

INSTITUTO TECNOLÓGICO AUTÓNOMO DE MÉXICO



DETECCIÓN DE COMUNIDADES EN REDES A PARTIR DE
LA OPTIMIZACIÓN DE LA ESTABILIDAD DE PARTICIÓN

TESIS
QUE PARA OBTENER EL TÍTULO DE
LICENCIADO EN MATEMÁTICAS APLICADAS
PRESENTA
TANIA MENDOZA SMITH

ASESOR: DR. JOSÉ LUIS FARAH IBÁÑEZ

CIUDAD DE MÉXICO

2016

Autorización

Con fundamento en el artículo 21 y 27 de la Ley Federal del Derecho de Autor y como titular de los derechos moral y patrimonial de la obra titulada "Detección de Comunidades en redes a partir de la Optimización de la Estabilidad de Partición", otorgo de manera gratuita y permanente al Instituto Tecnológico Autónomo de México y a la Biblioteca Raúl Baillères Jr. autorización para que fijen la obra en cualquier medio, incluido el electrónico y la divulguen entre sus usuarios, profesores, estudiantes o terceras personas, sin que pueda percibir por la divulgación una contraprestación.

Tania Mendoza Smith

Fecha

Firma

Dedicado a mi familia, amigos y profesores

*Para mis papás, Laura y Héctor
y mis hermanos, Rodrigo y Héctorin.*

Agradecimientos

Con esta tesis concluyo la que hasta ahora ha sido la mejor etapa de mi vida, mi vida como estudiante del ITAM. Y quiero agradecer y dedicarles este logro a todos los que formaron parte de esta experiencia.

Primero que nada, agradezco a quienes han hecho posible que mi vida esté llena de tan grandes momentos, mis padres. Gracias mamá y papá por todo el esfuerzo que pusieron en mi educación, por el tiempo que dedicaron a enriquecer mi vida y por los sacrificios que eso supuso para ustedes. A mis hermanos por ser mis compañeros de vida; a Rodrigo por su ayuda e inspiración durante mi carrera y a Héctor por su preocupación, consejos y ejemplo. A mis abuelos, tíos y primos por la vida que hemos recorrido juntos.

Gracias a mi asesor, José Luis Farah, por su ayuda y la motivación para terminar este trabajo. Gracias por sus consejos, por sus clases, la paciencia que tuvo conmigo y por la enriquecedora experiencia de hablar y trabajar con él. Igualmente, gracias a mis sinodales Ana Paulina Figueroa, Edgar Possani y Ramón Espinosa por su tiempo en revisar este trabajo y sus valiosas aportaciones que mejoraron enormemente este trabajo.

Gracias a todos los profesores del ITAM cuyas enseñanzas enriquecieron mi formación académica y personal. Especialmente, agradezco a Rafael Gamboa, Ernesto Barrios, Javier Alfaro, Vladimir Caetano, Marcelo Mejía y Luis Felipe González. Del mismo modo, agradezco a los profesores que formaron parte de mi educación básica, quienes formaron quién soy como estudiante, mi gusto por las matemáticas y por aprender, gracias a Josefa

Aliseda, Virginia Villanueva y al Profesor Andrés Gutiérrez.

Muchísimas gracias a mis maestros fuera del salón de clases, quiénes me ayudaron con mi primer trabajo y a desarrollarme profesionalmente. Gracias a Guillermo Garduño, a Elmer Garduño y Rodrigo Fortes por tantas enseñanzas, a Mariano por su ayuda y a mis amigos, Sonia, Andrea, Yuriko, Areli, Sergio y Vila. Gracias también a Ger, Ro, Virus, Damián, Fer y Germán por este último año de trabajo.

A mis compañeros y amigos del ITAM, en especial a mis mejores amigas Crispin y Panda por haber hecho mi vida más feliz y divertida desde que las conocí hasta ahora y a mis amigos matemáticos, ingenieros, economistas, etc, por las fiestas, horas de biblioteca y por haber hecho esta etapa tan divertida, en especial a Aguado, Magda, Rafa, Daniel, Mercedes, Karen, Juan Solis, Mariana, Johnny, Joaquín, Zermeño, Juan, Aldo, Estebán, Juan Luis y Yuri.

A mis roomies y amigos foráneos, gracias por las mejores fiestas, comidas y viajes en la mejor casa, Hidalgo 92. Gracias a Erihvan, Anne, Francís, Alice, Téllez, Tere y Paquito. A mis amigos de toda la vida, Lucy, Sofía, Natalia, Fer, Rosana, Marcela, Mathieu, Rodrigo, Aguado, Ditter, Giovanni y Bob. Gracias por la amistad que ha perdurado todos estos años. Los quiero muchísimo.

A Aldo por ser mi mejor amigo y compañero todos estos años, por siempre apoyarme y quererme. Mi vida es mucho mejor gracias a ti.

Resumen

En este trabajo se explica un método de detección de comunidades, llamado ‘Optimización de la Estabilidad de Partición de una Gráfica’, introducido por Lambiotte,^[1] el cual se distingue de otros métodos de detección de comunidades por ser un método estocástico y por brindar la posibilidad de encontrar comunidades con diferente resolución gracias a su dependencia de un parámetro de tiempo. El método se basa en la simulación de un proceso de difusión sobre la gráfica, a tiempos bajos, se tienen muchas comunidades, empezando por el caso trivial de una comunidad por nodo, y al pasar el tiempo los nodos se van agrupando en cada vez menos comunidades, hasta llegar a la partición trivial de una sola comunidad.

Índice general

Página de Título	I
Autorización	I
Dedicatoria	II
Agradecimientos	III
Resumen	V
Índice	VI
I Introducción y Marco Teórico	1
1. Introducción	2
2. Detección de Comunidades	5
2.1. Notación y definiciones básicas	6
2.2. Conglomerados en redes	7
2.3. Definición de Comunidad	9
2.4. Algoritmos para detección de comunidades	11
2.4.1. Centralidad de intermediación	11
2.4.2. Modularidad	14
2.4.3. Desarrollo e interpretación de la Modularidad	16
3. Estabilidad	19
3.1. Caminatas Aleatorias	20
3.1.1. Caminatas Aleatorias en Gráficas	21
3.2. Difusión	24
3.2.1. El Laplaciano de la gráfica	25
3.2.2. La ecuación de difusión	27

3.2.3. Solución a la ecuación de difusión	28
3.3. Estabilidad a tiempo discreto	29
3.4. Estabilidad a tiempo continuo	33
3.5. Estabilidad y Modularidad	35
3.5.1. Límite de resolución	36
4. Optimización de estabilidad	40
4.1. Algorítmico de Louvain	40
4.2. Maximización de la estabilidad de partición para un tiempo de Markov	44
4.3. Variación de la Información	45
4.4. Heurística para detección de comunidades a partir de la estabilidad de partición	47
II Resultados y Conclusión	49
5. Resultados	50
5.1. Zachary's Karate Club	50
5.1.1. Características de la red	51
5.1.2. Resultados sobre Karate Club	51
5.2. Los Miserables	53
5.2.1. Resultados	54
5.3. Análisis por Tiempo de Markov	58
5.3.1. Once comunidades	59
5.3.2. Diez comunidades	60
5.3.3. Ocho comunidades	60
5.3.4. Siete comunidades	63
5.3.5. Seis comunidades y óptimo de modularidad	64
5.3.6. Cinco comunidades	64
5.3.7. Cuatro comunidades	65
5.3.8. Tres comunidades	68
6. Conclusiones	71
A. Software y Reproducibilidad	76
B. Notación	77

C. Detalles	79
C.1. Gráfica Aleatoria de Erdös Rènyi	79
C.2. Eigenvectores y el Teorema Espectral	80

Parte I

Introducción y Marco Teórico

Capítulo 1

Introducción

Este primer capítulo explica los objetivos que se persiguen a lo largo del trabajo así como la referencia a los capítulos dónde puede encontrarse cada una de las partes que lo componen.

Hay cuatro objetivos principales sobre los que se centra este trabajo. El primero, introducir el problema de partición de una red, y en específico, el de encontrar una partición natural en comunidades. El segundo, explicar algunos métodos existentes para resolver el problema de detección de comunidades. El tercero, explicar un enfoque estocástico de detección de comunidades introduciendo, primero, una medida de calidad estocástica, llamada *estabilidad* y posteriormente, una heurística para detectar comunidades a través de la optimización de esta medida. Finalmente, mostrar que los resultados obtenidos a partir de esta heurística son competitivos con los resultados obtenidos con otros métodos como la *optimización de modularidad*; mostrando además la ventaja de tener la capacidad de encontrar

particiones con diferente número de comunidades utilizando al tiempo como parámetro de resolución. Se explicará el problema de detectar comunidades en redes sin dar información a priori sobre el número de partes que se desea encontrar ni el tamaño de estas, diferenciando así este problema del problema de partición. Veremos que la mayoría de los métodos existentes para detección de comunidades se centran en características estructurales de la red y que el método sobre el que trata este trabajo, se enfoca en lo que la simulación de un proceso estocástico sobre la red puede decírnos sobre estas características. Para ello, definiremos y explicaremos el concepto de caminata aleatoria sobre una red así como el proceso de difusión sobre una red. El primero se utilizará para definir la medida de estabilidad a tiempo discreto, y el segundo para definir la medida a tiempo continuo.

Adicionalmente, se explica la relación entre la medida de *estabilidad*, y una de las medidas de calidad de particiones más utilizadas, la *modularidad*. Esto, porque al ser la maximización de *modularidad* un problema muy estudiado, la relación nos permitirá hacer uso de las heurísticas utilizadas para resolver la optimización de *modularidad* en la maximización de la *estabilidad*.

Para finalizar la parte teórica, se desarrolla la heurística que se usa en este trabajo para optimizar la *estabilidad*, así como una medida para elegir resoluciones relevantes, la ‘Variación de la Información’.

Finalmente, se presentan los resultados obtenidos al correr el algoritmo de ‘Optimización de Estabilidad’ sobre 2 redes. La primera, Karate Club, una de las redes más famosas utilizada para probar la validez de los métodos de detección de comunidades. La segunda, la red de coaparición de

personajes en la novela de Victor Hugo, ‘Los Miserables’, que nos permite ver el impacto de los óptimos encontrados a distintas resoluciones en una red social. Exploramos la partición a distintos tiempos de Markov, haciendo uso de la ‘Variación de Información’, para descubrir comunidades anidadas y distintas relaciones entre los personajes.

La tesis se divide en 5 capítulos, además de esta introducción. En el segundo capítulo se introduce el problema de detección de comunidades, así como algunos algoritmos para resolverlo. En el tercer capítulo se explica la medida de calidad de particiones que utilizaremos, la *estabilidad*, así como su relación con la *modularidad*. En el cuarto capítulo se explica el algoritmo que utilizaremos para optimizar la *estabilidad*. El quinto capítulo muestra los resultados de correr el algoritmo sobre las redes antes mencionadas. Y por último, en el sexto capítulo se presentan las conclusiones de este trabajo, así como otras posibles aplicaciones y se sugiere el trabajo futuro a realizar. Cabe mencionar, que hay 3 apéndices que incluyen: la lista de notación utilizada a lo largo del trabajo, el URL donde puede encontrarse el código, los datos y los resultados de esta tesis así como algunas definiciones y explicaciones que pudieran interesar al lector.

Capítulo 2

Detección de Comunidades

En este capítulo explicaremos la importancia del problema de detección de comunidades. Así mismo, se darán las definiciones básicas que se utilizarán a lo largo del trabajo. Posteriormente, se introducirá el problema de dividir los vértices de una gráfica en conglomerados, y en particular el problema de detección de comunidades, para lo que se muestran también algunos de los algoritmos más utilizados para resolverlo.

La cantidad de datos existentes y almacenados ha crecido enormemente en los últimos años, lo que trae el reto de extraer información relevante e inteligible para simplificar descripciones y obtener resultados. Las redes con datos ‘reales’ suelen ser sistemas irregulares, es decir, son objetos donde el orden y el desorden coexisten. Muchos de los datos que pueden representarse como redes se dividen de manera natural en grupos. Por ejemplo, en una red social, tenemos grupos de amigos; en una red bioquímica, tenemos unidades funcionales, y, en una red de compras tenemos productos similares. Podemos

modelar a estas redes para su análisis como gráficas o *grafos* (Definición 2.1) y encontrar así las subestructuras en las que se dividen naturalmente, las *comunidades*, que como su raíz etimológica nos dice, nos pueden facilitar la extracción de las características comunes de los datos.

2.1. Notación y definiciones básicas

Se hace este apartado con la finalidad de definir algunos de los conceptos que se utilizarán en la mayor parte del trabajo y aclarar la notación, que será uniforme a lo largo de todos los capítulos. Para facilitar la lectura, la lista completa de notación puede encontrarse una lista en el Apéndice B.

A lo largo del trabajo, denotaremos por G una gráfica no dirigida con n nodos y m aristas.

Definición 2.1. Se define a una gráfica G no dirigida como el par ordenado $G = (V, E)$, donde: $V = \{v_i | i = 1, \dots, n\}$ es un conjunto de vértices o nodos, y $E = \{e_k | k = 1, \dots, m\}$ es un conjunto de aristas entre estos nodos, de manera que $e_{ij} = v_i v_j$ se refiere al arista que existe entre el nodo v_i y el nodo v_j , sin dirección. Tal que, $|V| = n$ y $|E| = m$.

Las aristas a lo largo del trabajo estarán representados por la matriz de adyacencias (Definición 2.2) de G , denotada por A .

Definición 2.2. Sea G una gráfica no dirigida. Se define a la *matriz de adyacencias*, A , como la matriz de de tamaño $n \times n$ que denota la existencia de una arista e_{ij} entre un par de vértices v_i, v_j tal que la ij-ésima entrada

de la matriz A está dada por:

$$A_{ij} = \begin{cases} 1 & e_{ij} = v_i v_j \in E \\ 0 & e.o.c. \end{cases}$$

Como trabajaremos con G como gráfica no dirigida, esta matriz es simétrica, es decir $A_{ij} = A_{ji} \forall i, j$.

Definición 2.3. Sea G una gráfica no dirigida, se define el *grado de un vértice*, como el número de aristas que inciden con él. Se denota d_i al grado de v_i por su nombre en inglés (*degree*) y se puede calcular a partir de la matriz de adyacencias como:

$$d_i = \sum_{j=1}^n A_{ij}$$

Definiremos a d como el vector de grados $d = (d_1, d_2, \dots, d_n)$, y de la misma manera, a la matriz diagonal $D = diag(d)$ tal que

$$D_{ij} = \begin{cases} d_i & i = j \\ 0 & e.o.c. \end{cases}$$

Para evitar confusiones, las derivadas están denotadas por ∂ .

2.2. Conglomerados en redes

El problema de dividir los vértices de una red en conglomerados, en general, tiene dos grandes clases que dan lugar a distintos algoritmos. Newman^[2] se refiere a estos dos problemas como ‘*partición de una gráfica*’ y ‘*detección de comunidades*’.

El problema de partición de una gráfica se refiere al problema de partitionar los nodos de la gráfica en k partes mutuamente excluyentes, de forma tal que el número de aristas entre los grupos sea minimizado. En este problema, están dadas de antemano restricciones sobre el número de partes, k , y el tamaño de estas, n_k .

Por otro lado, el problema de detección de comunidades también busca dividir los nodos de la gráfica en conglomerados mutuamente excluyentes. Pero difiere del problema de partición en que no tiene definidas restricciones sobre el número de conglomerados ni el tamaño de estos de antemano; su meta es encontrar la manera ‘natural’ en la que una red se separa. Esta es la mayor diferencia entre ambos problemas, y la razón por la que las comunidades se entienden como las partes ‘naturales’ en las que se divide la gráfica, ya que queremos obtener información de la red sin tener que dar restricciones de antemano. Nos interesa saber tanto el número de conglomerados en los que se dividen los nodos de la red, como los miembros que conforman cada una de ellos. Cabe mencionar que existen formulaciones del problema de detección de comunidades dónde se permite que las comunidades se intersecten,^[3] en esta tesis la solución se piensa como una división exclusiva.

Es claro que el problema de partición está matemáticamente definido, mientras que el de comunidad no lo está, y es más bien la noción de algo que desearíamos encontrar en la red, como veremos en la siguiente sección. Newman^[2] distingue las soluciones de ambos problemas, reservando la palabra ‘partición’ para el primer problema, y la palabra ‘comunidad’ para el segundo. En este trabajo haremos lo mismo.

2.3. Definición de Comunidad

No hay una definición única y matemáticamente formal para lo que se entiende por comunidad, pero una de las definiciones más aceptadas fue dada por Newman^[2] y se refiere a la idea general de lo que deseamos encontrar en una comunidad y sería:

Las comunidades de una red son grupos de nodos que tienen una fuerte conexión entre ellos y poca o menor interacción con el resto de la red.^[2]

Para entender qué gráficas poseen esta estructura de comunidad que las divide de manera natural, nos servirá introducir el concepto de *gráfica aleatoria*. Erdős y Rényi, en 1959, introdujeron la *gráfica aleatoria* (Apéndice C). En ella, la existencia de una arista entre cualquier par de vértices v_i, v_j es equiprobable. Las aristas de la gráfica aleatoria resultante se reparten a lo largo de los nodos de manera casi homogénea, ya que como su distribución de grados es binomial, la mayoría de los vértices tendrán un grado similar, dado por la esperanza de esta binomial.

