

RESEARCH

# Mitocondrias anormales en el tejido muscular

Julia Del Río Toledo<sup>\*</sup>  
, Javier Méndez Parrilla  
, Juan Jesús Rus Muñoz  
and Benjamín Cortés Fernández

<sup>\*</sup>Correspondence:  
[juliadelrio2003@uma.es](mailto:juliadelrio2003@uma.es)  
ETSI Informática, Universidad de  
Málaga, Málaga, España  
Full list of author information is  
available at the end of the article

## Abstract

**Keywords:** sample; article; author

## 1 Introducción

Las mitocondrias, además de ser la principal fuente de ATP, desempeñan funciones esenciales en la regulación del metabolismo celular, incluyendo la catabolización de nutrientes y la gestión de la homeostasis redox [1]. A lo largo de la evolución eucariota, su forma y comportamiento se adaptaron para garantizar la transmisión precisa de su genoma y responder a las demandas celulares [2]. En tejidos de alta demanda energética como el músculo esquelético, las mitocondrias juegan un papel crucial en la flexibilidad metabólica y la adaptación a estímulos como el ejercicio [3, 4]. La disfunción mitocondrial contribuye a diversas patologías [5], neurodegenerativas, cancerígenas [6] o musculoesqueléticas [7].

La relación entre las mitocondrias y el músculo esquelético es fundamental para comprender el metabolismo energético y la adaptación muscular. La biogénesis mitocondrial y la captación de glucosa estimulada por contracción son procesos clave que se ven influenciados por factores como el óxido nítrico [8] o el OPA1 (Optic Atrophy 1) [9]. Además, el ejercicio regular desencadena adaptaciones moleculares que optimizan el rendimiento muscular y la salud mitocondrial [10, 11]. Sin embargo, con el envejecimiento, la dinámica mitocondrial y la capacidad de mitofagia pueden verse comprometidas, afectando así la funcionalidad muscular [12, 13].

La mitofagia es el proceso mediante el cual las mitocondrias dañadas se eliminan selectivamente. Los defectos en dicho proceso pueden llevar a la acumulación de mitocondrias disfuncionales, afectando aún más la salud muscular. [13] La disfunción mitocondrial [14] en estos tejidos se ha asociado con varias patologías relacionadas con la edad, incluidas la insuficiencia cardíaca y la sarcopenia [15]. La reducción de la función mitocondrial en el músculo esquelético contribuye a la pérdida de masa muscular y fuerza, lo que afecta significativamente la movilidad y la calidad de vida. Otro factor determinante que provocan las mitocondrias disfuncionales es el aumento de ROS (especies reactivas de oxígeno), provocando daño oxidativo en músculos y otros tejidos. [16]

Además, la disrupción en la dinámica mitocondrial (fusión y fisión) contribuye a la atrofia muscular, con un aumento de mitocondrias hinchadas y fragmentadas. Estas alteraciones son comunes en enfermedades como la distrofia muscular, donde

la pérdida de proteínas estructurales agrava los efectos negativos de la disfunción mitocondrial en el tejido muscular.

En concreto, este estudio, examinará los genes vinculados a la patología HP:0008316 (mitocondrias anormales en el tejido muscular) mediante técnicas de biología de sistemas. Nuestro objetivo es investigar, construir y analizar redes biológicas para identificar las vías y procesos relacionados con las enfermedades asociadas a esta condición.

## 2 Materiales y métodos

### 2.1 Materiales

#### 2.1.1 *Programas*

**R** es un lenguaje de programación y un entorno estadístico muy utilizado en bioinformática, análisis de datos y ciencia de datos. Sirve para realizar análisis estadísticos complejos, visualización de datos y manipulación de datos biológicos, como análisis de expresión genética para lo que lo estamos utilizando en este caso. Fue desarrollado por Robert Gentleman y Ross Ihaka en la Universidad de Auckland, Nueva Zelanda, y es un proyecto de código abierto mantenido por la comunidad R.[17]

