lo que hace que los desarrollos en este lenguaje sean ejecutables en cualquier sistema operativo que ejecute una máquina virtual *Java.*

Java se ha utilizado en el desarrollo de *MetaApi.*

4.3.4. Jersey

Jersey [15] es una librería que nos permite desarrollar servicios web *RESTful* mediante su implementación de *JAX-RS.* Nos proporciona las herramientas básicas para construir nuestros servicios *REST* (*POST*, *GET*, *PUT* y *DELETE*) de manera cómoda y sencilla.

Jersey ha sido utilizado para el desarrollo de *MetaApi.*

4.3.5. R

R [16] es el lenguaje y entorno para computación estadística por antonomasia, nos permite tratar nuestros datos de manera matricial, facilitandonos el manejo tanto de nuestros voluminosos datos como de sus metadatos [13]. R ha sido utilizado para el desarrollo de *MetaFunR* y de *MetaPipeline*.

4.3.6. Plot.ly

Plot.ly [17] es una librería multiplataforma basada en D3, que permite la creación de gráficos interactivos y exportables. La versión utilizada de

la librería ha sido la realizada para *JavaScript* que es compatible con el desarrollo de *MetaFun* dado que, como se ha mencionado anteriormente, *TypeScript* es un superconjunto de *JavaScript.*

Plot.ly ha sido utilizado para generación de los gráficos de tipo caja, diagrama de puntos para la representación de los resultados del PCA, gráficos de bosque y gráficos de embudo.

5.
RESULTADOS

En este apartado se lleva a cabo un análisis completo que cubre el registro inicial en la web, la subida de datos, la ejecución de análisis y la obtención de resultados, mostrando las diferentes opciones de la web y de su funcionamiento. Durante el apartado 5.4 y sucesivos, las explicaciones se llevan a cabo siguiendo un ejemplo de metaanálisis de tres estudios con resultados conocidos. Es con este conjunto de datos, entre otros, con el que se han llevado a cabo las pruebas pertinentes para asegurar que los análisis se realizan correctamente.

Todo el código desarrollado durante este trabajo se puede encontrar en https://gitlab.com/metafundev.

En estas pruebas se lleva a cabo un estudio de diferencias de sexo en adenocarcinoma de pulmón. Los datos de los estudios utilizados se pueden encontrar en *GEO* con identificador *GSE19188, GSE75037 y GSE10072*.

5.1. PANTALLA DE INICIO

En la pantalla de inicio (Figura 10) se puede leer una breve descripción de las funcionalidades de la web así como proceder o bien al registro, a la autentificación del usuario o a la documentación.

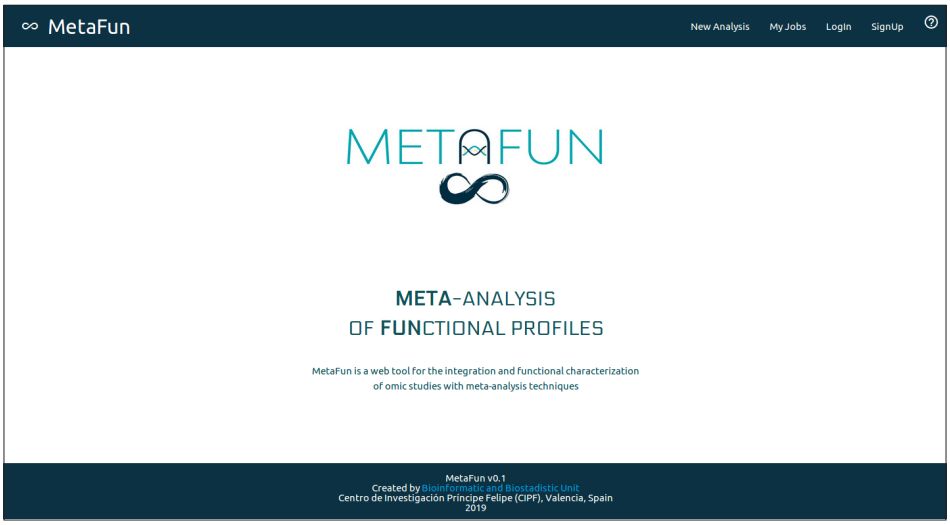

Figura 10. *Página de Inicio.*

Para llevar a cabo un análisis el usuario tiene que estar registrado en la web, para ello deben dirigirse a la opción *Sign Up* si es que no tiene cuenta o a *Log In* para autenticarse.

5.2. REGISTRO Y AUTENTIFICACIÓN

Apartado para la creación de un nuevo usuario en la herramienta web (Figura 11). Los datos necesarios para el correcto registro en el servicio son: nombre de usuario, correo electrónico y contraseña deseada.

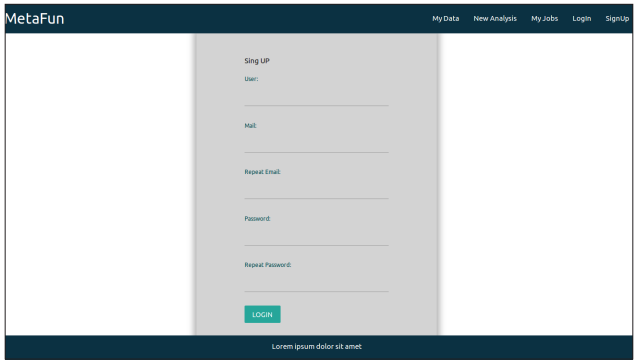

Figura 11. *Página de Registro.*

Para la validación de las credenciales de acceso es necesario credenciales (Figura 12) el correo electrónico y la contraseña correcta.

