

José David Núñez González
Paula Azqueta Tamés

ANÁLISIS COMPUTACIONAL DE REDES COMPLEJAS



Núñez González, José David

Análisis computacional de redes complejas [Recurso electrónico] / José David Nuñez González, Paula Azqueta Tamés. – Datos. – [Leioa]: Universidad del País Vasco / Euskal Herriko Unibertsitatea, Argitalpen Zerbitzua = Servicio Editorial, [2026]. – 1 recurso en línea: PDF (112 p.). – (Unibertsitateko Eskuliburuak = Manuales Universitarios)

Modo de acceso: World Wide Web

Bibliografía: p. 99-102.

ISBN: 978-84-1319-764-7

1. Grafos, Teoría de. 2. Redes de información. 3. Sistemas de recomendación (Filtrado de información). 4. Percolación (Física estadística). I. Azqueta Tamés, Paula, coaut.

(0.034) 681.324

Índice general

1. Fundamentos y Problemas Clásicos de la Teoría de Grafos	1
1.1. Introducción	1
1.2. Fundamentos de la teoría de grafos	1
1.2.1. Conceptos básicos	1
1.2.2. Representaciones	2
1.2.3. Algoritmos fundamentales	4
Algoritmo de Dijkstra: caminos mínimos	4
Algoritmo de Prim: árbol de expansión mínima (AEM)	6
Algoritmo de Kruskal: árbol de expansión mínima por orden de aristas	7
1.2.4. Resumen:	8
1.3. Problemas clásicos de la teoría de grafos	9
1.3.1. El problema de los puentes de Königsberg	9
1.3.2. El problema del viajante	10
1.3.3. El problema de la coloración de mapas	10
1.4. Combinatoria y teoría de Ramsey	11
1.4.1. Principios combinatorios en grafos	11
1.4.2. Teoría de Ramsey	12
1.5. Teoría de Redes Sociales	13
1.5.1. Tipología de Redes	13
1.5.2. Propiedades Estructurales	14
1.6. Dinámica y estructura en redes	14
1.6.1. Dinámica sobre una red	14
1.6.2. Detección de comunidades	15
1.7. Redes complejas y su tratamiento estadístico	16
1.7.1. Características de una red compleja	16
1.7.2. Tratamiento estadístico	16
2. Métricas Estructurales y Modelos de Redes	19
2.1. Métricas de Caracterización de Redes	19
2.1.1. Caminos y propiedades básicas	19
Camino sobre un grafo	19
Camino geodésico	19
Camino euleriano	20

	Camino hamiltoniano	20
2.1.2.	Grado y densidad	21
	Grado de un nodo	21
	Distribución de grados	22
	Densidad del grafo	22
2.1.3.	Medidas globales	23
	Diámetro de un grafo	23
	Distancia media sobre un grafo	23
	Problemas en el cálculo de distancias medias	23
2.2.	Conectividad y Componentes	25
2.2.1.	Componentes Conexas	25
2.2.2.	Componente Gigante	25
2.2.3.	Componentes Desconectadas	26
2.2.4.	Transición de Fase	26
2.2.5.	Estructura en Grafos Dirigidos	26
	Tendrils (zarcillos)	27
	Tubes (túneles)	27
2.3.	Propiedades estructurales y métricas	27
2.3.1.	Medidas de centralidad	27
	Centralidad de grado	28
	Centralidad de cercanía (closeness)	28
	Centralidad de intermediación (betweenness)	29
2.3.2.	Estructura local	31
	Coeficiente de <i>clustering</i>	31
	Correlación de grados entre vecinos	32
2.4.	Modelos de grafos aleatorios	34
2.4.1.	Grafos aleatorios generalizados	34
2.4.2.	Modelo de Erdős–Rényi	34
	Definición y propiedades	35
	Transición de fase y componente gigante	35
2.4.3.	Modelo de configuración	36
2.5.	Topologías y modelos de redes	37
2.5.1.	Redes Small World	37
	Modelo de Watts–Strogatz	37
	Transición regular–aleatoria	38
2.5.2.	Redes libres de escala	38
	Modelo de Barabási–Albert (Preferential Attachment)	39
	Modelo de Price	39
2.6.	Análisis de comunidades	41
2.6.1.	Conceptos básicos: cluster, comunidad y módulo	41
	Clustering espectral	42
	Método de Girvan–Newman	43

2.6.2. Comparación de modelos	43
3. Sistemas Recomendadores y Modelos de Confianza	45
3.1. Introducción a los Sistemas Recomendadores (SR)	45
3.1.1. Breve Perspectiva Histórica: Groupware y Workflow	45
Groupware (Software Colaborativo)	45
Workflow (Gestión de Flujos de Trabajo)	46
3.1.2. Definición y Naturaleza del SR	46
3.1.3. El SR como Pilar de la Economía de la Atención	47
3.1.4. Problemas Fundamentales de los Sistemas Recomendadores	47
3.2. Estructura del Conocimiento: Taxonomía vs. Folksonomía	47
3.3. Arquitectura General de un Sistema Recomendador	49
3.3.1. Mecanismos de Similitud	50
3.3.2. Mecanismos de Retroalimentación: Implícita vs. Explícita	50
Retroalimentación Explícita	51
Retroalimentación Implícita	51
El Desafío de la Preferencia Negativa	52
3.3.3. Recolección y Preprocesamiento de Datos	52
Fuentes de Datos y Características	52
Preprocesamiento y Normalización	52
3.3.4. Técnicas Predictoras Avanzadas	53
Regresión como Predictor	53
Predictor Slope One	53
3.3.5. Técnicas Predictoras	54
Descomposición en Valores Singulares (SVD)	55
3.3.6. Evaluación de Sistemas Recomendadores	56
3.4. Tipologías de Sistemas Recomendadores	57
3.4.1. Sistemas Recomendadores Colaborativos (CF)	57
Proceso del Filtrado Colaborativo	58
Tipos de CF y Modelos	58
Ejemplos de Sistemas Recomendadores (SR)	59
Ventajas y Desventajas Formales del CF	59
3.4.2. Sistemas Recomendadores Basados en Contenidos (CB)	59
3.4.3. Sistemas Recomendadores Híbridos	60
3.5. De la Recomendación a la confianza: modelos WoT	60
3.5.1. Predicción de Confianza (<i>Trust Prediction</i>)	61
Pipeline de Predicción	61
3.5.2. Web de Confianza (WoT)	62
3.5.3. Propiedades y Métricas de Confianza	62
Propiedades de la Confianza	62
Métricas de Confianza	62
3.5.4. Aplicaciones y Esquemas de Confianza en SR	62

Aplicaciones de la Confianza	62
Esquemas de Confianza (<i>Trust Schemes</i>)	63
3.6. Redes Dinámicas: Influencia y Propagación	63
3.6.1. Problemas Identificados en Sistemas Recomendadores	63
3.6.2. Generación de Recetas y Recomendación Inversa	63
Diseño Experimental y Satisfacción	64
Predicción Directa vs. Recomendación Inversa	64
3.6.3. Maximización de Influencia (<i>Influence Maximization</i>)	64
3.6.4. Web de Confianza (WoT)	64
Estrategias de Búsqueda	64
Modelos de Propagación Utilizados	65
Optimización Heurística para la Selección de Semillas	66
4. Percolación y Dinámicas de Transmisión	69
4.1. Introducción a la Percolación y Procesos de Transmisión	69
4.1.1. Aplicaciones de la Teoría de la Percolación	69
Física de Materiales y Química	70
Epidemiología y Salud Pública	70
Sociología y Redes Sociales	70
Biología y Neurociencia	71
4.2. Percolación en Redes Complejas	71
Conceptos Fundamentales de la Percolación	71
Percolación de Sitio (<i>Site Percolation</i>)	72
Percolación de Enlace (<i>Bond Percolation</i>)	72
4.2.1. Modelado y Análisis de la Percolación	72
Modelo de Bond Percolation	73
Percolación en Grafos Aleatorios (Erdos-Rényi)	74
Percolación en Grafos Arbitrarios	74
4.3. Grafos Geométricos Aleatorios y Percolación	75
4.3.1. Definición de Grafos Geométricos Aleatorios	75
4.3.2. Ejemplos y Propiedades de Grafos Geométricos	76
4.3.3. Percolación en Grafos Geométricos	77
4.4. Dinámicas de Difusión: de la Percolación a la Epidemiología	77
4.4.1. Transmisión sobre Redes	77
Ecuaciones de Transmisión	78
Interpretación de la Transmisión	79
4.4.2. Modelo SIS (<i>Susceptible-Infected-Susceptible</i>)	80
4.4.3. Modelo SIR (<i>Susceptible-Infected-Recovered</i>)	80
5. Dinámica de Sincronización en Redes	83
5.1. Emergencia del Orden: Sincronización en Sistemas Complejos	83
5.1.1. Clustering Basado en la Sincronización	83
5.2. Modelos de Fase y Comportamiento Colectivo	84

5.2.1.	Osciladores No Idénticos: El Modelo de Kuramoto	85
	Representación Mean-Field	85
	Umbral de Sincronización	86
5.2.2.	Sincronización Social: Modelos de Opinión y Consenso	87
	Modelo de Opinión: Opinion Changing Rate Model (Rate Model)	87
	Clustering Dinámico Basado en el Modelo OCR	87
	Clustering Dinámico: La Red Zachary Karate Club	88
5.3.	Sincronización de Sistemas Caóticos	89
5.3.1.	Dinámica Sincronizada y Sistemas Caóticos	89
	Sincronización de Dos Osciladores Acoplados Idénticos	89
5.3.2.	El Esquema de Sincronización de Pecora y Carroll (PC)	90
	La Configuración de Pecora y Carroll	90
	PC Synchronization Scheme	91
5.3.3.	Análisis de Estabilidad y Acoplamiento General	91
	Sincronización Completa: Estabilidad	92
	General Linear Coupling Scheme	92
	Naturaleza del Esquema de Acoplamiento	93
5.4.	Sincronización en Redes de Osciladores de Fase	93
5.4.1.	Osciladores Acoplados y Acoplamiento Lineal	94
5.4.2.	La Función de Estabilidad Maestra (<i>The Master Stability Function</i>)	94
Conclusiones		97

Índice de figuras

1.1. Representación abstracta de una red compleja y sus interconexiones. Fuente: [14].	2
1.2. Mapa de 1652 de la Ciudad de Königsberg con el agua y los puentes resaltados. Fuente: [36].	9
1.3. Visualización del número de Ramsey $R(3, 3) = 6$: en cualquier coloración de las aristas de un K_6 con dos colores, siempre aparece un triángulo monocolor. Fuente: [19, 10].	12
1.4. Ejemplo esquemático de una red compleja con hubs, comunidades y enlaces heterogéneos [4].	17
2.1. Comparativa entre camino euleriano y ciclo hamiltoniano. Fuente: Adaptado de [15, 10].	21
2.2. Representación del problema del camino más corto en un grafo ponderado. Fuente: [22, 14].	24
2.3. Representación de la emergencia de la componente gigante (GCC) en una red aleatoria. Fuente: [12, 4].	26
2.4. Estructura en grafos dirigidos: núcleos (GSCC), zarcillos (tendrils) y túneles (tubes). Fuente: [11, 28].	27
2.5. Visualización de una red social (Twitter) y sus métricas de centralidad. Fuente: [11].	30
2.6. El número de Erdős como ejemplo de distancias en redes de coautoría. Fuente: [29, 4].	35
2.7. Modelo Watts-Strogatz: transición de red regular a aleatoria. Fuente: [43].	37
2.8. Comparativa estructural entre redes monótonas, de mundo pequeño y aleatorias. Fuente: [28, 43].	40
2.9. Algoritmos de detección de comunidades en redes complejas. Fuente: [30, 17].	41
2.10. Red con comunidades, módulos y motifs [28]	42
3.1. Arquitectura general de los Sistemas de Recomendación. Fuente: [20].	49
3.2. Esquema del predictor clásico Slope One para filtrado colaborativo. Fuente: [20, 38].	54
3.3. Esquema conceptual: Comparativa entre Filtrado Colaborativo y Basado en Contenido. Fuente: [1, 20].	57

3.4. Modelos de confianza: Confianza delegada frente a Web of Trust (WoT). Fuente: [31].	61
3.5. Algoritmo de maximización de influencia basado en la reducción del espacio de búsqueda en redes sociales. Fuente: [31, 11].	65
3.6. Algoritmo de maximización de influencia basado en la reducción del search space en redes sociales [31]	65
3.7. Flujo del algoritmo de optimización heurístico clásico [31]	66
4.1. Percolación de Sitio y Enlace [28]	73
4.2. Ejemplo de umbral de percolación en el modelo Erdos-Renyi [29]	74
4.3. Ejemplo Grafos Geométricos [28]	76
5.1. Ejemplo de redes complejas aplicadas a neurología. Fuente: Adaptado de [6].	84
5.2. Red Zachary Karate Club: Clústeres detectados mediante la dinámica de sincronización. Fuente: [26].	89
5.3. Gráfico de dos osciladores acoplados bajo el modelo de Kuramoto. Fuente: [27].	90

Capítulo 1

Fundamentos y Problemas Clásicos de la Teoría de Grafos

1.1. Introducción

La teoría de grafos constituye una de las herramientas más poderosas para modelar sistemas complejos en matemáticas, informática, sociología y biología. Este trabajo presenta un recorrido desde los fundamentos de la teoría de grafos y la combinatoria hasta la teoría moderna de redes complejas, abordando tanto los aspectos teóricos como las aplicaciones en redes sociales y sistemas reales.

1.2. Fundamentos de la teoría de grafos

1.2.1. Conceptos básicos

Un **grafo** $G = (V, E)$ es una estructura matemática compuesta por un conjunto de *vértices* o *nodos* $V = \{v_1, v_2, \dots, v_n\}$ y un conjunto de *aristas* $E \subseteq V \times V$ que representan las conexiones entre ellos [10, 30].

Cada arista $e = (v_i, v_j)$ une dos vértices del grafo. Según el tipo de relación que representan, y tal como se detalla en la literatura clásica de redes complejas [4], se distinguen varios tipos de grafos:

- **Grafo no orientado:** las aristas no tienen dirección, es decir, (v_i, v_j) es equivalente a (v_j, v_i) .
- **Grafo orientado (dígrafo):** cada arista tiene una dirección asociada, representada por una flecha de v_i a v_j .
- **Grafo ponderado:** cada arista tiene asociado un valor o peso w_{ij} , que puede representar distancia, coste o capacidad [14].
- **Grafo bipartito:** el conjunto de vértices se puede dividir en dos subconjuntos V_1 y V_2 tales que todas las aristas conectan un vértice de V_1 con uno de V_2 .

Entre las propiedades fundamentales de un grafo se encuentran:

- **Grado:** número de aristas que inciden en un vértice. En grafos dirigidos se distingue entre *grado de entrada* (d^-) y *grado de salida* (d^+).
- **Camino:** secuencia de vértices conectados por aristas consecutivas.
- **Conectividad:** mide si es posible llegar de un vértice a otro mediante un camino.
- **Ciclo:** camino cerrado que comienza y termina en el mismo vértice sin repetir aristas.
- **Componente conexas:** subgrafo maximal en el que cualquier par de vértices está conectado mediante algún camino.



Figura 1.1: Representación abstracta de una red compleja y sus interconexiones. Fuente: [14].

1.2.2. Representaciones

Un grafo puede representarse de varias formas dependiendo de las operaciones que se deseen realizar y del nivel de detalle necesario [10]. Las tres más habituales son la **matriz de adyacencia**, la **lista de adyacencia** y la **matriz de incidencia** [30, 14].

Matriz de adyacencia

La *matriz de adyacencia* $A = [a_{ij}]$ de un grafo con n vértices es una matriz cuadrada de dimensión $n \times n$, donde cada elemento indica si existe una arista entre dos vértices:

$$a_{ij} = \begin{cases} 1, & \text{si existe una arista de } v_i \text{ a } v_j, \\ 0, & \text{en caso contrario.} \end{cases}$$

Por ejemplo, para el grafo con vértices A, B, C, D y aristas $\{(A, B), (A, C), (B, C), (B, D)\}$, la matriz de adyacencia es:

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Lista de adyacencia

La *lista de adyacencia* asocia a cada vértice el conjunto de vértices con los que está conectado. Esta forma ocupa menos memoria cuando el grafo tiene pocas aristas respecto al número de vértices [4].

Cuadro 1.1: Lista de adyacencia para el grafo ejemplo.

Vértice	Vecinos
A	B, C
B	A, C, D
C	A, B
D	B

Matriz de incidencia

La *matriz de incidencia* $M = [m_{ij}]$ relaciona vértices y aristas [10]. Si el grafo tiene n vértices y m aristas, M es una matriz $n \times m$ tal que:

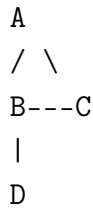
$$m_{ij} = \begin{cases} 1, & \text{si el vértice } v_i \text{ participa en la arista } e_j, \\ 0, & \text{en caso contrario.} \end{cases}$$

Cuadro 1.2: Matriz de incidencia correspondiente al grafo ejemplo.

	$e_1(A, B)$	$e_2(A, C)$	$e_3(B, C)$	$e_4(B, D)$
A	1	1	0	0
B	1	0	1	1
C	0	1	1	0
D	0	0	0	1

Esquema visual sencillo

Para hacerse una idea de la estructura del grafo ejemplo:



En este esquema, cada línea representa una arista. Así, por ejemplo, el vértice B está conectado con A, C y D.

Cuadro 1.3: Ventajas y desventajas de las representaciones de grafos.

Representación	Ventajas	Desventajas
Matriz de adyacencia	Acceso rápido a aristas; forma algebraica.	Alto uso de memoria si el grafo es disperso.
Lista de adyacencia	Eficiente en memoria; fácil de recorrer.	Acceso más lento a pares no conectados.
Matriz de incidencia	Buena para estudiar propiedades de aristas.	Menos intuitiva y más grande para grafos densos.

1.2.3. Algoritmos fundamentales

Los algoritmos de la teoría de grafos permiten resolver problemas de optimización y búsqueda de rutas de forma eficiente [30]. Entre los más importantes se encuentran el **algoritmo de Dijkstra**, el **algoritmo de Prim** y el **algoritmo de Kruskal**. Aunque los tres trabajan sobre grafos ponderados, cada uno tiene un objetivo distinto: Dijkstra busca el *camino más corto*, mientras que Prim y Kruskal buscan el *árbol de expansión mínima* [10, 14]. A continuación se explican sus bases conceptuales, sus pasos fundamentales y sus diferencias.

Algoritmo de Dijkstra: caminos mínimos

El algoritmo de Dijkstra encuentra la ruta más corta desde un vértice de origen hasta todos los demás vértices de un grafo con pesos no negativos [30]. Su idea se basa en construir progresivamente una tabla de distancias mínimas, eligiendo siempre el vértice más cercano al origen que aún no se haya visitado. Este es el algoritmo utilizado por Google Maps (GPS), sistemas de transporte público, redes ferroviarias y simulaciones de flujo de energía entre otros.

Ejemplo sencillo: Supongamos el siguiente grafo con pesos (distancias):

$$A \xrightarrow{2} B, \quad A \xrightarrow{5} C, \quad B \xrightarrow{1} C, \quad B \xrightarrow{3} D, \quad C \xrightarrow{2} D$$

Queremos las distancias mínimas desde el vértice A.

Cuadro 1.4: Ejecución paso a paso del algoritmo de Dijkstra

Paso	A	B	C	D	Vértice fijado
Inicial	0	∞	∞	∞	—
1	0	2	5	∞	A
2	0	2	3	5	B
3	0	2	3	5	C
4	0	2	3	5	D

Interpretación: Primero se parte del nodo A con distancia 0. Desde A, se actualizan las distancias de sus vecinos (B=2, C=5). Después se elige el nodo con menor distancia provisional (B) y se relajan sus aristas: como C puede alcanzarse por A→B→C con peso 3, se mejora su distancia (de 5 a 3). El proceso continúa hasta que todos los vértices estén “fijados”.

Resultado: las distancias mínimas son: $d(A, B) = 2$, $d(A, C) = 3$, $d(A, D) = 5$. Sea $G = (V, E)$ un grafo ponderado, con una función de pesos $w : E \rightarrow \mathbb{R}_{\geq 0}$. Se desea encontrar para un vértice origen $s \in V$ la distancia mínima $d(s, v)$ a cada vértice $v \in V$.

$$d(s, v) = \min_{P \in \mathcal{P}(s, v)} \sum_{(u, t) \in P} w(u, t)$$

donde $\mathcal{P}(s, v)$ es el conjunto de todos los caminos posibles entre s y v .

Idea matemática. Dijkstra define un conjunto S de vértices cuya distancia mínima desde s ya ha sido determinada, y mantiene para los demás vértices una estimación provisional $\hat{d}(v)$. Inicialmente:

$$\hat{d}(s) = 0, \quad \hat{d}(v) = +\infty, \quad \forall v \neq s$$

En cada iteración:

$$u = \arg \min_{v \notin S} \hat{d}(v)$$

$$S \leftarrow S \cup \{u\}$$

y se *relajan* las aristas adyacentes:

$$\hat{d}(v) \leftarrow \min(\hat{d}(v), \hat{d}(u) + w(u, v))$$

Este proceso garantiza que cuando un vértice u es añadido a S , $\hat{d}(u) = d(s, u)$. La demostración se basa en el principio de optimalidad y en la no negatividad de los pesos [10, 30] (si hubiera pesos negativos, una relajación posterior podría mejorar una distancia ya fijada, violando el invariante).

Complejidad:

$$T(|V|, |E|) = \begin{cases} O(|V|^2), & \text{si se usa matriz de adyacencia} \\ O((|V| + |E|) \log |V|), & \text{si se usa heap binario [14]} \end{cases}$$

Algoritmo de Prim: árbol de expansión mínima (AEM)

El algoritmo de Prim busca conectar todos los vértices de un grafo con el menor coste total posible, sin formar ciclos [10]. Esto se conoce como *árbol de expansión mínima*, porque conecta todos los nodos con el menor peso total de aristas. Este algoritmo es muy utilizado en compañías eléctricas, telecomunicaciones y redes urbanas entre otros [14].

La idea básica es muy intuitiva: 1. Comenzamos en cualquier vértice. 2. En cada paso, elegimos la arista más ligera que conecta un vértice ya incluido en el árbol con uno que aún no está. 3. Repetimos hasta conectar todos los vértices.

Ejemplo:

$$A \xrightarrow{2} B, \quad A \xrightarrow{3} C, \quad B \xrightarrow{1} C, \quad B \xrightarrow{4} D, \quad C \xrightarrow{2} D$$

Cuadro 1.5: Construcción paso a paso del árbol de Prim

Paso	Arista elegida	Árbol actual	Peso acumulado
1	(A,B) [2]	A-B	2
2	(B,C) [1]	A-B-C	3
3	(C,D) [2]	A-B-C-D	5

El resultado es un árbol con peso total $2 + 1 + 2 = 5$. Si dibujamos las aristas elegidas, veremos que todos los vértices están conectados sin formar ciclos.

Sea $G = (V, E)$ un grafo no dirigido, conexo y ponderado con pesos $w(e) > 0$. Se busca un **árbol de expansión mínima (AEM)**, es decir, un subgrafo $T = (V, E_T)$ tal que:

$$E_T \subseteq E, \quad |E_T| = |V| - 1, \quad T \text{ es conexo,}$$

y

$$w(T) = \sum_{e \in E_T} w(e)$$

es mínimo [30].

Formulación matemática. Prim parte de un vértice inicial $r \in V$, y mantiene dos conjuntos:

$$S \subseteq V \quad (\text{vértices ya incluidos en el árbol})$$

$$V \setminus S \quad (\text{vértices aún no conectados})$$

En cada iteración se selecciona la arista de peso mínimo que cruza el *corte* $(S, V \setminus S)$:

$$e^* = \arg \min_{(u,v) \in E, u \in S, v \notin S} w(u, v)$$

y se actualiza:

$$S \leftarrow S \cup \{v\}, \quad E_T \leftarrow E_T \cup \{e^*\}$$

Propiedad de corte (demostración de corrección). Sea $(S, V \setminus S)$ un corte cualquiera del grafo y e la arista de menor peso que lo cruza. Entonces existe al menos un árbol de expansión mínima que contiene e [10]. El algoritmo de Prim se basa en aplicar esta propiedad redundante o recursivamente, garantizando la optimalidad global por selección local mínima.

Complejidad:

$$T(|V|, |E|) = \begin{cases} O(|V|^2), & \text{si se representa con matriz de adyacencia} \\ O(|E| \log |V|), & \text{si se usan colas de prioridad [14]} \end{cases}$$

Algoritmo de Kruskal: árbol de expansión mínima por orden de aristas

Kruskal persigue el mismo objetivo que Prim (encontrar el AEM), pero lo hace de un modo diferente [10]. En lugar de construir desde un vértice, Kruskal parte de un grafo vacío y va añadiendo aristas por orden de peso creciente, siempre que no formen un ciclo con las ya elegidas. Este es muy usado en sistemas de cableado interurbano y satelital.

Pasos: 1. Ordenar todas las aristas de menor a mayor peso. 2. Ir añadiendo cada arista si conecta dos componentes distintas (evita ciclos). 3. Detenerse cuando haya $n - 1$ aristas.

Ejemplo (mismo grafo):

Aristas ordenadas: $(B, C)[1], (A, B)[2], (C, D)[2], (A, C)[3], (B, D)[4]$

Cuadro 1.6: Selección de aristas en el algoritmo de Kruskal

Paso	Arista	¿Aceptada?
1	(B,C) [1]	Sí
2	(A,B) [2]	Sí
3	(C,D) [2]	Sí
4	(A,C) [3]	No (forma ciclo)
5	(B,D) [4]	No (ya conectados)

El árbol mínimo obtenido es el mismo que con Prim, con peso total $1 + 2 + 2 = 5$ [30].

Kruskal persigue el mismo objetivo que Prim, pero desde una estrategia diferente. Define inicialmente un bosque de $|V|$ árboles, uno por cada vértice, y los va uniendo de forma incremental [14].

Formalmente, sea el conjunto de aristas ordenado por peso:

$$E = \{e_1, e_2, \dots, e_m\}, \quad \text{con } w(e_1) \leq w(e_2) \leq \dots \leq w(e_m)$$

Se recorre secuencialmente este conjunto y se añaden aristas mientras no formen ciclos:

$$E_T = \emptyset$$

$$\text{Para cada } e_i = (u, v) : \begin{cases} E_T \leftarrow E_T \cup \{e_i\}, & \text{si } u \text{ y } v \text{ están en componentes distintas} \\ \text{ignorar } e_i, & \text{si forma ciclo} \end{cases}$$

El criterio de parada se cumple cuando $|E_T| = |V| - 1$. **Propiedad de ciclo (justificación).** Si C es un ciclo en el grafo y $e_{\text{máx}}$ es la arista de mayor peso en ese ciclo, entonces existe un AEM que no incluye $e_{\text{máx}}$. Kruskal garantiza optimalidad al descartar las aristas que cerrarían un ciclo (siempre la más pesada dentro de él).

Implementación matemática. El algoritmo se apoya en una estructura de conjuntos disjuntos (Union-Find), que implementa dos operaciones:

Find(v) : devuelve el identificador del conjunto al que pertenece v

Union(u, v) : une los conjuntos que contienen u y v

Estas operaciones pueden implementarse con compresión de caminos y unión por rango, logrando complejidad casi lineal.

Complejidad:

$$T(|E|, |V|) = O(|E| \log |E|) + O(|E| \alpha(|V|)) \approx O(|E| \log |V|)$$

donde $\alpha(|V|)$ es la función inversa de Ackermann (crece extremadamente lento).

