

**Universidad  
Rey Juan Carlos**

Escuela Técnica Superior  
de Ingeniería Informática

**Grado en Matemáticas**

**Curso 2022-2023**

**Trabajo Fin de Grado**

**GRAFOS DE COMPETITIVIDAD: DETECCIÓN DE  
COMUNIDADES MEDIANTE EL ALGORITMO DE  
GIRVAN-NEWMAN Y APLICACIÓN AL ESTUDIO  
DEL IBEX 35**

**Autor:** Daniel Recuero Cordobés  
**Tutor:** Regino Criado Herrero



# Agradecimientos

No quiero dejar la ocasión para agradecer a todas y cada una de las personas que me han acompañado durante estos años de universidad.

En primer lugar, agradecer a mi familia, mi madre Merce, mi padre César y mi hermana Sheila, por mostrar siempre confianza hacia mí y sentirse orgullosos del proceso. También a mis abuelos, los que están y los que estuvieron, que allá donde estén siempre me acompañan. Además, a Irene por ser mi apoyo incondicional y estar siempre.

Gracias a mis amigos y amigas que me han visto crecer durante estos años y que hoy se sienten tan contentos como yo por haber alcanzado el objetivo.

Gracias a mis compañeros de carrera y a los amigos que me llevo, solo ellos podían haber hecho el camino menos duro. Especial mención a Marta Casanova por el equipo que hicimos juntos.

También agradecer a mi tutor Regino Criado por haberme dado la oportunidad de trabajar con él.



# Resumen

En este trabajo se recoge un análisis sobre las empresas que cotizan en el IBEX 35. Para ello, se utiliza la teoría de grafos y, más concretamente, los grafos de competitividad. Estos objetos matemáticos permiten analizar las posibles relaciones entre los diferentes individuos del estudio.

Previamente al desarrollo del estudio práctico, se establecen las bases de la teoría de grafos: definiciones, resultados teóricos y ejemplos. Todo ello ayuda a una mejor interpretación de lo que se trabajará posteriormente.

Seguidamente, en la parte práctica, se comienza el estudio sobre las acciones del IBEX 35. Estos activos se estudian a partir del concepto de *competitividad* que permitirá establecer, sobre la evolución de sus cotizaciones, las potenciales relaciones entre ellos. La bolsa de valores es un campo sobre el que existe mucho conocimiento pero aún no se ha encontrado una metodología perfecta que permita a los inversores predecir los posibles movimientos de las gráficas de cotización.

El objetivo de este análisis es encontrar las comunidades que pueden formar las empresas a partir de su representación en los grafos que se van a construir, atendiendo cada uno de ellos a un punto de vista distinto sobre la cotización en los mercados financieros. Para poder detectar las distintas comunidades, se recurre al algoritmo de Girvan-Newman, el cual es uno de los posibles algoritmos desarrollados que permite la identificación de dichas comunidades.

Por consiguiente, en este trabajo se presenta un desarrollo novedoso e interesante al ámbito de la inversión en los mercados financieros, uniendo las Matemáticas y la Economía.

## Palabras clave:

- Grafo de competitividad
- Matriz de adyacencia
- Competitividad
- Precios de cotización
- Algoritmo de Girvan-Newman
- Identificación de comunidades



# Índice de contenidos

<b>Índice de figuras</b>	<b>XI</b>
<b>Índice de pseudocódigos</b>	<b>XIII</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Contexto y alcance . . . . .	1
1.2. Estructura del documento . . . . .	2
<b>2. Objetivos</b>	<b>5</b>
<b>3. Contenidos principales</b>	<b>7</b>
3.1. Teoría de grafos . . . . .	7
3.1.1. Origen e historia . . . . .	8
3.1.2. Definiciones básicas en la teoría de grafos . . . . .	10
3.1.3. Grafos de competitividad . . . . .	19
3.2. Detección de comunidades: Algoritmo de Girvan-Newman . . . . .	25
3.3. Conceptos del contexto económico . . . . .	31
<b>4. Desarrollo del estudio</b>	<b>34</b>
4.1. Ingesta de datos . . . . .	35
4.2. Desarrollo de las diferentes perspectivas del estudio . . . . .	36
4.2.1. Desarrollo de las subvariantes para las perspectivas anteriores	39
4.3. Estudio del coeficiente de correlación de Kendall . . . . .	40
4.4. Aplicación del algoritmo de Girvan-Newman . . . . .	41
<b>5. Resultados</b>	<b>43</b>
5.1. Estudio del coeficiente de correlación de Kendall . . . . .	43
5.2. Aplicación del algoritmo de Girvan-Newman . . . . .	44
<b>6. Conclusiones y trabajos futuros</b>	<b>54</b>
<b>Bibliografía y Referencias</b>	<b>58</b>
<b>Apéndices</b>	<b>61</b>

<b>A. Funciones definidas y empleadas en la práctica</b>	<b>63</b>
<b>B. Resultados obtenidos en la práctica</b>	<b>66</b>
<b>C. Recursos para la extracción de conclusiones</b>	<b>75</b>
C.1. Caso Endesa y Sacyr . . . . .	75
C.2. Caso Solaria y Red Eléctrica . . . . .	76
<b>D. Aplicación del algoritmo de Girvan-Newman sobre los grafos comentados</b>	<b>77</b>
D.1. Perspectiva 1 . . . . .	77
D.2. Perspectiva 2 . . . . .	80
<b>E. Repositorio de archivos</b>	<b>83</b>
<b>F. Distribución del tiempo de trabajo</b>	<b>85</b>



# Índice de figuras

3.1.	Situación de los puentes en la ciudad de Königsberg . . . . .	8
3.2.	Ejemplo de grafo simple . . . . .	12
3.3.	Ejemplo de subgrafo . . . . .	13
3.4.	Ejemplos de grafos completos . . . . .	13
3.5.	Ejemplo de grafo bipartido (a) y de grafo bipartido completo (b) .	15
3.6.	Ejemplo de grafo de competitividad . . . . .	22
3.7.	Ejemplo de iteración del algoritmo de Girvan-Newman . . . . .	29
3.8.	Ejemplo aplicación del algoritmo de Girvan-Newman al problema del club de kárate de Zachary . . . . .	30
4.1.	Representación de las tres medidas umbral . . . . .	40
4.2.	Bloque de código de la primera perspectiva para las iteraciones del algoritmo de Girvan-Newman . . . . .	41
5.1.	Representación de la evolución del coeficiente de correlación de Kendall en las tres perspectivas . . . . .	44
5.2.	Aplicación del algoritmo de Girvan-Newman a las perspectivas 3.2 y 3.3: Iteración 7 e Iteración 1, respectivamente . . . . .	47
5.3.	Dendograma relativo a la aplicación del algoritmo de Girvan-Newman a las subvariante 3.3 . . . . .	49
5.4.	Cuadro de rentabilidades conjuntas en el período . . . . .	52
A.1.	Implementación de la función <i>ordenarRanking</i> . . . . .	63
A.2.	Implementación de la función <i>seCruza</i> . . . . .	64
A.3.	Implementación de la función <i>insertarColumnaDiferencia</i> . . . . .	64
A.4.	Implementación de la función <i>insertarColumnaVariacion</i> . . . . .	65
A.5.	Implementación de la función <i>coefKendall</i> . . . . .	65
B.1.	Ejemplo de ingesta de datos para Inditex . . . . .	66
B.2.	Ejemplo de ranking relativo a la perspectiva 1 . . . . .	67
B.3.	Ejemplo de ranking relativo a la perspectiva 2 . . . . .	68
B.4.	Ejemplo de ranking relativo a la perspectiva 3 . . . . .	69
B.5.	Último ranking relativo a la perspectiva 1 . . . . .	70
B.6.	Código relativo a la matriz de rankings de la perspectiva 1 . . . . .	71

B.7. Matriz de rankings relativa a la perspectiva 1 . . . . .	72
B.8. Código relativo a la matriz de adyacencia de la perspectiva 1 . . . . .	72
B.9. Matriz de adyacencia de la perspectiva 1 . . . . .	73
B.10. Ejemplo de código para la representación óptima de los grafos relativos a la perspectiva 1 . . . . .	74
D.1. Aplicación del algoritmo de Girvan-Newman a la perspectiva 1.1: Iteración 7 . . . . .	78
D.2. Aplicación del algoritmo de Girvan-Newman a las perspectivas 1.2 y 1.3: Iteración 5 e Iteración 3, respectivamente . . . . .	79
D.3. Aplicación del algoritmo de Girvan-Newman a las perspectivas 2.2 y 2.3: Iteración 6 e Iteración 7, respectivamente . . . . .	82
E.1. Repositorio GitHub . . . . .	84
F.1. Tiempo total empleado en el desarrollo del trabajo . . . . .	85
F.2. Distribución temporal 1 . . . . .	86
F.3. Distribución temporal 2 . . . . .	86
F.4. Distribución temporal 3 . . . . .	86
F.5. Distribución temporal 4 . . . . .	87



# Índice de pseudocódigos

1.	Algoritmo de Girvan-Newman . . . . .	29
----	--------------------------------------	----



# 1

## Introducción

Previamente a la exposición de conceptos teóricos que servirán de base para la posterior explicación del estudio práctico, en esta sección introductoria se presenta el marco y el contexto en el que se desarrolla este trabajo. Seguidamente, se establece la estructura del mismo.

### 1.1. Contexto y alcance

Desde los inicios, el ser humano ha tendido a comercializar con diferentes tipos de bienes. Es por ello que ciertas referencias sitúan el origen de la bolsa en la Antigua Grecia y en el Imperio Romano [5]. Sin embargo, en lo que sí coinciden todas ellas es que no fue hasta el siglo XVI cuando surgió la idea de la bolsa de valores tal como se conoce hoy en día. Aquí cobra importancia la familia Van der Buérse, miembros de la nobleza holandesa, que realizaban intercambios mercantiles en un edificio de su propiedad en la ciudad de Brujas [6]. Finalmente, sería en 1602 cuando se instauraría la primera bolsa de valores conocida como tal. Su localización fue Ámsterdam y sus creadores la Compañía de las Indias Orientales Holandesa. Su objetivo era poder financiar los viajes mediante la compra venta de títulos. Seguidamente, aparecerían a finales del siglo XVIII las bolsas de Nueva York y París, entre otras. Acercándose a nuestro contexto, la bolsa de Madrid fue fundada en 1831.

Desde entonces, la complejidad de los productos con los que se comercializa en estos lugares ha crecido, debido al amplio abanico que ofertan: acciones, fondos indexados, bonos, entre otros. Además, los lugares como bolsa de valores siguen

manteniéndose pero cada vez es más fácil poder invertir en ciertos activos de manera online a través de ciertas plataformas que lo facilitan. Acompañado de ello, están las circunstancias que rodean al sector económico: crisis financieras, inflación, epidemias y una larga e innumerable lista de factores a tener en cuenta para que estos activos tengan un mayor o un menor valor.

Es por todo esto que muchos autores, investigadores y corredores de bolsa buscan la técnicas perfectas que les ayuden a predecir los movimientos que realizarán las gráficas que representan la evolución en el tiempo del valor de estos productos.

Basándose en esta situación, el presente estudio pretende presentar una herramienta más a esta detección de patrones y tendencias de forma que se pueda disponer de un punto de vista adicional a las métricas y parámetros empleados usualmente al realizar una inversión.

Con todo esto, a partir de los datos recogidos se pretende obtener una información cuya extrapolación sirva para poder hacer una inversión basada en el análisis realizado.

Este estudio se apoya en la teoría de grafos, habida cuenta de que esta es un área importante dentro de las matemáticas cuya aplicación a distintos problemas de naturaleza, no siempre matemática, hace de este campo una herramienta de calidad para la resolución de los mismos.

Al margen del análisis presentado en este trabajo, los factores que hacen que las acciones varíen su precio son tan variados que es prácticamente imposible predecir sus movimientos, pues hasta una simple noticia puede tener como consecuencia movimientos bruscos en los precios de cotización para una empresa en un período de tiempo muy corto. La utilidad principal de este trabajo es añadir las conclusiones observadas a una larga lista de más investigaciones y, entre todas, mejorar el conocimiento del mercado financiero.

## 1.2. Estructura del documento

En primer lugar, tras esta parte introductoria, se van a definir los *Objetivos* del estudio.

Seguidamente, se tratarán los *Contenidos principales* del trabajo. En este apartado, se busca explicar todos los fundamentos teóricos que posteriormente se emplearán para el estudio práctico. Como fundamentos teóricos se tiene, en primer lugar, la teoría de grafos, la cual comienza con un poco de contexto sobre esta teoría y sus orígenes. A continuación, se dan los conceptos básicos y algún resultado a destacar. Posteriormente, se profundiza en lo que realmente se aplica al estudio, los grafos de competitividad y, de nuevo, se hace una revisión de sus

conceptos más importantes. Por último, puesto que el trabajo tiene relación con el contexto económico, es necesario incluir ciertos aspectos del mismo para una mejor interpretación del estudio.

Tras esta gran sección, aparece el *Desarrollo del estudio*, donde se comenta todo lo construido para la parte práctica. Este apartado es importante porque ayuda a entender qué es lo que realmente se busca con el estudio. En consecuencia, aparece el siguiente punto, llamado *Resultados*, donde se concentran las observaciones y el análisis sobre la aplicación del algoritmo de Girvan-Newman a las diferentes perspectivas presentadas en el estudio. Además, esta sección contiene una interpretación sobre la gráfica donde se representa la evolución del coeficiente de correlación de Kendall a lo largo de la familia de rankings.

Finalmente, se tiene una última sección, *Conclusiones y trabajos futuros* donde se recoge la información que realmente se buscaba con este estudio, es decir, con los resultados obtenidos, se interpretan para extraer conclusiones de las situaciones estudiadas. Además, se incluyen posibles variantes y mejoras a realizar que se podrían tener en cuenta para una posible evolución del estudio.

Adicionalmente a toda esta estructura anterior, existe un apartado de *Bibliografía y referencias* para posibles consultas de las fuentes empleadas y un amplio apartado de *Apéndices*. Esta última parte aparece de la necesidad de ilustrar cómo funciona parte del código de la práctica y de poder visualizar todos los resultados obtenidos. Aunque aquí se recogen los más importantes, existe un repositorio en GitHub que permite la consulta completa del código desarrollado.



# 2

## Objetivos

Como objetivo general de este trabajo se establece la revisión de conceptos y resultados de la teoría de grafos y la búsqueda de conclusiones de valor que puedan aportar algún conocimiento sobre una posible inversión en alguna empresa del IBEX 35, bien basándonos en extrapolar los datos analizados en el período temporal estudiado, bien utilizando las herramientas desarrolladas en períodos cercanos a la posible inversión. Además, si las técnicas empleadas fuesen potencialmente útiles, se podrían extrapolar para otros mercados e incluso para otros productos financieros.

Por otro lado, los objetivos específicos del estudio son:

1. Exploración de conceptos básicos sobre la teoría de grafos.
2. Ampliación del conocimiento sobre la teoría de grafos mediante el análisis y la aplicación del algoritmo de Girvan-Newman a grafos de competitividad.
3. Extrapolación de conclusiones del estudio a situaciones reales.
4. Aplicación de los conceptos teóricos a un caso práctico real.
5. Análisis de objetos de la realidad desde una perspectiva matemática.
6. Desarrollo de conclusiones a partir de observaciones originadas tras la adaptación autónoma de técnicas matemáticas.
7. Realización de un escrito formal que aporte valor real.



# 3

## Contenidos principales

En este apartado se busca sentar las bases teóricas del estudio. En primer lugar, dar respuesta a qué son los grafos y algunos resultados interesantes de la teoría de grafos relacionados con el trabajo que se presenta. Además, se profundizará en los grafos de competitividad, principal objeto de trabajo del estudio. Por último, se presentan conceptos relativos a la bolsa y, concretamente, a las acciones y a cómo se pueden estudiar para poder entender mejor los objetivos de esta investigación.

### 3.1. Teoría de grafos

La teoría de grafos es el eje central en torno al cual gira la base teórica del estudio. Es por ello que hay que conocer ciertas definiciones básicas y algunos resultados que puedan aportar cierto conocimiento al lector para una mejor comprensión de la aplicación práctica posterior.

Partiendo del origen de la teoría de grafos, el fin último es llegar a conocer los grafos de competitividad que serán sobre los cuales estará basada la parte práctica. Como referencias principales que han contribuido a dar forma a esta sección, el origen de la teoría de grafos y algunos apuntes históricos han sido extraídos de *Una breve introducción a la teoría de grafos*[1] y *Breve historia de la Teoría de Grafos*[2].

Por otro lado, los conceptos más básicos y unas primeras definiciones, junto

con los resultados teóricos, se han obtenido de los libros: *Introducción a la teoría de grafos*[3], *Puntos y flechas: teoría de los grafos*[8] y *Graph Theory*[4]. Además, se han tomado conceptos del texto *Un cuatrimestre de Matemática Discreta* [13].

Para poder definir y exponer conceptos de grafos de competitividad, sus componentes y los diferentes parámetros a estudiar, se han revisado y extraído los conceptos principales de los artículos: *Comparing rankings by means of competitiveness graphs: structural properties and computation*[9], *A new method for comparing rankings through complex networks: Model and analysis of competitiveness of major European soccer leagues*[10] y *Comparing series of rankings with ties by using complex networks: An analysis of the Spanish stock market (Ibex-35 Index)*[11].

### 3.1.1. Origen e historia

Desde el origen de la teoría de grafos, el mayor impulso para investigar en este campo han sido las aplicaciones prácticas que aporta a problemas concretos. Esto es debido a que es considerada como una teoría matemática perfecta por su rigurosidad conceptual y por su aplicación a la realidad.

#### PROBLEMA DE LOS PUENTES DE KÖNIGSBERG

El primer trabajo en este área de investigación fue publicado para tratar el conocido **Problema de los Puentes de Königsberg**. Königsberg, actualmente Kaliningrado (Rusia), era entonces la capital de Prusia Oriental y contaba con 7 puentes los cuales conectaban las diferentes partes de la ciudad que estaban separadas por las bifurcaciones del río Pregolya.

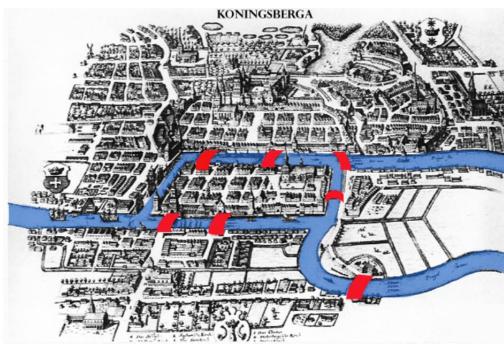


Figura 3.1: Situación de los puentes en la ciudad de Königsberg

La imagen 3.1 representa la situación que entonces tenía la ciudad. El problema buscaba encontrar el camino que pasara por todos los puentes una sola vez de forma que al terminarlo el punto final fuese el punto inicial de dicho camino. Durante años, los ciudadanos se plantearon esta problemática a "modo de

## Capítulo 3. Contenidos principales

---

juego”, pero fue en 1736 cuando Euler publicó el artículo con la solución a años de debate.

Leonhard Euler concluyó que **no** era posible encontrar el recorrido que cumpliese dichas características. Además, este resultado tuvo importancia por ser de los primeros resultados topológicos en geometría ya que no dependía de ninguna medida, y permitió iniciar también la relación entre la topología y la teoría de grafos.

### LEYES DE KIRCHHOFF

Años más tarde, en 1846, Gustav Kirchhoff, tras combinar el estudio de las redes eléctricas con los principios de la teoría de grafos, publicó las **Leyes de Kirchhoff**, las cuales permiten calcular en los circuitos eléctricos el voltaje y la intensidad que pasa por ellos.

### PROBLEMA DE COLORACIÓN

A mediados del siglo XIX, Francis Guthrie planteó si era posible colorear con cuatro colores cualquier mapa de países de forma que dos países colindantes no tuvieran el mismo color. El origen de este problema, llamado **Problema de Coloración**, está en 1852 y, aunque no fue resuelto por su pensador original, en 1976 Kenneth Appel y Wolfgang Haken encontraron la solución. Se dice entonces que dio inicio a lo que hoy se conoce como la teoría de grafos, ya que se sentaron los primeros conceptos, definiciones y resultados básicos de la teoría de grafos.

### PROBLEMA DE ENUMERACIÓN DE LOS ISÓMEROS

El matemático Arthur Cayley en 1857 planteó enumerar los isómeros, que son compuestos químicos que comparten fórmula química pero cuya estructura molecular es diferente. Para ello, planteó un grafo de tipo árbol que enlazaba átomos y así pudo aplicar la teoría de grafos a esta casuística.

Posteriormente, se adjudicaría el término ‘grafo’ a lo que se conoce hoy en día debido a la representación de los enlaces entre los átomos de una molécula.

### PROBLEMA DEL CAMINO MÍNIMO

Un problema tradicional en la teoría de grafos es el de **encontrar el camino mínimo entre dos puntos**, es decir, dado un par de puntos del grafo, como cada arista que une puntos del grafo tiene un determinado peso, se buscaba viajar de un punto dado a otro con un peso mínimo acumulado.

El primer autor en tratar el problema fue Lester Ford JR, que publicó en 1956 un algoritmo que lo resolvía. Posteriormente, trabajaría con Richard Bellman para desarrollar el algoritmo de Bellmann-Ford, el cual era más eficiente y trataba más casuísticas que el primero publicado.

Seguidamente, en 1959, apareció el algoritmo de Dijkstra, por Edsger Dijks-

tra, como su mismo nombre indica, el cual, junto con el anterior, es de los más reconocidos para la resolución de este problema.

Paralelamente a este, Bertrand Roy publicó el algoritmo de Floyd-Warshall, el cual sería actualizado por Robert Floyd en 1962 para darse a conocer con la versión actual. Este ya no trataba en un par de puntos, si no que a través de una matriz era capaz de dar el camino mínimo entre todos los puntos del grafo.

Este problema va a ser clave para la aplicación del algoritmo de detección de comunidades de Girvan-Newman que posteriormente se explicará. Se trata de la base para que el algoritmo funcione correctamente.

#### PROBLEMAS ACTUALES

En la actualidad, la aplicación de los grafos va mucho más allá y se puede ver en diversos campos como la sociología, para analizar las relaciones que se dan dentro de una red social, la economía, donde entraría esta investigación para poder conocer ciertas características de los mercados financieros, o para modelar las relaciones o interacciones entre los agentes que intervienen en un determinado fenómeno o sistema: individuos, sujetos, empresas...

### 3.1.2. Definiciones básicas en la teoría de grafos

Tras haber dado una idea del origen y las posibles aplicaciones de la teoría de grafos, es necesario recordar algunas definiciones básicas.

Se comenzará definiendo formalmente qué es un grafo, sus componentes y conceptos entorno a estas estructuras que son importantes a la hora de trabajar con ellos.

**Definición 1** (Grafo simple). *Un **grafo simple**  $G$  es un par  $(V(G), E(G))$  donde  $V(G)$  es un conjunto finito no vacío de elementos llamados **nodos** y  $E(G)$  es un conjunto finito de pares no ordenados no repetidos de elementos de  $V(G)$  llamados **aristas**.*

**Nota 1.**

1. *El hecho de que los pares de  $E(G)$  sean no ordenados hace que el grafo  $G$  reciba el nombre de grafo no dirigido.*
2.  *$E(G)$  se forma por pares no ordenados y no repetidos, lo que significa que dicho par solamente puede existir, como máximo, una vez en el conjunto  $E(G)$  debido a que  $G$  es un grafo simple.*

**Nota 2.** *A la hora de la visualización gráfica de un grafo no hay que confundir su representación con su definición. Para un grafo, la definición es única pero el número de representaciones del mismo es infinito.*

**Definición 2** (Nodos adyacentes y nodos incidentes). *Sea un grafo  $G=(V(G),E(G))$  y sean  $u,v \in V(G)$ , se dice que  $u$  y  $v$  son **nodos adyacentes** si existe una arista en  $E(G)$  que los une. Se dice entonces también que  $u$  y  $v$  son **nodos incidentes** en dicha arista.*

**Definición 3** (Aristas adyacentes). *Sea un grafo  $G=(V(G),E(G))$  y sean  $a,b \in E(G)$ , se dice que  $a$  y  $b$  son **aristas adyacentes** si tienen un nodo de  $V(G)$  en común.*

**Definición 4** (Grado de un nodo). *Sea un grafo  $G=(V(G),E(G))$  y sea  $u \in V(G)$ , se define el **grado del vértice**  $u$  como el número de aristas del grafo que inciden en él. Se denota por  $g(u)$ .*

**Nota 3.** El **cardinal de un conjunto**  $A$ , es decir, el número de elementos que este contiene vendrá dado por la notación  $|A|$ .

A continuación, se va a definir lo que se llama matriz de adyacencia asociada a un grafo.

**Definición 5** (Matriz de adyacencia). *Dado un grafo  $G=(V(G),E(G))$ , se define la **matriz de adyacencia**  $M$  asociada al grafo  $G$  como la matriz cuadrada donde las filas y las columnas hacen referencia a los nodos del grafo de forma que:*

$$M_{ij} = \begin{cases} 1 & \text{si existe arista entre el nodo } i \text{ y el nodo } j \\ 0 & \text{en otro caso} \end{cases}$$

Este objeto matemático es clave para poder trabajar con grafos ya que permite resumir toda la información en una estructura que resulta útil para la aplicación de algoritmos, estudiar propiedades del grafo, etcétera.

