

RESEARCH

Relación de genes con la aparición del papiloma como fenotipo patológico.

Eva M. Ayala Fernández, Raúl A. Castro Valderas, María J. Hidalgo Rodríguez, Francisco J. Rodríguez-cordoba Lucena and María Vida Montañez*

*Correspondence:
mariavida262001@uma.es
ETSI Informática, Universidad de
Málaga, Málaga, España
Full list of author information is
available at the end of the article

Abstract

Keywords: VPH; papiloma; papila; tumor; genes; cáncer

1 Introducción

Un papiloma es un tumor epitelial benigno que crece de manera exofítica [1], es decir, proyectándose hacia afuera en forma de proyecciones y de manera no agresiva ni propagándose por todo el cuerpo. En este contexto, "papila" se refiere a la proyección creada por el tumor, no a un tumor en una papila ya existente. Cuando se utiliza sin contexto específico, con frecuencia se refiere a infecciones causadas por el virus del papiloma humano (VPH). Existen casi 200 tipos distintos de VPH [2], y muchos de ellos son carcinogénicos. Sin embargo, también existen otras condiciones que pueden causar papilomas, así como muchos casos en los que la causa no se conoce. Las infecciones por el VPH de riesgo alto en ocasiones causan cáncer en las partes del cuerpo en las que el VPH infecta a las células. Por ejemplo, cáncer de cuello uterino, cáncer vulvar, cáncer vaginal, cáncer de pene, cáncer anal y cánceres orofaríngeos positivos para el VPH. La carga de los cánceres relacionados con el virus del papiloma humano causa cerca del 5% [3] de todos los cánceres en el mundo, se calcula que 570.000 mujeres y 60.000 hombres tienen un cáncer relacionado con el VPH cada año [3]. El cáncer de cuello uterino es el más frecuente de todos los causados por el VPH, debido a que este es la causa de casi todos los cánceres de cuello uterino del mundo [3].

1.1 Información sobre los genes a estudiar

A continuación, veremos una breve información de los genes con mayor grado de interconexión que hemos encontrado al realizar un análisis fenotípico (HPO).

AKT1: La proteína quinasa B (Akt1) tiene un papel fundamental en el crecimiento y la supervivencia celular al transducir señales en la cascada de señalización celular (PI3K)/AKT, la cual genera mensajeros que participan en la regulación de la progresión del ciclo celular, adhesión y migración. La vía PI3K/AKT es una de las que suelen estar afectadas en distintos tipos de cáncer en humanos, como el cáncer de ovario, de mama, y de Cowden. Además, se asocia los niveles altos de fosforilación de la proteína con los peores pronósticos de cáncer [4].

TP53: TP53 (tumor protein p53) es un gen supresor de tumores involucrado en procesos biológicos fundamentales para la estabilidad genética. Las mutaciones en

este gen han sido asociadas con un peor pronóstico para pacientes con carcinoma oral de células escamosas, dando lugar a un cáncer más agresivo al combinarse con el VPH [5].

HRAS: Este gen participa en la regulación de la vía de la proteína quinasa activada por mitógenos (MAPK) y mediada por la proteína quinasa Raf. En los últimos años se han definido una serie de síndromes con mutaciones en genes implicados en esta vía Ras/PAPK, entre ellos el síndrome de Costellos. Este síndrome refleja características cutáneas distintivas, como papillomas [4].

SDHD: Las mutaciones en este gen están asociadas con la formación de tumores, incluyendo el paraganglioma hereditario. La transmisión de la enfermedad ocurre casi exclusivamente a través del alelo paterno, lo que sugiere que este locus puede estar impreso maternalmente. Hay pseudogenes para este gen en los cromosomas 1, 2, 3, 7 y 18. Resultados de empalme alternativos en múltiples variantes de transcripción [6].

SDHB: Las mutaciones en este gen dan como resultado feocromocitoma y paraganglioma. Las alteraciones germinales y variaciones en este gen causan el síndrome de Cowden. Recientemente se ha reconocido el cáncer de endometrio como un componente importante de este síndrome [7].

1.2 Hipótesis del trabajo

Al mapear los genes asociados con el papiloma en una red de interacción proteína-proteína, podemos modelar este fenotipo y buscar grupos de genes que formen conglomerados dentro de la red. Al hacerlo, planteamos la hipótesis de que encontraremos grupos de genes involucrados en procesos subyacentes importantes relacionados con el crecimiento tumoral, y diversos cánceres como el de cuello uterino.

1.3 Objetivos

- 1 Obtener genes relacionados con el papiloma.
- 2 Utilizar estos genes para hacer una network para modelar los procesos subyacentes del papiloma.
- 3 Analizar esta red para encontrar grupos de genes que interactúan entre sí.
- 4 Investigar las funciones de los genes en estos grupos para obtener información sobre los subyacentes mecanismos que conducen al papiloma (aquí podemos ver si están relacionados en las mismas enfermedades).

2 Materiales y métodos

2.1 Materiales

A continuación, explicaremos los recursos y herramientas utilizados para los experimentos:

Human Phenotype Ontology

HPO es un vocabulario estandarizado de anomalías fenotípicas en enfermedades humanas que utiliza un fenotipado detallado para poder ser usado a nivel computacional [8].