1.2.4. Resumen:

- **Dijkstra** es como un explorador que sale del origen y va marcando las rutas más cortas conocidas, expandiéndose hacia los vértices más cercanos primero. - **Prim** actúa como si estuviera tendiendo una red eléctrica: empieza en un punto y va extendiendo los cables más baratos para conectar todos los nodos. - **Kruskal** trabaja como un ingeniero que analiza todo el mapa de conexiones posibles y va uniendo las piezas más baratas sin cerrar circuitos.

En resumen:

Dijkstra: rutas más cortas vs. Prim y Kruskal: árbol mínimo.

Los tres algoritmos son pilares de la teoría de grafos y se aplican en campos tan diversos como redes de transporte, telecomunicaciones, logística o sistemas de recomendación.

1.3. Problemas clásicos de la teoría de grafos

A lo largo de la historia, muchos problemas aparentemente cotidianos han servido de punto de partida para el desarrollo de la teoría de grafos. Tres de los más célebres son el **problema de los puentes de Königsberg**, el **problema del viajante** y el **problema de la coloración de mapas**. Cada uno representa un tipo distinto de desafío: el primero da origen al concepto de *recorrido euleriano*, el segundo inaugura la *optimización combinatoria* y el tercero introduce la *coloración de grafos* y la teoría de planos.

1.3.1. El problema de los puentes de Königsberg

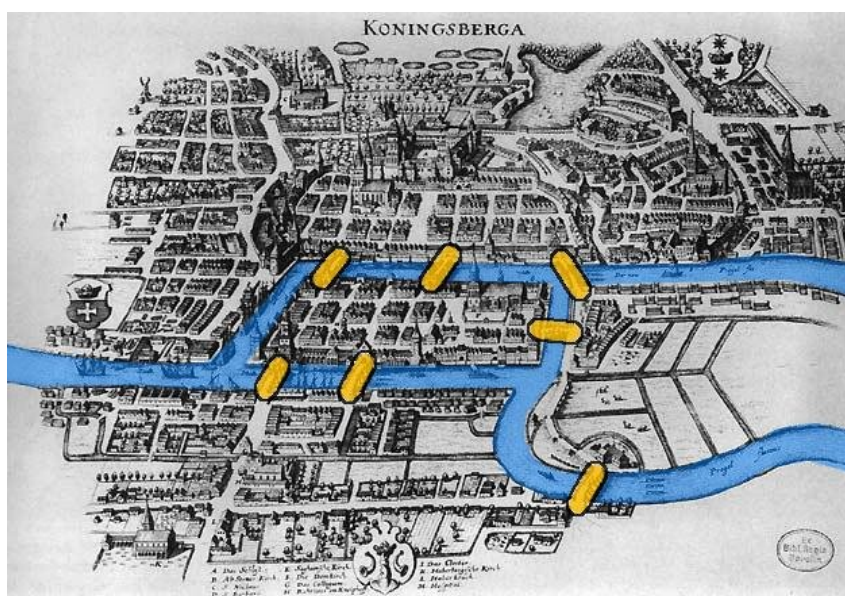


Figura 1.2: Mapa de 1652 de la Ciudad de Königsberg con el agua y los puentes resaltados. Fuente: [36].

1. Contexto histórico. El origen de la teoría de grafos se remonta al siglo XVIII, en la ciudad prusiana de Königsberg (hoy Kaliningrado). La ciudad estaba atravesada por el río Pregel, que dividía el territorio en cuatro zonas de tierra conectadas por siete puentes. Los habitantes se preguntaban si era posible realizar un paseo que cruzara todos los puentes exactamente una vez y regresara al punto de partida.

2. Modelización sencilla. Leonhard Euler (1736) representó cada zona terrestre como un *vértice* y cada puente como una *arista* que unía dos vértices [15]. Así transformó el problema físico en uno matemático:

¿Existe un recorrido que pase por todas las aristas exactamente una vez?

A este tipo de camino se le llama **recorrido euleriano**. Si además comienza y termina en el mismo vértice, se denomina **circuito euleriano** [10].

3. Formulación general. Sea $G = (V, E)$ un grafo no dirigido. Un recorrido euleriano es una secuencia $v_1, v_2, \dots, v_k$ tal que cada arista de E aparece exactamente una vez entre dos vértices consecutivos de la secuencia [30].

4. Teorema de Euler. Euler demostró que para un grafo conexo:

$$\begin{cases} \text{Circuito euleriano} \iff \text{todos los vértices de grado par.} \\ \text{Recorrido euleriano} \iff \text{exactamente dos vértices de grado impar.} \end{cases}$$

En el caso de Königsberg, todos los vértices tenían grado impar (tres o cinco), por lo que *no existía tal recorrido* [10, 14].

5. Interpretación y legado. La genialidad de Euler consistió en reducir un problema físico a una estructura abstracta de vértices y aristas. Ese paso fundó la teoría de grafos moderna. Hoy los recorridos eulerianos se aplican en problemas de trazado de rutas, planificación de calles o incluso en el ensamblaje genómico [30].

1.3.2. El problema del viajante

1. Introducción intuitiva. El **problema del viajante de comercio** (TSP, por sus siglas en inglés) plantea una situación donde un viajante debe visitar una lista de ciudades exactamente una vez y regresar al punto de partida recorriendo la distancia mínima posible [14]. A pesar de su sencillez, es uno de los más estudiados en optimización combinatoria.

2. Modelización formal. Sea $G = (V, E)$ un grafo completo donde cada vértice v_i representa una ciudad y cada arista (v_i, v_j) tiene un peso w_{ij} . El objetivo es encontrar el ciclo hamiltoniano de coste mínimo [10]:

$$\min_{\pi \in S_n} \sum_{i=1}^{n-1} w_{\pi(i)\pi(i+1)} + w_{\pi(n)\pi(1)}$$

3. Complejidad. El número de posibles rutas crece factorialmente con n . Esto implica que el TSP pertenece a la clase de problemas **NP-duros**, lo que significa que no existe un algoritmo que garantice la solución exacta en tiempo polinómico para casos generales [14].

4. Métodos de resolución. Existen diversas aproximaciones que van desde métodos exactos hasta heurísticas como algoritmos genéticos o colonia de hormigas [30]. El algoritmo de Christofides es una de las aproximaciones más destacadas para grafos métricos.

1.3.3. El problema de la coloración de mapas

1. Motivación sencilla. ¿Cuántos colores son necesarios para pintar un mapa de forma que dos regiones adyacentes nunca compartan el mismo color? Este problema fue planteado originalmente por Francis Guthrie en el siglo XIX [10].

2. Traducción a grafos. Cada región se representa como un vértice, y el problema se convierte en hallar el **número cromático** del grafo, denotado por $\chi(G)$ [10].

3. Teorema de los cuatro colores. Para grafos planos, el teorema afirma que $\chi(G) \leq 4$ [14].

4. Breve historia y demostración. Fue finalmente demostrado en 1976 por Appel y Haken mediante un método asistido por ordenador, marcando un hito en la historia de las matemáticas computacionales [10].

5. Generalización y Aplicaciones. El problema se generaliza a coloración de aristas y redes no planas, aplicándose en la planificación de horarios y asignación de frecuencias de radio [30, 14].

6. Interpretación conceptual. Mientras Euler se centraba en *recorridos*, y el viajante en *caminos óptimos*, la coloración aborda la *interferencia y compatibilidad* [14].

1.4. Combinatoria y teoría de Ramsey

La combinatoria y la teoría de grafos están estrechamente relacionadas: ambas estudian las posibles configuraciones que surgen al conectar o disponer elementos de un conjunto [10]. En la teoría de grafos, la combinatoria permite contar caminos, calcular probabilidades de conexión y analizar estructuras complejas [14]. Por otro lado, la **teoría de Ramsey** aporta una visión profunda: muestra que, en estructuras suficientemente grandes, el orden y los patrones son inevitables, incluso dentro del aparente caos [30].

1.4.1. Principios combinatorios en grafos

1. Introducción sencilla. La combinatoria se ocupa de responder preguntas del tipo “¿cuántas formas hay de...?”. En los grafos, esto significa contar cuántas posibles conexiones, caminos o subestructuras se pueden formar entre los vértices [10].

2. Conteo de aristas y subgrafos. En un grafo completo K_n , el número total de aristas se obtiene mediante el coeficiente binomial:

$$|E(K_n)| = \binom{n}{2} = \frac{n(n-1)}{2}$$

Si queremos contar el número de posibles subgrafos de K_n , observamos que cada arista tiene un estado binario (presente o ausente):

$$\text{Número total de subgrafos posibles} = 2^{|E(K_n)|} = 2^{\frac{n(n-1)}{2}}$$

Este crecimiento exponencial es fundamental para entender la complejidad de los algoritmos de búsqueda en redes [14].

3. Conteo de caminos. Un camino simple de longitud k puede considerarse como una variación sin repetición de vértices [10]:

$$P(n, k + 1) = \frac{n!}{(n - k - 1)!}$$

Estas fórmulas son la base de los cálculos de probabilidad en el modelo de grafos aleatorios de Erdős–Rényi [12].

1.4.2. Teoría de Ramsey

1. Idea intuitiva. La teoría de Ramsey afirma que en estructuras suficientemente grandes, algún tipo de orden debe aparecer necesariamente [10].

2. Formulación formal. El **Teorema de Ramsey** establece que para cualquier par de enteros r, s , existe un número $R(r, s)$ tal que todo grafo con ese número de vértices contiene o bien una clique de tamaño r , o bien un conjunto independiente de tamaño s [10, 30].

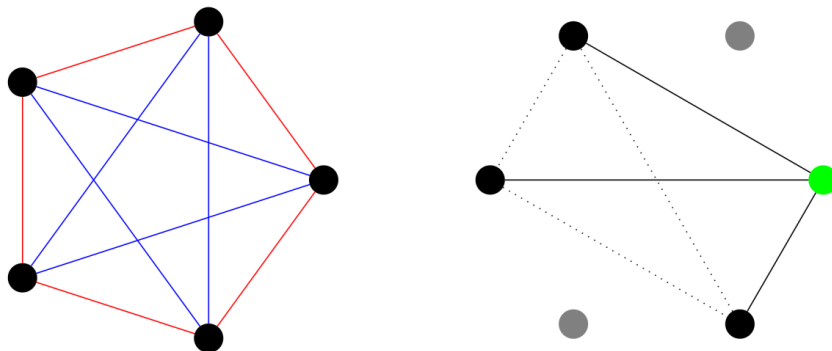


Figura 1.3: Visualización del número de Ramsey $R(3, 3) = 6$: en cualquier coloración de las aristas de un K_6 con dos colores, siempre aparece un triángulo monocolor. Fuente: [19, 10].

3. Interpretación combinatoria. En notación de coloración de aristas, el número de Ramsey se define como [10]:

$$R(r, s) = \min\{n \in \mathbb{N} : K_n \text{ coloreado con dos colores contiene un } K_r \text{ rojo o un } K_s \text{ azul}\}$$

4. Casos particulares y aplicaciones. **a) Teorema de la amistad.** Propuesto originalmente por Erdős, Rényi y Sós (1966), afirma que si en una red cada par de personas tiene exactamente un amigo en común, debe existir un individuo conectado con todos los demás [14].

b) Problema del final feliz. Planteado por Erdős y Szekeres (1935), conecta la geometría combinatoria con la teoría de Ramsey, demostrando que para cualquier n existe un número $N(n)$ de puntos en el plano que garantiza un polígono convexo de n lados [10].

c) Seis grados de separación. Aunque popularizado por Milgram, este principio se modela matemáticamente a través de las redes *small-world* [43]. Implica que el diámetro del grafo social humano es logarítmicamente pequeño respecto al número de habitantes [29, 4].

5. Significado profundo. El teorema de Ramsey ilustra el principio de que “**el orden es inevitable dentro del caos**”. Este concepto atraviesa la física estadística y la sociología matemática, proporcionando un marco para entender cómo surgen patrones en redes de gran escala [2, 3].

1.5. Teoría de Redes Sociales

La teoría de redes sociales aplica los conceptos y herramientas de la teoría de grafos al estudio de las relaciones entre individuos, grupos o entidades [30]. En lugar de vértices y aristas abstractos, las redes sociales representan personas (nodos) y vínculos como amistad, comunicación o influencia [4].

Comprender su estructura permite analizar fenómenos como la difusión de información, la propagación de ideas o la formación de comunidades [16]. A diferencia de los grafos teóricos clásicos, las redes sociales reales presentan patrones de conexión no aleatorios y propiedades emergentes que las distinguen de los modelos puramente aleatorios [2].

1.5.1. Tipología de Redes

1. Introducción general. Los estudios clásicos identificaron tres modelos fundamentales para describir las redes reales: las **redes aleatorias**, las **redes de tipo small-world** y las **redes libres de escala** [6]. Más tarde se añadieron modelos específicos, como las **redes bibliográficas**, que reflejan relaciones de coautoría o citación científica [29].

2. Redes aleatorias (modelo de Erdős–Rényi). El modelo más simple de red fue propuesto por Erdős y Rényi (1959). Parte de n vértices y conecta cada par con probabilidad p [12].

$$G(n, p) : \Pr((u, v) \in E) = p$$

Características principales:

- La distribución de grados sigue aproximadamente una distribución de Poisson para grandes n [30].
- La conectividad global aparece bruscamente cuando p supera el umbral crítico $p_c = \frac{\ln n}{n}$ [12].

3. Redes small-world (modelo de Watts–Strogatz). Duncan Watts y Steven Strogatz (1998) descubrieron que muchas redes presentan alta agrupación local y pequeña longitud media de camino simultáneamente [43].

El modelo de Watts–Strogatz conserva el agrupamiento (*clustering*) pero reduce drásticamente la distancia media entre nodos mediante la reconfiguración aleatoria de aristas [43].

Ejemplo social: El experimento de los “seis grados de separación” de Milgram se explica matemáticamente mediante este modelo de conectividad global [29].

4. Redes libres de escala (modelo de Barabási–Albert). Barabási y Albert (1999) demostraron que redes como Internet o las redes de citas siguen una distribución de grados de ley de potencia [5]:

$$P(k) \sim k^{-\gamma}$$

Mecanismo generador: Preferential attachment. Los nuevos nodos tienden a conectarse con aquellos que ya son populares (“el rico se hace más rico”) [2].

Consecuencias: Estas redes son robustas frente a fallos aleatorios pero extremadamente vulnerables ante ataques dirigidos a los *hubs* [4, 2].

5. Redes bibliográficas. Estas redes reflejan relaciones de coautoría o citación. Suelen mostrar propiedades híbridas: fuerte agrupación local y la presencia de autores influyentes que actúan como *hubs* [29]. El “índice de Erdős” es la aplicación práctica más conocida de la distancia geodésica en este tipo de redes académicas [14].

1.5.2. Propiedades Estructurales

1. Concepto general. Para comprender el comportamiento de una red, no basta con observar quién está conectado con quién: hay que medir **cómo** está conectada mediante métricas de centralidad, densidad y modularidad [6, 14].

1.6. Dinámica y estructura en redes

Hasta ahora hemos analizado las redes desde un punto de vista estructural. Sin embargo, las redes también sirven como soporte para procesos dinámicos: información que se difunde, enfermedades que se propagan o señales que se sincronizan [4]. El estudio de la **dinámica sobre redes** permite comprender fenómenos complejos en múltiples disciplinas [6]. Por otro lado, dentro de la estructura misma, pueden detectarse **comunidades, módulos y patrones recurrentes** que reflejan la organización interna del sistema [16].

1.6.1. Dinámica sobre una red

1. Introducción intuitiva. Las dinámicas de propagación, ya sean de rumores o virus, pueden modelarse matemáticamente como procesos que evolucionan sobre los vértices y

aristas de un grafo [30].

2. Difusión de información. La difusión se modela mediante un proceso de propagación probabilística donde la estructura de la red (densidad, hubs, modularidad) afecta la velocidad de difusión [2]. Las redes libres de escala, por ejemplo, difunden información más rápido debido a la presencia de *hubs* [5].

3. Propagación de epidemias (modelo SIR). El modelo clásico SIR divide la población en Susceptibles, Infectados y Recuperados [21]. Sobre una red, el parámetro clave es el número básico de reproducción R_0 , que depende críticamente de la topología [32]. En redes con alta heterogeneidad de grado, el umbral epidémico tiende a desaparecer, facilitando la propagación [1].

4. Sincronización. La sincronización describe cómo osciladores independientes se coordinan a través de sus conexiones [3]. Un modelo fundamental es el de **Kuramoto** [24]:

$$\frac{d\theta_i}{dt} = \omega_i + K \sum_j A_{ij} \sin(\theta_j - \theta_i)$$

La sincronización emerge cuando el acoplamiento K supera un umbral crítico que depende de la estructura de la red [40].

5. Flujos y transporte. El teorema de flujo máximo–corte mínimo de Ford–Fulkerson establece que el flujo máximo entre un origen y un destino es igual a la capacidad mínima del corte que los separa [14]. Este principio vincula la dinámica de transporte con la conectividad estructural.

1.6.2. Detección de comunidades

Las redes reales no se distribuyen de manera uniforme, sino que tienden a organizarse en **comunidades**: subgrupos de nodos densamente interconectados entre sí que mantienen una conectividad escasa con el resto del grafo [16, 17]. Este fenómeno topológico se analiza habitualmente mediante el concepto de **modularidad** (Q), una métrica introducida por Newman y Girvan [17] que cuantifica la calidad de una partición comparándola con un modelo nulo aleatorio, y cuya expresión formal es:

$$Q = \frac{1}{2m} \sum_{i,j} [A_{ij} - \frac{k_i k_j}{2m}] \delta(c_i, c_j).$$

Más allá de los métodos clásicos como el algoritmo de Girvan-Newman, basado en la intermediación de aristas, o el método de Louvain para redes de gran escala [16], la estructura de las redes revela organizaciones más complejas de tipo jerárquico o núcleo-periferia (*core-periphery*), fundamentales para la resiliencia sistémica [14, 4]. En esta

escala micro-estructural emergen los **motifs**, subgrafos recurrentes que actúan como bloques funcionales al aparecer con una frecuencia significativamente superior a la esperada en modelos aleatorios [6].

Sin embargo, es importante destacar que la identificación precisa de estas comunidades basándose únicamente en la topología estática sigue siendo un **problema abierto** y computacionalmente costoso en la teoría de redes actual. Mientras que los enfoques aquí descritos son puramente estructurales, la verdadera naturaleza funcional de una comunidad a menudo solo se revela a través de la actividad de sus nodos. Por ello, este desafío analítico se abordará desde una perspectiva radicalmente distinta en el **Bloque 5** (ver Sección 5.1.1), donde se demostrará cómo la **sincronización dinámica** de osciladores permite emerger clústeres funcionales que superan las limitaciones de los métodos basados exclusivamente en la matriz de adyacencia [3]. Esta transición conceptual de la estructura a la dinámica subraya que, en los sistemas complejos, la organización de una comunidad es inseparable de los procesos de sincronía que ocurren sobre sus enlaces.

1.7. Redes complejas y su tratamiento estadístico

Las **redes complejas** constituyen el nivel más avanzado del estudio de los grafos. A diferencia de los modelos clásicos, representan sistemas reales en los que las conexiones siguen patrones estructurales y estadísticos propios [29]. Estas redes aparecen en contextos diversos como la biología, la economía o las redes sociales digitales, y su análisis ha permitido descubrir leyes universales del comportamiento colectivo [2].

1.7.1. Características de una red compleja

Las redes complejas presentan una serie de rasgos distintivos [4]:

- **Heterogeneidad estructural:** los nodos no tienen el mismo número de conexiones; la presencia de *hubs* es una característica definitoria [5].
- **Jerarquía y modularidad:** las redes se organizan en comunidades interconectadas donde pequeños módulos forman estructuras de orden superior [17, 16].
- **Alta robustez y vulnerabilidad selectiva:** son resistentes ante fallos aleatorios pero extremadamente frágiles ante ataques dirigidos a los *hubs* [2].
- **Autoorganización:** surgen a través de procesos locales de crecimiento, como el *preferential attachment* [2].

1.7.2. Tratamiento estadístico

El análisis de redes complejas se basa en herramientas de la física estadística que describen su estructura [6, 14]:

- **Distribución de grados:** en redes libres de escala, sigue una ley de potencias $P(k) \sim k^{-\gamma}$ [5].
- **Medidas de centralidad:** cuantifican la influencia de un nodo. Destacan la *intermediación* (betweenness) y el *vector propio* (eigenvector), esenciales para identificar cuellos de botella y nodos influyentes [29].
- **Coefficiente de clustering:** mide la cohesión local y la tendencia a formar triángulos [43].
- **Detección de comunidades:** el uso de la modularidad permite identificar grupos con fuerte conexión interna, reflejando la organización funcional del sistema [17, 16].

Estas medidas permiten cuantificar propiedades globales como la eficiencia de transmisión o la resiliencia [3]. El tratamiento estadístico combina hoy matemáticas discretas, física y ciencias sociales para descifrar la complejidad de los sistemas interconectados [30].

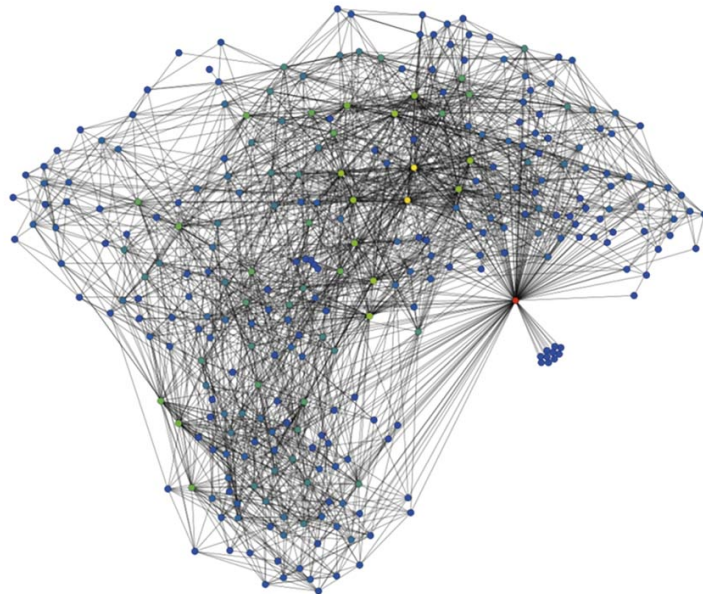


Figura 1.4: Ejemplo esquemático de una red compleja con hubs, comunidades y enlaces heterogéneos [4].

Cuadro 1.7: Resumen general de la teoría de grafos y redes complejas.

Bloque	Concepto central	Aplicaciones principales
Fundamentos	Vértices y aristas [10]	Modelos de relaciones
Algoritmos	Dijkstra, Prim, Kruskal [14]	Optimización de rutas y costes
Problemas clásicos	Euler, TSP, coloración [15]	Logística y planificación
Combinatoria	Ramsey y orden inevitable [10]	Probabilidad y patrones
Redes sociales	Small-world y leyes de potencia [43, 5]	Difusión e influencia
Dinámica	Modelo SIR y sincronía [32, 24]	Epidemias y coordinación
Redes complejas	Heterogeneidad y robustez [2]	Internet, biología e IA

*Una vez asentados los fundamentos de la teoría de grafos y comprendidos los algoritmos que permiten recorrer estructuras elementales, se hace evidente que las redes del mundo real poseen una complejidad que trasciende los grafos estáticos de pocos nodos. Problemas clásicos como el de los puentes de Königsberg o la coloración de mapas sirven de base pedagógica, pero no resultan suficientes para caracterizar sistemas masivos y dinámicos. Por ello, en el siguiente capítulo, nos adentraremos en las **Métricas Estructurales**, herramientas que nos permitirán "medir" la topología de las redes sociales y entender cómo su arquitectura intrínseca determina tanto la importancia de los individuos como la cohesión de los grupos.*

Capítulo 2

Métricas Estructurales y Modelos de Redes

2.1. Métricas de Caracterización de Redes

2.1.1. Caminos y propiedades básicas

Camino sobre un grafo

Un **camino** en un grafo $G = (V, E)$ es una secuencia finita de vértices

$$v_1, v_2, \dots, v_k$$

tal que cada par consecutivo de vértices (v_i, v_{i+1}) está conectado por una arista del grafo [10]. En un grafo orientado, las aristas deben seguir la dirección especificada [30].

- Si no se repite ningún vértice (excepto, opcionalmente, el primero y el último), el camino se denomina **camino simple**.
- Si el primer y el último vértice coinciden, el camino es un **ciclo**.

El número de aristas que lo componen se llama **longitud del camino** [28]. Por ejemplo, en el grafo con aristas $A-B$, $B-C$, $C-D$, la secuencia $A \rightarrow B \rightarrow C \rightarrow D$ es un camino de longitud 3.

Camino geodésico

Un **camino geodésico** entre dos vértices u y v es el camino de menor longitud que los conecta [30]. Su longitud define la **distancia geodésica** entre ambos nodos, denotada $d(u, v)$.

$$d(u, v) = \min_{\text{todos los caminos de } u \text{ a } v} \text{longitud del camino}$$

En grafos ponderados, donde las aristas tienen un peso w_{ij} , la longitud del camino se define como la suma de los pesos de las aristas que lo componen, y se utilizan algoritmos como Dijkstra o Floyd–Warshall para encontrar el camino geodésico [14, 22].

Ejemplo: Si $A-B-C$ y también existe la arista directa $A-C$, entonces:

$$d(A, C) = 1 \quad (\text{camino geodésico directo}).$$

El camino $A \rightarrow B \rightarrow C$ sería más largo, con distancia 2.

Camino euleriano

Un **camino euleriano** es aquel que recorre cada **arista** del grafo exactamente una vez. Si el camino comienza y termina en el mismo vértice, se denomina **circuito euleriano** [15].

Condiciones de existencia:

- En un grafo no orientado: Existe un camino euleriano si y solo si tiene **exactamente dos vértices de grado impar** [10]. Si todos los vértices tienen grado par, entonces existe un circuito euleriano.
- En un grafo orientado: Existe un camino euleriano si y solo si todos los vértices, salvo dos, tienen el mismo número de aristas de entrada y salida; los dos restantes difieren en una unidad.

El concepto proviene del famoso *problema de los puentes de Königsberg*, resuelto por Euler en 1736 [15], considerado el origen de la teoría de grafos.

Camino hamiltoniano

Un **camino hamiltoniano** es un camino que recorre cada **vértice** del grafo exactamente una vez [10]. Si además el camino forma un ciclo, es decir, empieza y termina en el mismo vértice, se denomina **circuito hamiltoniano**.

Diferencias con el camino euleriano:

- El camino euleriano pasa por todas las *aristas* una sola vez.
- El camino hamiltoniano pasa por todos los *vértices* una sola vez.

A diferencia del problema euleriano, ****no existe una condición simple**** que garantice la existencia de un camino hamiltoniano; su detección pertenece a los problemas NP-completos [14].