En realidad, las gráficas que modelan datos reales no suelen ser así de homogéneas, por lo que las gráficas aleatorias de Erdős y Rényi no son buenas candidatas para modelar, como lo ilustra la Figura 2.1. La distribución de grados de las redes con datos reales, suele tener una cola que sigue una ley de potencias,^[4] lo que significa que muchos nodos con grado bajo coexisten con otros de grado alto. Más aún, la distribución de grados es localmente no-homogénea, con altas concentraciones de aristas en algunos grupos de vértices y poca concentración entre estos grupos. A esta carac-

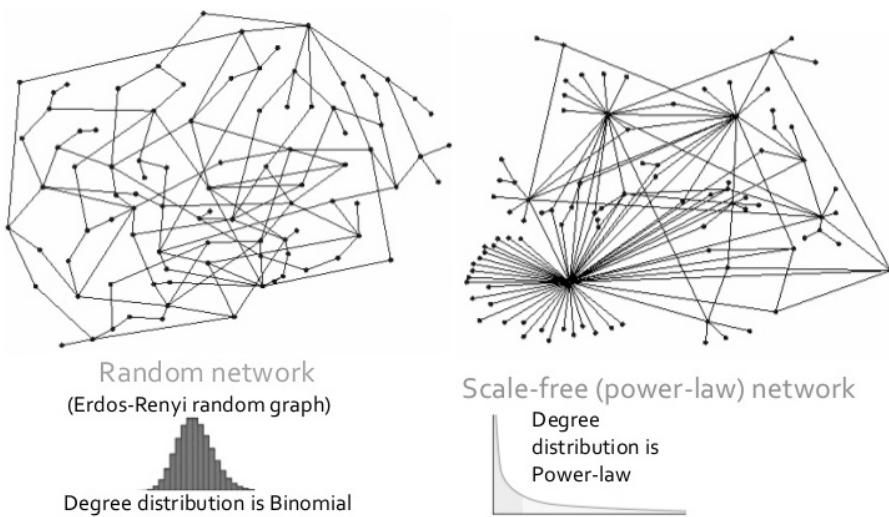


Figura 2.1: **Gráfica Aleatoria vs Red libre de escala:** La red libre de escala se ha tomado como un mejor modelo de aproximación a redes reales, en particular de redes sociales. - Cornell University. Poisson Vs Scale Free Network. 2009. Web.

terística, Girvan y Newman,^[2] le llamaron *estructura de comunidad* , y es una propiedad que las gráficas aleatorias no poseen. Es claro que esta no es una definición matemáticamente formal, ya que no queda definido qué entendemos por ‘poca’ o ‘mucha’ concentración, pero sirve para introducir la noción de cómo podríamos desarrollar algoritmos para encontrar los grupos con alta concentración y las aristas que los conectan.

Esta ambigüedad de definiciones, y la amplia variedad de problemas que existen, ha resultado en el desarrollo de numerosas heurísticas, medidas y métodos para la detección de comunidades, donde cada uno está basado en una noción particular de comunidad. Cada método utiliza aproximaciones distintas que capturan las propiedades que se desean detectar, por lo

que cada uno nos llevará a resultados distintos. A continuación explicamos algunos de estos métodos y las ideas en las que se basan.

2.4. Algoritmos para detección de comunidades

Algunos de los algoritmos de detección de comunidades más populares se basan en una de dos aproximaciones; *conglomerados jerárquicos* u *optimización de particiones*. Los algoritmos de conglomerados jerárquicos, inician considerando a cada nodo en su propia, y en cada iteración, se van uniendo comunidades usando alguna medida de similaridad de comunidades. Para los algoritmos de optimización de particiones, se inicia con un única comunidad que agrupa a todos los nodos y se va dividiendo hasta que se cumpla cierto criterio de optimalidad o medida de bondad.

En cualquiera de las dos aproximaciones, hay que decidir si los criterios de decisión van a estar basados en propiedades locales o globales de la red. Los métodos locales, están definidos por la estructura al rededor de un nodo o una subgráfica, sin tomar en cuenta al resto de la red, por ejemplo, sus adyacencias. Por otro lado, los métodos globales consideran a toda la red, por ejemplo, los caminos entre nodos.

2.4.1. Centralidad de intermediación

Para motivar la intuición del algoritmo de ‘centralidad de intermediación’, consideremos una gráfica y su partición natural en comunidades bajo la idea de comunidad presentada en la Sección 2.3. Si suponemos que existe esta partición natural para una gráfica, tenemos que existen aristas entre

nodos miembros de una misma comunidad y aristas entre nodos de distintas comunidades. Si lográramos eliminar todas las aristas que conectan nodos de distintas comunidades, nos quedaríamos con una red dividida en componentes conexas, en la que podemos entender cada componente conexa como una comunidad de esta partición natural. La Figura 2.2 ejemplifica esta idea, donde cada una de las subgráficas dentro de los círculos representan una comunidad.

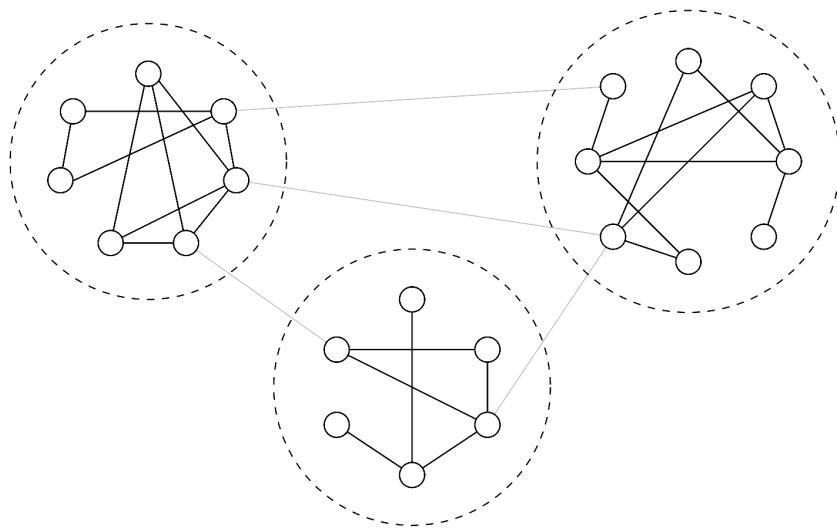


Figura 2.2: Ejemplo de una gráfica con comunidades subyacentes. Existen 3 comunidades, encerradas en círculos. Si quitáramos las aristas que conectan a los círculos, quedaría 3 componentes conexas; las comunidades.

Siguiendo la intuición del párrafo anterior, bastaría encontrar estas aristas entre comunidades para tener un algoritmo de detección de comuni-

dades. Una vez más, Girvan y Newman, presentaron un algoritmo basado en la *centralidad de intermediación* que sigue esta idea de eliminación.

Definición 2.4. Un *camino* de longitud n en una gráfica no dirigida, G es una secuencia de vértices $(v_1, v_2, \dots, v_{n+1})$ tal que v_i es adyacente a v_{i+1} para todo $i = 1, \dots, n$

Definición 2.5. El *camino geodésico* o camino más corto entre un par de vértices cuales quiera v_i, v_j es un camino entre ellos, tal que no existe ningún otro camino de longitud menor.

Definición 2.6. La *centralidad de intermediación*, b , por su nombre en inglés, *betweenness centrality* de una arista en una red es el número de caminos geodésicos en la red que pasan por dicha arista.

Algoritmo 1: Algoritmo de Centralidad

Input: Sea G una gráfica con n nodos y m aristas

- 1 Calcular la centralidad de intermediación $b_i \forall i = 1, 2, \dots, m$
 - 2 Eliminar la arista con la mayor centralidad b^*
 - 3 Con la nueva gráfica, recalcular las centralidades.
 - 4 Repetir hasta que no queden aristas
-

Este es un algoritmo divisivo, de conglomerados jerárquicos. A partir de alguna iteración, la red habrá perdido suficientes aristas para dividirse en dos componentes conexas, que consideramos como comunidades, y al seguir iterando, se irá dividiendo cada vez en más componentes hasta llegar a tener una red sin aristas donde cada nodo está en su propia comunidad. Estas subdivisiones pueden representarse con un dendograma, y uno puede elegir partir en el punto que crea más conveniente.

La razón de utilizar la centralidad de intermediación como medida para eliminar aristas es que, si consideramos comunidades que solo se conectan unas con otras a través de pocas aristas, en comparación con el número de aristas que existen dentro de la comunidad, todos los caminos más cortos entre nodos de distintas comunidades tienen que pasar por alguna de estas aristas. Es decir, las aristas que conectan comunidades de una gráfica como esta, tienen alta centralidad de intermediación.

La idea detrás de este algoritmo es muy convincente, sin embargo, es claro que el cálculo de la centralidad es muy caro, ya que hay que calcular todos los caminos geodésicos para cada par de nodos y recalcularlo en cada iteración. Por lo que resulta muy lento para redes muy grandes. Además, ha probado dar resultados poco convincentes para redes muy densas, ya que la diferencia entre la centralidad de aristas entre nodos de distintas comunidades y las aristas entre nodos de la misma comunidad es más baja para estas redes.

2.4.2. Modularidad

Uno de los métodos de detección de comunidades más populares se basa en la medida de calidad de particiones formulada por Girvan y Newman, la *modularidad* definida más adelante (Definición 2.9). El objetivo de esta medida es cuantificar qué tan buena es una partición, y así, poder compararla con otra, para decidir cuál es la mejor o para decidir si una gráfica tiene estructura de comunidad. Esta medida nos interesa, pues además de utilizarse para calificar particiones, puede maximizarse con la finalidad de encontrar comunidades.

Definición 2.7. Sea G una gráfica, con n nodos divididos en p_1, \dots, p_k partes. El *tamaño de corte*, R , de la partición se define como el número de aristas entre nodos v_i, v_j tales que $v_i \in p_r$ y $v_j \in p_s$, $r \neq s$. Es decir, el número de aristas que existen entre las partes.

Para entender la modularidad, recordemos el tamaño de corte (Definición 2.7). El algoritmo de partición de Kernighan-Lin^[5] se basa en la búsqueda de una partición con corte mínimo. Es decir, se basa en la idea de que una buena división es aquella en la que existen pocas aristas entre comunidades. La modularidad va un paso más allá, y dice que una buena división es aquella en la que existen menos aristas entre comunidades de las que se esperarían si las conexiones fueran aleatorias siguiendo a un modelo nulo.

Definición 2.8. Sea G una gráfica no dirigida. El modelo nulo de Newman-Girvan para G está dado por:

$$P(v_i v_j \in E) = \frac{d_i d_j}{2m}$$

El modelo nulo es una gráfica aleatoria que puede o no mantener algunas de las propiedades estructurales de la gráfica original. Dependiendo de la elección del modelo nulo, se obtienen diferentes ecuaciones de modularidad. La elección del modelo nulo es muy importante,^[6] pues repercute en los resultados. Una de las ideas más simples e ingenuas para el modelo nulo es asignar las probabilidades de manera uniforme, es decir, que para cualquier par de nodos v_i, v_j la existencia de una arista sea equiprobable (modelo de Erdős y Réyni - Apéndice C). Esta no es la mejor opción, ya

que se ha probado que este modelo no lleva a buenos resultados debido a que la distribución de grados de la gráfica resultante del modelo nulo no se asemejan a la gráfica original. Por ello, es preferible tomar un modelo nulo que replique las características estructurales de la gráfica original; esta es la práctica común en la literatura.^[6] El modelo nulo normalmente elegido para la modularidad, es el modelo nulo de Newman-Girvan (Definición 2.8), el cual genera una gráfica aleatoria que preserva la distribución de grados de la gráfica original.

De este modo, la posible existencia de comunidades se revela comparando la densidad de aristas en una subgráfica y la densidad que uno esperaría tener en dicha subgráfica si las adyacencias existieran sin importar la estructura de comunidad.

Definición 2.9. Dada una partición en n_c comunidades de los nodos de G , sea c_i la comunidad a la que pertenece el nodo v_i . La modularidad de Newman-Girvan, denotada por Q está dada por:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{d_i d_j}{2m}) \delta(c_i, c_j) \quad (2.1)$$

Donde una vez más, A es la matriz de adyacencias de G ,

A continuación, se explica la obtención de la ecuación de ‘modularidad de Newman-Girvan’, que será la utilizada en el resto del trabajo, a partir del modelo nulo de la Definición 2.8 y su significado.

2.4.3. Desarrollo e interpretación de la Modularidad

La ecuación de Modularidad de la Definición 2.9 se deriva como sigue:

Sea G una gráfica no dirigida (Definición 2.1) y consideremos una partición en n_c comunidades. El entero c_i denota la comunidad a la que pertenece el vértice v_i .

Entonces, el número total de aristas que existen entre vértices miembros de la misma comunidad está dado por:

$$\sum_{v_i, v_j \in E} \delta(c_i, c_j) = \frac{1}{2} \sum_{\forall ij} A_{ij} \delta(c_i, c_j) \quad (2.2)$$

Dónde $\delta(c_i, c_j)$ es la Delta de Kronecker, que es 1 si $c_i = c_j$ y 0 e.o.c.

Para calcular las aristas esperadas bajo el modelo nulo, consideraremos un par de nodos v_i y v_j . La probabilidad de que exista una arista entre ellos es proporcional al grado de ambos nodos, d_i y d_j respectivamente. Por definición, existen $2m$ incidencias de aristas en toda la gráfica (ya que cada arista incide con 2 nodos y existen m aristas),

Entonces, la probabilidad de adyacencia entre cualquier par de vértices v_i, v_j está dada por $\frac{d_i d_j}{2m}$

Así, obtenemos la expresión

$$\frac{1}{2} \sum_{ij} \frac{d_i d_j}{2m} \delta(c_i, c_j) \quad (2.3)$$

Haciendo la resta, obtenemos la modularidad:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{d_i d_j}{2m}) \delta(c_i, c_j) \quad \square \quad (2.4)$$

La intuición es que entre más exceda una comunidad el número de aristas internas al número esperado de aristas, mejor definida estará la comunidad. Así que, valores grandes de modularidad indican buenas particiones.

La modularidad siempre es menor a 1, pero puede ser negativa, lo que nos indicaría que no existen particiones con modularidad positiva, es decir, que la gráfica no tiene estructura de comunidad.

Es importante notar que el máximo de la modularidad crece si el tamaño de la gráfica o el número de comunidades bien separadas crecen. Por ello, la modularidad no debe emplearse para comparar la estructura de comunidad entre redes con tamaños muy distintos.

Se ha demostrado que el problema de optimización de la modularidad es NP-completo,^[7,8] por lo que está justificado el empleo de heurísticas para acercarse a una buena solución. Hay numerosas opciones para la optimización de la modularidad, una de ellas, y la más utilizada por su rapidez y la calidad de sus resultados es el algoritmo de *Louvain*, que será discutido en el capítulo siguiente.

Las comunidades resultantes de la optimización de la modularidad, en general son muy buenas. Sin embargo, se ha mostrado que esta medida tiene un límite de resolución, es decir, que aún en casos donde la división resulta muy clara, la modularidad devuelve particiones de tamaño balanceado, y tiende a agrupar más de lo necesario. Este límite de resolución se explica con más detalle en el Capítulo 3. Nos interesaba explicar esta medida, ya que el método de optimización de estabilidad busca romper este límite de resolución, además de beneficiarse de su relación con la modularidad para formar heurísticas de optimización. Esto también se verá en el Capítulo 3.

Capítulo 3

Estabilidad

El objetivo de este capítulo es definir la medida de calidad de particiones llamada ‘estabilidad’ la cual ha sido definida tanto a tiempo discreto como a tiempo continuo. Para ello, se necesita definir antes la caminata aleatoria sobre una gráfica (Definición 3.4) así como el proceso de difusión sobre una gráfica (Definición 3.6). Una vez definida la medida de calidad, se explica su relación con la ‘modularidad’, que se definió en el capítulo anterior (Definición 2.9). Esto, porque la relación entre estas dos medidas nos servirá para el algoritmo de optimización de estabilidad que se explica en el capítulo siguiente.

La mayor parte del trabajo hecho en el problema de detección de comunidades está basado en propiedades estructurales de la gráfica. La ‘estabilidad’, introducida por Lambiotte,^[1] es una medida de calidad basada en las propiedades estadísticas de un proceso dinámico que toma lugar en la gráfica. Una de las ideas centrales de esta medida es que, procesos dinámi-

cos, como una difusión, se ven afectados por la topología de la gráfica; por ejemplo, nodos con grado alto tienen mayor probabilidad de ser visitados varias veces por una caminata aleatoria y componentes interconectadas tienen probabilidad alta de retener un flujo por un tiempo mayor que componentes con baja interconexión. De esta manera, observar procesos dinámicos sobre una gráfica, pueden ayudarnos a extraer información de la topología de la red.^[8]

En este capítulo, se pretende explicar la medida de ‘estabilidad de una gráfica’, por lo que se explican también los procesos dinámicos en los que se basa.

3.1. Caminatas Aleatorias

Uno de los procesos dinámicos que podemos considerar en una gráfica, es una *caminata aleatoria*.

Una caminata aleatoria es la formalización matemática de un proceso que inicia en un punto y se mueve de manera aleatoria hacia otro. Algunas de las cosas que pueden modelarse como caminatas aleatorias serían: la trayectoria de una partícula o de un gas, los precios de una acción en la bolsa de valores, etcétera. Las caminatas aleatorias pueden ser muy complejas, e incluso existir en varias dimensiones, pero nos enfocaremos a las caminatas unidimensionales. A una caminata de este tipo, se le conoce también como una ‘cadena de Markov’.

