

<b>Asignatura:</b>	<b>Análisis de datos masivos</b>	<b>Docente:</b>	Ing. Juan Carlos García
--------------------	----------------------------------	-----------------	-------------------------

**Tarea****Proyecto Final****Nombre de estudiante:**

Darlyn Moran

Sebastián Bermeo

José Paye

Steven Lavayen

Xavier Cruz

**Curso:**

SIN-S-MA-10-6

**Ciclo:**

Ciclo I 2024 - 2025

## RESUMEN

En el análisis de coautoría de publicaciones científicas utilizando datos de un archivo CSV (`scopus.csv`), se construyeron dos tipos de grafos para visualizar las relaciones entre los autores. Primero, se generó un grafo no dirigido utilizando NetworkX, donde los nodos representan a los autores y las aristas reflejan las colaboraciones entre ellos. En este grafo, cada par de autores que colaboró en una misma publicación se conecta mediante una arista cuyo peso indica el número de publicaciones compartidas. La visualización con Plotly muestra una red en la que los autores frecuentemente colaborativos están más interconectados, proporcionando una visión general de la estructura de coautoría en la red científica.

Posteriormente, se construyó un grafo dirigido para explorar la dirección de las colaboraciones. En este modelo, las aristas dirigidas entre nodos no solo indican la existencia de una colaboración, sino también la dirección de esta, lo que podría reflejar la secuencia de la colaboración o el flujo de trabajo entre autores. Las flechas en la visualización, creadas también con Plotly, destacan estas direcciones, ofreciendo una perspectiva más matizada sobre cómo se desarrollan las colaboraciones en el campo científico. Ambos enfoques ofrecen valiosas perspectivas sobre las redes de coautoría, siendo el grafo dirigido útil para analizar la dirección de las colaboraciones, mientras que el grafo no dirigido proporciona una visión general de la conectividad entre autores.

## INTRODUCCIÓN

El análisis de redes de coautoría en la literatura científica es fundamental para entender las dinámicas de colaboración en el campo de la investigación. En este contexto, el archivo `scopus.csv` proporciona una base rica y exhaustiva de datos, recopilando información crítica sobre publicaciones científicas. Este archivo contiene 46 columnas que abarcan una amplia gama de detalles sobre cada publicación, ofreciendo una visión integral del panorama científico. Entre estas columnas se encuentran 'Authors' (autores), que identifica a los colaboradores de cada trabajo; 'Title' (título), que describe el tema de la publicación; 'Year' (año de publicación), que señala el momento de la investigación; y 'Cited by' (citas), que indica el impacto de la

publicación en el campo. Además, se incluyen 'DOI' (identificador de objeto digital), 'Author Keywords' (palabras clave de los autores), y 'Affiliations' (afiliaciones), entre otros. Esta riqueza de información permite una construcción detallada y precisa de grafos que representan las redes de coautoría.

La construcción de estos grafos se lleva a cabo mediante herramientas avanzadas como NetworkX y Plotly, que facilitan la visualización y análisis de las conexiones entre autores. En particular, se desarrollan tanto grafos dirigidos como no dirigidos para capturar diferentes aspectos de las colaboraciones científicas. El grafo no dirigido revela la intensidad de las conexiones entre autores, mientras que el grafo dirigido proporciona información sobre la dirección de estas colaboraciones, lo que puede ofrecer una perspectiva adicional sobre el flujo de trabajo y las interacciones entre los investigadores. La creación de estos grafos a partir del dataset `scopus.csv` no solo permite una visualización clara de las redes de coautoría, sino que también proporciona herramientas analíticas para identificar patrones significativos en la colaboración científica. Estos análisis son esenciales para comprender mejor las estructuras de las redes de investigación y para identificar clusters de colaboración intensa y los roles específicos de los autores dentro de estas redes.