R

R, en su esencia, es un lenguaje destinado a la exploración estadística y la creación

de representaciones gráficas. Se configura como un entorno de programación compuesto por un conjunto de herramientas altamente adaptables, cuya funcionalidad puede ser expandida con facilidad a través de la integración de paquetes, bibliotecas o mediante la creación de funciones personalizadas. Además, se destaca por ser una plataforma de código abierto y gratuita, enmarcada en el proyecto GNU, compartiendo este enfoque con sistemas como Linux o aplicaciones como Mozilla Firefox. En nuestro caso hemos trabajado con la versión 4.3.1 de R [9].

Python

Python es un lenguaje de programación versátil de alto nivel ampliamente empleado en el desarrollo de diversas aplicaciones. A diferencia de lenguajes como Java o .NET, Python es interpretado, lo que significa que no requiere un proceso de compilación antes de ejecutar las aplicaciones. En lugar de eso, las aplicaciones escritas en Python se ejecutan directamente en la computadora utilizando un intérprete, eliminando la necesidad de traducción a lenguaje de máquina previamente. Este enfoque agiliza el desarrollo y ejecución de programas en Python. La versión utilizada de Python fue la 3.12.0 [10].

String

String es una base de datos que alberga información sobre interacciones entre proteínas, tanto aquellas conocidas como las predichas. Estas interacciones abarcan desde asociaciones directas (físicas) hasta indirectas (funcionales). La base de datos recopila datos de diversas fuentes, que incluyen repositorios experimentales, métodos de predicción computacional y colecciones de textos públicos. Cada interacción está evaluada con una puntuación de condensación combinada que sintetiza las diversas evidencias disponibles[11].

iGraph

iGraph es una biblioteca rápida y de código abierto para el análisis de grafos o redes. El núcleo de esta librería está implementado en C y dispone de enlaces para su uso con lenguajes de alto nivel como R, Python y Mathematica [12].

Pandas

Pandas es una biblioteca de programación en Python diseñada para facilitar el análisis y la manipulación de datos. Se centra en estructuras de datos como el "DataFrame", una tabla bidimensional, y ofrece funciones para cargar, limpiar y transformar datos de manera eficiente. Es ampliamente utilizado en ciencia de datos y análisis estadístico [13].

Algoritmos de clusterización

Los algoritmos de clusterización son técnicas utilizadas en análisis de datos para dividir un conjunto de datos en grupos o "clústeres" basándose en similitudes entre los elementos. El objetivo es agrupar datos que sean más similares entre sí y más diferentes de otros grupos. Estos algoritmos ayudan a descubrir patrones y estructuras intrínsecas en los datos sin necesidad de etiquetas preexistentes. Los que se utilizaron para el trabajo fueron los siguientes:

- **Algoritmo de Givan-Newman:** es un método utilizado para detectar comunidades en redes complejas. Este algoritmo se basa en la idea de eliminar gradualmente los enlaces más importantes de una red para revelar su estructura de comunidad.

•**Algoritmo de optimización voraz:** también conocido como algoritmo ávido o greedy, es un enfoque de resolución de problemas que toma decisiones locales en cada etapa con la esperanza de encontrar una solución óptima global. En cada paso, el algoritmo selecciona la mejor opción disponible en ese momento, sin considerar las posibles consecuencias a largo plazo.

•**Algoritmo de propagación de etiquetas:** también conocido como "propagación de la afinidad," es un método de agrupamiento basado en la similitud entre los datos. A diferencia de otros algoritmos de agrupamiento que requieren la especificación del número de clústeres, la propagación de etiquetas es un algoritmo de agrupamiento sin la necesidad de definir previamente el número de clústeres.

•**Algoritmo de Louvain:** es un algoritmo de optimización utilizado para la detección de comunidades en redes o grafos. Su objetivo es encontrar una partición modular del grafo que maximice la modularidad. La modularidad es una medida que cuantifica la calidad de la partición de un grafo en comunidades, considerando la densidad de conexiones dentro de las comunidades y la rareza de conexiones entre ellas.

Linkcomm

Las comunidades de enlaces revelan la estructura anidada y superpuesta en las redes y descubren los nodos clave que forman conexiones con varias comunidades. Linkcomm proporciona un conjunto de herramientas para generar, visualizar y analizar comunidades de enlaces en redes de tamaño y tipo arbitrarios. El paquete linkcomm también incluye herramientas para generar, visualizar y analizar comunidades de Generadores de Clúster Superpuestos (OCG) [14].

2.2 Métodos

En primer lugar creamos una función que busca genes asociados a un fenotipo específico utilizando la API de **HPO**. La función recibe el parámetro fenotipo, que es el código HPO para el fenotipo de interés. Realizamos una solicitud GET a la API de HPO para obtener información sobre genes asociados al fenotipo. Si la solicitud es exitosa (código de estado 200), extraemos y devolvemos los genes asociados al fenotipo. Si no, imprimimos un mensaje de error y devolvemos una lista vacía.

Utilizamos la función *buscar_genes_por_fenotipo* para obtener los genes asociados al fenotipo "Papilloma" (código HPO: HP:0012740). Si encontramos genes asociados, imprimimos los nombres de los genes. Si no encontramos genes asociados, mostramos un mensaje indicando que no se encontraron genes.