Definición 3.1. Una *cadena de Markov* es una secuencia de variables aleatorias X_1, X_2, X_3, \dots que cumplen con la propiedad de Markov.

$$P(X_{n+1} = x \mid X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) = P(X_{n+1} = x \mid X_n = x_n)$$

Definición 3.2. Dada una cadena de Markov, se define al *espacio de estados*, S , que es el conjunto finito o numerable formado por los posibles valores de X_i .

En palabras, la propiedad de Markov se refiere a que, dado el estado presente, el futuro y el pasado son independientes.

Definición 3.3. Sea X_t una cadena de Markov sobre el espacio de estados S . Se define a la *matriz estocástica* o *de transición*, M , como la matriz que describe la probabilidad de pasar de el estado i al estado j en un paso. Tal que $M_{ij} = P(X_t = j \mid X_{t-1} = i)$

Haciendo uso de estas definiciones, podemos pensar a una cadena de Markov como una caminata aleatoria sobre una red dirigida,^[2] dónde las aristas de la gráfica están marcadas con las probabilidades de pasar de un estado a otro y el espacio de estados S está dado por los nodos, como definiremos a continuación.

3.1.1. Caminatas Aleatorias en Gráficas

Una caminata aleatoria sobre una gráfica,^[9] inicia en un vértice dado y en cada paso de la caminata, el caminante ‘elige’ de manera uniforme entre todas las aristas adyacentes al vértice actual la arista sobre la cuál se va a mover. Al llegar al otro nodo adyacente, vuelve a elegir y así sucesivamente. En nuestro caso, permitiremos al caminante pasar por las mismas aristas y vértices más de una vez, y regresar sobre una arista recién cruzada.

Definición 3.4. Dada una gráfica no dirigida, G , como la definida en (Definición 2.1) A la cadena de Markov a tiempo discreto X con espacio de estados S y matriz de transición de un paso dada por: $M = AD^{-1}$ la definimos como la *caminata aleatoria sobre G* . Donde A es la matriz de adyacencias de G y D la matriz diagonal de grados.

La matriz de transición de la caminata aleatoria sobre la gráfica no necesariamente tiene que estar dada por AD^{-1} , sin embargo, se toma esta matriz de transición pues es la que asigna la probabilidad de salto de manera uniforme entre aristas adyacentes; tal que, estando en v_i , la probabilidad de brincar a cualquiera de sus d_i aristas adyacentes está dada por $\frac{1}{d_i}$ donde d_i es el grado de v_i .

Vista por entradas, haciendo uso de (Definición 3.3), tenemos que

$$M_{ij} = P(X_t = j | X_{t-1} = i) = \frac{A_{ij}}{d_i} \quad (3.1)$$

La matriz de adyacencias A (Definición 2.2), hace que $M_{ij} = 0$ cuando v_i y v_j no están conectados por una arista.

Ahora, buscamos obtener la ecuación de la dinámica de esta caminata. Para obtenerla, consideremos que al tiempo t que la caminata se encuentra sobre el vértice v_i . Denotamos como $p_i(t)$ la probabilidad de que la caminante se encuentre en v_i al tiempo t . Calculemos entonces la probabilidad de que esto sucediera:

$$p_i(t) = \sum_j \frac{A_{ji}}{d_j} p_j(t-1)$$

Escrito de forma matricial, para todos los nodos, tenemos:

$$p(t) = AD^{-1}p(t-1) \quad (3.2)$$

donde $p(t) = (p_1(t), \dots, p_n(t))$ es el vector de probabilidades al tiempo t , $p(t - 1)$ el del paso anterior y AD^{-1} la matriz de transición.

Dada esta dinámica, encontremos la *distribución estacionaria* para esta cadena de Markov.

Definición 3.5. La *distribución estacionaria* π es un vector (π_1, \dots, π_n) tal que $\sum_{i=1}^n \pi_i = 1$ y que permanece invariante a la operación de la matriz de transición M , es decir $\pi M = \pi$

En una cadena de Markov, la distribución estacionaria está dada por la distribución límite, es decir, la distribución que se encuentra cuando t tiende a ∞ . A continuación, desarrollamos el proceso para encontrar esta distribución estacionaria π .

Proposición 3.1. La distribución estacionaria para la cadena de Markov de la ecuación 3.2 está dada por

$$\pi = \frac{d^T}{2m} \quad (3.3)$$

donde $d = (d_1, d_2, \dots, d_n)$ es el vector de grados.

Demostración 3.1. Hagamos $t \rightarrow \infty$, entonces

$$\pi_i = p_i(\infty) = \sum_{j=1}^n \frac{A_{ij}}{d_j} p_j(\infty)$$

Escrito de forma matricial, tenemos:

$$p(\infty) = AD^{-1}p(\infty)$$

Llamaremos L a la resta $D - A$, que definiremos más adelante (Definición 3.7). Desarrollando la ecuación anterior y reescribiendo $p(\infty) = p_\infty$,

$$p_\infty - AD^{-1}p_\infty = (I - AD^{-1})p_\infty = (D - A)D^{-1}p_\infty = LD^{-1}p_\infty = 0$$

Entonces, tenemos que el vector $D^{-1}p$ es un eigenvector de L con $\lambda = 0$.

En una gráfica con una sola componente, demostraremos más adelante (Demostración 3.2) que el único eigenvector de L con eigenvalor $\lambda = 0$ es aquel con todas sus componentes iguales. Haciendo uso de esta proposición, tenemos,

$D^{-1}p_\infty = \alpha\mathbf{1}$ con $\alpha \in \mathbb{R}$ y $\mathbf{1} = (1, \dots, 1)$ el vector de unos de tamaño n .

Entonces $p_\infty = \alpha D\mathbf{1}$ y, visto entrada a entrada, tenemos que $\Rightarrow p_i(\infty) = \alpha d_i$

Esto significa que la probabilidad de que una caminata se encuentre en v_i en el límite $t \rightarrow \infty$ es proporcional al grado del nodo, d_i . Si normalizamos este vector, para que el vector $p(\infty)$ sea tal que $\sum_{i=1}^n p_i(\infty) = 1$, obtenemos:

$$p_i(\infty) = \frac{\alpha d_i}{\alpha \sum_j d_j} = \frac{d_i}{\sum_j d_j} = \frac{d_i}{2m}$$

Con lo que finalmente, tenemos que:

$$\pi = \frac{d^T}{2m} \quad \square$$

Este resultado obedece a la intuición mencionada anteriormente de que es más probable visitar vértices con grado alto, ya que existen más maneras de llegar a ellos.

3.2. Difusión

Una difusión es un proceso por el que algún fluido, por ejemplo un gas o un líquido se mueven. En el caso de un gas, este movimiento sería efecto de

la presión que lleva al gas de regiones de baja densidad a regiones de alta densidad. También podemos verlo cuando una gota de tinta colorea el agua de un recipiente.

Matemáticamente, un ‘proceso de difusión’ es un proceso de Markov con trayectorias continuas, es decir, un proceso de Markov sin saltos. Podemos entonces, definir una difusión en una gráfica, y podemos pensarlo como una sustancia moviéndose hacia los nodos a través de las aristas.

Definición 3.6. Sea G una gráfica, se define como *proceso de difusión* en la gráfica al movimiento de una sustancia entre los vértices a través de las aristas a una razón de $C(\phi_j - \phi_i)$ donde ϕ_i es la concentración de la sustancia en v_i y C la *constante de difusión*

Para continuar con el desarrollo del proceso de difusión, necesitamos introducir formalmente al Laplaciano de la gráfica, L

3.2.1. El Laplaciano de la gráfica

Definición 3.7. El *Laplaciano* de la gráfica es una matriz simétrica dada por $L = D - A$. Que por elementos, se escribe:

$$L_{ij} = \begin{cases} d_i & i = j \\ -1 & i \neq j \text{ y } v_i v_j \in E \\ 0 & e.o.c. \end{cases}$$

La matriz L fue llamada Laplaciano porque la ecuación de difusión (Ecuación 3.4) tiene la forma de la ecuación ordinaria de difusión de un gas, y en el lugar del operador Laplaciano, ∇^2 , tenemos a L .

La matriz L tiene muchas implicaciones importantes, ahora demostraremos una que fue utilizada en la Demostración 3.1.

Proposición 3.2. La matriz L siempre tiene al menos un eigenvalor $\lambda = 0$

Demostración 3.2. Sea $x = \mathbf{1} = (1, 1, \dots, 1)$. Queremos ver que este vector sea el eigenvector correspondiente a $\lambda = 0$.

En otras palabras, queremos ver para que λ se cumple la igualdad $Lx = \lambda x$

Si observamos el vector resultante componente a componente, tenemos

$$(L\mathbf{1})_i = \sum_{j=1}^n L_{ij} \times 1 = \sum_{j=1}^n (\delta_{ij}d_i - A_{ij}) = d_i - \sum_{j=1}^n A_{ij} = d_i - d_i = 0$$

$$\text{Entonces } L\mathbf{1} = 0$$

Por lo que, $\mathbf{1}$ siempre es un eigenvector de L con $\lambda = 0$ ■

Además de este resultado, aunque no se demostrarán, los siguientes resultados sobre L también son importantes:^[2]

- L sólo tiene eigenvalores reales no negativos.
- L no tiene inversa.
- L es singular.
- El número de eigenvalores 0 de L es igual a el número de componentes de la gráfica.
- La solución a la ecuación de difusión de cualquier gráfica contiene únicamente exponenciales decrecientes, por lo que la solución tiende a un equilibrio en el límite $t \rightarrow \infty$.

3.2.2. La ecuación de difusión

Proposición 3.3. La ecuación de la dinámica para el proceso de difusión en la gráfica (Definición 3.6) está dada por:

$$\frac{\partial \phi}{\partial t} + CL\phi = 0 \quad (3.4)$$

Demostración 3.3. Al ocurrir el proceso de difusión (Definición 3.6) sobre G , en un pequeño intervalo de tiempo, la cantidad de fluido que atraviesa de v_j a v_i está dada por $C(\phi_j - \phi_i)\partial t$. Y la razón con la que cambia la cantidad de sustancia en el nodo v_i está dada por :

$$\begin{aligned} \frac{\partial \phi_i}{\partial t} &= C \sum_j A_{ij}(\phi_j - \phi_i) \\ \Rightarrow \frac{\partial \phi_i}{\partial t} &= C \sum_j A_{ij}(\phi_j - \phi_i) \\ &= C \sum_j A_{ij}\phi_j - C\phi_i \sum_j A_{ij} \\ &= C \sum_j A_{ij}\phi_j - C\phi_i d_i \end{aligned} \quad (3.5)$$

Entonces la ecuación 3.5, puede escribirse en forma matricial como :

$$\frac{\partial \phi}{\partial t} = C(A - D)\phi \quad (3.6)$$

Donde ϕ es el vector que tiene por elementos (ϕ_1, \dots, ϕ_n) .

Sustituyendo la matriz L de (Definición 3.7) en la ecuación 3.6 tenemos que

$$\frac{\partial \phi}{\partial t} + CL\phi = 0 \quad \blacksquare$$

3.2.3. Solución a la ecuación de difusión

Ecuación 3.4 : $\frac{\partial \phi}{\partial t} + CL\phi = 0$

Podemos resolver la ecuación diferencial con ayuda de las propiedades de L como sigue:

Escribimos al vector ϕ como combinación lineal de los eigenvectores de L , $\{x_1, \dots, x_n\}$ dónde la dependencia temporal del vector estará dada por los coeficientes $\{\alpha_1(t), \dots, \alpha_n(t)\}$ como sigue:

$$\phi(t) = \sum_{i=1}^n \alpha_i(t)x_i$$

Sustituyendo en la ecuación (3.4) y haciendo uso de la definición de eigenvectores $Lx_i = \lambda_i x_i$ tenemos,

$$\begin{aligned} & \frac{\partial \phi}{\partial t} + CL\phi \\ &= \frac{\partial \sum_i \alpha_i(t)x_i}{\partial t} + CL \sum_{i=1}^n \alpha_i(t)x_i \\ &= \sum_{i=1}^n \left[\frac{\partial \alpha_i(t)x_i}{\partial t} + CL\alpha_i(t)x_i \right] \\ &= \sum_{i=1}^n \left[\frac{\partial \alpha_i(t)x_i}{\partial t} + C\alpha_i(t)\lambda_i x_i \right] \\ &= \sum_{i=1}^n \left[\frac{\partial \alpha_i(t)}{\partial t} + C\alpha_i(t)\lambda_i \right] x_i = 0 \end{aligned} \tag{3.7}$$

Como L es simétrica, por el Teorema Espectral (Apéndice C.2), puede ser diagonalizada por una matriz ortogonal de eigenvectores. Podemos tomar a $\{x_i, \dots, x_n\}$ como una base ortonormal de eigenvectores de L .

Sea $\beta_i = \frac{\partial \alpha_i(t)}{\partial t} + C\alpha_i(t)\lambda_i$, entonces, haciendo uso de que $\{x_i, \dots, x_n\}$ está formada por vectores linealmente independientes y sustituyendo en la

ecuación 3.7, tenemos que,

$$\sum_{i=1}^n \beta_i x_i = 0 \text{ si solo si tenemos que } \beta_i = 0 \quad \forall i$$

$$\text{Entonces, } \beta_i = \frac{\partial \alpha_i}{\partial t} + C\alpha_i \lambda_i = 0 \quad \forall i$$

$$\text{Por lo tanto, } \frac{\partial \alpha_i}{\partial t} = -C\alpha_i \lambda_i \quad \forall i$$

$$\text{Integrando, } \int \frac{1}{\alpha_i} \partial \alpha_i = \int -C\lambda_i \partial t \quad \forall i$$

$$\text{Resolviendo, } \ln |\alpha_i| = \int -C\lambda_i \partial t \quad \forall i$$

$$\text{Despejando, } |\alpha_i| = e^{-C\lambda_i t} \quad \forall i$$

Si tenemos la condición inicial $\alpha_i(0)$, obtenemos la ecuación:

$$\alpha_i(t) = \alpha_i(0)e^{-C\lambda_i t} \quad \forall i \tag{3.8}$$

Esta ecuación nos será útil más adelante para definir la estabilidad a tiempo continuo (Definición 3.11).

3.3. Estabilidad a tiempo discreto

El método de detección de comunidades utilizado en esta tesis será la *optimización de la estabilidad a tiempo continuo*, pero la primera definición de esta medida de calidad fue a tiempo discreto, por lo que iniciaremos el desarrollo de la medida de esta manera. Como mencionamos antes, existe una relación entre una gráfica y una cadena de Markov, dada por el hecho de que a cualquier gráfica puede asociársele una caminata aleatoria. Esta relación nos da una interpretación de comunidad de manera dinámica:

La partición natural de la gráfica en comunidades a un tiempo dado corresponde a los atractores del sistema dinámico, es decir, a los espacios de estados de los que el escape es poco probable dada la escala de tiempo.^[10]

Para establecer esta correspondencia, se utiliza la medida de ‘autocovarianza de conglomerados del proceso de Markov’, que mide qué tan persistente es una partición en el tiempo. Definiremos esta medida a continuación.

Consideremos una caminata aleatoria sobre G como la discutida en la sección anterior dada por la ecuación (3.2) y con la distribución estacionaria que encontramos, dada por la ecuación (3.3). Hay que recordar que en una caminata de este tipo, la matriz de transición a t pasos está dada por M^t .

Definición 3.8. Sea $X(t)$ un vector aleatorio que describe la difusión de la partícula, es decir, $X_i(t) = 1$ si la partícula está en v_i al tiempo t y 0 en otro caso. Ahora, definamos la *Autocovarianza* de este vector aleatorio:

$$W(t) = cov[X(\tau), X(\tau + t)] = \Pi M^t - \pi^T \pi \quad (3.9)$$

Donde M^t es la matriz de transición (Ecuación 3.1) de t pasos, π es el vector de la distribución estacionaria (Definición 3.3) y la matriz $\Pi = diag(\pi)$ es una matriz diagonal.

Si observamos a W entrada a entrada, vemos que describe la probabilidad de transferencia entre nodos:

$$W_{ij} = P(\text{empezar en } v_i \text{ y terminar en } v_j \text{ después de } t \text{ pasos}) -$$

$$P(\text{Dos caminatas estén en } v_i \text{ y en } v_j \text{ evaluados estacionariamente})$$

Consideremos ahora, una partición exclusiva de los nodos (un nodo en una única comunidad) con c comunidades, y guardemos esta partición en la matriz H de tamaño $n \times c$ con entradas

$$H_{ij} = \begin{cases} 1 & v_i \in c_j \\ 0 & e.o.c. \end{cases}$$

Definimos ahora al vector aleatorio $Y(t)$ que describe la difusión de la partícula en términos de las comunidades, es decir que $Y_k(t) = 1$ si la partícula está en la comunidad k al tiempo t y 0 en otro caso, y está dado por $Y(t) = H^T X(t)$.