Ejemplo comparativo: Camino Euleriano vs Hamiltoniano

Consideremos el siguiente grafo no orientado:

$$A - B - C - D - A - C$$

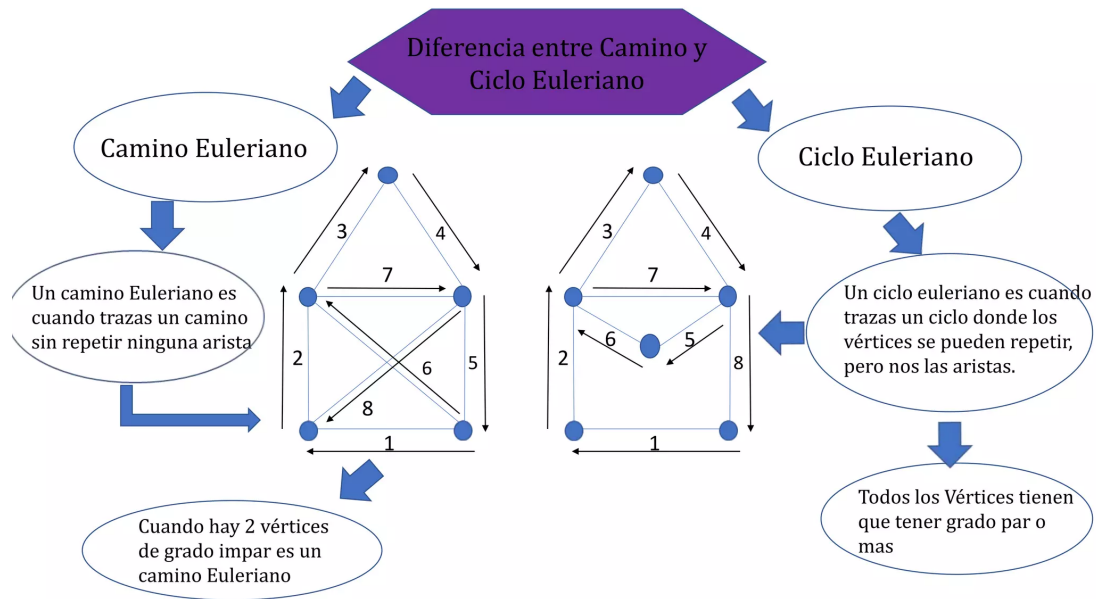


Figura 2.1: Comparativa entre camino euleriano y ciclo hamiltoniano. Fuente: Adaptado de [15, 10].

- **Camino euleriano:** Un posible recorrido euleriano es: $A \rightarrow B \rightarrow C \rightarrow A \rightarrow D \rightarrow C$. Aquí se recorren todas las aristas una vez, pero el nodo A se visita más de una vez (lo cual está permitido).
- **Camino hamiltoniano:** Un posible camino hamiltoniano es: $A \rightarrow B \rightarrow C \rightarrow D$. En este caso se visita cada vértice exactamente una vez, pero no necesariamente todas las aristas.

2.1.2. Grado y densidad

Comprender cómo están conectados los nodos dentro de un grafo es como analizar una red de amistades, llamadas o conexiones neuronales [11]. Algunas personas (nodos) son más populares que otras, algunas redes están llenas de vínculos, y otras son más esporádicas. Para medir esto matemáticamente entran en juego tres conceptos clave: **grado**, **distribución de grados** y **densidad** [4, 26].

Grado de un nodo

El **grado** de un nodo v , denotado como k_v , es el número de conexiones (aristas) que tiene con otros nodos [10].

$$k_v = |\{ u \in V : (u, v) \in E \}|$$

- En un **grafo no orientado**, el grado es simplemente el número de enlaces.

- En un **grafo orientado**, distinguimos grado de entrada (k_{in}) y grado de salida (k_{out}).

Observación: La suma de todos los grados en un grafo no orientado es siempre el doble del número de aristas: $\sum_{v \in V} k_v = 2|E|$. Esta identidad se conoce como el *teorema del apretón de manos* [30].

Distribución de grados

La **distribución de grados** describe la probabilidad de que un nodo elegido al azar tenga un grado determinado k [2]:

$$P(k) = \frac{\text{Número de nodos con grado } k}{N}$$

Es una de las formas más importantes de caracterizar la estructura de una red [5].

Tres formas típicas de distribuciones:

1. **Distribución binomial (redes aleatorias tipo Erdős–Rényi):** Característica del modelo $G(N, p)$ [12]. La mayor parte de los nodos tiene grados cercanos a la media. Este comportamiento explica por qué estas redes suelen tener distancias cortas pero bajo clustering [43].
2. **Distribución homogénea (redes regulares):** Todos los nodos tienen el mismo grado. Estas redes sirven como referencia para la transición regular–aleatoria del modelo de Watts–Strogatz [43].
3. **Distribución de ley de potencias (redes libres de escala):** Propia de modelos como Barabási–Albert [5, 9], donde la aparición de *hubs* genera estructuras altamente heterogéneas: $P(k) \sim k^{-\gamma}$. Esta distribución es clave para explicar la resiliencia frente a fallos y la vulnerabilidad a ataques [2].

Densidad del grafo

La **densidad** mide qué tan lleno de conexiones está un grafo respecto al número máximo posible de aristas [28]:

$$D(G) = \frac{2|E|}{N(N-1)}$$

Ejemplo: Si una red tiene $N = 5$ nodos y $E = 4$ aristas: $D(G) = \frac{8}{20} = 0,4$. La red tiene el 40 % de las conexiones posibles. Una densidad de 1 significa un grafo completo; una densidad baja significa grupos aislados [26].

Concepto	Símbolo / Fórmula	Significado intuitivo
Grado de un nodo	k_v	Número de conexiones de un nodo
Distribución de grados	$P(k)$	Probabilidad de que un nodo tenga grado k
Densidad del grafo	$D(G) = \frac{2E}{N(N-1)}$	Porcentaje de conexiones posibles existentes

Cuadro 2.1: Resumen de medidas locales de conectividad en un grafo.

2.1.3. Medidas globales

Diámetro de un grafo

El **diámetro** es la mayor distancia geodésica existente entre cualquier par de vértices del grafo [30]:

$$D(G) = \max_{u,v \in V} d(u, v)$$

El diámetro cuantifica cuán “extendida” es una red. En redes reales, tiende a ser sorprendentemente pequeño incluso para redes enormes, fenómeno conocido como el *efecto mundo pequeño* [43, 11].

Distancia media sobre un grafo

La **distancia media** $L(G)$ mide cuán lejos están, en promedio, dos nodos cualesquiera del grafo [29]:

$$L(G) = \frac{1}{N(N-1)} \sum_{\substack{u,v \in V \\ u \neq v}} d(u, v)$$

Un valor pequeño indica una red eficiente. En redes reales, $L(G)$ suele crecer como $\log N$ [43].

Problemas en el cálculo de distancias medias

El cálculo presenta diversas dificultades [42, 22]:

1. **Grafo no conexo:** Si existen pares sin camino, $d(u, v) = \infty$. Suele calcularse sobre la *componente gigante* [30].
2. **Coste computacional:** Calcular todas las distancias tiene complejidad $O(N^3)$ [14].
3. **Pesos y direccionalidad:** Complican el cálculo en redes dirigidas o ponderadas.
4. **Efecto de la topología:** Las redes de "mundo pequeño" reducen drásticamente la distancia media gracias a los hubs [4].

En la Figura 2 se muestra un grafo ponderado. El **problema del camino más corto** consiste en determinar la ruta con menor coste. Para calcular la **distancia media**, se obtiene el promedio de las longitudes de los caminos más cortos entre todos los pares:

$$\bar{d} = \frac{1}{\binom{n}{2}} \sum_{i \neq j} d(i, j)$$

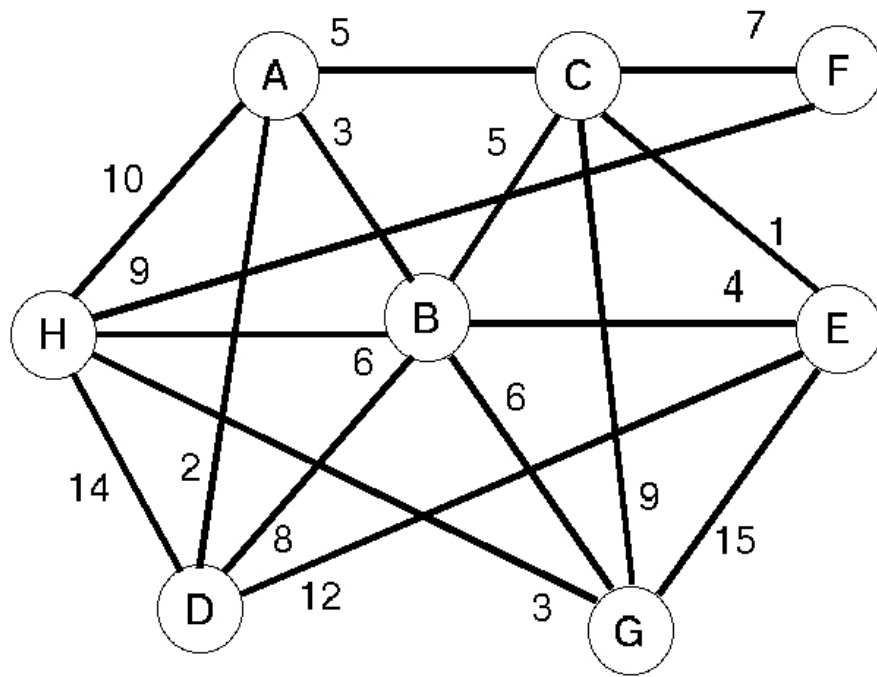


Figura 2.2: Representación del problema del camino más corto en un grafo ponderado.
Fuente: [22, 14].

Este cálculo requiere aplicar algoritmos como Dijkstra o Floyd-Warshall [22, 14].

2.2. Conectividad y Componentes

2.2.1. Componentes Conexas

Una **componente conexa** es un conjunto maximal de nodos donde todos están comunicados mediante algún camino [10]. Formalmente, dos vértices u, v pertenecen a la misma componente si existe un camino entre ellos. El grafo puede descomponerse como:

$$V = C_1 \cup C_2 \cup \dots \cup C_k, \quad C_i \cap C_j = \emptyset.$$

Ejemplo: Con aristas $(A, B), (B, C), (D, E)$, las componentes son $\{A, B, C\}$ y $\{D, E\}$. En particular, en grafos dirigidos se distinguen dos tipos fundamentales de conectividad [30, 26]:

- **Componente fuertemente conexa (SCC):** todos los nodos son mutuamente alcanzables, es decir, existen caminos $u \rightarrow v$ y $v \rightarrow u$.
- **Componente débilmente conexa:** conjunto conexo al ignorar la dirección de las aristas.

Ejemplo: En $A \rightarrow B \rightarrow C$, los nodos son débilmente conexos pero no fuertemente conexos.

Nodos mutuamente alcanzables: Dos nodos son mutuamente alcanzables si existe un camino en ambos sentidos [10]. El conjunto maximal de nodos mutuamente alcanzables forma una **SCC**, núcleo estructural de un grafo dirigido [28].

2.2.2. Componente Gigante

La **componente gigante** se define como la componente conexa más extensa de un grafo, la cual alberga una fracción finita y significativa del total de nodos del sistema (N) [4]. Según la teoría fundamental de grafos aleatorios, la emergencia de esta estructura macroscópica no es un proceso lineal, sino un fenómeno crítico que surge abruptamente cuando el grado medio de la red supera el umbral de criticidad $\langle k \rangle_c = 1$ [12, 2]. Bajo esta transición, si $\langle k \rangle < 1$, el sistema se mantiene en una fase subcrítica compuesta por pequeñas islas o clústeres aislados, mientras que para $\langle k \rangle > 1$, la red entra en una fase supracrítica donde la mayoría de los nodos se integran en una única estructura global de comunicación.

En el contexto de las redes dirigidas de gran escala, como la World Wide Web, esta organización se vuelve más compleja, distinguiéndose el **GSCC** (*Giant Strongly Connected Component*) como un núcleo central donde existe un camino bidireccional entre cualquier par de nodos, y la **GCC** (*Giant Connected Component*), que representa la conectividad global ignorando la dirección de los enlaces e integrando las regiones periféricas [29].

Es fundamental comprender que la aparición de esta componente gigante es, en esencia, el resultado directo de un proceso de **percolación**. Aunque en esta sección se analiza

desde un punto de vista estático y descriptivo, la componente gigante representa el *clúster percolante* cuya formación y estabilidad frente a la eliminación de nodos o enlaces constituye el núcleo del análisis de resiliencia. Esta relación intrínseca entre la estructura aquí presentada y la dinámica de fallos será explorada en profundidad en el **Bloque 4** (ver Sección 4.1), donde se estudiará cómo la teoría de la percolación permite predecir el colapso de la conectividad global ante ataques dirigidos o fallos aleatorios [32].

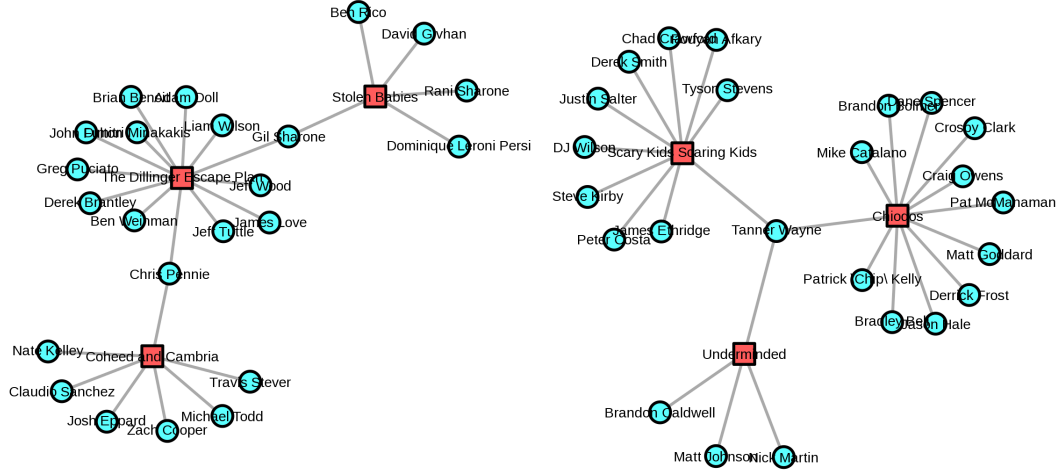


Figura 2.3: Representación de la emergencia de la componente gigante (GCC) en una red aleatoria. Fuente: [12, 4].

2.2.3. Componentes Desconectadas

Se refieren a subgrafos completamente aislados, sin relación con la componente gigante [14]. Ejemplos comunes incluyen páginas web sin enlaces entrantes ni salientes o usuarios sin conexiones en una red social.

2.2.4. Transición de Fase

El concepto de *transición de fase* permite entender cómo, al incrementar ligeramente la probabilidad de conexión entre nodos, una red aleatoria pasa de estar fragmentada en pequeñas islas a formar una estructura conectada a gran escala [12, 30]. En modelos como $G(N, p)$, este cambio ocurre de manera abrupta alrededor del punto crítico [4]:

$$p_c \approx \frac{1}{N}.$$

Este fenómeno es análogo a los procesos de percolación estudiados en física estadística [32].

2.2.5. Estructura en Grafos Dirigidos

Siguiendo el modelo de "pajarita" (*bow-tie structure*) propuesto para la red global [11], se identifican:

Tendrils (zarcillos)

Nodos conectados a la GSCC sin formar parte de su núcleo de mutua alcanzabilidad [28]:

- **IN-tendrils:** nodos que pueden llegar a la GSCC, pero desde los cuales no se puede regresar.
- **OUT-tendrils:** nodos a los que se puede llegar desde la GSCC, pero que no tienen camino de retorno al núcleo.

Tubes (túneles)

Los *tubes* conectan nodos de la región IN con nodos de la región OUT sin pasar necesariamente por la GSCC, actuando como puentes laterales entre regiones periféricas de la red [11].

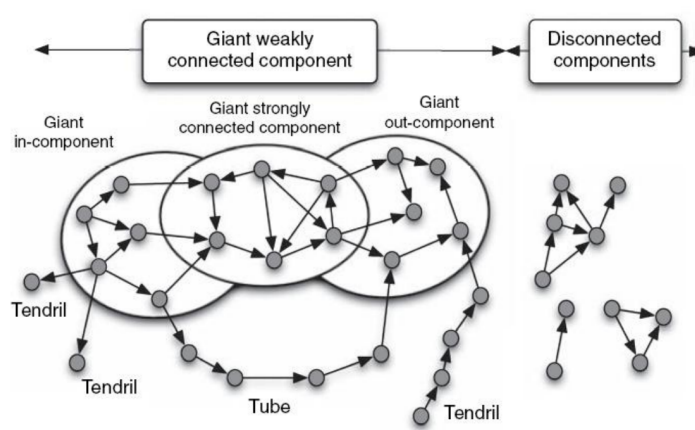


Figura 2.4: Estructura en grafos dirigidos: núcleos (GSCC), zarcillos (tendrils) y túneles (tubes). Fuente: [11, 28].

2.3. Propiedades estructurales y métricas

2.3.1. Medidas de centralidad

Las **medidas de centralidad** permiten identificar qué nodos son más importantes o influyentes en una red [30, 14]. Cada medida responde a una pregunta distinta:

- **Grado (Degree):** ¿Cuántas conexiones directas tiene un nodo?
- **Cercanía (Closeness):** ¿Qué nodo está más cerca del resto?
- **Intermediación (Betweenness):** ¿Qué nodo actúa como puente entre otros?

Regla mnemotécnica: *Grado = Amigos, Cercanía = Atajos, Intermediación = Puentes*. (Fácil de recordar: “A-P-P” → Amigos, Pasos, Puentes).

Centralidad de grado

La **centralidad de grado** mide el número de conexiones directas de un nodo [10, 30].

$$C_D(v) = k_v$$

Si se desea comparar redes de distinto tamaño, puede normalizarse:

$$C_D(v) = \frac{k_v}{N - 1}$$

donde N es el número total de nodos.

Ejemplo:

Supongamos el siguiente grafo no orientado:

Las aristas son: A–B, A–D, B–C y B–E.

Nodo	k_v
A	2 (B, D)
B	3 (A, C, E)
C	1 (B)
D	1 (A)
E	1 (B)

El nodo más central según esta medida es **B**, con 3 conexiones directas. Si sumas todos los grados, $2 + 3 + 1 + 1 + 1 = 8 = 2|E|$. Esto cumple el *teorema del apretón de manos* [30].

Centralidad de cercanía (closeness)

La **centralidad de cercanía** mide cuán “cerca” está un nodo del resto, es decir, la inversa de la distancia media a todos los demás [30, 26].

$$C_C(v) = \frac{N - 1}{\sum_{u \neq v} d(v, u)}$$

donde $d(v, u)$ es la longitud del camino más corto entre v y u [11].

Usando el mismo grafo que en el ejemplo anterior:

De... / A...	A	B	C	D	E
A	0	1	2	1	2
B	1	0	1	2	1
C	2	1	0	3	2
D	1	2	3	0	3
E	2	1	2	3	0

Sumando distancias y aplicando la fórmula:

Nodo	Suma de distancias	$C_C(v) = \frac{N-1}{\sum d(v,u)}$
A	6	$\frac{4}{6} = 0,667$
B	5	$\frac{4}{5} = 0,800$
C	8	$\frac{4}{8} = 0,500$
D	9	$\frac{4}{9} = 0,444$
E	8	$\frac{4}{8} = 0,500$

El nodo **B** tiene la mayor cercanía (0.8): es el que puede comunicarse con el resto en menos pasos promedio. Por tanto, B es el “centro geográfico” de la red; está a pocos saltos de todos [14].

Truco: *Cuanto más cerca del resto, mayor cercanía $\rightarrow$ menos pasos = más central.*

Centralidad de intermediación (betweenness)

La **centralidad de intermediación** mide cuántos caminos más cortos entre pares de nodos pasan por el nodo v [30]. Si un nodo aparece con frecuencia en los caminos más cortos, actúa como “puente” o “intermediario” [14].

$$C_B(v) = \sum_{\substack{s \neq v \neq t \\ s \neq t}} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

donde:

- σ_{st} = número total de caminos más cortos entre los nodos s y t ,
- $\sigma_{st}(v)$ = número de esos caminos que pasan por v .

Usamos de nuevo el grafo, y recuerda: *Betweenness* = “Entre” $\rightarrow$ mide quién está entre los demás. Los caminos más cortos entre todos los pares de nodos son:

Pares (s,t)	Camino más corto	Nodos intermedios
A–B	A–B	—
A–C	A–B–C	B
A–D	A–D	—
A–E	A–B–E	B
B–C	B–C	—
B–D	B–A–D	A
B–E	B–E	—
C–D	C–B–A–D	B, A
C–E	C–B–E	B
D–E	D–A–B–E	A, B

Contamos en cuántos caminos mínimos actúa cada nodo como intermediario (sin contar los extremos):

Nodo	Número de caminos donde actúa como intermediario
A	2 ($B - D, D - E$)
B	6 ($A - C, A - E, C - D, C - E, D - E, A - B - E$)
C	0
D	0
E	0

El nodo **B** tiene la mayor intermediación: aparece en la mayoría de los caminos más cortos entre otros nodos. Por tanto, se podría decir que B es el puente de comunicación entre grupos; si B desaparece, la red se fragmenta [14].

Curiosidad: Esta medida se utiliza en el algoritmo de *Girvan–Newman* para detectar comunidades: se eliminan las aristas con mayor betweenness y la red se divide de forma natural [17].

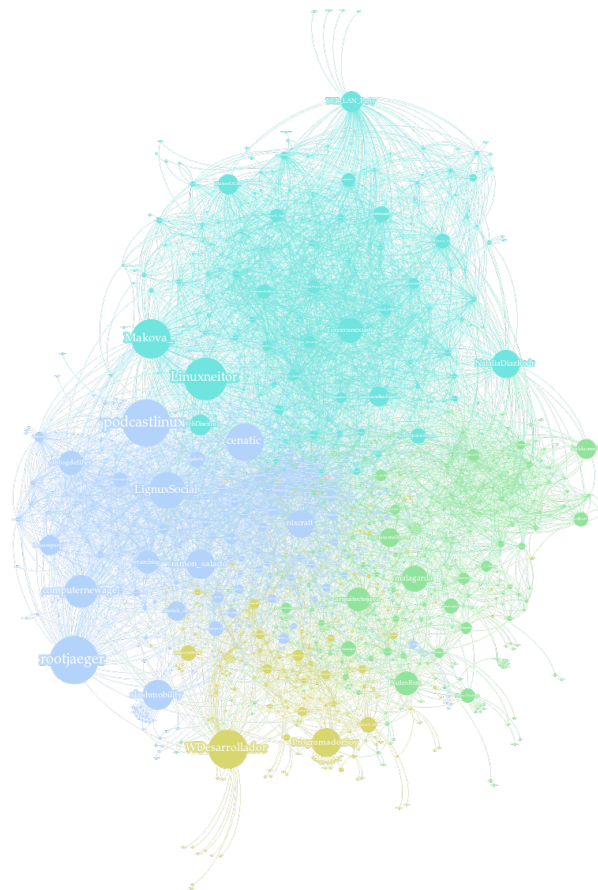


Figura 2.5: Visualización de una red social (Twitter) y sus métricas de centralidad. Fuente: [11].

En la red de la Figura 2.5, las principales medidas de centralidad permiten identificar distintos roles estructurales [11]:

- **Grado (degree):** los nodos con mayor número de conexiones representan usuarios altamente activos o populares.

- **Cercanía (closeness):** mide qué tan rápido puede un nodo alcanzar al resto. Los nodos más centrales presentan una menor distancia media hacia otros usuarios.
- **Intermediación (betweenness):** identifica usuarios que actúan como puentes entre comunidades, facilitando el flujo de información entre ellos [17].

Comparativa

Centralidad	Qué mide	Fórmula base	Recuerdo
Grado	Popularidad local	k_v	<i>Amigos</i>
Cercanía	Accesibilidad global	$\frac{N-1}{\sum d(v,u)}$	<i>Pasos</i>
Intermediación	Puente/Corretaje	$\sum \sigma_{st}(v)/\sigma_{st}$	<i>Puentes</i>

Consejos prácticos

1. Si tu red está fragmentada, usa **cercanía armónica** en lugar de la clásica [28].
2. Normaliza (cuando compares) para redes de distinto tamaño [30].
3. En ponderadas, revisa el significado del “peso”: ¿más peso es “más cercano” o “más costoso”? Ajusta antes de calcular caminos mínimos [22].

2.3.2. Estructura local

Antes de estudiar propiedades globales de una red, es útil analizar su *estructura local*, es decir, la organización que rodea a cada nodo y a sus vecinos inmediatos [43]. Estas medidas describen el “microentorno” de un vértice: cómo se relacionan sus amigos, con quién se conectan los hubs o cómo es la forma del subgrafo que lo rodea [30].

En este contexto, una **definición local** es aquella que se calcula usando únicamente la información del nodo y de sus vecinos directos, sin considerar toda la red completa.

Coefficiente de *clustering*

Idea intuitiva Si tus amigos también son amigos entre sí, tu barrio social es “cozy”: hay triángulos por todas partes. El **clustering** mide cuán probable es que *los amigos de un nodo sean amigos entre sí* [43].

Definición local. Sea v un nodo con grado k_v y T_v el número de *triángulos* que tienen a v como vértice. El número máximo de enlaces entre los vecinos de v es $\binom{k_v}{2} = \frac{k_v(k_v-1)}{2}$. El **coeficiente de clustering local** de v es [43, 4]:

$$C_v = \frac{\text{aristas observadas entre vecinos de } v}{\text{aristas posibles entre vecinos de } v} = \frac{2T_v}{k_v(k_v-1)}, \quad k_v \geq 2,$$

y por convenio $C_v = 0$ si $k_v < 2$.

Como $0 \leq T_v \leq \binom{k_v}{2}$, se sigue inmediatamente $0 \leq C_v \leq 1$. Además, $C_v = 1$ si y solo si los vecinos de v forman un clique, y $C_v = 0$ si no hay ninguna arista entre sus vecinos [10].

Ejemplo. Si $k_v = 4$ y entre los 4 vecinos existen 3 aristas (formando un triángulo y una arista suelta), entonces $T_v = 1$ y:

$$C_v = \frac{2 \cdot 1}{4 \cdot 3} = \frac{1}{6}.$$

Clustering = Triángulos. Piensa en “amigos de mis amigos” $\Rightarrow$ triángulos $\Rightarrow$ comodidad local.

Correlación de grados entre vecinos

Idea intuitiva ¿Los nodos con muchos enlaces tienden a conectarse con otros muy conectados (*asortativos*), o más bien con nodos de bajo grado (*disasortativos*)? Esto capta *con quién se juntan los hubs* [29].

Grado medio de los vecinos. Una forma descriptiva es la función [6]:

$k_{nn}(k)$ = promedio del grado de los vecinos de nodos con grado k .

- Si $k_{nn}(k)$ *crece* con k , la red es **asortativa**: los de alto grado tienden a conectarse con altos.
- Si $k_{nn}(k)$ *decrece* con k , la red es **disasortativa**: los hubs conectan con periféricos.

Coefficiente de asortatividad de Newman (definición por Pearson). Sea $M = |E|$ el número de aristas. Para cada arista $e = \{i, j\}$, denote k_i, k_j los grados de sus extremos. El **coeficiente de asortatividad** r es la correlación de Pearson entre los grados en los extremos de una arista [29, 30]:

$$r = \frac{\frac{1}{M} \sum_{\{i,j\} \in E} k_i k_j - \left(\frac{1}{2M} \sum_{\{i,j\} \in E} (k_i + k_j) \right)^2}{\frac{1}{2M} \sum_{\{i,j\} \in E} (k_i^2 + k_j^2) - \left(\frac{1}{2M} \sum_{\{i,j\} \in E} (k_i + k_j) \right)^2}.$$

- $r \in [-1, 1]$; $r > 0$ indica **asortatividad** y $r < 0$ **disasortatividad** [29].
- *Equivalente elegante (con grados de exceso).* Si e_{jk} es la prob. de que una arista conecte nodos de *exceso de grado* j y k y q_k la distribución marginal, entonces

[30, 14]:

$$r = \frac{\sum_{jk} jk (e_{jk} - q_j q_k)}{\sigma_q^2},$$

donde σ_q^2 es la varianza de q_k . Esta forma resalta que r mide *desviación respecto a independencia*.