**Observación 1.** Es importante tener en cuenta que la ordenación de los nodos para la creación de la matriz de adyacencia es esencial, debido a que a diferente ordenación de los mismos, distinta matriz de adyacencia se obtendrá, aunque reflejen la misma información. Por tanto, a la hora de aplicarla en algún cálculo, esto afectaría al resultado que se obtenga.

**Observación 2.** Debido a la teoría algebraica, la matriz de adyacencia permite calcular directamente algunos de los elementos del grafo:

- El número de vértices de un grafo es el número de filas o el número de columnas de su matriz de adyacencia.
- El grado de un nodo es la suma de los números de su fila o de su columna.
- El número de aristas de un grafo ( $|E(G)|$ ) es la mitad de la suma de los números de la matriz de adyacencia.

**Definición 6** (Nodo aislado y nodo terminal). *Sea un grafo  $G=(V(G),E(G))$  y sea  $u \in V(G)$ , se dice que  $u$  es un **nodo aislado** si es de grado cero, y se dice que  $u$  es un **nodo terminal** si es de grado uno.*

**Nota 4.** Los nodos también pueden ser referidos como vértices o puntos y las aristas se pueden llamar arcos o líneas.

**Definición 7** (Clique). *Sea un grafo  $G=(V(G),E(G))$ , se define un **clique**  $C \subseteq V(G)$  como un subconjunto de nodos tales que para cada par de nodos de  $C$  existe una arista en  $E(G)$  que los une.*

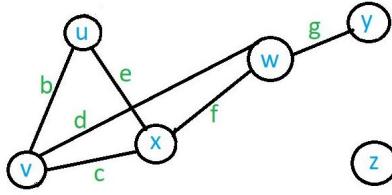


Figura 3.2: Ejemplo de grafo simple

Así pues, en este ejemplo 3.2 se tiene el grafo  $G=(V(G),E(G))$ , donde:

- ❖  $V(G) = \{u, v, w, x, y, z\}$ , con  $|V(G)| = 6$ ,
- ❖  $E(G) = \{b = \{u, v\}, e = \{u, x\}, d = \{v, w\}, c = \{v, x\}, f = \{w, x\}, g = \{w, y\}\}$ , con  $|E(G)| = 6$ ,
- ❖ Los nodos  $x, w$  son adyacentes ya que existe la arista  $f$  tal que  $f = \{w, x\}$ ; por ejemplo,
- ❖ Los nodos incidentes en la arista  $e = \{u, x\}$  son  $u, x$ ; por ejemplo,
- ❖ Las aristas incidentes en el nodo  $u$  son  $b, e$ ; por ejemplo,
- ❖ El nodo  $w$  tiene grado  $g(w) = 3$ ,
- ❖ El nodo  $y$  es un nodo terminal y el nodo  $z$  es un nodo aislado,
- ❖ El subconjunto  $C = \{u, v, x\}$  es un clique,
- ❖ Ordenando los nodos alfabéticamente en filas y columnas ( $\{u, v, w, x, y, z\}$ ) se tiene la siguiente matriz de adyacencia para ese grafo:

$$\begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.1)$$

Una vez definidos los elementos básicos de los grafos, se pasa entonces a recordar algunos grafos característicos con sus definiciones formales.

**Definición 8** (Subgrafo). *Sea un grafo  $G = (V(G), E(G))$ , se define un **subgrafo**  $H = (V(H), E(H))$  como un grafo tal que  $V(H) \subseteq V(G)$  y  $E(H) \subseteq E(G)$ . Esto se denota como  $H \prec G$ .*

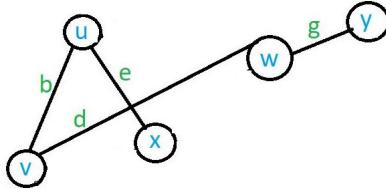


Figura 3.3: Ejemplo de subgrafo

Como se puede observar, en el ejemplo de subgrafo 3.3, el grafo  $H = (V(H), E(H))$  es un subgrafo del grafo  $G$  del ejemplo 3.2, ya que  $V(H) = \{u, v, w, x, y\} \subseteq V(G)$  y  $E(H) = \{b = \{u, v\}, e = \{u, x\}, d = \{v, w\}, g = \{w, y\}\} \subseteq E(G)$ .

Seguidamente, hay que recordar las definiciones de algunos tipos de grafo que por su singularidad merecen ser mencionados. Estos grafos, en numerosas bibliografías, reciben el nombre de **grafos notables**.

**Definición 9** (Grafo completo). *Se define el **grafo completo** como el grafo en el que para cualquier par de nodos existe una arista, es decir, todos sus vértices son adyacentes. Se denota por  $K_n$ , donde  $n$  es el cardinal de  $V(G)$ . Matemáticamente, se define como:*

1.  $V(G) = \{u, v, w, \dots, z\}$  con  $|V(G)| = n$
2.  $E(G) = \{\{x, y\} \mid x, y \in V(G), x \neq y\}$

**Observación 3.** *El número de aristas de un grafo completo de  $n$  vértices es  $\frac{1}{2}n(n - 1)$ . Esto se debe a que por combinatoria como las aristas se forman por un par de nodos, la elección de todos los pares de nodos entre los  $n$  posibles es  $n(n - 1)$  y luego como las aristas son no dirigidas, es decir, la arista  $\{u, v\}$  es la misma que la  $\{v, u\}$ , hay que dividir el total entre 2, resultando así el número total de aristas indicado.*

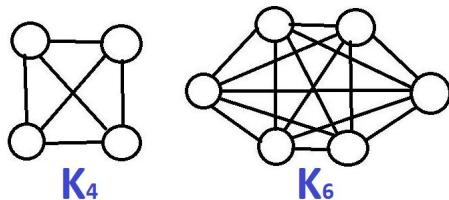


Figura 3.4: Ejemplos de grafos completos

En la figura 3.4 se tiene una representación de los grafos completos  $K_4$  y  $K_6$ . Aplicando la Observación 3 se tiene, además de por conteo sobre las figuras, que:

- Para  $K_4$  se tiene que  $\frac{1}{2}4(4 - 1) = 6$  aristas.
- Para  $K_6$  se tiene que  $\frac{1}{2}6(6 - 1) = 15$  aristas.

**Definición 10** (Grafo regular). *Se dice que un grafo es un **grafo regular** si todos sus nodos tienen el mismo grado.*

**Observación 4.** *Un grafo completo de  $n$  nodos es un grafo regular en el que el grado de todos sus nodos es  $n - 1$ .*

**Definición 11** (Grafo bipartido). *Sea un grafo  $G=(V(G),E(G))$ , se dice que este es un **grafo bipartido** cuando se puede dividir el conjunto  $V(G)$  en dos subconjuntos disjuntos de forma que cada arista del conjunto  $E(G)$  tenga un nodo de cada uno de los dos subconjuntos. Matemáticamente, se expresa como:*

1.  $V(G) = A \cup B$  con  $A \cap B = \emptyset$  ( $A, B$  subconjuntos disjuntos de  $V(G)$ ).
2. Si  $\{u, v\} \in E(G)$ , entonces  $u \in A$  y  $v \in B$  o bien  $v \in A$  y  $u \in B$ .

**Observación 5.** *Para comprobar si un grafo es bipartido o no, basta con seguir el siguiente algoritmo de bipartición de un grafo:*

1. Se elige un vértice al azar y se marca como tipo  $A$ .
2. Todos los vértices adyacentes al anterior nodo se marcan como tipo  $B$ .
3. Entre los vértices anteriores se elige uno al azar y se repiten los pasos 1 y 2.

Una vez concluido esto, o simultáneamente al proceso, pueden darse dos casuísticas:

- Se obtiene un vértice (o varios) que se clasifica como tipo  $A$  y  $B$  simultáneamente. En este caso, el grafo es no bipartido.
- Se ha finalizado el proceso y cada nodo corresponde a un y solo a un tipo de vértice  $A$  o  $B$ . En este caso, el grafo es bipartido y se ha obtenido la **bipartición** del mismo.

**Definición 12** (Grafo bipartido completo). *Sea un **grafo bipartido**  $G=(V(G),E(G))$ , se dice que este es **bipartido completo** cuando es un grafo bipartido en el que cada vértice de cada uno de los subconjuntos disjuntos está unido a todos los nodos del otro subconjunto. Este tipo de grafos se denotan por  $K_{m,n}$ , donde  $m, n$  son los números de nodos que pertenecen a cada uno de los conjuntos disjuntos. Matemáticamente, se define como:*

1.  $V(G) = A \cup B$  con  $|A| = m$  y  $|B| = n$  y  $A \cap B = \emptyset$ .
2. El grafo es un grafo bipartido con partición  $A, B$ .
3. Si  $u \in A$ , entonces  $\forall v \in B \quad \exists \{u, v\} \in E(G)$ , o viceversa cambiando el conjunto  $A$  por el  $B$  y el  $B$  por el  $A$  en esta condición.

**Observación 6.** De la propia definición se deduce que el cardinal de  $E(G)$  de un grafo bipartido completo es  $m * n$ . Esto se debe a que, por ejemplo, tomando los nodos del primer "grupo", cada uno tendrá una arista con cada uno de los nodos del segundo "grupo", luego, de cada nodo del primer grupo saldrán  $n$  aristas. Como el primer grupo tiene  $m$  nodos, en total habrá  $m * n$  aristas, ya que si se tomase la misma visión pero desde los nodos del segundo grupo, las aristas correspondientes ya estarían contempladas con la primera casuística.



Figura 3.5: Ejemplo de grafo bipartido (a) y de grafo bipartido completo (b)

Como se ve en la figura 3.5, ambos grafos son bipartidos porque cumplen con la definición para serlo, siendo los dos conjuntos disjuntos diferenciados por colores. Además, en el caso (b) se trata de un grafo bipartido completo, concretamente  $K_{4,2}$ , ya que todos los nodos de cada conjunto tienen una arista hacia los del otro conjunto y viceversa. Tomando entonces la Observación 6, se tiene que el número total de aristas es  $4 * 2 = 8$ , como se puede ver por conteo.

Además de los elementos básicos que se han definido anteriormente sobre los grafos, existen operaciones que se pueden realizar sobre ellos. Estas son las siguientes definiciones.

**Definición 13** (Camino). Un **camino** en un grafo  $G=(V(G),E(G))$  es una secuencia de aristas tal que el final de una es el inicio de la siguiente. Para denotarlo, se puede escribir entre paréntesis la secuencia de nodos o la secuencia de aristas y se puede referir a él con letras griegas.

**Definición 14** (Camino simple y camino elemental). Sea un camino en un grafo  $G=(V(G),E(G))$ :

- El camino es **simple** si no se repite ninguna arista.

- *El camino es **elemental** si no se repite ningún nodo.*

**Definición 15** (Círculo). *Un **círculo** en un grafo  $G=(V(G),E(G))$  es un camino donde el nodo inicial y el nodo final es el mismo.*

**Nota 5.** *El concepto de circuito también acoge las definiciones simple y elemental dadas para los caminos.*

**Definición 16** (Longitud de un camino). *Se define la **longitud de un camino** en un grafo como el número de aristas que posee. Así pues, un camino  $\beta$  tendrá una longitud  $l(\beta)$ .*

**Definición 17** (Distancia entre dos nodos). *Dado un grafo  $G=(V(G),E(G))$  y dos nodos  $u,v \in V(G)$ , se define la **distancia entre los nodos**  $u$  y  $v$  como la mínima longitud de todos los posibles caminos que unen estos dos vértices. Se denota por  $d(u,v)$ .*

**Definición 18** (Diámetro de un grafo). *Dado un grafo  $G=(V(G),E(G))$  se define el **diámetro del grafo** como la mayor distancia existente entre dos nodos del grafo.*

**Nota 6.** *Si en un grafo no existiese camino posible entre dos nodos, entonces, por convenio, la distancia entre ellos es  $\infty$  y el diámetro del grafo pasaría a ser también  $\infty$ .*

Volviendo al grafo de la figura 3.2, se pueden poner ejemplos de los conceptos anteriores:

- ❖ Se tiene que  $\alpha = (d,f,c) = (v,w,x,v)$  es un camino y, además, simple ya que no repite ninguna arista, aunque no sería elemental por pasar dos veces por el vértice  $v$ .
- ❖ Sea  $\gamma = (b,e,f,g)$ , este es un camino elemental al no repetirse ningún nodo y, también, simple por no repetirse ninguna arista.
- ❖ El camino  $\alpha$  es un circuito simple y elemental, ya que aunque repita un vértice, este es el de origen y final.
- ❖ Para los caminos anteriores se tiene, que  $l(\alpha) = 3$  y  $l(\gamma) = 4$ .
- ❖ La distancia entre los nodos  $v, y$  es 2, ya que el menor camino entre ellos es el camino  $\sigma = (d,g) = (v,w,y)$ .
- ❖ El diámetro del nodo de la figura 3.2 es  $\infty$ , debido a que existe el nodo aislado  $z$  al cual no se puede llegar desde ningún nodo.

Tras tener definidos los conceptos de caminos y circuitos, se pueden introducir las siguientes definiciones.

**Definición 19** (Grafo conexo y grafo inconexo). *Se dice que grafo  $G=(V(G),E(G))$  es un **grafo conexo** si para cada par de nodos de  $V(G)$  existe un camino posible de aristas de  $E(G)$  para llegar de un vértice al otro. En caso contrario, cuando para un par no existe dicho camino, el grafo se dice que es un **grafo inconexo**.*

De nuevo, tomando la figura 3.2, este es un grafo inconexo debido a que el vértice  $z$  no tiene caminos posibles al resto de los nodos. Sin embargo, el ejemplo 3.3 sí es un grafo conexo.

**Definición 20** (Componente conexa). *Sea un grafo  $G=(V(G),E(G))$ , se define una **componente conexa** del mismo como un subgrafo  $H=(V(H),E(H))$  que verifica:*

- i)  $H$  es conexo,
- ii) Si  $\{u,v\} \in E(G)$  y  $u \in V(H)$ , entonces  $v \in V(H)$  y  $\{u,v\} \in E(H)$ .

*En otras palabras, las componentes conexas son subgrafos conexos maximales, en el sentido de que no están contenidos en otro subgrafo conexo del grafo original.*

**Observación 7.** *De la propia definición de componente conexa aparece la siguiente propiedad:*

*Si un grafo es conexo este posee una única componente conexa que es él mismo.*

Para verlo con un ejemplo, en el caso del grafo 3.2 se tienen dos componentes conexas: la primera formada por el nodo aislado  $\{z\}$  y la segunda formada por los nodos  $\{u, v, w, x, y\}$  y sus respectivas aristas incidentes en ellos. Otro ejemplo sería la figura 3.5 donde el grafo (a) tiene 3 componentes conexas mientras que el grafo (b) es un grafo conexo y, por tanto, tiene una única componente conexa.

**Nota 7.** *Aquí hay que recordar que, por definición, un grafo es subgrafo de él mismo ya que la definición de subgrafo está dada con un "contenido o igual". Es por ello que se puede hablar de un grafo conexo como un grafo que tiene una componente conexa, que es el grafo en sí mismo.*

Tras establecer los principales elementos de la teoría de grafos, es conveniente dar algunos resultados teóricos importantes como proposiciones, lemas o corolarios.

Para empezar, se expone el lema del apretón de manos y un corolario suyo.

**Lema 1** (Lema del apretón de manos). *Sea un grafo  $G=(V(G),E(G))$  simple y no dirigido, se tiene que la suma de los grados de sus nodos es un número par. Además, cumple que:*

$$\sum_{u \in V(G)} g(u) = 2|E(G)|.$$

**Corolario 1** (Corolario del lema del apretón de manos). *En cualquier grafo el número de vértices de grado impar tiene que ser un número par.*

No se incluye demostración ninguna debido a que es una obviedad. Puesto que cada arista tiene dos extremos, que son nodos, al sumar todos los grados de todos ellos, cada arista se cuenta dos veces. Ello implica que el número total de la suma de los grados de los nodos sea par y, además, sea dos veces el número de aristas totales del grafo.

Por otro lado, en el caso de que hubiese un nodo de grado impar, siempre tiene que haber otro nodo de grado impar al que se una, luego el número de nodos de grado impar en un grafo simple es un número par.

Estos resultados son útiles a la hora de tratar un grafo, ya que dan una propiedad obvia pero que no siempre se tiene en cuenta durante el trabajo con estos objetos matemáticos.

A continuación, se expone un resultado sobre grafos conexos.

**Proposición 1.** *Sea  $G=(V(G),E(G))$  un grafo simple conexo, entonces existe un camino simple entre cualquier par de vértices.*

Demostración Hay que recordar que un camino simple es aquel que no repite ninguna arista. En ese sentido, por ser el grafo conexo, dados dos nodos  $u, v \in V(G)$  existe un camino entre ellos y, para la demostración, se toma el de menor longitud entre ellos. Por reducción al absurdo, si no fuese simple, existiría una arista  $a \in E(G)$ , incidente en los nodos  $u, v \in V(G)$ , por la que se pasaría más de una vez. Pero entonces, el camino que se ha tomado no es el de menor longitud porque habría aristas que se podrían suprimir sobre el camino que se había tomado. Por ello, contradicción y sí existe un camino simple entre cada par de nodos.

*c.q.d.*

Por último, se da una proposición acerca de una propiedad de los grafos bipartidos.

**Proposición 2.** *Sea  $G=(V(G),E(G))$  un grafo bipartido, para cualquier circuito que se construya, este es de longitud par.*

Nuevamente, no se expone una prueba puesto que es bastante sencillo de ver. En un grafo bipartido los nodos están divididos en dos subconjuntos y además entre dos nodos del mismo subconjunto no hay aristas incidentes. Por tanto, suponiendo que el nodo inicial sea del subconjunto  $A$ , para volver a dicho nodo habrá que pasar por al menos uno del subconjunto  $B$ , habiendo recorrido al menos dos aristas. Repitiendo esta condición tantas veces como sea preciso por el circuito, el número de aristas por las que se cruza es un múltiplo de 2. Por tanto, la longitud del circuito es par.

### 3.1.3. Grafos de competitividad

Para poder desarrollar la parte práctica hay que empezar por definir qué es un grafo de competitividad y en qué contexto se utiliza este concepto.

Lo primero es saber que los grafos de competitividad se van a utilizar para estudiar rankings y hay que analizar qué significa competir en este ámbito y qué es un ranking.

La primera definición que se va a tratar en este apartado es un concepto común a muchos estudios, pero se le va a dar sentido en este contexto.

**Definición 21** (Puntuación). *Sea un conjunto de  $n$  elementos o competidores  $A = \{x, y, \dots, n\}$ . Se define la **puntuación** como una función  $p : \{x, y, \dots, n\} \rightarrow \mathbb{R}$  donde a cada elemento del conjunto  $A$  se le asocia un número que tendrá distinta connotación según el contexto.*

Esta definición es importante para explicar el siguiente concepto porque tener una puntuación, también llamada valoración, de los elementos induce a hacer un ranking de ellos en función de este puntuaje.

**Definición 22** (Ranking). *Dado un conjunto de  $n$  elementos o competidores  $A = \{x, y, \dots, n\}$ , un **ranking**  $r$  es una función  $r : \{1, 2, \dots, n\} \rightarrow A$ . Se dice que un **ranking** es una lista ordenada de elementos en función de un criterio, el cuál suele ser una puntuación asociada a ellos. Es decir, un ranking se puede considerar como una permutación de los elementos que en él participan.*

Ejemplos claros de generar una puntuación sobre los elementos y crear a partir de esta rankings se verán más adelante cuando se explique la parte práctica del estudio.

**Nota 8.** *Puesto que se va a trabajar con una familia de rankings  $R = \{r_1, r_2, \dots, r_m\}$ , a partir de este punto el conjunto de elementos de los rankings vendrá dado por la notación  $V(G_R)$ , o  $V(G_{r_k})$ , con  $k \leq m$ , cuando sea para referirse a un ranking en particular (aunque hay que recordar que los elementos que ordenan son los mismos).*

*Además, se hablará de nodos del grafo y de elementos de los rankings indistintamente, haciendo referencia a los mismos objetos.*

**Nota 9.** *Dado un ranking  $r$  y un par de elementos  $u_i, u_j \in V(G_R)$ , con  $i \neq j$ ,  $i, j \leq n$ , se denota por  $r(u_i) > r(u_j)$  la situación en la que el elemento  $u_i$  está situado por encima del elemento  $u_j$  en el ranking  $r$ .*

*También se escribe  $r(u_i) \geq r(u_j)$  si  $r(u_i) > r(u_j)$  o  $i = j$ . Esto cobra sentido debido a que un ranking se puede representar como un vector  $(u_1, u_2, \dots, u_n)$  donde  $u_1$  es el primer elemento del ranking y  $u_n$  el último.*

Tras esta aclaración se establece la siguiente relación: sea  $p$  una puntuación y  $r$  un ranking sobre un conjunto  $A$ , con  $i, j \in A$ , entonces:

$$r(u_i) \geq r(u_j) \Leftrightarrow p(u_i) \geq p(u_j)$$

**Observación 8.** *Hay que destacar que la relación binaria de orden  $\geq$  en un ranking es una relación de orden total, ya que se cumple:*

1. *Reflexividad: sea  $u_i \in V(G_R)$ , claramente  $r(u_i) \geq r(u_i)$ .*
2. *Antisimetría: sean  $u_i, u_j \in V(G_R)$  tales que  $r(u_i) \geq r(u_j)$  y  $r(u_j) \geq r(u_i)$ , entonces  $u_i = u_j$ .*
3. *Transitividad: sean  $u_i, u_j, u_k \in V(G_R)$  tales que  $r(u_i) \geq r(u_j)$  y  $r(u_j) \geq r(u_k)$ , entonces  $r(u_i) \geq r(u_k)$ .*
4. *Total: sea  $u_i, u_j \in V(G_R)$  tal que  $u_i \neq u_j$ , entonces  $r(u_i) \geq r(u_j)$  o  $r(u_j) \geq r(u_i)$ .*

**Definición 23** (Competir). *Sea una familia de  $m \in \mathbb{N}$  rankings  $R = \{r_1, r_2, \dots, r_m\}$ , con todos ellos ordenando los mismos elementos de un conjunto  $V(G_R)$ . Dado un par de individuos  $u_i, u_j \in V(G_R)$  con  $i \neq j$ ,  $i, j \leq n$ , se dice que ambos **compiten** si han intercambiado sus posiciones relativas en dos rankings consecutivos. Es decir, si existe  $k < m$  tal que  $r_k(u_i) > r_k(u_j)$  y  $r_{k+1}(u_j) > r_{k+1}(u_i)$ , o viceversa.*

Este concepto de *competencia* es el que va a permitir establecer criterios para construir lo que se llaman grafos de competitividad.

**Observación 9.** *Obviamente, la ordenación que se haga a lo largo de los rankings debe ser bajo un criterio fijo, para que el concepto de competencia se mantenga a lo largo de la familia de rankings.*

Antes de llegar a qué es un grafo de competitividad, hay que definir qué es un grafo ponderado y qué son los pesos de las aristas.

**Definición 24** (Grafo ponderado y Peso de una arista). *Sea  $G = (V(G), E(G))$  un grafo, se dice que este es un **grafo ponderado** si cada arista del conjunto  $E(G)$  tiene un peso asociado.*

*Sea  $a \in E(G)$  una arista del grafo, el **peso** de esta arista es un número asociado a la misma que puede tener cualquier connotación dentro del contexto que rodea al grafo. Si  $a = \{u, v\}$  con  $u, v \in V(G)$ , el peso de  $a$  se denota por  $w_{u,v}$ .*

En este punto, es posible definir qué es un grafo de competitividad.

**Definición 25** (Grafo de competitividad). *Sea  $R = \{r_1, r_2, \dots, r_m\}$  una familia de rankings, con elementos del conjunto  $V(G_R)$ , se define el **grafo de competitividad** como el grafo simple  $G=(V(G_R),E(G_R))$  donde los nodos son los elementos de los rankings y las aristas existen si para cada par de nodos estos han competido entre sí en la familia de rankings  $R$ .*

En otras palabras, el grafo de competitividad permite representar información sobre pares de rankings adyacentes. Cuando el grafo de competitividad se refiere a solamente dos rankings adyacentes, el grafo se llama **grafo de permutación**. Ambos a su vez se engloban en la familia de **grafos de comparabilidad**.

Una evolución del grafo de competitividad surge a partir del siguiente concepto:

**Definición 26** (Competir  $h$ -veces). *Sea  $R = \{r_1, r_2, \dots, r_m\}$  una familia de rankings, con elementos del conjunto  $V(G_R)$ , se dice que dos nodos han **competido  $h$ -veces** si han intercambiado su posición relativa a lo largo de los  $m$  rankings en  $h$  ocasiones.*

**Definición 27** (Grafo de competitividad evolutivo). *Sea  $G_R = (V(G_R), E(G_R))$  un grafo de competitividad. A partir de este se puede generar el **grafo de competitividad evolutivo** como el grafo simple pesado  $G=(V(G_R),E(G_R))$  donde los nodos son los elementos competidores que aparecen en los rankings y cada arista tiene como peso el número de veces que han competido entre sí los correspondientes nodos adyacentes a la arista.*

Formalmente, dados  $u_i, u_j \in V(G_R)$  nodos del grafo, la arista tendrá un peso  $w_{i,j}$  que vendrá dado por el número de veces que los nodos  $u_i, u_j$  compiten a lo largo de la familia de rankings  $R$ .

