

RESEARCH

Red de los targets de SARS-CoV2

Irene Romero Granados*, Paula Andújar Zambrano*, Rosario García Morales* and Soledad del Castillo Carrera*

*Correspondence:
ireero99@uma.es;
paulandujar@uma.es;
0619884107@uma.es;
delcastillosoledad@uma.es
ETSI Informática, Universidad de
Málaga, Málaga, España
Full list of author information is
available at the end of the article

Abstract

Este proyecto pretende estudiar las interacciones que se producen entre las 29 proteínas del virus SARS-COV-2 y el interactoma humano. Para ello, se recopilarán los datos de interacción comentados, con los que se procederá a realizar un análisis sobre ellos y construir la red de proteínas que conforman los objetivos principales para el virus. En cuanto a las herramientas que se utilizarán, serán la base de datos biológica UniProt para la obtención de los datos de entrada, y el lenguaje de R para los métodos de análisis.

Keywords: SARS-COV-2; interactoma; R

1 Introducción

La familia de los coronavirus son virus infecciosos a los que se llama así debido a que en su superficie tienen puntas en forma de corona. A esta familia se les unió en 2019 el conocido SARS-CoV-2, que ha dado lugar al coronavirus 2 o COVID-19. Esta enfermedad es una enfermedad infecciosa que afecta a las vías respiratorias, de manera leve a moderada. Sin embargo esta enfermedad en personas mayores o con patologías previas puede hacer que se desarrolle la enfermedad con consecuencias o síntomas más graves, pudiendo producir hasta la muerte.

El coronavirus actualmente es considerado un problema de salud global, ya que debido a esta pandemia se han contagiado hasta ahora unas 369.955.862 personas y han fallecido un total de 5.650.738 personas.

Es por esto que es esencial el estudio de este virus, tanto de sus genes, sus proteínas o como interacciona con el ser humano.

A día de hoy tras toda la inversión mundial que se ha hecho para poder poner fin a este virus, se sabe que el SARS-CoV-2 está formado por 29 proteínas que interactúan con las células del ser humano pudiendo producir síntomas respiratorios graves hasta poder causar la muerte. A estas interacciones moleculares binarias proteína-proteína se les llama interactoma.

El interactoma sirve como de punto de partida para estudiar los posibles fármacos que podrían bloquear dichas interacciones y así evitar que el virus entre a la célula y se replique. Gracias al estudio del interactoma ha sido posible la realización de vacunas contra el COVID-19.

En este proyecto vamos a crear y estudiar la red de interacciones de las proteínas del SARS-CoV-2 con las proteínas humanas, y así poder obtener cuales son las principales funciones biológicas humanas en las que este virus interviene y relacionarlo con la realidad. Todos los recursos usados para la obtención de dicha información la podremos encontrar en el GitHub proporcionado.

2 Materiales y métodos

2.1 Carga de librerías y datos

Para poder llevar a cabo este trabajo, el primer paso a realizar es la carga de librerías necesarias y la carga de datos. Antes de cargar los datos, estos han sido descargados de Uniprot (<https://www.uniprot.org/>) en formato .csv para poder llevar a cabo el análisis de la red. Una vez ha sido añadido este fichero al directorio correspondiente (data), se procederá a la carga de librerías. Las librerías que han sido utilizadas en este proyecto son las siguientes:

- **igraph**: Esta librería permite realizar análisis de redes, por lo cual, proporciona funciones para manipular gráficos con facilidad.
- **dplyr**: Esta librería proporciona métodos para poder manejar los ficheros de datos.
- **ggplot2**: Esta librería es un paquete de visualización de datos.
- **zoo**: Esta librería está especialmente dirigida a series temporales irregulares de vectores/matrices y factores numéricos.
- **STRINGdb**: Este paquete proporciona una interfaz para la base de datos STRING de interacciones proteína-proteína.

Después de tener las librerías necesarias y saber la funcionalidad de cada una de ellas, se procederá a cargar el archivo en una variable llamada "data" mediante el método "read.csv()"

Seguidamente, se filtrarán las entradas utilizando el paquete "dplyr" en las que la columna Entry.Name contenga en su nombre "HUMAN" ya que estos son los datos que interesan en esta práctica.

2.2 Mapeo y primera capa de la red

A continuación, utilizando la librería STRINGdb, se realiza un mapeo con los datos ya filtrados. Se guardará los hits de string en una imagen png en el directorio de los resultados (results)

Seguidamente, se creará la primera capa de la red y se guardará el resultado de esta primera capa en una imagen png.