**Python** es un lenguaje de programación, conocido por su simplicidad, versatilidad y eficiencia a la hora de programar. Se utiliza en análisis de datos, desarrollo de modelos, análisis bioinformático, y en el manejo y procesamiento de grandes volúmenes de datos biológicos. Fue creado por Guido van Rossum y ha crecido hasta convertirse en uno de los lenguajes más populares, con una comunidad activa que proporciona múltiples herramientas y bibliotecas.[18]

#### 2.1.2 *Bases de Datos*

**HPO** (Human Phenotype Ontology) es una base de datos que proporciona una clasificación estándar de los términos fenotípicos humanos, cada uno con un identificador único. En el contexto de la investigación genética, HPO permite relacionar los fenotipos observables ("mitocondrias anormales en el tejido muscular", identificado por HP:0008316) con genes y enfermedades específicas. Esto facilita el análisis y busca de datos genéticos en función de características fenotípicas. Es mantenido por la comunidad científica, específicamente por el equipo de Monarch Initiative, para mejorar la precisión en los estudios de fenotipos y genética humana.[19]

**StringDB** es una base de datos que contiene información sobre interacciones proteína-proteína (PPI), recopilada de diversas fuentes como experimentos, bases de datos públicas y predicciones computacionales. Esta base de datos permite investigar redes de proteínas interconectadas, facilitando el análisis de cómo las proteínas interactúan en diferentes contextos biológicos, algo esencial en estudios genéticos como los relacionados con HPO. Desarrollada por un consorcio de investigadores y está disponible como un recurso en línea gratuito para la comunidad científica.[20]

El **NCBI** (National Center for Biotechnology Information) es un centro de investigación en biotecnología de Estados Unidos que proporciona acceso a una vasta

colección de bases de datos biológicas y genómicas. Es utilizado para acceder a datos sobre genes, proteínas y secuencias genómicas, y proporciona herramientas de análisis. Nosotros lo utilizamos junto a HPO, para facilitar la correlación entre genes específicos y fenotipos. Fue fundado por los Institutos Nacionales de Salud (NIH) de Estados Unidos y es un recurso fundamental para la investigación genética y bioinformática.[21]

### 2.1.3 Bibliotecas

**Igraph** es una biblioteca para el análisis y la visualización de redes complejas, desarrollada en múltiples lenguajes, incluyendo Python y R. En bioinformática, se usa para construir y analizar redes de interacciones biológicas, como redes de proteínas o genes, ayudando a visualizar cómo las proteínas relacionadas con un fenotipo específico pueden interactuar entre sí. Desarrollado por una comunidad internacional de programadores y científicos.[22]

**Pandas** es una biblioteca en Python que facilita la manipulación y el análisis de datos mediante estructuras de datos como DataFrames. En estudios bioinformáticos, pandas se utiliza para manejar y procesar grandes volúmenes de datos, como los obtenidos de bases de datos genéticas. Es útil para organizar los datos de HPO y facilitar su análisis. Fue creada por Wes McKinney y es ahora una biblioteca ampliamente adoptada en el ámbito de la ciencia de datos.[23]

**Requests** es una biblioteca de Python que permite realizar solicitudes HTTP de forma sencilla. Es utilizada para acceder a APIs y extraer datos de bases de datos en línea, como HPO o StringDB, permitiendo automatizar la obtención de datos relevantes para el estudio de fenotipos y genes. Fue desarrollada por Kenneth Reitz y es una de las bibliotecas más populares para trabajar con APIs en Python.[24]

**Networkx** es una biblioteca en Python para la creación, manipulación y análisis de redes complejas. Se utiliza para modelar y analizar redes de interacciones genéticas o de proteínas. Por ejemplo, en el análisis de datos de StringDB, networkx permite estudiar las relaciones entre proteínas de forma visual y cuantitativa. Es una biblioteca de código abierto mantenida por la comunidad, utilizada ampliamente en estudios de redes biológicas y sociales.[25]

**Matplotlib.pyplot** es una biblioteca de Python que permite crear gráficos y visualizaciones de datos de forma detallada y personalizada. Lo utilizamos para representar datos biológicos y resultados de análisis de manera visual, lo cual facilita la interpretación de datos complejos, como los patrones de interacción entre proteínas o genes relacionados con un fenotipo específico. Fue desarrollada por John D. Hunter y es mantenida por la comunidad científica y de datos.[26]

## 2.2 Métodos

### 2.2.1 Obtención de datos y construcción de la red de interacciones

El análisis comenzó con la identificación de genes asociados al fenotipo de interés, definido por el código HP:0008316, que corresponde a "mitocondrias anormales

en tejido muscular” en la Ontología del Fenotipo Humano (HPO, por sus siglas en inglés). Este paso fue crucial para establecer una base de datos inicial que sustentara el estudio.

