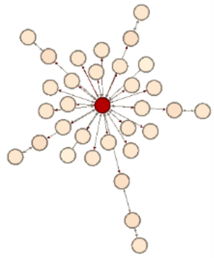
Jorge Fábrega

Redes Sociales



1

Objetivos de la clase

- Introducir aplicaciones estadísticas clave en teoría de redes que permiten modelar formación y evolución de vínculos.
- Mostrar cómo distintos modelos (ERGMs, Ising, EGA) capturan regularidades estructurales, dinámicas y latentes.
- Conectar intuiciones sociales con fundamentos matemáticos

2

¿Por qué estudiar estadísticas en redes?

- Las redes sociales no son aleatorias: muestran regularidades estructurales (reciprocidad, homofilia, cierre triádico).
- Los grafos empíricos son solo una muestra parcial: necesitamos inferir y generalizar.
- Nos interesa explicar cómo emergen configuraciones colectivas a partir de decisiones locales.
- Ante eso: La estadística nos permite:
  - Comparar la red observada con un mundo contrafactual (qué pasaría “por azar”).
  - Identificar mecanismos sociales ocultos detrás de los enlaces.

3

Tres enfoques principales

- **Estructurales**
  - Pregunta: ***¿Qué patrones locales explican la red?***
  - Ejemplos: ERGMs, block models.
  - Enfoque: regularidades en triángulos, reciprocidad, centralidad.
- **Dinámicos**
  - Pregunta: ***¿Cómo cambian los enlaces en el tiempo?***
  - Ejemplos: TERGMs, preferential attachment.
  - Enfoque: evolución de tríadas, coevolución de atributos y vínculos.
- **Latentes**
  - Pregunta: ***¿Qué estructuras invisibles generan la red?***
  - Ejemplos: Ising, Expected Graph Models (EGA).
  - Enfoque: detectar dimensiones ocultas (ideología, rasgos psicológicos, etc.).

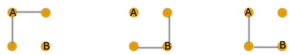
4

¿Qué patrones locales explican la red?

5

Exponential Random Graph Models

6



¿cuál es la probabilidad de que A sea amigo de B?

¿qué combinación de lazos de amistad en toda la clase es más probable, dadas las características de los nodos y de la red en su conjunto?

7

ERGMs

- Es el "equivalente" en redes a un general lineal model
- Expected Random Graph Model (ERGM) es una familia de modelos que permite testear hipótesis sobre qué explica la formación/características de una red.
  - Por ejemplo: Si para una red G, tu hipótesis implica que:
    - La cantidad de conexiones importa → testear densidad
    - La reciprocidad importa → testear relaciones no dirigidas
    - La formación de comunidades importa → testear tríadas
    - Los atributos de los nodos importa → testear su distribución, homofilia,...

    Etc...

8

La intuición

- "Es como" una regresión logística, pero para redes.
- Básicamente, buscamos probabilidad de que exista un link entre los nodos *i* y *j*, $P(a_{ij}=1)$ dado el resto de la estructura del grafo en las características que consideras que explican la red, *G*. Es decir: $P(a_{ij}=1|G)$
- El output de un ERGM es un odd-ratio para cada una de esas características del siguiente modo:

  $log\ (P(a_{ij}=1|G)\ /\ P(a_{ij}=0|G)\ )$
- Por ejemplo, si tu modelo tiene una sola característica, digamos, el número de links y el valor que obtuviste fue: -1.19, la probabilidad es 0,23
  - ¿Por qué? Porque es un odd-ratio, debes transformarlo: exp(-1,19)/[1+exp(-1,19)]

9

Ahora formalmente

- Sea *G=(V,E)* una red. Sea $Y_{ij} = Y_{ji}$ una variable binaria que adquiere valor 1 si $e_{ij} \in E$.
- Entonces $Y = [Y_{ij}]$ es la *matriz aleatoria adyacente* de *G*. Tal que $y = [y_{ij}]$ es una realización particular de dicha matriz.
- Un ERGM es la distribución de probabilidades conjunta de los elementos de $Y = y$:

$$P_\theta(Y = y) = \left(\frac{1}{k}\right) exp\left\{\sum_H \theta_H g_H(y)\right\}$$

- Donde: *H* es una "configuración" o set de posibles edges entre los vértices o nodos de *G*
- $g_H(y) = \prod_{y_{ij \in H}} y_{ij}$, por lo tanto es 1 si la configuración *H* ocurre y 0 si no.
- k es una constante de normalización: $k = k(\theta) = exp\{\sum_H \theta_H g_H(y)\}$

10

¿Por qué una exponencial?