Definición 3.9. Sea H la matriz de partición en comunidades y $Y(t)$ el vector aleatorio de la difusión de una partícula en las comunidades dadas por H . Se define la *matriz de autocovarianza de conglomerados* como:

$$\begin{aligned} R(t) &= cov[Y(\tau), Y(\tau + t)] = H^T cov[X(\tau), X(\tau + t)] H \\ &= H^T [\Pi P(t) - \pi^T \pi] H \end{aligned} \quad (3.10)$$

Con el desarrollo de la Definición 3.9, tenemos una interpretación equivalente a la de la matriz W , pero describe la transferencia entre comunidades, en vez de entre nodos.

La idea de una buena partición, sería que hubiera una alta probabilidad de mantenerse en la misma comunidad a través del tiempo,^[10] es decir, que los elementos diagonales $R(t)_{ii}$ fueran más grandes que los elementos $R(t)_{ij}$ con $i \neq j$. Esta idea nos lleva a la definición de ‘estabilidad a tiempo discreto’:

Definición 3.10. Sea H la matriz de partición en comunidades y $R(s)$ la matriz de autocovarianza de conglomerados al tiempo s . Se define la estabilidad de la partición H al tiempo t como,

$$r(t; H) = \min_{0 \leq s \leq t} \sum_{i=1}^c R(s)_{ii} = \min_{0 \leq s \leq t} \text{traza}[R(s)] \quad (3.11)$$

El motivo de agregar la minimización sobre el intervalo de tiempo $[0, t]$ es que se desea asignar a la estabilidad un valor grande solamente si éste es grande para todos los tiempos en la escala hasta t . Haciéndolo de este modo, le damos un valor bajo a las particiones que tienen alta probabilidad de salir una comunidad y regresar posteriormente, como es el caso de las gráficas bipartitas.

Esta minimización no suele ser necesaria en la mayoría de los casos. De hecho, experimentos numéricos^[8] han demostrado que

$$r(t; H) = \min_{0 \leq s \leq t} \text{traza}[R(s)] \approx \text{traza}[R(t)] \quad (3.12)$$

Esta simplificación reduce dramáticamente el tiempo de cómputo y la supondremos como cierta.

La primera idea de cómo utilizar la estabilidad para encontrar comunidades relevantes sería teniendo una colección de todas las particiones H posibles, calculando su estabilidad para cada tiempo t y eligiendo para cada uno de estos tiempos la partición con la estabilidad más alta. De este modo, se obtiene la ‘Curva de Estabilidad’:

$$r(t) = \max_H r(t; H) \quad (3.13)$$

Podemos ver que el tiempo de Markov, t , actúa como un parámetro de resolución, estableciendo una jerarquía de particiones de muchas a pocas

comunidades. Podríamos decidir considerar como comunidades relevantes a aquellas que permanecen óptimas durante largos períodos de tiempo.^[8] Hay que notar que optimizar de este modo resultaría infactible, pero podemos encontrar heurísticas que intenten buscar este óptimo, como se describirá más adelante.

3.4. Estabilidad a tiempo continuo

Ahora, definimos de manera distinta la dinámica que toma lugar en la red, consideraremos un proceso de difusión Markoviano dado por una caminata aleatoria a tiempo continuo como sigue:

$$\frac{\partial p(t)}{\partial t} = -p(t)[D^{-1}L] \quad (3.14)$$

Una vez más, el estado estacionario está dado por la ecuación (3.3). Mientras que en el tiempo discreto, la caminata brinca de un nodo a otro en intervalos unitarios de tiempo, en el tiempo continuo, la caminata tiene un tiempo de espera entre saltos que es una variable aleatoria continua y sin memoria distribuida exponencialmente con esperanza unitaria. Cuando ocurre un salto, las probabilidades de los dos procesos son idénticas, por lo que se puede ver a la caminata en tiempo discreto como una aproximación a la de tiempo continuo, y de hecho, se ha reportado que llevan a estructuras de comunidad similares.^[11]

La estabilidad a tiempo continuo, se define de manera similar a la de tiempo discreto, cambiando la matriz de transición. En el tiempo continuo,

tendremos a la matriz

$$M(t) = \exp(-tD^{-1}L)$$

Por lo tanto, la matriz de autocovarianza de conglomerados está dada por:

$$R(t) = H^T[\Pi M(t) - \pi^T\pi]H$$

Definición 3.11. Se define la estabilidad a tiempo continuo de H al tiempo t como:

$$r(t; H) = \text{traza}R(t)$$

Como vimos en la sección anterior, la estabilidad a tiempo discreto incluye una minimización sobre el intervalo de tiempo $(0, t)$. En el tiempo continuo esto no es necesario, ya que la función decrece monótonamente con el tiempo, como se demuestra a continuación.

Proposición 3.4. En el caso continuo, la $\text{traza}(R(t))$ es monotonamente decreciente con respecto al tiempo

Demostración 3.4. Consideremos al Laplaciano normalizado

$$\mathcal{L} = D^{-1/2}LD^{-1/2}$$

Donde la matriz diagonal $D^{-1/2}$ es tal que

$$D_{ii}^{-1/2} = \frac{1}{\sqrt{d_i}}$$

Con él podemos reescribir a la matriz de transición como

$$M(t) = D^{-1/2}\exp(-t\mathcal{L})D^{1/2}$$

Haciendo uso de la definición de la exponencial y el teorema espectral para diagonalizar a \mathcal{L} .

Tomemos la derivada con respecto al tiempo de la estabilidad,

$$\frac{\partial(r(t; h))}{\partial t} = -\text{traza}(H^T \Pi D^{-1/2} \mathcal{L}^{1/2} \exp(-t\mathcal{L}) \mathcal{L}^{1/2} D^{1/2} H)$$

Esta es estrictamente negativa, por lo que podemos afirmar que la estabilidad es estrictamente decreciente, y la minimización no es necesaria, pues podemos tomar el tiempo t como el punto de mínima estabilidad. ■

Habiendo definido la estabilidad, hay que aclarar que la intención no es encontrar a la mejor partición en comunidades, sino descubrir particiones relevantes a diferentes resoluciones. Buscaremos que nuestras particiones sean persistentes en un intervalo de tiempo y robustas ante variaciones en la estructura de la gráfica o el método de optimización.

3.5. Estabilidad y Modularidad

La estabilidad provee un marco unificador de distintas heurísticas y medidas, en particular con la modularidad.

Proposición 3.5. Dada una red no dirigida, la estabilidad a tiempo discreto, en $t = 1$ equivale a la modularidad.

Demostración 3.5. Sea $r_1(H)$ la estabilidad a tiempo discreto al tiempo $t=1$, entonces

$$\begin{aligned} r_1(H) &= \text{traza}[H^T (\Pi D^{-1} A - \pi^T \pi) H] \\ &= \frac{1}{2m} \text{traza}[H^T (A - \frac{dd^T}{2m}) H] \end{aligned}$$

Haciendo uso de que las matrices Π y D^{-1} son diagonales y de que

$$(\Pi D^{-1})_{ii} = \frac{d_i}{2m} \times \frac{1}{d_i}$$

Escrito en forma matricial;

$$\Pi D^{-1} = \frac{I}{2m}$$

Sea X una matriz cualquiera de tamaño $n \times n$, la matriz que resulta de la multiplicación $H^T X H$ es una matriz de tamaño $c \times c$ que tendrá entrada a entrada la suma de las entradas de la matriz X agrupadas por comunidad, como sigue: $(H^T X H)_{kl} = \sum_{i \in k, j \in l} X_{ij}$

Haciendo uso de esto, sea $X = A - \frac{dd^T}{2m}$

$$\text{entonces, } (H^T X H)_{kl} = \sum_{i \in k, j \in l} A_{ij} - \frac{d_i d_j}{2m}$$

$$\frac{1}{2m} \text{traza}(H^T X H) = \frac{1}{2m} \sum_{k=1}^c (H^T X H)_{kk} = \frac{1}{2m} \sum_{k=1}^c \sum_{i,j \in k} A_{ij} - \frac{d_i d_j}{2m}$$

$$r_1(H) = \frac{1}{2m} \sum_c \sum_{i,j \in c} A_{ij} - \frac{d_i d_j}{2m}$$

Por lo tanto, $r_1(H) = \text{Modularidad}$ ■

Esta demostración sólo es válida para el caso de redes no dirigidas y a tiempo discreto. Sin embargo, esta igualdad nos será útil para facilitar la optimización de la estabilidad.

3.5.1. Límite de resolución

Se ha detectado que al utilizar la optimización de modularidad como método de detección de comunidades, el método puede resultar incapaz

de identificar comunidades naturales,^[12] dando particiones muy gruesas, es decir, sobre agrupando. Esto se debe al llamado ‘límite de resolución’ que se refiere a un límite que depende del tamaño total de la gráfica y de qué tan conectados estén sus nodos. Podemos encontrar dos casos peligrosos en los que la partición se vería afectada por este límite:

1. Dos comunidades tienen un balance perfecto entre aristas dentro de la comunidad y aristas fuera de él. Así que están en el límite de conformar una comunidad. (Figura 3.1)
2. Dos comunidades tienen una única arista conectándolas con el resto de la gráfica, y sólo una arista conectándolas al uno con el otro. En este caso, el método de modularidad no podría separarlas aunque se tratara de dos gráficas completas unidas por una única arista. (Figura 3.2)

Las figuras 3.1, 3.2 muestran ejemplos de gráficas dónde al optimizar la modularidad, ocurre este límite de resolución. Aunque las gráficas no modelan datos reales, sino que son ejemplos creados para estos casos limitantes, no es improbable encontrar alguna estructura similar en la realidad donde la modularidad sería incapaz de agrupar correctamente.

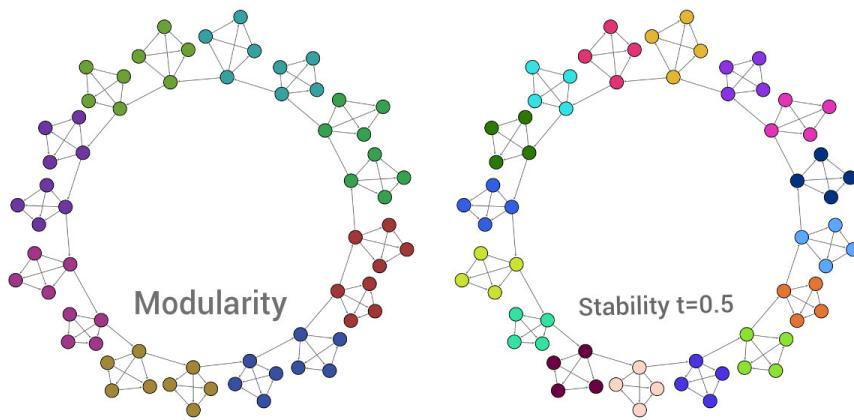


Figura 3.1: Ejemplo de una red con comunidades formadas por subgráficas completas de tamaño 4. La modularidad es incapaz de separarlas, mientras que la estabilidad lo resuelve para $t=0.5$

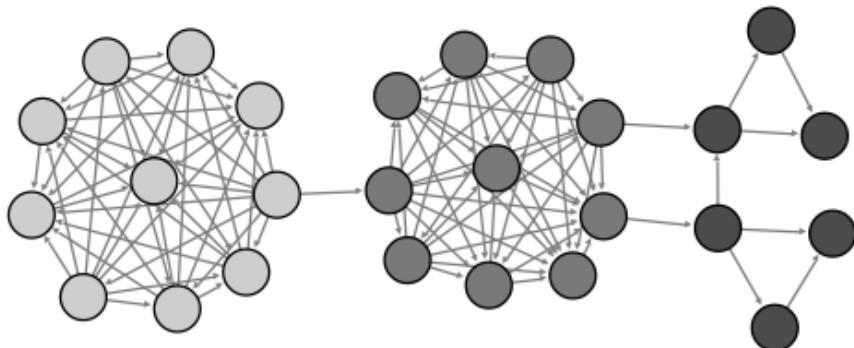


Figura 3.2: Ejemplo de una red con dos comunidades conformadas por subgráfica completa de tamaño 10, y dos comunidades por subgráficas completas de tamaño 3. Por la diferencia en tamaños, la modularidad agrupa juntas a las comunidades pequeñas.

Si observamos el comportamiento de la estabilidad en el límite, podemos observar que el tiempo de Markov actúa como un parámetro de resolución, y de hecho, ayuda a vencer el límite de resolución de la modularidad.

1. En $t = 0$, la estabilidad $R(0)$ es máxima cuando tenemos una partición donde cada nodo está en su propia comunidad.
2. En $t = \infty$, la estabilidad $R(\infty)$ normalmente es maximizada por una partición en 2 comunidades.

Estas condiciones límite nos permite ver que $R(t)$ optimizada a tiempos distintos da una secuencia de particiones dónde el número de comunidades va decreciendo, por lo que t actúa como un parámetro de resolución.

Utilizando la estabilidad, las particiones que rompen el límite de resolución de la modularidad se obtienen optimizando para tiempos pequeños, cercanos a 0.

Hasta ahora, hemos explicado la teoría detrás de la medida de ‘estabilidad de partición’ y hemos definido esta medida. Así mismo, la hemos relacionado con la medida de ‘modularidad’. Esta relación la hemos hecho con la finalidad de justificar el uso del algoritmo de Louvain (Sección 4.1), el cual fue desarrollado como heurística para la maximización de modularidad como ayuda para optimizar la estabilidad. El siguiente capítulo se enfoca en explicar una heurística que busca maximizar la medida de estabilidad con la finalidad de encontrar comunidades.

Capítulo 4

Optimización de estabilidad

Como hemos visto, la estabilidad es una medida de calidad de particiones. En principio podríamos utilizarla simplemente para calificar particiones encontradas por algún otro algoritmo. Pero en este capítulo veremos cómo usarla para definir un método de detección de comunidades, usándola como función objetivo. Para cada tiempo t , encontraremos una partición H con estabilidad máxima. Al igual que la optimización de modularidad, hacer esta optimización sobre todo el espacio de las particiones es un problema NP-completo, por lo que necesitaremos hacer uso de heurísticas.

4.1. Algorítmico de Louvain

En esta tesis haremos uso de que siempre es posible reescribir el problema de estabilidad de una gráfica como la modularidad de otra gráfica (Sección 3.5). La importancia de esto es que la modularidad ha sido muy estudiada y existen muchas heurísticas eficientes para optimizarla, pudien-

do utilizar cualquiera de ellas para ayudarnos a optimizar la estabilidad, en particular el algoritmo de *Louvain*.^[13]

La razón por la que existe un problema de modularidad equivalente a la estabilidad, es similar a lo demostrado en el capítulo anterior para estabilidad a tiempo discreto en $t = 1$. Para la estabilidad a tiempo continuo, podemos decir que

$$r(t; H) = \text{Modularidad}(G')$$

Donde $r(t; H)$ es la estabilidad a tiempo continuo de una gráfica G con matriz de adyacencia A y G' es una gráfica con pesos dependientes del tiempo, con matriz de adyacencia $X(t) = \exp(-tD^{-1}L)$. Por construcción, $X(t)$ recorre más nodos de la red, lo que le da menos peso a los caminos de longitud 1, que son los dados por la matriz A . Esto significa que se espera encontrar comunidades más grandes cuando se optimiza la modularidad de $X(t)$ cuando t crece.

Ahora que sabemos que podemos replantear el problema de estabilidad como uno de modularidad, explicaremos el algoritmo que será utilizado para optimizar la modularidad, el algoritmo de *Louvain*. El método original fue idea de Etienne Lefebvre en el 2007, y fue mejorado por Blondel, Guillaume y Lambiotte.^[13] Se conoce como ‘*Louvain*’ porque los autores estudiaban en la Universidad Católica de Louvain. Es uno de los algoritmos más rápidos y ha probado dar resultados prácticos muy buenos.

El algoritmo se divide en varias corridas, cada una consistente de 2 fases. En cada iteración de fases, se hará uso del cálculo de la *ganancia en modularidad*. La ganancia está dada por la siguiente fórmula:

$$\Delta Q = \left[\frac{\sum_C + d_{i,C}}{2m} - \left(\frac{\sum_{adj} + d_i}{2m} \right)^2 \right] - \left[\frac{\sum_C}{2m} - \left(\frac{\sum_{adj}}{2m} \right)^2 - \left(\frac{d_i}{2m} \right)^2 \right]$$

Donde

$$\sum_C = \sum_{j,k \in C} w_{jk} \quad , \quad d_{i,C} = \sum_{j \in C} w_{ij} \quad , \quad \sum_{adj} = \sum_{j \in C, k \notin C} w_{jk}$$

El proceso se inicia con una gráfica G con n nodos.

Fase 1:

1. Asignarle a cada nodo su propia comunidad, es decir, en principio tenemos n comunidades.
2. Para cada nodo v_i , nos fijamos en sus d_i vecinos y evaluamos la *ganancia en modularidad* que resultaría de quitar a v_i de su propia comunidad y ponerlo en la de cada uno de sus vecinos.
3. Ponemos a v_i en la comunidad del vecino v_j que da la mayor ganancia, siempre y cuando esta sea estrictamente positiva. En caso de no existir movimiento que suponga una ganancia positiva, dejamos a v_i en su propia comunidad.
4. Este proceso se repite secuencialmente para todos los nodos hasta que no se pueda obtener mejoría de hacer estos movimientos individuales. En esta fase se obtiene un óptimo local.