2.4. Modelos de grafos aleatorios

Introducción. Los modelos de grafos aleatorios permiten generar redes sintéticas para analizar propiedades estructurales y fenómenos emergentes, como la aparición de componentes gigantes o la distribución de distancias [30, 4]. Los dos modelos más comunes son el **modelo de Erdős–Rényi** y el **modelo de configuración**, que difieren en cómo controlan los grados de los nodos [14].

2.4.1. Grafos aleatorios generalizados

Los **grafos aleatorios generalizados** se utilizan para estudiar propiedades globales de redes heterogéneas, como la aparición de componentes gigantes y la distribución de distancias [29].

Funciones generadoras. Se define la función generadora del grado [30]:

$$G_0(x) = \sum_{k=0}^{\infty} P(k) x^k$$

y la función generadora del *exceso de grado* [28]:

$$G_1(x) = \frac{G'_0(x)}{G'_0(1)} = \sum_{k=0}^{\infty} \frac{(k+1)P(k+1)}{\langle k \rangle} x^k$$

Estas funciones permiten calcular de manera compacta el tamaño esperado de las componentes y la aparición de la componente gigante [29, 6].

Componente gigante. Si u es la probabilidad de que, al seguir una arista al azar, no se llegue a la componente gigante, entonces u satisface [30]:

$$u = G_1(u), \quad S_{\text{GCC}} = 1 - G_0(u)$$

Valores bajos de u indican que muchas trayectorias alcanzan la red principal (alta conectividad), mientras que $u \approx 1$ refleja una red fragmentada [14].

2.4.2. Modelo de Erdős–Rényi

Motivación: número de Erdős. Antes de definir el modelo, es útil ilustrar la idea de distancia geodésica con el **número de Erdős** en un grafo de coautorías [30]:

- La **egonet** de un nodo v es el subgrafo inducido por v y sus vecinos directos [26].
- El **número de Erdős** de un autor x es la longitud mínima del camino desde x hasta Paul Erdős [29]:

$$\text{Erds}(x) = d_{G_{\text{coaut}}}(x, \text{Erdős})$$

donde G_{coaut} es el grafo no orientado de coautorías.

La mayoría de autores están a pocos saltos de Erdős, ilustrando un ejemplo de *small world* [43] (Figura 2.6).

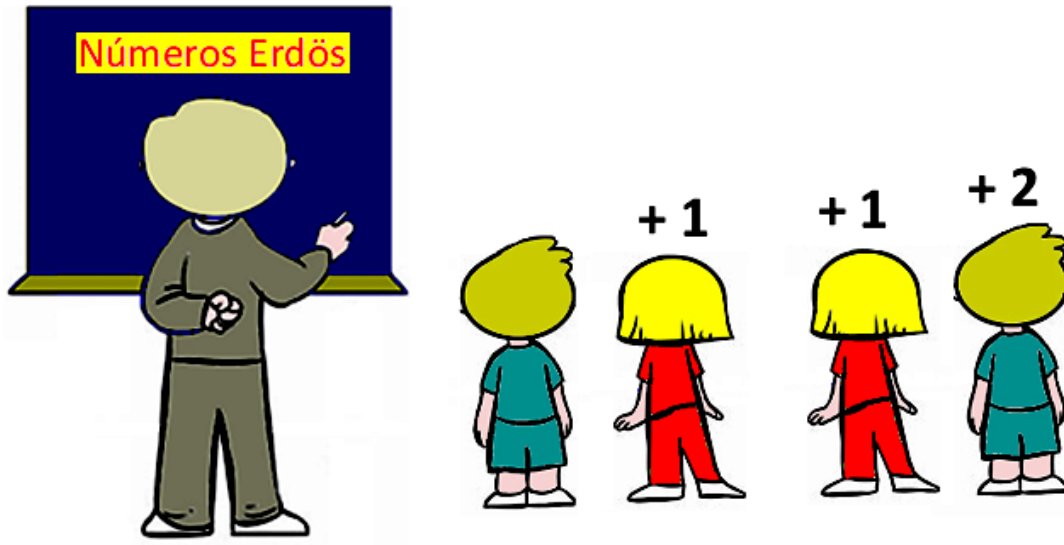


Figura 2.6: El número de Erdős como ejemplo de distancias en redes de coautoría. Fuente: [29, 4].

Definición y propiedades

El modelo de Erdős–Rényi $G(N, p)$ genera grafos aleatorios con N nodos, donde cada par de vértices se conecta de manera independiente con probabilidad p [12]. Propiedades clave [4]:

- Grado medio: $\langle k \rangle = p(N - 1)$
- Número esperado de aristas: $\mathbb{E}[|E|] = p \binom{N}{2}$
- Distribución de grado: aproximadamente binomial, que se aproxima a Poisson para N grande y p pequeño [30]:

$$P(k) \approx e^{-\langle k \rangle} \frac{\langle k \rangle^k}{k!}$$

- No hay correlación ni agrupamiento local ($C \approx p$) [2].

Transición de fase y componente gigante

A medida que aumenta p , el grafo pasa de muchas pequeñas componentes a una **componente gigante** que contiene una fracción significativa de los nodos [12, 29]:

$$\text{Umbral crítico: } \langle k \rangle_c = 1$$

- $\langle k \rangle < 1$: solo hay grupos aislados.
- $\langle k \rangle > 1$: surge la componente gigante [4].

Analogía: como una reacción en cadena: al superar un nivel crítico de enlaces, la red “enciende” su conectividad global.

2.4.3. Modelo de configuración

El **modelo de configuración** genera grafos aleatorios con una distribución de grados arbitraria $P(k)$ [29, 26]. A cada nodo i se le asigna un número de *stubs* igual a su grado k_i ; luego, los stubs se emparejan aleatoriamente hasta formar $M = \frac{1}{2} \sum_i k_i$ aristas [30].

Ventaja: reproduce la heterogeneidad de los grados reales, a diferencia del modelo de Erdős–Rényi [14].

- Grado medio: $\langle k \rangle = \sum_k kP(k)$
- Varianza del grado: $\langle k^2 \rangle - \langle k \rangle^2$
- Clustering bajo, ya que las conexiones siguen siendo aleatorias [2].

El modelo conserva exactamente los grados individuales k_i , pero destruye cualquier estructura de comunidad o correlación entre vecinos [26]. Cuando $P(k)$ sigue una ley de potencias, el grafo generado es **libre de escala** [5].

Un comentario importante: la probabilidad de que un nodo pertenezca a la componente gigante depende de los momentos de la distribución de grados [29]:

$$\langle k^2 \rangle - 2\langle k \rangle > 0 \quad \Rightarrow \quad \text{surge una componente gigante.}$$

Por debajo de este umbral, la red permanece fragmentada. Este criterio generaliza el umbral de Erdős–Rényi ($\langle k \rangle = 1$) y puede interpretarse como la manifestación práctica de la *transición de fase* [2]: a partir de cierta conectividad, una gran fracción de nodos queda interconectada.

2.5. Topologías y modelos de redes

2.5.1. Redes Small World

Las **redes de mundo pequeño** combinan dos propiedades que a primera vista parecen contradictorias:

- **Alta agrupación local:** los vecinos de un nodo suelen estar conectados entre sí, formando triángulos [43].
- **Baja distancia media global:** cualquier nodo puede alcanzarse desde cualquier otro en pocos pasos [11].

Este tipo de estructura explica fenómenos observados en redes sociales, biológicas o tecnológicas, donde “todos estamos conectados por solo unos pocos saltos” [29, 4].

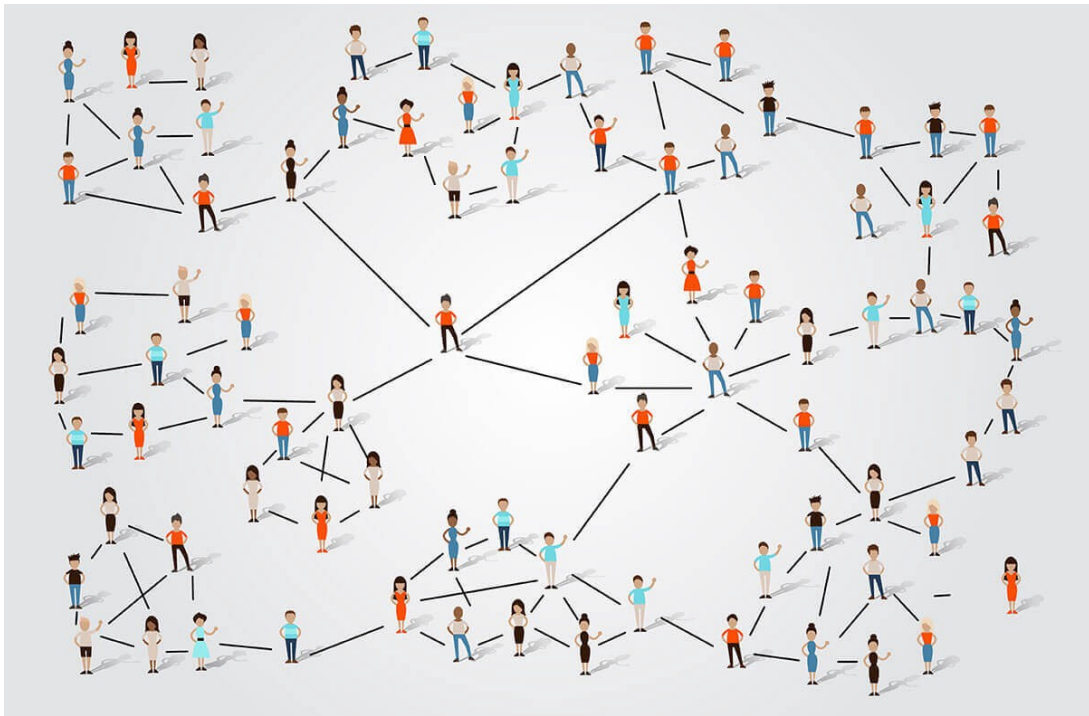


Figura 2.7: Modelo Watts-Strogatz: transición de red regular a aleatoria. Fuente: [43].

Modelo de Watts–Strogatz

El **modelo de Watts–Strogatz (WS)** formaliza estas ideas [43]. Se parte de una red regular, donde cada nodo está conectado a sus k vecinos más próximos, y luego se **reconecta aleatoriamente** cada arista con probabilidad p :

$$G(N, k, p)$$

- $p = 0$: red completamente regular (alta agrupación C , grandes distancias L).

- $p = 1$: red completamente aleatoria (bajo clustering, corta distancia media) [12].
- $0 < p \ll 1$: régimen de **mundo pequeño** — se mantienen triángulos locales y se reducen drásticamente las distancias globales [43].

Fenómeno clave: incluso unos pocos enlaces aleatorios reducen rápidamente la distancia media $L(p)$, mientras el coeficiente de clustering $C(p)$ apenas disminuye [30]:

$$L(p) \downarrow \text{rápidamente}, \quad C(p) \approx \text{constante}.$$

$$\begin{aligned} C(p) &\gg C_{\text{ER}} \approx p, & (\text{alta densidad de triángulos}) \\ L(p) &\approx L_{\text{ER}} \sim \log N, & (\text{distancias cortas típicas de redes aleatorias}) \end{aligned}$$

Ejemplo intuitivo: Tus amigos cercanos están fuertemente conectados entre sí (alto C), pero unos pocos “atajos” permiten llegar al resto del mundo en pocos pasos, mostrando el efecto de los *seis grados de separación* [29, 4].

Mnemotecnia: *Mundo pequeño = triángulos locales + atajos globales.*

Transición regular–aleatoria

El parámetro p actúa como un **control continuo de topología**, y la red evoluciona gradualmente de regular a aleatoria [28]:

$$p : 0 \longrightarrow 1$$

- $p \approx 0$: red regular, alta agrupación, grandes distancias.
- $0 < p \ll 1$: **mundo pequeño** — equilibrio entre cohesión local y conectividad global.
- $p \approx 1$: red casi aleatoria — bajo clustering, distancias cortas.

Esta transición ilustra cómo unos pocos enlaces aleatorios (“atajos”) pueden reducir drásticamente la distancia media sin destruir la cohesión local, mostrando la esencia de las redes Small World [43, 14].

2.5.2. Redes libres de escala

Hasta ahora hemos visto modelos como Erdős–Rényi (aleatorio) y Watts–Strogatz (Small World). Sin embargo, muchas redes reales presentan una heterogeneidad extrema en los grados de sus nodos: unos pocos **hubs** dominan la estructura [5, 2]. Este fenómeno se describe mediante las **redes libres de escala**, cuya distribución de grados sigue una ley de potencias [9]:

$$P(k) \sim k^{-\gamma}, \quad 2 < \gamma < 3$$

En ellas coexisten muchos nodos con pocos enlaces y unos pocos nodos extremadamente conectados (**hubs**) [4]. Se observan en Internet, metabolismo celular, aeropuertos y redes sociales [29, 6].

Consecuencias de la ley de potencias:

- Alta heterogeneidad de grados [2].
- Distancia media muy baja: $L \sim \log \log N$.
- Robustez frente a fallos aleatorios, pero vulnerabilidad ante ataques dirigidos a hubs [2].

Modelo de Barabási–Albert (Preferential Attachment)

Para explicar la aparición de esta ley de potencias, Barabási y Albert (1999) propusieron un mecanismo de **crecimiento con preferencia adjunta** [5]:

1. Comenzar con un pequeño grafo inicial de m_0 nodos.
2. En cada paso, añadir un nuevo nodo con $m \leq m_0$ enlaces.
3. La probabilidad de conectarse a un nodo existente i es proporcional a su grado [2]:

$$\Pi(k_i) = \frac{k_i}{\sum_j k_j}$$

Los nodos más conectados atraen más enlaces: el clásico efecto “**el rico se hace más rico**” [4].

Modelo de Price

El modelo de **Price** (1976) anticipa la idea de Barabási–Albert aplicándola a redes de citas científicas [29, 14]. Cada nuevo artículo cita trabajos anteriores con probabilidad proporcional al éxito previo:

$$\Pi(k_i) = \frac{k_i + k_0}{\sum_j (k_j + k_0)}$$

donde k_0 es una “atracción inicial”. Este mecanismo también produce una distribución de ley de potencias [30]. Price fue el pionero en detectar este principio décadas antes de Barabási–Albert.

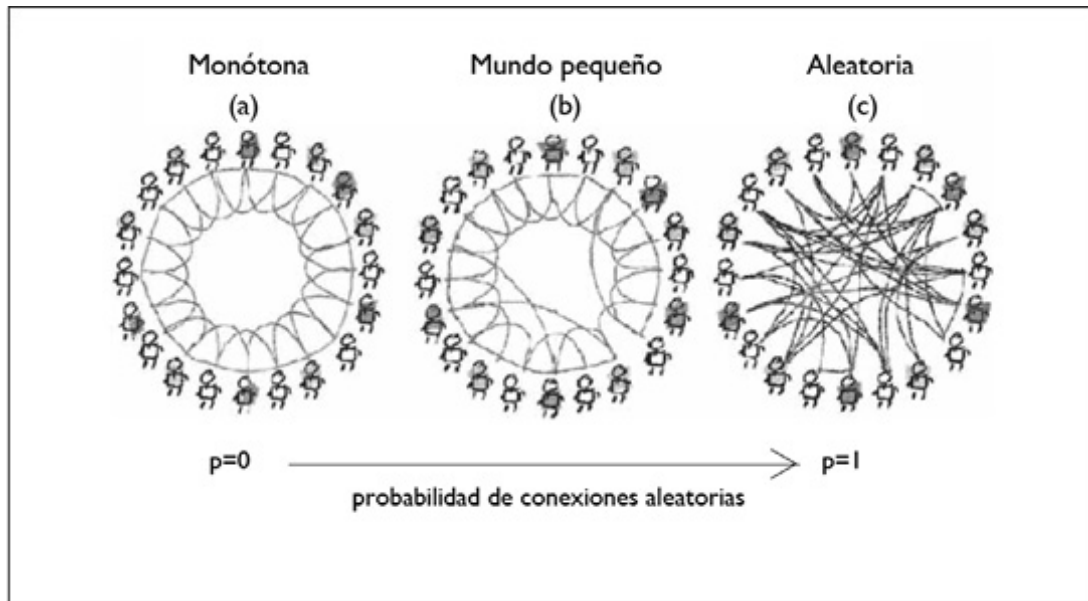


Figura 2.8: Comparativa estructural entre redes monótonas, de mundo pequeño y aleatorias. Fuente: [28, 43].

Tipo de red	Agrupación	Distancia media	Distribución de grados	Referencia
Regular	Alta	Alta	Constante	[43]
Aleatoria	Baja	Baja	Binomial / Poisson	[12]
Mundo pequeño	Alta	Baja	Intermedia	[43]
Libre de escala	Variable	Muy baja	Ley de potencias	[5]

Cuadro 2.2: Comparación de topologías de red basada en la literatura clásica.

2.6. Análisis de comunidades

Las **comunidades** son grupos de nodos dentro de una red que están más densamente conectados entre sí que con el resto del grafo [17]. Detectarlas permite entender la estructura modular de sistemas complejos —como grupos sociales, funciones biológicas o temas en la web [16].

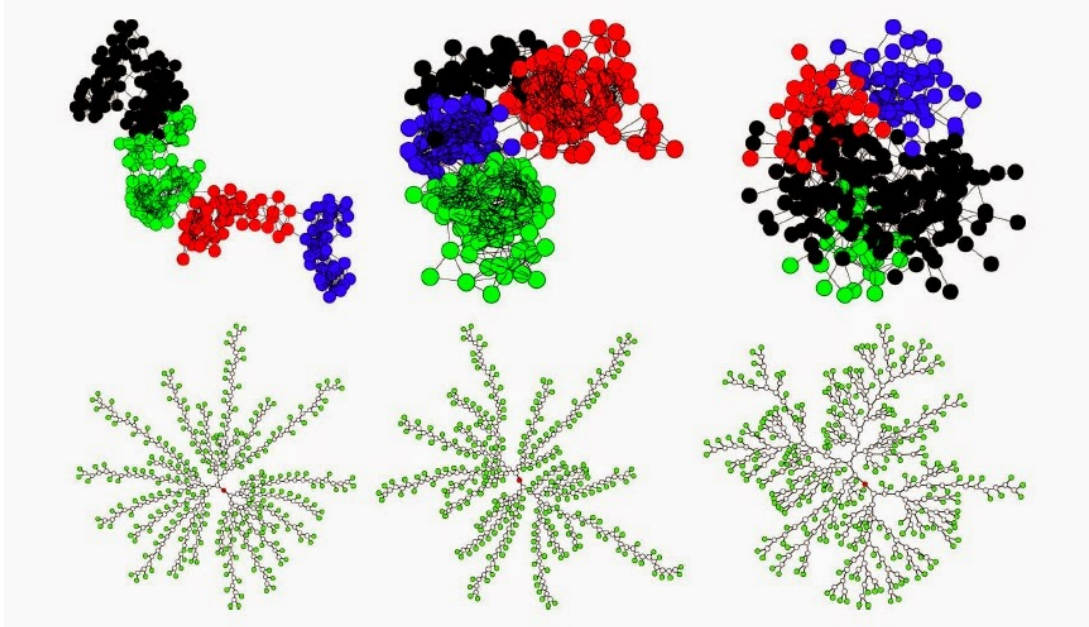


Figura 2.9: Algoritmos de detección de comunidades en redes complejas. Fuente: [30, 17].

2.6.1. Conceptos básicos: cluster, comunidad y módulo

En análisis de redes conviene distinguir entre **cluster** y **comunidad** [26]:

Cluster (agrupamiento local) Un *cluster* es un conjunto de nodos vecinos que tienden a estar conectados entre sí, medido localmente por el **coeficiente de clustering** C_v [43]. Se centra en la densidad de triángulos alrededor de cada nodo.

Comunidad Una *comunidad* es un subconjunto de nodos con muchas conexiones internas y relativamente pocas externas [17]. Se trata de una propiedad global de la red y se cuantifica mediante la **modularidad** Q introducida por Newman y Girvan [29, 18]:

$$Q = \frac{1}{2m} \sum_{i,j} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j)$$

don de:

- A_{ij} : elementos de la matriz de adyacencia,
- k_i, k_j : grados de los nodos i y j ,

- m : número total de aristas,
- $\delta(c_i, c_j) = 1$ si los nodos i y j pertenecen a la misma comunidad.

Un valor alto de Q (cercano a 1) indica que la partición en comunidades está bien definida, es decir, los nodos tienden a conectarse más dentro de su comunidad que con nodos externos [16].

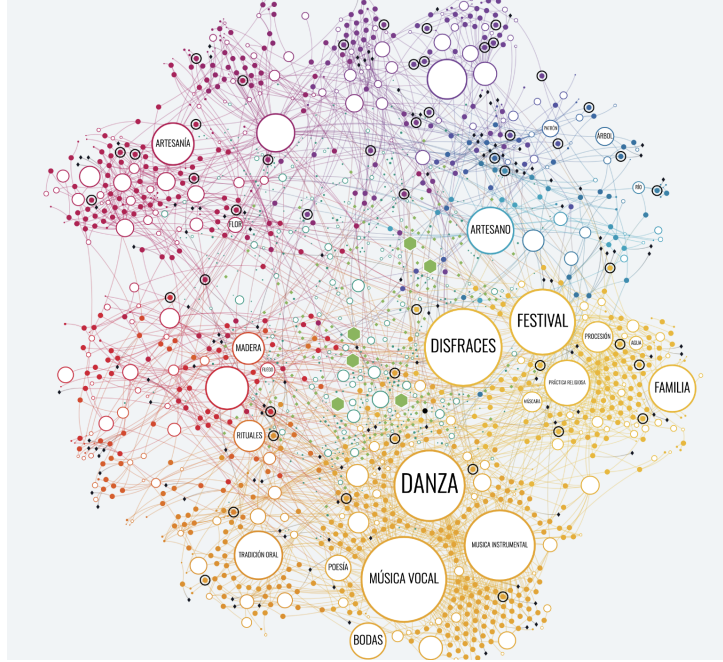


Figura 2.10: Red con comunidades, módulos y motifs [28]

En la Figura se observan **comunidades** (clusters) diferenciadas por color: por ejemplo, la comunidad amarilla agrupa conceptos como *Danza*, *Música vocal* y *Bodas*, mientras que la roja incluye *Artesanía*, *Madera* y *Rituales*.

Dentro de cada comunidad existen **módulos**, subgrupos con alta cohesión interna [26]. Finalmente, se identifican **motifs**, patrones recurrentes como triángulos y estructuras tipo estrella en nodos centrales que actúan como hubs [6].

Clustering espectral

El **clustering espectral** es un método para detectar comunidades basado en las propiedades de la *matriz Laplaciana* del grafo [25]:

$$L = D - A,$$

donde A es la matriz de adyacencia y D la matriz diagonal de grados.

Idea central El espectro de L (sus autovalores y autovectores) refleja la conectividad de la red [14]. En particular, los **autovectores asociados a los menores autovalores** capturan la estructura de comunidades [25, 26].

Procedimiento típico 1. Calcular los k autovectores correspondientes a los menores k autovalores de L . 2. Construir una matriz $U \in \mathbb{R}^{N \times k}$ cuyos renglones representan a los nodos en un espacio k -dimensional. 3. Aplicar un algoritmo de clustering clásico, como k -means, sobre los renglones de U . 4. Asignar cada nodo a la comunidad resultante [13].

Método de Girvan–Newman

El **método de Girvan–Newman** detecta comunidades basándose en la idea de que los enlaces que conectan diferentes comunidades son “puentes” con alta *centralidad de intermediación* (*betweenness*) [17].

Procedimiento 1. Calcular la centralidad de intermediación de todas las aristas [30]. 2. Eliminar la arista con mayor centralidad. 3. Recalcular las centralidades y repetir el proceso [16]. 4. La red se fragmenta en subgrafos que corresponden a comunidades naturales.

Mnemotecnia *Girvan–Newman = cortar puentes; Clustering espectral = usar autovectores para separar comunidades.*

2.6.2. Comparación de modelos

Modelo	Tipo de red	Clustering	Distancia media	Referencia
Erdős–Rényi	Aleatoria	Bajo	Bajo	[12]
Watts–Strogatz	Mundo pequeño	Alto	Bajo	[43]
Barabási–Albert	Libre de escala	Variable	Muy bajo	[5]
Regular	Ordenada	Alto	Alto	[26]

*El análisis de las métricas de centralidad, los modelos de grafos aleatorios y la detección de comunidades nos ha proporcionado una radiografía precisa de la estructura y las tipologías de las redes. Sin embargo, en el ámbito de la inteligencia computacional aplicada, el objetivo no es solo medir la red como un objeto inerte, sino utilizar su conectividad para predecir el comportamiento humano y optimizar el acceso a la información. En el próximo bloque, exploraremos cómo estas propiedades estructurales se traducen en **Sistemas Recomendadores**, transformando los lazos de confianza en motores de predicción que definen la economía de la atención en la era digital.*

Capítulo 3

Sistemas Recomendadores y Modelos de Confianza

3.1. Introducción a los Sistemas Recomendadores (SR)

La explosión de contenido disponible en la era digital ha hecho que la **información sea abundante, pero la atención humana escasa**. Esta paradoja ha generado el problema de la **sobrecarga de información** (*information overload*), donde los usuarios son incapaces de procesar la vasta cantidad de datos, productos o noticias que se presentan ante ellos, lo que dificulta la toma de decisiones [37, 1].

3.1.1. Breve Perspectiva Histórica: Groupware y Workflow

Aunque los Sistemas Recomendadores (SR) emergieron formalmente en la década de 1990, sus raíces conceptuales se extienden a campos anteriores que se enfocaban en la gestión y el filtrado de información en entornos colaborativos [37]. Dos de estos campos son el **Groupware** y el **Workflow** [39].

Aunque Groupware y Workflow se centraron en la **productividad** empresarial, su manejo de la complejidad informativa y su enfoque en la **interacción social** (Groupware) y la **secuencia de eventos** (Workflow) proporcionaron la base arquitectónica y los primeros retos de filtrado que más tarde serían formalizados y resueltos por los **Sistemas de Filtrado de Información** y, finalmente, por los **Sistemas Recomendadores** basados en técnicas de Machine Learning [7, 20]:

Groupware (Software Colaborativo)

El *Groupware* (o software colaborativo) se desarrolló en los años 80 y principios de los 90 con el objetivo de facilitar el trabajo de grupos de personas que comparten una tarea o meta común [39]. El foco era principalmente la coordinación, comunicación y colaboración entre usuarios.

Sin embargo, estos sistemas se enfrentaron tempranamente al desafío de la **gestión de la información compartida**. Al manejar grandes volúmenes de documentos, mensajes

y decisiones, los sistemas de Groupware desarrollaron primitivas formas de filtrado. Por ejemplo, al categorizar documentos por relevancia o al priorizar mensajes, sentaron las bases para que los SR modernos identifiquen y jerarquicen la información en función del contexto social y de la tarea [37].

Workflow (Gestión de Flujos de Trabajo)

Los sistemas de *Workflow* (Gestión de Flujos de Trabajo) se enfocaron en la automatización y la estructuración de procesos de negocio repetitivos [39]. En este caso, se centraba principalmente en la regulación de secuencias de tareas, asegurando que la información o un ítem de trabajo pase por las manos adecuadas en el orden correcto.

En este aspecto, los sistemas de Workflow son precursores conceptuales de los SR en dos aspectos:

- (a) **Modelado de Procesos:** Demostraron cómo modelar el **flujo de la información** y la **dependencia** de un ítem respecto a las acciones de diferentes agentes.
- (b) **Sugerencia Basada en Contexto:** Desarrollaron la capacidad de *sugerir* el siguiente paso o al siguiente actor en un proceso, lo cual es análogo a la función de un SR que sugiere el siguiente ítem o acción más probable [1].