- Porque las distribuciones exponenciales tienen una característica muy atractiva: NO TIENEN MEMORIA.

(una breve explicación aquí: https://llc.stat.purdue.edu/2014/41600/notes/prob3204.pdf)

- Entonces, podemos estudiar las probabilidades condicionales de cada link independiente de los otros, "como si" fuesen una cadena de eventos independientes.

11

Teorema de Hammersley–Clifford (1971)

- Toda distribución de probabilidad puede representarse como una cadena de Markov si cumple con ciertos requisitos en la distribución de probabilidades.
- Las distribuciones exponenciales cumplen con esas propiedades.
- Implicancia: toda familia de modelos de redes puede ser expresado (potencialmente) como un ERGM si se cuenta con las estadísticas fundamentales (S1,S2,...) de dicha familia de modelos

$$Pr(g) = \frac{e^{\theta_1 S_1(g) + \theta_2 S_2(g) + \cdots + \theta_k S_k(g)}}{\sum_{g' \epsilon G} e^{\theta_1 S_1(g') + \theta_2 S_2(g') + \cdots + \theta_k S_k(g')}}$$

12

MCMC: Markov Chain Monte Carlo

- En redes muy pequeñas ($n < 8$) es posible estimar todas las configuraciones para un set de hipótesis/configuraciones (véase paquete *ergmito* en R)
- En redes más grandes deben hacerse aproximaciones estadísticas
- MCMC
  - Markov Chain: Sólo el estado inmediatamente anterior importa (separar estimación en bloques)
  - Monte Carlo: Aleatorización para sacar estimaciones de parámetros
  - Juntos: Dados los valores de estadísticos de la red, simulaciones sucesivas para buscar la distribución que mejor explica las estadísticas observadas

13

Pasos para la estimación vía ERGMs

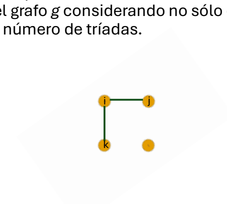
- Paso 1: Cada link es tratado como una variable aleatoria
- Paso 2: Un conjunto de hipótesis se postula para explicar formación de links
  - Ejemplo: links aleatorios, links por cercanía, por homofilia, triadic closure, etc…
- Paso 3: Cada hipótesis es una configuración que se suma a otras posibles configuraciones.
  - Ejemplo: El link *ij* puede ser por aleatoridad + por cercanía + por homofilia, …
- Paso 4: Las configuraciones, aunque locales, son homogéneas en toda la red (reducción de parámetros)
- Paso 5: Estimación e interpretación de los parámetros del modelo

14

Ejemplo

- Supongamos que queremos estudiar simultáneamente la probabilidad del grafo *g* considerando no sólo el número de links sino también el número de tríadas.

1/15



15

Ejemplo

Aplicado a una red de Erdös-Renyi (en la que todos los links son independientes unos de otros, i.e. θ_T=0):

$$Pr(g) = p^m(1-p)^{\binom{n}{2}-m} = \frac{p^m}{(1-p)^m} * (1-p)^{\binom{n}{2}}$$

$$Pr(g) = e^{\log\left(\frac{p}{(1-p)}\right)*m+constante}$$

$Pr(g) = e^{\theta_m m(g)+constante}$ sobre todos los posibles grafos *g'*

16

Ejemplo

- Entonces, sea *m(g)* el número de links y *T(g)* el número de tríadas de la red *g*:

La probabilidad del grafo *g* depende de:

$$\theta_m m(g) + \theta_T T(g)$$

$$\Downarrow$$

$$e^{\theta_m m(g) + \theta_T T(g)}$$

$$\Downarrow$$

$$Pr(g) = \frac{e^{\theta_m m(g) + \theta_T T(g)}}{\sum_{g' \epsilon G} e^{\theta_m m(g') + \theta_T T(g')}} \longleftarrow \text{ERGM}$$

17

¿Cómo se hace en la práctica?