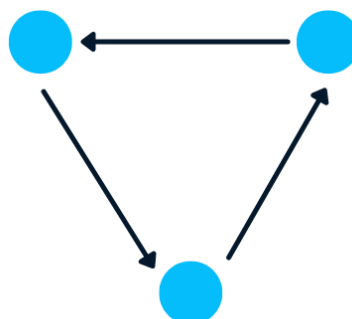
Uno de los gráficos presentados es un grafo dirigido que ilustra las relaciones de coautoría entre los autores en el dataset `scopus.csv`. Este grafo proporciona una representación visual de cómo los autores colaboran entre sí en el ámbito de la investigación científica. En este grafo, cada nodo representa a un autor individual, etiquetado con su nombre y coloreado de azul claro para una identificación clara. Las aristas, representadas por líneas grises, conectan estos nodos, indicando que los autores han trabajado conjuntamente en al menos una publicación. La particularidad de este grafo es el uso de flechas rojas, que señalan la dirección de la colaboración. Aunque en el contexto de coautoría la dirección de las flechas puede no tener un significado específico a menos que se defina claramente, estas flechas ayudan a resaltar el carácter dirigido del grafo. El diseño de los nodos y aristas utiliza un layout de resorte (`spring_layout`), que organiza los nodos de manera que las aristas tengan una longitud relativamente igual y minimiza las intersecciones, facilitando una visualización ordenada. Además, el gráfico está configurado para no mostrar cuadrículas ni líneas de cero en los ejes, enfocándose

completamente en la estructura del grafo para una interpretación más clara de las relaciones de colaboración.

El gráfico presentado es un grafo no dirigido que representa las relaciones de coautoría entre los autores del dataset scopus.csv. Este grafo ofrece una visión detallada de las colaboraciones entre autores en el ámbito científico. En el grafo, cada nodo simboliza a un autor, etiquetado con su nombre y coloreado de azul claro para asegurar una distinción visual fácil. Las aristas, que son líneas grises, conectan estos nodos, demostrando que los pares de autores han colaborado en al menos una publicación. Aunque el grosor de las líneas es uniforme en este gráfico, en una visualización más detallada, el grosor de las aristas podría reflejar la cantidad de colaboraciones entre los autores. El layout utilizado, el diseño de resorte (spring\_layout), distribuye los nodos de manera que las aristas mantengan una longitud más o menos igual y minimiza las intersecciones, contribuyendo a una presentación clara y ordenada. El gráfico está diseñado para no mostrar cuadrículas ni líneas de cero en los ejes, lo que permite una visualización limpia y sin distracciones, enfocándose en la estructura general del grafo y en las relaciones de colaboración entre los autores.

## MARCO TEÓRICO

Primero que nada, un grafo es un tipo de estructura de datos que contiene nodos y aristas. Un nodo puede ser una persona, un lugar o una cosa, y las aristas definen la relación entre nodos. Las aristas pueden ser dirigidas y no dirigidas en función de las dependencias direccionales. En el ejemplo siguiente, los círculos azules son nodos y las flechas son aristas. La dirección de las aristas define las dependencias entre dos nodos.



*Ilustración 1: Imagen de un grafo*

El análisis de redes de coautoría en la literatura científica se ha convertido en una herramienta fundamental para comprender la estructura y dinámica de la investigación

académica. Esta metodología utiliza la teoría de grafos para representar y analizar las relaciones entre autores a través de sus colaboraciones en publicaciones científicas. En este contexto, un grafo se define como una estructura compuesta por nodos (representando autores) y aristas (representando colaboraciones entre ellos). La teoría de grafos proporciona el marco matemático necesario para entender cómo estas conexiones estructuran la red de investigación.

Una de las contribuciones más significativas a la teoría de redes fue realizada por Barabási y Albert (1999) con el modelo de redes de escala libre. Este modelo explica cómo las redes complejas, incluidas las redes de coautoría, tienden a formar hubs o nodos altamente conectados que actúan como centros de influencia. En el ámbito de la coautoría científica, estos hubs a menudo representan investigadores clave que tienen una amplia red de colaboradores y desempeñan un papel crucial en la propagación de nuevos conocimientos y en la formación de grupos de investigación interdisciplinarios (Barabási & Albert).