3.1.2. Definición y Naturaleza del SR

Bajo este legado, los **Sistemas de Recomendación** se definen como mediadores inteligentes que emplean técnicas de **filtrado de información** para presentar ítems personalizados a un usuario específico [38]. Su mecanismo fundamental consiste en contrastar el **perfil del usuario** con las **características de los ítems**, buscando predecir la preferencia sobre elementos aún no consumidos. En la actual **Economía de la Atención**, estos sistemas no solo facilitan el descubrimiento (*serendipity*), sino que maximizan el *engagement* y el valor comercial de las plataformas [11, 23].

El SR actúa como un mediador inteligente, ayudando al usuario a navegar por un espacio de ítems potencialmente infinito [11].

El objetivo principal de un SR es:

- **Predicción de Preferencia:** Buscar predecir el *ranking* o ponderación que el usuario le daría a un ítem que el sistema aún no ha considerado [23].
- **Generación de Características:** Las características utilizadas para esta comparación pueden basarse en la relación o acercamiento del usuario con el tema, o en el ambiente social del mismo usuario [20].

El resultado final de este proceso es una lista de ítems ordenados que se espera maximicen la probabilidad de interacción, compra o consumo por parte del usuario [38].

3.1.3. El SR como Pilar de la Economía de la Atención

La relevancia de los SR trasciende la mera conveniencia para el usuario; se han convertido en pilares del modelo de negocio en la economía digital [11]. En la **Economía de la Atención**, donde el recurso más valioso es el tiempo del usuario, el SR es la herramienta esencial para:

- **Aumentar el Engagement:** Plataformas de contenido utilizan los SR para aumentar el tiempo de permanencia del usuario (*engagement*) al sugerir la próxima película o canción [23].
- **Maximizar el Valor de por Vida del Cliente (CLV):** En el e-commerce, los SR mejoran las ventas al sugerir productos relevantes (*cross-selling*) [38].
- **Descubrimiento (*Serendipity*):** Facilitar que los usuarios encuentren contenido o productos inesperados pero relevantes [7].

3.1.4. Problemas Fundamentales de los Sistemas Recomendadores

A pesar de su eficacia, todos los Sistemas Recomendadores enfrentan desafíos inherentes que los distintos modelos (Colaborativos, Basados en Contenidos, Híbridos) buscan mitigar [1, 20]:

- **Problema del Arranque en Frío (*Cold Start Problem*):** Se presenta cuando un **nuevo usuario** o un **nuevo ítem** se introduce en el sistema. El SR no tiene suficientes datos de interacción para generar un perfil fiable o para hacer comparaciones [7].
- **Problema de la Escasez de Datos (*Sparsity*):** Incluso en sistemas grandes, la matriz de interacciones usuario-ítem suele ser muy escasa (*sparse*), ya que la mayoría de los usuarios solo han interactuado con una pequeña fracción de todos los ítems disponibles [38, 23]. Esto dificulta encontrar patrones de similitud fiables.

3.2. Estructura del Conocimiento: Taxonomía vs. Folksonomía

Para que un motor de recomendación funcione, el sistema debe "entender" qué está recomendando. La organización del conocimiento determina cómo se indexan los ítems, influyendo directamente en el diseño de los algoritmos [1].

Taxonomía: El Orden Centralizado Representa la clasificación jerárquica y rígida definida por expertos. Aunque ofrece alta consistencia y precisión, su falta de escalabilidad ante temas emergentes limita la frescura de las recomendaciones en entornos volátiles [11].

- **Definición:** Es un esquema de clasificación formal y rígido donde los ítems se agrupan en categorías y subcategorías que siguen una estructura de árbol (*parent-child relation*) [37]. Las etiquetas son creadas por un equipo editorial, expertos temáticos o mediante estándares industriales.
- **Ventajas:**
 - (a) **Consistencia:** Garantiza que un ítem siempre se etiquetará de la misma forma, ofreciendo alta precisión y uniformidad.
 - (b) **Precisión:** Al ser definida por expertos, la estructura lógica es robusta y precisa.
 - (c) **Integración:** Fácil de integrar con bases de datos relacionales y sistemas de búsqueda tradicionales [38].
- **Desventajas e Impacto en SR:**
 - (a) **Falta de Escalabilidad:** La estructura es costosa y lenta de mantener. Añadir nuevas categorías es un proceso manual.
 - (b) **Rigidez:** Dificultad para capturar matices o temas emergentes, lo que limita la capacidad del SR para identificar características subjetivas [1].

Folksonomía: El Orden Emergente Basada en el etiquetado colaborativo (*tags*), permite que la estructura del conocimiento evolucione orgánicamente con la comunidad. Esta riqueza semántica es vital para que los SR capturen intereses de nicho, aunque introduce ruido y ambigüedad que deben ser procesados algorítmicamente [20, 7].

- **Definición:** Es un sistema de clasificación abierto y distribuido donde los ítems son etiquetados libremente por la comunidad (*crowd-sourcing*) [20]. Las etiquetas se conocen como **tags** o **etiquetas sociales**. Plataformas como Flickr o YouTube son ejemplos clásicos [1].
- **Ventajas:**
 - (a) **Escalabilidad:** La estructura se adapta rápidamente a la cantidad masiva de datos no estructurados [11].
 - (b) **Riqueza Semántica:** Refleja el lenguaje natural y los **temas emergentes**, facilitando a los SR capturar características de nicho [7].
 - (c) **Soporte para CF:** Las etiquetas proporcionan datos adicionales que los SR Colaborativos utilizan para mitigar la escasez de datos [38, 23].
- **Desventajas e Impacto en SR:**
 - (a) **Ambigüedad y Ruido:** El ruido es inherente debido a la falta de control (sinónimos, errores ortográficos) [20].

(b) **Vaguedad:** Dificultad para la búsqueda precisa por la falta de jerarquía.

Cuadro 3.1: Comparativa de Tipologías de Sistemas Recomendadores

Tipología	Definición	Ventajas	Desventajas
Colaborativo	Basado en interacciones usuario-ítem para encontrar similitudes [38].	• Alta capacidad para la <i>Serendipity</i>. • No requiere metadatos [23].	• Arranque en Frío [1]. • Escasez de Datos [38].
Contenidos	Recomienda ítems similares a los consumidos, basado en atributos [7].	• Robusto ante el <i>Cold Start</i> de usuarios. • Independencia de otros usuarios [1].	• Sobre-especialización [20]. • Depende de calidad de metadatos.
Híbridos	Combina CF + CB para mitigar debilidades [7].	• Mitiga <i>Cold Start</i> y <i>Sparsity</i>. • Mejora la precisión [1].	• Mayor complejidad. • Costo computacional alto [20].

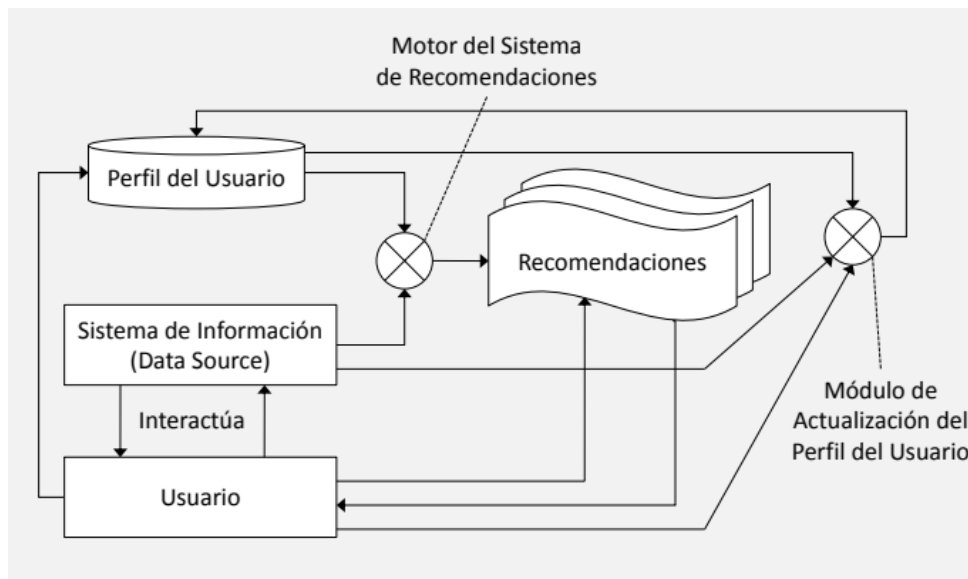


Figura 3.1: Arquitectura general de los Sistemas de Recomendación. Fuente: [20].

3.3. Arquitectura General de un Sistema Recomendador

Habiendo definido cómo se organiza el conocimiento, es necesario analizar la "tubería" (*pipeline*) que transforma datos brutos en sugerencias. Esta arquitectura se apoya en

tres pilares funcionales interconectados: un motor de comparación, un motor de predicción/aprendizaje y un módulo de evaluación. El objetivo de esta estructura es traducir los datos brutos de interacción y preferencias en recomendaciones accionables.

3.3.1. Mecanismos de Similitud

Los mecanismos de similitud son esenciales en los SR, particularmente en el **Filtrado Colaborativo (CF)**, ya que permiten cuantificar la **proximidad** entre dos usuarios (para encontrar "vecinos con gustos similares") o entre dos ítems (para encontrar ítems que son sustitutos cercanos) [38]. Estos mecanismos transforman las preferencias de los usuarios (a menudo representadas como vectores en un espacio multidimensional) en un único valor escalar [1].

- **Similaridad del Coseno (*Cosine Similarity*):** Mide el **ángulo** entre dos vectores [38]. Es ideal cuando la magnitud de las calificaciones no es tan importante como su dirección (es decir, el patrón de calificaciones). Es robusta frente a las diferencias en la escala de las calificaciones entre usuarios, ya que solo considera la orientación [20]. Se calcula como el producto punto de los vectores dividido por el producto de sus magnitudes:

$$\text{sim}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}$$

- **Correlación de Pearson (*Pearson Correlation*):** Mide el **grado de relación lineal** entre dos variables (usuarios o ítems), ajustando las diferencias en la media de las calificaciones de cada entidad [1]. Es útil cuando se quiere compensar el sesgo del usuario (si un usuario tiende a calificar siempre alto y otro siempre bajo) [37]. Se calcula como la covarianza de los vectores dividida por el producto de sus desviaciones estándar.
- **Distancia Euclidiana (*Euclidean Distance*):** Mide la **distancia geométrica directa** entre dos puntos en un espacio multidimensional. A diferencia de la similitud del coseno, sí está influenciada por la magnitud de las calificaciones [20]. En SR, se suele transformar en una métrica de similitud tomando su inverso o restándola de un valor máximo, ya que una distancia pequeña implica alta similitud.

3.3.2. Mecanismos de Retroalimentación: Implícita vs. Explícita

La base de cualquier Sistema Recomendador es el conocimiento de las **preferencias del usuario** [37]. Esta información se obtiene a través de **Mecanismos de Retroalimentación** que capturan la interacción del usuario con los ítems. Existen dos categorías principales de retroalimentación, que difieren en la intencionalidad del usuario y en la calidad del dato recolectado [1, 20]. La fuente primaria son los datos de interacción. La **retroalimentación explícita** (ratings, likes) ofrece precisión pero sufre de escasez (*sparsity*), mientras que la **implícita** (clics, tiempo de permanencia) es abundante pero ambigua [23, 20].

Retroalimentación Explícita

La Retroalimentación Explícita (*Explicit Feedback*) ocurre cuando el usuario **proporciona directamente** su opinión sobre un ítem con la intención de informar al sistema [23].

- **Definición:** Datos directamente expresados por el usuario. Son medidas de preferencia conscientes y objetivas.
- **Ejemplos:**
 - (a) **Calificaciones (*Ratings*):** Otorgar estrellas (ej., 1 a 5) o una puntuación numérica a un producto o contenido [38].
 - (b) **Reseñas (*Reviews*):** Comentarios de texto que acompañan una calificación, aportando contexto cualitativo.
 - (c) **Me Gusta/No Me Gusta (*Likes/Dislikes*):** Juicios binarios sobre un ítem.
- **Ventajas:** Los datos son **precisos** y reflejan claramente la intensidad de la preferencia. Esto facilita el cálculo de la precisión del modelo (RMSE, MAE) [23].
- **Desventajas:** Los datos son **escasos** (*sparse*), ya que la mayoría de los usuarios solo califica una pequeña fracción de ítems [38]. Además, pueden estar sujetos a **sesgo** (ej., el usuario solo califica lo que ama o lo que odia).

Retroalimentación Implícita

La Retroalimentación Implícita (*Implicit Feedback*) infiere las preferencias del usuario a partir de su **comportamiento o interacciones** observadas dentro del sistema [20].

- **Definición:** Datos recogidos de forma pasiva por el sistema sin una acción directa de calificación por parte del usuario [23]. Estos datos sugieren interés, pero no la satisfacción.
- **Ejemplos:**
 - (a) **Vistas y Clics:** El usuario hizo clic en un ítem.
 - (b) **Tiempo de Permanencia:** La duración que el usuario pasó viendo un vídeo o leyendo un artículo.
 - (c) **Historial de Compras:** La adquisición de un producto es una fuerte señal de preferencia [38].
 - (d) **Navegación:** Artículos añadidos al carrito o a la lista de deseos.
- **Ventajas:** Los datos son **abundantes**, ya que la mayoría de las interacciones en una plataforma son implícitas. Esto ayuda a combatir el problema de la escasez de datos (*sparsity*) [1].
- **Desventajas:** Es **ambigua**. El sistema sabe lo que el usuario hizo (ej., vio una película), pero no si le gustó. Un clic no equivale a una preferencia positiva [23].

El Desafío de la Preferencia Negativa

Un reto crítico en la retroalimentación implícita es determinar la **Preferencia Negativa** [20]. Si un usuario ha interactuado con el ítem A pero no con el ítem B , el sistema puede inferir que B no es de su interés. Sin embargo, hay dos posibles interpretaciones para la ausencia de interacción [1]:

1. **Desconocimiento:** El usuario simplemente **nunca vio** el ítem B porque no fue expuesto a él (la ausencia no es un voto negativo).
2. **Rechazo:** El usuario fue expuesto al ítem B y **decidió activamente ignorarlo** (la ausencia es un voto negativo).

Los modelos de SR avanzados deben utilizar técnicas de *sampling* y modelos de *ranking* para diferenciar entre estas dos ausencias y mitigar el sesgo hacia ítems ya vistos [23].

3.3.3. Recolección y Preprocesamiento de Datos

Antes de aplicar cualquier algoritmo de recomendación, es fundamental establecer un proceso robusto para obtener los datos que definen las preferencias del usuario y las características del ítem [1].

Fuentes de Datos y Características

La recolección de datos alimenta la matriz de interacción usuario-ítem, que es la base de los Sistemas Recomendadores (SR) [38]. Las principales fuentes son:

- **Datos de Interacción:** Recolectados mediante **Mecanismos de Retroalimentación** (discutidos anteriormente). Estos definen la matriz de interacciones [20].
- **Datos de Contenido:** Atributos de los ítems (metadatos, género, director, keywords). Estos son vitales para los SR Basados en Contenidos y para mitigar el *cold start* [7].
- **Datos Demográficos/Contextuales:** Características del usuario (edad, ubicación, historial de búsqueda). Útiles para personalizar recomendaciones y para el *cold start* de nuevos usuarios [1].

Preprocesamiento y Normalización

El preprocesamiento es crucial para lidiar con la naturaleza ruidosa y escasa de los datos de los SR [38]:

- **Normalización de Calificaciones:** En la retroalimentación explícita, es común normalizar las calificaciones para compensar el **sesgo del usuario** (ej., un usuario que siempre califica alto). Esto se logra restando la media de las calificaciones del usuario a cada una de sus calificaciones [37].

- **Manejo de la Escasez (*Sparsity*):** La matriz de interacción suele tener muchos ceros [38]. Se emplean técnicas como la **Factorización Matricial** para inferir los valores faltantes en el espacio latente o **muestreo negativo** en la retroalimentación implícita [23].

3.3.4. Técnicas Predictoras Avanzadas

Dentro de las Técnicas Predictoras (Sección 3.2), existen modelos específicos que ofrecen enfoques eficientes o más robustos que la simple suma ponderada [1].

Regresión como Predictor

La Regresión puede ser utilizada tanto en modelos Basados en Contenidos como en modelos Híbridos/Colaborativos para estimar el *rating* desconocido $\hat{r}_{u,i}$ [7, 1].

- **Uso en CB:** El perfil del usuario (vector de preferencias) y el vector de características del ítem son utilizados como **variables independientes** (X). El modelo de regresión (ej., Regresión Lineal o SVR) se entrena con los *ratings* conocidos (Y) para aprender la función que mapea las características a la calificación [20].
- **Uso en CF (Modelos de Sesgo):** La regresión se utiliza para modelar los **sesgos** intrínsecos de los usuarios y los ítems [23]. La predicción se calcula como la suma de la media global (μ), el sesgo del usuario (b_u) y el sesgo del ítem (b_i), más un error:

$$\hat{r}_{u,i} = \mu + b_u + b_i + \epsilon_{u,i}$$

Los parámetros de sesgo (b_u y b_i) se aprenden minimizando la función de error cuadrático [23].

Predictor Slope One

Slope One es un algoritmo de Filtrado Colaborativo **basado en ítems** que destaca por su **simplicidad, facilidad de implementación y eficiencia computacional** [20].

- **Principio:** Asume que la diferencia en las calificaciones que un usuario da a dos ítems (i y j) es una constante para todos los usuarios.
- **Cálculo de la Diferencia (Slope):** El valor central del algoritmo es la **desviación promedio** ($dev_{i,j}$) entre los *ratings* de los ítems i y j :

$$dev_{i,j} = \frac{\sum_{u \in U_{i,j}} (r_{u,i} - r_{u,j})}{|U_{i,j}|}$$

Donde $U_{i,j}$ es el conjunto de usuarios que han calificado **ambos** ítems, i y j .

- **Cálculo de la Predicción:** La predicción para el ítem i para el usuario u se obtiene promediando la calificación conocida del ítem j con la desviación promedio entre i y j :

$$\hat{r}_{u,i} = \frac{\sum_{j \in R_u} (dev_{i,j} + r_{u,j}) \cdot w_{i,j}}{\sum_{j \in R_u} w_{i,j}}$$

Donde R_u es el conjunto de ítems que el usuario u ya ha calificado, y $w_{i,j}$ es un peso (generalmente la cardinalidad $|U_{i,j}|$) que representa cuántos usuarios han calificado el par $[20]$.

- **Ventajas:** Es incremental (fácilmente actualizable), rápido de entrenar y predecir, y es una excelente opción cuando la escalabilidad es una preocupación.

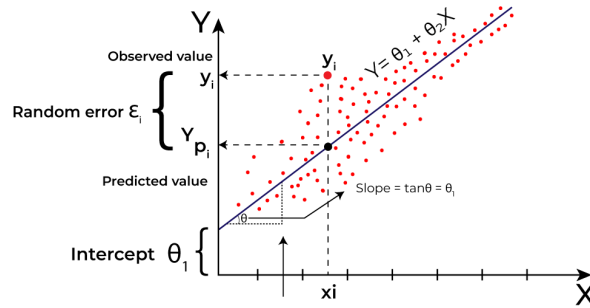


Figura 3.2: Esquema del predictor clásico Slope One para filtrado colaborativo. Fuente: [20, 38].

3.3.5. Técnicas Predictoras

A pesar de la sofisticación de modelos como la Descomposición en Valores Singulares (SVD) o el predictor Slope One, estas técnicas operan bajo una premisa puramente estadística: asumen que toda interacción en la base de datos es legítima y fidedigna. Las técnicas predictoras son el núcleo del SR. Una vez que se ha establecido la similitud, estos modelos utilizan los datos históricos y las relaciones descubiertas para **estimar la calificación** ($\hat{r}_{u,i}$) que un usuario u daría a un ítem i que aún no ha visto [1, 38].

- **Métodos Basados en Vecinos (*Neighborhood-based Methods*):** Característicos del Filtrado Colaborativo. La predicción se calcula como una media ponderada de las calificaciones que los **usuarios similares** (*user-based CF*) o las calificaciones dadas por el usuario a **ítems similares** (*item-based CF*) le han dado al ítem objetivo [37, 38].

$$\hat{r}_{u,i} = \frac{\sum_{v \in \text{vecinos}(u)} \text{sim}(u, v) \cdot r_{v,i}}{\sum_{v \in \text{vecinos}(u)} |\text{sim}(u, v)|}$$

- **Métodos Basados en Modelos (*Model-based Methods*):** Utilizan el Aprendizaje Automático para construir un modelo predictivo, siendo la **Factorización Matricial** (*Matrix Factorization*) el ejemplo más prominente [23].

Descomposición en Valores Singulares (SVD)

La **Descomposición en Valores Singulares (SVD)** es una técnica de álgebra lineal que constituye el pilar fundamental de los modelos de factorización matricial modernos. Su objetivo primordial es la reducción de la dimensionalidad de la matriz de interacciones original, permitiendo descubrir estructuras latentes (preferencias ocultas) que no son evidentes en los datos brutos.

Formalismo Matemático Dada una matriz de calificaciones R de dimensiones $m \times n$, donde m es el número de usuarios y n el de ítems, el teorema de SVD establece que R puede descomponerse en el producto de tres matrices:

$$R = U\Sigma V^T$$

Donde:

- U : Es una matriz ortogonal de $m \times m$, cuyas columnas representan a los **usuarios** en un espacio de características latentes.
- Σ : Es una matriz diagonal de $m \times n$ que contiene los **valores singulares**, los cuales cuantifican la relevancia de cada factor latente.
- V^T : Es la transpuesta de una matriz ortogonal de $n \times n$, cuyas filas representan a los **ítems** en el mismo espacio latente.

Aproximación de Bajo Rango y Predicción Debido a que la matriz R suele ser altamente dispersa (*sparse*), en este trabajo se emplea una aproximación de bajo rango mediante la selección de los k valores singulares más significativos ($k \ll n, m$). Esto permite proyectar tanto a usuarios como a ítems en un espacio vectorial compartido de dimensión k . La predicción de una calificación desconocida se calcula entonces como el producto escalar de los vectores latentes correspondientes:

$$\hat{r}_{u,i} \approx \mathbf{p}_u \cdot \mathbf{q}_i^T$$

Donde $\mathbf{p}_u$ es el vector representativo del usuario u y $\mathbf{q}_i$ es el vector representativo del ítem i .

Esta técnica ofrece tres ventajas críticas para la implementación del SR:

1. **Generalización:** Capta relaciones semánticas entre ítems (ej. si un usuario consume contenido de ciencia ficción, el sistema asocia factores latentes similares entre ítems de ese género).

2. **Escalabilidad:** Reduce drásticamente la cantidad de datos necesarios para realizar una predicción una vez entrenado el modelo.
3. **Reducción de Ruido:** Al ignorar los valores singulares menores, el modelo se centra en los patrones de comportamiento más consistentes, ignorando fluctuaciones aleatorias en las calificaciones.

3.3.6. Evaluación de Sistemas Recomendadores

La evaluación es esencial para medir la calidad de las predicciones del SR y comparar distintos enfoques [1, 20]. Las métricas se dividen según el objetivo de la predicción: estimar un valor numérico (precisión) o estimar una pertenencia binaria (clasificación).

- **Métricas de Precisión (Estimación de Error):** Se utilizan cuando el SR predice una calificación numérica (*rating*, ej., de 1 a 5 estrellas). Miden cuán cerca está la predicción ($\hat{r}_{u,i}$) de la calificación real ($r_{u,i}$) [23, 20].
 - **Métricas de Precisión (Estimación de Error):** Se utilizan cuando el SR predice una calificación numérica (*rating*). Miden cuán cerca está la predicción ($\hat{r}_{u,i}$) de la calificación real ($r_{u,i}$) [23, 20].
 - **MSE (*Mean Squared Error*):** Mide el promedio de los errores al cuadrado. Al no tener la raíz cuadrada, es útil en procesos de optimización matemática, aunque su unidad de medida no es la misma que la de las calificaciones originales.

$$\text{MSE} = \frac{1}{|\mathcal{T}|} \sum_{(u,i) \in \mathcal{T}} (r_{u,i} - \hat{r}_{u,i})^2$$

- **RMSE (*Root Mean Square Error*):** Es la raíz cuadrada del MSE. Penaliza fuertemente los errores grandes y devuelve el error a la misma escala que las calificaciones originales (ej. estrellas), lo que facilita su interpretación [38].

$$\text{RMSE} = \sqrt{\frac{1}{|\mathcal{T}|} \sum_{(u,i) \in \mathcal{T}} (r_{u,i} - \hat{r}_{u,i})^2}$$

- **MAE (*Mean Absolute Error*):** Mide el promedio de las diferencias absolutas. A diferencia del RMSE, trata todos los errores de manera lineal, siendo menos sensible a valores atípicos [1].

$$\text{MAE} = \frac{1}{|\mathcal{T}|} \sum_{(u,i) \in \mathcal{T}} |r_{u,i} - \hat{r}_{u,i}|$$

- **Métricas de Clasificación/Ranking:** Se utilizan cuando el SR predice si un usuario interactuará con un ítem o si lo consumirá. La métrica se centra en la calidad de la lista de recomendación [20].

- **Precisión ($Precision@k$):** De los k ítems recomendados, ¿cuántos son relevantes? [38]
- **Recall ($Recall@k$):** Del total de ítems relevantes para el usuario, ¿cuántos fueron incluidos en la lista de k recomendaciones? [1]
- **F1-score:** Es la media armónica entre Precisión y Recall.

Métricas de Cobertura y Diversidad: Miden la calidad estratégica de las recomendaciones, abordando el impacto en el negocio y la experiencia del usuario [20, 7].

- **Cobertura del Catálogo ($Catalog Coverage$):** Mide la proporción de ítems únicos en el catálogo que han sido alguna vez recomendados. Es vital para evitar el sesgo hacia ítems populares [20].
- **Diversidad ($Diversity$):** Mide cuán diferentes son los ítems dentro de una lista de recomendación, previniendo listas monótonas [1].
- **Serendipity:** Mide la capacidad del SR para sorprender al usuario con ítems relevantes que probablemente no habrían descubierto por sí mismos [7, 1].

3.4. Tipologías de Sistemas Recomendadores

Los Sistemas Recomendadores se clasifican principalmente en tres grandes tipologías, cada una con un enfoque distinto para modelar las preferencias del usuario y generar la lista de ítems sugeridos [1]. La elección de la tipología depende de la naturaleza de los datos disponibles (interacciones vs. atributos de ítems) y de los problemas que se buscan mitigar [7, 20].

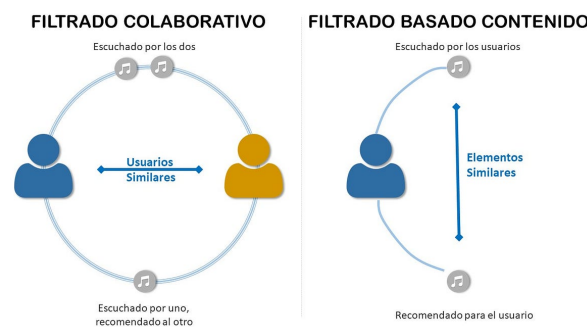


Figura 3.3: Esquema conceptual: Comparativa entre Filtrado Colaborativo y Basado en Contenido. Fuente: [1, 20].