Tras esto construimos una URL con los símbolos de los genes obtenidos anteriormente para realizar una solicitud a la API de StringDB, que proporciona información sobre interacciones de proteínas. Realizamos una solicitud GET a la API de StringDB para obtener una red de interacciones de proteínas. Guardamos la respuesta como una imagen llamada 'img.png'.

Imprimimos y filtramos los resultados del análisis de enriquecimiento por categorías ('Process' y 'KEGG'). Realizamos búsquedas específicas en los resultados de enfermedades y genes utilizando palabras clave y patrones de genes relevantes, respectivamente. Guardamos los resultados de las búsquedas en archivos CSV.

Realizamos una solicitud GET a la API de StringDB para descargar la red de interacciones en formato TSV. Si la descarga es exitosa, guardamos el archivo con el

nombre proporcionado por la respuesta o como *'red_descargada.tsv'* si no se obtiene un nombre de archivo. Si hay un error en la descarga, imprimimos un mensaje de error.

Cargamos el archivo TSV descargado. Seleccionamos columnas específicas (*'preferredName'_A* y *'preferredName'_B*) y eliminamos duplicados. Guardamos estas columnas en un nuevo archivo de texto llamado *'genes_igraph.txt'*.

2.3 Propiedades de la red

Nosotros usamos la biblioteca de **igraph** disponible en el lenguaje de programación de R para realizar el análisis y la visualización de la red. En primer lugar, importamos la librería y leímos el fichero que obtuvimos al generar la red con la API de STRINGDB y la guardamos en un grafo. A partir de este, estudiamos las **propiedades del grafo**, si todos los nodos estaban conectados con la función *is_connected()*, si era o no dirigido con la función *is.directed()*, el grado de centralidad que nos informaba del número de conexiones de cada gen con la función *degree()*, la centralidad de cercanía con la que obtuvimos la distancia promedio entre un nodo y todos los demás nodos mediante la aplicación de *closeness()*, y la conectividad que nos indicaba la fortaleza de la conexión del nodo aplicando *edge_density()*.

2.4 Detección de comunidades

En segundo lugar, llevamos a cabo la **identificación de comunidades** mediante distintos **algoritmos de clusterización**: método de Girvan-Newman, algoritmo de optimización voraz, propagación de etiquetas y el algoritmo de Louvain.

- 1 El método de **Girvan Newman** (función *cluster_edge_betweenness()*, en el fichero *biologiasistemas_igraph.R* en el repositorio GitHub del proyecto) detecta comunidades basándose en la centralidad de intermediación de los nodos, en otras palabras, va eliminando gradualmente las aristas más importantes para identificar las comunidades de la red [15].
- 2 El **algoritmo voraz** (función *cluster_fast_greedy()*, en el fichero *biologiasistemas_igraph.R* en el repositorio GitHub del proyecto), busca formar grupos de datos de manera iterativa tomando en cada paso la elección más beneficiosa para fusionar o dividir clusters con el objetivo de maximizar un criterio local [16].
- 3 La **propagación de etiquetas** (función *cluster_label_prop()*, en el fichero *biologiasistemas_igraph.R* en el repositorio GitHub del proyecto) es un enfoque basado en la difusión de información a través de la red que agrupa los nodos que están fuertemente conectados [17].
- 4 Por último el **algoritmo de Louvain** (función *cluster_louvain()*, en el fichero *biologiasistemas_igraph.R* en el repositorio GitHub del proyecto) busca organizar los nodos de una red en comunidades de manera que la modularidad global de la red sea máxima, o en otras palabras, cómo de bien se dividen los nodos de una red en grupos o comunidades distintas [18].

Para estudiar mejor a qué comunidad pertenece cada nodo, visualizamos el resultado de la aplicación del algoritmo de Louvain mediante la aplicación de **link communities** (función *getLinkCommunities()*, en el fichero *biologiasistemas_igraph.R* en el repositorio GitHub del proyecto) que nos permitía identificar si alguno de los nodos se incluían en varias comunidades.

2.5 Interacción de genes de interés

En tercer lugar, estudiamos la interacción de nuestros **genes de interés** especificados anteriormente: *TP53*, *HRAS*, *AKT1*, *SDHB*, *SDHD*. Para ellos creamos una función (función *encontrar_vecinos_interesantes()*, en el fichero *biologiasistemas_igraph.R* en el repositorio GitHub del proyecto) que nos devolvía una tabla con los vecinos de cada uno de estos genes para saber si entre ellos estaban relacionados.

2.6 Enriquecimiento funcional de la comunidad de interés.

Tras estudiar la detección de comunidades optamos por realizar un enriquecimiento funcional de la comunidad en la que se encuentran nuestros genes de interés. Para ello, se busca la comunidad en la que se encuentran estos genes, se identifica esta y se guardan los genes que la conforman en el entorno de R. Esto lo hemos hecho con el comando `nodos_genes_interes <- V(net)$name[in%genes_interes]` donde *net* es nuestro grafo principal y *genes_interes* es un vector donde se encuentran definidos los genes de interés del estudio. Este comando encuentra los índices de los nodos que representan los genes de interés nuestro de nuestro grafo. Seguimos, obteniendo el índice de la comunidad que contiene estos genes de interés con el comando `cluster_containing_genes <- membership(cfg)[nodos_genes_interes]` donde *cfg* es la variable donde se encuentran las diferentes comunidades que hemos identificado en nuestro grafo principal. Tras esto, con el comando `genes_cluster <- V(net)$name[membership(cfg)==cluster_containing_genes]` guardamos todos los genes que conforman esta comunidad en una variable. Esta variable la guardamos en un fichero de texto con la función `write.table()`.