Para llevar a cabo este análisis de manera sistemática, desarrollamos un script en Python llamado *hpo\_genes\_fetcher.py* que interactuaba con la API de HPO. La elección de Python como lenguaje principal se basó en su flexibilidad y en su amplio ecosistema de bibliotecas, como *requests* y *pandas*, que son ideales para manejar datos y realizar solicitudes con las APIs. El script se diseñó para automatizar la extracción de genes relacionados con el fenotipo, lo que evitaba errores manuales y garantizaba la reproducibilidad del proceso.

Se utilizó la biblioteca *requests* para realizar solicitudes HTTP de tipo GET a los endpoints específicos de la API de HPO. El identificador del fenotipo, HP:0008316, se incluyó como parámetro en la URL de la solicitud, asegurando que los datos recuperados estuvieran específicamente relacionados con este término fenotípico. La respuesta de la API, en formato JSON (estándar ampliamente utilizado para el intercambio de datos estructurados en aplicaciones web), contenía una lista dentro del campo “genes”, que incluía los nombres de los genes asociados al fenotipo. Este formato estructurado fue procesado utilizando la biblioteca *pandas*, lo que permitió organizar la información en un DataFrame, una estructura de datos tabular que facilita su análisis y manipulación.

Una vez organizados, los datos se depuraron para eliminar duplicados o entradas incompletas, con lo que se garantizó que la lista de genes obtenida estuviera lista para su uso en análisis posteriores. Finalmente, los genes se almacenaron en un archivo de texto estructurado para facilitar su acceso y uso en las etapas siguientes del estudio.

La lista de genes recopilada fue utilizada como entrada en un segundo script de Python, *fetch\_interactions.py*, que se encargó de analizar las interacciones proteicas entre los genes utilizando la API de **STRINGdb**. Este análisis es crucial para comprender cómo interactúan entre sí las proteínas codificadas por los genes, lo que puede revelar información sobre sus funciones en el fenotipo.

Utilizando dicha API, el script envió solicitudes POST que incluían los nombres de los genes formateados con %0d como delimitador, cumpliendo con los requisitos específicos de la API. Se configuraron parámetros como la especie de interés (Homo sapiens, código NCBI: 9606) y un umbral de confianza de 0.5, significando esto que las interacciones incluidas tienen al menos un 50% de probabilidad de ser funcionalmente relevantes, según los cálculos de STRINGdb.

La respuesta de la API, en formato JSON, proporcionó información detallada sobre las interacciones, incluyendo los genes conectados, la naturaleza de las interacciones y una puntuación de confianza. Además, el script incluyó una función para descargar y guardar una representación visual de la red directamente desde STRINGdb en formato PNG, lo que facilitó la documentación gráfica de las interacciones proteicas.

### 2.2.2 Análisis de la red y visualización avanzada

El análisis detallado de la red de interacciones proteicas se realizó en **R** utilizando la biblioteca **iGraph**. Este paso fue fundamental para profundizar en las propiedades

estructurales y funcionales de la red, permitiendo explorar relaciones clave entre los genes asociados a nuestro fenotipo. Para comenzar, los datos generados previamente con STRINGdb, que contenían la lista de nodos y aristas, fueron exportados en formato CSV y posteriormente importados en R mediante las funciones estándar *read.csv()*. Este enfoque aseguró la interoperabilidad entre Python y R, facilitando la integración del flujo de trabajo.

En R, los datos se transformaron en un grafo dirigido utilizando la función *graph\_from\_data\_frame()*. Este grafo representaba los genes como nodos y las interacciones entre ellos como aristas ponderadas, donde el peso correspondía a la puntuación de confianza calculada por STRINGdb. La dirección de aristas permitió representar el flujo funcional entre los genes.