3.4.1. Sistemas Recomendadores Colaborativos (CF)

El **Filtrado Colaborativo** (*Collaborative Filtering* o **CF**) es el enfoque más extendido y fundamental en los Sistemas Recomendadores (SR). Se basa en la premisa sociológica

de que si dos usuarios tuvieron preferencias similares en el pasado, es probable que las tengan en el futuro [37, 38].

El sistema opera basándose únicamente en las **interacciones pasadas de los usuarios** (*user-item interactions*), ignorando los atributos intrínsecos de los ítems [23].

- **Principio Central:** La predicción para un usuario u se basa en el comportamiento de otros usuarios que son **similares** a u [1].

Proceso del Filtrado Colaborativo

El proceso de CF (específicamente en los métodos basados en vecinos) sigue un flujo de cuatro pasos para generar la recomendación [38, 20]:

1. **Recolección de Datos:** Se construye la matriz dispersa de interacción usuario-ítem, utilizando retroalimentación explícita (ratings) o implícita (compras, clics).
2. **Cálculo de Similitud:** Se cuantifica la proximidad entre usuarios (CF basado en Usuario) o entre ítems (CF basado en Ítem) utilizando métricas como la Similaridad del Coseno o la Correlación de Pearson [1].
3. **Identificación de Vecinos:** Se seleccionan los K vecinos más cercanos al usuario o al ítem objetivo.
4. **Generación de Predicción:** Se calcula el *rating* desconocido $\hat{r}_{u,i}$ como la suma ponderada de las calificaciones de los vecinos.
5. **Generación de Recomendaciones:** Los ítems con la predicción más alta (que el usuario aún no ha visto) se presentan en la lista final.

Tipos de CF y Modelos

Los enfoques de CF se dividen en métodos basados en memoria (vecinos) y métodos basados en modelos [1, 38]:

- **CF basado en Usuarios (*User-based CF*):** Busca a los **vecinos** más similares al usuario activo u y recomienda los ítems que esos vecinos mejor han valorado [37]. Este enfoque es intuitivo pero sufre de problemas de escalabilidad.
- **CF basado en Ítems (*Item-based CF*):** Mide la similitud entre el ítem i y otros ítems que el usuario u ha valorado [38]. Es generalmente más robusto y escalable que el basado en usuarios.
- **Métodos basados en Modelos:** Entrenan modelos para aprender la estructura latente [23].

- (a) **Factorización Matricial (*Matrix Factorization*):** Descompone la matriz de interacción en dos matrices de rango inferior (características latentes de usuarios y ítems), permitiendo capturar relaciones que no son evidentes [23]. El **SVD** (*Singular Value Decomposition*) es un método canónico.

- **Otros Tipos de CF:**

- (c) **CF Basado en Gráfico/Gráfico de Conocimiento:** Utiliza algoritmos de recorrido de grafos para modelar las relaciones complejas entre usuarios, ítems y atributos [11, 26].
- (d) **CF Basado en Redes Neuronales (*Deep CF*):** Emplea arquitecturas de *Deep Learning* para aprender representaciones no lineales de las características latentes [20].

Ejemplos de Sistemas Recomendadores (SR)

La mayoría de los grandes SR utilizan variaciones del Filtrado Colaborativo, a menudo en un formato híbrido [7]:

- **Netflix:** Utiliza principalmente CF basado en Modelos (Factorización Matricial) y CF profundo para sugerir películas y series basadas en el historial de visualización [23].
- **Amazon:** Combina el CF basado en Ítems ("Los clientes que compraron X, también compraron Y") con reglas basadas en contenido e historial de navegación [38].
- **Spotify:** Emplea CF para sus listas de reproducción Discover Weekly y CF basado en el tiempo [1].

Ventajas y Desventajas Formales del CF

El CF presenta un *trade-off* crucial entre su capacidad predictiva y su robustez ante la falta de datos, lo que justifica la necesidad de modelos híbridos [1, 7].

3.4.2. Sistemas Recomendadores Basados en Contenidos (CB)

Los SR Basados en Contenidos (*Content-Based*) se enfocan en las propiedades de los ítems y las preferencias históricas del usuario sobre esas propiedades [1]. El sistema recomienda ítems que son **similares a los que el usuario ya ha consumido** y valorado positivamente en el pasado [20].

- **Principio:** "Si te gustó el ítem A , te gustarán otros ítems con las mismas características que A " [7].

- **Representación del Contenido:** Los ítems se representan mediante un conjunto de atributos. En el caso de texto, es común usar el algoritmo **TF-IDF** (*Term Frequency-Inverse Document Frequency*) para ponderar la importancia de las palabras clave [1].
- **Perfil del Usuario:** Se construye un perfil que refleja las preferencias del usuario basándose en los atributos de los ítems que ha valorado positivamente.
- **Ventaja Principal:** Es robusto frente al problema del **arranque en frío de usuarios** y no sufre de la escasez de datos de interacción [1].
- **Desventaja:** Tiene limitada capacidad para la **serendipity**, ya que tiende a recomendar solo ítems que coinciden con el historial conocido del usuario (*over-specialization problem*) [20].

3.4.3. Sistemas Recomendadores Híbridos

Los Sistemas Recomendadores Híbridos combinan elementos de dos o más enfoques (generalmente CF y CB) con el objetivo de **mitigar las debilidades individuales** de cada método y aprovechar sus fortalezas [7, 1].

- **Objetivo Principal:** Resolver los problemas de **arranque en frío** (*cold-start*) y **escasez de datos** (*sparsity*) [1]. Por ejemplo, un usuario nuevo que aún no tiene interacciones puede recibir recomendaciones basadas en sus datos demográficos o en los atributos de su primera interacción (usando CB).
- **Técnicas de Hibridación:** Existen diversas formas de combinar los modelos [7, 20]:
 - (a) **Ponderado (*Weighted*):** Las puntuaciones de múltiples modelos se combinan linealmente.
 - (b) **Mixto (*Mixed*):** Las recomendaciones de diferentes modelos se presentan lado a lado.
 - (c) **Fusionado (*Cascading/Switching*):** Un modelo actúa como respaldo del otro. Por ejemplo, se usa CF, pero si el usuario es nuevo, el sistema **cambia** al modelo CB.

3.5. De la Recomendación a la confianza: modelos WoT

Una limitación intrínseca del filtrado colaborativo tradicional es la premisa de que todos los usuarios son igualmente fiables. Los modelos de **Web of Trust (WoT)** surgen para mitigar esta vulnerabilidad, introduciendo la confianza como un factor de ponderación crítico que mejora la robustez de las predicciones en entornos descentralizados [31].

Mientras que el Filtrado Colaborativo (CF) se apoya en la similitud estadística de opiniones, el factor confianza añade una capa de fiabilidad cualitativa a dichas relaciones [1, 20]. A diferencia de la similitud —que suele tratarse como una métrica simétrica—, la confianza se modela como un **grafo dirigido** y asimétrico. Esta naturaleza permite que la confianza se propague a través de la red social, facilitando la mitigación de ataques de perfiles falsos (*shilling attacks*) y resolviendo el problema del *cold start* mediante las declaraciones de fiabilidad iniciales de los usuarios [11, 26].

3.5.1. Predicción de Confianza (*Trust Prediction*)

El objetivo es predecir la confianza que un usuario A depositaría en la opinión o recomendación de otro usuario B . Esto se modela a menudo como un problema de **clasificación** o **regresión** que estima el grado de fiabilidad de la fuente [31].

Pipeline de Predicción

La predicción de confianza sigue un proceso estructurado [1]:

- **Extracción de Features (*Feature Extraction*):** Identificación de características que influyen en la confianza. Estas pueden ser la consistencia histórica del evaluador, su experiencia, su reputación, o su actividad reciente [31].
- **Resultados (*Epinions*):** Epinions.com es un caso de estudio clásico en la investigación de confianza [26]. Es una plataforma donde los usuarios pueden calificar productos y explícitamente designar a otros usuarios como "confiables" o "no confiables" (*Web of Trust*), proporcionando un *dataset* ideal para entrenar y validar modelos [11].

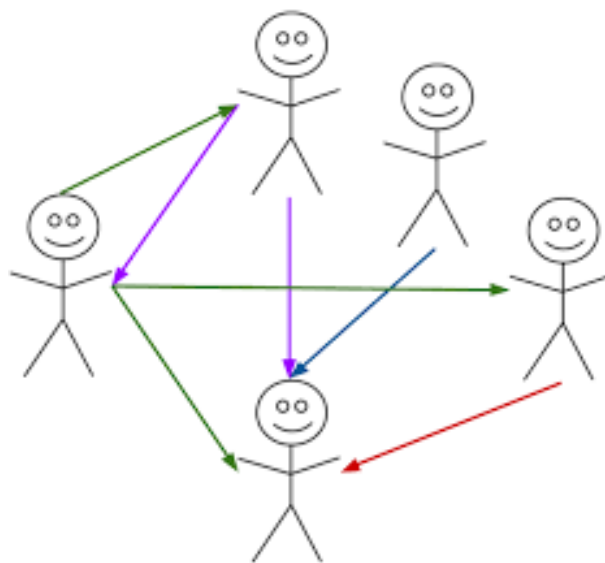


Figura 3.4: Modelos de confianza: Confianza delegada frente a Web of Trust (WoT). Fuente: [31].

3.5.2. Web de Confianza (WoT)

La Web de Confianza (*Web of Trust* o **WoT**) es una estructura conceptual o una red social implícita dentro de un SR que mapea las relaciones de confianza entre los usuarios [11].

- **Estructura:** Se modela como un **grafo dirigido** donde los nodos son los usuarios y las aristas representan una relación de confianza [28].
- **Propagación:** La confianza se propaga a lo largo de las rutas del grafo. Por ejemplo, si A confía en B, y B confía en C, A podría confiar indirectamente en C [11]. Los algoritmos deben calcular esta propagación y atenuarla con la distancia [31].

3.5.3. Propiedades y Métricas de Confianza

La confianza se distingue de la similitud en sus propiedades fundamentales [31]:

Propiedades de la Confianza

- **Asimetría:** Si el usuario A confía en B, no necesariamente implica que B confíe en A [11].
- **No Transitividad Estricta:** A pesar de que la confianza se propaga, esta se diluye con la distancia [31].
- **Contextualidad:** La confianza puede depender del dominio temático [1].

Métricas de Confianza

Estas métricas cuantifican la reputación de un nodo o la fuerza de un enlace en la WoT [26]:

- **Confianza Local:** Grado de confianza directa entre dos usuarios.
- **Confianza Global/Reputación:** Medida de la fiabilidad general de un usuario en toda la red [11].
- **Confianza Propagada:** Cálculo del camino de confianza más fuerte entre nodos no conectados directamente [31].

3.5.4. Aplicaciones y Esquemas de Confianza en SR

Aplicaciones de la Confianza

- **Mitigación de Ataques:** El factor confianza ayuda a mitigar ataques de *shilling* (perfiles falsos) [1].

- **Mejora de la Precisión de CF:** Los modelos priorizan las opiniones de vecinos confiables sobre los meramente similares [23].
- **Arranque en Frío (Cold Start):** Permite *arrancar*^a un usuario nuevo a través de sus declaraciones de confianza iniciales [7].

Esquemas de Confianza (*Trust Schemes*)

Algoritmos utilizados para incorporar la confianza en la predicción [31]:

- **Integración Pre-Filtrado:** Seleccionar vecinos confiables antes del CF [38].
- **Integración Ponderada:** Usar la confianza como factor de ponderación adicional en la suma del CF [1].
- **Modelos Basados en Propagación:** Uso de algoritmos como *Random Walk* para estimar confianza remota [11].

3.6. Redes Dinámicas: Influencia y Propagación

El paso final en la evolución de un SR es dejar de ver la recomendación como un evento aislado y verla como el inicio de un proceso dinámico de difusión. Esta sección aborda problemas específicos identificados en la implementación práctica de los Sistemas Recomendadores (SR) y presenta aplicaciones especializadas que extienden las metodologías de recomendación a la optimización de la difusión de información [1, 20].

3.6.1. Problemas Identificados en Sistemas Recomendadores

Aunque las tipologías tradicionales (CF, CB, Híbridos) mitigan la escasez de datos, persisten desafíos prácticos en la calidad y diversidad de las recomendaciones [7].

- **Sesgo de Popularidad:** Los SR tienden a recomendar ítems populares, marginando el "Long Tail" del catálogo [4].
- **Sensibilidad Contextual:** Dificultad para adaptar las recomendaciones a cambios rápidos en el contexto o el estado de ánimo del usuario [1].
- **Efectos de Filtro Burbuja:** Limitar la exposición del usuario a nuevos intereses al sobre-especializar las recomendaciones [11].

3.6.2. Generación de Recetas y Recomendación Inversa

Esta aplicación extiende los SR más allá de la simple predicción de un *rating* de un ítem existente, enfocándose en la síntesis de un nuevo ítem y la predicción de satisfacción [23].

Diseño Experimental y Satisfacción

Se describe el marco experimental donde el ítem es una receta (una combinación compleja de ingredientes y pasos).

- **Receta y Satisfacción:** Se mide la relación entre las características estructurales de la receta (ingredientes, tiempo) y el *feedback* de satisfacción obtenido del usuario [1].

Predicción Directa vs. Recomendación Inversa

Se distingue entre los dos problemas predictivos que surgen de este contexto [20]:

- **Predicción Directa:** ¿Cuál será el *satisfaction feedback* obtenido del usuario para una receta dada? (Problema estándar de regresión/clasificación).
- **Recomendación Inversa:** ¿Cuál es la receta que necesito generar para obtener una satisfacción específica (S_{target})? (Problema de optimización/generación).

3.6.3. Maximización de Influencia (*Influence Maximization*)

La maximización de influencia representa una de las extensiones más potentes de la teoría de redes aplicada, combinando la topología estructural con algoritmos de optimización para resolver problemas críticos de difusión en entornos sociales [11]. El objetivo fundamental consiste en identificar el conjunto mínimo de nodos iniciales, denominado *seed set*, que sea capaz de desencadenar la mayor cascada de adopción de un producto o información a través de la red [4, 31].

Este proceso de difusión no es aleatorio; depende intrínsecamente de la capacidad de los nodos elegidos para actuar como puentes entre distintas comunidades. Desde un punto de vista analítico, maximizar la influencia es, en esencia, un proceso de **percolación de información**, donde la señal debe "permear.^a a través de los enlaces disponibles hasta alcanzar una componente macroscópica de la red. Esta analogía técnica resulta fundamental, ya que los modelos de propagación utilizados aquí (como el modelo de Cascada Independiente o el de Umbral Lineal) comparten la misma base matemática que los procesos de transmisión que se detallarán en el **Bloque 4** (ver Sección 4.1). Allí se profundizará en cómo la estructura geométrica del grafo determina si una idea se extingue localmente o si logra percolar globalmente, transformándose en un fenómeno de masas [32].

3.6.4. Web de Confianza (WoT)

Estrategias de Búsqueda

- **Búsqueda Exhaustiva:** Analizar todas las combinaciones posibles de **seed sets**, siendo computacionalmente intratable para redes grandes [30].
- **Búsqueda Heurística:** Utilizar algoritmos que encuentran soluciones subóptimas pero eficientes en tiempo polinomial (ej., algoritmo *greedy* aproximado) [31, 14].

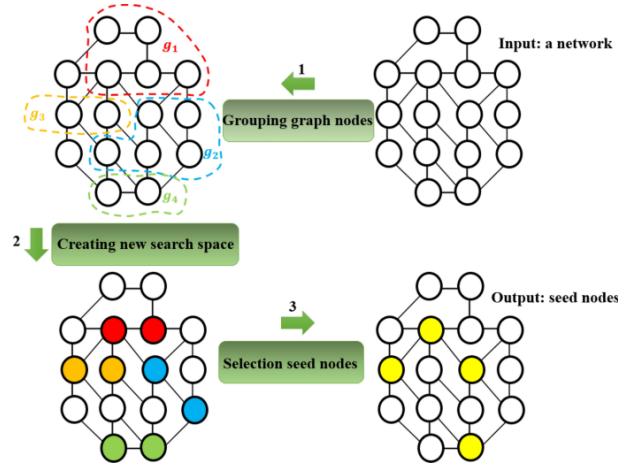


Figura 3.5: Algoritmo de maximización de influencia basado en la reducción del espacio de búsqueda en redes sociales. Fuente: [31, 11].

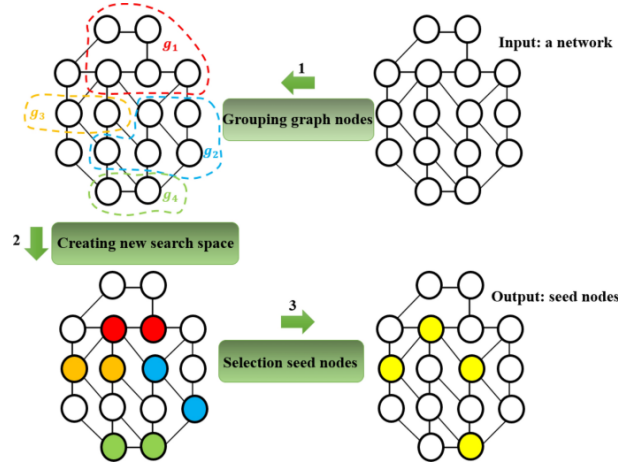


Figura 3.6: Algoritmo de maximización de influencia basado en la reducción del search space en redes sociales [31]

Modelos de Propagación Utilizados

La eficacia de una estrategia de difusión depende de la naturaleza del proceso de contagio. Se utilizan modelos dinámicos para simular cómo la información o influencia fluye a través de los enlaces de la red [32]:

- **ICM Model (*Independent Cascade Model*):** En este modelo estocástico, la activación de un nodo otorga una única oportunidad para activar a cada uno de sus vecinos con una probabilidad fija p . Es un proceso centrado en el evento de transmisión [11].
- **LTM Model (*Linear Threshold Model*):** A diferencia del anterior, este es un modelo centrado en la acumulación de influencia. Un nodo se activa solo cuando la suma ponderada de la influencia de sus vecinos activos excede un umbral crítico θ , representando fenómenos de presión social o adopción colectiva [11].

Optimización Heurística para la Selección de Semillas

Identificar el conjunto de usuarios iniciales (*seed set*) que maximice la propagación bajo los modelos ICM o LTM es un problema de optimización combinatoria de clase **NP-hard** [11]. Debido a la explosión del espacio de búsqueda en redes sociales de gran escala, el uso de técnicas de fuerza bruta es inviable, lo que justifica la implementación de algoritmos de inteligencia computacional.

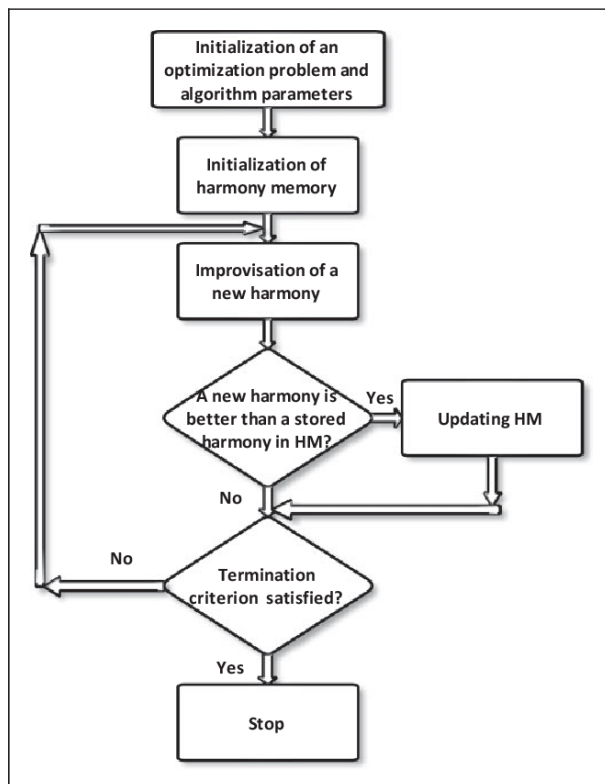


Figura 3.7: Flujo del algoritmo de optimización heurístico clásico [31]

En este contexto, se emplean algoritmos metaheurísticos para encontrar soluciones subóptimas pero altamente eficientes. Una de las técnicas destacadas es:

- **Harmony Search (HS):** Es un algoritmo bioinspirado en el proceso de improvisación de los músicos para encontrar una armonía perfecta. En la maximización de influencia, cada armonía representa un posible conjunto de semillas. El algoritmo evalúa la calidad de estos conjuntos simulando la propagación (mediante ICM o LTM) y refina la selección a través de procesos de memoria y mutación, permitiendo explorar grandes grafos de manera robusta [31].

Habiendo analizado cómo la confianza y los algoritmos de filtrado colaborativo procesan la información para el usuario, surge una pregunta crítica: ¿cómo se propaga esa información a gran escala a través de la arquitectura que hemos descrito? El éxito de un sistema recomendador depende de su capacidad de influencia, pero dicha influencia está sujeta a los límites físicos y críticos de

la conectividad global. En el capítulo siguiente, estudiaremos la **Teoría de la Percolación y las Dinámicas de Transmisión**, explorando los umbrales donde una red deja de ser una colección de nodos aislados para convertirse en un conductor masivo de difusión.

Capítulo 4

Percolación y Dinámicas de Transmisión

4.1. Introducción a la Percolación y Procesos de Transmisión

Si en los bloques anteriores analizamos la arquitectura de la red y cómo recomendar contenido en ella, en este capítulo abordaremos su fisiología: cómo la información o las enfermedades fluyen a través de sus enlaces. Para ello, utilizaremos la Teoría de la Percolación, que nos permitirá entender cuándo una red es lo suficientemente robusta para mantenerse unida o lo suficientemente vulnerable para permitir una epidemia global.

La **Teoría de la Percolación** es un marco matemático y conceptual fundamental dentro de la física estadística y la teoría de grafos que estudia la conectividad y el flujo en sistemas aleatorios o discretos [30]. Su núcleo reside en el análisis de cómo la eliminación o adición de elementos (nodos o enlaces) afecta la estructura de gran escala de una red, particularmente la formación de una **componente gigante** o un clúster conexo que abarca todo el sistema [4].

El concepto central de la percolación es la **Transición de Fase**. Cuando se modifica la probabilidad p de que un nodo o enlace esté activo, el sistema pasa abruptamente de estar compuesto por pequeños fragmentos aislados a tener una estructura macroscópica y altamente conectada [29]. La identificación del **umbral de percolación** (p_c) es crucial, ya que marca el punto crítico donde la conectividad global emerge [14].

La relevancia de la percolación para el estudio de las redes complejas se manifiesta en dos aspectos primordiales: la **robustez** estructural de la red (su capacidad para mantener la conectividad bajo ataque o fallo de sus componentes [2]) y su capacidad para **transmitir** fenómenos (información, rumores, enfermedades) a través de sus rutas [32].

4.1.1. Aplicaciones de la Teoría de la Percolación

La versatilidad de la Teoría de la Percolación le ha permitido trascender sus orígenes en la física de materiales para convertirse en una herramienta indispensable en el análisis

de la estructura y dinámica de sistemas complejos en diversas disciplinas [6]:

Física de Materiales y Química

Históricamente, la percolación se desarrolló para modelar fenómenos físicos de conectividad en materiales desordenados [13].

- **Conductividad Eléctrica:** Se utiliza para determinar el umbral crítico de concentración de partículas conductoras necesarias para que un material compuesto pase de ser aislante a conductor [30].
- **Flujo de Fluidos en Medios Porosos:** Modela el flujo de petróleo o agua a través de rocas o filtros. El umbral de percolación define la cantidad mínima de poros conectados necesaria para que el fluido atraviese el material [40].
- **Gelificación y Polimerización:** Describe la formación de redes poliméricas tridimensionales, determinando el punto exacto en el que las moléculas forman una única macromolécula que abarca todo el volumen [6].

Epidemiología y Salud Pública

En este campo, la percolación se relaciona directamente con los modelos de transmisión (SIR, SIS), enfocándose en la dispersión de patógenos o información [32, 21].

- **Contagio y Componente Crítica:** La eliminación aleatoria de nodos (vacunación) o enlaces (restricciones de contacto) se modela mediante percolación para hallar el umbral de inmunidad de rebaño necesario para evitar la propagación global [32].
- **Estrategias de Intervención:** En redes heterogéneas, la percolación demuestra que atacar los nodos de alto grado (*hubs*) es significativamente más eficiente que la eliminación aleatoria [2, 4].

Sociología y Redes Sociales

En las redes sociales y de comunicación, la percolación es clave para entender la difusión de información y la robustez social [11].

- **Difusión de Información/Rumores:** Modela cómo un rumor o un producto se propaga. El umbral de percolación representa el porcentaje mínimo de adoptantes para que la idea se convierta en un fenómeno generalizado [43, 31].
- **Robustez de Infraestructuras:** Se analiza la capacidad de internet o redes eléctricas para resistir fallos. La teoría revela que las redes libres de escala son robustas frente a fallos aleatorios pero vulnerables a ataques dirigidos [2].

Biología y Neurociencia

La conectividad funcional es un tema central en la biología de sistemas [18].

- **Redes Genéticas y Metabólicas:** La percolación estudia cómo la eliminación de un gen afecta la capacidad funcional de una célula [2].
- **Conectividad Neuronal:** Modelado de la transmisión de señales en el cerebro para entender cómo lesiones afectan la comunicación funcional a larga distancia [6].

4.2. Percolación en Redes Complejas

La **Teoría de la Percolación** ofrece un marco matemático esencial para estudiar cómo la **conectividad** y la **robustez** de una red se ven afectadas por la alteración aleatoria o selectiva de sus componentes [30]. Su núcleo reside en el análisis de determinar el punto en el que una red pierde su coherencia estructural o, inversamente, el punto donde una estructura funcional emerge [4].

El concepto central es la **transición de fase**, el cambio abrupto en la propiedad de conectividad global de la red (la aparición de una componente gigante) al variar un parámetro de probabilidad [2].

Conceptos Fundamentales de la Percolación

El proceso de percolación modela la probabilidad de que exista un camino entre dos nodos cualesquiera en un grafo, a menudo en presencia de fallos o de crecimiento aleatorio [14]. La clave es la formación del **Clúster Percolante** o **Componente Gigante** [29].

Componente Gigante Es el clúster o subgrafo más grande y conexo, cuya dimensión es del orden de la dimensión total del sistema (N) [4]. La existencia de la componente gigante permite el flujo o la propagación global de cualquier fenómeno a través de toda la red [32].

[Image of the emergence of a giant component in a random network during percolation]

Umbral de Percolación (p_c) Es el valor crítico de probabilidad que marca la transición de fase [40].

- Si la probabilidad de activación/supervivencia $p < p_c$, la red consiste principalmente en pequeños fragmentos aislados; el sistema **no percola**.
- Si $p \geq p_c$, se forma la componente gigante, y el sistema **percola**, permitiendo la conectividad global [30].

El valor de p_c es una propiedad estructural intrínseca de la red. Por ejemplo, en el modelo de Erdos-Rényi, $p_c \approx \frac{1}{\langle k \rangle}$ [12].

La teoría distingue fundamentalmente dos modos operativos: la percolación de sitio y la percolación de enlace [6, 26].

Percolación de Sitio (*Site Percolation*)

El modelo de percolación de sitio simula la **eliminación o desactivación de los nodos** (o vértices) de la red [30].

- **Mecanismo:** Cada nodo se activa con una probabilidad p . Los enlaces se mantienen si y solo si ambos nodos que conecta están activos [14].
- **Contexto de Aplicación:** Modela la **robustez** de una red frente al fallo de sus componentes individuales [2]:
 - (a) **Fallos en Infraestructuras:** Simulación del colapso de servidores en una red eléctrica o Internet [2].
 - (b) **Vacunación Aleatoria:** Inmunización de individuos (nodos) de manera no selectiva [32].
- **Robustez:** La resistencia de la red se mide por qué tan alto debe ser $1 - p$ antes de que la componente gigante se desintegre [4].