Fase 2:

1. Construimos una nueva gráfica donde las comunidades encontradas en la fase 1 se vuelven nodos.
2. Ahora, las aristas tendrán pesos, que se construyen sumando los pesos

de las aristas entre comunidades, de la siguiente manera:

$$\hat{w}_{ij} = \sum_{v_k \in C_i, v_l \in C_k} w_{kl}$$

Donde \hat{w}_{ij} es el peso de los nodos formados por la comunidad i y la comunidad j y w_{kl} es el peso de la arista entre el nodo v_k y v_l si esta arista existe. Las aristas entre nodos de una misma comunidad se vuelven lazos. Cabe mencionar que si iniciamos con una gráfica sin pesos, tomamos $w_{kl} = 1$ para toda arista $v_k v_l \in E$

El algoritmo sigue corriendo hasta que no se encuentre mejoría en la modularidad, y en cada corrida el número de comunidades, claramente, va decreciendo.

La razón por la que el método resulta tan rápido es por lo fácil que es calcular la ganancia en modularidad al hacer un cambio.

Es importante notar que la partición resultante depende del orden en el que se revisan los nodos. En la práctica, se corren varios procesos completos con inicios aleatorio y se toma el que da el mejor resultado. Los autores de este método, dicen que éste podría resolver el problema de resolución de la modularidad, ya que las soluciones intermedias pueden ser tomadas como particiones óptimas a diferentes resoluciones. Esto no ha probado ser cierto,^[14] y de hecho, en el método para maximizar la estabilidad, tomaremos solo la solución con máxima modularidad, y ni siquiera veremos las ‘otras’ resoluciones.

4.2. Maximización de la estabilidad de partición para un tiempo de Markov

En el capítulo anterior, en la definición de la medida de estabilidad, se puede ver que ésta depende de un parámetro de tiempo; el tiempo de Markov. También mencionamos que este parámetro de tiempo actúa como un parámetro de resolución, en el que iniciamos con muchas comunidades pequeñas, y terminamos con la partición trivial en una única comunidad. El algoritmo para maximizar la estabilidad a un tiempo de Markov t está dado por:

Algoritmo 2: Partición óptima

Input: A : matriz de adyacencias, t : tiempo de Markov, r : número de inicios aleatorios

Output: H^* : partición óptima en t

Calcular $P(t) = \exp(-tD^{-1}L)$

for $i = 1 \dots r$

Correr Louvain (aleatorio) para $P(t)$ y obtener la partición H_i

Calcular la estabilidad $r(t; H_i)$

Guardar H_i si $r(t; H_i) \geq r(t; H_{i-1})$

end for

Lo que hacemos es optimizar la modularidad para la matriz de transición del tiempo t , dada por $P(t)$ como si esta fuera una matriz de adyacencias. Posteriormente, calculamos la estabilidad de la partición resultante. Repetimos este proceso varias veces, ya que el algoritmo de Louvain tiene

inicios aleatorios, los cuales pueden arrojar resultados distintos. Utilizamos la mejor partición que arroja Louvain, de acuerdo a la medida de estabilidad.

Hasta este punto, hemos explicado cómo se encuentra la partición óptima a aun tiempo de Markov, pero nos interesa encontrar particiones ‘relevantes’ a distintos tiempos. Para elegir los tiempos de Markov que pudieran resultar interesantes, se utiliza una medida llamada ‘Variación de Información’. La cual, dada una secuencia de tiempos de Markov (t_1, \dots, t_s) compara la partición óptima de t_i con los óptimos en t_{i-1} y t_{i+1} para buscar particiones ‘robustas’. Esta medida se explica en el apartado siguiente.

4.3. Variación de la Información

La importancia de una partición a un tiempo, debe ir asociada a alguna medida de robustez. Un resultado se considera robusto si los resultados no se alteran bajo pequeñas modificaciones a la optimización, la red, o la medida de calidad. Para nosotros, la robustez de una partición óptima estará dada por la persistencia de ésta sobre períodos ‘largos’ de tiempo.

Para ello, necesitaremos comparar las particiones y buscar intervalos de tiempo donde éstas son muy similares. La medida que utilizaremos es la *Variación de la Información*.

Definición 4.1. Sean P_1 y P_2 dos particiones que se desea comparar, se define la *Variación de la Información* entre P_1 y P_2 como:

$$\widehat{VI}(P_1, P_2) = [2H(P_1, P_2) - H(P_1) - H(P_2)]/\log N \quad (4.1)$$

Dónde $H(P) = -\sum_C p(C)\log p(C)$ es la entropía de Shannon, y $H(P_1, P_2)$ es la entropía de la probabilidad conjunta. Y la frecuencia relativa de que un nodo esté en la comunidad C en la partición P_α es $p(C_\alpha) = n_{C_\alpha}/N$. Esta versión de la Variación de la Información está normalizada por su máximo que es $\log N$, por lo que toma valores en $[0, 1]$ y se va a 0 sólo cuando las particiones son idénticas. Denotaremos a esta normalización o media de la Variación de Información por \widehat{VI}

Para analizar los resultados de la maximización de estabilidad, queremos 2 cosas:

1. Que la partición sea persistente sobre un rango amplio de tiempos.
Esto se observaría en la \widehat{VI} como mesetas que se extienden sobre intervalos largos de tiempo.
2. Evaluar la dispersión que se introduce por la componente aleatoria de la heurística. Como mencionamos antes, para cada tiempo se corren r optimizaciones Louvain aleatorias. Una \widehat{VI} baja indica que se encontraron particiones óptimas muy similares en cada una de estas optimizaciones aleatorias.

El cálculo de la \widehat{VI} es muy costoso. Sin embargo es necesaria para tener un criterio de decisión para elegir tiempos de Markov relevantes y no explorar las particiones que arroja la maximización a demasiados tiempos.

4.4. Heurística para detección de comunidades a partir de la estabilidad de partición

Como hemos dicho antes, la estabilidad no busca encontrar la mejor partición, sino una serie de particiones relevantes a tiempos distintos. La heurística que vamos a utilizar ha sido utilizada antes por Delvenne, Schaub, Yaliraki y Barahona.^[8,15] El método que se utiliza en esta tesis, entonces, tiene 2 partes.

1. Componente exploratoria: Busca la partición óptima para una secuencia de tiempos. Devuelve una colección de particiones. Probablemente no nos interesen todas, pero las utilizamos para analizar cuáles son relevantes con ayuda de la Variación de Información 4.1.
2. Obtener una partición óptima: Tras haber analizado la gráfica de VI, podemos decidir qué tiempo (o tiempos) nos resultan interesantes, y obtener la partición correspondiente a éste.

Para la componente exploratoria, básicamente repetiremos para cada tiempo de Markov de una lista de tiempos la optimización descrita en el algoritmo 2 y utilizaremos las particiones encontradas como óptimas para calcular la \widehat{VI} . Esto se describe a continuación:

Como vemos, la segunda parte se utiliza también en la primera. La secuencia de particiones obtenida en la primera parte, descubre la estructura en comunidades que tiene la red a distintas escalas. Esta es la primera parte de nuestro objetivo, de ahí sigue elegir las escalas más relevantes, para lo que

Algoritmo 3: Componente exploratoria

Input: A : matriz de adyacencias, \hat{t} : vector de tiempos

Output: VI: variación de información

for t en \hat{t}

 Correr **Algoritmo 3**

end for

Calcular VI con las H_t^*

necesitaremos un criterio de ‘relevancia’, dado por la \widehat{VI} . Una vez elegidos los tiempos relevantes, recuperamos el óptimo encontrado para ellos.

El costo computacional de la primera parte de la heurística es muy alto, tanto por el número de optimizaciones que se corren a cada tiempo, como por el cálculo de la \widehat{VI} . El costo de calcular la \widehat{VI} no puede ser disminuido ya que es la que nos permite descubrir más de la estructura de la gráfica. El costo podría reducirse significativamente utilizando una versión linealizada de la estabilidad, o paralelizando los cálculos costosos; sin embargo, esto no se implementó para este trabajo.

Parte II

Resultados y Conclusión

Capítulo 5

Resultados

5.1. Zachary's Karate Club

La primera prueba de la heurística será sobre la famosa red de ‘Karate Club’, la cual ha sido utilizada en un gran número de publicaciones para probar los métodos de detección de comunidades.

La red fue resultado de un estudio antropológico realizado en los años 70’s al Club de Karate de una universidad, en el que se estudiaba la ‘*asociación voluntaria*’. Tras años de análisis, existió un problema interno entre el instructor y el presidente del Club, lo que provocó que se desintegrase en dos nuevos clubes. Wayne W. Zachary creó con estos datos una red social, donde los miembros del club son los nodos y los arcos existen si los miembros tenían interacción social fuera de las actividades del club de karate. Su estudio trataba sobre predecir, con la red, quiénes se irían con el instructor y quiénes con el presidente, bajo la hipótesis de que la gente no iría sola,

sino con sus amigos.

Nos interesa probar el algoritmo con esta gráfica, ya que se utiliza en la investigación de detección de comunidades como un problema básico que tiene que resolverse con el óptimo de 4 o 2 comunidades para considerar los resultados del método como válidos.

5.1.1. Características de la red

Karate Club es una gráfica no dirigida de 34 nodos y 78 aristas. Podemos observar en la Fig. 5.1 que los nodos más importantes son el #1 y el #34, estos corresponden, de hecho, al instructor y al presidente respectivamente. Podemos ver, también, que existen algunos nodos en la frontera, que están conectados casi con la misma fuerza a una y otra facción.

5.1.2. Resultados sobre Karate Club

Se corrió la componente exploratoria sobre la red con semilla 1. Esta componente maximiza la estabilidad para una lista de tiempos. Para cada tiempo, se hicieron 5 diferentes optimizaciones. Con esto, se dibuja la Variación de la Información. La estabilidad a cada tiempo no se graficó, ya que siempre es decreciente con respecto al tiempo, y no es relevante la comparación. Los tiempos de Markov están espaciados logarítmicamente en el intervalo (0, 100).

El algoritmo se detiene al encontrar la partición trivial con una comunidad. Como podemos ver en la gráfica, esta red encuentra su partición trivial, en un tiempo muy pequeño. Este comportamiento es común en redes pequeñas.

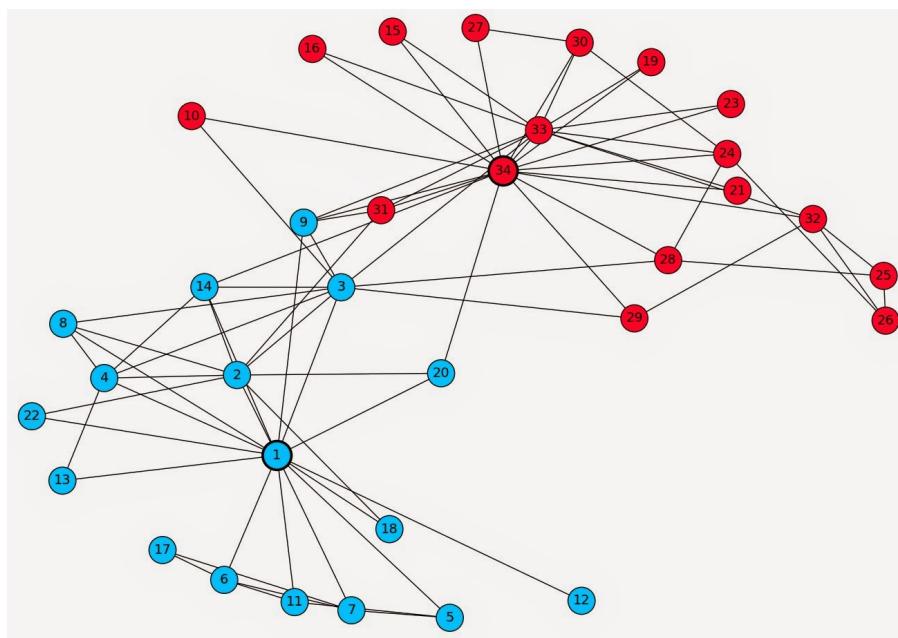
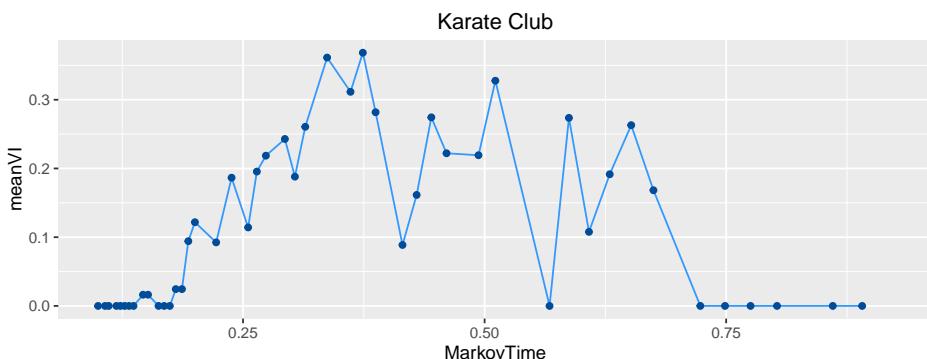


Figura 5.1: ‘Karate Club’ Grupos tras la desintegración (ground truth)

Figura 5.2: ‘Karate Club’ \widehat{VI}

Utilizando la heurística de la Variación de Información, buscamos decrecimientos o períodos de tiempo estables para tratar de encontrar una

buenas particiones. En este caso, el ejemplo solo busca replicar el resultado presentado en la sección anterior; para mostrar la validez de este método. Se eligen 2 tiempos de Markov relevantes, el primero en $t = 0.74881$ donde se encuentran 4 comunidades y el segundo en $t = 0.922198$ con 2 comunidades.

En el primer tiempo relevante, se obtiene el equivalente al resultado de la Modularidad, que arroja una partición en 4 comunidades. Y, en el segundo tiempo relevante se obtiene ‘casi’ la partición real del club de Karate, excepto por el nodo #9 que cae en la comunidad equivocada. Este error, es el mismo error que obtuvo el autor de la red, Zachary, al usar el algoritmo de Ford & Fulkerson para partir la red en grupos, y es el mismo error en el que caen los métodos de detección de comunidades que llegan a una partición en 2 grupos, pues el comportamiento de la persona de este nodo fue extraño.

Ambos resultados se muestran a continuación:

5.2. Los Miserables

La red anterior sirvió para determinar si los resultados este método tienen sentido en la partición en comunidades que ha sido encontrada con otros. En esta sección veremos las posibilidades que brinda la resolución. Para esto, utilizaremos la red creada por D.E. Knuth de *The Stanford GraphBase*.

La red está formada por los personajes de la novela ‘Los Miserables’ de Victor Hugo como nodos, existiendo un arco entre un par de nodos si estos personajes aparecen en el mismo capítulo. La gráfica resultante, es una gráfica no dirigida formada por 77 nodos y 254 arcos, cuyos pesos están

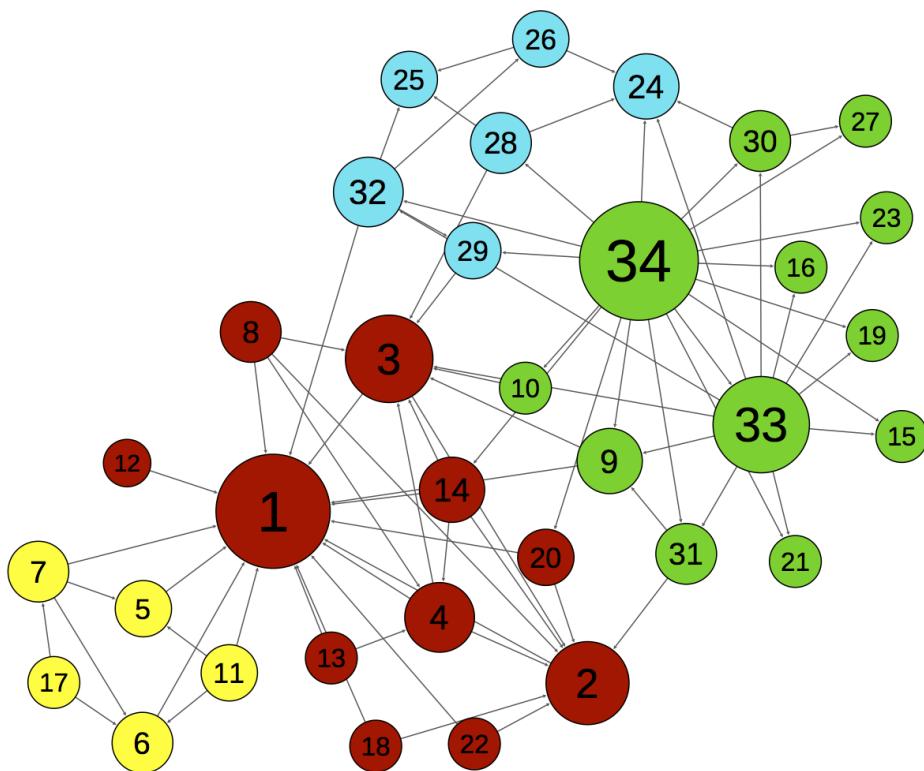


Figura 5.3: ‘Karate Club’ Partición en 4 comunidades. Equivalente a modularidad.

dados por el número de capítulos en el que co-aparecen los personajes.