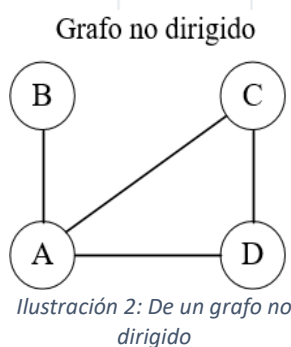
Por otro lado, el trabajo de Newman (2001) sobre redes de colaboración científica resalta la importancia de identificar comunidades de autores que colaboran en áreas temáticas similares. Este enfoque permite descubrir patrones de agrupamiento y colaboración intensiva dentro de la red, proporcionando información valiosa sobre la estructura de las comunidades científicas y su evolución a lo largo del tiempo. Los métodos de análisis de redes comunitarias, como el algoritmo de modularidad de Newman y Girvan (2004), son ampliamente utilizados para detectar estas comunidades y evaluar la cohesión dentro de la red (Newman. J ; Girvan. M., 2001 ; 2004).

En el campo del aprendizaje automático y el análisis de redes, las redes neuronales han introducido métodos avanzados para el análisis de grafos. Los embeddings de nodos, como Node2Vec y DeepWalk, transforman nodos en vectores de características en un espacio de menor dimensión, lo que permite aplicar modelos de aprendizaje profundo a los datos de red. Estas técnicas facilitan tareas como la clasificación de nodos y la predicción de enlaces, mejorando la capacidad para detectar patrones complejos y realizar inferencias precisas sobre la estructura de la red (Grover, A., & Leskovec, J., 2016) ; (Perozzi, B., Al-Rfou, R., & Skiena, S., 2014)

Las redes neuronales convolucionales en grafos (GCN) y las redes neuronales de atención en grafos (GAT) representan avances significativos en el análisis de redes complejas (Abid Ali Awan, 2024). Las GCN extienden las redes neuronales convolucionales tradicionales a datos de grafos, permitiendo la propagación de información a través de la estructura del grafo y el aprendizaje de representaciones de nodos basadas en la información de sus vecinos (Kipf, T. N., & Welling, M., 2017). Las GAT, por su parte, introducen mecanismos de atención que ponderan la importancia de las conexiones entre nodos, mejorando la capacidad del modelo para capturar relaciones complejas y realizar tareas de clasificación y análisis de importancia de las conexiones (Velickovic, 2018).

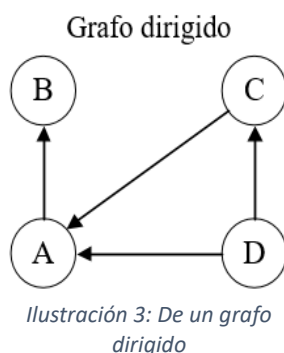
## MÉTODOS

### Creación y Análisis de Grafos:



**Grafo No Dirigido:** La construcción de un grafo no dirigido para representar las colaboraciones entre autores implica crear nodos para cada autor y aristas para cada par de autores que han colaborado en al menos una publicación. En este grafo, la relación de coautoría se considera simétrica, ya que la colaboración entre dos autores no tiene una dirección específica.

La visualización del grafo se realiza utilizando un layout de resorte (spring\_layout), que organiza los nodos de manera que las aristas tienen una longitud más o menos uniforme y se minimizan las intersecciones. Este enfoque permite una representación clara de la estructura general de la red de coautoría, facilitando la identificación de patrones de colaboración y comunidades.



**Grafo Dirigido:** Para capturar la dirección de las colaboraciones, se construye un grafo dirigido en el que las aristas tienen una orientación que puede reflejar la secuencia o jerarquía en las colaboraciones. Las flechas en el grafo dirigido indican la dirección de la colaboración, lo que puede ser útil para analizar patrones de flujo y dependencia entre autores. Aunque la dirección de las aristas puede no tener un significado

intrínseco sin una definición clara, su inclusión permite un análisis más detallado de las relaciones entre autores y la evolución de las colaboraciones.

### Análisis del grafo dirigido

Este un **grafo dirigido** que muestra las relaciones de coautoría entre autores con dataset de scopus.csv. Aquí se detalla un resumen de lo que presenta:

1. **Nodos:** Cada nodo representa a un autor. Los nodos están etiquetados con los nombres de los autores y están coloreados de azul claro.
2. **Aristas:** Las líneas grises conectan los nodos y representan las coautorías entre pares de autores. Cada línea indica que los dos autores colaboraron en al menos un trabajo juntos.
3. **Flechas:** Las flechas rojas indican la dirección de la colaboración. Aunque en este caso específico, las flechas se usan para resaltar la dirección en un grafo dirigido, pero en términos prácticos de coautoría, la dirección puede no tener un significado específico a menos que se defina.
4. **Layout:** El diseño de los nodos y aristas se organiza usando un layout de resorte (spring\_layout), que distribuye los nodos de manera que las aristas tengan una longitud más o menos igual y se minimicen las intersecciones.
5. **Configuración:** El gráfico no muestra cuadrículas ni líneas de cero en los ejes para enfocar la visualización en la estructura del grafo.