2.3 Linked Communities

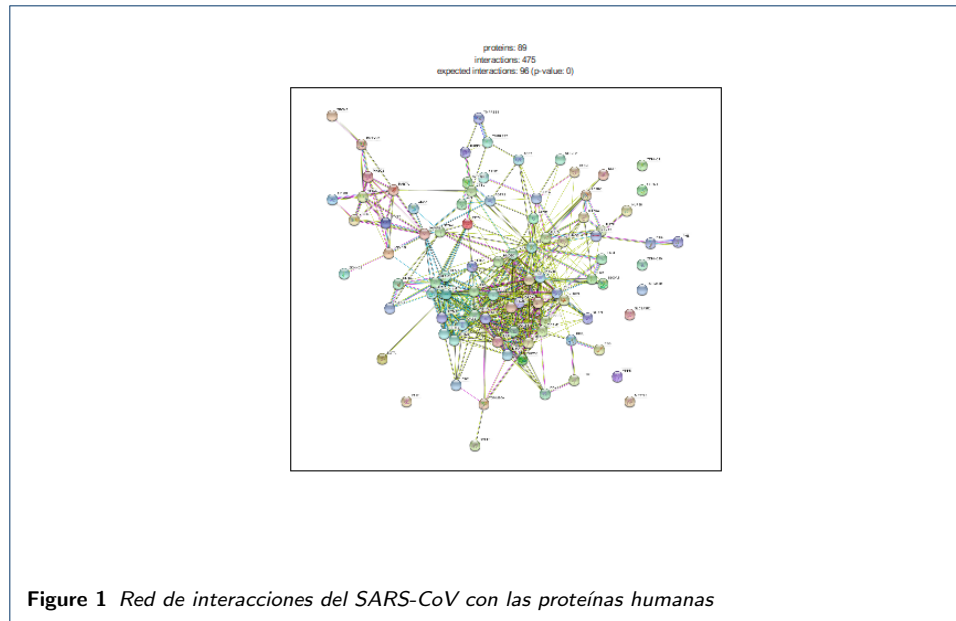
Una vez se tiene los datos ya mapeados y filtrados, se pasa a realizar el clustering. El clustering consiste en agrupar los elementos o nodos en grupos con características similares y así, determinar patrones.

Utilizando el paquete linkcomm, se ha realizado la búsqueda de comunidades. Linkcomm proporciona las herramientas necesarias para generar, visualizar y analizar comunidades dentro de un grafo.

Al obtener las comunidades vinculadas, estas serán guardadas en la carpeta de results y, además, se han obtenido los tamaños de los clusters y han sido clasificados por comunidad/modularidad para así obtener los 6 mejores.

2.4 Enriquecimiento Funcional

Al realizar el clustering y el agrupamiento de estas comunidades, se realizará el enriquecimiento funcional. Este enriquecimiento es utilizado para obtener los procesos biológicos de los grupos que han sido formados en el clustering. En primer lugar,



ha sido diseñado el enriquecimiento mediante STRINGdb. Después, se ha realizado el enriquecimiento con GO, es decir, con una ontología génica y también, se ha realizado el enriquecimiento con la ontología KEGG.

El enriquecimiento ha sido utilizado con los clusters de mayor tamaño y de mayor modularidad.

2.5 Robustez

Por último, utilizando los métodos proporcionados en el campus virtual, se ha calculado la robustez de la red de genes. Esta funciona para conocer si la red que se está estudiando es un sistema fuerte y esta no tiende a tener errores. Además, ha sido calculada frente a ataques aleatorios como a ataques dirigidos, pero también, han sido combinados ambos ataques.

3 Resultados

3.1 Red de interacciones y robustez

La siguiente imagen muestra la red de interacciones del ser humano con las proteínas del SARS-CoV. Como podemos ver el SARS-CoV interacciona con 89 proteínas humanas, produciendo un total de 475 interacciones.

Tras eliminar los nodos que no están conectados, hemos obtenido la red real de interacciones que podemos ver a continuación. Sin embargo hay demasiadas conexiones como para poder distinguir los nodos. Es por ello que realizaremos los pasos siguientes de clustering, para así poder extraer la información relevante de la red.

Para poder estudiar cual es la capacidad de nuestra red de mantener sus funciones frente a la presencia de "ataques" y ver cuán de adaptable es, usamos la robustez. Podemos observar que para ataques aleatorios es bastante robusta, mientras que para ataques dirigidos es más débil.