Una vez que tenemos el fichero de texto con los genes que conforman la comunidad de interés procedemos a realizar un enriquecimiento funcional. Para ello, en un nuevo script de Python que hemos llamado *enriquecimiento_cluster.py*. Comenzamos importando las librerías necesarias y construyendo la URL para realizar una solicitud a la API de STRINGDB. Definimos la categoría funcional de interés, en nuestro caso HPO (para relacionar los genes con fenotipos patológicos), definimos la especie de nuestros genes, en este caso 9606 (referente a Homo Sapiens) y el archivo de genes donde se encuentran nuestra comunidad de interés. Seguidamente recogemos los parámetros para el análisis funcional ya definidos en una solicitud a la API de StringDB. Creamos un dataframe con las columnas *Term*, *PreferredName*, *Category*, *Description* que muestran el término HPO, el nombre de los genes, la categoría funcional (en este caso, HPO) y una descripción del fenotipo patológico asociado. Este dataset se guarda en un archivo csv con la función `.to_csv()`.

2.7 Consulta de la relación entre genes asociados y fenotipos patológicos.

Por último, hemos consultado manualmente el archivo csv creado con el dataframe anteriormente descrito y hemos identificado relaciones de genes asociados con fenotipos patológicos (estas consultas han sido filtradas por palabras clave como son: papiloma, genitales, renal, carcinoma, ovario y derivados, útero y derivados).

3 Resultados

3.1 Red de interacción entre genes

Se identificaron un total de 51 genes asociados al fenotipo **Papilloma** y obtuvimos una representación visual de la red de interacción entre los genes (Fig. 1). La red cuenta con todos los nodos con grado mayor o igual que 1 menos 6 nodos que se encuentran aislados. Más detalles sobre la red serán explicados a partir de la sección 3.3.

3.2 Análisis de enriquecimiento funcional

Se realizó un análisis de enriquecimiento funcional destacando las categorías **Process** y **KEGG**, obteniendo relaciones entre algunos grupos de genes con problemas del sistema inmune, carcinoma y las adipoquinas.

Los términos y categorías relacionados con enfermedades como **cervix**, **ovarian**, **HPV**, **herpes**, **papillomavirus**, **costellos** y **cowden** fueron encontrados en el análisis de enriquecimiento funcional, indicando posibles vínculos entre estos términos y los genes asociados al fenotipo estudiado.

Se realizaron búsquedas específicas de genes clave como *TP53*, *AKT1*, *SDHB*, *SDHD*, *HRAS* dentro de las categorías significativas. Estos genes podrían tener una importancia particular en relación con el fenotipo de interés, evidenciando su posible relevancia funcional.

3.3 Propiedades de la red y detección de comunidades

En nuestra red vimos que todos los nodos estaban conectados entre sí y que se trataba de un grafo no dirigido.

Obtuvimos una tabla con el **grado de centralidad** de cada gen (tabla 1). Pudimos observar como el gen de interés *TP53* es el que presentaba un mayor grado de centralidad y por tanto más relaciones con otros genes.

Gen	Grado Centralidad
CD4	15
CD79A	13
SPL1	11
DDB2	9
LRR8A	4
FLT4	5
TCF3	10
IL7	10
PIK3CA	11
MSH3	11
TP53	30
XPC	9
ERCC3	9
ERCC4	9
USF3	1
TMC8	5
IGLL1	7
CARMIL2	1
SEC23B	2
GATA2	10
SASH3	3

Gen	Grado Centralidad
SDHC	5
NRAS	8
PTEN	15
STK4	2
XPA	9
SDHB	7
SDHD	5
TPP2	2
DCLRE1C	10
RHOH	8
ERCC2	8
CD79B	11
CXCR4	12
HRAS	9
PIK3R1	12
AKT1	20
TMC6	4
IKBKG	4
DOCK8	7
ERCC5	9

Table 1: Grado Centralidad

En cuanto al **grado de conectividad** obtuvimos un valor del 19%, el cual era bastante pequeño y nos indicaba que nuestro grafo no era muy fuerte. Esto puede

deberse a que las comunidades entre si no tenían una gran dependencia y teníamos varios genes que no nos interesaban.

Empleamos distintos algoritmos para **detectar comunidades**. En el caso del método de Girvan-Newman había nodos que no pertenecían a ninguna comunidad (Fig. 2-A). Esto podía deberse a la forma en que el algoritmo de betweenness identifica comunidades. Puede detectar comunidades basándose en la centralidad de intermediación de los nodos, y algunos nodos pueden no estar claramente vinculados a una comunidad en función de esta medida.