Este gráfico permite visualizar las conexiones entre autores y cómo se distribuyen las colaboraciones en el conjunto de datos.

```
import pandas as pd
import networkx as nx
import plotly.graph_objects as go

# Leer el archivo CSV
df = pd.read_csv('/content/drive/MyDrive/scopus.csv')

# Crear un grafo dirigido vacío
G = nx.DiGraph()

# Iterar sobre cada fila del DataFrame
for index, row in df.iterrows():
    authors = row['Authors']

    if pd.notna(authors):
```



```
# Dividir los autores por comas y eliminar espacios
adicionales
authors_list = [author.strip() for author in
authors.split(',')]

# Agregar nodos y aristas dirigidas para cada par de
autores
for i in range(len(authors_list)):
    for j in range(i + 1, len(authors_list)):
        author1 = authors_list[i]
        author2 = authors_list[j]

        # Añadir nodos (autores) y una arista dirigida
entre ellos
        if G.has_edge(author1, author2):
            G[author1][author2]['weight'] += 1
        else:
            G.add_edge(author1, author2, weight=1)

# Posicionamiento de los nodos
pos = nx.spring_layout(G, k=0.3)

# Extraer posiciones de nodos
nodes_x = [pos[node][0] for node in G.nodes()]
nodes_y = [pos[node][1] for node in G.nodes()]

# Extraer aristas
edges_x = []
edges_y = []
for edge in G.edges():
    x0, y0 = pos[edge[0]]
    x1, y1 = pos[edge[1]]
    edges_x.extend([x0, x1, None])
    edges_y.extend([y0, y1, None])

# Extraer aristas dirigidas
arrows_x = []
arrows_y = []
for edge in G.edges():
    x0, y0 = pos[edge[0]]
    x1, y1 = pos[edge[1]]
    arrows_x.append(x1)
    arrows_y.append(y1)

# Crear la figura con Plotly
fig = go.Figure()

# Añadir nodos
```



```
fig.add_trace(go.Scatter(x=nodes_x, y=nodes_y,
                        mode='markers+text',
                        text=[node for node in G.nodes()],
                        textposition='top center',
                        marker=dict(size=10, color='lightblue'),
                        name='Nodos'))

# Añadir aristas
fig.add_trace(go.Scatter(x=edges_x, y=edges_y,
                        mode='lines',
                        line=dict(width=1, color='gray'),
                        name='Aristas'))

# Añadir flechas para aristas dirigidas
fig.add_trace(go.Scatter(x=arrows_x, y=arrows_y,
                        mode='markers',
                        marker=dict(size=5, symbol='arrow-bar-
up', color='red'),
                        name='Flechas'))

# Configuración del layout
fig.update_layout(title='Grafo Dirigido de Coautores',
                  showlegend=False,
                  xaxis=dict(showgrid=False, zeroline=False),
                  yaxis=dict(showgrid=False, zeroline=False))

# Mostrar la figura
fig.show()
```

Ilustración 4: Código de grafo dirigido

### Análisis del grafo no dirigido

Este es un **grafo no dirigido** que muestra las relaciones de coautoría entre autores en la dataset de scopus.csv. Aquí se detalla lo siguiente:

1. **Nodos:** Cada nodo representa a un autor. Los nodos están etiquetados con los nombres de los autores y están coloreados de azul claro.
2. **Aristas:** Las líneas grises conectan los nodos y representan las colaboraciones entre pares de autores. Cada línea indica que los dos autores han colaborado en al menos un trabajo juntos. La grosor de la línea puede reflejar la cantidad de trabajos en los que los autores han colaborado, aunque en este gráfico las líneas tienen un ancho uniforme.