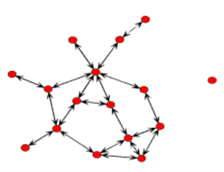
En R se hace con el paquete statnet (compilado de varios paquetes)
Referencias: https://statnet.org/trac

Algoritmo:
1. Se estiman $\theta_1, \theta_2, \theta_3$ ... (varios métodos, maximum likelihood)
2. Se simulan redes con propiedades $\theta_1, \theta_2, \theta_3$ ...
3. Se actualizan los valores de $\theta_1, \theta_2, \theta_3$ ... a partir de esas redes simuladas
4. Se repite 2... hasta lograr convergencia

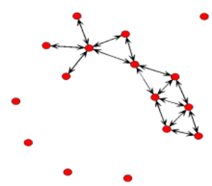
18

Ejemplo: Medici

Matrimonios



Negocios



20 links de 120 posibles en teoría

19

1.479

Medici – ergm en base al número de links

θ_m

```
==========================
Summary of model fit
==========================
Formula:   flomarriage ~ edges
Iterations:  5 out of 20
Monte Carlo MLE Results:
      Estimate Std. Error MCMC % p-value
edges  -1.6094   0.2449     0  <1e-04 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
   Null Deviance: 166.4  on 120  degrees of freedom
 Residual Deviance: 108.1  on 119  degrees of freedom
 AIC: 110.1    BIC: 112.9    (Smaller is better.)
```

Odd ratio = -1,609 * 1 (cambio de 0 a 1)

$$\Pr(l_{ij} = 1) = \frac{e^{-1{,}609}}{1 + e^{-1{,}609}} = 0{,}1667$$

... que es justamente lo que esperábamos porque:

20 links reales /120 posibles = 0,1667

20

Medici – ergm en base al número de links + tríadas

```
==========================
Summary of model fit
==========================
Formula:   flomarriage ~ edges + triangle
Iterations:  2 out of 20
Monte Carlo MLE Results:
         Estimate Std. Error MCMC % p-value
edges     -1.6603     0.3556      0  <1e-04 ***
triangle   0.1300     0.5944      0   0.827
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
   Null Deviance: 166.4  on 120  degrees of freedom
 Residual Deviance: 108.1  on 118  degrees of freedom
 AIC: 112.1    BIC: 117.6    (Smaller is better.)
```

Odd ratio = -1,609 * 1 + 0,13*T

Si T = 0 → odd ratio: -1,609

Si T = 1 → odd ratio: -1,609+0,13=-1,47

$$\Pr(l_{ij} = 1 | T = 1) = 0{,}185$$

... y sube en aprox.2 pts por tríada

21

```
==========================
Summary of model fit
==========================
Formula:   flomarriage ~ edges + triangle + nodecov("wealth")
Iterations:  2 out of 20
Monte Carlo MLE Results:
               Estimate Std. Error MCMC % p-value
edges       -2.580037   0.536265     0 <1e-04 ***
triangle     -0.116286   0.620783     0  0.8517
nodecov.wealth  0.010886   0.005018     0  0.0321 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
    Null Deviance: 166.4  on 120  degrees of freedom
 Residual Deviance: 103.1  on 117  degrees of freedom
 AIC: 109.1    BIC: 117.4    (Smaller is better.)
```

Como *wealth* son atributos de nodos, se consideran ambas riquezas al ver la probabilidad de cada link *ij*.