La visualización inicial del grafo se realizó utilizando la función *plot()* de iGraph, con personalizaciones que resaltaban características clave de la red. Se ajustaron colores y tamaños de los nodos según su grado de conectividad, lo que permitió identificar rápidamente los genes más conectados o "hubs". Para optimizar la interpretación visual, se probaron diferentes algoritmos de disposición, como *layout\_with\_fr* (disposición de Fruchterman-Reingold) y *layout\_nicely*, que distribuyen los nodos de manera equilibrada según sus conexiones. Estas configuraciones facilitaron la identificación de patrones estructurales en la red y permitieron una representación gráfica intuitiva de las interacciones proteicas.

Además del grado de los nodos, se calcularon otras métricas topológicas relevantes, como la centralidad de intermediación, que identifica genes clave para el flujo de información, y el coeficiente de agrupamiento, que mide la tendencia de los genes a formar clústeres densos.

La distribución del grado de los nodos se realizó mediante histogramas, que mostraban cómo se distribuían las conexiones entre los genes. Adicionalmente, se generaron mapas de calor que representaban la intensidad de las interacciones (pesos de las aristas), proporcionando una visión global de las relaciones más relevantes. Estas visualizaciones se realizaron utilizando bibliotecas de R como *ggplot2* y *heatmap()*, que facilitaron la interpretación de los datos.

Finalmente, se evaluaron las comunidades detectadas mediante algoritmos de agrupamiento. Cada comunidad fue analizada en términos de su relevancia funcional en el contexto del fenotipo, considerando la conectividad interna y los genes principales en cada módulo. Este enfoque permitió identificar subgrupos funcionales de genes, lo que ayudó a interpretar mejor las relaciones biológicas subyacentes.

### 2.2.3 Análisis de enriquecimiento funcional

## 3 Resultados

## 4 Discusión

## 5 Conclusiones

### Abreviaciones

Indicar lista de abreviaciones mostrando cada acrónimo a que corresponde

### Disponibilidad de datos y materiales

[Enlace al proyecto en GitHub](#)

### Contribución de los autores

Usando las iniciales que habéis definido al comienzo del documento, debeis indicar la contribución al proyecto en el estilo: J.E : Encargado del análisis de coexpresión con R, escritura de resultados; J.R.S : modelado de red con python y automatizado del código, escritura de métodos; ... OJO: que sea realista con los registros que hay en vuestros repositorios de github.

**Author details**

ETSI Informática, Universidad de Málaga, Málaga, España.