3. **Layout:** El diseño de los nodos y aristas se organiza usando un layout de resorte (spring\_layout), que distribuye los nodos de manera que las aristas tengan una longitud más o menos igual y se minimicen las intersecciones.
4. **Configuración:** El gráfico está configurado para no mostrar cuadrículas ni líneas de cero en los ejes, lo que facilita la visualización clara de la estructura del grafo.

Este gráfico permite visualizar las conexiones entre autores y cómo se distribuyen las colaboraciones en el conjunto de datos.

```
import pandas as pd
import networkx as nx
import plotly.graph_objects as go

# Leer el archivo CSV
df = pd.read_csv('/content/drive/MyDrive/scopus.csv')

# Crear un grafo no dirigido vacío
G = nx.Graph()

# Iterar sobre cada fila del DataFrame
for index, row in df.iterrows():
    authors = row['Authors']

    if pd.isna(authors):
        # Dividir los autores por comas y eliminar espacios
        # adicionales
        authors_list = [author.strip() for author in
authors.split(',')]

        # Agregar nodos y aristas para cada par de autores
        for i in range(len(authors_list)):
            for j in range(i + 1, len(authors_list)):
                author1 = authors_list[i]
                author2 = authors_list[j]

                # Añadir nodos (autores) y una arista entre
ellos

                if G.has_edge(author1, author2):
                    G[author1][author2]['weight'] += 1
                else:
                    G.add_edge(author1, author2, weight=1)

# Posicionamiento de los nodos
```

```
pos = nx.spring_layout(G, k=0.3)

# Extraer posiciones de nodos
nodes_x = [pos[node][0] for node in G.nodes()]
nodes_y = [pos[node][1] for node in G.nodes()]

# Extraer aristas
edges_x = []
edges_y = []
for edge in G.edges():
    x0, y0 = pos[edge[0]]
    x1, y1 = pos[edge[1]]
    edges_x.extend([x0, x1, None])
    edges_y.extend([y0, y1, None])

# Crear la figura con Plotly
fig = go.Figure()

# Añadir nodos
fig.add_trace(go.Scatter(x=nodes_x, y=nodes_y,
                        mode='markers+text',
                        text=[node for node in G.nodes()],
                        textposition='top center',
                        marker=dict(size=10, color='lightblue'),
                        name='Nodos'))

# Añadir aristas
fig.add_trace(go.Scatter(x=edges_x, y=edges_y,
                        mode='lines',
                        line=dict(width=1, color='gray'),
                        name='Aristas'))

# Configuración del layout
fig.update_layout(title='Grafo de Coautores',
                  showlegend=False,
                  xaxis=dict(showgrid=False, zeroline=False),
                  yaxis=dict(showgrid=False, zeroline=False))

# Mostrar la figura
fig.show()
```

Ilustración 5: Grafo no dirigido

## Diferencias y Comparaciones

Para realizar diferencias y comparación suponemos que el Código 1 es el grafo dirigido y el código 2 es el grafo no dirigido:

### 1. Tipo de Grafo:

- **Código 1:** Se crea un grafo dirigido (DiGraph), lo que significa que las aristas tienen una dirección. Esto puede ser útil si se desea mostrar la dirección de la colaboración (aunque en este caso, las flechas son solo una representación visual).
- **Código 2:** Se crea un grafo no dirigido (Graph), donde las aristas no tienen dirección. Las colaboraciones se representan sin considerar un flujo o dirección específica.

## 2. Visualización de Aristas:

- **Código 1:** Se añade flechas rojas en el gráfico para mostrar la dirección de las aristas, aunque en la práctica, para un grafo no dirigido, esto puede no tener mucha utilidad a menos que se esté interesado en representar algún tipo de flujo o jerarquía en la colaboración.
- **Código 2:** Solo muestra las aristas como líneas grises, sin dirección.

## 3. Uso de Aristas Dirigidas:

- **Código 1:** Utiliza `nx.DiGraph` que permite la representación de aristas dirigidas. Las aristas se añaden con una dirección, y el gráfico muestra flechas para indicar esta dirección.
- **Código 2:** Utiliza `nx.Graph`, que no tiene en cuenta la dirección de las aristas. Todas las aristas son bidireccionales por defecto.