**Observación 10.** Aunque se ha incluido el término *grafo de competitividad evolutivo*, en este estudio se hablará solamente de *grafo de competitividad*, entendiendo que este es el *evolutivo (ponderado)*. Este término se ha incluido porque puede verse en otras bibliografías como dos grafos diferenciados.

Para clarificar la construcción de un grafo de competitividad, se adjunta el siguiente ejemplo:

**Ejemplo 1.** Sea la familia de rankings  $R = \{r_1, r_2, r_3\}$  con elementos  $V(G_R) = \{u_1, u_2, u_3, u_4, u_5, u_6\}$ , de forma que :

1.  $r_1 = (u_2, u_6, u_3, u_1, u_5, u_4)$
2.  $r_2 = (u_6, u_5, u_3, u_2, u_1, u_4)$
3.  $r_3 = (u_3, u_5, u_1, u_6, u_2, u_4)$

Con estos rankings, se tiene el siguiente grafo de competitividad:

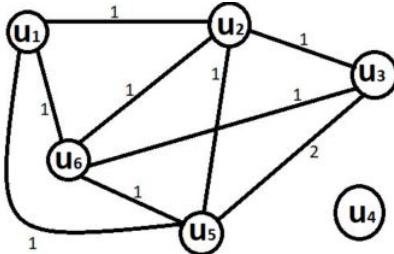


Figura 3.6: Ejemplo de grafo de competitividad

El grafo 3.6 es el grafo de competitividad generado por estos tres rankings, donde se puede observar, entre otras cosas, que los elementos  $u_1, u_3$  no han competido entre ellos, mientras que  $u_3, u_5$  han competido más que el resto. Además, como el elemento  $u_4$  se ha mantenido siempre el último en los rankings, este no compite con ningún otro; es por ello que es un nodo aislado en el grafo.

**Observación 11.** Sea  $m$  el número de rankings de la familia de rankings, entonces el máximo número de veces que puede competir un par de elementos de los rankings, es decir, el peso máximo de la arista entre dos elementos en el grafo de competitividad, es  $m - 1$ .

Antes de pasar a tratar aspectos técnicos de los grafos de competitividad, hay que definir qué es la matriz de adyacencia de un grafo ponderado.

**Definición 28** (Matriz de adyacencia de un grafo ponderado). *Dado un grafo ponderado  $G = (V(G), E(G))$ , se define la **matriz de adyacencia  $M$**  asociada al grafo  $G$  como la matriz cuadrada donde las filas y las columnas hacen referencia a los nodos del grafo de forma que:*

$$M_{ij} = \begin{cases} w_{i,j} & \text{si el nodo } u_i \text{ y el nodo } u_j \text{ compiten} \\ 0 & \text{en otro caso} \end{cases}$$

donde hay que recordar que  $w_{i,j}$  hace referencia al peso de la arista asociada a los nodos  $u_i, u_j$ .

**Nota 10.** Puesto que los grafos con los que se va a trabajar en este estudio son simples y no dirigidos, la matriz de adyacencia es simétrica ( $M_{ij} = M_{ji}$ ) con diagonal nula.

Tomando de nuevo el ejemplo de la 3.6, si se ordenan los nodos en filas y columnas ( $\{u_1, u_2, u_3, u_4, u_5, u_6\}$ ) se tiene la siguiente matriz de adyacencia para

ese grafo:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 2 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 \end{pmatrix} \quad (3.2)$$

Una vez definida la base central de lo que se aplicará a la práctica después, hay conceptos relevantes que pueden ser útiles para la parte posterior del estudio.

**Definición 29** (Conjunto de competitividad de un nodo). *Sea  $R = \{r_1, r_2, \dots, r_m\}$  una familia de rankings, con nodos en  $V(G_R)$ . Dado un nodo  $u \in V(G_R)$ , se define su **conjunto de competitividad** como el conjunto de nodos que compiten con  $u$ , y  $u$ , es decir:*

$$C(u) = \{v \in V(G_R) \mid (u, v) \text{ compiten}\} \cup \{u\}$$

Este último concepto resulta útil para poder identificar los vecinos que tiene en el grafo de competitividad un elemento concreto.

**Definición 30** (Conjunto de competidores). *Sea  $R = \{r_1, r_2, \dots, r_m\}$  una familia de rankings, con nodos en  $V(G_R)$ . Sea  $J \subseteq V(G_R)$ , se denomina **conjunto de competidores** si este es un conjunto maximal que cumple la propiedad de que sus elementos compiten entre sí.*

**Nota 11.** *Hay que recordar que un conjunto maximal, en este caso, es un subconjunto del conjunto de nodos del grafo tal que dados los elementos que contiene no se puede añadir otro elemento más del total de nodos tal que se siga manteniendo la condición de que todos los elementos del subconjunto compiten entre sí.*

Por ejemplo, retomando el ejemplo 3.6, el conjunto  $J=\{u_1, u_2, u_5, u_6\}$  es un conjunto de competidores ya que todos los elementos compiten entre sí y además es maximal, porque no hay otro nodo que se pueda añadir al conjunto de forma que todos sus elementos sigan compitiendo entre sí. Otro ejemplo sería el conjunto  $M=\{u_2, u_3, u_5, u_6\}$ .

**Observación 12.** *A consecuencia de la propia definición de conjunto de competidores, a partir de este conjunto se puede generar un subgrafo completo maximal del grafo de competitividad original.*

**Observación 13.** *Dos elementos compiten entre sí si se encuentran en el mismo conjunto de competidores. Luego, aparece la siguiente condición:*

$$J \subseteq V(G_R) \text{ es un conjunto de competidores} \Leftrightarrow J = \bigcap_{u \in J} C(u)$$

Otro aspecto que puede resultar interesante estudiar es cuando dos elementos están compitiendo indirectamente. Aparece así el término de *competir eventualmente*:

**Definición 31** (Competir eventualmente). *Sea  $R = \{r_1, r_2, \dots, r_m\}$  una familia de rankings con nodos en  $V(G_R)$ . Sean  $u, v \in V(G_R)$ , se dice que  $u$  y  $v$  **compiten eventualmente** si existe  $p \in \mathbb{N}$  y nodos  $x_1, \dots, x_p \in V(G_R)$  tal que  $\{u, x_1\}, \dots, \{x_{p-1}, x_p\}, \{x_p, v\}$  compiten.*

Siguiendo la línea de las definiciones anteriores, aparece el siguiente concepto:

**Definición 32** (Conjunto de competidores eventuales). *Sea el subconjunto  $J \subseteq V(G_R)$ , se dice que este es un **conjunto de competidores eventuales** si es un conjunto maximal donde todos sus elementos compiten eventualmente entre sí.*

Volviendo al ejemplo 3.6, se tiene que el conjunto  $V(G_R) \setminus \{u_4\}$  es un conjunto de competidores eventuales, ya que es un subconjunto del total, todos sus elementos compiten eventualmente entre ellos y es maximal.

**Observación 14.** *Por definición, competir implica, claramente, competir eventualmente.*

*Como ocurría en el caso de competir, dos nodos compiten eventualmente si y solo si pertenecen al mismo conjunto de competidores eventuales. Si no, ni siquiera podrán competir entre ellos.*

*Además, dos nodos compiten eventualmente si y solo si existe un camino entre ellos en el grafo de competitividad.*

### Coeficiente para el estudio de los grafos de competitividad

Una vez se tiene el grafo de competitividad en función de una familia de rankings, se emplean una serie de parámetros para poder estudiarlo y extraer conclusiones.

Para este estudio, se va a emplear el que se conoce como coeficiente de correlación Kendall  $\tau$ .

**Definición 33** (Coeficiente de correlación de Kendall  $\tau$ ). *Sea  $V(G_R)$  el conjunto de nodos con  $|V(G_R)| = n$ , ordenados en dos rankings  $r_1$  y  $r_2$ . Se define el **coeficiente de correlación de Kendall** como el cociente de la diferencia de los pares de nodos que compiten y no compiten en esos dos ranking entre el número de posibles pares que compitan en esos dos mismos rankings. Matemáticamente, resulta la siguiente expresión:*

$$\tau(r_1, r_2) = \frac{\bar{K}(r_1, r_2) - K(r_1, r_2)}{\binom{n}{2}}$$

donde  $\bar{K}(r_1, r_2)$  es el número de pares de nodos que no compiten con respecto de  $\{r_1, r_2\}$  y  $K(r_1, r_2)$  es el número de pares de nodos que sí compiten respecto de ambos rankings.

**Observación 15.** Claramente, como los pares posibles a competir se forman por dos elementos, sobre un total de  $n$  elementos las posibles combinaciones vienen dadas por el número  $\binom{n}{2}$ .

De la definición y esta última observación, se puede establecer una expresión equivalente para calcular el coeficiente de Kendall.

**Observación 16.** Sean los rankings  $r_1$  y  $r_2$  los de la definición del coeficiente de Kendall, formando la familia de rankings  $R = \{r_1, r_2\}$ . Sea  $G_R$  el grafo de competitividad (grafo simple) generado a partir de  $R$ . Se tiene entonces que:  $K(r_1, r_2) = |E_R|$  y que  $\bar{K}(r_1, r_2) = \binom{n}{2} - |E_R|$ , luego se tiene que:

$$\tau(r_1, r_2) = \frac{\binom{n}{2} - |E_R| - |E_R|}{\binom{n}{2}} = 1 - \frac{2|E_R|}{\binom{n}{2}} = 1 - \frac{4|E_R|}{n * (n - 1)}$$

Según se sabe, el coeficiente de correlación Kendall  $\tau$  toma valores entre -1 y +1, y en consecuencia de este resultado se puede afirmar que:

- Valores para el coeficiente de correlación de Kendall cercanos al +1 en una familia de rankings indican que estos presentan una alta correlación.
- Cuanto más alejado del valor +1 se encuentre el valor del coeficiente de Kendall, mayor es la diferencia entre dichos rankings.

En este sentido, cuando se trabaje con varias familias de rankings, como pasará después en la parte práctica, será útil la representación de los valores obtenidos para el coeficiente de correlación Kendall a lo largo de los pares de rankings continuos de las familias de rankings de forma que con todos estos pares superpuestos se pueda visualizar qué familias presentan una mayor o menor correlación entre sus elementos.

### 3.2. Detección de comunidades: Algoritmo de Girvan-Newman

Para la parte práctica de este estudio, una vez se hayan revisado los conceptos anteriores, se van a generar varios grafos de competitividad, los cuales irán asociados a distintas puntuaciones. A estos se les va a aplicar el algoritmo de

### 3.2. Detección de comunidades: Algoritmo de Girvan-Newman

---

Girvan-Newman que va a permitir establecer comunidades entre los diferentes valores del IBEX 35.

La detección de comunidades es una tipología de problema empleada en distintos ámbitos como la medicina, por ejemplo, en el funcionamiento de las redes neuronales, en la sociología para la detección de comunidades dentro de una red social, y un largo etcétera.

Con el fin de entender la base teórica de esta idea, lo primero es definir qué es una comunidad para entender su importancia en el estudio.

**Definición 34** (Comunidad). *Una **comunidad** dentro de un grafo es un conjunto de nodos caracterizados porque el número de conexiones entre los elementos de la comunidad es muy superior al número de conexiones que tienen con el resto de nodos del grafo.*

Para la detección de comunidades existen numerosos algoritmos como son:

- Método del corte mínimo: Establece divisiones del grafo con un número de nodos aproximado en cada una de ellas basándose en la idea de elegir grupos con un número mínimo de aristas. Un ejemplo está en la bibliografía [19]
- Algoritmo de k-medias: Permite dividir el conjunto total en k grupos basándose en una idea de distancias entre los nodos. Un ejemplo se encuentra en [21].
- Algoritmo de maximización de modularidad: Este algoritmo se basa en encontrar aquellas particiones del grafo con valores altos de modularidad, que es una función de medida de la calidad de una partición dentro del grafo. [18]

Entre otros, y cada uno de ellos se centra en una idea distinta para la identificación de las comunidades. El motivo de este abanico para la identificación de comunidades es que no existe un criterio definido perfecto y único para determinar qué característica hace que un nodo esté en una comunidad o en otra. Además, son múltiples los modelos que se basan en la idea de que un mismo nodo puede pertenecer a diferentes comunidades, lo que se conoce como superposición, aunque para este estudio no se tratará ninguno de esta tipología.

Sin embargo, para este análisis, se ha escogido el algoritmo de Girvan-Newman debido a que es uno de los más utilizado. En numerosas referencias es considerado como el algoritmo óptimo para la resolución de este tipo de problema.

Para esta sección se han tomado los siguientes artículos, papers y libros: *Community structure in social and biological networks*[14], *Algoritmos de detección de comunidades en grafos y aplicación en conjuntos de datos de tránsito en aeropuertos y trasvase de alumnos entre titulaciones de la UPV*[18], *Finding and*

*evaluating community structure in networks* [17] y *Revising the Newman-Girvan algorithm* [7].

El **algoritmo de Girvan-Newman** (2002) es una técnica para la detección y el análisis de comunidades dentro de un grafo. Este algoritmo fue resultado de los estudios de Michelle Girvan y Mark Newman. Además de establecer comunidades dentro de un grafo estudiaron otras propiedades de estos como, por ejemplo, vieron que se podía calcular la probabilidad de relación entre dos nodos que tenían un nodo adyacente en común.

Se trata de un algoritmo jerárquico ya que su objetivo es establecer una jerarquía entre las diferentes comunidades que se van formando a lo largo de sus iteraciones, con distintos niveles de profundidad. Esto se puede observar de manera muy intuitiva mediante un tipo de representación llamada dendograma, que se verá posteriormente.

Pero lo que concierne en este estudio es el conocer cómo establecer comunidades entre los nodos de un grafo, es decir, aquellos conjuntos de nodos donde sus elementos tienen conexiones más densas que con elementos de fuera de su comunidad. Además de establecer comunidades se pueden establecer metacomunidades, agrupaciones de comunidades, y establecer jerarquías para agrupaciones de agrupaciones y otros conceptos que quedan fuera del alcance de los objetivos de este estudio.

El algoritmo se basa en la eliminación progresiva de las aristas con mayor coeficiente de intermediación (betweenness) del grafo original. Así, en las iteraciones siguientes, el algoritmo va generando las comunidades, que serán aquellos nodos que siguen conectados tras la eliminación progresiva de las aristas. A modo de visión general, el algoritmo se centra en mantener aquellos enlaces que son más probables entre comunidades y no, como otros algoritmos, en encontrar los nodos centrales de las comunidades (como podría ser con los centroides de un clustering).

Existen dos enfoques para la eliminación de elementos del grafo:

1. La intermediación de nodos, que se basa en la siguiente idea:
  - i) Sea el nodo  $u \in V(G)$  un nodo del grafo, se toman pares de nodos del conjunto  $V(G) \setminus \{u\}$  y se calculan sus caminos más cortos entre pares (problema del camino mínimo entre nodos).
  - ii) Con todos los caminos calculados, la intermediación del nodo  $u$  es el número de caminos seleccionados en el paso anterior que pasan por el nodo  $u$ .
  - iii) Repetir los pasos i) y ii) con el resto de nodos del conjunto  $V(G)$  para obtener la intermediación de todos los vértices.

Esta medida sobre cada nodo del grafo sirve para cuantificar la centralidad y la influencia que tiene cada uno de ellos sobre el flujo de información dentro del grafo entre el resto de nodos.

2. La intermediación de aristas, que se basa en:

- i) Sea  $a \in E(G)$  una arista del grafo, se toman pares de nodos del conjunto  $V(G)$  y se calculan los caminos más cortos entre ellos (problema del camino mínimo entre nodos).
- ii) Con todos los caminos mínimos calculados, la intermediación de la arista  $a$  es el número de caminos seleccionados en el paso anterior que pasan por dicha arista  $a$ .
- iii) Repetir los pasos i) y ii) con el resto de aristas del conjunto  $E(G)$  para obtener la intermediación de todas las aristas.

En este caso la idea es, análogamente, calcular la influencia de cada arista sobre el flujo de información del grafo.

Para el cálculo de la intermediación puede ocurrir que haya más de un camino con la característica de ser el más corto entre un par de nodos. Para resolver el conflicto, a cada camino se le asigna la misma ponderación que el resto, de forma que entre todas las ponderaciones de todos los caminos mínimos sumen uno.

El objetivo de esto es conseguir que si las comunidades están unidas por aristas con poco peso, entonces los caminos mínimos pasarán por estas. El hecho de que estas aristas sean comunes a muchos caminos mínimos hace que tengan una elevada intermediación y serán las que se eliminan del grafo en las sucesivas iteraciones del algoritmo. Esto se explicará a continuación.

El concepto anterior, intermediación en este estudio, es frecuentemente llamado en numerosas bibliografías como betweenness, su nombre en inglés aunque en muchos textos lo mantienen. Con esto explicado, se puede exponer el algoritmo de Girvan-Newman. Los pasos que sigue son los siguientes:

1. Calcular la intermediación de todas las aristas que hay en el grafo.
2. Eliminar la arista que tenga la intermediación más alta. En caso de empate entre aristas, se toma una de ellas al azar.
3. Para el resto de aristas que queden en el grafo, calcular su intermediación.
4. Repetir los pasos 2 y 3 hasta que no quede ninguna arista.

Una vez establecido el algoritmo, se escribe su pseudocódigo [1] para una posible implementación futura.

**Algoritmo 1** Algoritmo de Girvan-Newman

---

```

repeat
     $n = \text{Total number of edges in graph } G$ 
    for  $i = 0$  to  $n - 1$  do
         $\text{Betweenness}[i] = \text{Betweenness of edge } i$ 
        if  $\text{Betweenness}[i] > \text{Max_Betweenness}$  then
             $\text{Max_Betweenness} = \text{Betweenness}[i]$ 
             $\text{Max_Betweenness_Edge} = i$ 
        end if
    end for
    remove edge Max_Betweenness_Edge from graph G
until  $n = 0$ 

```

---

Para clarificar cómo funciona el algoritmo, se va a simular una iteración del mismo en un ejemplo. Supongamos que se tiene el grafo 3.7 donde todas las aristas tienen el mismo peso, por ejemplo, la unidad. En la figura izquierda se ha calculado la intermediación de todas las aristas. Claramente, la arista con mayor intermediación tenía que ser la central debido a que para realizar cualquier camino entre ambas zonas del grafo hay que pasar siempre por dicha arista.

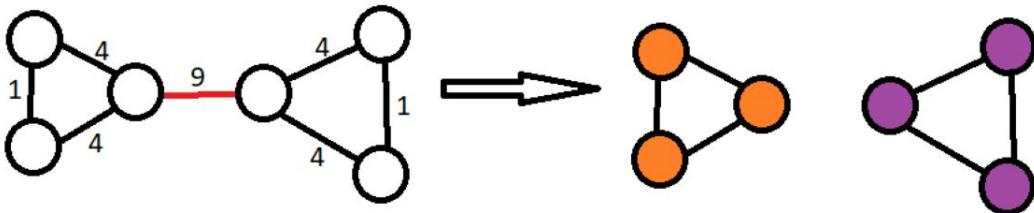


Figura 3.7: Ejemplo de iteración del algoritmo de Girvan-Newman

Una vez se tiene la arista con mayor intermediación, se elimina dicha arista y así surgen las dos primeras comunidades, como se observa en la parte derecha de 3.7.

Para el cálculo de la intermediación de las aristas, Newman desarrolló otro algoritmo que resuelve este problema para un conjunto de  $k$  aristas sobre  $n$  nodos con una complejidad  $O(kn)$ . Ahora bien, como este proceso ha de calcularse para eliminar las  $k$  aristas, el algoritmo de Girvan-Newman recibe una complejidad de  $O(k^2n)$ . En todo caso, como a medida que se van eliminando aristas hay que calcular la intermediación sobre un número menor de aristas, el tiempo de resolución del algoritmo podrá ser menor. Además, el tiempo de ejecución también dependerá de la predisposición que presente el grafo para la división en comunidades, ya que habrá grafos más densos respecto a otros.

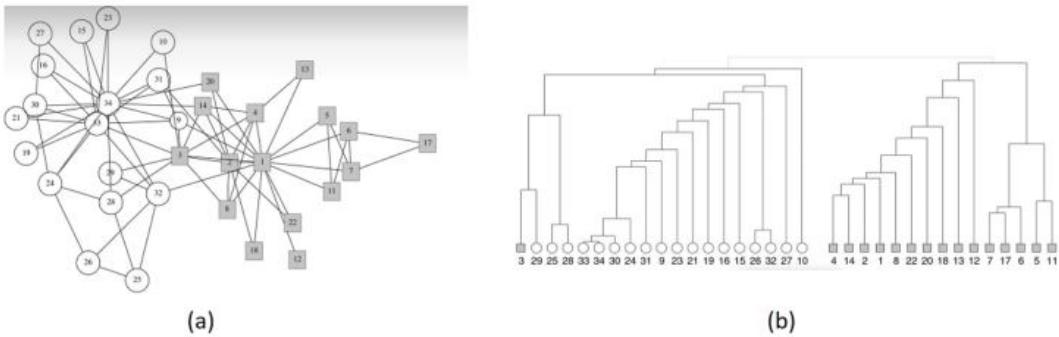


Figura 3.8: Ejemplo aplicación del algoritmo de Girvan-Newman al problema del club de kárate de Zachary

Los autores del algoritmo plantean una observación para el cálculo de la intermediación de las aristas: podría ser tentativo calcular solamente en un primer paso la intermediación de todas las aristas y sobre esos valores ir eliminándolas a lo largo del algoritmo sin recalcular la intermediación de cada una, pero esto no contemplaría todos los casos ya que a medida que se van eliminando aristas la betweenness de las restantes pueden cambiar.

A lo largo de las iteraciones sucesivas que hace el algoritmo se van haciendo las separaciones entre nodos y se van generando las comunidades. Por lo que en iteraciones primeras el número de comunidades será muy bajo y en las últimas iteraciones las comunidades estarán formadas, casi todas, por un único nodo.

Tras la aplicación del algoritmo, la información tratada puede representarse de dos formas:

- Simultáneamente al desarrollo de las iteraciones se pueden representar las comunidades que existen en el grafo original mediante agrupaciones de nodos o mediante una representación por coloración, o similares, de forma que los nodos de cada comunidad se diferencien.
- Tras la última iteración puede representarse un dendograma donde se ven reflejadas todas las iteraciones del algoritmo. En este dendrograma la idea es ir separando, mediante la eliminación progresiva de aristas, nodos del conjunto total de forma que se vayan generando las comunidades. De esta manera, al finalizar las iteraciones, que corresponderá con la parte final del dendrograma, todas las hojas del gráfico serán las nodos divididos individualmente.

Un ejemplo de aplicación de este algoritmo es [3.8](#), que se puede encontrar más detallado en [\[14\]](#).

Este ejemplo de grafo fue desarrollado por Wayne W. Zachary para mostrar cómo 34 miembros de un club de kárate se separan en dos grupos distintos. El origen de estos dos grupos es un desacuerdo entre el administrador del club y un profesor que terminaría en la apertura de otro club por parte de este último. Aunque en el problema original se utilizasen otras medidas para la determinación de los grupos, Girvan y Newman utilizaron este ejemplo real para conocer si su algoritmo se adecuaba a la realidad.

Los resultados obtenidos al aplicar el algoritmo en este ejemplo fueron:

- (a) Representación del grafo con los 34 individuos agrupados en dos comunidades.
- (b) Dendograma de la separación en comunidades a lo largo de las iteraciones del algoritmo.

### 3.3. Conceptos del contexto económico

La finalidad de esta sección es dar contexto a todos los conceptos que en la práctica posterior van a ser tratados. En primer lugar, hay que saber qué son las acciones y explicar un poco acerca de ellas. Para ello, se ha recurrido a diferentes fuentes. [15][16]

**Definición 35** (Acción). *Una **acción** de una empresa se define como un activo que representa una parte de la propiedad de dicha empresa. Los beneficios que tiene la persona, física o jurídica, que posee acciones de una entidad son beneficios de la misma proporcionales a su porcentaje de propiedad.*

El origen de las acciones está en que las empresas puedan financiarse a través de la venta de una parte de la propiedad de la misma. Con ella, los dueños originales pierden parte de propiedad pero ganan presupuesto que se podrá destinar a proyectos u otras operaciones, por ejemplo. Todas estas acciones por parte de entidades empresariales están reguladas por multitud de entidades gubernamentales superiores como la CNMV, en España, o la AEVM, en la Unión Europea.

Cuando una empresa decide "salir a bolsa", a través de un proceso previo se determina un precio inicial, estudiando diferentes factores, para cada una de las acciones que emite. Una vez se ponen en circulación, el precio de dichas acciones comienza a fluctuar y a variar en el tiempo debido a innumerables sucesos que hacen que suban o bajen de precio. Aparece entonces el siguiente concepto.

**Definición 36** (Precio de cotización). *Dentro de un mercado financiero, el **precio de cotización** de un producto financiero es el precio en el que se realizó la última transacción para ese producto.*

Inherentemente, aparece lo que sí se empleará en la práctica que es el **precio de cotización al cierre de la sesión**, el cual, como su propio nombre indica, se refiere al último precio de negociación de ese activo en esa sesión.

Más conceptos útiles en este contexto es saber qué es una **sesión**. La bolsa, como lugar donde se comercializa con los productos financieros, tiene un horario y un calendario asociado fuera de los cuales no es posible operar con tales productos. Es precisamente la comercialización en un determinado día y cumpliendo un horario lo que se conoce como sesión.