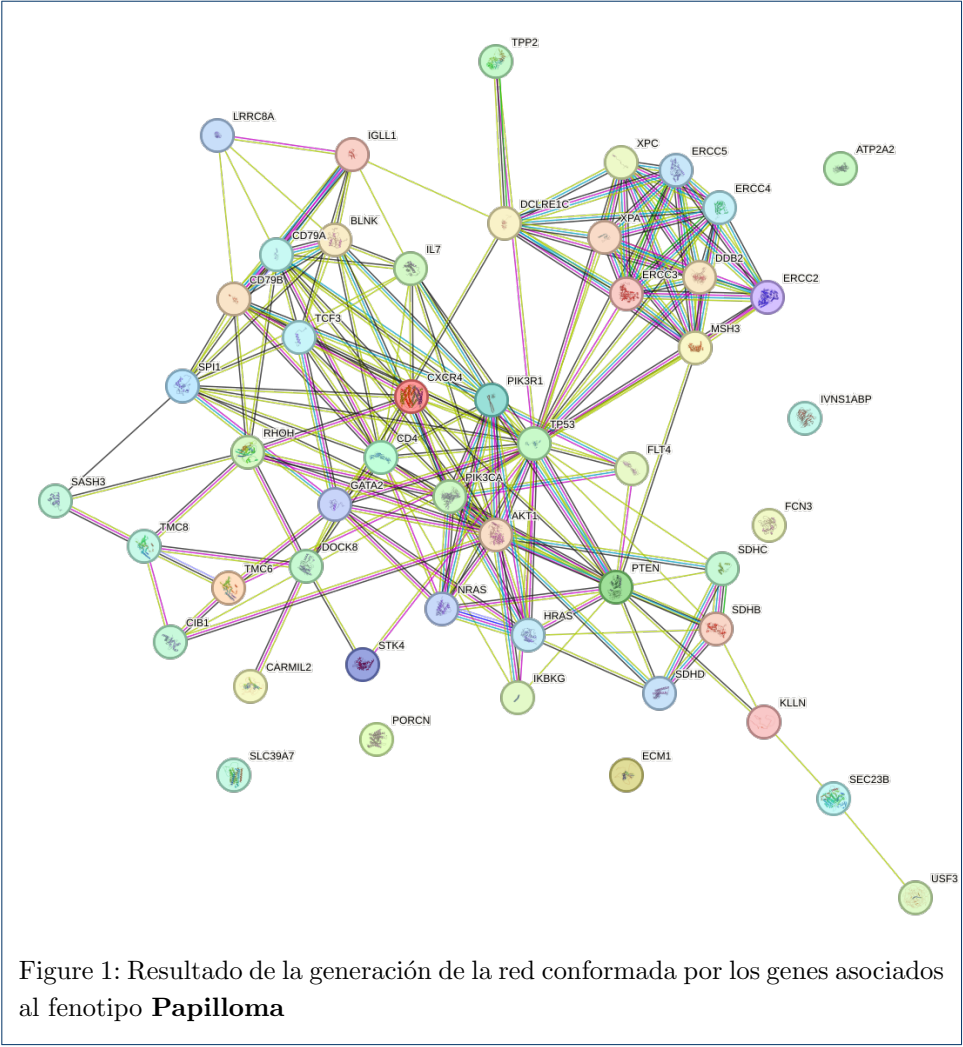
Con el algoritmo voraz vimos que nuestros genes de interés se encontraban en el mismo cluster de color amarillo (*TP53*, *AKT1*, *SDHB*, *SDHD*, *HRAS*). Además, todos los nodos pertenecían a alguna comunidad (Fig. 2-B), al igual que con el algoritmo de propagación de etiquetas (Fig. 2-C). Sin embargo, en este último obtuvimos tan solo 3 comunidades por lo que la clusterización es mínima. Con la aplicación del método de Louvain también resultaban menos comunidades que con el algoritmo voraz. Observamos 4 comunidades (Fig. 2-D).