## 4. Configuración de la Visualización:

- **Código 1:** Se incluye flechas para representar la dirección de las aristas, lo que puede ayudar a visualizar mejor la estructura dirigida del grafo.
- **Código 2:** Solo se visualiza las aristas sin dirección, lo que es más adecuado para un grafo no dirigido.

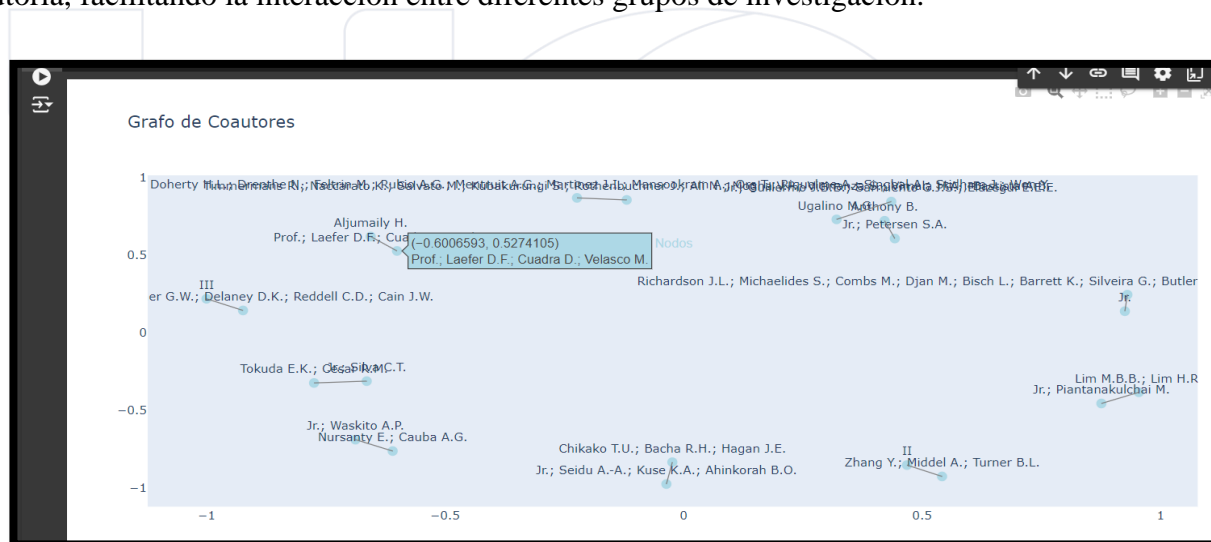
En resumen, si se requiere mostrar la dirección de las colaboraciones entre autores, el primer código es más adecuado. Si simplemente se requiere mostrar las colaboraciones sin tener en cuenta por la dirección, el segundo código es más apropiado.

## RESULTADOS

### Grafo No Dirigido:

En el grafo no dirigido, se pueden identificar clústeres de nodos que representan grupos de autores que tienden a colaborar frecuentemente entre sí. Estos clústeres indican la formación de comunidades de investigación en áreas temáticas específicas, reflejando una alta cohesión entre los miembros de cada grupo.

La distribución de los nodos en el layout de resorte revela la centralidad de ciertos autores dentro de la red. Algunos nodos ocupan posiciones más centrales, indicando que estos autores tienen un mayor número de conexiones (colaboraciones) en comparación con otros. Esto sugiere la existencia de autores clave que actúan como "hubs" en la red de coautoría, facilitando la interacción entre diferentes grupos de investigación.



*Ilustración 6: Resultado de grafo no dirigido*

### Grafo Dirigido:

En el grafo dirigido, las flechas indican la dirección de las colaboraciones. Aunque en el contexto de la coautoría la dirección puede no tener un significado práctico directo, este tipo de grafo permite explorar relaciones jerárquicas o secuenciales si existieran en los datos.

La estructura del grafo dirigido también permite identificar posibles flujos de conocimiento o influencia entre autores, donde ciertos nodos pueden aparecer como



centrales en la red, no solo son altamente productivos sino también influyentes en la formación de subredes de colaboración. Esto coincide con estudios previos que demuestran cómo ciertos investigadores actúan como "hubs" o "brokers" dentro de sus comunidades científicas, facilitando la conexión entre diferentes grupos de trabajo y promoviendo la interdisciplinariedad.