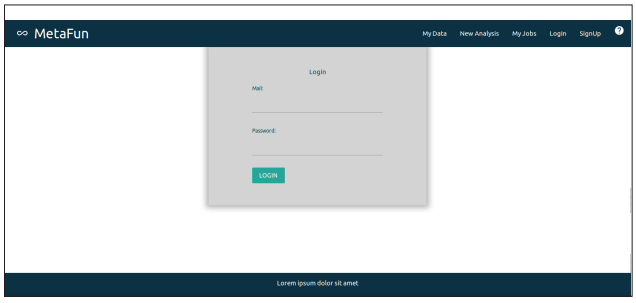

Figura 12. *Página de Acceso.*

5.3. PÁGINA DE USUARIO

Para que el usuario sea capaz de realizar los metaanálisis en *MetaFun*, es necesario que tengan los datos de los estudios a analizar almacenados en su usuario (Figura 13). Por tanto es necesario que haya un apartado en la que puedan realizar las operaciones de subida, eliminado, renombrado y modificación de los datos. Se ofrece también un navegador de archivos para que el manejo de los mismos sea lo más intuitivo posible.

Cuando un usuario nuevo entra en este apartado tendrá que añadir los *datasets* que quiere analizar, y también sus diseños experimentales, para ello deberá hacer click en *Upload Data*, esta acción hará aparecer una ventana emergente donde se encuentran las opciones de subida.

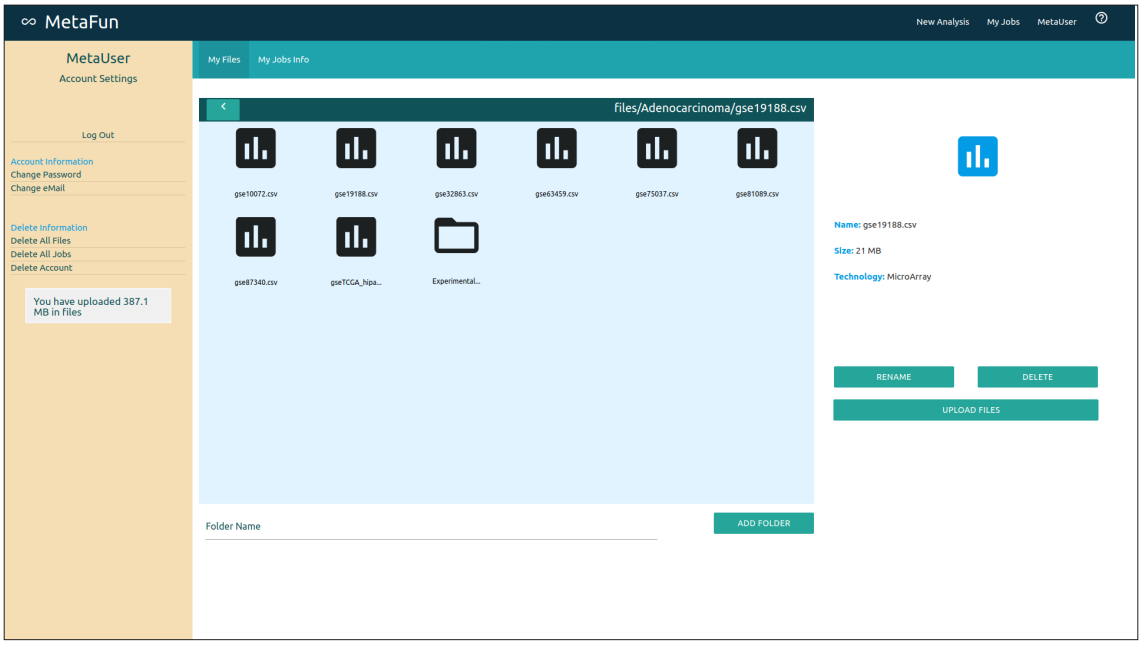

Figura 13. *Página de Usuario.*

Una vez tenemos los archivos en el servidor el usuario será capaz de seleccionarlos en el siguiente apartado.

5.4. CREACIÓN DE LOS ANÁLISIS

Cuando los usuarios ya tienen sus datos subidos a nuestro servidor, tendrán que elegir las diferentes configuraciones (Figura 14) del metaanálisis que quieren realizar. Tendrán que elegir el método de metaanálisis, si quieren utilizar un modelo de efecto fijo o aleatorio, si quieren realizar la caracterización funcional mediante GSEA o mediante Hipathia, y por último, si los datos con los que se llevará a cabo el metaanálisis son de Humano, Ratón o Rata.

En nuestro caso ejemplo seleccionaremos *Random Effect Model, GSEA y Homo Sapiens*.

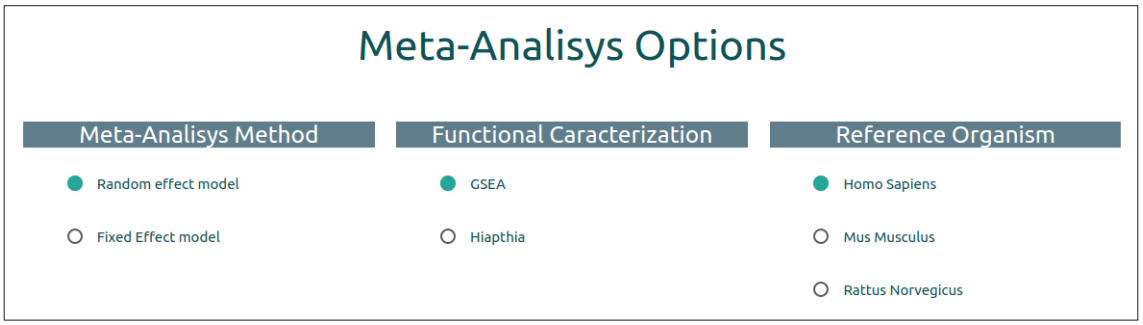

Figura 14. *Selección de Opciones del Metaanálisis.*

Después de seleccionar la opciones del metaanálisis tendrán que decidir cuáles son los estudios que estarán implicados. La experiencia de usuario elegida para esta tarea se basa en el arrastre de los estudios (Figura 15) a sus apartados correspondientes, en caso de ser los ficheros de expresión irán en el primer cajón y en caso de ser un fichero con el diseño experimental del estudio en el segundo. Además, tendrán que ordenarlos de tal manera que cada uno de los estudios quede alineado con su correspondiente diseño experimental (Figura 16).