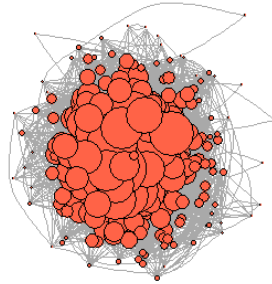


Figure 2 Red de interacciones del SARS-CoV con las proteínas humanas tras un proceso de filtrado

3.2 Linked Communities

En esta sección se explicarán los resultados obtenidos aplicar los métodos de comunidades enlazadas a nuestra red, para los cuales hemos usado el paquete linkcomm.

Para ello, primero hemos obtenido estas comunidades aplicando un 'single clustering'. Se ha guardado una imagen del resumen de estas comunidades para un resultado más visual.

Podemos observar que este método ha definido 165 comunidades diferentes, la mayor de estas tiene 70 nodos lo cual podemos deducir que hay varias comunidades que comparten nodos.

Para poder analizar un poco las comunidades las hemos filtrado por tamaño y modularidad. Las gráficas generadas para este propósito las mostramos a continuación.

De estas gráficas hemos podido extraer la comunidad 112 siendo esta la más grande y las comunidades 78 y 22 con una mayor modularidad. Estas comunidades serán utilizadas posteriormente para un análisis funcional.

Para una mejor visualización de los resultados, se ha cambiado el diseño de la gráfica al de Fruchterman Reingold, mostrando solo nodos que pertenezcan a 10 o más comunidades.

Por otra parte, se han obtenido las comunidades anidadas y se han filtrado para que se muestren las que son independientes de las demás.