Wealth:

[1] 10 36 55 44 20 32 8 42 103 48 49 3 27 10 146 48

Ej: link con una tríada entre familias con riqueza 10 y 146:

-2,58 – 0,11 + 0,01*(10+146) = -1,13

En prob: 0,24

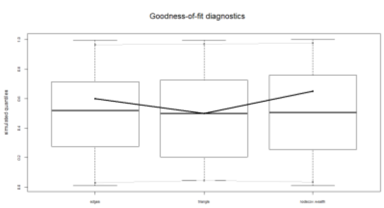
Este modelo sugiere mayor probabilidad de un link a mayor riqueza de ambas familias

22

Goodness of fit en los Medici

¿el modelo cómo se ajusta a los datos?

A partir del modelo se simulan redes y se calculan las mismas variables del modelo

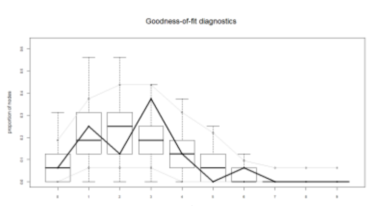


23

Goodness of fit en los Medici

Y también otras que son descriptores básicos de la red, pero no estaban en el modelo:

Distribución de grado

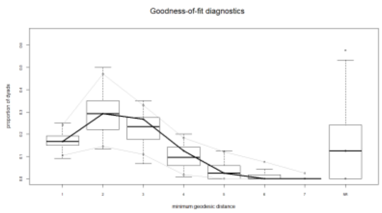


24

Goodness of fit en los Medici

Y también otras que son descriptores básicos de la red, pero no estaban en el modelo:

Geodésicas

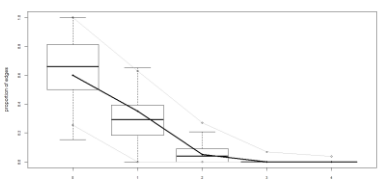


25

Goodness of fit en los Medici

Y también otras que son descriptores básicos de la red, pero no estaban en el modelo:

Vecinos compartidos



26

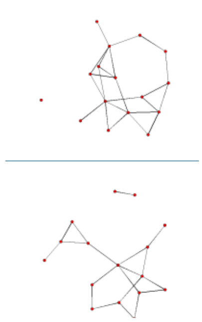
Simulaciones de redes

- Para poder hacer GOF fue necesario hacer simulaciones.

Stored network statistics:

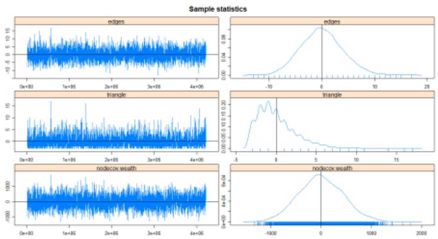
	edges	triangle	nodecov.wealth
[1,]	23	5	2374
[2,]	20	4	1961
[3,]	22	4	2585
[4,]	19	0	2092
[5,]	18	0	1494
[6,]	20	5	2378
[7,]	21	2	2308
[8,]	19	2	1661

...



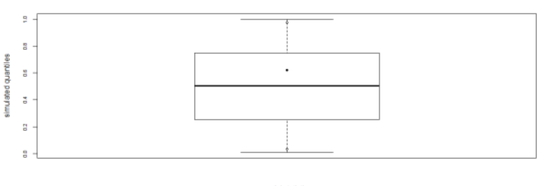
27

Diagnóstico



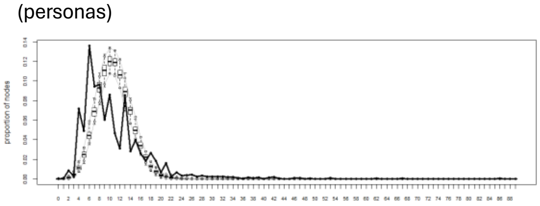
28

Un grafo aleatorio ajustado a los datos de interlocking (personas)



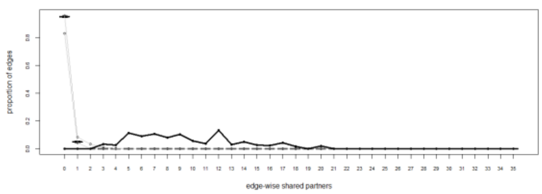
29

Un grafo aleatorio ajustado a los datos de interlocking (personas)