Como el precio de cotización no es fijo en el tiempo, aparece el concepto de volatilidad.

**Definición 37** (Volatilidad). *La volatilidad dentro de un mercado financiero se define como la medida de la variación de los precios de cotización de un producto financiero concreto. Esta medida tiene en cuenta la frecuencia de esos cambios y si han sido grandes o pequeños. Hay que puntualizar que pueden ser movimientos alcistas o bajistas, indistintamente.*

El concepto de volatilidad [20] es útil para comparar varios productos financieros. Así, se dirá que uno es más volátil que otro si los precios del primero varían más que para el segundo en el mismo período temporal.

La volatilidad es uno de los aspectos más estudiados en este contexto debido a que guarda una estrecha relación con los precios de cotización de los productos, lo que permitirá actuar sobre ellos para obtener beneficios.

Ciento es que en varias fuentes definen la volatilidad estrictamente a partir de la desviación estándar como un concepto plenamente estadístico pero, como se va a ver en este trabajo, no siempre la única forma de afrontar el estudio de la misma es a través de herramientas estadísticas.



# 4

## Desarrollo del estudio

En este apartado se va a desarrollar la parte práctica de este estudio. Una vez se han establecido los conceptos teóricos que fundamentan esta parte, es momento de ponerlos en práctica. La idea es explicar en este apartado los puntos más relevantes para entender cómo se ha llevado a cabo la práctica. Adicionalmente a esta sección, se puede consultar el archivo .py en el apéndice E, el cual recoge todos los archivos relativos al desarrollo práctico de este estudio.

Para el desarrollo del código se ha utilizado el lenguaje de programación Python en la herramienta Jupyter Notebook.

Dados los objetivos que se vienen comentando a lo largo de todo este escrito, es necesario, en primer lugar, saber cómo se van a obtener los datos, de qué tipo son y su origen. Para estas cuestiones aparece la primera subsección.

Seguido del primer punto, aparece la explicación de lo que en este trabajo se llaman *perspectivas*, las cuales permiten dar diferentes enfoques al estudio, generando así diferentes grafos de competitividad con distintas relaciones entre los nodos. Estas visiones es posible compararlas desde el punto de vista de la correlación de su familia de rankings. Para ello, se utiliza el coeficiente de correlación de Kendall explicado anteriormente.

Por último, para finalizar la práctica se aplica el algoritmo de Girvan-Newman a todas estas perspectivas y sus subvariantes, que más tarde se explicarán.

## 4.1. Ingesta de datos

Para la obtención de los datos acerca de las cotizaciones de las 35 empresas que forman el IBEX 35 se ha tomado como referencia la web oficial de Bolsas y Mercados Españoles [12]. Esta fuente de datos permite conocer el histórico de cualquier empresa que cotice en el IBEX 35 además de una ficha propia para cada una de ellas donde se encuentra toda la información relativa a la misma.

Con ello, para poder ingestar los datos, en primer lugar para cada una de las 35 empresas se ha tenido que filtrar las fechas tomando los últimos 6 meses desde la ingesta y, puesto que dicha ingesta tuvo lugar el 23 de octubre de 2022, la información que se tiene es desde la sesión del 22 de abril de 2022 hasta la sesión del 21 de octubre de 2022, ambas incluidas.

A continuación, estas tablas de información se han recogido en un archivo .csv para su posterior subida al entorno de trabajo. Así, por cada empresa se tiene un archivo .csv.

Para entender mejor qué información se está tomando de cada empresa se ha adjuntado un ejemplo de Inditex B.1 que permite extraer la información para el resto de empresas.

Puesto que el estudio se va a realizar trabajando con el precio de la acción al cierre de la sesión, la información que se necesita para el desarrollo de la práctica es la relativa a las columnas: Fecha, referente a la fecha de la sesión que se registra, y Cierre, que registra el precio de la acción al cierre de la sesión.

El resto de columnas se refieren a diferentes precios en la acción con los cuales no se va a trabajar. Además, se puede observar que se tiene información sobre un total de 131 sesiones.

Para realizar esta importación se ha trabajado con Dataframes, objetos de la librería *pandas*, los cuales permiten un fácil almacenamiento de datos de estas características.

Posterior a la ingesta de todos estos datos, se construyeron variables para almacenar en una lista los nombres de las empresas que se iban a emplear y el número de empresas y de rankings que se necesitarán posteriormente en el estudio.

## 4.2. Desarrollo de las diferentes perspectivas del estudio

Para llevar a cabo la construcción de diferentes grafos de competitividad se han desarrollado distintas ideas para la generación de sus ranking asociados. De aquí a continuación para referirse a cada una de ellas se hará a través de los conceptos: perspectivas, visiones, enfoques o subvariantes.

Previamente a la explicación y desarrollo de los tres enfoques hay que aclarar que la estructura establecida para los rankings es de dos columnas donde la primera almacena los nombres de las empresas y la segunda su valor ordenado/a ordenar en el ranking asociado a dicha empresa.

La primera perspectiva se basa en que bajo una inversión inicial de 1000 € en cada empresa, se ha obtenido el valor de la inversión realizada al final de cada sesión y con el precio de cierre de cada sesión se ha ido construyendo un ranking según el valor que iba tomando cada una de dichas inversiones.

Para entender mejor esta idea, en [B.2](#) se tiene el primer ranking creado asociado a la perspectiva 1. La idea es que esos 1000 € iniciales han pasado a tener otro valor y es con este con el que se construyen y ordenan los rankings.

En relación a la situación anterior, aparece la segunda perspectiva, la cual emplea la variación (ganancia o pérdida) de la inversión para cada valor para construir la familia de rankings, es decir, se toma la diferencia entre valores consecutivos y no el valor en sí. Debido a esto, la visión también puede verse como la que estudia la **volatilidad** ya que trabaja con cuánto ha variado el valor de una sesión a otra. Para ilustrar la idea, se adjunta el primer ranking construido para este enfoque en [B.3](#).

La explicación de por qué tomar dos medidas sobre el mismo origen, que es la inversión inicial de 1000 € en cada empresa, es que en este segundo caso va a existir mayor competitividad puesto que empresas que hayan tenido tendencias contrarias a lo largo de varias sesiones seguidas distarán mucho en cuanto a sus valores pero puede que sí compitan observando el incremento o decremento del mismo.

Por último, aparece la tercera perspectiva, la cual trabaja también con el concepto de **volatilidad** pero a través de la evaluación del porcentaje de ganacia o pérdida entre cada par de sesiones contiguas. Es decir, tomando como referencia el valor de la empresa al cierre de la sesión anterior, se evalúa el porcentaje de subida o bajada en la siguiente sesión. Nuevamente, se adjunta el primer ranking relativo a este último enfoque en [B.4](#).

La segunda y la tercera perspectiva tienen una relación muy estrecha debido a que en ambas se estudia la volatilidad pero en cada una de ellas los valores son

distintos.

Una vez explicadas las perspectivas, hay que tratar las funciones que se han implementado. Por un lado, se ha definido la función *ordenarRanking*. Esta toma como argumento un DataFrame con dos columnas: la primera con el nombre de las empresas y la segunda con sus valores asociados. Esta estructura es la que se puede visualizar en [B.2](#) y solamente se ordenarán aquellos rankings que la respeten. La función *ordenarRanking* [A.1](#) es una adaptación del método de ordenación **Selection sort**, que ordena las empresas y sus valores asociados en orden decreciente.

Puesto que se tienen registradas 131 sesiones y la primera no se emplea para la creación de un ranking específico, se han tenido que construir para cada perspectiva 130 rankings.

Para escalabilizar todo ello, el método ha sido generar dataFrames para cada perspectiva de forma que contuviesen 35 filas relativas a cada una de las empresas que se ordenan en cada ranking y 130 columnas, una por cada ranking construido. Un ejemplo relativo a la perspectiva 1 se ve en [B.6](#), el código, y [B.7](#), el resultado, la matriz de rankings. Para la construcción de estos DataFrame, se ha realizado lo siguiente en cada enfoque:

1. Mediante un bucle *for* se han recorrido todas las sesiones registradas.
2. En cada iteración, para cada sesión se ha creado el ranking asociado y se ha almacenado en una variable auxiliar. Para crear cada ranking se ha tenido que almacenar la información del ranking de forma que se pueda aplicar la función *ordenarRanking*.
3. Se aplica a cada ranking, que se almacena en cada iteración en la variable auxiliar, la función *ordenarRanking* para ordenarlo.
4. Por último, se toma simplemente la columna con los nombres de las empresas ordenados, desechando el valor utilizado para la ordenación, y se almacena en la matriz de rankings.

Con estas construcciones, se obtiene tres dataFrames de información que serán útiles para las construcciones de las matrices de adyacencia posteriores.

A modo de comprobación, puesto que al inicio de cada perspectiva se crearon el primer, el segundo y el último ranking, se ha ejecutado un bloque de código donde se comparan las dos primeras columnas y la última de estas tablas de rankings con ellos, de forma que se pueda conocer de forma automática si son idénticos o no, en cuyo caso se puede saber si se ha escalado bien el modelo.

El siguiente paso es emplear los dataFrames de los rankings para construir las matrices de adyacencia de cada una de las perspectivas. Para ello, hay que conocer

si dadas dos empresas y dos rankings, contiguos por la naturaleza del problema, estas compiten entre sí. Es por ello que se ha implementado la función *seCruza*, definida en [A.2](#). Esta función toma dos *dataFrame* de datos, los cuales serán dos columnas de rankings, y dos nombres, que en este caso serán dos nombres de empresas. Dicha función se basa en encontrar las posiciones en las que se encuentra cada empresa en cada ranking y así poder compararlas para saber si la que en el primer ranking estaba primera ahora está segunda, o viceversa, es decir, si sus posiciones relativas se han intercambiado y por tanto las empresas compiten.

Finalmente, para obtener las matrices de adyacencia, en primer lugar, se construyen *dataFrames* a modo de matrices cuadradas 35x35 donde cada columna y cada fila hace referencia a una empresa concreta. Para dar entrada a estas empresas, se recurre a la lista inicial donde estaban almacenados todos los nombres de las empresas ordenados alfabéticamente. Esta construcción permite interpretar que en el cruce de la empresa de la fila *i* con la de la columna *j* se encuentra el número de veces que han competido estas empresas a lo largo de la familia de rankings.

Estas matrices presentan la característica de ser simétricas y con diagonal nula, como ya se explicó anteriormente en los fundamentos teóricos. Un ejemplo para la perspectiva 1 está en [B.8](#), el código, y [B.9](#), el resultado.

Bajo esta estructura, para cada perspectiva se recorren los *dataFrame* de rankings mediante bucles *for* y para cada par de rankings contiguos se recorre toda la lista de empresas tomando todos los pares posibles. Con ello, se evalúa si bajo ese par de rankings ese par de empresas compiten y en caso afirmativo se suma 1 en las posiciones  $[i,j]$  y  $[j,i]$  de la matriz, entendiendo *i* como la fila y *j* como la columna.

Con esta última parte se obtienen las matrices de adyacencia de las tres perspectivas anteriores, las cuales que permitirán la representación de los grafos de competitividad asociados.

Para la representación de los grafos se han ido probando diferentes modalidades de posicionamiento de los nodos de la librería *networkx* y finalmente en todos los grafos representados se ha escogido la disposición circular mediante la sentencia *circular\_layout(G)*. El objetivo de este punto es optimizar la colocación de los elementos del grafo para que se pudiese observar correctamente. Un ejemplo de este código relativo al grafo de la perspectiva 1 está en [B.10](#), extrapolable al resto de las representaciones. Además, como se puede ver en dicho ejemplo, las aristas no se representan con su peso asociado, si no con un grosor proporcional a su peso.

Los resultados obtenidos en los grafos de estas tres perspectivas, que se pueden observar en el GitHub del anexo [E](#), manifiestan que no es posible obtener con-

clusiones claras de ellos debido a la alta competitividad que presentan los nodos entre sí. Es por ello que aparece la necesidad del desarrollo de las subvariantes, las cuales serán explicadas en el siguiente subapartado.

Antes de terminar este bloque hay que añadir la creación de dos funciones que han sido útiles para este estudio.

La primera es la función *insertarColumnaDiferencia* que toma un DataFrame con la estructura de las cuatro primeras columnas de [B.3](#) y calcula una quinta con la diferencia entre la columna 4 y la columna 3 por fila. Esto será útil porque es el valor que se va a introducir en el ranking, la diferencia de la inversión entre las sesiones. Esta función, ilustrada en [A.3](#) se emplea cada vez que se quiere generar un ranking para poder crearlos con la estructura definida anteriormente.

Por último, también se generó una función para incluir una columna con la diferencia porcentual entre las sesiones en los DataFrame de las empresas. Su nombre es *insertarColumnaVariacion A.4*. Básicamente, ejecuta la diferencia porcentual definida anteriormente entre el precio de la sesión actual y de la sesión anterior y lo rellena para todas las sesiones. Obviamente, para la primera sesión la rellena con un 0, referente a que no hay información de la sesión anterior. Lógicamente, esta función se ha tenido que aplicar a los 35 casos de las empresas para completar su información.

#### 4.2.1. Desarrollo de las subvariantes para las perspectivas anteriores

Observando los resultados obtenidos en los grafos de las tres perspectivas anteriores se ve que en las tres casuísticas los grafos presentan un alto volumen de aristas lo que significa que presentan una alta competitividad. Esto implica que cuando se vaya a aplicar el algoritmo de Girvan-Newman no se van a obtener resultados que permitan extraer conclusiones claras para el estudio.

Para ello, la alternativa escogida es tomar de cada grafo solo aquellas aristas que presenten pesos mayores que cierta cantidad umbral, de forma que solo queden en el grafo aquellas aristas que aporten mayor información. Durante el desarrollo de la práctica se ha llegado a tres valores umbrales:

- Media aritmética de aquellos pesos mayores que cero. En primer lugar, se ha creado una lista, recorriendo la mitad superior derecha de la matriz de adyacencia para no repetir aristas, con los pesos de las aristas mayores que cero. Una vez se tenían todos los valores recogidos, se ha calculado la media a través de la función *mean* de la librería *numpy*.
- Medida 1. Para este caso se ha tomado el valor intermedio entre la media y el peso máximo en la matriz de adyacencia.

- Medida 2. Para el último límite se ha tomado un valor un poco mayor que el valor intermedio entre la media y el peso máximo en la matriz de adyacencia. Es decir, si la medida 1 estaba en la posición relativa 0.5, esta medida está en la 0.6.

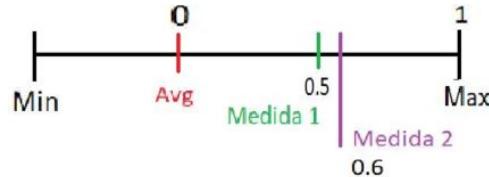


Figura 4.1: Representación de las tres medidas umbral

Estos valores se aplican para cada una de las tres perspectivas, resultando 3 subvariantes para cada enfoque. En 4.1 puede observarse como la primera que se calcula es la media (AVG en el gráfico) y a partir de esta, que se fija como valor más bajo, se calcula la medida 1 y la medida 2.

El origen de las dos medidas últimas es el hecho de que al aplicar la media como valor límite aún el volumen de aristas era muy elevado. Con la medida 1 ya para las perspectivas 1 y 3 el grafo es bastante ligero de aristas. Sin embargo, como en la perspectiva 2 aún había bastantes aristas, se estableció la medida 2.

La medida 2 está en la posición relativa 0.6 pero se probó en diferentes localizaciones, incluso en la 0.9. Finalmente se estableció la que se ha tomado porque con una posición más alta se simplificaba demasiado la información que recogían los grafos.

Todos estos grafos generados, así como los originales, serán empleados para la aplicación del algoritmo de Girvan-Newman sobre los mismos, con el fin de analizar el comportamiento de este algoritmo sobre ellos.

Adicionalmente, todos los grafos generados en esta sección se pueden visualizar en el anexo de GitHub [E](#).

### 4.3. Estudio del coeficiente de correlación de Kendall

Previamente a la aplicación del algoritmo de Girvan-Newman, se recurre al estudio del coeficiente de correlación de Kendall para las tres perspectivas definidas anteriormente.

Rescatando la definición para el algoritmo de Girvan-Newman de los fundamentos teóricos, se ha implementado la siguiente función. Su nombre es *coefKen-*

*dall* y se basa en recoger dos rankings, dataFrames, y una lista de nombres, en este caso de empresas, e ir evaluando si para cada par de empresas estas compiten o no, en cuyo caso se lleva un contador u otro que recoge ambas casuísticas. Con ambos valores, se calcula el coeficiente de correlación de Kendall para estos dos rankings.

Como se puede observar en [A.5](#), la función calcula el valor del coeficiente para dos rankings concretos, por lo que para calcularlo de todos los rankings consecutivos de las tres perspectivas hubo que iterar mediante un bucle *for* para recorrer todos los dataFrames de rankings de cada uno de los tres enfoques.

Los resultados de esta parte del estudio se podrán ver en el siguiente capítulo de *Resultados* ([5](#)).

## 4.4. Aplicación del algoritmo de Girvan-Newman

Este último apartado es el más importante a la hora de poder extraer las conclusiones del estudio. Tras el desarrollo anterior, se han originado 12 grafos los cuales hay que analizar aplicándoles el algoritmo de Girvan-Newman.

Para la evaluación del algoritmo en todas las variantes y subvariantes, se ha decidido iterar el algoritmo como en [4.2](#) para todas ellas. De esta forma, se puede visualizar la evolución del algoritmo a lo largo de las iteraciones y cómo se van formando las comunidades. Concretamente, para todos se han establecido 7 iteraciones con el fin de que las comunidades que resulten sean más o menos informativas para extraer conclusiones.

```
#Con este bloque se consiguen las k primeras iteraciones del algoritmo
list_communities_inv = []
n_it_inv = 7
comp_inv = community.girvan_newman(G_inv)
for communities in itertools.islice(comp_inv, n_it_inv):
    print(tuple(sorted(c) for c in communities))
    list_communities_inv.append(tuple(sorted(c) for c in communities))
```

Figura 4.2: Bloque de código de la primera perspectiva para las iteraciones del algoritmo de Girvan-Newman

Para cada una de estas iteraciones se ha hecho una representación de la situación del grafo, coloreando los nodos según la comunidad a la que pertenezcan.



# 5

## Resultados

El objetivo de este capítulo es comentar los resultados obtenidos con el coeficiente de correlación de Kendall y los obtenidos tras la aplicación de el algoritmo de Girvan-Newman a los 12 grafos creados. Esto permitirá una extracción de conclusiones más sencilla.

### 5.1. Estudio del coeficiente de correlación de Kendall

Tras la ejecución de los bloques de código relativos al coeficiente de correlación de Kendall, se ha representado la evolución del mismo a lo largo de los pares de rankings consecutivos [5.1](#).

La situación es que la perspectiva 1 presenta valores muy elevados, cercanos a 1 en muchas ocasiones, y las perspectivas 2 y 3 se mueven de forma prácticamente conjunta sobre los mismos valores, 0.5 como máximo y -0.6 como mínimo, oscilando entre ellos.

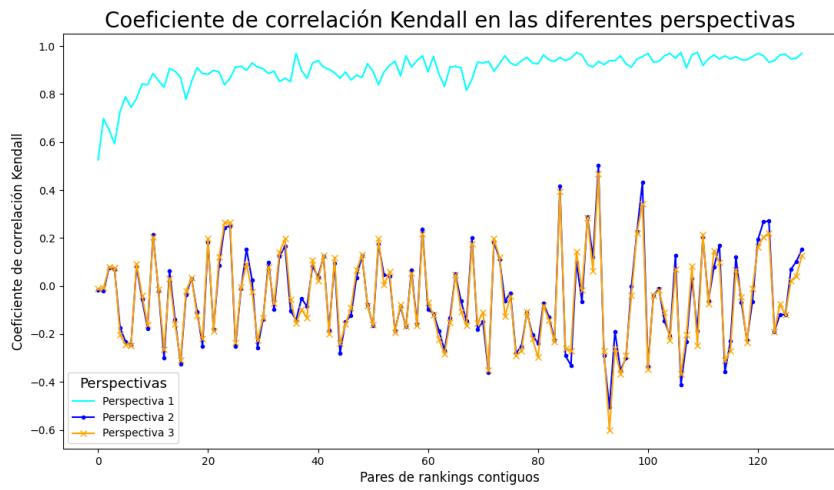


Figura 5.1: Representación de la evolución del coeficiente de correlación de Kendall en las tres perspectivas

## 5.2. Aplicación del algoritmo de Girvan-Newman

Para explicar los resultados obtenidos en la aplicación del algoritmo para la detección de comunidades se va a explicar la situación obtenida por perspectivas y subvariantes. A modo visión global, y como era de esperar, la aplicación del algoritmo a las perspectivas relativas a las medida 1 y medida 2 son las que mayor información aportan.

En primer lugar, para las perspectivas **1**, **2**, **2.1**, **3** y **3.1**, los resultados han sido que en la primera iteración se generan dos comunidades. La primera con un solo nodo y otra con los nodos restantes. A partir de aquí, la "mega-comunidad" con todos los nodos se va desagregando en comunidades de un solo nodo a lo largo de las iteraciones sucesivas. La generación de estas "mono-comunidades" hay ocasiones que se generan en el orden de los nodos y otras ocasiones en las que se van creando de forma aleatoria. Claramente, esto no permite extraer conclusiones debido a que no se obtienen comunidades bien definidas con varios nodos relacionados.

Para estudiar los resultados de las variantes restantes se ha tomado una iteración representativa de cada una de ellas y se comentará la situación obtenida. Estos grafos con la aplicación del algoritmo pueden verse en [D.1](#) para la perspectiva 1 y en [D.2](#) para la perspectiva 2. Puesto que la perspectiva 3 es la que ha ofrecido resultados más interesantes, se detallará posteriormente en esta sección. Adicionalmente, todos los grafos con todas las iteraciones del algoritmo de Girvan-Newman sobre ellas se pueden encontrar en [E](#).

Para la **primera perspectiva y sus subvariantes**, concretamente **1.1**, **1.2** y **1.3**, revisadas en [D.1](#), se ha obtenido:

- Para el caso 1.1, aparecían Arcelormittal, Banco Santander, Fluidra, IAG e Inmobiliaria Colonial bajo una comunidad, siendo Arcelormittal la empresa que se mantiene en la comunidad con el avance de las iteraciones.
- En el caso 1.2, Amadeus y Enagás se encuentran bajo una misma comunidad.
- La relación en 1.2 y 1.3 entre Endesa y Sacyr.
- La fuerte relación entre Solaria y Red Eléctrica, que forman una comunidad, junto con Iberdrola en menor grado y Ferrovial en un grado aún menor, que solo se unen a las dos primeras en iteraciones tempranas y Ferrovial en el caso 1.3 no es capaz de ello.
- Arcelormittal y Rovi en las dos últimas subvariantes forman una comunidad sólida que en iteraciones anteriores es acompañada por Pharma Mar y en el caso 1.2 por Cellnex. También para ese caso estas dos últimas forman una comunidad que puede aportar cierta información.

Se pasa entonces a estudiar la **segunda perspectiva y sus subvariantes**, concretamente **2.2** y **2.3**, revisadas en [D.2](#), con los siguientes resultados:

- Acciona y Rovi guardan una alta relación, pues en ambos casos están integradas en una misma comunidad y además se mantienen a lo largo de las iteraciones. Pharma Mar y Rovi también tienen estrecha relación como se ve en el caso 2.3, en el que son la última comunidad en separarse. Sin embargo, Acciona y Pharma Mar no guardan relación alguna, de hecho no presentan una arista común en el grafo de competitividad.

Derivado de 2.3, Mapfre y Meliá Hotels se presentan en una comunidad conjunta que además guarda relación con todas las empresas anteriores.

- Amadeus e Inmobiliaria Colonial se ubican en una misma comunidad en ambos casos. Adicionalmente en 2.2 con Fluidra y Siemens Gamesa, y en 2.3 con Aena. Las empresas comentadas para 2.2, en este grafo, presentan una comunidad más fuerte que Amadeus e Inmobiliaria Colonial.
- De ambos casos estudiados se deduce que Caixabank, IAG y Solaria junto con Inditex y Repsol tienen cierta relación pero son estas dos últimas empresas, conjuntamente, independientes del resto.

## 5.2. Aplicación del algoritmo de Girvan-Newman

---

- Bankinter, Iberdrola e Indra presentan en 2.3 una comunidad y en 2.2 guardan estrecha relación con Acerinox, Aena, Endesa, BBVA y Red Eléctrica. Pese a ello, las tres primeras empresas junto con Endesa formarán en la subvariante 2.2 una comunidad.
- De 2.3 se obtiene que Acs, Acerinox, Arcelormittal y Endesa comparten comunidad mientras que lo que se vio en 2.2 es que las dos primeras empresas formaban una sola comunidad que se relacionaba con la formada por Pharma Mar, Sacyr, Merlin Properties, Naturgy y Banco Santander.
- De este último punto se tiene en 2.3 que Banco Santander, Sacyr y Merlin Properties forman una sola comunidad y que Naturgy con Enagás y Cellnex forman otra. Sin embargo, estas dos empresas en 2.2 forman una comunidad con Mapfre y Meliá Hotels.