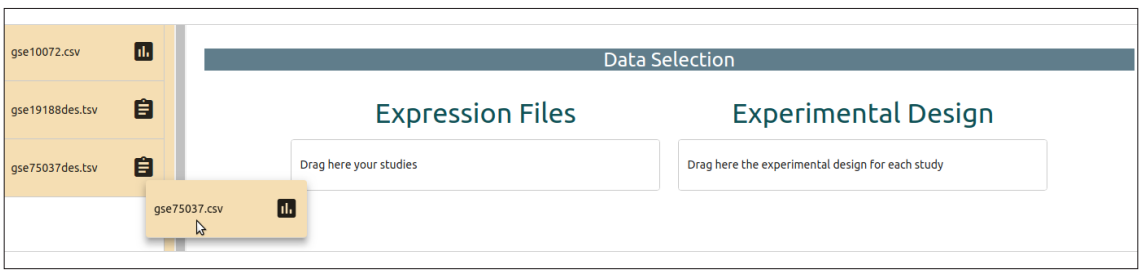

Figura 15. *Selección de Estudios para el Metaanálisis.*

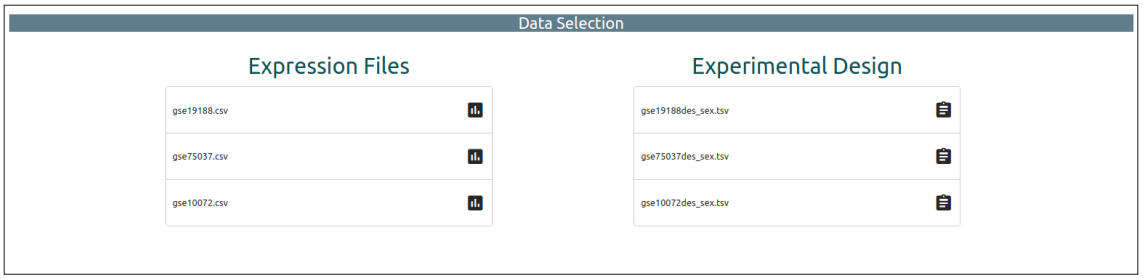

Figura 16. *Estudios Seleccionados Ordenados.*

Una vez los estudios están correctamente relacionados, se permitirá la opción de elegir el contraste que quieren llevar a cabo con los datos y seleccionar qué etiqueta se corresponde con la del contraste (Figura 17). Actualmente en *MetaFun* es posible realizar metaanálisis centrados tanto en un contraste Caso-Control así como en un contraste de diferencias de sexo [(Mujeres Enfermas - Mujeres Control) - (Hombres Enfermos - Hombres Control)].

Cuando todos estos detalles han sido cumplimentados, solo tienen que elegir el nombre que tendrá el estudio y lanzar el trabajo.

En este apartado tendremos que seleccionar *Adenocarcinoma_Female* como mujer enferma, *Control_Female* como mujer sana, *Adenocarcinoma_Male* para hombre enfermo, y *Control_Male* para hombre sano.

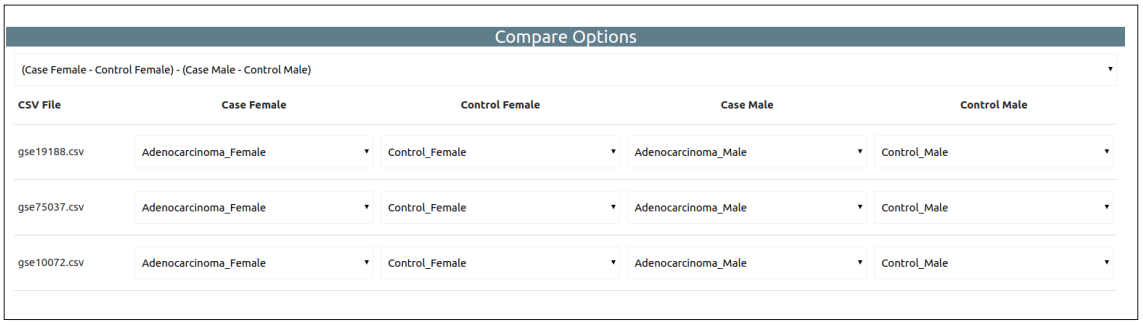

Figura 17. *Opciones de Contraste de Variables.*

5.5. RESULTADOS

Cuando la ejecución del metaanálisis ha concluido, el usuario podrá acceder a los mismos de manera interactiva a través de la pestaña "*My Jobs*". En este apartado tendrá que seleccionar entre sus trabajos finalizados cual de ellos es el que quiere ver.

Cuando ha seleccionado uno de sus análisis podrá ver los resultados del análisis exploratorio, expresión diferencial, enriquecimiento funcional y los resultados finales del metaanálisis así como un resumen de las opciones de lanzamiento del análisis, y un resumen cuantitativo del los resultados.