5.2.1. Resultados

Los resultados que aquí se muestran fueron obtenidos sin tomar en cuenta los pesos de la red, es decir, considerando todos los arcos con peso igual a 1. Esto porque el análisis de la red sin pesos resulta más acertado,

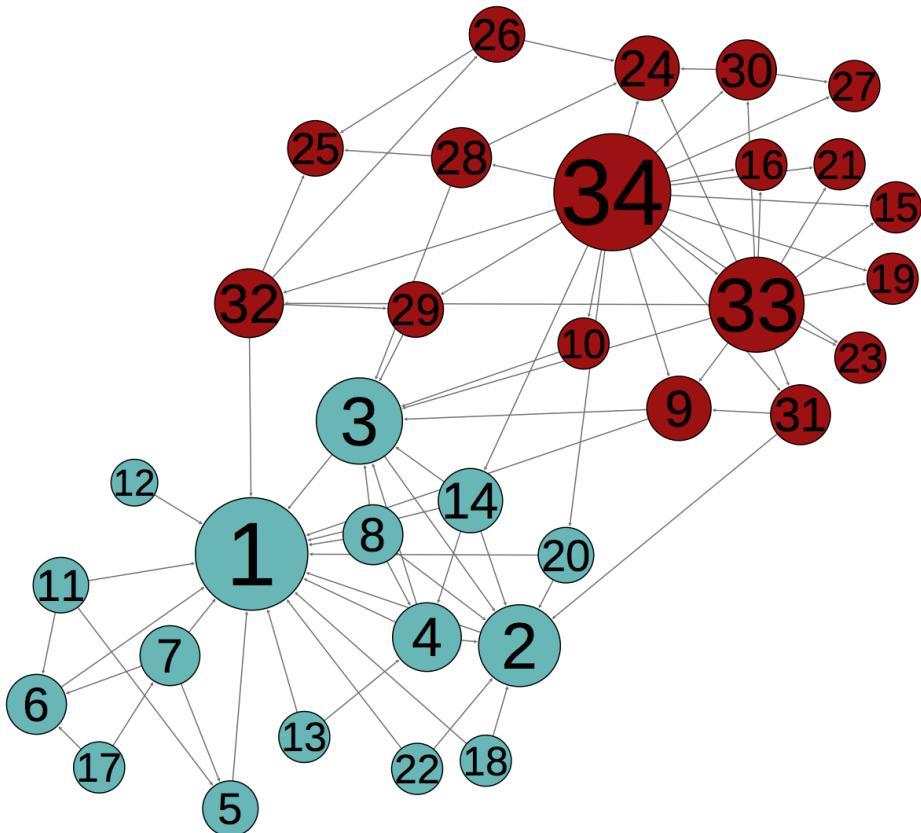


Figura 5.4: ‘Karate Club’ Partición en 2 comunidades. Recupera el ground-truth

ya que al utilizarlos, resultó que prevenían la separación de algunos nodos y empujaban a otros, con conexiones débiles, a quedar en comunidades aisladas.

Se corrió la componente exploratoria sobre la red con semilla = 1. Una

vez más se hicieron 5 optimizaciones con inicios aleatorios para la optimización, y 5 iteraciones para cada una de ellas. Con esto, se grafica la Variación de la Información, que nos ayudará una vez más a encontrar tiempos con particiones relevantes.

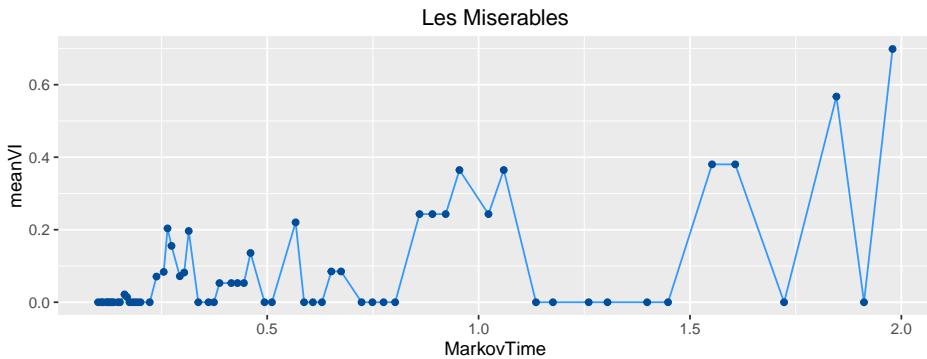


Figura 5.5: ‘Los Miserables’ \widehat{VI}

Para ilustrar la elección de los puntos considerados relevantes de esta gráfica, se muestra la Figura(5.5) que muestra el número de comunidades encontrado para cada tiempo y su respectiva \widehat{VI} . Como nos interesa analizar el significado de las comunidades, no se toman en cuenta los tiempos donde existen demasiadas comunidades; empezamos a explorar a partir del tiempo 0.337006 dónde se da la primera caída importante de la \widehat{VI} con un número de comunidades razonable.

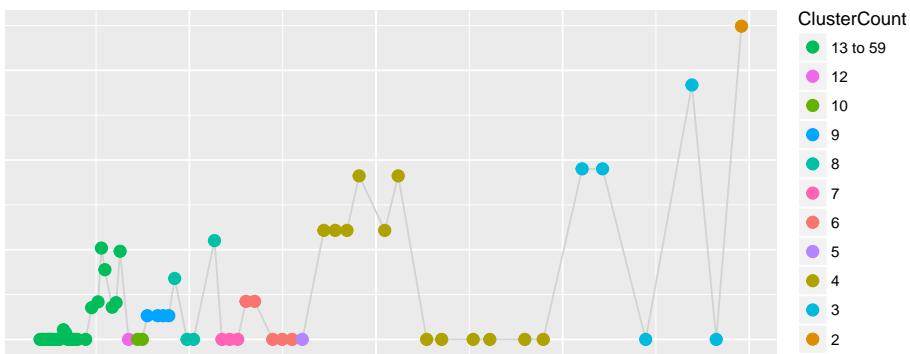


Figura 5.6: ‘Los Miserables’ Número de comunidades encontradas.

Los tiempos considerados como relevantes, que se describirán a continuación, fueron elegidos buscando caídas en la \widehat{VI} . En la gráfica, observamos varias caídas, pero las más importantes, son las drásticas, o las que bajan a 0. En algunos casos, tenemos varios tiempos consecutivos con $\widehat{VI} = 0$, en estos casos, se eligió un tiempo centrado, o dos tiempos si el número de comunidades cambió. El Cuadro(5.1) muestra los tiempos elegidos.

Tiempo	Estabilidad	Comunidades	\widehat{VI}
0.337006	0.381	11	0.0
0.361234	0.375	10	0.0
0.493705	0.357	8	0.0
0.608022	0.338	7	0.0
0.74881	0.325	6	0.0
0.802643	0.318	5	0.0
1.26038	0.253	4	0
1.72259	0.219	3	0.0
1.91164	0.123	3	0.0

Cuadro 5.1: **Los Miserables:** Análisis exploratorio, tiempos seleccionados

5.3. Análisis por Tiempo de Markov

En esta sección analizaremos, para cada uno de los tiempos seleccionados como relevantes, el significado de las comunidades que se encontraron.

Una de las formas más comunes para explicar una comunidad es utilizar los nodos más relevantes dentro de ella. Por ello, los nodos con mayor grado serán los guías para explicar a la comunidad.

Iniciamos con el tiempo más bajo, que nos da el mayor número de comunidades. Para explicar las comunidades, las nombraremos de acuerdo al número que arroja el algoritmo. Dado que el ‘número’ de la comunidad es arrojado de manera aleatoria, utilizaremos el orden de nombres del primer análisis para facilitar la relación.

5.3.1. Once comunidades

Se encuentran 11 comunidades para $t = 0.337006$. La Figura(5.7) nos muestra el resultado para este tiempo. A continuación, se explicará cada una de las comunidades encontradas para esta resolución, y estas comunidades serán utilizadas como referencia en adelante.

- 0 -** Jean Valjean y personajes poco importantes en la novela. Marguerite y Scaufflier aparecen en el ‘Volumen 1: Fantine’, al final del volumen, cuando su camino cruza el de Jean. Gervais, es un niño al que roba Jean Valjean, cuando él aún es un criminal.
- 1 -** Revolucionarios de la barricada, y Madame Hucheloup, dueña del Corrinthe, donde se establece la barricada.
- 2 -** El Obispo Myriel y personajes que solo aparecen con él, como su hermana, Mlle. Baptistine y su ama de llaves, Mme. Magliore.
- 3 -** La familia de Marius, como sus abuelos Gillenormand y su padre, Georges Pontmercy. Y Cosette, esposa de Marius.
- 4 -** La juventud de Fantine; sus amigas, y los amigos del padre de Cosette, Félix Tholomyes, quienes fueron amantes.
- 5 -** Personajes del juicio contra el supuesto Jean Valjean, Champmathieu. Brevet, Cochepaille y Chenildieu son criminales que acusan a Champmathieu.
- 6 -** Javert es miembro de esta comunidad, junto con personajes poco importantes, como las monjas Simplice y Perpetue que salvan de Javert

a Jean Valjean cuando este visita a Fantine.

- 7 - Los criminales del grupo Patron-Minette.
- 8 - La familia Thenardier, excepto Eponine, quien está en la comunidad de criminales.
- 9 - Gavroche, y sus hermanos que se unen a él en la barricada.
- 10 - Mabeuf y su sirvienta Plutarch; también pelearon en la barricada.

En esta resolución encontramos comunidades pequeñas con relaciones muy cercanas en los casos donde los nodos comparten grado similar, como los criminales o los revolucionarios; y relaciones sin importancia para nodos con grado muy alto que conviven con personajes de grado muy bajo, como Valjean y Javert.

5.3.2. Diez comunidades

Se encontraron 10 comunidades para $t = 0.361234$. La Figura(5.8) nos muestra el resultado para este tiempo. Las comunidades permanecen idénticas a la resolución anterior, con la excepción de que las comunidades 7 (criminales Patron-Minette) y la familia Thenardier se unen para formar una misma comunidad. Esta unión hace sentido, pues juntos cometen el robo a la casa Gorbeau.

5.3.3. Ocho comunidades

Se encontraron 8 comunidades para $t = 0.493705$. La Figura(5.9) nos muestra el resultado para este tiempo. La mayoría de las comunidades per-

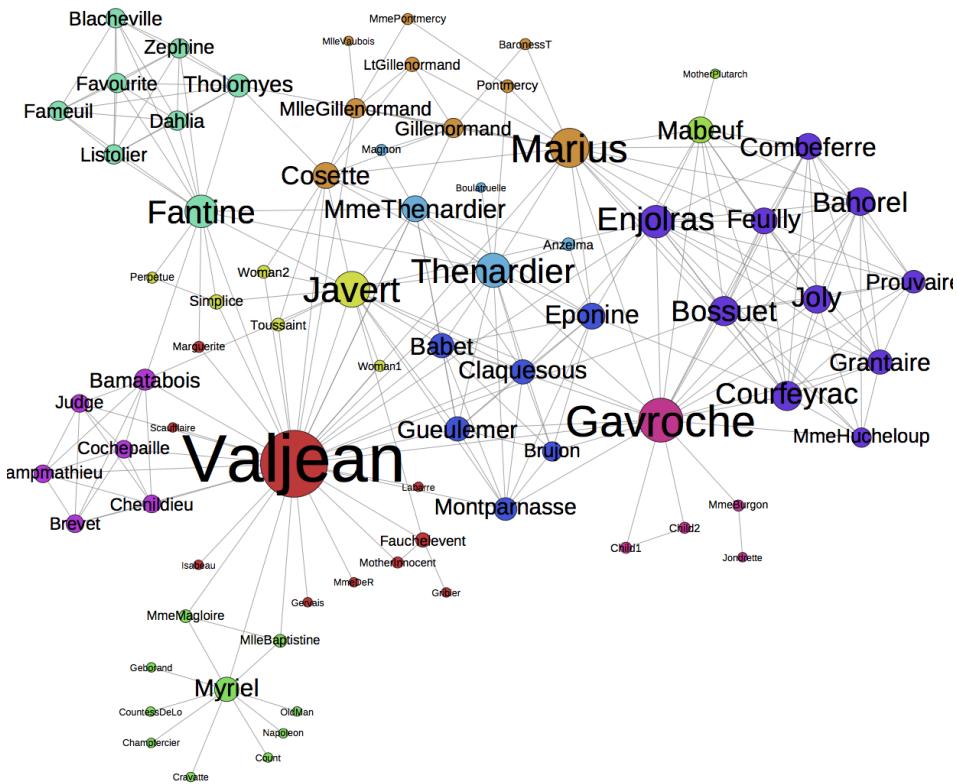


Figura 5.7: ‘Los Miserables’ 11 comunidades

manecen iguales a las de la resolución anterior. Pero, encontramos 3 cambios importantes.

- La comunidad 4, de la juventud de Fantine, incluye ahora también a los personajes que compartía con las comunidades 0 y 6 de Jean Valjean y Javert respectivamente. Estos personajes son las hermanas que cuidaron a Fantine en el hospital, y su vecina, Marguerite.
- Las comunidades 0 y 6, de Jean Valjean y Javert finalmente se unen,

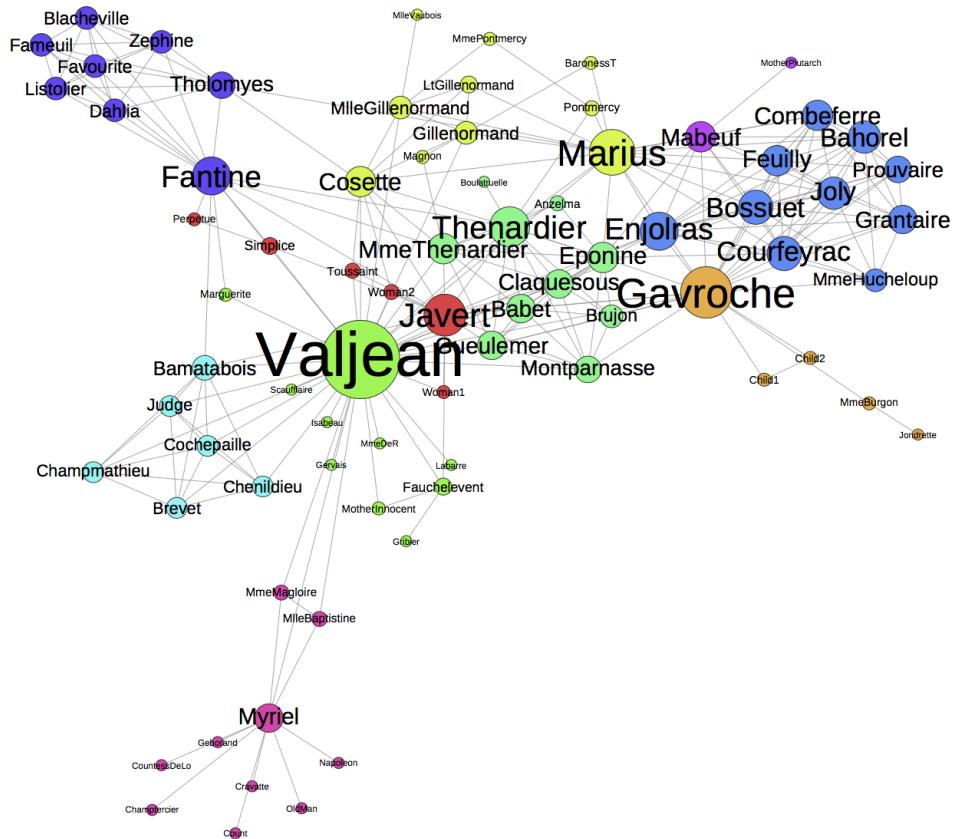


Figura 5.8: ‘Los Miserables’ 10 comunidades

aún acompañados de sus personajes poco importantes.

- Asimismo, la comunidad 10 que solo incluía Mabeuf y su sirviente, se unen con la comunidad 1 de revolucionarios de barricada, a la que realmente pertenecen.

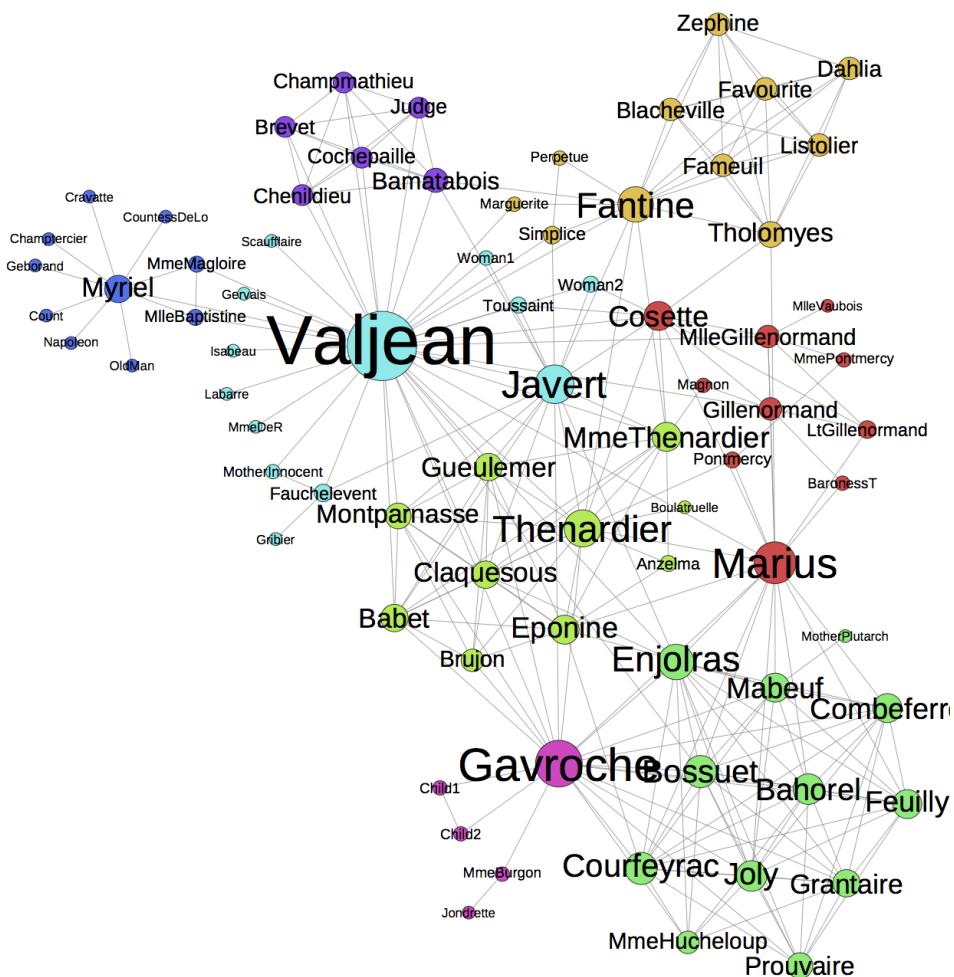


Figura 5.9: ‘Los Miserables’ 8 comunidades

5.3.4. Siete comunidades

Se encontraron 7 comunidades para $t = 0.608022$. El análisis de esta comunidad no merece la pena, ya que es idéntica a la siguiente resolución, salvo porque la comunidad de Gavroche se separa en 2. Esta separación

empeora el análisis de comunidades.