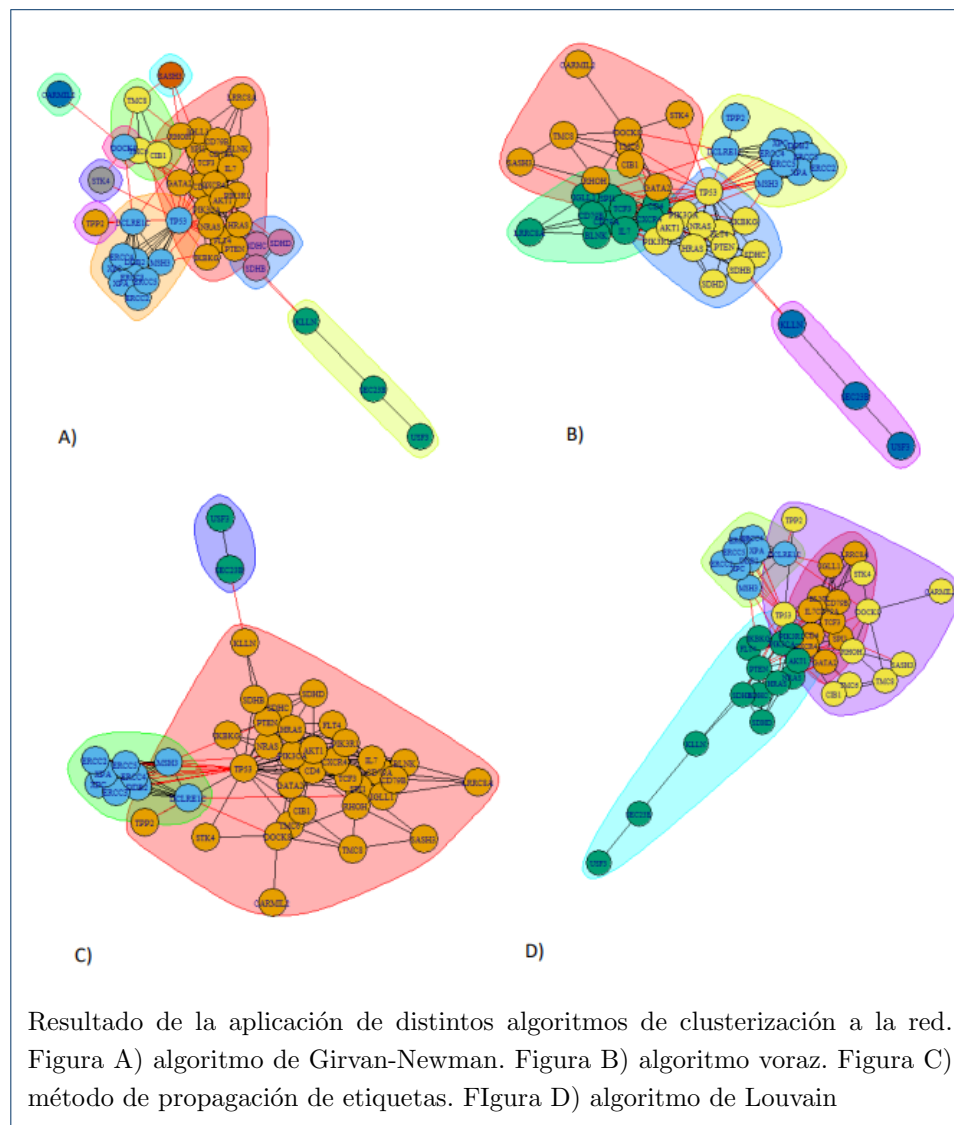
Los algoritmos anteriores tenían la desventaja de que no reflejaban con exactitud la posibilidad de que un gen perteneciera a más de una comunidad. Para estudiar esto hicimos uso de **Link Communities**, y obtuvimos una gráfica donde cada gen era representado por un diagrama de sectores (Fig. 3). Observamos que la mayoría de los genes pertenecen a más de una comunidad. Al centrarnos en nuestros genes de interés detectamos que el gen *TP53* tenía relación con 4 clusters distintos mientras que los otros genes de interés, *SDHB*, *SDHb*, *HRAS*, *AKT1* se encontraban en una comunidad más diferenciada de color rosa.

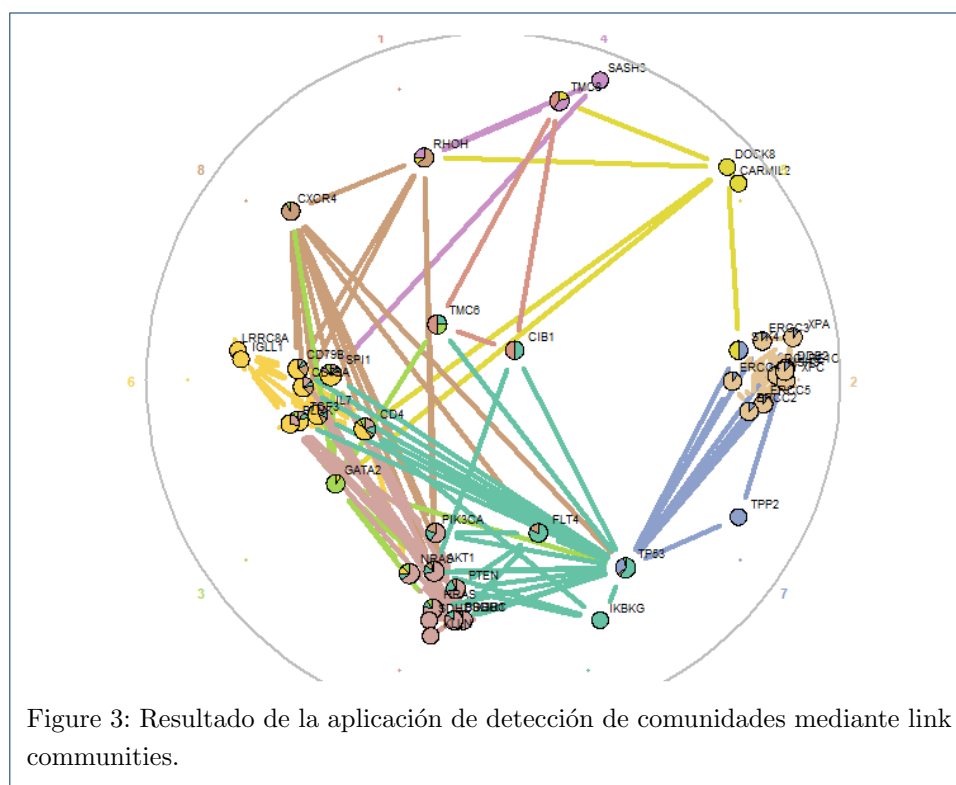
Para el **estudio genes de interés** obtuvimos la tabla 2 con el gen en concreto y sus genes vecinos. Observamos que todos los genes estaban relacionados entre sí menos el caso de *TP53* y *SDHD*.