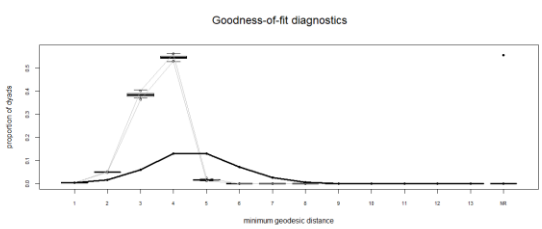
30

Un grafo aleatorio ajustado a los datos de interlocking (personas)



31

Un grafo aleatorio ajustado a los datos de interlocking (personas)



32

Referencias

- https://statnet.org/trac
- Buscar también p* models

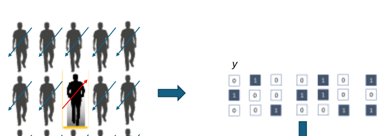
33

¿Qué estructuras invisibles generan la red?

34

Ising model

35

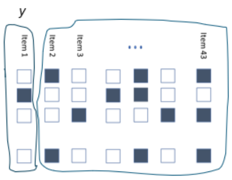


$$P(Y = y) = \frac{1}{Z} e^{-\beta H(y)}$$

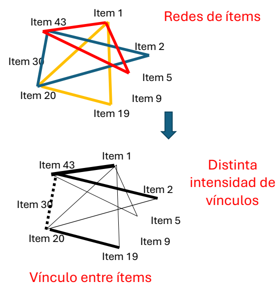
$$H(y) = -\sum_i t_i y_i - \sum_{ij} w_{ij} y_i y_j$$

Modelo de Ising

36



Nodos: Items son variables aleatorias
Links: Asociación condicional entre ítems



37

Exploratory Graph Analysis (EGA)

	Ansiedad	Fatiga	Preocupación	Insomnio	Anhedonia ...
Individuo 1	1	2	2	1	1
Individuo 2	3	2	3	3	2
Individuo 3	4	1	4	7	3
Indivi...					

Matriz X de n*p → Matriz de covarianzas o de correlaciones o alguna otra forma de asociación $\mathbf{R} = [r_{ij}]_{i,j=1}^{p} \xrightarrow{\text{GLASSO}} \mathbf{\Omega} = \mathbf{R}^{-1}$

Matriz de precisión Con alto p y poco n sobreajusta, por eso se regulariza (básicamente llevar a cero correlaciones bajas)

	Ansiedad	Fatiga	Preocupación	Insomnio	Anhedonia ...
Ansiedad	w11	w12	w13	w14	w15
Fatiga	w21	w22	w23	w24	w25
Preocupación	w31	w32	w33	w34	w35
Insom...					

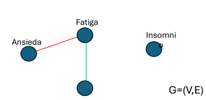
$$\rho_{ij|\text{resto}} = -\frac{\omega_{ij}}{\sqrt{\omega_{ii}\omega_{jj}}}$$

38

	Ansiedad	Fatiga	Preocupación	Insomnio	Anhedonia ...
Ansiedad	w11	w12	w13	w14	w15
Fatiga	w21	w22	w23	w24	w25
Preocupación	w31	w32	w33	w34	w35
Insom...					

$$\rho_{ij|\text{resto}} = -\frac{\omega_{ij}}{\sqrt{\omega_{ii}\omega_{jj}}}$$

	Ansiedad	Fatiga	Preocupación...
Ansiedad	-	-0.1	0
Fatiga	-0.1	-	0.8
Preocupación	0	0.8	-
Insom...			