5.3.5. Seis comunidades y óptimo de modularidad

Se encontraron 6 comunidades para $t = 0.74881$. La Figura(5.10) nos muestra el resultado para este tiempo. Cabe mencionar, que este es el resultado obtenido de usar Louvain para maximizar la modularidad. Una vez más, la mayoría de las comunidades se mantienen, y tenemos 4 cambios importantes.

- Las comunidades 0 y 6, de Jean Valjean y Javert ya no están unidas.
- Javert se une ahora a la comunidad de delincuentes que se formó en la resolución anterior.
- La comunidad 0, de Jean Valjean, absorbe otra parte de su historia, los personajes del juicio de la comunidad 5.
- Continúa incrementándose la comunidad revolucionaria, ahora con la unión de la comunidad 9, de Gavroche, con los revolucionarios de barricada de la resolución anterior.

5.3.6. Cinco comunidades

Se encontraron 5 comunidades para $t = 0.802643$. La Figura(5.11) nos muestra el resultado para este tiempo. Una vez más, la mayoría de las comunidades se mantienen, y solo tenemos un cambio importante. la comunidad de Jean Valjean absorbe a otra comunidad relacionada con su historia; la del Obispo Myriel.

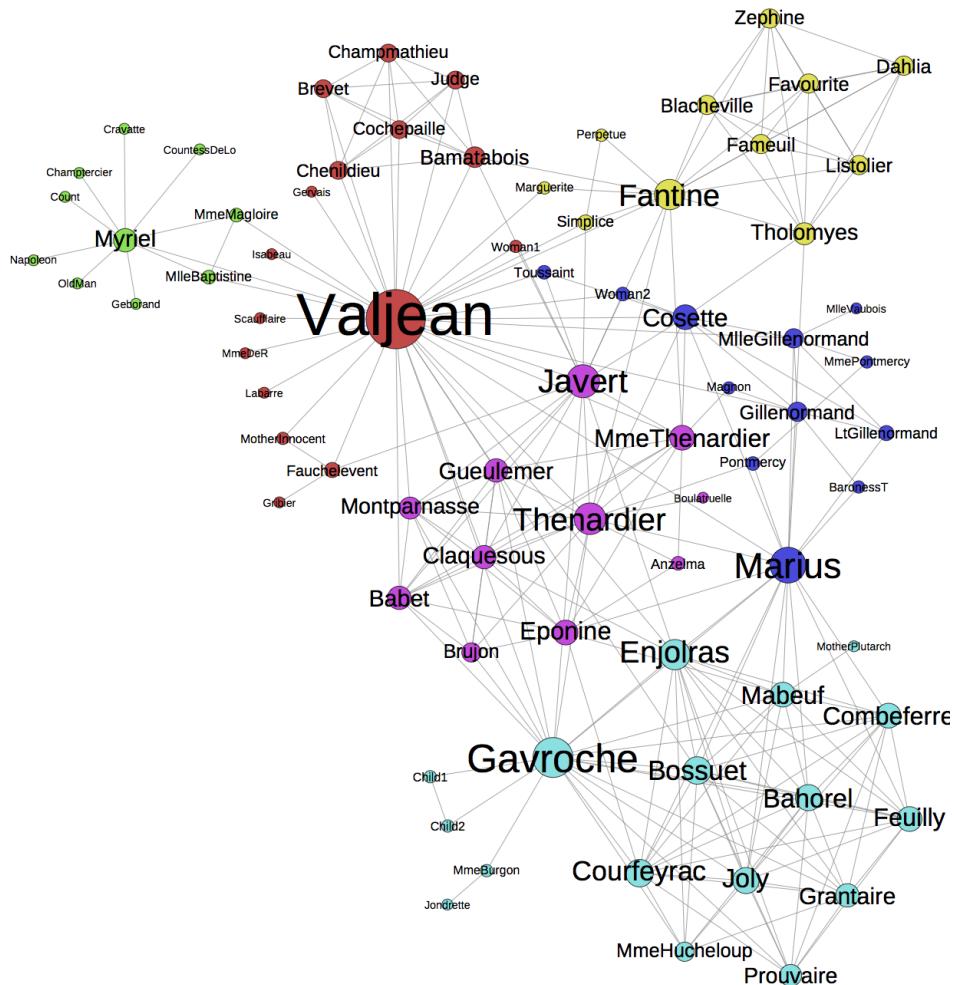


Figura 5.10: ‘Los Miserables’ 6 comunidades y óptimo de modularidad

5.3.7. Cuatro comunidades

Encontramos 4 comunidades para dos tiempos de Markov considerados relevantes, uno es una meseta con $\widehat{VI} > 0$ y el otro, con $\widehat{VI} = 0$. Ambas

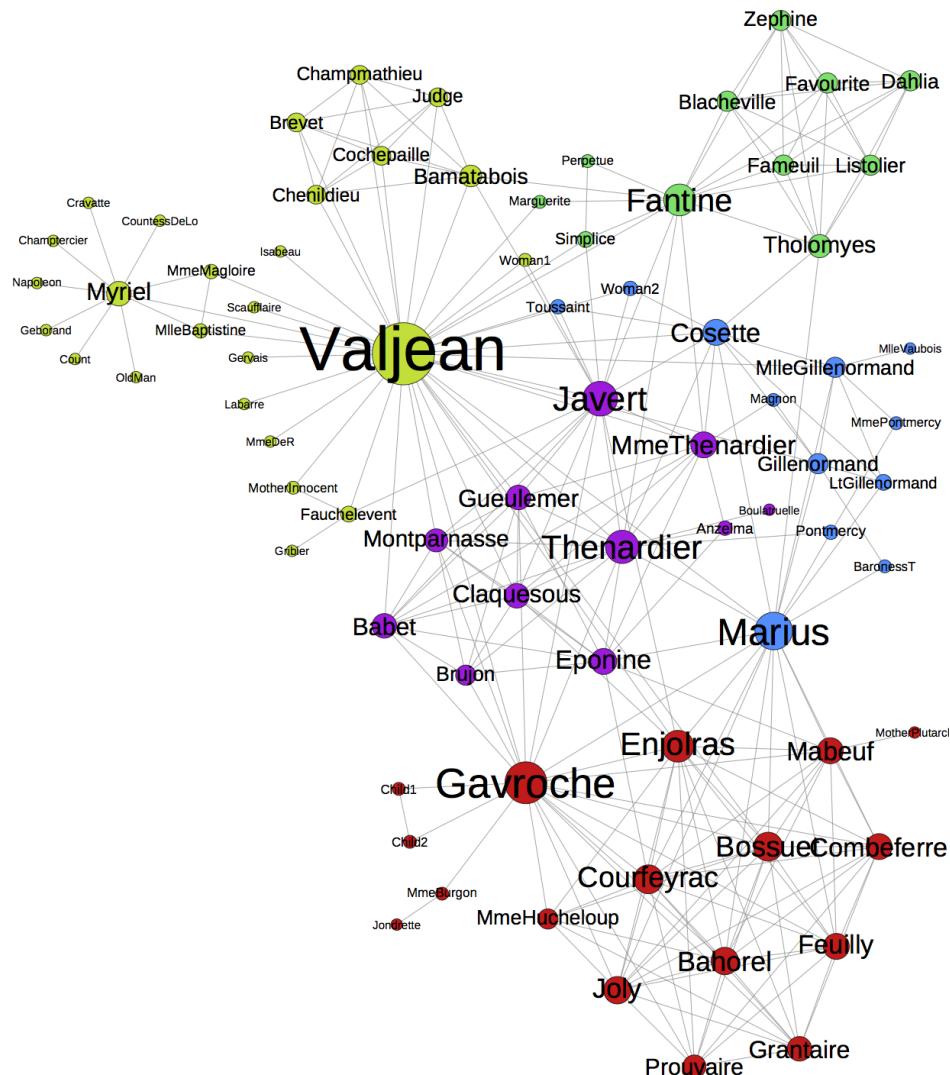


Figura 5.11: 'Los Miserables' 5 comunidades

son importantes, y muestran resultados complementarios.

La Figura(5.12) nos muestra el resultado para el tiempo $t = 0.890735$.

En esta resolución, tenemos 2 cambios importantes:

- Marius deja de pertenecer a su comunidad, y se une a la comunidad de los revolucionarios. De este modo, todos los personajes de la barricada quedan en la misma comunidad.
- La comunidad a la que pertenecía Marius, pasa a formar parte de la comunidad de Javert y los delincuentes. En este caso, esa unión parecería no tener sentido, pero se atraen, ya que el nodo más importante de la comunidad ahora es Cosette, que está muy relacionada con los Thenardier, su familia adoptiva. Asimismo, porque los Thenardier se presentan en la boda de Cosette, a la que asiste la familia de Marius.

En este primer tiempo, tenemos 4 comunidades perfectamente delineadas, la de Jean Valjean, la de Fantine, la de Marius y los revolucionarios y la de Cosette. Javert, queda en la comunidad de Cosette por su aparición con los delincuentes. Estas comunidades corresponden a los 5 volúmenes de la obra, titulados como estos personajes principales. Los personajes del Volumen IV están repartidos tanto en la comunidad de Marius como en la de Cosette, pues es el volumen sobre el inicio de la revolución.

La Figura(5.13) nos muestra el resultado para el tiempo $t = 1.26038$. Esta partición en 4 comunidades tiene un resultado menos interesante que el anterior, pero muestra la unión de las historias de Jean Valjean y Cosette, quienes pasan gran parte de su vida juntos.

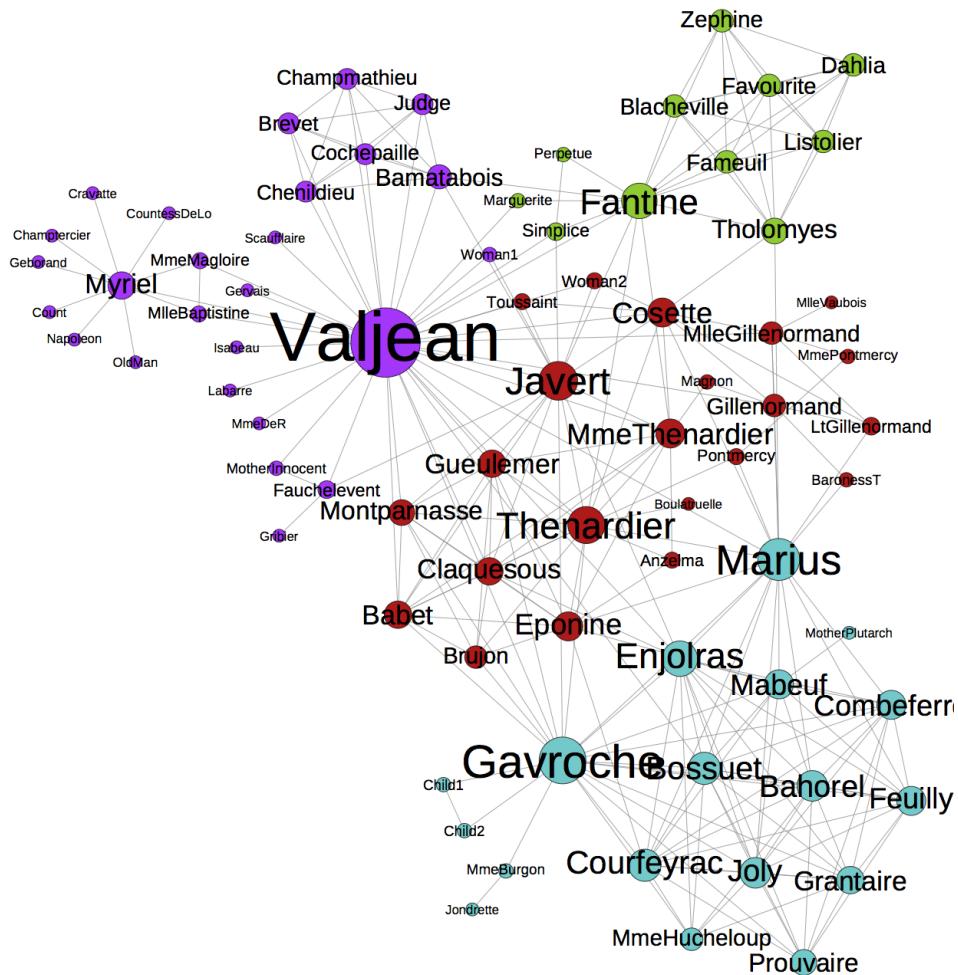


Figura 5.12: ‘Los Miserables’ 4 comunidades

5.3.8. Tres comunidades

Finalmente, encontramos 3 comunidades para dos tiempos de Markov considerados relevantes, ambos con caídas a $\widehat{VI} = 0$; en los tiempos $t =$

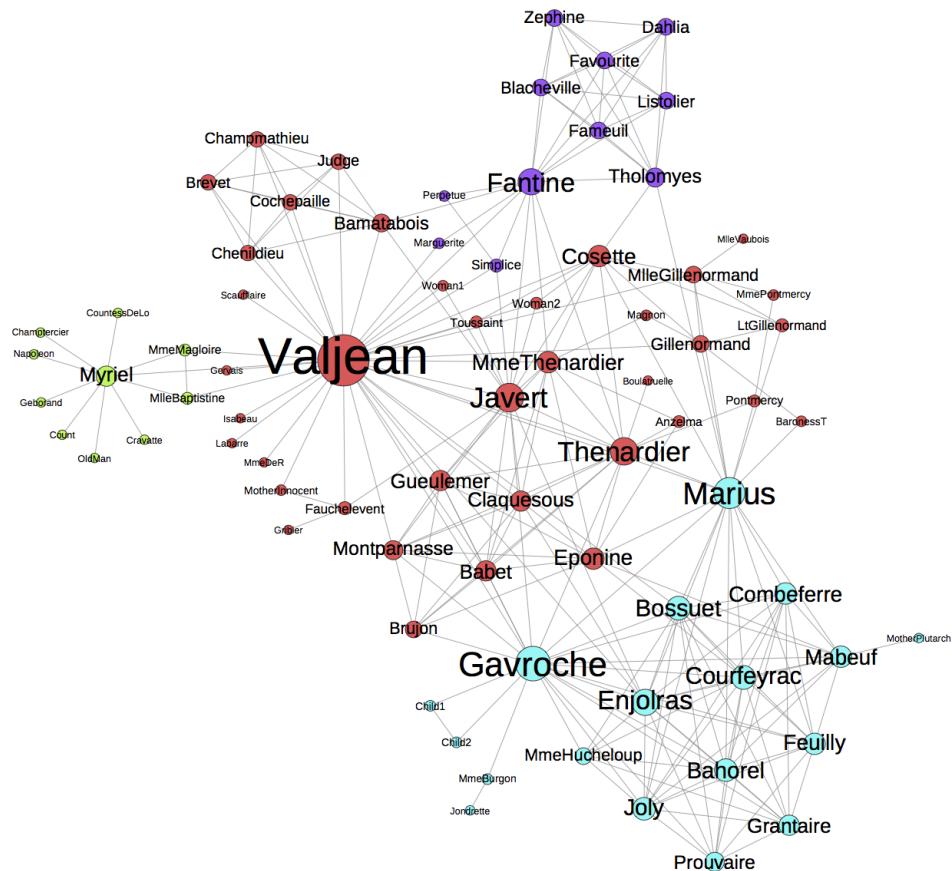


Figura 5.13: ‘Los Miserables’ 4 comunidades

1.72259 y $t = 1.91164$. La segunda de estas particiones no se reporta, ya que empeora nuestro análisis.

La Figura(5.14) nos muestra el resultado para el tiempo $t = 1.72259$. En ella observamos 3 comunidades. Se mantiene la comunidad de Fantine y la comunidad revolucionaria con Marius. Y por otro lado, se une la historia completa de Jean Valjean, con la de Cosette.