Table 2: Genes de interes y genes relacionados

Genes	Genes interes
TP53	SDHB,HRAS,AKT1
HRAS	TP53,SDHB,SDHS,AKT1
AKT1	TP53,SDHB,SDHD,HRAS
SDHB	TP53,SDHD,HRAS,AKT1
SDHD	SDHB,HRAS,AKT1







3.4 Relación de los genes de interés con fenotipos patológicos.

Tras enriquecer el conjunto de genes de interés obtuvimos 729 entradas en las cuales encontramos relaciones de genes asociados con fenotipos patológicos que calificamos en estas tablas 3, 4, 5, 6, 7, 8, según la palabra clave por la cual buscamos:

Table 3: Fenotipos patológicos con palabra clave papiloma

Genes	Fenotipos patológicos asociados
FLT4, PIK3CA, TP53, SDHC, NRAS, PTEN, SDHB, SDHD, HRAS, PIK3R1, AKT1, IKBKG	Papiloma
TP53, NRAS	Papiloma del plexo coroideo

Table 4: Fenotipos patológicos con palabra clave genital

Genes	Fenotipos patológicos asociados
PIK3CA, TP53, SDHC, NRAS, PTEN, SDHB, SDHD, AKT1	Neoplasia genital
FLT4, PIK3CA, TP53, SDHC, NRAS, PTEN, SDHB, SDHD, HRAS, AKT1	Anomalía de los genitales externos masculinos
PIK3CA, TP53, SDHC, NRAS, PTEN, SDHB, SDHD, AKT1	Morfología anormal de los genitales internos femeninos

Table 5: Fenotipos patológicos con palabra clave renal

Genes	Fenotipos patológicos asociados
PIK3CA, TP53, SDHC, NRAS, PTEN, SDHB, SDHD, AKT1	Neoplasia renal
PIK3CA, SDHC, NRAS, PTEN, SDHB, SDHD, AKT1	Carcinoma de células renales
PIK3CA, TP53, SDHC, PTEN, SDHB, SDHD, AKT1	Anomalía de las glándulas suprarrenales
TP53, SDHC, PTEN, SDHB, SDHD	Neoplasia de la glándula suprarrenal
SDHC, SDHB, SDHD	Feocromocitoma extraadrenal
SDHC, SDHB, SDHD	Feocromocitoma suprarrenal

Table 6: Fenotipos patológicos con palabra clave carcinoma

Genes	Fenotipos patológicos asociados
PIK3CA, SDHC, NRAS, PTEN, SDHB, SDHD, HRAS, AKT1	Carcinoma folicular de tiroides
PIK3CA, TP53, SDHC, NRAS, PTEN, SDHB, SDHD, HRAS, AKT1	Carcinoma de tiroides
PIK3CA, TP53, SDHC, PTEN, SDHB, SDHD, AKT1	Carcinoma de mama
PIK3CA, SDHC, NRAS, PTEN, SDHB, SDHD, AKT1	Carcinoma de células renales
PIK3CA, SDHC, PTEN, SDHB, SDHD, AKT1	Carcinoma de endometrio
PIK3CA, NRAS, PTEN, HRAS, AKT1	Carcinoma de células transicionales de vejiga
PIK3CA, NRAS, AKT1	Carcinoma colorrectal hereditario no polipósico
NRAS, HRAS1	Carcinoma no medular de tiroides
PIK3CA, AKT1	Adenocarcinoma papilar de ovario
PIK3CA, TP53	Adenocarcinoma de pulmón
NRAS, HRAS	Carcinoma papilar de tiroides
NRAS, HRAS	Carcinoma basocelular
PIK3CA, TP53	Carcinoma hepatocelular
TP53, AKT1	Carcinoma

Table 7: Fenotipos patológicos con palabra clave ovario y derivados

Genes	Fenotipos patológicos asociados
PIK3CA, TP53, PTEN, AKT1	Neoplasia ovárica
PIK3CA, AKT1	Adenocarcinoma papilar de ovario
PIK3CA, TP53, SDHC, PTEN, SDHB, SDHD, AKT1	Anomalia del ovario
PIK3CA, SDHC, PTEN, SDHB, SDHD, AKT1	Ovarios poliquísticos agrandados

Table 8: Fenotipos patológicos con palabra clave útero y derivados

Genes	Fenotipos patológicos asociados
PIK3CA, SDHC, NRAS, PTEN, SDHB, SDHD, AKT1	Neoplasia uterina
PIK3CA, NRAS, AKT1	Leiomioma uterino
FLT4, SDHB, SDHD, PIK3R1	Retraso del crecimiento intrauterino

4 Discusión

5 Conclusiones

Abreviaciones

VPH: virus del papiloma humano HPO: human phenotype ontology

Disponibilidad de datos y materiales

<https://github.com/mariavidaa/HPO.Papilloma>

Contribución de los autores

E.M.A.F. hipótesis y materiales, R.A.C.V. introducción y, métodos y resultados en Python, M.J.H.R. introducción y, métodos y resultados en R y Python F.J.R.L. objetivos y, métodos y resultados en Python, M.V.M introducción y, métodos y resultados en R

Author details

ETSI Informática, Universidad de Málaga, Málaga, España.