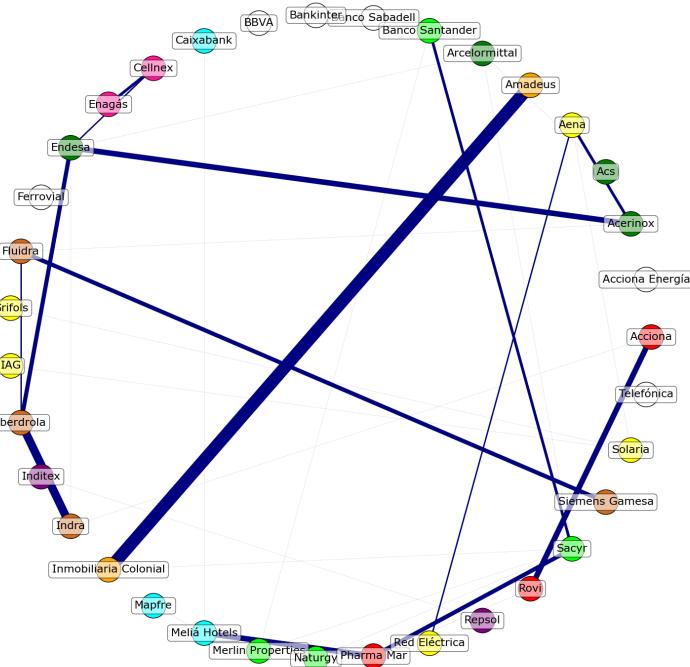
En comparación con el primer enfoque, esta segunda perspectiva presenta muchísimas más sinergias y solapamientos de ideas sobre las comunidades por lo que habrá que ver qué conclusiones se pueden sacar de toda esta información.

Finalmente, se estudia la tercera perspectiva y sus subvariantes a través de los enfoques **2.2** y **2.3**, visibles en [5.2](#). Hay que recordar que este caso se va a explicar detalladamente aquí como se ha hecho para los casos anteriores en [D](#).

En la subvariante **3.2** ocurre como para los casos de las subvariantes del enfoque 2 analizadas, en las que se generan dos grandes comunidades y a partir de estas se van generando el resto de comunidades desagregándose las dos "mega-comunidades". Sin embargo, para el caso **3.3**, en la primera iteración ya sale un número de comunidades importante, concretamente siete.

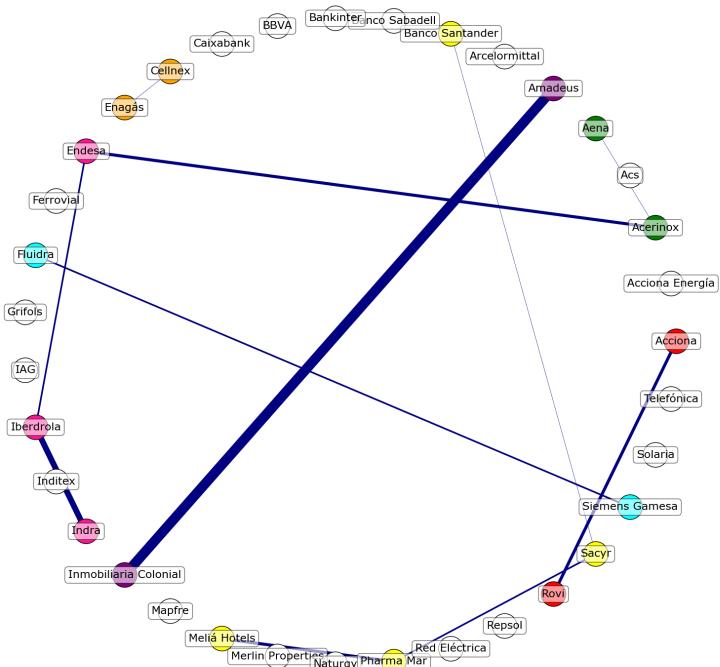
- Para analizar la subvariante **3.2** se toma la iteración 7 ([5.2 \(a\)](#)), que tiene las siguientes comunidades:
  1. La comunidad **verde** procede de una comunidad anterior que se separó en esta primera y en la color **chocolate**. La agrupación **verde** la forman Acerinox, Acs, Arcelormittal y Endesa mientras que la **chocolate** la componen Fluidra, Iberdrola, Indra y Siemens Gamesa. Esta última se descompone posteriormente en Iberdrola e Indra por un lado y las restantes por otro. Además, la comunidad **verde** se descompone mucho antes para llegar a comunidades de un solo nodo.
  2. Aena, Grifols, IAG, Red Eléctrica y Solaria forman la comunidad **amarilla**. Esta surge en las primeras iteraciones y las empresas conviven hasta que Grifols se desprende de ellas y las restantes forman dos comunidades con el mismo cardinal. Es decir, a un lado queda Aena y Red Eléctrica y al otro IAG y Solaria. Ninguna de estas comunidades sobrevive hasta las últimas iteraciones del algoritmo.

**Application of Girvan Newman algorithm: Perspective 3.2, iteration 7**



(a)

**Application of Girvan Newman algorithm: Perspective 3.3, iteration 1**



(b)

Figura 5.2: Aplicación del algoritmo de Girvan-Newman a las perspectivas 3.2 y 3.3: Iteración 7 e Iteración 1, respectivamente

3. Adicionalmente, se tiene la comunidad **cyan**, que cuenta con Caixa-bank, Mapfre y Meliá Hotels, originaria de la comunidad **roja** que se explicará seguidamente. La comunidad, que luego quedará sin Caixa-bank, será de las últimas en separarse.
  4. En esta iteración Amadeus e Inmobiliaria Colonial forman la comunidad **naranja**, la cual procede de la comunidad de color **roja** que desprendió esta comunidad naranja que, además, contaba con la actual comunidad **lima**, que recoge a Banco Santander, Merlin Properties, Naturgy y Sacyr. Esta última comunidad, salvo Naturgy, se mantendrá a lo largo de las iteraciones hasta que se desprenda el Banco Santander para quedar solamente dos en la misma.
  5. La comunidad entre Cellnex y Enagás, en **rosa**, es originaria del conglomerado **verde**. Esta comunidad es una de las que más se mantienen a lo largo de las iteraciones.
  6. La **morada**, con Inditex y Repsol, surge en la primera iteración y se mantiene varias iteraciones.
  7. Por último, la comunidad colerada en **rojo**, que en esta iteración cuenta con Acciona, Rovi y Pharma Mar, contaba en un origen con las actuales **naranja**, **lima** y **cyan**. Esta comunidad cuenta con Pharma Mar y Rovi que será el último enlace en romperse antes de finalizar la aplicación del algoritmo.
- Por último, para la subperspectiva **3.3** se ha escogido la iteración 1 ([5.2 \(b\)](#)) ya que, como se ha comentado previamente, en esta primera ya se tiene un número de comunidades considerable y los nodos están bien clasificados en ellas. En varias de estas, las comunidades las constituyen un par de nodos que solamente se relacionan entre ellos:
    1. **Verde**: Aena y Acerinox.
    2. **Roja**: Acciona y Rovi.
    3. **Morada**: Amadeus e Inmobiliaria Colonial.
    4. **Amarilla**: Banco Santander, Meliá Hotels, Pharma Mar y Sacyr. Aunque posteriormente se separarán para que Banco Santander y Sacyr formen una comunidad por sí solas y las otras dos empresas restantes formen otra.
    5. **Naranja**: Cellnex y Enagás.
    6. **Rosa**: Endesa, Iberdrola e Indra, de la que luego Endesa se desprenderá.
    7. **Cyan**: Fluidra y Siemens Gamesa.

Para esta subvariante, las comunidades que más se mantienen a lo largo de las iteraciones son la de Iberdrola e Indra, procedente de la **rosa**, y la de Meliá Hotels y Pharma mar, originaria en la **amarilla**.

Antes de acabar con esta perspectiva, para esta visión en concreto, la 3.3, se ha desarrollado un dendograma sobre las iteraciones sucesivas del algoritmo de Girvan-Newman.

**Dendograma de la aplicación del algoritmo de Girvan - Newman a la perspectiva 3.3**

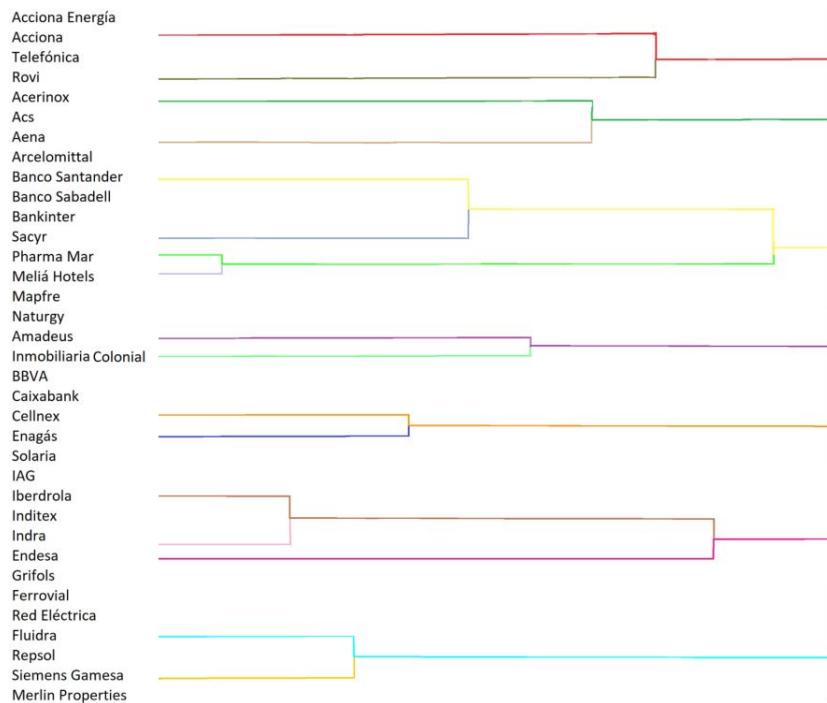


Figura 5.3: Dendograma relativo a la aplicación del algoritmo de Girvan-Newman a las subvariante 3.3

Con este dendrograma 5.3 se pueden visualizar de derecha a izquierda cómo se han ido construyendo las comunidades a lo largo de las iteraciones del algoritmo sobre el grafo de competitividad. En concreto, cuenta con 11 iteraciones. Como se puede además observar, finaliza cuando todas las comunidades presentan nodos hoja, es decir, comunidades formadas por un solo nodo.

Para finalizar este exhaustivo apartado, se concluye para el enfoque 3 y sus subvariantes lo siguiente:

- De la 3.2 se deduce que Inditex y Repsol pueden encuadrarse en la misma comunidad. En 3.3 no hay referencias de este hecho porque no presentan aristas adyacentes.
- Acciona y Rovi se presentan en una misma comunidad en ambos casos. Aunque en 3.2 también está junto a Pharma Mar, se revela que estas empresas presentan alta competitividad relativa entre ellas. De hecho, es en 3.2 el enlace entre Pharma Mar y Rovi destacado por ser el último en romperse.

- Junto a las empresas anteriores se puede encontrar en 3.2 cierta relación con la comunidad que forman Amadeus e Inmobiliaria Colonial, bien separadas en 3.3 y en 3.2. Esta pareja de empresas también ha de ser estudiada para posibles conclusiones del estudio.
- En la línea de estos dos puntos anteriores, Banco Santander y Sacyr, que formarán luego una sola comunidad, se encuentran en 3.3 en una comunidad con Pharma Mar y Meliá Hotels, de donde se desvela también la alta relación de estas dos últimas que incluso se conserva hasta la última iteración del algoritmo.
- En esta línea, en 3.2, aparece una comunidad entre Mapfre, Caixabank y Meliá Hotels. Habrá que ver qué relación guarda Pharma Mar entre esta comunidad y la descrita en el anterior punto.
- También siguiendo lo anterior, Banco Santander, Merlin Properties, Naturgy y Sacyr están en una misma comunidad en 3.2.
- Por otro lado, aparecen Cellnex y Enagás, altamente relacionadas en ambos casos. Además, en 3.2 se puede relacionar, en primeras iteraciones, con la comunidad que contiene las empresas del siguiente punto.
- Además, se puede deducir fuerte conexión entre Fluidra y Siemens Gamesa y entre Endesa, Iberdrola e Indra, aunque con Endesa en menor grado. Esta idea, proveniente de 3.3, también se puede deducir de 3.2, donde sin contar con Endesa, las empresas anteriores están en una misma comunidad.
- La comunidad de Endesa en 3.2 que comparte con Acerinox, Acs y Arcelormittal tiene estrecha conexión con la de Iberdrola, Siemens Gamesa, Fluidra e Indra.
- Por último, aunque en 3.3 Aena y Acerinox comparten comunidad, formada por ellas mismas, en 3.2 Aena aparece junto a Grifols, Solaria, Red Eléctrica e IAG, la cual no tiene cabida en 3.3 debido a que no compiten con ninguna otra de ese grafo.

Con todos estos puntos observados, habrá que estudiarlos para poder extraer conclusiones en el siguiente y último capítulo. Además, hay que tener especial cuidado, a priori, cuando se estudien los grafos relativos a las perspectivas 2 y 3 debido a que ambas referencian la volatilidad y podrían presentar disposiciones semejantes que permitirán extraer unas conclusiones interesantes.

Adicionalmente al análisis anterior, se pueden obtener conclusiones a partir del último ranking de la perspectiva 1 [B.5](#), donde se encuentran las empresas que mayor rentabilidad han generado en este período de tiempo. En este sentido, las empresas que lideran el ranking son: Siemens Gamesa, Caixabank y Bankinter. Se procede entonces a una revisión exhaustiva de las comunidades a las que pertenecen

estas empresas en los diferentes grafos relativos a las tres perspectivas y sus subvariantes, recopilando la situación de las mismas a lo largo de las iteraciones de aplicación del algoritmo de Girvan-Newman.

1. Para Siemens Gamesa se ha observado que en la perspectiva 2.2 forma una sólida comunidad con Amedeus, Fluidra e Inmobiliaria Colonial. En lo relativo al enfoque 3.2 forma en una iteración avanzada una comunidad destacable con Fluidra, Iberdrola e Indra aunque esta agrupación procede de otra con Acerinox, Acs, Arcelormittal y Endesa, lo que significa que Siemens Gamesa tiene también relación con estas empresas pero en menor grado. En la última casuística, la 3.3, esta empresa forma comunidad con Fluidra. Para el resto de enfoques o bien Siemens Gamesa no competía con ninguna otra empresa o la densidad de aristas del grafo no permitía la detección de comunidades de manera clara.

Tras el paso por todas las variantes, queda claro que Siemens Gamesa guarda una estrecha relación con Fluidra y, en menor grado, con las empresas mencionadas anteriormente.

2. Repitiendo el ejercicio para Caixabank se observa que en 2.2 forma comunidad con Inditex, Repsol y Solaria, procedente de una comunidad más grande junto con Acciona, Ferrovial, IAG y Rovi. En línea de este último comentario, en 2.3 forma comunidad con algunas de ellas, concretamente con IAG, Inditex, Repsol y Solaria, la cual es bastante sólida hasta que Caixabank se separa con IAG y con Solaria. Por otro lado, en 3.2 esta empresa se agrupa junto con Mapfre y Meliá Hotels, que en iteraciones anteriores se agrupaban con Acciona, Pharma Mar y Rovi. Para el resto de enfoques o bien Caixabank no competía con ninguna otra empresa o la densidad de aristas del grafo no permitía la detección de comunidades de manera clara.

Como resultado, se tiene que Caixabank guarda una estrecha relación con Inditex, Repsol y Solaria y, en menor grado, con IAG, Mapfre y Meliá Hotels.

3. Por último, para Bankinter se ha observado que en el enfoque 2.2 forma comunidad junto a Acerinox, Aena, Endesa, Iberdrola, Indra y Red Eléctrica. Parecido al caso anterior, en 2.3 forma una comunidad bastante grande con algunas de las empresas anteriores y otras en las primeras iteraciones pero finalmente con las empresas con las que forma una comunidad más pequeña después es con Iberdrola e Indra.

A modo final, se ha visto que Bankinter en muchas ocasiones no tenían aristas adyacentes, lo que indica que su competitividad con el resto de empresas es baja. Además, las empresas con las que mayor relación guarda son Iberdrola e Indra que son las empresas comunes a sus comunidades más destacables.

Revisando los resultados anteriores se obtiene lo siguiente. Además, hay que añadir que solo unas pocas empresas han conseguido obtener en este período rentabilidades no negativas, por lo que cuando se evalúe si algunas comunidades son rentables o no habrá que poner atención en este hecho. Para el estudio de rentabilidades conjuntas se han ido estudiando los porcentajes de rentabilidad para diferentes combinaciones de empresas 5.4.

Agrupaciones	Rentabilidad
Siemens Gamesa + Fluidra	-17.5269
Caixabank + Inditex + Repsol	7.7177
Caixabank + Inditex + Repsol + Solaria	-2.1341
Bankinter + Indra + Iberdrola	-0.9281

Figura 5.4: Cuadro de rentabilidades conjuntas en el período

- Siemens Gamesa compartía comunidad con Fluidra en varias aplicaciones del algoritmo pero una inversión conjunta de estas dos empresas ha generado una rentabilidad de -17.5269 %. Obviamente, para el resto de comunidades que se han generado, como todas tenían empresas con rentabilidades negativas, estas comunidades hubiesen generado rentabilidad negativa.
- Una inversión conjuntamente en Caixabank, Inditex y Repsol, ha generado una rentabilidad de 7.7177 %, mientras que tomando las tres empresas anteriores y Solaria como indicaba uno de los resultados obtenidos ha dado una rentabilidad de -2.1341 %. Como ocurría para el punto anterior, para el resto de comunidades que se han desarrollado, la rentabilidad conjunta hubiese sido negativa debido a la rentabilidad negativa de las propias empresas que las forman.
- Aunque Bankinter no competía en numerosas ocasiones, tras la revisión se ha observado que junto con Iberdrola e Indra forma una comunidad bastante sólida. La rentabilidad de esta agrupación ha sido en el período de -0.9281 %.



# 6

## Conclusiones y trabajos futuros

Este último capítulo tiene como objetivos la exposición de las conclusiones del estudio, es decir, ante una posible inversión futura, basándose en estos datos, en qué empresas puede ser razonable invertir debido a su historial de rentabilidades positivas en el período analizado y, por otro lado, plantear variantes y caminos alternativos a este estudio para poder mejorarlo o, simplemente, ideas que podrían desarrollarse y que, a priori, podrían aportar valor a este análisis.

Hay que tener en cuenta que los comentarios sobre las situaciones estudiadas han sido extraídos de datos que están recogidos desde el 22 de abril de 2022 hasta el 21 de octubre de 2022.

Revisando los puntos comentados anteriormente en la sección *Resultados 5*, se obtienen las siguientes **conclusiones**:

- Del coeficiente de correlación de Kendall, del que se puede ver representada su evolución en la figura 5.1, se obtiene que la perspectiva 1 presenta valores cercanos a 1, lo que indica una alta correlación entre los rankings, así como una baja competitividad entre los nodos. La consecuencia de esto es que el volumen de aristas, pese a ser alto por la cantidad de información recogida, no aporta tanta información como las perspectivas restantes. Al fin y al cabo, debido a que hay 130 rankings es de esperar que todas compitan con todas, pero no existe una alta competitividad. Las perspectivas 2 y 3 compiten de forma muy similar pues las gráficas se suelen mover en los mismos valores aunque no siempre idénticos. Además, los valores del coeficiente indican que pese a que hay rankings bastante correlados, éstos son pocos y

la competitividad entre ellos es notoria. Es por ello que estas dos variantes aportan más información que la primera. Además, debido a la naturaleza de los valores que toman los rankings, hace que ambas perspectivas nos muestren una idea acerca de la volatilidad de las acciones.

- Del primer enfoque se obtuvo que Endesa y Sacyr (C.1) compartían comunidad. Además, en la perspectiva 1 es la pareja que más compite y se observa que cuando una de ellas comienza un movimiento ascendente o descendente en su valor, la otra copia el movimiento. Concretamente, este suceso se ha dado en el 57,6 % de las sesiones estudiadas. Esto sería útil para que cuando una de ellas comience una tendencia clara se invierta en la otra en búsqueda de rentabilidad.
- Las empresas Acciona y Rovi han compartido comunidad en las perspectivas 2 y 3. De los rankings generados, se observa que ante posiciones extremas en los rankings de una de ellas, normalmente, la otra responde con una volatilidad alta en sentido contrario. Aunque hay casos en los que presentan una volatilidad similar, es recurrente ver que cuando una se hunde en el ranking la otra se lanza hacia arriba. Además, a nivel valor, presentan competitividad baja aunque más alta que lo siguiente que se va a ver, por lo que se podría decir que ante un aumento de valor de una, la otra pocas veces va a poder superarla.

En estas perspectivas también aparecen relacionadas con Pharma Mar, pero esta empresa ocupa posiciones intermedias en los rankings y no permite extraer conclusiones claras en este contexto.

Para cerrar este punto, se concluye que un movimiento en la volatilidad brusco de Acciona o Rovi implica una réplica de la otra empresa en sentido contrario, en la mayoría de los casos.

- De los enfoques 2 y 3 se ha obtenido que Inditex y Repsol forman una comunidad fuerte. Esto se traduce en una conclusión similar al anterior punto, agudizado cuando una de las dos toma posiciones extremas dentro de los rankings. En cuanto a los valores totales, las empresas compiten muy poco. Para Mapfre y Meliá Hotels ocurre prácticamente una casuística similar.
- También en los enfoques 2 y 3 las empresas Acerinox, Acs y Arcelomittal han sido integradas en una misma comunidad. La situación es prácticamente análoga a los puntos anteriores, pero tratando las tres empresas conjuntamente si dos de ellas presentan posiciones muy extremas en un sentido, la otra empresa suele replicar en el contrario, lo que es bastante importante ante una posible inversión porque, además, las condiciones en este caso se refuerzan con la observación de tres empresas y no de dos.
- En la perspectiva 1 se observó que Solaria y Red Eléctrica (C.2) compartían comunidad debido a su alta competitividad. Tras revisar los rankings de esta

---

perspectiva se ha visto que, aunque Red Eléctrica suele tener más valor que Solaria, cuando esta supera a la primera, se mantiene unas sesiones y rápidamente Red eléctrica vuelve a superarla. Desde el punto de vista inversor esto puede ser interesante en dos sentidos:

1. Cuando Solaria supere a Red Eléctrica puede ser buen momento para operar en corto con Solaria porque su valor bajará unas sesiones más tarde.
  2. Cuando Red Eléctrica es superada por Solaria, es buen momento para invertir en la primera y cuando supere a Solaria se debería estar en ganancias.
- Invertir simultáneamente en Caixabank, Inditex y Repsol ha generado una rentabilidad positiva en este período (7,718 %) por lo que si la tendencia siguiese sería una buena inversión a corto plazo.

Una vez planteadas las conclusiones del estudio, se han desarrollado paralelamente al mismo **posibles mejoras para futuros evolutivos del trabajo:**

1. Se podría considerar trabajar con el precio de apertura o con otra medida en torno al precio. Con ellas, la idea sería replicar los rankings y los grafos, aplicar el algoritmo y observar si existe algún patrón. Un análisis prácticamente similar.
2. Con los comentarios exhaustivos hechos en el capítulo *Resultados 5*, se podrían extraer más conclusiones, no tan importantes como las comentadas anteriormente, pero se observaron parejas de empresas, e incluso tríos de empresas que se relacionaban en una misma comunidad y que pueden aportar también valor a la extracción de conclusiones. Es el caso de, por ejemplo, Amadeus e Inmobiliaria Colonial que también se ha visto que presentan comportamientos singulares al analizar su volatilidad.
3. Sería útil, para agilizar el análisis, la implementación de un método que dado dos, tres o las empresas que se deseen represente la evolución de las posiciones de las mismas a lo largo de las familias de rankings. Esto permitiría observar los cruces que se dan entre las posiciones de las empresas y ello puede llevar a detectar ciertos patrones de comportamiento de alguna de ellas en función del comportamiento del resto.
4. Se podría valorar la aplicación de algún otro algoritmo de detección de comunidades para observar en qué se diferencian los resultados de los obtenidos a través del de Girvan-Newman.
5. El análisis que se ha hecho para las empresas que más valor tienen al final del estudio podría replicarse para las empresas que han dado menor rentabilidad de cara a una inversión a corto plazo.