Percolación de Enlace (*Bond Percolation*)

El modelo de percolación de enlace simula la **eliminación o activación de las aristas** (o enlaces) de la red [30].

- **Mecanismo:** Cada enlace se activa con una probabilidad p , y se desactiva con probabilidad $1 - p$ [13].
- **Contexto de Aplicación:** Modela la fiabilidad de las conexiones o la capacidad de flujo [6]:
 - (a) **Redes de Comunicación:** Simulación de la caída de líneas de transmisión.
 - (b) **Propagación de Epidemias:** Probabilidad de infección entre dos individuos susceptibles [32].
- **Enfoque en la Conectividad:** La percolación de enlace es a menudo el enfoque preferido para estudiar los **procesos de transmisión** [29].

4.2.1. Modelado y Análisis de la Percolación

La Teoría de la Percolación ofrece modelos analíticos para predecir el umbral de percolación (p_c) basándose en la topología subyacente [30, 14].

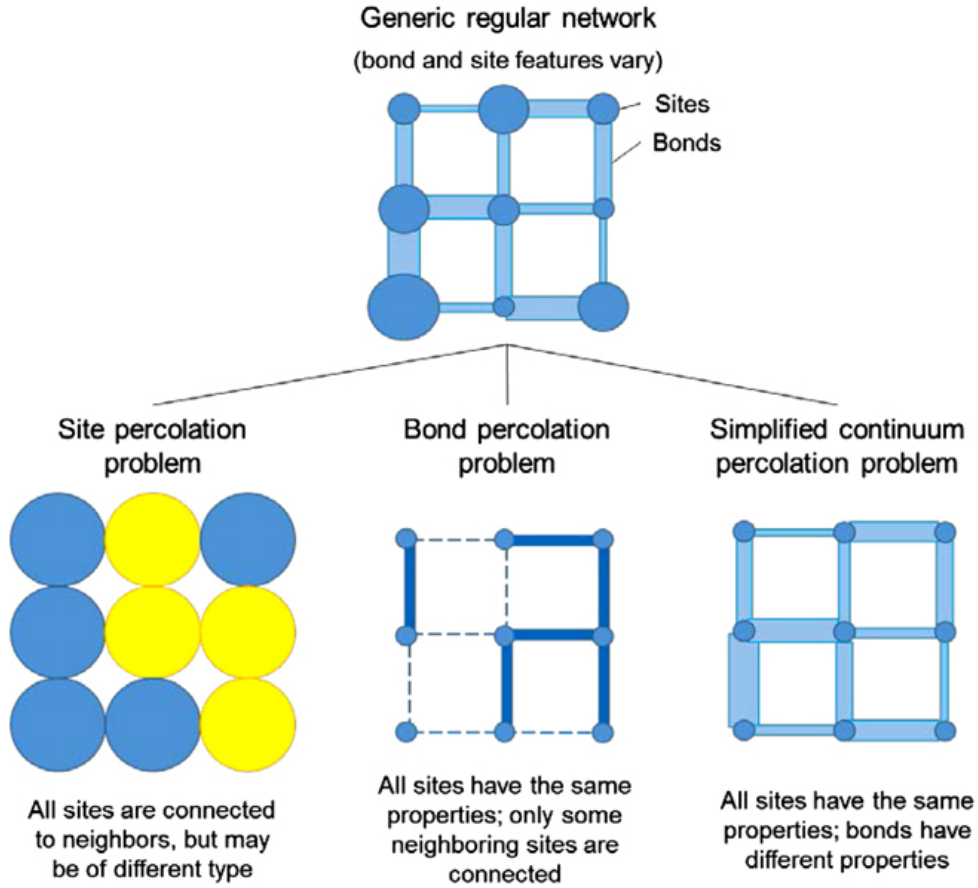


Figura 4.1: Percolación de Sitio y Enlace [28]

Modelo de Bond Percolation

El **Modelo de Percolación de Enlace** es el marco de análisis más directo para estudiar la conectividad [29].

Método de Función Generadora Este método matemático no es solo una abstracción; es lo que nos permite predecir el momento exacto en que la red social alcanza la masa crítica para que un contenido se vuelva viral.

Para redes con distribuciones de grado específicas, el análisis se realiza utilizando el método de la **Función Generadora de Probabilidad** [30]. Si $P(k)$ es la distribución de grado, la probabilidad $G_0(z)$ es:

$$G_0(z) = \sum_{k=0}^{\infty} P(k)z^k$$

Este método permite el cálculo explícito de p_c y el tamaño de la componente gigante $P_{\infty}(p)$ [27].

Percolación en Grafos Aleatorios (Erdos-Rényi)

El **Modelo de Erdos-Rényi** (ER) es la topología canónica para el estudio de la percolación [12].

Umbral Crítico (p_c) En el límite de $N \rightarrow \infty$, el umbral crítico para la aparición de la componente gigante ocurre cuando el grado promedio $\langle k \rangle$ es igual a 1 [30]:

$$\langle k \rangle_c = 1$$

Dado que $\langle k \rangle = Np_{ER}$, el umbral de percolación de enlace p_c es [4]:

$$p_c = \frac{1}{N}$$

[Image of the phase transition in Erdos-Rényi graphs highlighting the critical point at average degree 1]

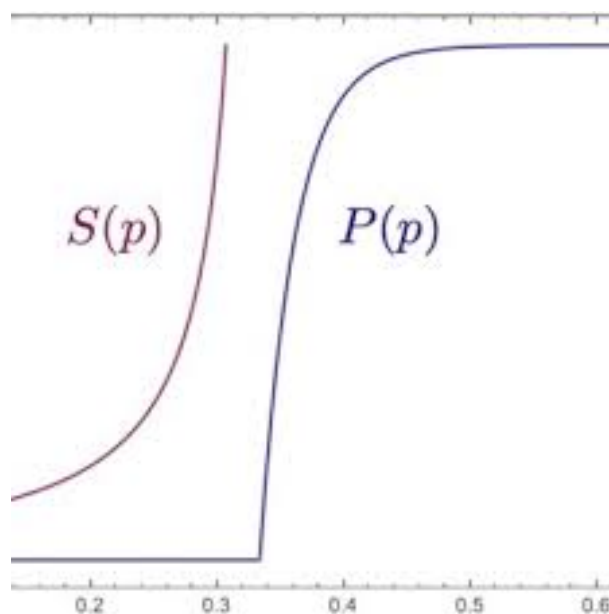


Figura 4.2: Ejemplo de umbral de percolación en el modelo Erdos-Rényi [29]

Percolación en Grafos Arbitrarios

La generalización de la teoría a **Grafos Arbitrarios** se centra en redes heterogéneas como las redes libres de escala [2].

Umbral de Percolación en Redes Heterogéneas Para una distribución de grado $P(k)$, el umbral de percolación de sitio p_c está determinado por [28]:

$$p_c = \frac{\langle k \rangle}{\langle k^2 \rangle - \langle k \rangle}$$

Fragilidad de las Redes Libres de Escala En redes ****Libres de Escala**** ($P(k) \sim k^{-\gamma}$):

- Si $\gamma \leq 3$, el segundo momento $\langle k^2 \rangle$ **diverge** [2].
- Esto resulta en un umbral crítico p_c que tiende a **cero** ($p_c \rightarrow 0$) [5].

Esto implica que las redes libres de escala son extraordinariamente **robustas frente a fallos aleatorios** pero extremadamente **frágiles ante ataques dirigidos** a los *hubs* [2, 4].

[Image of network robustness comparing random failure and targeted attack in scale-free networks]

Generalización La **heterogeneidad del grado** es el factor topológico dominante que determina la robustez estructural de una red compleja [6, 26].

4.3. Grafos Geométricos Aleatorios y Percolación

A diferencia de los modelos previos (ER) donde la conexión es puramente probabilística, los Grafos Geométricos Aleatorios introducen la métrica espacial. Esto es muy importante porque, en la vida real, los virus o los datos no saltan a cualquier nodo, sino que dependen de la proximidad física. Aquí, la percolación se convierte en un problema de 'alcance geográfico', donde el radio de conexión r determina si el sistema colapsa o permanece conectado.

Mientras que los modelos de grafos aleatorios (como Erdos-Rényi) se enfocan en la conectividad sin considerar la posición espacial [12], los **Grafos Geométricos Aleatorios** (*Random Geometric Graphs* o RGGs) introducen la **métrica espacial** como el factor determinante de la conectividad [35].

4.3.1. Definición de Grafos Geométricos Aleatorios

Un Grafo Geométrico Aleatorio $G(N, r)$ se construye bajo dos condiciones clave [35]:

1. **Colocación Espacial:** Se generan N nodos de manera aleatoria e independiente, distribuidos uniformemente sobre un espacio métrico subyacente, usualmente un cuadrado unitario en $\mathbb{R}^2$ [14].
2. **Regla de Conexión Basada en Distancia:** Dos nodos i y j se conectan con un enlace si y solo si la **distancia euclidiana** $d(i, j)$ entre ellos es menor o igual a un umbral fijo, llamado **radio de conexión** r .

$$G(N, r) = \{(i, j) \mid d(i, j) \leq r\}$$

Este modelo es fundamental para estudiar redes donde la **física de la proximidad** impone restricciones de conexión, como en las comunicaciones inalámbricas [6].

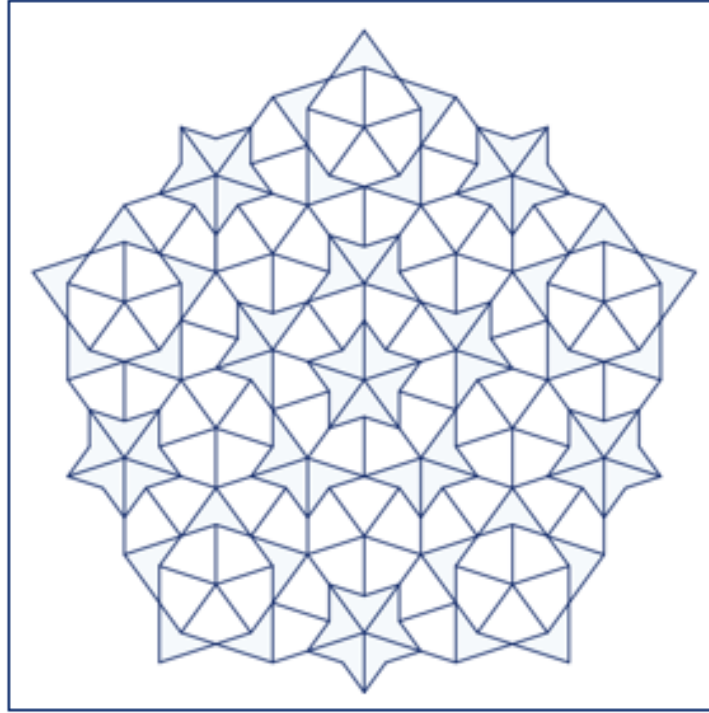


Figura 4.3: Ejemplo Grafos Geométricos [28]

4.3.2. Ejemplos y Propiedades de Grafos Geométricos

Los grafos geométricos son ideales para modelar sistemas donde la ubicación física es una limitación funcional [4].

- **Ejemplos de Redes Reales:**

- (a) **Redes de Sensores Inalámbricos:** Los nodos tienen un rango de transmisión limitado determinado por el radio r [13].
- (b) **Comunicación Inalámbrica Ad-Hoc:** Dispositivos que se conectan directamente según su distancia instantánea [11].
- (c) **Redes Neuronales Corticales:** Modelado de la conectividad local densa a cortas distancias [6].

- **Propiedades Topológicas Clave:**

- (a) **Alto Coeficiente de Agrupamiento (*Clustering*):** Los RGGs tienen un alto agrupamiento local intrínseco. Si A está cerca de B y C , es muy probable que B y C estén cerca entre sí.
- (b) **Relación entre Grado y Radio:** El grado promedio $\langle k \rangle$ es proporcional al área del círculo de conexión [35]:

$$\langle k \rangle \approx (N - 1)\pi r^2$$

4.3.3. Percolación en Grafos Geométricos

La percolación en RGGs estudia cómo la conectividad global depende del radio r o de la densidad de nodos ρ [14].

Dependencia del Radio Crítico (r_c) Existe una transición de fase nítida gobernada por un **radio crítico** [35]:

- Si $r < r_c$, la red está fragmentada en componentes pequeñas [26].
- Si $r \geq r_c$, se forma una componente gigante que atraviesa el espacio; la red **percola**[33].

Diferencia con Erdos-Rényi (ER) La geometría impone restricciones que diferencian ambos modelos [30]:

1. **Conectividad Local:** En ER es puramente probabilística; en RGGs está sujeta a la métrica euclidiana [35].
2. **Eficiencia:** En ER, la componente gigante emerge con $\langle k \rangle \approx 1$. En RGGs, se requiere un grado promedio $\langle k \rangle$ significativamente mayor (típicamente $\langle k \rangle \approx 4,5$ en 2D) para superar los obstáculos geométricos [8, 4].

4.4. Dinámicas de Difusión: de la Percolación a la Epidemiología

Habiendo comprendido la estática de la red mediante la percolación, pasamos ahora a la dinámica temporal. La propagación de una epidemia o un rumor es, en esencia, un proceso de percolación que ocurre a lo largo del tiempo. Veremos cómo el éxito de un virus (medido por el factor R_0) no depende solo de su agresividad, sino de si la topología de la red permite que el proceso 'percole' hasta alcanzar a la mayoría de la población.

Los Modelos de Transmisión son una aplicación directa y poderosa de la teoría de redes [30]. Su objetivo es entender y predecir la **dinámica temporal** de un fenómeno que se propaga de nodo a nodo (persona a persona, servidor a servidor) a través de los enlaces que definen la red [32]. Estos modelos permiten analizar la velocidad de la propagación, la extensión final del fenómeno y, crucialmente, las condiciones necesarias para su erradicación [4].

4.4.1. Transmisión sobre Redes

El estudio de la transmisión sobre redes se desarrolla en dos contextos principales que, aunque temáticamente distintos, comparten la misma estructura matemática de propagación en grafos: el contexto **epidemiológico** y el **social/tecnológico** [29, 32].

Contexto Epidemiológico Los modelos de transmisión nacieron históricamente para simular la difusión de enfermedades infecciosas (epidemias) [21]. En este contexto, la red representa a la población, los nodos son individuos y los enlaces representan contactos físicos o sociales que permiten el contagio. El principal objetivo es calcular el **Factor de Reproducción Básico** (R_0), que determina si una enfermedad puede propagarse de forma sostenida en la población [32].

Contexto Social y Tecnológico La propagación no se limita a patógenos. Estos modelos se aplican para simular [11, 31]:

- **Difusión de Innovación:** Cómo una nueva tecnología o producto se propaga a través de redes de influencia social.
- **Propagación de Rumores/Noticias:** La velocidad con la que la información (verdadera o falsa) se mueve en plataformas sociales o redes de comunicación.
- **Propagación de Malware:** El flujo de virus informáticos a través de redes de servidores.

En ambos contextos, la **topología de la red** (homogénea vs. heterogénea) juega un papel decisivo [4]. Por ejemplo, la presencia de *hubs* en redes libres de escala acelera drásticamente la propagación, haciendo que el umbral de criticidad sea mucho menor que en redes aleatorias [2].

Ecuaciones de Transmisión

Los procesos de propagación se modelan formalmente mediante sistemas de **ecuaciones diferenciales** o **ecuaciones de recurrencia** (en tiempo discreto) [30]. Estas ecuaciones describen cómo cambia la fracción de individuos en cada estado (ej., susceptibles, infectados, recuperados) a lo largo del tiempo.

Modelado de Compartimentos Los modelos más simples dividen la población total (N) en distintos **compartimentos** o estados mutuos [21]:

$$N = S(t) + I(t) + R(t) + \dots$$

Donde $S(t)$, $I(t)$, $R(t)$ son el número de individuos Susceptibles, Infectados y Recuperados, respectivamente, en el tiempo t .

Ecuaciones Diferenciales Ordinarias (EDOs) Las tasas de cambio entre los compartimentos se definen mediante EDOs [32]. El cambio en el número de infectados, por ejemplo, está determinado por dos términos opuestos:

1. **Tasa de Contagio (Flujo $S \rightarrow I$):** Proporcional al número de contactos entre susceptibles (S) e infectados (I) y a la probabilidad de transmisión (β).

2. **Tasa de Recuperación/Remoción (Flujo $I \rightarrow R$ o $I \rightarrow S$):** Proporcional al número de infectados (I) y a la tasa de remoción (γ).

En una aproximación de población homogénea (sin estructura de red explícita), la ecuación de cambio para el estado de infección (I) típicamente toma la forma [21]:

$$\frac{dI}{dt} = \beta \frac{SI}{N} - \gamma I$$

Esta formulación clásica debe ser adaptada en la Teoría de Redes para incluir el **grado de los nodos** y la **probabilidad de encuentro** en topologías específicas [32, 4].

Interpretación de la Transmisión

La principal herramienta analítica para interpretar y predecir el comportamiento de un proceso de transmisión es el **Factor de Reproducción Básico** (R_0) [32].

Factor de Reproducción Básico (R_0) R_0 se define como el número promedio de nuevas infecciones que un individuo infectado introduce en una población totalmente susceptible [21]. Este valor es crucial para determinar si una epidemia puede sostenerse o extinguirse.

- Si $R_0 < 1$: La epidemia **se extingue**. Cada infectado genera, en promedio, menos de una nueva infección.
- Si $R_0 = 1$: La epidemia se encuentra en un estado estacionario (endémico).
- Si $R_0 > 1$: La epidemia **crece** de manera exponencial, llevando a una propagación masiva.

Impacto de la Topología de la Red En redes complejas, R_0 se interpreta en el contexto de la estructura subyacente del grafo [29]. En una red con distribución de grado arbitraria $P(k)$, se puede demostrar que R_0 es proporcional al segundo momento de la distribución de grado, $\langle k^2 \rangle$ [32, 4]:

$$R_0 \propto \frac{\langle k^2 \rangle}{\langle k \rangle}$$

- **Redes Homogéneas (Bajo $\langle k^2 \rangle$):** El umbral crítico de propagación (la condición $R_0 > 1$) es relativamente alto. Se necesita una alta tasa de transmisión (β) para que la enfermedad se propague [30].
- **Redes Heterogéneas (Alto $\langle k^2 \rangle$):** En redes libres de escala, donde $\langle k^2 \rangle$ puede divergir, R_0 puede ser muy alto incluso para tasas de transmisión (β) extremadamente bajas [2]. Esto implica que, en estas redes, una enfermedad **siempre puede propagarse** de forma sostenida si existe alguna transmisibilidad ($\beta > 0$) [32]. La presencia de *hubs* actúa como super-propagador, eliminando el umbral de criticidad.

4.4.2. Modelo SIS (*Susceptible-Infected-Susceptible*)

El Modelo SIS es uno de los modelos compartimentales más simples, asumiendo que el individuo que se recupera de la infección no adquiere inmunidad y vuelve inmediatamente al estado de susceptible [32].

Definición Divide la población en dos estados:

- **S** (*Susceptible*): Individuos que pueden contraer la infección.
- **I** (*Infected*): Individuos que están infectados y son capaces de transmitir la enfermedad.

El flujo de estados es cíclico: $\mathbf{S} \rightarrow \mathbf{I} \rightarrow \mathbf{S}$.

Ecuaciones Clásicas (Sin Red) El sistema se describe mediante las tasas de cambio [32, 21]:

- La tasa de infección (β) mueve individuos de **S** a **I**.
- La tasa de recuperación (γ) mueve individuos de **I** de vuelta a **S**.

$$\begin{aligned}\frac{dI}{dt} &= \beta \frac{SI}{N} - \gamma I \\ \frac{dS}{dt} &= -\beta \frac{SI}{N} + \gamma I\end{aligned}$$

En el contexto de la red, estas ecuaciones se modifican para incluir la estructura del grafo y la probabilidad de contacto solo con los vecinos infectados [30, 4].

4.4.3. Modelo SIR (*Susceptible-Infected-Recovered*)

El Modelo SIR es el modelo más simple para describir una epidemia con inmunidad temporal o permanente [21]. Fue formulado por **Lowell Reed y Wade Hampton** y popularizado por Kermack y McKendrick.

Definición Divide la población en tres compartimentos no cíclicos:

- **S** (*Susceptible*): Individuos que pueden contraer la infección.
- **I** (*Infected*): Individuos infectados que son contagiosos.
- **R** (*Recovered/Removidos*): Individuos que se han recuperado, han muerto o han sido puestos en cuarentena, y que han adquirido inmunidad, por lo que **no pueden volver a ser susceptibles**.

El flujo de estados es secuencial y unidireccional: $\mathbf{S} \rightarrow \mathbf{I} \rightarrow \mathbf{R}$.

Ecuaciones Clásicas (Sin Red) El sistema de ecuaciones diferenciales que describe el flujo es [21]:

$$\begin{aligned}\frac{dS}{dt} &= -\beta \frac{SI}{N} \\ \frac{dI}{dt} &= \beta \frac{SI}{N} - \gamma I \\ \frac{dR}{dt} &= \gamma I\end{aligned}$$

Donde β es la tasa de infección y γ es la tasa de remoción/recuperación. En este modelo, si la infección comienza, se detiene cuando $\frac{dI}{dt} \leq 0$, lo que ocurre cuando la fracción de susceptibles restantes es menor que el umbral de inmunidad de grupo [32].

*El análisis de la percolación y los modelos compartimentales nos permite concluir que la estructura de la red es el determinante crítico de su resiliencia. Se ha verificado que factores como el umbral de propagación y el valor R_0 dependen de la heterogeneidad del grafo, asumiendo hasta ahora una interacción estocástica. No obstante, existe un nivel de interacción más profundo: la **sincronía temporal**. En el bloque final, trascenderemos el flujo de información para abordar la **Dinámica de Sincronización**, el fenómeno donde la estructura y la transmisión convergen para generar orden colectivo y permitir la detección funcional de comunidades desde una perspectiva dinámica.*

Capítulo 5

Dinámica de Sincronización en Redes

5.1. Emergencia del Orden: Sincronización en Sistemas Complejos

Si bien técnicas como la Factorización Matricial o la Descomposición en Valores Singulares (SVD), detalladas en la Sección 3.3.5, optimizan la precisión de la predicción mediante la identificación de patrones latentes, estos modelos operan en un vacío social donde se asume que cada interacción registrada es legítima y fidedigna.

Esta omisión de la fiabilidad de la fuente motiva la transición hacia modelos basados en la confianza, los cuales permiten discernir entre la similitud estadística y la credibilidad del actor dentro de la red.

El estudio de la **Dinámica de Sincronización** en redes complejas es un campo interdisciplinario que une la teoría de grafos, los sistemas dinámicos y la física estadística [3]. Su objetivo es entender cómo la **interacción local** entre componentes acoplados (los nodos) conduce a la emergencia de un **comportamiento coherente y ordenado** a nivel global [6].

Este fenómeno es ubicuo en la naturaleza y la tecnología. Se observa en [40]:

- La biología (ej., el parpadeo de las luciérnagas, el ritmo cardíaco y los ciclos circadianos).
- La neurociencia (ej., la actividad coordinada de las neuronas en el cerebro).
- La ingeniería (ej., el funcionamiento coordinado de los generadores eléctricos en una red de energía).

En todos estos casos, la topología subyacente de la red es el factor dominante que dicta si la sincronización es posible y bajo qué condiciones de acoplamiento [3, 27].

5.1.1. Clustering Basado en la Sincronización

El **Clustering Basado en la Sincronización** propone una perspectiva dinámica para la detección de comunidades, ofreciendo una alternativa a los métodos puramente

pico [40]. El modelo de Kuramoto es la herramienta canónica para abordar este problema [24].

5.2.1. Osciladores No Idénticos: El Modelo de Kuramoto

El **Modelo de Kuramoto** es el paradigma matemático para entender la sincronización en sistemas de elementos acoplados [24, 40]. Reduce la dinámica de cada unidad a una sola variable: su **fase** [3].

Definición Matemática La dinámica de la fase θ_i del oscilador i se rige por la siguiente Ecuación Diferencial Ordinaria (ODE) [24]:

$$\frac{d\theta_i}{dt} = \omega_i + \frac{K}{N} \sum_{j=1}^N \sin(\theta_j - \theta_i)$$

Donde:

- ω_i : Es la **frecuencia natural** del oscilador i , extraída de una distribución $g(\omega)$ [40].
- K : Es la **fuerza de acoplamiento** global [3].
- $\sin(\theta_j - \theta_i)$: Es la función de acoplamiento que promueve la sincronía al reducir la diferencia de fase [40].

El modelo predice una **transición de fase continua** donde un aumento en K provoca que una fracción creciente de los osciladores se sincronice [24, 3].

Representación Mean-Field

La **Representación Mean-Field** es la clave para analizar el modelo de Kuramoto cuando N es muy grande [40].

El Parámetro de Orden (r) El grado de sincronización se describe mediante el **Parámetro de Orden Complejo** $Z(t)$ [24]:

$$Z(t) = r(t)e^{i\psi(t)} = \frac{1}{N} \sum_{j=1}^N e^{i\theta_j}$$

Donde:

- $r(t) = |Z(t)|$: Es el **grado de coherencia**. Escala de $r \approx 0$ (estado incoherente) a $r \approx 1$ (estado sincronizado) [40, 3].
- $\psi(t)$: Es la **fase promedio** del clúster sincronizado.

Evolución del Parámetro de Orden En el límite $N \rightarrow \infty$, la ecuación se simplifica a [40]:

$$\frac{d\theta_i}{dt} = \omega_i + Kr \sin(\psi - \theta_i)$$

Esta ecuación demuestra que la dinámica de cada oscilador ya no depende de las fases individuales, sino únicamente de la **fuerza del campo medio** (Kr) y de la **fase promedio** (ψ) del colectivo [3].

Umbral de Sincronización

El **Umbral de Sincronización** (K_c) es el valor crítico de la fuerza de acoplamiento global K a partir del cual el sistema de osciladores de Kuramoto pasa de un estado **incoherente** ($r \approx 0$) a un estado de **sincronización parcial** ($r > 0$) [24]. Este es un valor fundamental, ya que caracteriza la propensión de un sistema a alcanzar el orden colectivo [40].

Cálculo del Umbral El valor de K_c se deriva analíticamente considerando la estabilidad del estado no sincronizado (donde $r = 0$). Para el modelo Kuramoto con acoplamiento global y en el límite $N \rightarrow \infty$, Kuramoto demostró que el umbral está determinado por la densidad de la distribución de las frecuencias naturales $g(\omega)$ en el origen ($\omega = 0$) [24, 3]:

$$K_c = \frac{2}{\pi g(0)}$$

Donde $g(0)$ es la altura de la distribución de frecuencias en el punto $\omega = 0$.

Significado Físico

- **Condición de Emergencia:** K_c representa la fuerza mínima de interacción necesaria para superar la tendencia natural de los osciladores a dispersarse debido a sus distintas frecuencias ω_i [40].
- **Distribuciones Estrechas:** Si la distribución $g(\omega)$ es estrecha (es decir, $g(0)$ es alta), K_c es **bajo**. La sincronización ocurre fácilmente [3].
- **Distribuciones Amplias:** Si $g(\omega)$ es ancha ($g(0)$ es baja), K_c es **alto**. Se requiere una fuerza de acoplamiento mucho mayor para forzar la sincronización [27].

El umbral es una característica intrínseca de la diversidad del sistema. Si todos los osciladores fueran idénticos, K_c sería cero, y la sincronización se alcanzaría instantáneamente [40].