References

- Kozomara, R.J., Brankovic-Magic, M.V., Jovic, N.R., Stosic, S.M., Magic, Z.M.: Prognostic significance of tp53 mutations in oral squamous cell carcinoma with human papilloma virus infection. *The International Journal of Biological Markers* **22**, 252–257 (2007). doi:10.1177/172460080702200403. PMID: 18161655
- Ljubojevic, S., Skerlev, M.: Hpv-associated diseases. *Clinics in Dermatology* **32**, 227–234 (2014). doi:10.1016/j.clindermatol.2013.08.007. Update on Sexually Transmitted Infections
- del Cáncer, I.N.: El virus del papiloma humano (VPH) y el cáncer - Instituto Nacional del Cáncer (2021)
- Siegel, D.H., Mann, J.A., Krol, A.L., Rauen, K.A.: Dermatological phenotype in costello syndrome: Consequences of ras dysregulation in development. *British Journal of Dermatology* **166** (2012). doi:10.1111/j.1365-2133.2011.10744.x
- McKenna, M., Balasuriya, N., Zhong, S., Li, S.S.C., O'Donoghue, P.: Phospho-form specific substrates of protein kinase b (akt1). *Frontiers in Bioengineering and Biotechnology* **8** (2021). doi:10.3389/fbioe.2020.619252
- Hensen, E.F., Jordanova, E.S., van Minderhout, I.J.H.M., Hogendoorn, P.C.W., Taschner, P.E.M., van der Mey, A.G.L., Devilee, P., Cornelisse, C.J.: Somatic loss of maternal chromosome 11 causes parent-of-origin-dependent inheritance in sdhd-linked paraganglioma and pheochromocytoma families. *Oncogene* **23**, 4076–4083 (2004). doi:10.1038/sj.onc.1207591
- Mahdi, H., Mester, J.L., Nizialek, E.A., Ngeow, J., Michener, C., Eng, C.: Germline pten, sdhb-d, and klln alterations in endometrial cancer patients with cowden and cowden-like syndromes: An international, multicenter, prospective study. *Cancer* **121** (2015). doi:10.1002/cncr.29106
- Köhler, S., Gargano, M., Matentzoglou, N., Carmody, L.C., Lewis-Smith, D., Vasilevsky, N.A., Danis, D., Balagura, G., Baynam, G., Brower, A.M., Callahan, T.J., Chute, C.G., Est, J.L., Galer, P.D., Ganesan, S., Griese, M., Haimel, M., Pazmandi, J., Hanauer, M., Harris, N.L., Hartnett, M.J., Hastreiter, M., Hauck, F., He, Y., Jeske, T., Kearney, H., Kindle, G., Klein, C., Knoflach, K., Krause, R., Lagorce, D., McMurry, J.A., Miller, J.A., Munoz-Torres, M.C., Peters, R.L., Rapp, C.K., Rath, A.M., Rind, S.A., Rosenberg, A.Z., Segal, M.M., Seidel, M.G., Smedley, D., Talmy, T., Thomas, Y., Wiafe, S.A., Xian, J., Yüksel, Z., Helbig, I., Mungall, C.J., Haendel, M.A., Robinson, P.N.: The human phenotype ontology in 2021. *Nucleic Acids Research* **49** (2021). doi:10.1093/nar/gkaa1043
- Giorgi, F.M., Ceraolo, C., Mercatelli, D.: The R Language: An Engine for Bioinformatics and Data Science (2022). doi:10.3390/life12050648
- Mehare, H.B., Anilkumar, J.P., Usmani, N.A.: The Python Programming Language, (2023). doi:10.1007/978-3-031-22206-1_2
- Szklarczyk, D., Gable, A.L., Nastou, K.C., Lyon, D., Kirsch, R., Pyysalo, S., Doncheva, N.T., Legeay, M., Fang, T., Bork, P., Jensen, L.J., von Mering, C.: The string database in 2021: Customizable protein-protein networks, and functional characterization of user-uploaded gene/measurement sets. *Nucleic Acids Research* **49** (2021). doi:10.1093/nar/gkaa1074
- Csardi, G., Nepusz, T.: The igraph software package for complex network research. *InterJournal Complex Systems* **Complex Sy** (2006)
- Sneekun, R., Vachiyatwala, K., Author, C.: Data analysis using pandas library of python. *Acta Scientific COMPUTER SCIENCES* **4** (2022)
- Kalinka, A.T.: The generation , visualization , and analysis of link communities in arbitrary networks with the r package linkcomm. *Molecular Cell* (2011)
- Zahiri, M., Mohammadzadeh, J., Harifi, S.: An improved girvan–newman community detection algorithm using trust-based centrality. *Journal of Ambient Intelligence and Humanized Computing* **14** (2023). doi:10.1007/s12652-021-03508-y
- Curtis, S.A.: The classification of greedy algorithms. *Science of Computer Programming* **49** (2003). doi:10.1016/j.scico.2003.09.001
- Garza, S.E., Schaeffer, S.E.: Community detection with the Label Propagation Algorithm: A survey (2019). doi:10.1016/j.physa.2019.122058
- Zhang, J., Fei, J., Song, X., Feng, J.: An improved louvain algorithm for community detection. *Mathematical Problems in Engineering* **2021** (2021). doi:10.1155/2021/1485592