5.5.1. Resumen del Análisis

En este apartado se resumen descriptiva y cuantitativamente las características del análisis (Figura 18). Se pueden ver las opciones seleccionadas para lanzar el metaanálisis, el número de muestras de cada condición por estudio, y en total, las estadísticas referentes al número de genes y/o funciones significativas en los análisis de expresión diferencial, enriquecimiento funcional y el propio metaanálisis.

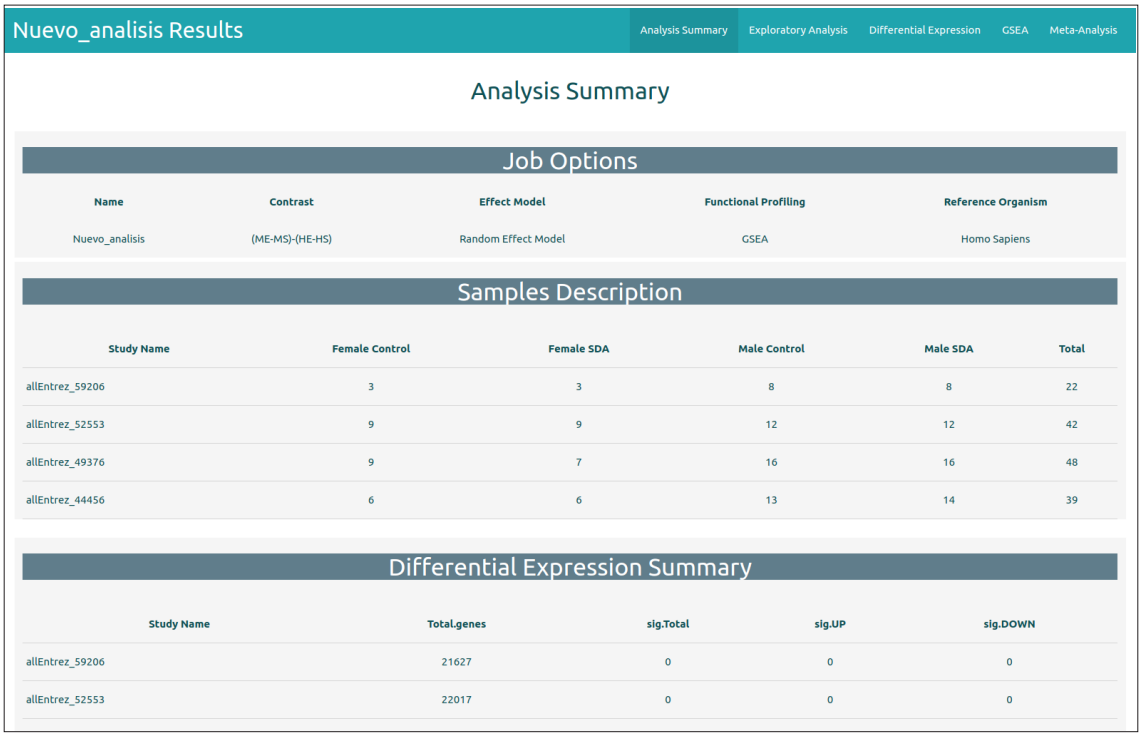

Figura 18. *Resumen Cuantitativo del Análisis.*

5.5.2. Análisis Exploratorio

En este apartado se pueden ver los resultados del análisis preliminar de los datos de cada estudio analizados por separado (Figura 19) para comprobar su variabilidad y la agrupación de muestras.

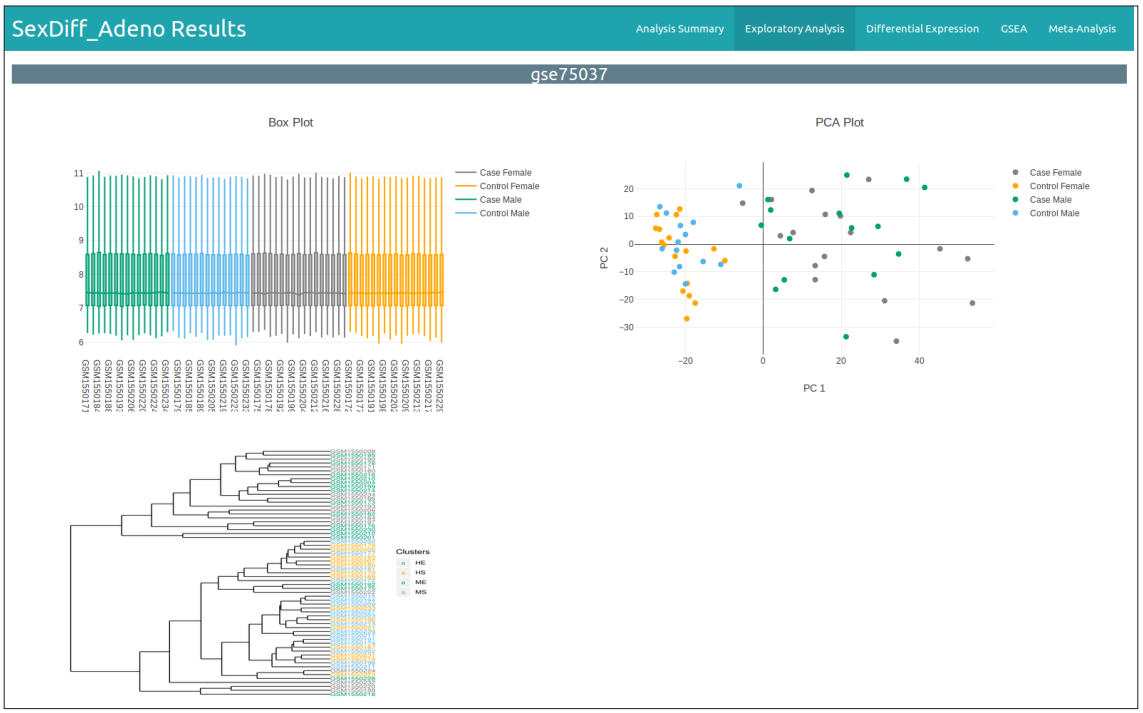

Figura 19. *Análisis Exploratorio por Estudio.*

5.5.2.1. *Boxplots*

Estos *boxplots* (Figura 20) son interactivos se puede ver cada una de las muestras de manera individual o por grupos, además se ha añadido una línea que une todas las medianas, para poder observar mejor la variabilidad de las muestras entre ellas. También pasando el ratón por encima de cada muestra aparecen los valores referentes a ella.