# Bibliografía y referencias

- [1] Menéndez Velázquez, Amador; *Una breve introducción a la teoría de grafos*, junio 1998, Suma., n. 28; p. 11-26.  
<https://redined.educacion.gob.es/xmlui/bitstream/handle/11162/13526/011-026.pdf?sequence=1>
- [2] Verbel de León, Arturo; *Breve historia de la Teoría de Grafos*, junio 2018, Mi camino Master.  
<http://micaminomaster.com.co/grafo-algoritmo/historia-teoria-grafos/>
- [3] Wilson, Robin J.; *Introducción a la teoría de grafos*, 1983, Alianza, Versión Española De Enrique García Camarero.
- [4] Tutte, W.T.; *Graph Theory*, 1984, Addison-Wesley, Volume 21, Cambridge Mathematical Library.  
<https://books.google.es/books?hl=es&lr=&id=uTGhooU37h4C&oi=fnd&pg=PR13&dq=graph+theory&ots=DJt6UKIt0Z&sig=RzE24jzdebaFm4yf4-3Kr-Qs8PE#v=onepage&q=graph20theory&f=false>
- [5] Referencia contexto económico 1.  
[https://www.expansion.com/economia-para-todos/inversion/que-es-la-bolsa-cual-fue-su-origen-y-funcion.html#:~:text=Las%20bolsas%20son%20antigu%C3%ADsimas%20y,comerciantes%20en%20Brujas%20\(B%C3%A9lgica\)](https://www.expansion.com/economia-para-todos/inversion/que-es-la-bolsa-cual-fue-su-origen-y-funcion.html#:~:text=Las%20bolsas%20son%20antigu%C3%ADsimas%20y,comerciantes%20en%20Brujas%20(B%C3%A9lgica))
- [6] Referencia contexto económico 2.  
<https://blog.caixabank.es/blogcaixabank/la-historia-de-la-bolsa-como-empezo-todo/#:~:text=El20concepto20de20bolsa20de,modo20de20escudo20de20armas>
- [7] Coronicová Hurajová, Jana y Madaras, Tomáš; *Revising the Newman-Girvan algorithm*, 2016, ITAT, Volume 1649, p. 200-205.  
<https://ceur-ws.org/Vol-1649/200.pdf>
- [8] Kaufmann, A. *Puntos Y Flechas : Teoría De Los Grafos*. Marcombo, 1976. Print.
- [9] Criado, R.; García, E.; Pedroche, F. y Romance, M.; *Comparing rankings by means of competitiveness graphs: structural properties and computation*, 25 marzo 2014, arXiv.  
<https://arxiv.org/abs/1310.6921>
- [10] Criado, R.; García, E.; Pedroche, F.; Romance, M.; *A new method for comparing rankings through complex networks: Model and analysis of competitiveness of major European soccer leagues*, 31 octubre 2013, Chaos 23, 043114 (2013).  
<https://doi.org/10.1063/1.4826446>

## BIBLIOGRAFÍA Y REFERENCIAS

---

- [11] Pedroche, F.; Criado, R.; García, E.; Romance, M. y Sánchez, V.E.; *Comparing series of rankings with ties by using complex networks: An analysis of the Spanish stock market (Ibex-35 Index)*, marzo 2015, Networks and Heterogeneous Media, American Institute of Mathematical Sciences, Volume 10, Number 1, p.101-125.  
<http://www.aims sciences.org/journals/displayArticlesnew.jsp?paperID=10842>
- [12] Referencia información de cotización de las empresas del IBEX 35: Bolsas y Mercados Españoles.  
<https://www.bolsasymercados.es/bme-exchange/es/Mercados-y-Cotizaciones/Acciones/Mercado-Continuo/Precios/ibex-35-ESOSI0000005>
- [13] Criado, R.; Muñoz, R.; *Un cuatrimestre de Matemática Discreta*, 29 septiembre 2005.  
<https://reginocriado.files.wordpress.com/2013/01/md.pdf>
- [14] Girvan, M. y Newman, M. E. J.; *Community structure in social and biological networks*, 11 junio 2022; PNAS; Vol. 99, No. 12.  
<https://doi.org/10.1073/pnas.122653799>
- [15] Referencia económica 1.  
<https://www.ig.com/en/glossary-trading-terms/share-price-definition#:~:text=A%20share%20price%20%E2%80%93or%20a,company%20isn't%20meeting%20expectations>
- [16] Hayes, Adam; Silberstein, Samantha; Kazel, Melody. *Stocks: What They Are, Main Types, How They Differ From Bonds*, 06 julio 2022, Investopedia.  
<https://www.investopedia.com/terms/s/stock.asp>
- [17] Newman, M. E. J. y Girvan, M.; *Finding and evaluating community structure in networks*, 26 febrero 2004, Physical Review E 69, American Physical Society, 026113 (2004).  
[http://www.cse.cuhk.edu.hk/~cslui/CMSC5734/newman\\_community\\_struct\\_networks\\_phys\\_rev.pdf](http://www.cse.cuhk.edu.hk/~cslui/CMSC5734/newman_community_struct_networks_phys_rev.pdf)
- [18] Méndez Carmona, Alejandro; Morillas Gómez, Samuel; Hervás Jorge, Antonio; *Algoritmos de detección de comunidades en grafos y aplicación en conjuntos de datos de tránsito en aeropuertos y trasvase de alumnos entre titulaciones de la UPV*, curso 2021/2022, Universidad Politécnica de València.  
<https://riunet.upv.es/bitstream/handle/10251/185654/Mendez%20-%20Algoritmos%20de%20deteccion%20de%20comunidades%20en%20grafos%20y%20aplicacion%20en%20conjuntos%20de%20datos%20de%20t....pdf?sequence=1&isAllowed=y>
- [19] Newman, M. E. J.; *Detecting community structure in networks*, 01 marzo 2004, Eur. Phys. J. B 38, 321–330 (2004).  
<https://link.springer.com/article/10.1140/epjb/e2004-00124-y>
- [20] Curry, Benjamin; Ashford, Kate; Schmidt, John; *What Is Market Volatility - And How Should You Manage It?*. 25 julio 2022, Forbes Advisor.  
<https://www.forbes.com/advisor/investing/what-is-volatility/>
- [21] Likas, Aristidis; Vlassis, Nikos; Verbeek, Jakob J.; *The global k-means clustering algorithm*, 2003, Pattern Recognition, Volume 36, Issue 2, 2003, p. 451-461, ISSN 0031-3203.  
[https://www.sciencedirect.com/science/article/pii/S0031320302000602?casa\\_token=oK9t-8vlndnOAAAAAA:kWmYFD8\\_7fsiPsU2gX-8Fr0uhs15RmgSsDjUrcUPy1LWvFOM1n2bJSIADEu7QWJ2QlNRIOE](https://www.sciencedirect.com/science/article/pii/S0031320302000602?casa_token=oK9t-8vlndnOAAAAAA:kWmYFD8_7fsiPsU2gX-8Fr0uhs15RmgSsDjUrcUPy1LWvFOM1n2bJSIADEu7QWJ2QlNRIOE)



# Apéndices



# A

## Funciones definidas y empleadas en la práctica

En este apéndice se recogen los códigos de las funciones que han sido definidas en la práctica y que han sido necesarias para el desarrollo del análisis práctico.

En primer lugar, A.1 hace referencia a la función *ordenarRanking* que se emplea para ordenar los rankings de las 3 perspectivas que se han planteado.

```
#Algoritmo para ordenar los rankings
def ordenarRanking(data):
    n = len(data)
    for i in range(n-1):
        minimo = i
        temp_val = data.iloc[i,1]
        temp_name = data.iloc[i,0]
        for j in range(i,n):
            if data.iloc[j,1] > temp_val:
                minimo = j
                temp_val = data.iloc[j,1]
                temp_name = data.iloc[j,0]
        data.iloc[minimo,1] = data.iloc[i,1]
        data.iloc[minimo,0] = data.iloc[i,0]
        data.iloc[i,1] = temp_val
        data.iloc[i,0] = temp_name
    return None
```

Figura A.1: Implementación de la función *ordenarRanking*

Por otro lado, la función *seCruza*, A.2, es la empleada para conocer si dadas dos empresas y dos rankings estas intercambian sus posiciones relativas, es decir, si se cruzan para esa pareja de rankings.

---

```

#Algoritmo para saber si dado un par de acciones estas se cruzan en dos rankings dados
def seCruza(data1,data2,stock1,stock2):
    n = len(data1)
    find1_1 = False
    find1_2 = False
    i = 0

    while (i<n and (find1_1==False or find1_2==False)):
        if data1[i] == stock1:
            find1_1 = True
            pos1_1 = i
        if data1[i] == stock2:
            find1_2 = True
            pos1_2 = i
        i=i+1

    find2_1 = False
    find2_2 = False
    j = 0

    while (j<n and (find2_1==False or find2_2==False)):
        if data2[j] == stock1:
            find2_1 = True
            pos2_1 = j
        if data2[j] == stock2:
            find2_2 = True
            pos2_2 = j
        j=j+1

    up1= stock1 if (pos1_1>pos1_2) else stock2
    up2= stock1 if (pos2_1>pos2_2) else stock2

    flag = not(up1==up2)

    return flag

```

Figura A.2: Implementación de la función *seCruza*

Otra función que también se emplea en la práctica es *insertarColumnaDiferencia* A.3, que calcula la diferencia entre los valores de dos columnas para cada fila. Esta función es importante para la perspectiva 2 porque establece los valores que van asociados a cada empresa en cada ranking para posteriormente poder ordenar dichas empresas. Concretamente, calcula la diferencia entre los valores de la inversión para pares de sesiones consecutivas.

```

def insertarColumnaDiferencia(data):
    n_rows = len(data)
    diferencia = []
    for i in range(0,n_rows):
        aux = data.iloc[i,3]-data.iloc[i,2]
        diferencia.append(aux)

    data.insert(4,"Diferencia valor",diferencia,True) #Permite duplicados
    return None

```

Figura A.3: Implementación de la función *insertarColumnaDiferencia*

También se ha definido y empleado la función *insertarColumnaVariación* que se utiliza para crear en cada DataFrame de cada empresa la columna relativa a la variación porcentual diaria del precio de cotización de una sesión respecto a su anterior, utilizada en la perspectiva 3.

```
def insertarColumnaVariacion(data):
    n_sesions = len(data)
    variacion = [0]
    for i in range(1,n_sesions):
        aux = ((data.iloc[i,1]-data.iloc[i-1,1])*100/data.iloc[i-1,1])
        variacion.append(round(aux,4))

    data.insert(9,"% Variacion cierre sesion",variacion,True)
    return None
```

Figura A.4: Implementación de la función *insertarColumnaVariacion*

Por último, la función *coefKendall* es la empleada para calcular el coeficiente de correlación de Kendall en las tres perspectivas. Hay que recordar que servía para analizar la correlación entre los rankings de cada una de las tres casuísticas.

```
def coefKendall (data1,data2,listStocks):
    n = len(data1)
    n_compete = 0
    n_non_compete = 0

    for i in range(n):
        for j in range(i+1,n):
            if (seCruza(data1,data2,listStocks[i],listStocks[j])):
                n_compete = n_compete + 1
            else:
                n_non_compete = n_non_compete + 1

    coef = (n_non_compete - n_compete)/((n*(n-1))/2)
    return coef
```

Figura A.5: Implementación de la función *coefKendall*

# B

## Resultados obtenidos en la práctica

En este apéndice se recogen parte de los resultados obtenidos al hacer el análisis práctico. Puesto que las estructuras generadas al ejecutar el código son bastante repetitivas, aquí se almacenan ejemplos de cada una de ellas para poder entender mejor lo obtenido en el estudio. Para poder consultar las restantes, se puede hacer a través del repositorio de archivos de GitHub [E](#).

En primer lugar, los archivos.csv que se han abierto en la práctica relativos a la información histórica de cada empresa tienen el siguiente aspecto. En este caso para la empresa *Inditex*, pero las 34 restantes presentan la misma configuración.

	Fecha	Cierre	Referencia	Volumen	Efectivo	Ultimo	Maximo	Mínimo	Medio
0	22/04/2022	20.32	21.090	16.716.449	346.525.123,56	20.32	20.92	20.320	20.5115
1	25/04/2022	20.25	20.320	4.190.699	84.440.298,37	20.25	20.38	19.895	20.1495
2	26/04/2022	20.26	20.250	17.156.684	351.031.637,03	20.26	20.85	20.260	20.4903
3	27/04/2022	20.31	20.260	6.015.487	122.393.751,21	20.31	20.49	20.130	20.3194
4	28/04/2022	20.14	19.845	2.813.693	56.551.614,54	20.14	20.23	19.900	20.0987
...	...	...	...	...	...	...	...	...	...
126	17/10/2022	22.38	21.870	2.735.570	60.873.877,69	22.38	22.38	21.860	22.2789
127	18/10/2022	22.75	22.380	7.626.535	173.886.817,04	22.75	23.08	22.440	22.7843
128	19/10/2022	22.73	22.750	24.285.791	549.740.894,11	22.73	22.99	22.450	22.7217
129	20/10/2022	22.99	22.730	3.121.176	71.253.356,48	22.99	23.06	22.470	22.9013
130	21/10/2022	22.29	22.990	5.655.174	125.486.304,40	22.29	22.71	21.760	22.1738

131 rows x 9 columns

Figura B.1: Ejemplo de ingesta de datos para Inditex

A continuación, se adjunta el primer ranking generado para la perspectiva 1 [B.2](#), ya ordenado por la función para ordenar rankings explicada anteriormente.

## Capítulo B. Resultados obtenidos en la práctica

---

	Nombre	Valor
0	Grifols	1047.443007
1	Meliá Hotels	1033.595425
2	Siemens Gamesa	1027.044855
3	Pharma Mar	1015.203145
4	Iberdrola	1014.374701
5	Naturgy	1010.790115
6	Acciona	1010.063559
7	Amadeus	1009.133965
8	Red Eléctrica	1008.084074
9	Endesa	1008.056395
10	Cellnex	1007.532122
11	Bankinter	1006.104129
12	Acciona Energía	1005.813953
13	Enagás	1001.953125
14	Merlin Properties	1000.000000
15	Inmobiliaria Colonial	997.530975
16	Inditex	996.555118
17	Indra	994.594595
18	Ferrovial	993.461381
19	Acs	991.856678
20	Solaria	990.561588
21	Telefónica	990.299278
22	Sacyr	989.682540
23	Rovi	989.600000
24	Aena	988.243198
25	IAG	985.401460
26	Mapfre	985.263158
27	Caixabank	978.317152
28	Banco Sabadell	972.585191
29	Fluidra	967.741935
30	Banco Santander	964.624131
31	BBVA	960.046864
32	Repsol	955.286093
33	Acerinox	955.058444
34	Arcelormittal	913.705584

Figura B.2: Ejemplo de ranking relativo a la perspectiva 1

Similarmente a lo anterior, se adjunta cómo se construyen los rankings para la perspectiva 2 (B.3). En estos DataFrame se almacena el valor evolutivo de la inversión inicial en cada empresa relativa a dos sesiones, la anterior y la actual a analizar, y se calcula la última columna mediante la función *insertarcolumnaDiferencia*.

	Nombre	Número de acciones	Valor sesión anterior	Valor sesión actual	Diferencia valor
0	Acciona	5.290610	1000.0	1010.063559	10.063559
1	Acciona Energía	29.069767	1000.0	1005.813953	5.813953
2	Acerinox	100.765820	1000.0	955.058444	-44.941556
3	Acs	40.716612	1000.0	991.856678	-8.143322
4	Aena	6.718173	1000.0	988.243198	-11.756802
5	Amadeus	16.914750	1000.0	1009.133965	9.133965
6	ArcelorMittal	33.840948	1000.0	913.705684	-86.294416
7	Banco Santander	315.855970	1000.0	964.624131	-35.375869
8	Banco Sabadell	1281.065847	1000.0	972.585191	-27.414809
9	Bankinter	179.533214	1000.0	1006.104129	6.104129
10	BBVA	199.760288	1000.0	980.946864	-39.053136
11	Caixabank	323.624595	1000.0	978.317152	-21.682848
12	Cellnex	22.153301	1000.0	1007.532122	7.532122
13	Enagás	48.828125	1000.0	1001.953125	1.953125
14	Endesa	50.352467	1000.0	1008.056395	8.056395
15	Ferrovial	40.866367	1000.0	993.461381	-6.53619
16	Fluidra	38.402458	1000.0	987.741935	-32.258065
17	Grifols	61.614295	1000.0	1047.443007	47.443007
18	IAG	561.482313	1000.0	985.401460	-14.586540
19	Iberdrola	95.831337	1000.0	1014.374701	14.374701
20	Inditex	49.212598	1000.0	996.555118	-3.446882
21	Indra	108.108108	1000.0	994.594565	-5.405405
22	Inmobiliaria Colonial	123.001230	1000.0	997.530975	-2.480025
23	Mapfre	526.315789	1000.0	985.263158	-14.736842
24	Meliá Hotels	142.959257	1000.0	1033.595425	33.595425
25	Merlin Properties	92.421442	1000.0	1000.000000	0.000000
26	Naturgy	34.806822	1000.0	1010.790115	10.790115
27	Pharma Mar	13.106160	1000.0	1015.203145	15.203145
28	Red Eléctrica	53.893829	1000.0	1008.084074	8.084074
29	Repsol	75.789283	1000.0	955.288093	-44.713907
30	Rovi	16.000000	1000.0	989.600000	-10.400000
31	Sacyr	396.825397	1000.0	989.682540	-10.317460
32	Siemens Gamesa	65.963061	1000.0	1027.044855	27.044855
33	Solaria	47.192072	1000.0	990.561586	-9.438414
34	Telefónica	208.398349	1000.0	990.290278	-9.700722

Figura B.3: Ejemplo de ranking relativo a la perspectiva 2

En este caso B.3, el ranking se construiría con la primera y la última columna pero se puede observar que el valor que se toma para el ranking es la diferencia entre las dos columnas centrales, relativas al valor de la sesión anterior y de la sesión actual.

Comparando B.2 con B.3 la distinción es que en esta segunda se trabaja con la diferencia en el valor o no con el valor en sí. Es decir, ambos rankings son los primeros de cada perspectiva, donde su sesión anterior fue la primera donde se tenía 1000 € invertidos en cada empresa. Yendo al caso de Grifols, por ejemplo, al cierre de la sesión 2, el valor de la inversión pasó a ser de 1047.443007 €. Para la perspectiva 1 es necesario ese valor pero para la perspectiva dos se toma 47.443007 €, que es la diferencia con respecto de la sesión anterior.

Siguiendo la línea de los dos casos anteriores, se adjunta un ejemplo de ranking de la tercera perspectiva B.4.

## Capítulo B. Resultados obtenidos en la práctica

---

	Nombre	Sesión anterior	Sesión actual	% Variacion cierre sesión
0	Acciona	188.8000	190.7000	1.0064
1	Acciona Energía	34.4000	34.6000	0.5814
2	Acerinox	9.9240	9.4780	-4.4942
3	Acs	24.5600	24.3600	-0.8143
4	Aena	148.8500	147.1000	-1.1757
5	Amadeus	59.1200	59.6600	0.9134
6	Arcelormittal	29.5500	27.0000	-8.6294
7	Banco Santander	3.1860	3.0540	-3.5376
8	Banco Sabadell	0.7806	0.7562	-2.7415
9	Bankinter	5.5700	5.8040	0.6104
10	BBVA	5.0060	4.8105	-3.9053
11	Caixabank	3.0900	3.0230	-2.1683
12	Cellnex	45.1400	45.4800	0.7532
13	Enagás	20.4800	20.5200	0.1953
14	Endesa	19.8800	20.0200	0.8056
15	Ferrovial	24.4700	24.3100	-0.6539
16	Fluidra	26.0400	25.2000	-3.2258
17	Grifols	16.2300	17.0000	4.7443
18	IAG	1.7810	1.7550	-1.4599
19	Iberdrola	10.4350	10.5850	1.4375
20	Inditex	20.3200	20.2500	-0.3445
21	Indra	9.2500	9.2000	-0.5405
22	Inmobiliaria Colonial	8.1300	8.1100	-0.2480
23	Mapfre	1.9000	1.8720	-1.4737
24	Meliá Hotels	6.9950	7.2300	3.3595
25	Merlin Properties	10.8200	10.8200	0.0000
26	Naturgy	28.7300	29.0400	1.0790
27	Pharma Mar	76.3000	77.4600	1.5203
28	Red Eléctrica	18.5550	18.7050	0.8084
29	Repsol	13.1950	12.8050	-4.4714
30	Rovi	62.5000	61.8500	-1.0400
31	Sacyr	2.5200	2.4940	-1.0317
32	Siemens Gamesa	15.1600	15.5700	2.7045
33	Solaria	21.1900	20.9900	-0.9438
34	Telefónica	4.8450	4.7980	-0.9701

Figura B.4: Ejemplo de ranking relativo a la perspectiva 3

A modo similar que en el ejemplo B.3, para esta perspectiva, como se puede ver en B.4, se han añadido dos columnas centrales que son las que se han empleado para la construcción del valor que trabaja el ranking, que es la última columna. Esta junto a la primera forman lo que sería el primer ranking.

Adicionalmente, se adjunta el último ranking relativo a la primera perspectiva debido a que es empleado durante el análisis para la extracción de conclusiones ya que permite ver qué empresas lideran el ranking final en cuanto a rentabilidad positiva.

---

	Nombre	Valor
0	Siemens Gamesa	1177.110818
1	Caixabank	1132.686084
2	Bankinter	1108.642729
3	Inditex	1096.948819
4	Acciona Energía	1055.232558
5	Banco Sabadell	1011.529593
6	Repsol	1001.894657
7	BBVA	998.701558
8	Acs	985.798046
9	Indra	952.432432
10	Ferrovial	932.570494
11	Iberdrola	913.080977
12	Sacyr	909.523810
13	Acciona	880.296610
14	Mapfre	855.789474
15	Acerinox	854.292624
16	Amadeus	843.200271
17	Banco Santander	839.545187
18	Naturgy	825.269753
19	Red Eléctrica	818.916734
20	ArcelorMittal	788.663283
21	Endesa	749.244713
22	Enagás	742.675781
23	Merlin Properties	729.205176
24	IAG	728.242560
25	Aena	709.439033
26	Pharma Mar	703.800786
27	Rovi	698.240000
28	Solaria	683.105238
29	Telefónica	670.588235
30	Cellnex	638.015064
31	Meliá Hotels	629.592566
32	Inmobiliaria Colonial	597.047970
33	Grifols	501.047443
34	Fluidra	472.350230

Figura B.5: Último ranking relativo a la perspectiva 1

Como ejemplos de escalabilidad de las ideas desarrolladas para la construcción de los rankings, se adjunta el código de creación de las matrices de rankings, en este caso para la perspectiva 1 [B.6](#), y un fragmento de dicha matriz construida [B.7](#), también de la perspectiva 1.