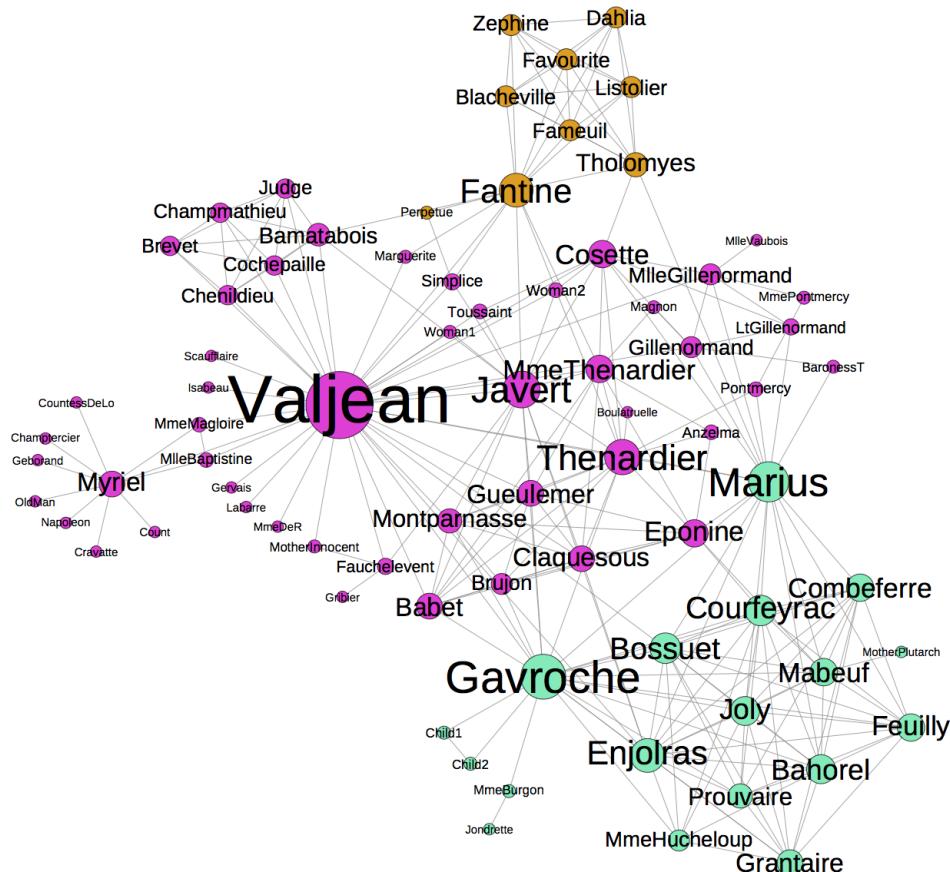


Figura 5.14: 'Los Miserables' 3 comunidades

Capítulo 6

Conclusiones

En este trabajo, se explicó el problema de detección de comunidades, así como algunos de los algoritmos que han sido utilizados para tratar de resolverlo. Se dijo también, que los resultados dependían del significado que se le diera a lo que uno espera encontrar en una comunidad, ya que estas estructuras no tienen una definición matemática definitiva. Posteriormente, se explicó la medida de modularidad, y se hizo énfasis en que la detección de comunidades basada en la maximización de la modularidad sufre de un límite de resolución para encontrar comunidades. Se explicaron después la caminata aleatoria y el proceso de difusión en una gráfica, para luego definir la medida de estabilidad a tiempo discreto, basado en la caminata y a tiempo continuo, basado en la difusión. Tras haber definido esta medida, se mostró la relación que tiene la estabilidad con la modularidad por dos razones principales; la primera, para justificar el uso del algoritmo de Louvain, creado para la modularidad, en la optimización de la estabilidad

y la segunda, porque la heurística de detección de comunidades basada en la estabilidad busca solucionar el límite de resolución de la modularidad. Finalmente, se explicó el algoritmo que se utilizó para maximizar la estabilidad, donde se introdujo la medida de variación de información. Utilizando este algoritmo, se ejemplificaron los resultados a distintas resoluciones que arroja utilizando la red de personajes de ‘Los Miserables’.

Con ayuda de los ejemplos prácticos de este método podemos concluir los siguientes puntos:

- El ejemplo de ‘Los Miserables’ es familiar para la mayoría de las personas, por lo que nos permitió justificar y corroborar los resultados a distintas resoluciones de esta heurística. La estructura de esta red es un buen ejemplo de una red social, de la que, si no conociéramos a los actores, podríamos esclarecer relaciones e interacciones.
- Asimismo, en este ejemplo, aunque prácticamente exploramos particiones de todos los tamaños, con el fin de demostrar la transición de una resolución a otra, podemos elegir solo algunas de estas resoluciones como verdaderamente informativas para analizar las interacciones sociales de los personajes.
 - Por ejemplo, en la partición en 11 comunidades, nos encontramos perfectamente delimitada a la célula criminal ‘Patron-Minette’ y a los niños de la barricada, liderados por Gavroche.
 - A partir de la partición en 4 comunidades, podemos encontrar juntos a todos aquellos que participaron en la revolución de las

barricadas, y, si observamos las resoluciones anteriores, podemos ver lo cerca que esta comunidad está de su antítesis aristócrata, con la familia de Marius.

Con estos ejemplos pudimos ver que distintas resoluciones nos ayudan a detectar distintos niveles de interacción. En este punto radica la bondad de este método en comparación de otros métodos de detección de comunidades, como la modularidad. Sin embargo, no podemos decir que sea un mejor método de detección de comunidades en cuanto a la rapidez con la que se llega a un resultado. Es claro que nuestro método es más costoso, y por consiguiente más lento. Si tomáramos en cuenta solamente un tiempo de Markov, el costo crece por el cómputo de la exponencial de la matriz para calcular $P(t)$ y que existen r inicios aleatorios de Louvain. Si tomamos en cuenta la componente exploratoria, esto crece, pues el cómputo se realiza para cada tiempo de Markov, y además hay que calcular la variación de información entre cada uno de los óptimos.

Otras aplicaciones posibles

En este trabajo se mostró la bondad de la resolución para redes sociales, que nos permitió agrupar personajes de una novela por distintos momentos de su vida o interacciones sociales. Sin embargo, no es este el único tipo de red en la que se puede sacar provecho de la resolución, otros ejemplos interesantes serían...

- Detección de temas en un corpus de documentos. Podría generarse una red de palabras por coaparición de palabras en un mismo docu-

mento, con lo que las palabras se agruparían en distintas comunidades. Y con ellas, podrían luego clasificarse los documentos en temas. De este modo, la resolución podría ayudar a detectar subtemas anidados dentro de temas más grandes.

- Detección de gustos en un catálogo de productos. Teniendo una red de compra de productos, donde una arista entre 2 productos está dado si una misma persona compró ambos productos, podríamos encontrar canastas de productos. Estas canastas pueden ayudar a definir tendencias en el caso de productos de moda. Y esto podría ayudar a crear un sistema de recomendación.

Trabajo futuro

Para este trabajo se implementó el algoritmo de detección de comunidades en *java*, y usando la versión no linealizada. Esto resulta en un incremento importante del costo computacional del método al crecer el número de nodos y arcos de la red. El algoritmo se intentó correr para una red de 5 mil productos y el tiempo que tomó la componente exploratoria fue de alrededor de 20 horas, usando 12Gb de RAM. Y, por las estructuras utilizadas para las matrices, la red original de 300 mil productos no pudo ser probada, ya que sobrepasaba el límite del tamaño de arreglo de *java*. Además del problema de memoria propio del lenguaje, la implementación de este algoritmo tarda mucho en arrojar resultados, sobre todo para la componente exploratoria.

Estos problemas podrían ser disminuidos utilizando el cómputo linea-

lizado de la estabilidad, eliminando la exploración de la \widehat{VI} , centrándose mejor en el número de comunidades obtenidas. Además, implementar el método en un ambiente paralelizado, o utilizando estructuras *sparse* para evitar sobrepasar la memoria de la máquina virtual, en el caso de java.

Apéndice A

Software y Reproducibilidad

Se puede encontrar una copia de este trabajo en:

[https://github.com/tmendozas/tesis.](https://github.com/tmendozas/tesis)

Ahí se encuentra el código para el algoritmo de optimización de estabilidad, el código para generar las gráficas y los datos utilizados en esta tesis.

Software y lenguajes utilizados:

- R
- L^AT_EX
- java
- gephi

Nota: La implementación de Louvain utilizada es una copia parcial del código de Ludo Waltman and Nees Jan van Eck, que puede encontrarse en:
<http://www.ludowaltman.nl/slm/>

Apéndice B

Notación

G	Gráfica con n nodos y m aristas
n	Número de nodos de G
m	Número de aristas de G
v_i	i -ésimo nodo o vértice de G
A	Matriz de adyacencias de G de tamaño $n \times n$
d_i	Grado del nodo v_i
d	Vector de grados, con componentes d_i $i = 1, \dots, n$
D	Matriz diagonal de grados $diag(d)$
k	Número de grupos de partición
p_k	k -ésima partición

n_k	Tamaño de la k-ésima partición
$p_i(t)$	Probabilidad de estar en v_i al tiempo t
$p(t)$	Vector de probabilidades de estado al tiempo t
π	Vector de la distribución estacionaria
M	Matriz de transición
S	Espacio de estados de la cadena de Markov.
L	Laplaciano de la gráfica $L = D - A$
$X(t)$	Vector aleatorio de un proceso de difusión en G
$X_i(t)$	Indicadora del estado de la difusión en v_i al tiempo t
$W(t)$	Matriz de Autocovarianza de $X(t)$
H	Matriz de partición en c comunidadesd e tamaño $n \times c$.
$Y(t)$	Vector aleatorio de un proceso de difusión en comunidades.
$R(t)$	Matriz de Autocovarianza de Conglomerados (<i>i.e.</i> de $Y(t)$)
$r(t; H)$	Estabilidad de la partición H al tiempo t
$M(t)$	Matriz de transición de t pasos
\mathcal{L}	Laplaciano normalizado

Apéndice C

Detalles

C.1. Gráfica Aleatoria de Erdös Rènyi

En la gráfica aleatoria, se tienen dos parámetros, el número de vértices, n y la probabilidad de conexión entre cualquier par de vértices p . Como mencionamos, la existencia de aristas es equiprobable y se distribuye independientemente, de manera que

$$P(e_{ij} = (v_i, v_j) \in E) = p \quad \forall v_i, v_j \in V$$

Si vemos a los grados de los vértices d_1, \dots, d_n como variables aleatorias, podemos ver que se distribuyen $\text{Bin}(n, p)$ ya que,

$$P(d_i = k) = \binom{n-1}{k} p^k (1-p)^{n-1-k}$$

Explicando la ecuación término a término, la probabilidad de que el nodo v_i esté conectado con k y no lo esté con los $n - 1 - k$ restantes está dada por $p^k(1-p)^{n-1-k}$ y $\binom{n-1}{k}$ indica el número de opciones que tenemos para que estas conexiones sucedan.

En el límite cuando n tiende a ∞ , sucede que la distribución de grados se distribuye Poisson con $\lambda = np$ donde una vez más n es el número de vértices y p la probabilidad de conexión de cualquier par de vértices.

Para gráficas donde n no es muy grande, el valor esperado de la distribución binomial, $\mu = np$, nos indica que el grado de los vértices de una gráfica construida con estos parámetros, es homogéneo entre los vértices. Es por esta razón que intuitivamente, las aristas están repartidas entre los nodos de manera tal que sería muy difícil encontrar conjuntos de vértices con mayor conexión entre ellas y menor conexión con otros vértices o conjuntos de vértices. Por eso, se menciona en el Capítulo 2 que las gráficas aleatorias no cuentan con estructura de comunidad.

Por otro lado, cuando n es muy grande y la distribución de grados Binomial converge a una Poisson, se tiene que la distribución de grados tampoco corresponde a la de una red con datos reales. Ya que, las redes con datos reales tienen una cola que sigue una ley de potencias, la cual decrece con menor rapidez que la cola de una exponencial, como la Poisson.

C.2. Eigenvectores y el Teorema Espectral

Definición C.1. Sea A una matriz real cuadrada de tamaño $n \times n$. Al vector $\mathbf{u} = (u_1, u_2, \dots, u_n)$ se le llama *eigenvector* de A si satisface la ecuación:

$$A\mathbf{u} = \lambda\mathbf{u}$$

Donde $\lambda \in \mathbb{R}$ es el *eigenvalor* correspondiente a \mathbf{u}

Es decir, los eigenvectores son aquellos vectores a los que la transfor-

mación lineal de A afecta con un reescalamiento, más no varía su dirección. También pueden llamárseles *vectores propios* o *vectores característicos*

Para el Teorema Espectral enunciaremos solo el teorema que resulta pertinente a este trabajo; el teorema que habla de la descomposición de matrices simétricas. Esto nos será útil para la matriz de adyacencias A utilizada a lo largo del trabajo, ya que al ser la matriz correspondiente a una gráfica no dirigida, es una matriz simétrica.

Teorema C.1. *Teorema Espectral* - Sea A una matriz simétrica. Existe una base ortonormal de eigenvectores de A con todos sus eigenvalores en los números reales.

Teorema C.2. *Descomposición Espectral* - Sea A una matriz simétrica y $\{q_1, q_2, \dots, q_n\}$ una base ortonormal de eigenvectores de A . Podemos descomponer a la matriz A como sigue:

$$A = Q\Lambda Q^T$$

Donde, Λ es la matriz diagonal con los eigenvalores $\lambda_1, \dots, \lambda_n$ en su diagonal y Q es la matriz que tiene por columnas a los vectores q_1, q_2, \dots, q_n

Bibliografía

- [¹] R. Lambiotte, J. C. Delvenne, and M. Barahona. Laplacian dynamics and multiscale modular structure in networks, Nov 2008.
- [²] Mark Newman. *Networks: An Introduction*. Oxford University Press, Inc., New York, NY, USA, 2010.
- [³] Marianne McKenzie. *Stability-optimization algorithms for the detection of community structure in networks*. PhD thesis, Keble College, University of Oxford, 2012.
- [⁴] M. E. J. Newman. Power laws, Pareto distributions and Zipf's law. *Contemporary Physics*, 2005.
- [⁵] B.W. Kernighan and S. Lin. An Efficient Heuristic Procedure for Partitioning Graphs. *The Bell Systems Technical Journal*, 49(2), 1970.
- [⁶] M. Sarzynska, E. A. Leicht, G. Chowell, and M. A. Porter. Null Models for Community Detection in Spatially-Embedded, Temporal Networks. *ArXiv e-prints*, 2014.
- [⁷] U. Brandes, D. Delling, M. Gaertler, R. Goerke, M. Hoefer, Z. Nikoloski, and D. Wagner. Maximizing Modularity is hard. *ArXiv Physics e-prints*, 2006.
- [⁸] Jean-Charles Delvenne, Michael T. Schaub, Sophia N. Yaliraki, and Mauricio Barahona. *Dynamics On and Of Complex Networks, Volume 2: Applications to Time-Varying Dynamical Systems*, chapter The Stability of a Graph Partition: A Dynamics-Based Framework for Community Detection, pages 221–242. Springer New York, New York, NY, 2013.

- [9] Laszlo Lovasz. Random walks on graphs: A survey, 1993.
- [10] J.-C. Delvenne, S. N. Yaliraki, and M. Barahona. Stability of graph communities across time scales. *Proceedings of the National Academy of Sciences*, 107(29):12755–12760, 2010.
- [11] M. Beguerisse-Díaz, B. Vangelov, and M. Barahona. Finding role communities in directed networks using role-based similarity, markov stability and the relaxed minimum spanning tree. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 937–940, Dec 2013.
- [12] Santo Fortunato and Marc Barthélemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41, 2007.
- [13] Vincent D Blondel, Jean loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks, 2008.
- [14] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486:75 – 174, 2010.
- [15] Mariano Beguerisse-Díaz, Guillermo Garduño-Hernandez, Borislav Vangelov, Sophia N. Yaliraki, and Mauricio Barahona. Interest communities and flow roles in directed networks: the twitter network of the uk riots. *Journal of The Royal Society Interface*, 11(101), 2014.
- [16] Animesh Mukherjee, Monojit Choudhury, Fernando Peruani, Niloy Ganguly, and Bivas Mitra. *Dynamics On and Of Complex Networks, Volume 2: Applications to Time-Varying Dynamical Systems*. Birkhäuser Basel, 2013.
- [17] John Hamal Hubbard and Beverly Henderson West. *Differential equations : a dynamical systems approach. [Part 2]. , Higher-dimensional systems*. Texts in applied mathematics. Springer, New York, 1995.
- [18] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69:026113, Feb 2004.
- [19] R. Lambiotte, J. C. Delvenne, and M. Barahona. Random walks, markov processes and the multiscale modular organization of complex networks.

- IEEE Transactions on Network Science and Engineering*, 1(2):76–90, July 2014.
- [20] Erwan Le Martelot and Chris Hankin. Multi-scale community detection using stability optimisation. *The International Journal of Web Based Communities (IJWBC) Special Issue on: Community Structure in Complex Networks*, 2012.
- [21] Mason A Porter, Jukka-Pekka Onnela, and Peter J Mucha. Communities in networks. *Notices of the AMS*, 56:1082–1097, 2009.
- [22] Ludo Waltman. A smart local moving algorithm for large-scale modularity-based community detection.