Figura 20. *Análisis Exploratorio (Boxplots).*

5.5.2.2. *PCA*

Los gráficos correspondientes al análisis de componentes principales (Figura 21) también son interactivos, pudiendo hacerse zoom y aislar las muestras. Igualmente si el puntero del ratón pasa por encima de una de las muestras, aparecerá la información relativa a dicha muestra.

Figura 21. *Análisis Exploratorio (PCA).*

5.5.2.3. *Análisis cluster*

En este apartado se pueden ver las gráficas generadas durante la ejecución del *pipeline* (Figura 22) que representan la separación de las muestras de forma distinta al apartado anterior para confirmar la presencia de muestras con comportamiento anómalo y verificar la distribución de las muestras en los clusters, y actuar en consideración.

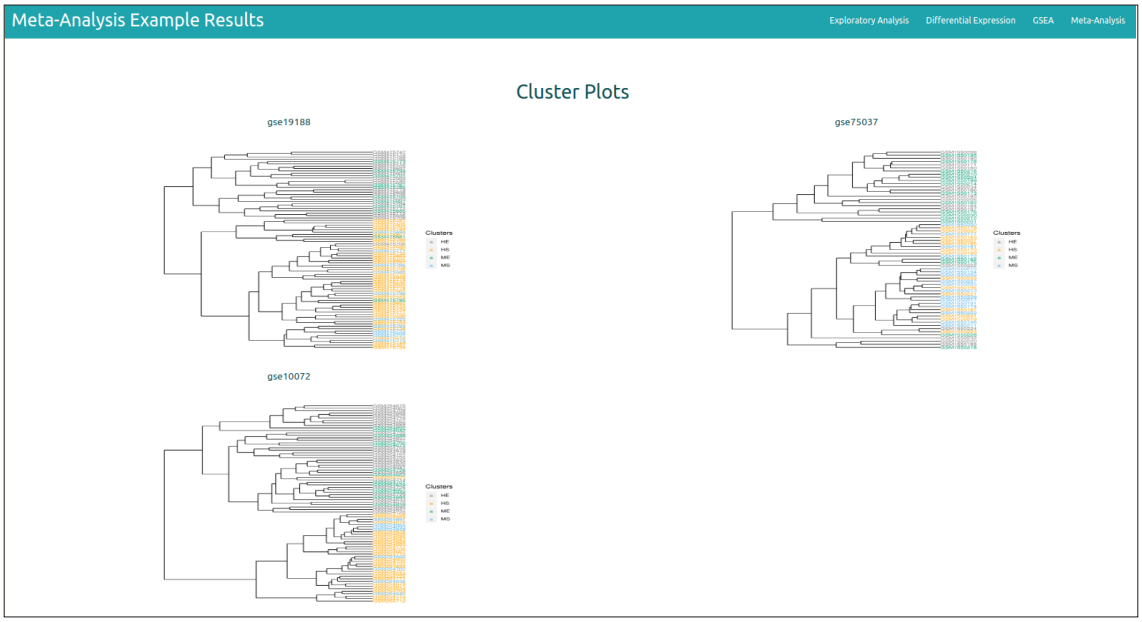

Figura 22. *Análisis Exploratorio (Cluster Plots).*

5.5.2.4. *Comentarios sobre el análisis exploratorio de los estudios*

A partir de los resultados del análisis exploratorio se puede ver gracias a la representación de las muestras en los boxplot cómo los estudios *GSE19188* y *GSE10072* están desbalanceados en cuanto a número de hombres y mujeres en los estudios. También se ve como el estudio *GSE10072* tiene una mayor dispersión en los datos (Figura 20).

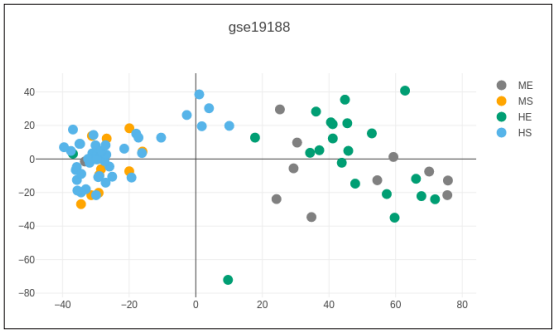

Figura 23. *PCA Estudio GSE19188.*

A partir de los gráficos PCA y del análisis cluster se puede ver que las muestras *GSM475706* y *GSM475780* del estudio *GSE19188* (Figura 23) se han agrupado con los controles habiendo sido etiquetadas como casos. Esto se puede ver en las Figuras 24 y 25, que muestran un zoom de la Figura 23 en las que se destacan las muestras anteriores. Este hecho podría deberse a un error de etiquetado, pero también podrían ser datos válidos. Para este ejemplo de metaanálisis, no se han eliminado.