## Capítulo B. Resultados obtenidos en la práctica

---

```
#####
#Escalabilidad del modelo #####
#Creación de un Data Frame con tantas filas como empresas y tantas columnas como rankings
df_rankings_inv = pd.DataFrame(columns=rankings,index=classification)

#Bucle
for i in range(1,n_sesiones):
    data_aux_inv = [[ 'Acciona',df_table_n_stocks.iloc[0,1]*df_acciona.iloc[i,1]],
                    [ 'Acciona Energía',df_table_n_stocks.iloc[1,1]*df_accionaEnergia.iloc[i,1]],
                    [ 'Acerinox',df_table_n_stocks.iloc[2,1]*df_acerinox.iloc[i,1]],
                    [ 'Acs',df_table_n_stocks.iloc[3,1]*df_acs.iloc[i,1]],
                    [ 'Aena',df_table_n_stocks.iloc[4,1]*df_aena.iloc[i,1]],
                    [ 'Amadeus',df_table_n_stocks.iloc[5,1]*df_amadeus.iloc[i,1]],
                    [ 'Arcelormittal',df_table_n_stocks.iloc[6,1]*df_arcelormittal.iloc[i,1]],
                    [ 'Banco Santander',df_table_n_stocks.iloc[7,1]*df_santander.iloc[i,1]],
                    [ 'Banco Sabadell',df_table_n_stocks.iloc[8,1]*df_sabadell.iloc[i,1]],
                    [ 'Bankinter',df_table_n_stocks.iloc[9,1]*df_bankinter.iloc[i,1]],
                    [ 'BBVA',df_table_n_stocks.iloc[10,1]*df_bbva.iloc[i,1]],
                    [ 'Caixabank',df_table_n_stocks.iloc[11,1]*df_caixabank.iloc[i,1]],
                    [ 'Cellnex',df_table_n_stocks.iloc[12,1]*df_cellnex.iloc[i,1]],
                    [ 'Enagás',df_table_n_stocks.iloc[13,1]*df_enagas.iloc[i,1]],
                    [ 'Endesa',df_table_n_stocks.iloc[14,1]*df_endesa.iloc[i,1]],
                    [ 'Ferrovial',df_table_n_stocks.iloc[15,1]*df_ferrovial.iloc[i,1]],
                    [ 'Fluidra',df_table_n_stocks.iloc[16,1]*df_fluidra.iloc[i,1]],
                    [ 'Grifols',df_table_n_stocks.iloc[17,1]*df_grifols.iloc[i,1]],
                    [ 'IAG',df_table_n_stocks.iloc[18,1]*df_IAG.iloc[i,1]],
                    [ 'Iberdrola',df_table_n_stocks.iloc[19,1]*df_iberdrola.iloc[i,1]],
                    [ 'Inditex',df_table_n_stocks.iloc[20,1]*df_inditex.iloc[i,1]],
                    [ 'Indra',df_table_n_stocks.iloc[21,1]*df_indra.iloc[i,1]],
                    [ 'Inmobiliaria Colonial',df_table_n_stocks.iloc[22,1]*df_colonial.iloc[i,1]],
                    [ 'Mapfre',df_table_n_stocks.iloc[23,1]*df_mapfre.iloc[i,1]],
                    [ 'Meliá Hotels',df_table_n_stocks.iloc[24,1]*df_melia.iloc[i,1]],
                    [ 'Merlin Properties',df_table_n_stocks.iloc[25,1]*df_merlin.iloc[i,1]],
                    [ 'Naturgy',df_table_n_stocks.iloc[26,1]*df_naturgy.iloc[i,1]],
                    [ 'Pharma Mar',df_table_n_stocks.iloc[27,1]*df_pharmaMar.iloc[i,1]],
                    [ 'Red Eléctrica',df_table_n_stocks.iloc[28,1]*df_redElectrica.iloc[i,1]],
                    [ 'Repsol',df_table_n_stocks.iloc[29,1]*df_repsol.iloc[i,1]],
                    [ 'Rovi',df_table_n_stocks.iloc[30,1]*df_rovi.iloc[i,1]],
                    [ 'Sacyr',df_table_n_stocks.iloc[31,1]*df_sacyr.iloc[i,1]],
                    [ 'Siemens Gamesa',df_table_n_stocks.iloc[32,1]*df_siemens.iloc[i,1]],
                    [ 'Solaria',df_table_n_stocks.iloc[33,1]*df_solaria.iloc[i,1]],
                    [ 'Telefónica',df_table_n_stocks.iloc[34,1]*df_telefonica.iloc[i,1]]]
    df_aux_inv = pd.DataFrame(data_aux_inv,columns=[ 'Nombre','Valor'])
    ordenarRanking(df_aux_inv)
    df_rankings_inv.iloc[:,i-1]=df_aux_inv.iloc[:,0]
df_rankings_inv
```

Figura B.6: Código relativo a la matriz de rankings de la perspectiva 1

	U	1	2	3	4	5	6	7	8	9	12B	Start Gen
1	Melis Hotels	Melis Hotels	Melis Hotels	Melis Hotels	Hed Elektrica	Iberdrola	Hed Elektrica	Hed Elektrica	Hed Elektrica	Hed Elektrica	Gefusa	Acme Energia
2	Siemens Gmessa	Femontel	Natura	Natura	Natura	Natura	Natura	Iberdrola	Iberdrola	Iberdrola	Gefusa	Acme Energia
3	Intermar Mar	Bancolombia	Natura	Natura	Hovar	Iberdrola	Iberdrola	Iberdrola	Iberdrola	Iberdrola	Intermar	Catolitico
4	Iberdrola	Hed Elektrica	Acme Energia	Repsol	Rosif	Sacyr	Gefusa	Sacyr	Sacyr	Hed Elektrica	Bancolombia	Bancolombia
5	Natura	Embras	Celtric	Natura	Natura	Neel Amadeus	Hed Elektrica	Hed Elektrica	Hed Elektrica	Hed Elektrica	Hed Elektrica	Hed Elektrica
6	Acmea	Natura	Enebras	Sacyr	Acme Energia	Hed Elektrica	Amadeus	Gefusa	Indesa	Amadeus	Femontel	Hed Elektrica
7	Amadeus	Immobiliare Colom	Enebras	Enebras	Sacyr	Sacyr	Gefusa	Catolbank	Enebras	Enebras	Enebras	Acmea
8	Hed Elektrica	Acmea	Sacyr	Gefusa	Amadeus	Acme Energia	Sacyr	Endesa	Hovar	Acmea	Banca Satelital	Banca Satelital
9	Endesa	Aca	Femontel	Endesa	Acimont	Bancolombia	Acme Energia	Acme Energia	Natura	Amadeus	Hovar	Acmea
10	Celtric	Amadeus	Immobiliare Colom	Femontel	Bancolombia	Endesa	Siemens Gmessa	Siemens Gmessa	Sacyr	Sacyr	Aca	Acmea
11	Bancolombia	Indesa	Aca	Acme Energia	Siemens Gmessa	Hovar	Enebras	Siemens Gmessa	Natura	Indesa	Iberdrola	Iberdrola
12	Acmea Energia	Sacyr	Solaris	Solaris	BBVA	Enebras	Itival	Hovar	Endesa	Sacyr	Sacyr	Sacyr
13	Enebras	Celtric	Indesa	Acmea	Endesa	Indesa	Bancolombia	Indesa	Femontel	Acmea Energia	Acmea Energia	Acmea Energia
14	Martin Properties	Rosif	Acme Energia	Aca	Solaris	Acimont	Indesa	Femontel	Aca	Aca	Indesa	Indesa
15	Immobiliare Colom	Enebras	Hovar	Amadeus	Femontel	Aca	Endesa	Amadeus	Indevisa	Natura	Mapfe	Mapfe
16	Indesa	Martin Properties	Amadeus	Acimont	Enebras	Catolbank	Aca	Aca	Indevisa	Indevisa	Red Electric	Acmea
17	Indesa	Aera	Bancolombia	Immobiliare Colom	Fludra	Natura	Acme Energia	Acme Energia	Acimont	Femontel	Acmea	Acmea
18	Aca	Solaris	Gefusa	Indesa	Hovar	Aca	Siemens Gmessa	BBVA	Catolbank	Solaris	Solaris	Natura
19	Solaris	Acme Energia	Acimont	Indesa	Celtric	Femontel	Acme Energia	Acme Energia	Celtric	Acmea	Endesa	Endesa
20	Solaris	Acme Energia	Acimont	Indesa	Celtric	Acme Energia	BBVA	Solaris	Telefónica	Immobiliare Colom	Immobiliare Colom	Solaris
21	Telefónica	Mapfe	Siemens Gmessa	Siemens Gmessa	Acmea	BBVA	Solaris	Telefónica	Immobiliare Colom	Immobiliare Colom	Indevisa	Indevisa
22	Sacyr	Telefónica	Mapfe	Martin Properties	Celtric	Indesa	Acme Energia	Acme Energia	Celtric	Acmea	Acimont	Acimont
23	Hovar	Siemens Gmessa	Mapfe	Martin Properties	Celtric	Celtric	Celtric	Celtric	Hovar	Bancolombia	Bancolombia	Bancolombia
24	Aera	Bancolombia	Repcol	Telefónica	Gefusa	Solaris	Celtric	IAG	IAG	Celtric	Mapfe	Mapfe
25	IAG	Hudha	Telefónica	IAG	Immobiliare Colom	Pharma Mar	Indevisa	Indevisa	Indevisa	Indevisa	Pharma Mar	Pharma Mar
26	Mapfe	Repcol	Fludra	Fludra	BBVA	Indevisa	Indevisa	BBVA	BBVA	BBVA	Mapfe	Mapfe
27	Catolbank	Acimont	Catolbank	Catolbank	Mapfe	Martin Properties	IAG	Fludra	Pharma Mar	Fludra	Catolbank	Catolbank
28	Banca Satelital	Catolbank	Aera	BBVA	Pharma Mar	Pharma Mar	Pharma Mar	Pharma Mar	Pharma Mar	Pharma Mar	Banca Satelital	Banca Satelital
29	Fludra	IAG	Acimont	Pharma Mar	Telefónica	Martin Properties	Banca Satelital	Fludra	Aera	Mapfe	Hovar	Hovar
30	Banca Satelital	BBVA	Pharma Mar	Acimont	Banca Satelital	Banca Satelital	Acimont	Banca Satelital	Banca Satelital	BBVA	Aera	IAG
31	BBVA	Pharma Mar	IAG	Acimont	BBVA	Aera	Acimont	Acimont	Mapfe	Acimont	Banca Satelital	Banca Satelital
32	Repcol	Acimont	BBVA	Banca Satelital	Mapfe	Aera	Mapfe	Acimont	Acimont	Acimont	Banca Satelital	Immobiliare Colom
33	Acimont	Banca Satelital	Banca Satelital	BBVA	Mapfe	Aera	Mapfe	Aera	Mapfe	IAG	Hudha	Hudha
34	Acimont	Banca Satelital	Banca Satelital	BBVA	Banca Satelital	Banca Satelital	Banca Satelital	Banca Satelital	Banca Satelital	Banca Satelital	Gefusa	Gefusa

Figura B.7: Matriz de rankings relativa a la perspectiva 1

Análogamente a las dos últimas ilustraciones, se repite la exposición de código B.6 y resultado para la matriz de adyacencia B.9 de los grafos originales de las perspectivas, también en este caso relativos a la perspectiva 1.

```

#Definición de la matriz de adyacencia del grafo, donde sus filas y sus columnas son las acciones
adj_matrix_inv = pd.DataFrame(columns=stocks,index=stocks)

#Dicha matriz se rellena de 0's para asegurarse de que posteriormente se puede llenar de la forma que se desea en el estudio
for i in range(n_stocks):
    for j in range(n_stocks):
        adj_matrix_inv.iloc[i,j]=0

#Iteraciones para completarla con el DataFrame de rankings: Se toma una acción y cada una de sus sucesivas a lo largo
#de todos los rankings, tomados de dos en dos de forma contigua
for k in range(n_sessions-2):
    for i in range(n_stocks):
        for j in range(i+1,n_stocks):
            if (seCruza(df_rankings_inv.iloc[:,k],df_rankings_inv.iloc[:,k+1],stocks[i],stocks[j])):
                adj_matrix_inv.iloc[i,j] = adj_matrix_inv.iloc[i,j]+1
                adj_matrix_inv.iloc[j,i] = adj_matrix_inv.iloc[j,i]+1

adj_matrix_inv

```

Figura B.8: Código relativo a la matriz de adyacencia de la perspectiva 1

## Capítulo B. Resultados obtenidos en la práctica

---

	Acciona	Acciona Energía	Acerinox	Acs	Aena	Amadeus	Arcelormittal	Banco Santander	Banco Sabadell	Bankinter	...	Merlin Properties	Naturgy	Pharma Mar	Red Eléctrica
Acciona	0	1	4	5	2	4	2	2	5	9	...	8	13	3	6
Acciona Energía	1	0	8	10	2	5	0	0	8	12	...	2	5	1	5
Acerinox	4	8	0	12	1	15	0	3	4	18	...	5	13	3	9
Acs	5	10	12	0	0	9	0	0	15	18	...	7	13	7	5
Aena	2	2	1	0	0	2	9	13	19	2	...	8	2	17	0
Amadeus	4	5	15	9	2	0	2	2	7	5	...	18	11	13	4
Arcelormittal	2	0	0	0	9	2	0	10	8	0	...	9	4	23	4
Banco Santander	2	0	3	0	13	2	10	0	0	0	...	1	3	9	1
Banco Sabadell	5	8	4	15	19	7	8	0	0	12	...	7	9	5	9
Bankinter	9	12	16	16	2	5	0	0	12	0	...	10	5	5	7
BBVA	11	0	8	11	13	7	6	5	14	8	...	9	7	11	3
Caixabank	5	7	8	15	1	11	0	0	0	3	...	11	11	5	5
Cellnex	2	1	7	9	11	14	9	7	5	9	...	17	0	20	0
Enagás	2	8	19	7	0	20	1	3	11	14	...	8	2	5	4
Endesa	4	3	13	13	0	20	7	1	13	17	...	6	10	3	4
Ferrovial	19	4	6	7	0	7	2	0	7	10	...	3	13	1	11
Fluidra	4	0	1	2	8	0	7	9	2	0	...	2	2	12	0
Grifols	7	5	5	5	3	5	5	5	3	7	...	1	7	5	9
IAG	2	2	1	0	3	0	13	3	1	0	...	4	0	9	0
Iberdrola	4	3	2	3	0	2	2	0	7	3	...	2	10	3	14
Inditex	3	17	6	8	0	7	0	0	6	14	...	1	5	1	3
Indra	5	6	12	9	2	9	0	0	7	8	...	7	7	1	19
Inmobiliaria Colonial	10	2	1	1	1	2	15	7	3	4	...	4	2	10	0
Mapfre	0	0	12	2	17	9	6	4	17	2	...	7	7	13	1
Meliá Hotels	5	5	1	7	9	5	5	5	11	13	...	13	5	5	5
Merlin Properties	8	2	5	7	8	18	9	1	7	10	...	0	6	15	0
Naturgy	13	5	13	13	2	11	4	3	9	5	...	6	0	5	6
Pharma Mar	3	1	3	7	17	13	23	9	5	5	...	15	5	0	3
Red Eléctrica	6	5	9	5	0	4	4	1	9	7	...	0	6	3	0
Repsol	9	2	8	7	1	5	0	1	10	12	...	5	13	3	15
Rovi	8	8	5	6	9	12	23	3	9	8	...	10	14	16	2
Sacyr	7	10	16	12	0	7	0	0	13	14	...	11	11	5	7
Siemens Gamesa	4	10	8	6	4	4	2	0	2	4	...	6	6	2	2
Solaria	20	8	7	20	5	8	1	1	11	14	...	2	18	2	24
Telefónica	8	6	7	2	7	12	5	3	3	12	...	12	6	8	6

Figura B.9: Matriz de adyacencia de la perspectiva 1

Por último, y cerrando este apéndice, se presenta el código relativo a la optimización de la representación de los grafos B.10, en este caso nuevamente para el grafo relativo a la perspectiva 1.

---

```

#Primero se genera el tipo de diseño del grafo para que los nodos estén bien diferenciados en la representación
pos_inv = nx.circular_layout(G_inv)

#Para representar las aristas se va a representar el grosor de la misma en proporcion a su peso
edgew_inv = [0.15*G_inv.get_edge_data(u, v)['weight'] for u, v in G_inv.edges()]
#Con la siguiente expresión se consigue establecer el ancho (x) y el alto (y) del espacio donde se va a representar el grafo
x_inv, y_inv = plt.subplots(figsize=(12, 12))
#Título del grafo
y_inv.set_title("Competitiviy graph: Perspective 1", dict(fontweight="bold", fontsize=22, color="black"), loc="center")

#Representación de los objetos del grafo
    #Aristas
nx.draw_networkx_edges(G_inv, pos_inv, width=edgew_inv, edge_color="navy", style='--')
    #Nodos
nx.draw_networkx_nodes(G_inv, pos_inv, node_size=700, node_shape='o', node_color="gold", linewidths=1, edgecolors="black")
    #Etiquetas sobre nodos
nx.draw_networkx_labels(G_inv, pos_inv, font_size=11, bbox=dict(ec="black", boxstyle="round", facecolor="white", pad=0.3,
alpha=0.6), font_weight='normal', font_family='Verdana', horizontalalignment='center',
verticalalignment='center', clip_on=False)

#Modificaciones sobre el espacio de representación para optimizar la representación
x_inv.tight_layout()
plt.axis("off") #Para no representar los ejes
y_inv.margins(0.05, 0.05)
plt.savefig("G1.png") #Para guardar la imagen para exportarla al documento de texto
plt.show() #Para mostrar el grafo

```

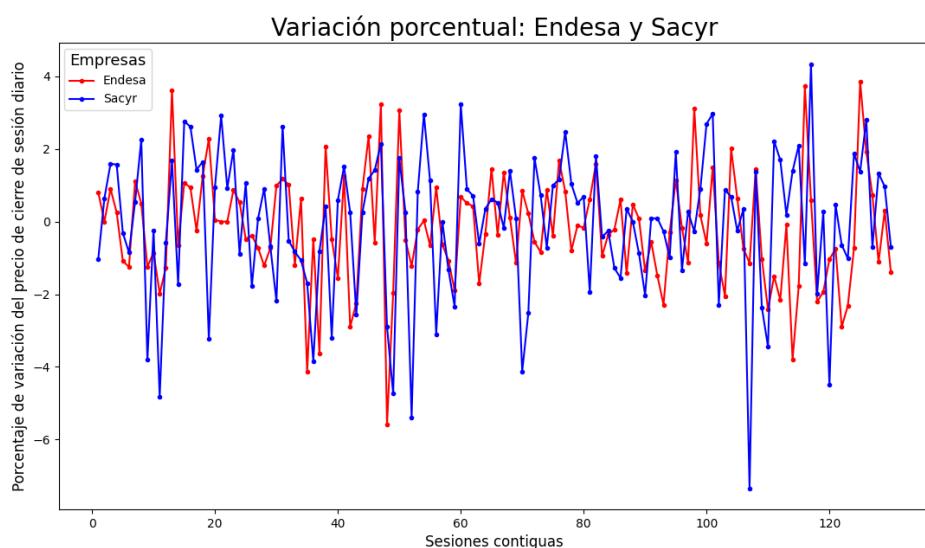
Figura B.10: Ejemplo de código para la representación óptima de los grafos relativo a la perspectiva 1

# C

## Recursos para la extracción de conclusiones

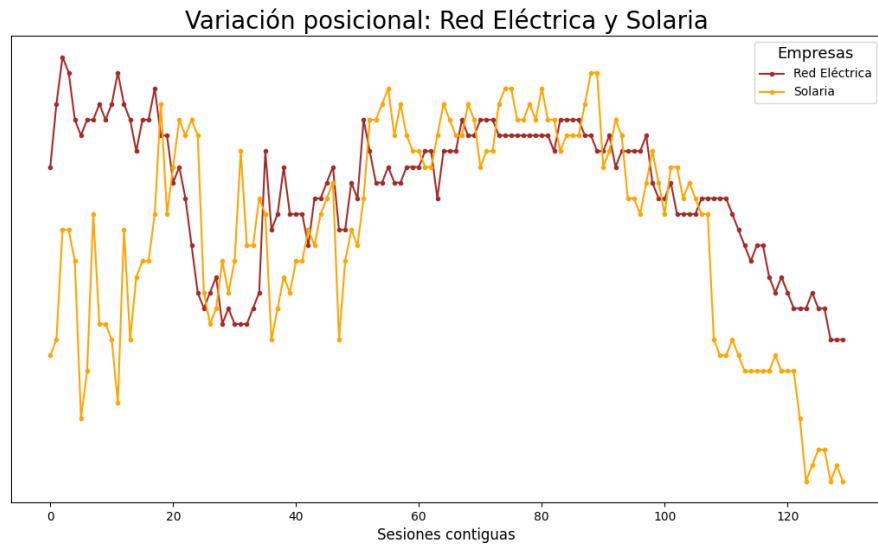
En este apéndice se recogen dos casos comentados en conclusiones con gráficas y cálculos que verifican las conclusiones extraídas anteriormente.

### C.1. Caso Endesa y Sacyr



Relativo a las empresas Endesa y Sacyr se ha observado que un movimiento ascendente o descendente en el valor de una implica, en numerosas ocasiones, un movimiento en el mismo sentido en la otra empresa. Este suceso se ha dado en el 57,6 % de las sesiones.

## C.2. Caso Solaria y Red Eléctrica



Del primer enfoque se ha extraído que Solaria y Red Eléctrica compartían comunidad debido a su alta competitividad.

Como conclusión, se ha observado que Red Eléctrica suele estar en posiciones superiores a Solaria en cuanto al valor de la inversión y que, cuando esta segunda empresa adelanta a la primera, solo unas sesiones después Red Eléctrica vuelve a ponerse por delante.

# D

## Aplicación del algoritmo de Girvan-Newman sobre los grafos comentados

En este apéndice se recogen las iteraciones relativas a la aplicación del algoritmo de Girvan-Newman sobre los grafos desarrollados que se han comentado en el punto 5. Concretamente, los relativos a la primera y segunda perspectiva. Adicionalmente, estos grafos están acompañados de una explicación sobre los resultados obtenidos tras la aplicación del algoritmo sobre ellos.

### D.1. Perspectiva 1

En este primer punto se recogen los detalles relativos a la perspectiva 1 en sus subvariantes 1.1, 1.2 y 1.3.

- Para la subvariante 1.1 aparece en la iteración 2 desde la "mega-comunidad" una comunidad con 5 nodos, concretamente son Arcelormittal, Banco Santander, Fluidra, IAG e Inmobiliaria Colonial, que se mantienen en la misma comunidad hasta iteraciones avanzadas.

En la imagen D.1 se puede ver como a pesar de haber generado más "mono-comunidades" posteriormente que no aportan información alguna, la comunidad **morada**, comentada anteriormente, sigue manteniéndose. Es por ello que es candidata para poder extraer conclusiones de la misma.

- En las subvariantes 1.2 y 1.3 se generan varias comunidades diferenciadas ya en la primera iteración. A partir de estas, las iteraciones sucesivas sirven para ir separando comunidades de 3 y 4 elementos e ir dejando nodos aislados.

En 1.2 (D.2 (a)) se ha tomado la iteración 5, donde destacan las comunidades:

1. La comunidad **morada** formada por Amadeus y Enagás con una fuerte conexión que se mantiene durante las 6 primeras iteraciones. Además, originariamente, esta comunidad estaba unida con la comunidad **rosa** formada en este caso por Endesa y Sacyr, que son las empresas con mayor competitividad del grafo.
2. El caso de los nodos **lima**, formada por Iberdrola, Solaria y Red Eléctrica, que en iteraciones anteriores formaba una comunidad con Ferrovial. En esta comunidad son las dos últimas empresas las que mantienen la relación hasta la última iteración.
3. La comunidad **amarilla** integrada por Arcelormittal y Rovi que anteriormente formaban una sola comunidad con la actualmente color **cyan** que integra Cellnex y Pharma Mar.
4. El resto de comunidades se deben analizar desde el punto de vista de si tienen conexión con algún nodo de alguna comunidad o no, pero al ser "mono-comunidades" tienen muchísima menor influencia sobre los nodos de otra comunidad.

Para el caso **1.3 (D.2 (b))** se tiene comunidades bien diferenciadas porque desde un principio hay nodos que están en una misma comunidad que solamente se relacionan entre ellos, por lo que se facilita la detección de comunidades. Para este caso, en su iteración 3, se tiene:

1. La comunidad **roja** con Arcelormittal y Rovi como componentes, anteriormente formando una comunidad de 3 con Pharma Mar.
2. Sacyr junto a Endesa forman la comunidad **verde** que mantienen hasta dos iteraciones más tarde.
3. Como en el caso **1.2**, Solaria y Red Eléctrica forman la comunidad más fuerte que se mantiene hasta la última iteración. En este caso es la **rosa**. Además, en un inicio aparecían con Iberdrola formando una única comunidad.
4. Los comentarios para las comunidades con un solo nodo son análogos al caso **1.2**.

**Application of Girvan Newman algorithm: Perspective 1.1, iteration 7**

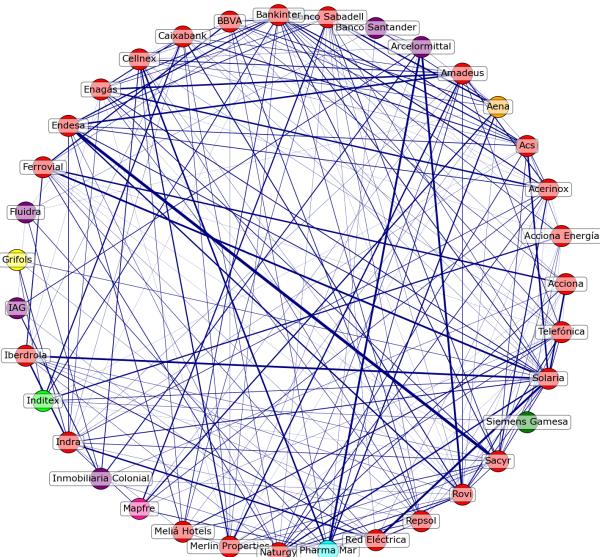


Figura D.1: Aplicación del algoritmo de Girvan-Newman a la perspectiva 1.1: Iteración 7

Capítulo D. Aplicación del algoritmo de Girvan-Newman sobre los grafos comentados

---

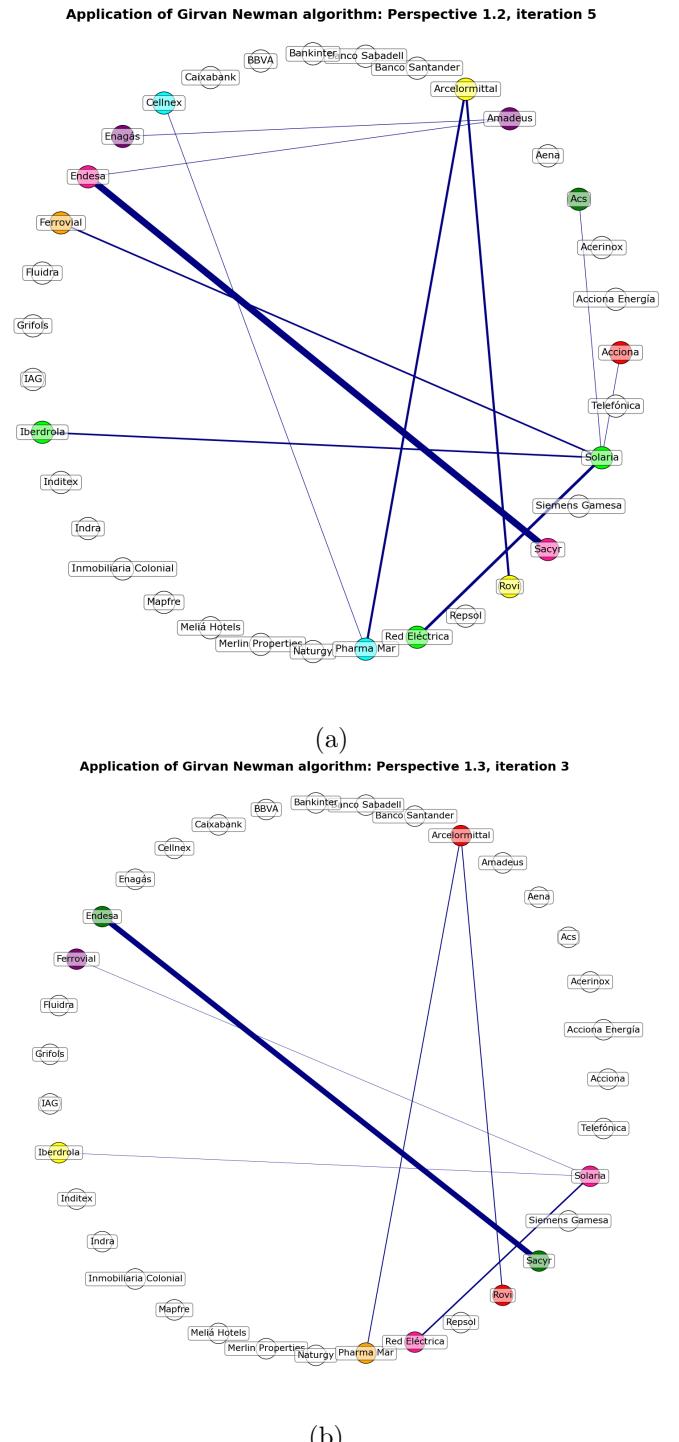


Figura D.2: Aplicación del algoritmo de Girvan-Newman a las perspectivas 1.2 y 1.3: Iteración 5 e Iteración 3, respectivamente

Un resumen global de los resultados obtenidos para esta perspectiva pueden consultarse en la sección *Resultados 5*.