Por otro lado, la estructura jerárquica sugerida por el grafo dirigido plantea preguntas sobre las dinámicas de liderazgo y seguimiento en las colaboraciones científicas. Si bien en la coautoría típica, la dirección puede no ser explícita, la presencia de aristas dirigidas en ciertos contextos podría reflejar relaciones de mentoría o la secuencia en que los proyectos de investigación son desarrollados y compartidos entre diferentes autores. Esto resuena con teorías que sugieren que las redes académicas no solo son espacios de colaboración, sino también de transmisión de conocimiento y de formación de nuevos investigadores.

El análisis de la cohesión de los clústeres y las comunidades de autores muestra cómo los investigadores tienden a formar grupos estrechamente conectados, posiblemente trabajando en temas afines o dentro de los mismos grupos de investigación. Esto refleja una tendencia natural a la colaboración intensa dentro de ciertas disciplinas o temas de investigación, lo cual puede tener implicaciones importantes para la innovación y el desarrollo científico. Sin embargo, la fuerte cohesión también podría limitar la difusión de nuevas ideas fuera de estos grupos, un fenómeno conocido como "homofilia" en las redes sociales y académicas, donde la diversidad de perspectivas podría verse limitada.

### Conclusiones

Este estudio ha permitido una exploración detallada de la red de coautoría en la literatura científica utilizando grafos dirigidos y no dirigidos, proporcionando una visualización clara de las relaciones de colaboración entre autores. Se ha identificado que ciertos autores juegan un rol central dentro de sus comunidades científicas, actuando como nodos de alta conectividad que facilitan la cohesión y la interacción dentro de sus subredes. Además, la estructura de la red sugiere que estas colaboraciones no solo son frecuentes sino también estratégicamente importantes para la generación de conocimiento dentro de las disciplinas estudiadas.



Los hallazgos sugieren que el análisis de grafos es una herramienta poderosa para entender las dinámicas de colaboración en la academia, permitiendo identificar tanto los patrones de interacción más comunes como las excepciones significativas que podrían indicar tendencias emergentes o áreas de innovación. Sin embargo, la interpretación de estos grafos también requiere una comprensión contextual de las relaciones entre los autores, y cómo estas relaciones reflejan las dinámicas más amplias de producción y disseminación de conocimiento en la literatura científica.

En resumen, este estudio no solo visualiza las relaciones de coautoría, sino que también proporciona insights sobre la estructura social del proceso científico, destacando la importancia de ciertos actores y la naturaleza colaborativa de la investigación. Los resultados obtenidos podrían servir como base para futuras investigaciones que exploren cómo estas dinámicas de colaboración evolucionan con el tiempo y cómo afectan la producción científica en diferentes campos.

## BIBLIOGRAFÍA

Abid Ali Awan. (2024). *Introducción completa a las redes neuronales gráficas*.

DATA CAMP.

Obtenido

de

<https://www.datacamp.com/es/tutorial/comprehensive-introduction-graph-neural-networks-gnns-tutorial>

Barabási & Albert. (s.f.). *Emergence of scaling in random networks*. Science. Obtenido de <https://barabasi.com/f/67.pdf>

Grover, A., & Leskovec, J. (2016). *Node2Vec: Scalable feature learning for networks*. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 855-864. Obtenido de <https://dl.acm.org/doi/10.1145/2939672.2939754>

Kipf, T. N., & Welling, M. (2017). *Semi-supervised classification with graph convolutional networks*. *Proceedings of the 5th International Conference on Learning Representations (ICLR)*. Obtenido de <https://arxiv.org/pdf/1609.02907>

Newman. J ; Girvan. M. (2001 ; 2004). *Finding and evaluating community structure in networks*. Physical Review E, 64(1), 016131. Obtenido de <https://journals.aps.org/pre/abstract/10.1103/PhysRevE.64.016131> ;  
<https://journals.aps.org/pre/abstract/10.1103/PhysRevE.69.026113>

Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). *DeepWalk: Online learning of social representations*. Obtenido de <https://dl.acm.org/doi/10.1145/2623330.2623732>

Velickovic, P. C. (2018). *Graph attention networks*. Proceedings of the 6th International Conference on Learning Representations (ICLR). Obtenido de <https://arxiv.org/pdf/1710.10903>