**References**

1. Spinelli, J.B., Haigis, M.C.: The multifaceted contributions of mitochondria to cellular metabolism (2018). doi:[10.1038/s41556-018-0124-1](https://doi.org/10.1038/s41556-018-0124-1)
2. Friedman, J.R., Nunnari, J.: Mitochondrial form and function (2014). doi:[10.1038/nature12985](https://doi.org/10.1038/nature12985)
3. Memme, J.M., Erlich, A.T., Phukan, G., Hood, D.A.: Exercise and mitochondrial health. *Journal of Physiology* **599** (2021). doi:[10.1113/JP278853](https://doi.org/10.1113/JP278853)
4. Smith, J.A.B., Murach, K.A., Dyar, K.A., Zierath, J.R.: Exercise metabolism and adaptation in skeletal muscle (2023). doi:[10.1038/s41580-023-00606-x](https://doi.org/10.1038/s41580-023-00606-x)
5. Quintana-Cabrera, R., Scorrano, L.: Determinants and outcomes of mitochondrial dynamics (2023). doi:[10.1016/j.molcel.2023.02.012](https://doi.org/10.1016/j.molcel.2023.02.012)
6. Chan, D.C.: Mitochondrial dynamics and its involvement in disease. *Annual Review of Pathology: Mechanisms of Disease* **15** (2020). doi:[10.1146/annurev-pathmechdis-012419-032711](https://doi.org/10.1146/annurev-pathmechdis-012419-032711)
7. Liu, S.Z., Marcinek, D.J.: Skeletal muscle bioenergetics in aging and heart failure. *Heart Failure Reviews* **22** (2017). doi:[10.1007/s10741-016-9586-z](https://doi.org/10.1007/s10741-016-9586-z)
8. McConell, G.K., Wadley, G.D.: Potential role of nitric oxide in contraction-stimulated glucose uptake and mitochondrial biogenesis in skeletal muscle. In: *Clinical and Experimental Pharmacology and Physiology*, vol. 35 (2008). doi:[10.1111/j.1440-1681.2008.05038.x](https://doi.org/10.1111/j.1440-1681.2008.05038.x)
9. Noone, J., O’Gorman, D.J., Kenny, H.C.: OPA1 regulation of mitochondrial dynamics in skeletal and cardiac muscle (2022). doi:[10.1016/j.tem.2022.07.003](https://doi.org/10.1016/j.tem.2022.07.003)
10. Hargreaves, M., Spriet, L.L.: Skeletal muscle energy metabolism during exercise (2020). doi:[10.1038/s42255-020-0251-4](https://doi.org/10.1038/s42255-020-0251-4)
11. Egan, B., Zierath, J.R.: Exercise metabolism and the molecular regulation of skeletal muscle adaptation (2013). doi:[10.1016/j.cmet.2012.12.012](https://doi.org/10.1016/j.cmet.2012.12.012)
12. Hood, D.A., Memme, J.M., Oliveira, A.N., Triolo, M.: Maintenance of Skeletal Muscle Mitochondria in Health, Exercise, and Aging (2019). doi:[10.1146/annurev-physiol-020518-114310](https://doi.org/10.1146/annurev-physiol-020518-114310)
13. J, L.-G.: Mitochondrial dynamics and mitophagy in skeletal muscle health and aging. *International Journal of Molecular Sciences* **22** (2021). doi:[10.3390/ijms22158179](https://doi.org/10.3390/ijms22158179)
14. Chen, T.H.: Mitochondrial dysfunction as an underlying cause of skeletal muscle disorders. *International Journal of Molecular Sciences* **23** (2022). doi:[10.3390/ijms232112926](https://doi.org/10.3390/ijms232112926)
15. Boengler, K.: Mitochondria and ageing: role in heart, skeletal muscle and adipose tissue. *Journal of Cachexia, Sarcopenia and Muscle* **8** (2017). doi:[10.1002/jcsm.12178](https://doi.org/10.1002/jcsm.12178)
16. Javadov, S.: Mitochondria-targeted antioxidant preserves contractile properties and mitochondrial function of skeletal muscle in aged rats. *Oncotarget* **6** (2015). doi:[10.18632/oncotarget.5783](https://doi.org/10.18632/oncotarget.5783)
17. Jiménez, J.U.: Introducción a r y rstudio (2019)
18. Teams, P.: What is Python? Executive Summary (2023)
19. Köhler, S., Doelken, S.C., Mungall, C.J., Bauer, S., Firth, H.V., Bailleul-Forestier, I., Black, G.C., Brown, D.L., Brudno, M., Campbell, J., *et al.*: The human phenotype ontology project: linking molecular biology and disease through phenotype data. *Nucleic acids research* **42**(D1), 966–974 (2014)
20. Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J., Simonovic, M., Roth, A., Santos, A., Tsafou, K.P., *et al.*: String v10: protein–protein interaction networks, integrated over the tree of life. *Nucleic acids research* **43**(D1), 447–452 (2015)
21. Jenuth, J.P.: The ncbi: Publicly available tools and resources on the web. *Bioinformatics methods and protocols*, 301–312 (1999)
22. Valdez, B.: Análisis de grafos usando r e igraph. *altamira* **1**(1), 1 (2016)
23. McKinney, W., *et al.*: pandas: a foundational python library for data analysis and statistics. *Python for high performance and scientific computing* **14**(9), 1–9 (2011)
24. Chandra, R.V., Varanasi, B.S.: *Python Requests Essentials*. Packt Publishing Birmingham, UK, ??? (2015)
25. Hagberg, A., Conway, D.: *Networkx: Network analysis with python*. URL: <https://networkx.github.io> (2020)
26. Ari, N., Ustazhanov, M.: Matplotlib in python. In: 2014 11th International Conference on Electronics, Computer and Computation (ICECCO), pp. 1–6 (2014). IEEE