## D.2. Perspectiva 2

En este último punto se recogen los detalles relativos a la perspectiva 2 en sus subvariantes **2.2** y **2.3**.

En estas subvariantes se generan al principio dos comunidades con muchos nodos cada una y es a partir de estas donde se comienzan a generar comunidades más pequeñas.

- En la subvariante **2.2** se ha tomado la iteración 6 (**D.3 (a)**) debido a que es justo esta la última en la que las comunidades tienen más de un nodo. Las comunidades a tratar son:
  1. **Roja**: Cuenta con Acciona, Ferrovial, IAG y Rovi. De la cual se desprenderán en comunidades unitarias Ferrovial e IAG en posteriores iteraciones.
  2. **Verde**: Integrada por Acerinox, Aena, Bankinter, BBVA, Endesa, Iberdrola, Indra y Red Eléctrica. Esta posteriormente se dividirá en dos comunidades formadas por Bankinter, Endesa, Iberdrola e Indra, la primera, y por el resto de empresas la segunda.
  3. **Amarilla**: Integrada solo por Acs y Arcelormittal. Sin embargo, hay que añadir que anteriormente formaban una sola comunidad con la de color **cyan**: Banco Santander, Sacyr, Merlin Properties, Naturgy y Pharma Mar. De esta última, serán Banco Santander y Merlin Properties, más tarde, las que abandonarán la comunidad para formar una por sí solas.
  4. **Morada**: Formada por Amadeus, Inmobiliaria Colonial, Fluidra y Siemens Gamesa, la cual se desglosará más tarde en las dos primeras y las dos segundas, presentando una mayor fuerza la segunda.
  5. **Rosa**: Contiene Caixabank, Solaria, Inditex y Repsol. Esta comunidad procede de una separación de la comunidad roja pero nuevamente se separará en dos comunidades formadas por las dos primeras y las dos últimas empresas, siendo la última comunidad la que se mantenga más iteraciones.
  6. **Naranja**: Esta última cuenta con Cellnex, Enagás, Mapfre y Meliá Hotels. Nuevamente, se separarán en dos comunidades de dos elementos, los anteriores respectivamente en ese orden y será precisamente la última la que más se mantenga a lo largo de las iteraciones.
- Por otro lado, para el caso **2.3** en la iteración 7 (**D.3 (b)**), se tiene:
  1. **Roja**: Formada por Acciona, Mapfre, Meliá Hotels, Pharma Mar y Rovi. Posteriormente, en varias iteraciones resultarán tres comunidades donde Acciona está sola, Mapfre y Meliá quedan unidas y las dos empresas restantes forman la tercera. Además, esta tercera será la última comunidad en separarse para terminar de aplicar el algoritmo.
  2. **Morada**: Cuenta con Acerinox, Acs, Arcelormittal y Endesa. Este caso proviene de una división con la comunidad **cyan** que la forman Bankinter, Iberdrola e Indra. Ambas se separarán posteriormente formando, en un primer momento, una "mono-comunidad" la empresa Bankinter y dos comunidades diferenciadas Acs con Arcelormittal y Endesa con Acerinox. De ellas, será esta última la que menos dure en las sucesivas iteraciones y la que proviene de la comunidad **cyan** la que más.
  3. **Naranja**: Contiene Aena, Amadeus e Inmobiliaria Colonial, siendo la primera la que antes se desapegue de las otras dos.

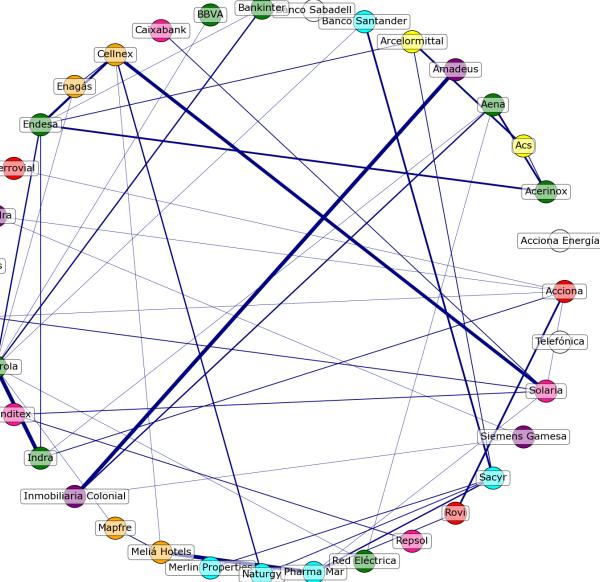
## Capítulo D. Aplicación del algoritmo de Girvan-Newman sobre los grafos comentados

---

4. De una comunidad anterior surgen dos comunidades. La **amarilla** integrada por Banco Santander, Merlin Properties y Sacyr, y la **rosa** que cuenta con Cellnex, Enagás y Naturgy, siendo Banco Santander y Enagás las primeras en formar una comunidad por sí solas.
5. **Verde:** La constituyen Caixabank, IAG y Solaria y procede de una comunidad con la unión de las empresas localizadas en la **lima**, comunidad de apenas dos empresas, Inditex y Repsol. De la **verde** serán IAG y Solaria las que mantengan la comunidad.

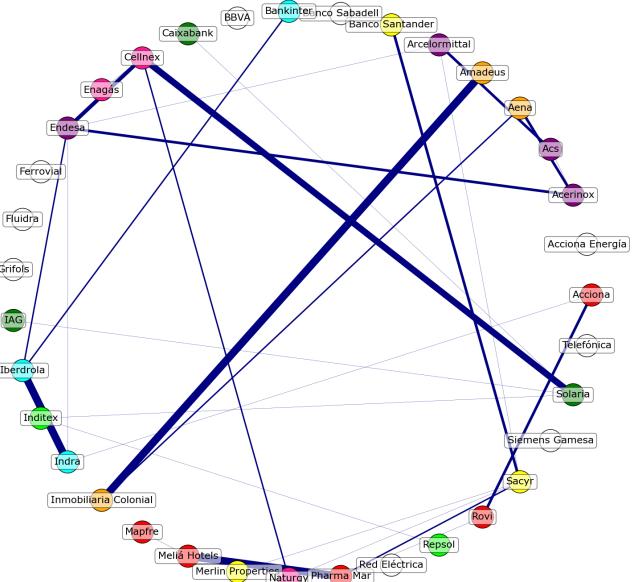
Un resumen global de los resultados obtenidos para esta perspectiva pueden consultarse en la sección *Resultados* 5.

Application of Girvan Newman algorithm: Perspective 2.2, iteration 6



(a)

Application of Girvan Newman algorithm: Perspective 2.3, iteration 7



(b)

Figura D.3: Aplicación del algoritmo de Girvan-Newman a las perspectivas 2.2 y 2.3: Iteración 6 e Iteración 7, respectivamente

# E

## Repositorio de archivos

En este apéndice se adjunta el repositorio de GitHub donde se recogen todos los archivos empleados para este estudio y todos los archivos que se han generado durante el mismo.

Como se puede ver en la figura E.1, el repositorio contiene una carpeta llamada *Data*, con la información relativa a cada una de las empresas, y una carpeta, *Grafos*, donde se pueden visualizar todos los grafos generados tanto antes de la aplicación del algoritmo de Girvan-Newman como en cada una de las iteraciones posteriores generadas tras su aplicación sobre los mismos.

También se puede encontrar este mismo archivo .pdf y un archivo .zip que contiene el código fuente de generación de esta memoria, el cual contiene un archivo .tex. En línea con lo anterior, se tiene el archivo .py referente al desarrollo práctico creado.

Por último, el repositorio contiene adicionalmente dos archivos .xlsx: El primero, que contiene las Rentabilidades derivadas de la perspectiva 1 para cada empresa y cada comunidad estudiada en esa perspectiva, y el segundo, que es el Horario con la distribución temporal llevada a cabo para desarrollar este trabajo. Este último, además, se puede visualizar en el siguiente apéndice (F).

Para finalizar, se adjunta el link a dicho repositorio.

Link: <https://github.com/drecuero24/Trabajo-Fin-de-Grado-Matematicas-Daniel-Recuero-Cordobes-URJC.git>

The screenshot shows a GitHub repository page for 'drecuero24'. The repository name is 'Código fuente - Memoria del Trabajo Fin de Grado'. It contains 14 commits, 1 branch, and 0 tags. The repository details include:

- Readme**: Base de datos de cotización de cada empresa del IBEX 35
- Grafos**: Grafos obtenidos en el desarrollo práctico
- Grafos\_de\_competitividad\_detecció...**: Memoria del Trabajo Fin de Grado
- Grafos\_de\_competitividad\_detecció...**: Código fuente - Memoria del Trabajo Fin de Grado
- Horario.xlsx**: Distribución temporal del trabajo
- README.md**: Update README.md
- Rentabilidades.xlsx**: Rentabilidad al finalizar el estudio
- TFG Desarrollo Práctico - DRC.py**: Desarrollo práctico TFG

The repository has 0 stars, 1 watching, and 0 forks. It was created by Daniel Recuero Cordobés from URJC. The repository is associated with the project 'Trabajo Fin de Grado Matemáticas'.

**About**

Trabajo Fin de Grado Matemáticas -  
Daniel Recuero Cordobés - URJC

**Readme**

**Releases**

No releases published  
Create a new release

**Packages**

No packages published  
Publish your first package

**Languages**

Python 100.0%

**README.md**

**Grafos de competitividad detección de comunidades mediante el algoritmo de Girvan Newman y aplicación al estudio del IBEX 35**

En este repositorio se encuentra el código, los datos y la memoria, así como resultados obtenidos y archivos empleados en el desarrollo del trabajo, relativos al Trabajo Fin de Grado de Matemáticas de Daniel Recuero Cordobés, graduado por la Universidad Rey Juan Carlos.

Ante cualquier error o necesidad contactar en: [danielrecuercord@gmail.com](mailto:danielrecuercord@gmail.com)

Figura E.1: Repositorio GitHub

# F

## Distribución del tiempo de trabajo

En este último apéndice se recoge la distribución temporal llevada a cabo para realizar este estudio.

Como se puede observar en F.1, el tiempo total empleado para realizar este trabajo ha sido de 178 horas, siendo noviembre el mes con mayor avance. Hay que señalar que no fue hasta finales de septiembre cuando se tuvo bien definida la idea a desarrollar y se comenzó el estudio.

2022						2023			Total
Junio	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre	Enero	Febrero	
4:45	1:35	8:15	8:30	43:15:00	46:55:00	42:40:00	17:15:00	4:50:00	178:00:00

Figura F.1: Tiempo total empleado en el desarrollo del trabajo

Adicionalmente a la tabla total anterior, se adjunta la tabla siguiente donde se recogen las tareas realizadas con su fecha y horario empleado para ello.

Fecha	Hora Inicio	Hora Fin	Duración	Tarea
12/06/2022	10:00	10:40	0:40	Lectura de paper: Comparing rankings by means of competitiveness graphs: structural properties and computation
14/06/2022	22:00	23:00	1:00	Lectura de paper: Comparing series of rankings with ties by using complex networks: An analysis of the spanish stock market (IBEX-35 index)
16/06/2022	22:00	22:45	0:45	Lectura de paper: A new method for comparing rankings through complex networks: Model and analysis of competitiveness of major European soccer leagues
21/06/2022	23:00	0:00	1:00	Búsqueda de información e investigación sobre la teoría de competitividad
23/06/2022	22:00	23:10	1:10	Lectura de paper: Comparing series of rankings with ties by using complex networks: An analysis of the spanish stock market (IBEX-35 index) 2
01/07/2022	9:40	10:30	0:50	Primera toma de contacto con LaTeX
05/07/2022	13:00	13:45	0:45	Búsqueda de aspectos relevantes a estudio sobre las acciones y búsqueda de fuentes de información sobre la teoría de grafos
26/08/2022	15:30	16:30	1:00	Investigación sobre el concepto de hipérgrafos
27/08/2022	11:15	12:05	0:50	Lectura de paper: Graph theory and its applications to the theory of graphs. Rödl, J. Wilson
				Redacción en LaTeX de las primeras definiciones y conceptos básicos de la teoría de grafos
28/08/2022	9:30	12:10	2:40	Lectura de paper: <i>Puntos y lneas</i> ; A. Kaufmann.
				Redacción en el texto de más definiciones y conceptos básicos de la teoría de grafos.
29/08/2022	9:15	13:00	3:45	Redacción de definiciones básicas restantes y algunos resultados posibles a considerar.
04/09/2022	12:45	14:00	1:15	Búsqueda de las posibles fuentes de datos para extraer los datos del estudio.
				Preparación de los datos para su uso en Python y su análisis en un archivo Notebook.
				Búsqueda de que existen de forma gratuita en internet y su descarga.
				Instalación de Anaconda, instalación de Python y decisión de utilizar Jupyter notebooks.
17/09/2022	17:20	18:50	1:30	Carga de los datos de Acciona y Telefónica en el entorno de desarrollo para poder trabajar con ellos. Además, solucionar problemas emergentes durante dicha carga.
				Carga de los datos de más empresas.
18/09/2022	10:25	13:20	2:55	Creación de una función para ordenar los datos de acuerdo a las empresas.
				Desarrollo del algoritmo de ordenamiento propuesto por mí para ordenar los dos primeros rankings.
19/09/2022	18:35	20:25	1:50	Eliminación de uno de los dos algoritmos implementados por falta de eficacia y desarrollo del escogido finalmente.
				Funcionamiento correcto del algoritmo que solo ordenaba los porcentajes pero no ordenaba los nombres de las empresas en consecuencia.
23/09/2022	19:05	20:05	1:00	Primera etapa del algoritmo para contar cruce entre empresas en los rankings.
				Creación de una función para saber si se cruzan las empresas dada una lista de ellas y una par de rankings.
01/10/2022	11:45	14:10	2:25	Creación de la matriz de adyacencia para los rankings.
02/10/2022	11:00	11:50	0:50	Escalarización de la tarea ya realizada anteriormente para poder aplicarla al modelo global.
03/10/2022	19:50	20:50	1:00	Modificación de la función seCrusal() para adaptarla a la estructura de los rankings.
03/10/2022	11:35	13:35	2:00	Prueba con diferentes tipos de grafos, pero los pesos no se colocan de forma correcta.
03/10/2022	19:00	21:20	2:20	Extraer la parte de teoría de grafos teóricos.
10/10/2022	19:25	21:35	2:10	Comenzar la sección de Teoría de grafos: <i>Gráficos de competitividad</i> .
				Trabajar con las referencias y la inserción de imágenes.
				Desarrollo en Overleaf del apartado de Teoría de grafos: <i>Definiciones básicas</i> ; completando con ejemplos e introduciendo explicaciones de los conceptos

Figura F.2: Distribución temporal 1

Fecha	Hora Inicio	Hora Fin	Duración	Tarea
12/10/2022	9:45	12:00	2:15	Comenzar el apartado de Teoría de grafos: <i>Comenzar con los grafos</i> .
12/10/2022	16:40	20:10	3:30	Terminar el apartado de Teoría de grafos: <i>Comenzar con los grafos</i> .
15/10/2022	19:25	21:00	1:35	Comenzar el apartado de Teoría de grafos: <i>Gráficos de competitividad</i> .
16/10/2022	8:30	12:50	4:20	Continuar dudas de parte de fundamentos de cara a consultar para su corrección.
16/10/2022	21:25	22:50	1:25	Continuar el apartado de Teoría de grafos: <i>Gráficos de competitividad</i> .
				Empezar sección Teoría de grafos: <i>Teoría de grafos: Gráficos de competitividad</i> .
				Continuar sección Teoría de grafos: <i>Gráficos de competitividad</i> .
				Continuar sección Teoría de grafos: <i>Gráficos de competitividad</i> .
				Continuar resto de documento y establecer secciones y subsecciones.
				Establecer próximos pasos para completar la parte de fundamentos teóricos.
				Ingesta de datos de conexión de las 35 empresas del IBEX-35 con su correspondiente actualización temporal de los últimos 6 meses desde hoy (23/10/2022).
				Modificación del código existente adaptándolo a los nuevos cambios.
25/10/2022	22:25	23:00	0:35	Finalización de la modificación del código existente adaptándolo a las nuevas versiones.
				Trabajo sobre la matriz de adyacencia y la representación de los grafos que tienen en cuenta el efectivo de cierto diario (actualización con las versiones; mantener aquellas aristas con peso mayor que la medida de pesos y mantener aquellas aristas con menor peso que la medida).
23/10/2022	16:30	20:25	3:55	Comienzo de otra perspectiva del estudio: Invertir 1000 euros en cada una de las empresas.
				Modificación de las bases de datos y creación de dos funciones para crear las columnas nuevas sobre las bases de datos relativas a porcentajes (motivado por la siguiente perspectiva).
				Comienzo de la última perspectiva del estudio: Estudiar el porcentaje de cierre de sesión acumulado a lo largo de las sesiones.
30/10/2022	9:15	13:55	4:40	Primeras tomas de contacto con el algoritmo de Girvan-Newman.
01/11/2022	11:15	13:15	2:00	Aplicación del algoritmo de Girvan-Newman sobre todos los grafos creados anteriormente y revisión de los resultados obtenidos.
02/11/2022	19:10	19:50	0:40	Modificar anotaciones sobre los fundamentos teóricos.
05/11/2022	17:25	18:05	1:40	Trabajo sobre la tercera perspectiva sobre aquellas aristas con peso mayor que la medida y con peso menor que la medida 1.
06/11/2022	17:20	18:40	1:20	Aplicación del parámetro de Kendall de la librería Numpy a los 3 variantes del estudio que se han originado anteriormente.
07/11/2022	20:35	22:55	2:20	Modificaciones en las representaciones de los grafos para una visualización óptima.
09/11/2022	15:15	19:25	4:10	Cambios sobre los fundamentos teóricos e introducción de algún concepto básico sobre la teoría de grafos (complementado con nueva bibliografía).
10/11/2022	19:15	21:40	2:25	Cambios de aspectos teóricos de grafos de competitividad revisando los papers.
12/11/2022	16:20	18:30	2:10	Continuación con los cambios teóricos sobre grafos de competitividad revisando los papers.
13/11/2022	9:55	14:40	4:45	Adhesión a los fundamentos teóricos explicación del algoritmo de Girvan-Newman.
				Depuración de dudas.
				Cambio de la perspectiva del estudio donde se invierten 1000 euros debido a que estaba mal planteada.

Figura F.3: Distribución temporal 2

Fecha	Hora Inicio	Hora Fin	Duración	Tarea
13/11/2022	9:55	14:40	4:45	Depuración de dudas.
17/11/2022	19:00	21:05	2:05	Cambio de la perspectiva del estudio donde se invierten 1000 euros debido a que estaba mal planteada.
				Introducción de texto en la perspectiva de invertir 1000 euros para su estructuración.
				Finalización con los cambios de la sesión anterior.
18/11/2022	16:10	18:00	1:50	- Cambio de la perspectiva del estudio donde se invierten 1000 euros debido a que estaba mal planteada.
21/11/2022	18:40	21:10	2:30	- Introducción de texto en la parte práctica para mejorar su estructuración.
23/11/2022	20:35	23:10	2:35	Añadir de la perspectiva del estudio donde se invierte 1000 euros y se observando la evolución diaria del cambio en el valor de la inversión entre la sesión actual y la anterior.
				Organización de todas las perspectivas del estudio con la redacción de texto en el archivo Jupyter Notebook para darle legibilidad y facilitar su introducción al documento escrito.
24/11/2022	20:25	22:10	1:45	Continuación con la organización de todas las perspectivas del estudio con la redacción de texto en el archivo Jupyter Notebook para darle legibilidad y facilitar su introducción al documento escrito final. Concretamente en la 2ª perspectiva final.
25/11/2022	22:30	0:45	2:15	Continuación con la organización de todas las perspectivas del estudio con la redacción de texto en el archivo Jupyter Notebook para darle legibilidad y facilitar su introducción al documento escrito final. Concretamente en la 2ª perspectiva final.
26/11/2022	17:20	21:20	4:00	Finalización con la organización de todas las perspectivas del estudio con la redacción de texto en el archivo Jupyter Notebook para darle legibilidad y facilitar su introducción al documento escrito final. Concretamente en la 2ª perspectiva final.
27/11/2022	11:30	14:00	2:30	Desarrollo del apartado de cálculo y aplicación de la correlación Kendall (definición de una función para su cálculo).
27/11/2022	16:45	18:50	2:05	Adición de la tercera subvariante a todas las perspectivas con pesos mayores que la medida 2.
27/11/2022	22:20	23:45	1:25	Aplicación del apartado de algoritmo Girvan-Newman a todas las perspectivas y sus subvariaciones desarrolladas.
29/11/2022	20:50	21:55	1:05	Cambios sobre los fundamentos teóricos e introducción de una proposición con la demostración del coeficiente de correlación Kendall.
30/11/2022	18:30	19:50	1:20	Continuación con el apartado teórico. Detección de comunidades - Algoritmo Girvan - Newman.
				Mejoras sobre los fundamentos teóricos y su revisión en la reunión.
				Redacción apartado de la introducción. Convenio y alcance.
				Depuración apartado Objetivo.
				Depuración parte práctica.
02/12/2022	17:45	20:55	3:10	- Definición de medida 1 y medida 2 y apreciar que la medida 1 tenía otro nombre.
02/12/2022	22:30	23:55	1:25	- Eliminación de la medida 2 ya que el código no tenía esa otra medida.
03/12/2022	17:05	18:50	1:45	Depuración parte práctica: Cambios en el apartado de aplicación del algoritmo Girvan-Newman adecuados a la situación de cada grafo para la primera perspectiva.
04/12/2022	9:20	15:00	5:40	Continuación con Depuración parte práctica: Cambios en el apartado de aplicación del algoritmo Girvan-Newman adecuados a la situación de cada grafo para las restantes perspectivas.
06/12/2022	10:55	14:05	3:10	Búsqueda de herramientas para la representación de un dendrograma en Python.
06/12/2022	13:00	21:50	2:50	Finalización y revisión de la parte práctica del trabajo final.
				Introducción de la parte práctica al trabajo final.
				Finalización de la introducción de la parte práctica al trabajo final.
				Cambios en los fundamentos teóricos tras la revisión.
08/12/2022	9:15	14:05	4:50	Complejar Contenidos principales: <i>Conceptos del comercio económico</i> .
09/12/2022	17:30	20:30	3:00	Complejar Contenidos principales: <i>Detección de comunidades - Algoritmo Girvan - Newman</i> .
10/12/2022	16:50	19:55	3:05	Realización del dendrograma para la subvariante 3.3.

Figura F.4: Distribución temporal 3

## Capítulo F. Distribución del tiempo de trabajo

---

Fecha	Hora Inicio	Hora Fin	Duración	Tareas
11/12/2022	9:25	15:00	5:35	Desarrollo del capítulo <i>Resultados</i>
11/12/2022	17:30	20:55	3:25	Desarrollo del capítulo <i>Conclusiones y trabajos futuros</i>
17/12/2022	16:45	18:35	1:50	Adición del pseudocódigo del algoritmo Givian-Newman a los contenidos teóricos.
				Cambios en la bibliografía.
18/12/2022	10:05	13:00	2:55	Depuración de las dudas.
				Revisión del documento y cambios y mejoras sobre el mismo.
04/01/2023	11:05	13:10	2:05	Cambios sobre el documento final tras la revisión del mismo por parte del tutor.
				Modificación de las conclusiones extraídas.
08/01/2023	9:05	11:00	1:55	Finalización de las conclusiones extraídas.
				Desarrollo del fichero <i>Apéndices</i> .
				Revisión de dudas.
				Creación de un anexo para almacenar las funciones definidas e implementadas en la práctica.
				Creación de un anexo para almacenar los gráficos, con la aplicación del algoritmo de Givian-Newman sobre ellos, comentados en los resultados de la práctica.
15/01/2023	16:20	18:30	2:10	Modificaciones y mejoras sobre el documento.
17/01/2023	20:00	21:20	1:20	Creación de apéndices y reestructuración de los contenidos.
20/01/2023	23:30	0:35	1:05	Reducción del punto 4 donde se explica la práctica. Movimiento de ciertos textos a los apéndices y mejoras sobre el texto.
				Reducción del punto 5 donde se explican los resultados obtenidos. Movimiento de ciertos textos a los apéndices y mejoras sobre el texto.
21/01/2023	9:30	11:10	1:40	Reforma acción para aclarar el punto 4 de conclusiones y los apéndices.
22/01/2023	8:35	15:35	7:00	Revisión del documento completo y redacción de los apéndices.
01/02/2023	19:30	22:00	2:30	Revisión del documento completo y anotación de cambios posibles.
02/02/2023	19:40	22:00	2:20	Cambios sobre el documento, adición de elementos al gitHub y redacción de los dos últimos apéndices.

Figura F.5: Distribución temporal 4