Mecanismos para detectar comunidades e importancia relativa de nodos (centralidades). Las comunidades son interpretadas como "factores" o variables latentes

39

Ejemplo

$$\rho_{ij|\text{resto}} = -\frac{\omega_{ij}}{\sqrt{\omega_{ii}\omega_{jj}}}$$

	Anhedonia	Insomnio	Fatiga	Irritabilidad	Ansiedad	Preocupacion	Nerviosismo	Inquietud
Anhedonia	0.00000000	0.33826200	0.609789545	0.000000000	-0.01153948	-0.03252217	-0.01526122	-0.05895483
Insomnio	0.33826200	0.00000000	0.235124011	0.042614595	0.05408452	0.10632328	0.05728066	0.07648377
Fatiga	0.60978954	0.23512401	0.000000000	-0.008633799	0.00000000	0.00000000	0.00000000	-0.05244861
Irritabilidad	0.00000000	0.04261459	-0.008633799	0.000000000	0.21940004	0.17640901	0.23452143	0.22784454
Ansiedad	-0.01153948	0.05408452	0.000000000	0.219400042	0.00000000	0.32521116	0.22713263	0.18962141
Preocupacion	-0.03252217	0.10632328	0.000000000	0.176409014	0.32521116	0.00000000	0.26009430	0.13346242
Nerviosismo	-0.01526122	0.05728066	0.000000000	0.234521430	0.22713263	0.26009430	0.00000000	0.28238449
Inquietud	-0.05895483	0.07648377	-0.052448609	0.227844540	0.18962141	0.13346242	0.28238449	0.00000000

40

Ejemplo

```
Model: GLASSO (EBIC with gamma = 0.5)
Correlations: auto
Lambda: 0.0898249126041102 (n = 100, ratio = 0.1)

Number of nodes: 8
Number of edges: 24
Edge density: 0.857

Non-zero edge weights:
     M    SD    Min   Max
 0.151 0.158 -0.059 0.610

----

Algorithm:  Walktrap

Number of communities:  2

    Anhedonia      Insomnio        Fatiga Irritabilidad      Ansiedad  Preocupacion   Nerviosismo
            1             1             1             2             2             2             2
    Inquietud
            2

----

Unidimensional Method: Louvain
Unidimensional: No

----

TEFI: -5.576
```

41

Ejemplo

Configuración del modelo:
Indica la regularización hecha
Gamma – complejidad aceptada
Lambda – penalización utilizada


```
Model: GLASSO (EBIC with gamma = 0.5)
Correlations: auto
Lambda: 0.0898249126041102 (n = 100, ratio = 0.1)

Number of nodes: 8
Number of edges: 24
Edge density: 0.857

Non-zero edge weights:
     M    SD    Min   Max
 0.151 0.158 -0.059 0.610

----

Algorithm:  Walktrap

Number of communities:  2

    Anhedonia      Insomnio        Fatiga Irritabilidad      Ansiedad  Preocupacion   Nerviosismo
            1             1             1             2             2             2             2
    Inquietud
            2

----

Unidimensional Method: Louvain
Unidimensional: No

----

TEFI: -5.576
```


42

Ejemplo

Descripción de la red

```
Model: GLASSO (EBIC with gamma = 0.5)
Correlations: auto
Lambda: 0.0898249126041102 (n = 100, ratio = 0.1)

Number of nodes: 8
Number of edges: 24
Edge density: 0.857

Non-zero edge weights:
     M    SD    Min   Max
 0.151 0.158 -0.059 0.610

----

Algorithm:  Walktrap

Number of communities:  2

    Anhedonia      Insomnio        Fatiga Irritabilidad      Ansiedad  Preocupacion   Nerviosismo
            1             1             1             2             2             2             2
    Inquietud
            2

----

Unidimensional Method: Louvain
Unidimensional: No

----

TEFI: -5.576
```

43

Ejemplo

Peso de las aristas


```
Model: GLASSO (EBIC with gamma = 0.5)
Correlations: auto
Lambda: 0.0898249126041102 (n = 100, ratio = 0.1)

Number of nodes: 8
Number of edges: 24
Edge density: 0.857

Non-zero edge weights:
     M    SD    Min   Max
 0.151 0.158 -0.059 0.610

----

Algorithm:  Walktrap

Number of communities:  2

    Anhedonia      Insomnio        Fatiga Irritabilidad      Ansiedad  Preocupacion   Nerviosismo
            1             1             1             2             2             2             2
    Inquietud
            2

----

Unidimensional Method: Louvain
Unidimensional: No

----

TEFI: -5.576
```