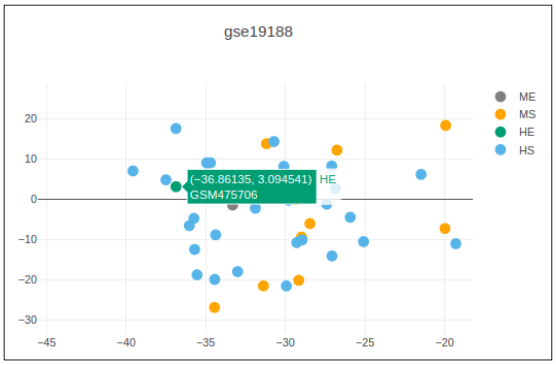

Figura 24. *Muestra GSM475706 Estudio GSE19188.*

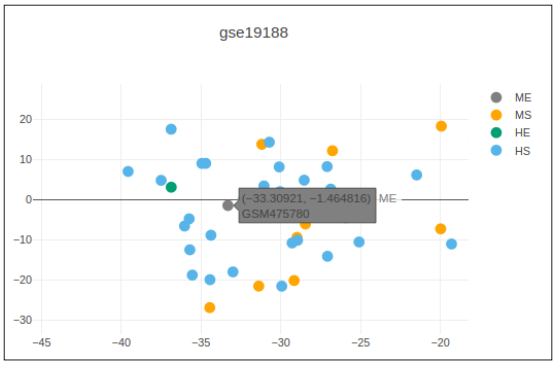

Figura 25. *Muestra GSM 475780 Estudio GSE19188.*

5.5.3. Expresión Diferencial

En este apartado se pueden ver los cincuenta mejores resultados de la expresión diferencial (Figura 26) de cada uno de los estudios por separado así como ver el gen a partir de su identificador *ENTREZ* en *NCBI*. Si el usuario desea todos los resultados podrá descargar el fichero que contiene todos los datos en formato CSV.

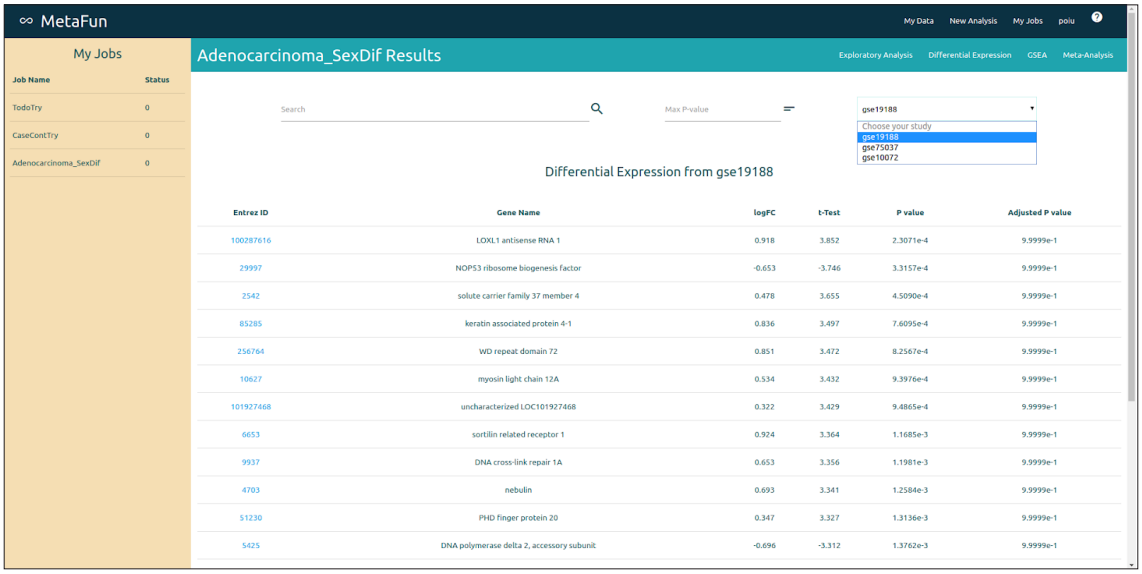

Figura 26. *Resultados, Expresión diferencial.*

5.5.3.1. *Comentarios sobre el análisis de expresión diferencial de los estudios*

Repasando los p-valores ajustados se puede ver que no hay ningún gen que resulte significativo en ninguno de los estudios. Este es uno de los casos en los que la caracterización funcional nos permitirá enriquecer nuestros resultados para poder obtener después conclusiones a partir de la función que desarrolla el gen en la célula.

5.5.4. Enriquecimiento Funcional

En este apartado se pueden ver los mejores resultados del enriquecimiento funcional (Figura 27) de cada uno de los estudios por separado así como ver la información relevante a partir del identificador del término GO en la web *QuickGO* del Instituto Europeo de Bioinformática (EBI). Si el usuario desea todos los resultados podrá descargar el fichero que contiene todos los datos en formato CSV.