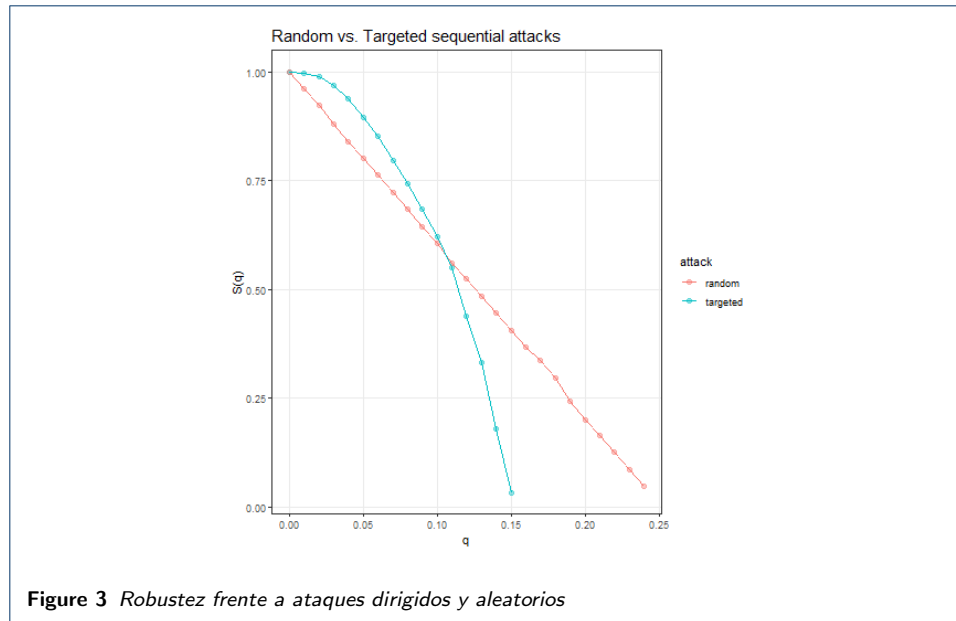
4 Discusión

5 Conclusiones

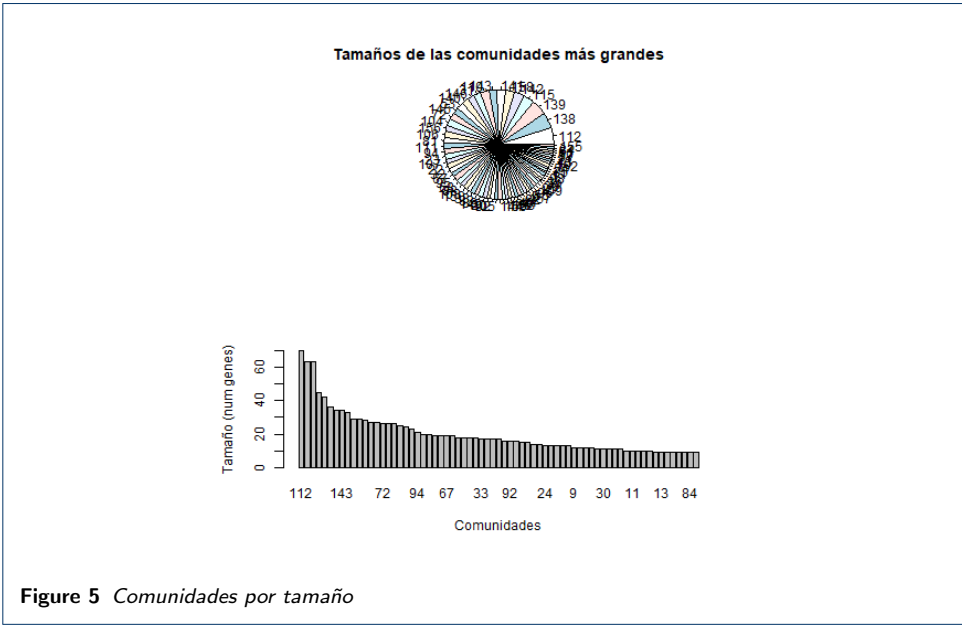
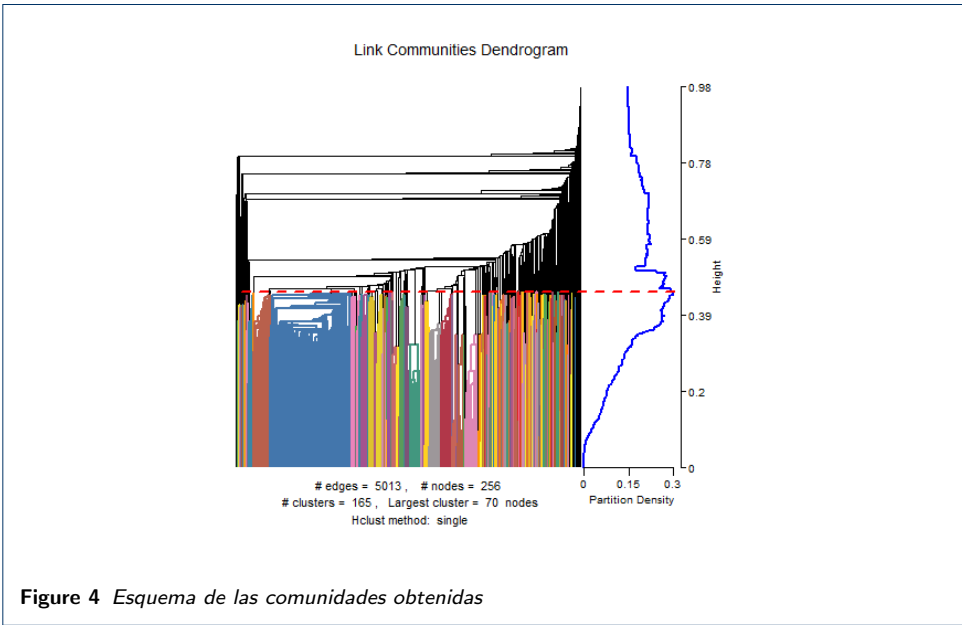
Abreviaciones

Disponibilidad de datos y materiales

https://github.com/Paulandujar/project_template

**Contribución de los autores**

Usando las iniciales que habéis definido al comienzo del documento, debéis 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.



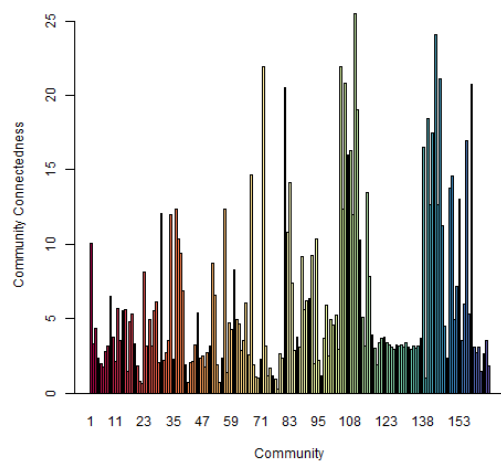


Figure 6 *Comunidades por modularidad*

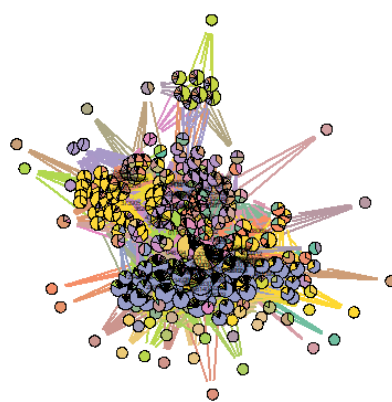


Figure 7 *Fruchterman Reingold graph*