5.2.2. Sincronización Social: Modelos de Opinión y Consenso

Los modelos de osciladores de fase (Kuramoto) han inspirado modelos dinámicos en las ciencias sociales [3]. Los **Modelos de Tasa** (*Rate Models*) transportan la idea de la evolución de fase a la dinámica de atributos o estados sociales, como la **opinión** [25].

Modelo de Opinión: Opinion Changing Rate Model (Rate Model)

El **Modelo de Tasa de Cambio de Opinión** (*Opinion Changing Rate Model* o OCR) es un mecanismo dinámico que modela la evolución de una opinión escalar de un individuo ($x_i(t)$) dentro de una red de influencia social [25]. Se define como un mecanismo dinámico de **influencia social**.

Definición y Mecanismo El modelo postula que la tasa de cambio de la opinión de un individuo i es proporcional a la diferencia entre su opinión actual y la opinión promedio de sus vecinos en la red [25]. A diferencia de los modelos de fase que utilizan el $\sin(\Delta\theta)$, los modelos de opinión a menudo utilizan la diferencia lineal o una función truncada [3].

La dinámica del cambio de opinión está dada por:

$$\frac{dx_i}{dt} = \sum_{j \in \mathcal{N}(i)} J_{ij} \cdot f(x_j - x_i)$$

Donde x_i es la opinión, $\mathcal{N}(i)$ es el vecindario, J_{ij} es el peso de la influencia y f es la función de interacción [25, 26].

Relevancia para el Clustering La capacidad de estos modelos para alcanzar el consenso o la polarización está fuertemente ligada a la **topología** [26]. En redes con estructuras de comunidad claras, los individuos dentro de un mismo grupo tenderán a alcanzar un consenso interno, mientras que los grupos separados pueden llevar a la **polarización** y a la formación de clústeres dinámicos [6].

Clustering Dinámico Basado en el Modelo OCR

El **Clustering Dinámico** utiliza modelos como el OCR para inferir la estructura comunitaria [25]. El principio radica en que la **evolución dinámica** del sistema revelará los grupos densamente conectados [26].

Mecanismo de Formación de Clústeres Cuando una red posee una estructura modular, la dinámica se comporta de la siguiente manera [25]:

1. **Consenso Intracomunitario:** Los nodos dentro de un clúster tienen fuertes enlaces mutuos. Esto provoca que sus opiniones converjan rápidamente hacia un consenso local [26].

2. **Polarización Intercomunitaria:** Los enlaces entre diferentes clústeres son más débiles, lo que causa que las opiniones de los grupos distintos se separen o **polaricen** [6].

Ventajas del Enfoque Dinámico Este método ofrece una perspectiva funcional sobre el *clustering*, evaluando la **coherencia funcional** de los grupos bajo un proceso de influencia [25, 6].

Clustering Dinámico: La Red Zachary Karate Club

La **Zachary Karate Club Network** constituye el *dataset* canónico y el *benchmark* fundamental en el análisis de redes sociales para la validación de algoritmos de detección de comunidades [26]. Este caso de estudio permite realizar el cierre conceptual de la presente investigación, al demostrar cómo el problema de la partición de grafos —abordado desde una perspectiva puramente estructural en el **Bloque 2** (ver Sección 2.6)— halla una solución natural y elegante a través de la física de los procesos dinámicos.

Contexto y Dualidad Estructura-Dinámica La red documenta las interacciones de 34 miembros de un club de kárate que, debido a un conflicto interno entre el instructor y el administrador, se fracturó en dos facciones claramente diferenciadas. Mientras que en la teoría de grafos estática intentamos predecir esta ruptura optimizando métricas como la modularidad (Q), el enfoque de sincronización permite que la división emerja orgánicamente a partir de las tasas de interacción entre los nodos [3, 26].

Resolución Dinámica de la División Al someter la red a un modelo de osciladores o de tasa de cambio de opinión (OCR), el sistema resuelve la detección de comunidades mediante la evolución temporal de sus estados [25]:

1. **Inicialización y Desorden:** Se asignan fases (θ_i) u opiniones (x_i) aleatorias, representando un estado de máxima entropía o falta de consenso inicial.
2. **Atracción Local:** Debido a la alta densidad de enlaces internos (cohesionados en el Bloque 2), los nodos dentro de una misma facción ejercen una fuerza de atracción mutua superior, arrastrando sus fases hacia un valor promedio local [26].
3. **Polarización y Consenso Emergente:** El sistema **converge** hacia un estado de sincronización parcial donde los dos clústeres se polarizan en valores opuestos. Este resultado no solo replica la división sociológica histórica, sino que valida que la sincronización es una herramienta de *clustering* más robusta que los métodos estáticos, ya que integra la topología en el flujo dinámico de la información [25].

Esta convergencia final une el principio y el fin de este trabajo, demostrando que la estructura (grafos) y la función (sincronización) son dos caras de la misma moneda en el estudio de la Inteligencia Computacional en redes sociales.

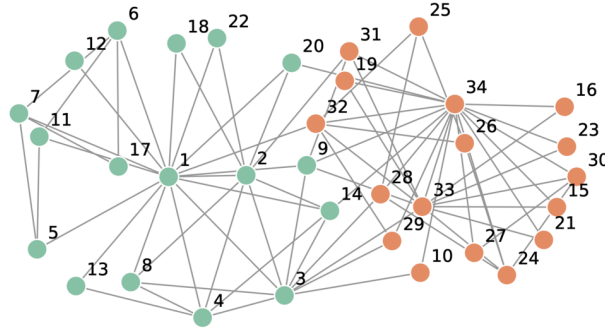


Figura 5.2: Red Zachary Karate Club: Clústeres detectados mediante la dinámica de sincronización. Fuente: [26].

5.3. Sincronización de Sistemas Caóticos

La sincronización de sistemas caóticos permite que sistemas deterministas pero impredecibles logren evolucionar de manera coherente debido a un acoplamiento [33]. Este campo ha encontrado aplicaciones en la comunicación segura y la ingeniería de circuitos [6].

5.3.1. Dinámica Sincronizada y Sistemas Caóticos

Definición de Sistemas Caóticos Un **Sistema Caótico** es un sistema dinámico determinista caracterizado por la **sensibilidad extrema a las condiciones iniciales** [40]. Ejemplos clásicos incluyen el oscilador de Lorenz y el circuito de Chua [6].

El Fenómeno de la Sincronización La sincronización en este contexto no es trivial. Dada la naturaleza divergente de las trayectorias, el acoplamiento debe ser lo suficientemente fuerte para superar las fuerzas expansivas del caos [33]. La sincronización se logra cuando $\|X(t) - Y(t)\| \rightarrow 0$ [6].

Sincronización de Dos Osciladores Acoplados Idénticos

El caso base es el acoplamiento de dos sistemas idénticos [6].

Definición del Caso Base Consideremos dos sistemas caóticos idénticos X y Y , descritos por el mismo conjunto de ecuaciones diferenciales, pero con diferentes condiciones iniciales [6]:

$$\begin{aligned}\frac{dX}{dt} &= F(X) + C(Y - X) \\ \frac{dY}{dt} &= F(Y) + C(X - Y)\end{aligned}$$

Donde F es la función dinámica no lineal y C es la matriz de acoplamiento [3].

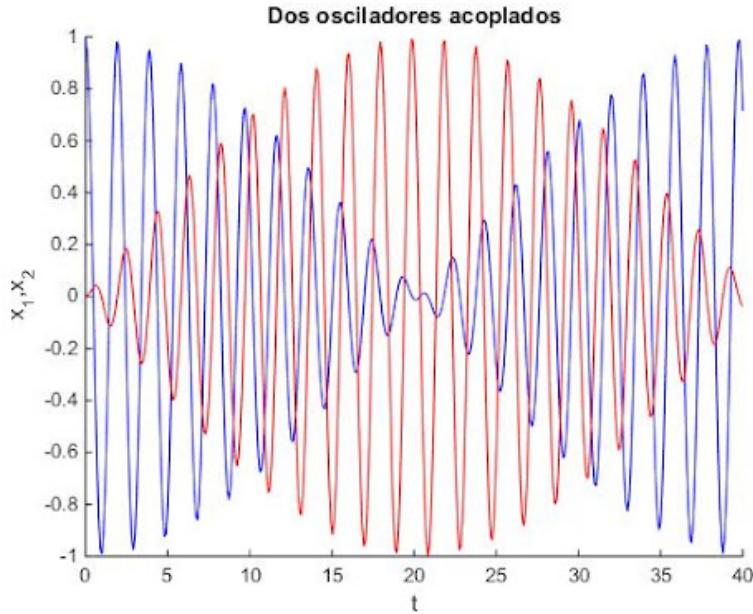


Figura 5.3: Gráfico de dos osciladores acoplados bajo el modelo de Kuramoto. Fuente: [27].

Análisis de la Sincronización Completa (*Complete Synchronization*) La **Sincronización Completa (CS)** se logra cuando las trayectorias de X y Y coinciden exactamente. Para analizar la estabilidad, se utiliza el **vector de error** $E = X - Y$ [27].

La dinámica del error E debe tender a cero [6]:

$$\frac{dE}{dt} = DF(X)E - 2CE$$

La estabilidad está determinada por los **Exponentes de Lyapunov Condicionales** (λ_c) [33]. Para la estabilidad, todos los exponentes en el subespacio transversal deben ser ****negativos**** [3, 34].

5.3.2. El Esquema de Sincronización de Pecora y Carroll (PC)

El trabajo seminal de Pecora y Carroll en 1990 demostró que es posible lograr la sincronización entre sistemas caóticos idénticos utilizando acoplamiento **unidireccional**.

La Configuración de Pecora y Carroll

La configuración PC divide el sistema dinámico idéntico en dos componentes: **conducción** y **respuesta**.

División del Sistema Cualquier sistema dinámico autónomo puede descomponerse en dos subsistemas [33, 6]:

1. **Subsistema de Conducción (*Driving System*) - X :** El sistema generador de la señal caótica.

2. **Subsistema de Respuesta (*Response System*) - Y :** El sistema esclavo que imita la dinámica del conductor.

PC Synchronization Scheme

El esquema PC se implementa forzando a variables del sistema de respuesta a seguir la evolución de las del sistema de conducción [33].

Implementación Unidireccional Si elegimos la variable x_1 como señal de conducción, la dinámica acoplada es [33, 6]:

1. **Sistema Conductor (X):** $\dot{x}_1 = f_1(X), \dot{x}_2 = f_2(X), \dot{x}_3 = f_3(X)$
2. **Sistema de Respuesta (Y):** $y_1 = x_1$ (Forzamiento), $\dot{y}_2 = f_2(x_1, y_2, y_3), \dot{y}_3 = f_3(x_1, y_2, y_3)$

La estabilidad se determina mediante los **Exponentes de Lyapunov Condicionales** (λ_c) del subsistema de respuesta [33, 3]. Para el éxito de la sincronización, **todos** los λ_c deben ser **negativos**.

Análisis de Estabilidad (Exponentes de Lyapunov Condicionales) La clave para determinar si este esquema funciona es analizar la estabilidad de la variedad de sincronización $y_i = x_i$ para las variables de respuesta no forzadas (y_2, y_3) [33, 34].

La estabilidad se determina calculando los **Exponentes de Lyapunov Condicionales** (λ_c) del subsistema de respuesta.

- Los λ_c se calculan considerando al sistema de conducción X como una **entrada externa fija** para el sistema de respuesta Y [33].
- Para que la sincronización sea exitosa (es decir, para que el error $E = Y_r - X_r$ converja a cero), **todos** los Exponentes de Lyapunov Condicionales deben ser **negativos** [33, 6].

El teorema de Pecora y Carroll demostró que no todas las divisiones de un sistema caótico permiten la sincronización estable, y la estabilidad depende de qué variables se elijan como señales de conducción [33].

5.3.3. Análisis de Estabilidad y Acoplamiento General

El análisis de la estabilidad es crucial para determinar la viabilidad de la sincronización en sistemas caóticos acoplados. Este análisis no solo confirma si el estado sincronizado puede alcanzarse, sino también bajo qué condiciones estructurales y paramétricas se mantiene [3].

Sincronización Completa: Estabilidad

La **Sincronización Completa (CS)** en sistemas caóticos acoplados ocurre cuando todas las trayectorias de los osciladores X_i son idénticas, es decir, $X_i(t) = S(t)$, donde $S(t)$ es la trayectoria del colector de sincronización [6].

Condición de Estabilidad La estabilidad del estado sincronizado está determinada por el comportamiento de pequeñas perturbaciones aplicadas en la dirección **transversal** (perpendicular) al colector de sincronización [3, 27].

- **Subespacio Tangencial:** Las perturbaciones a lo largo del colector se mantienen estables, ya que el sistema caótico sigue su trayectoria ($S(t)$).
- **Subespacio Transversal:** Para que la sincronización sea estable, las perturbaciones en el subespacio ortogonal al colector deben decaer exponencialmente con el tiempo [3]. Esto se garantiza si todos los **Exponentes de Lyapunov Condicionales (CLEs)** asociados a las direcciones transversales son **negativos** [6].

Los CLEs miden la tasa promedio de crecimiento o decaimiento de las perturbaciones [6]. El acoplamiento debe ser lo suficientemente fuerte para que la fuerza de convergencia supere la tasa natural de divergencia inherente al caos (dada por los Exponentes de Lyapunov positivos del sistema sin acoplar) [6, 3].

General Linear Coupling Scheme

En el estudio de redes complejas, la forma más común de modelar la interacción entre nodos idénticos es mediante el **Esquema de Acoplamiento Lineal General (GLCS)** [6, 3].

Definición Matemática Consideremos una red de N sistemas caóticos idénticos X_i , donde $i = 1, \dots, N$. La dinámica de cada nodo se define como [6, 26]:

$$\frac{dX_i}{dt} = F(X_i) + \sigma \sum_{j=1}^N L_{ij} H(X_j)$$

Donde:

- $F(X_i)$: Describe la dinámica intrínseca y no lineal del nodo individual i [3].
- σ : Es la **fuerza global de acoplamiento** (un escalar positivo) [6].
- L : Es la **Matriz Laplaciana Generalizada** de la red, donde L_{ij} codifica la topología [3].
- H : Es la **función de acoplamiento local**, que mapea el vector de estado del nodo X_j al subespacio a través del cual se produce la interacción [6].

Este formalismo es crucial, ya que permite separar la **dinámica local** (F) de la **estructura de la red** (L) y de la **forma de acoplamiento** (H) [3, 6].

Naturaleza del Esquema de Acoplamiento

La matriz Laplaciana L en el GLCS define la naturaleza del acoplamiento entre los nodos de la red, distinguiendo entre la direccionalidad del flujo de la señal [6, 3].

- **Unidireccional (*Unidirectional Case*):** El flujo de la señal es en un solo sentido [3, 6].
 - (a) Si el nodo A influye en B, el enlace $A \rightarrow B$ existe y es representado en la matriz Laplaciana [6].
 - (b) El nodo B **no influye en A**, por lo que el enlace $B \rightarrow A$ está ausente [6].
 - (c) En este caso, la matriz de conectividad L es **asimétrica** [3]. Este esquema es esencial para modelar la sincronización tipo Conductor-Respuesta (como en el esquema Pecora y Carroll) [33].
- **Bidireccional (*Bidirectional Case*):** Existe **acoplamiento mutuo** entre los nodos [3].
 - (a) Si el nodo A influye en B, B también influye en A con la misma fuerza o con una fuerza relacionada [3, 6].
 - (b) En redes no dirigidas o bidireccionales, la matriz de conectividad L es **simétrica** ($L_{ij} = L_{ji}$) [6]. Este esquema se utiliza para modelar la interacción simétrica entre vecinos, común en redes físicas donde la interacción es recíproca [3, 26].

Esta representación como grafo dirigido rescata los conceptos de **centralidad y conectividad** analizados previamente en el Capítulo 1.7, pero aplicados ahora a la validación de la fuente de información. Bajo este prisma, no basta con que un nodo presente un perfil de gustos similar al nuestro; para que la recomendación sea robusta, el nodo debe ser calificado como *fiable* dentro del camino de confianza que se propaga por la red social.

5.4. Sincronización en Redes de Osciladores de Fase

La transición del modelo de Kuramoto globalmente acoplado a la sincronización en redes complejas topológicamente estructuradas requiere la generalización del esquema de acoplamiento para incorporar la matriz de conectividad del grafo [3, 6].

5.4.1. Osciladores Acoplados y Acoplamiento Lineal

En una red compleja con N nodos, la dinámica de fase de cada oscilador i se ve influenciada únicamente por sus vecinos directos, en contraste con el acoplamiento global del modelo original de Kuramoto [24, 40].

Ecuación Generalizada de Osciladores Acoplados La dinámica del oscilador de fase i en una red con topología definida por la matriz de adyacencia A se describe mediante la siguiente ecuación [3]:

$$\frac{d\theta_i}{dt} = \omega_i + \sum_{j=1}^N A_{ij} H(\theta_j - \theta_i)$$

Donde:

- A_{ij} : Es el elemento de la matriz de adyacencia que codifica la topología de la red [30]. Si el acoplamiento es bidireccional, A es simétrica.
- $H(\theta_j - \theta_i)$: Es la función de acoplamiento, siendo $\sin(\theta_j - \theta_i)$ el caso más común [24].

Acoplamiento Lineal Para generalizar el análisis de estabilidad a sistemas dinámicos más complejos se utiliza el **Acoplamiento Lineal General**, donde la interacción se modela como una combinación lineal de las diferencias de sus estados [6, 26].

La dinámica del nodo i es [6]:

$$\frac{dX_i}{dt} = F(X_i) + \sigma \sum_{j=1}^N L_{ij} G(X_j)$$

Donde $F(X_i)$ es la dinámica local, σ es la fuerza de acoplamiento, L es la matriz Laplaciana de la red, y G es la función de acoplamiento [3]. Este formalismo permite el análisis de la sincronización en redes donde los nodos son sistemas caóticos [6].

5.4.2. La Función de Estabilidad Maestra (*The Master Stability Function*)

La **Función de Estabilidad Maestra (MSF)**, introducida por Pecora y Carroll, es una herramienta analítica clave que desacopla la **dinámica del nodo** de la **topología de la red** [34].

Desacoplamiento Topológico-Dinámico El análisis de la estabilidad se logra linealizando las ecuaciones de error en torno a la variedad de sincronización [6]. Este proceso

conduce a un sistema donde las **contribuciones topológicas** (autovalores de la Laplaciana) y las **dinámicas** aparecen separadas [34, 27]:

$$\frac{d\delta X_k}{dt} = [DF(S(t)) - \sigma \lambda_k DG(S(t))] \delta X_k$$

Donde δX_k es el vector de perturbación asociado al autovalor λ_k de la matriz Laplaciana L .

Definición y Cálculo de la MSF La MSF se define como el máximo Exponente de Lyapunov de un sistema variacional [34]:

$$\lambda_{\max}(\alpha) = \text{MSF}(\alpha)$$

Donde α toma los valores $\sigma \lambda_k$ [3].

1. **Cálculo de la MSF:** Se realiza una **única vez** para un sistema local dado, independientemente de la topología [34].
2. **Análisis de Estabilidad:** Se evalúa la MSF en los valores $\alpha = \sigma \lambda_k$ para todos los autovalores no nulos de L [6].
3. **Condición de Sincronización:** La sincronización es **estable** si y solo si la $\text{MSF}(\sigma \lambda_k)$ es **negativa** para todos los autovalores λ_k no nulos [34, 25].

La MSF permite predecir si una red con cualquier topología alcanzará la sincronización para una fuerza de acoplamiento σ dada [6, 27].

Conclusiones

A lo largo del presente trabajo se ha explorado la intersección entre la estructura topológica de las redes y los procesos dinámicos que definen la inteligencia computacional moderna. Este análisis permite concluir que la topología de la red no es un mero escenario pasivo, sino el sustrato fundamental que dicta la eficiencia de cualquier algoritmo. Mientras que la teoría de grafos clásica proporciona las herramientas de medición esenciales —como el grado, la densidad y la centralidad—, la aplicación de modelos de red compleja como *Small World* o *Preferential Attachment* es lo que realmente permite explicar la sorprendente rapidez y eficiencia de los procesos de comunicación en las sociedades contemporáneas.

En este contexto, el estudio de los sistemas recomendadores ha subrayado que el éxito en la gestión de la economía de la atención no depende exclusivamente del volumen de datos procesados, sino de la calidad del filtrado colaborativo basado en la confianza. La implementación de estructuras tipo *Web of Trust* (WoT) logra transformar el grafo social en un sistema de validación distribuida, permitiendo una personalización más profunda y, sobre todo, dotando a la red de una resiliencia necesaria frente a intentos de manipulación externa. Esta relación entre confianza y estructura se complementa con el análisis de la percolación, el cual revela que la conectividad global es extremadamente sensible a la integridad de los nodos críticos. En términos de dinámica de transmisión, esto demuestra que tanto para mitigar una epidemia (modelos SIR y SIS) como para fomentar la difusión de una innovación, la estrategia óptima debe basarse en la geometría local de la red y no solo en las tasas de contagio individuales.

Finalmente, el estudio de la sincronización y el comportamiento colectivo en redes revela que el orden macroscópico es un fenómeno emergente que surge de la interacción de osciladores individuales. Mediante el uso de herramientas analíticas avanzadas como la *Master Stability Function*, se ha verificado que la estabilidad de un sistema sincronizado es una propiedad intrínseca del acoplamiento lineal y la topología del grafo. Este hallazgo resulta fundamental para comprender cómo se forman las corrientes de opinión y los comportamientos de masas en las plataformas digitales actuales. En definitiva, la inteligencia computacional aplicada a redes sociales debe ser entendida como una disciplina unificada que utiliza la física estadística y la dinámica no lineal para descifrar, con precisión matemática, la complejidad de la interacción humana en la era digital.

Bibliografía

- [1] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.
- [2] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–97, 2002.
- [3] Alex Arenas, Albert Díaz-Guilera, Jürgen Kurths, Yamir Moreno, and Changsong Zhou. Synchronization in complex networks. *Physics Reports*, 469(3):93–153, 2008.
- [4] Albert-László Barabási. *Network Science*. Cambridge University Press, 2016.
- [5] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.
- [6] Stefano Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang. Complex networks: Structure and dynamics. *Physics Reports*, 424(4–5):175–308, 2006.
- [7] Robin Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–350, 2002.
- [8] Vincenza Carchiolo, Michele Malgeri, and Giuseppe Nicosia. Dynamical clustering on social networks. In *2008 International Conference on Computing, Communication and Control Technologies*, pages 108–113, 2008.
- [9] Aaron Clauset, Cosma Rohilla Shalizi, and Mark E. J. Newman. Power-law distributions in empirical data. *SIAM Review*, 51(4):661–703, 2009. Relacionado con el estudio de PNAS 0601602103 sobre leyes de potencia.
- [10] Reinhard Diestel. *Graph Theory*. Springer, 5th edition, 2017.
- [11] David Easley and Jon Kleinberg. *Networks, Crowds, and Markets: Reasoning about a Highly Connected World*. Cambridge University Press, 2010. Disponible en: <https://www.cs.cornell.edu/home/kleinber/networks-book/>.
- [12] Paul Erdős and Alfréd Rényi. On random graphs. *Publicationes Mathematicae*, 6:290–297, 1959.

- [13] Ernesto Estrada. Combinatorial and geometric aspects of networks. In *The Structure of Complex Networks*, pages 51–72. Springer, 2011.
- [14] Ernesto Estrada. *The Structure of Complex Networks: Theory and Applications*. Oxford University Press, 2011.
- [15] Leonhard Euler. Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae*, 8:128–140, 1736.
- [16] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3–5):75–174, 2010.
- [17] Michelle Girvan and Mark E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.
- [18] Michelle Girvan and Mark E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences (PNAS)*, 99(12):7821–7826, 2002.
- [19] Ronald L. Graham, Bruce L. Rothschild, and Joel H. Spencer. *Ramsey Theory*. John Wiley Sons, 2nd edition, 1990.
- [20] G. Huecas and J. Salvachúa. Filtros colaborativos y sistemas de recomendación. Material docente / Universidad Politécnica de Madrid, 2024. Enfoque en algoritmos de recomendación basados en grafos.
- [21] William O. Kermack and Anderson G. McKendrick. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A*, 115(772):700–721, 1927.
- [22] Bernd Klein. Graphs in python: Analysis and implementation. <https://python-course.eu/applications-python/graphs-python.php>, 2024. Accedido: 8 de enero de 2026.
- [23] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
- [24] Yoshiki Kuramoto. Self-entrainment of a population of coupled non-linear oscillators. In *International symposium on mathematical problems in theoretical physics*, volume 39, pages 420–422, 1975.
- [25] Abdelmalik Moujahid and Blanca Cases. *Clustering basado en la sincronización*. Grupo de Inteligencia Computacional, Universidad del País Vasco (UPV/EHU), 2011. Curso 2010-2011.

- [26] Abdelmalik Moujahid and Blanca Cases. *Community Structure in Graphs*. Grupo de Inteligencia Computacional, Universidad del País Vasco (UPV/EHU), 2012. Curso 2011-2012.
- [27] Abdelmalik Moujahid and Blanca Cases. *The synchronized dynamics of complex networks*. Grupo de Inteligencia Computacional, Universidad del País Vasco (UPV/EHU), 2014. Curso 2013-2014.
- [28] Abdelmalik Moujahid and Blanca Cases. *Introduction to network theory and Community Structure in Complex Networks*. Universidad del País Vasco (UPV/EHU), March 2015.
- [29] Mark E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.
- [30] Mark E. J. Newman. *Networks: An Introduction*. Oxford University Press, 2010.
- [31] José David Núñez-González. *Computational Intelligent Methods for Trusting in Social Networks*. PhD thesis, University of the Basque Country (UPV/EHU), Donostia–San Sebastián, February 2016. Thesis dissertation supervised by Prof. Manuel Graña.
- [32] Romualdo Pastor-Satorras, Claudio Castellano, Piet Van Mieghem, and Alessandro Vespignani. Epidemic processes in complex networks. *Reviews of Modern Physics*, 87(3):925, 2015.
- [33] Louis M. Pecora and Thomas L. Carroll. Synchronization in chaotic systems. *Physical Review Letters*, 64(8):821, 1990.
- [34] Louis M. Pecora and Thomas L. Carroll. Master stability function for synchronized coupled systems. *Physical Review Letters*, 80(10):2109, 1998.
- [35] Mathew D. Penrose. *Random Geometric Graphs*. Oxford University Press, 2007.
- [36] Antoni Pérez-Navarro. Análisis de redes: de euler a la complejidad de las redes sociales. Repositorio institucional y ResearchGate, 2018. Universitat Oberta de Catalunya (UOC).
- [37] Paul Resnick and Hal R. Varian. *Recommender systems*. Cambridge University Press, 2002.
- [38] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation techniques. In *Proceedings of the 10th international conference on World Wide Web (WWW)*, pages 285–295, 2001.
- [39] Douglas Schmidt. Groupware: Coordinated activities in software development. *IEEE Software*, 11(3):19–27, 1994.

- [40] Steven H. Strogatz. From kuramoto to the continuous limit: a simple model for synchronization. *Physica D: Nonlinear Phenomena*, 143(1–4):1–20, 2000.
- [41] Amitabha Tripathi, Sushmita Venugopalan, and Douglas B. West. A short constructive proof of the erdős–gallai characterization of graphic lists. *Discrete Mathematics*, 310(4):843–844, 2010.
- [42] W.C. Tucker. Network analysis with r. <https://rpubs.com/wctucker/302110>, 2017. Accedido: 8 de enero de 2026.
- [43] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of ‘small-world’ networks. *Nature*, 393:440–442, 1998.