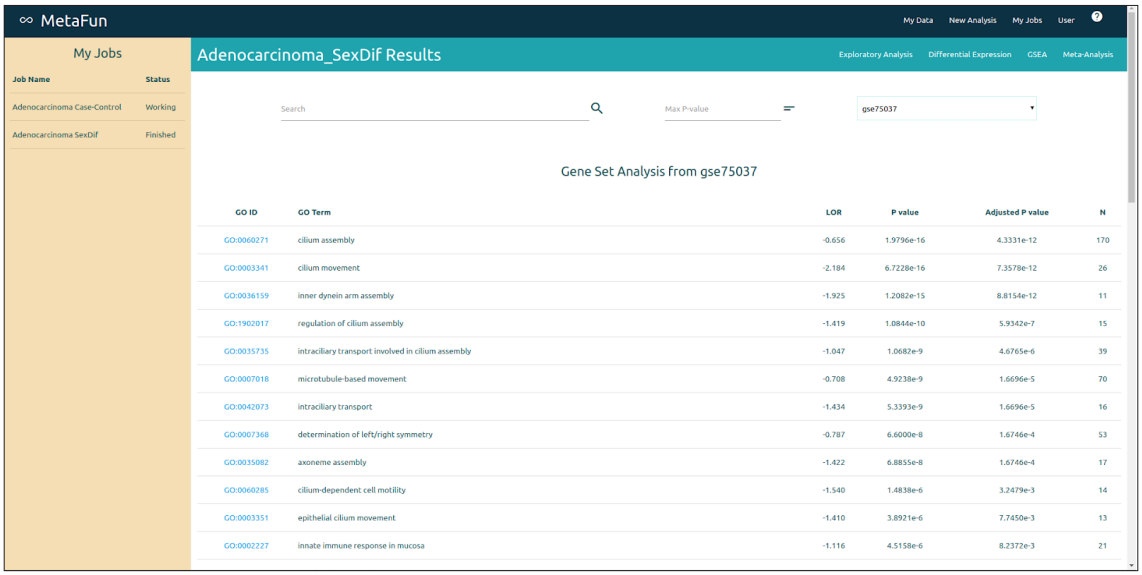

Figura 27. *Resultados, Enriquecimiento Funcional.*

5.5.4.1. *Comentarios sobre el análisis de enriquecimiento funcional*

Teniendo en cuenta el desbalanceo de muestras en los estudios comentado durante el análisis exploratorio, cabe esperar un sesgo que conduce a las funciones directamente relacionadas con el cáncer, es el caso por ejemplo del término GO:0003341 (*cilium movement*) que aparece en dos de los tres estudios con un *Log Odds Ratio* importante y un p-valor ajustado muy bajo. Este término es relevante dado que se ha demostrado que los cilios forman parte de un centro de transducción crucial que involucra vías de señalización relevantes para el desarrollo y las enfermedades como el cáncer (Higgins, Obaidi & McMorrow, 2019) [18].

5.5.4. Metaanálisis

En este apartado el usuario podrá ver los resultados finales a nivel de función genérica así como comprobar lo relevante que es cada estudio en el metaanálisis gracias a los gráficos de bosque y embudo (Figura 26).

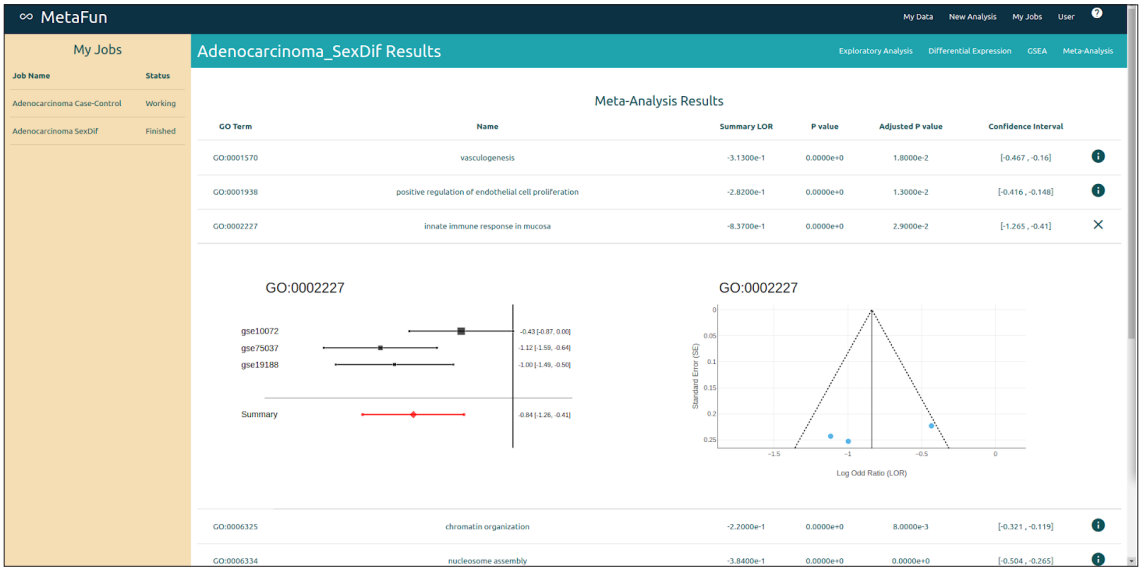

Figura 28. *Resultados, Metaanálisis.*

5.5.4.1. *Comentarios sobre los resultados del metaanálisis*

En el metaanálisis se observan funciones que indican un mayor crecimiento celular en hombres, por ejemplo, el término GO:0006335 *(DNA replication-dependent nucleosome assembly)* tiene un *Log Odds Ratio* de 0.409 y un p-valor ajustado de 0.034. Se puede decir que esta función está sobrerrepresentada en hombres ya que la comparación que se realiza es [(Mujeres Enfermas - Mujeres Sanas) - (Hombres Enfermos - Hombres Sanos)], y por tanto todos los *Log Odds Ratio* positivos harán referencia a mujeres y los *Log Odds Ratio* negativos harán referencia a una mayor expresión de la función en hombres.

Este resultado es interesante, ya que esta es una de las funciones relacionadas con el crecimiento celular lo cual ha sido previamente destacado (Pérez Díez, I. 2019) [19] por su sobrerrepresentación en hombres en adenocarcinoma de pulmón.

No obstante, estos resultados podrían ser un artefacto ya que, como se ha comentado a lo largo del trabajo, la selección de los estudios con los que se realice un

metaanálisis ha de cumplir unos requisitos estrictos de homogeneidad en lo que a tamaño muestral se refiere. En el caso de ejemplo, como se ha comentado durante la explicación del análisis exploratorio, los estudios escogidos no están balanceados en términos de sexo, y por tanto las funciones sobrerrepresentadas en hombres podrían estar sesgando los resultados.