44

Ejemplo

Detección de comunidades y asignación de síntomas a las comunidade


```
Model: GLASSO (EBIC with gamma = 0.5)
Correlations: auto
Lambda: 0.0898249126041102 (n = 100, ratio = 0.1)

Number of nodes: 8
Number of edges: 24
Edge density: 0.857

Non-zero edge weights:
     M    SD    Min   Max
 0.151 0.158 -0.059 0.610

----

Algorithm:  Walktrap

Number of communities:  2

    Anhedonia      Insomnio        Fatiga Irritabilidad      Ansiedad  Preocupacion   Nerviosismo
            1             1             1             2             2             2             2
    Inquietud
            2

----

Unidimensional Method: Louvain
Unidimensional: No

----

TEFI: -5.576
```


45

Ejemplo

```
Model: GLASSO (EBIC with gamma = 0.5)
Correlations: auto
Lambda: 0.0898249126041102 (n = 100, ratio = 0.1)

Number of nodes: 8
Number of edges: 24
Edge density: 0.857

Non-zero edge weights:
     M    SD    Min   Max
 0.151 0.158 -0.059 0.610

----

Algorithm:  Walktrap

Number of communities:  2

    Anhedonia      Insomnio        Fatiga Irritabilidad      Ansiedad  Preocupacion   Nerviosismo
            1             1             1             2             2             2             2
    Inquietud
            2

----

Unidimensional Method: Louvain
Unidimensional: No

----

TEFI: -5.576
```

Test de unidimensionalidad

46

Ejemplo

```
Model: GLASSO (EBIC with gamma = 0.5)
Correlations: auto
Lambda: 0.0898249126041102 (n = 100, ratio = 0.1)

Number of nodes: 8
Number of edges: 24
Edge density: 0.857

Non-zero edge weights:
     M    SD    Min   Max
 0.151 0.158 -0.059 0.610

----

Algorithm:  Walktrap

Number of communities:  2

    Anhedonia      Insomnio        Fatiga Irritabilidad      Ansiedad  Preocupacion   Nerviosismo
            1             1             1             2             2             2             2
    Inquietud
            2

----

Unidimensional Method: Louvain
Unidimensional: No

----

TEFI: -5.576
```

Total Entropy Fit Index: mientras más negativo mejor ajuste

47

Volviendo al punto del algoritmo walktrap…

- El 4to paso era detectar comunidades y nodos relevantes…

Detección de comunidades y asignación de síntomas a las comunidades

```
Model: GLASSO (EBIC with gamma = 0.5)
Correlations: auto
Lambda: 0.0898249126041102 (n = 100, ratio = 0.1)

Number of nodes: 8
Number of edges: 24
Edge density: 0.857

Non-zero edge weights:
     M    SD    Min   Max
 0.151 0.158 -0.059 0.610

----

Algorithm:  Walktrap

Number of communities:  2

    Anhedonia      Insomnio        Fatiga Irritabilidad      Ansiedad  Preocupacion   Nerviosismo
            1             1             1             2             2             2             2
    Inquietud
            2

----

Unidimensional Method: Louvain
Unidimensional: No

----

TEFI: -5.576
```

48

Detección de comunidades

- Básicamente encontrar dónde están las densidades locales más fuertes y un criterio para separar nodos en subgrafos.
- El principal: Modularidad → maximizar Q:

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

1 si ambos están en la misma comunidad
0 en caso contrario

- Algoritmo más popular: Louvain

49

Detección de comunidades

- Otros algoritmos frecuentes
- Walktrap
  - Intuición: Caminatas aleatorias en el grafo deberían rescatar cuando te estás dando vuelta en un mismo vecindario.
  - Se calculan las distancias entre todos los pares de nodos usando caminatas aleatorias.
  - Los nodos cercanos según esta distancia se agrupan iterativamente, creando una jerarquía de posibles particiones.
- Infomap
  - Objetivo: encontrar la partición de la red que comprime mejor la información de la caminata aleatoria

50