6.
CONCLUSIONES

A partir del proceso de desarrollo expuesto a lo largo de esta memoria se puede concluir que:

1. El objetivo principal, consistente en el desarrollo de una herramienta web accesible y enfocada en el metaanálisis de datos ómicos desde un enfoque funcional, ha sido cumplido.

2. La gestión de la información de los usuarios es confidencial, la comunicación cliente-servidor es segura.

3. Se resume eficientemente la información proporcionada por el pipeline desarrollado para la realización del metaanálisis, y esta es además, descargable en formato *CSV.*

4. La arquitectura web es robusta y escalable.

5. Aunque la carga de la información es aceptablemente fluida, es posible mejorar este apartado y ya se está trabajando en ello.

6. El funcionamiento actual del apartado de manejo de datos es sencillo y correcto, aunque se está refinando para una mejor integración en el diseño final de la aplicación web.

6.1. LÍNEAS FUTURAS DE TRABAJO

Como se ha comentado durante las conclusiones, ya se está trabajando en la mejora de ciertos aspectos de la web. Sin embargo hay proyectadas varias funcionalidades que serán añadidas en cuanto sea posible.

Algunas de ellas son:

- Habilitación de un usuario anónimo para la realización de metaanálisis que no requieran mucho tiempo de cómputo.

- Interoperabilidad con repositorios de datos públicos, como *GEO* o *ArrayExpress* para que el usuario no tenga que subir los datos de todos los estudios.

- Desarrollo de un apartado para la visualización completa de los datos a partir de los resultados completos.

- Generación de informes HTML a partir de los resultados completos.

- Posibilidad de incluir anotaciones funcionales (información gen-función o cualquier otro elemento biológico) que el propio usuario pueda generar. De esta forma, la potencialidad de la herramienta se ampliará a otras ómicas (Proteómica, Metabolómica…).

- Desarrollo de un apartado que permita generar los archivos de diseño experimental desde la propia web.

7.
BIBLIOGRAFÍA

[1] Wilkinson, M., Dumontier, M., Aalbersberg, I., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J., da Silva Santos, L., Bourne, P., Bouwman, J., Brookes, A., Clark, T., Crosas, M., Dillo. I,. Dumon, O., Edmunds, S., Evelo, C., Finkers, R., Gonzalez-Beltran, A., Gray, A., Groth, P,. Goble, C., Grethe, J., Heringa, J., 't Hoen, P., Hooft, R., Kuhn, T., Kok, R., Kok, J., Lusher, S., Martone, M., Mons, A., Packer, A., Persson, B., Rocca-Serra, P., Roos, M., van Schaik, R., Sansone, S., Schultes, E., Sengstag, T., Slater, T., Strawn, G., Swertz, M., Thompson, M., van der Lei, J., van Mulligen, E., Velterop, J., Waagmeester, A., Wittenburg, P., Wolstencroft, K., Zhao, J., Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data, 3*(1). doi: 10.1038/sdata.2016.18

[2] Al-Shahrour, F., Carbonell, J., Minguez, P., Goetz, S., Conesa, A., Tarraga, J., Medina, I., Alloza, I., Montaner, D., Dopazo, J. (2008). Babelomics: advanced functional profiling of transcriptomics, proteomics and genomics experiments. *Nucleic Acids Research, 36*(Web Server), W341-W346. doi: 10.1093/nar/gkn318

[3] Hidalgo, M.R., Çubuk, C., Amadoz, A., Salavert, F., Carbonell-Caballero, J., Dopazo, J. (2017). "High throughput estimation of functional cell activities reveals disease mechanisms and predicts relevant clinical outcomes." *Oncotarget, 3*(8), 5160-5178.

[4] García-García F. (2016). *Métodos de Análisis de Enriquecimiento Funcional en Estudios Genómicos*. 2016; Universitat de València.

[5] Tukey, J. (1962). The Future of Data Analysis. *The Annals Of Mathematical Statistics, 33*(1), 1-67. doi: 10.1214/aoms/1177704711

[6] Ritchie, M., Phipson, B., Wu, D., Hu, Y., Law, C., Shi, W., & Smyth, G. (2015). Limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Research, 43*(7), e47-e47. doi: 10.1093/nar/gkv007

[7] Montaner, D., & Dopazo, J. (2010). Multidimensional Gene Set Analysis of Genomic Data. *Plos ONE, 5*(4), e10348. doi: 10.1371/journal.pone.0010348.

[8] Glass, G.V. (1976). "Primary, Secondary, and Meta-Analysis of Research." *Educational Researcher, 5*(10), 3–8.

[9] MongoDB. https://www.mongodb.com/

[10] R. Rivest. The MD5 Message-Digest Algorithm [rfc1321]

[11] Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software, 36*(3), 1-48. URL: http://www.jstatsoft.org/v36/i03/

[12] Angular Framework Documentation. https://angular.io/guide/architecture

[13] TypeScript. https://www.typescriptlang.org/

[14] Java. https://www.java.com/en/

[15] Jersey. https://jersey.github.io/

[16] R. https://www.r-project.org/

[17] Plot.ly. https://plot.ly/

[18] Higgins, M., Obaidi, I., & McMorrow, T. (2019). Primary cilia and their role in cancer (Review). *Oncology Letters.* doi: 10.3892/ol.2019.9942

[19] Pérez Díez, I. (2019). Metaanálisis Funcional de las diferencias de sexo en estudios ómicos de adenocarcinoma de pulmón (TFM). Máster Universitario en Bioinformática, Universitat de València.