Assortativity and Mixing

Last updated: 2022/08/28, 08:34:20 EDT

Principles of Complex Systems, Vols. 1, 2, & 3D
CSYS/MATH 300, 303, & 394, 2022–2023 | @pocsvox

Prof. Peter Sheridan Dodds | @peterdodds

Computational Story Lab | Vermont Complex Systems Center
Santa Fe Institute | University of Vermont

These slides are brought to you by:

These slides are also brought to you by:

Special Guest Executive Producer

On Instagram at pratchett_the_cat

Outline

Basic idea:

- Random networks with arbitrary degree distributions cover much territory but do not represent all networks.

Basic idea:

- Random networks with arbitrary degree distributions cover much territory but do not represent all networks.
- Moving away from pure random networks was a key first step.

Basic idea:

- Random networks with arbitrary degree distributions cover much territory but do not represent all networks.
- Moving away from pure random networks was a key first step.
- We can extend in many other directions and a natural one is to introduce correlations between different kinds of nodes.

Basic idea:

- Random networks with arbitrary degree distributions cover much territory but do not represent all networks.
- Moving away from pure random networks was a key first step.
- We can extend in many other directions and a natural one is to introduce correlations between different kinds of nodes.
- Node attributes may be anything, e.g.:
  1. degree
  2. demographics (age, gender, etc.)
  3. group affiliation

Basic idea:

- Random networks with arbitrary degree distributions cover much territory but do not represent all networks.
- Moving away from pure random networks was a key first step.
- We can extend in many other directions and a natural one is to introduce correlations between different kinds of nodes.
- Node attributes may be anything, e.g.:
  1. degree
  2. demographics (age, gender, etc.)
  3. group affiliation
- We speak of mixing patterns, correlations, biases...

Basic idea:

- Random networks with arbitrary degree distributions cover much territory but do not represent all networks.
- Moving away from pure random networks was a key first step.
- We can extend in many other directions and a natural one is to introduce correlations between different kinds of nodes.
- Node attributes may be anything, e.g.:
  1. degree
  2. demographics (age, gender, etc.)
  3. group affiliation
- We speak of mixing patterns, correlations, biases...
- Networks are still random at base but now have more global structure.

Basic idea:

- Random networks with arbitrary degree distributions cover much territory but do not represent all networks.
- Moving away from pure random networks was a key first step.
- We can extend in many other directions and a natural one is to introduce correlations between different kinds of nodes.
- Node attributes may be anything, e.g.:
  1. degree
  2. demographics (age, gender, etc.)
  3. group affiliation
- We speak of mixing patterns, correlations, biases...
- Networks are still random at base but now have more global structure.
- Build on work by Newman [5, 6], and Boguñá and Serano. [1].

General mixing between node categories

General mixing between node categories

- Assume types of nodes are countable, and are assigned numbers 1, 2, 3, ...

General mixing between node categories

- Assume types of nodes are countable, and are assigned numbers 1, 2, 3, ...
- Consider networks with directed edges.

General mixing between node categories

- Assume types of nodes are countable, and are assigned numbers 1, 2, 3, ...
- Consider networks with directed edges.

$$e_{\mu\nu} = \mathbf{Pr}\left(\begin{array}{l}\text{an edge connects a node of type } \mu \\ \text{to a node of type } \nu\end{array}\right)$$

General mixing between node categories

- Assume types of nodes are countable, and are assigned numbers 1, 2, 3, ...
- Consider networks with directed edges.

$$e_{\mu\nu} = \mathbf{Pr}\left(\begin{array}{l} \text{an edge connects a node of type } \mu \\ \text{to a node of type } \nu \end{array} \right)$$

$$a_{\mu} = \mathbf{Pr}(\text{an edge comes from a node of type } \mu)$$

General mixing between node categories

- Assume types of nodes are countable, and are assigned numbers 1, 2, 3, ...
- Consider networks with directed edges.

$$e_{\mu\nu} = \mathbf{Pr}\left(\begin{array}{l} \text{an edge connects a node of type } \mu \\ \text{to a node of type } \nu \end{array} \right)$$

$$a_{\mu} = \mathbf{Pr}(\text{an edge comes from a node of type } \mu)$$

$$b_{\nu} = \mathbf{Pr}(\text{an edge leads to a node of type } \nu)$$

General mixing between node categories

- Assume types of nodes are countable, and are assigned numbers 1, 2, 3, ...
- Consider networks with directed edges.

$$e_{\mu\nu} = \mathbf{Pr}\left(\begin{array}{l}\text{an edge connects a node of type } \mu \\ \text{to a node of type } \nu\end{array}\right)$$

$$a_{\mu} = \mathbf{Pr}(\text{an edge comes from a node of type } \mu)$$

$$b_{\nu} = \mathbf{Pr}(\text{an edge leads to a node of type } \nu)$$

- Write $\mathbf{E} = [e_{\mu\nu}]$, $\vec{a} = [a_{\mu}]$, and $\vec{b} = [b_{\nu}]$.

General mixing between node categories

- Assume types of nodes are countable, and are assigned numbers 1, 2, 3, ...
- Consider networks with directed edges.

$$e_{\mu\nu} = \mathbf{Pr}\left(\begin{array}{l}\text{an edge connects a node of type } \mu \\ \text{to a node of type } \nu\end{array}\right)$$

$$a_{\mu} = \mathbf{Pr}(\text{an edge comes from a node of type } \mu)$$

$$b_{\nu} = \mathbf{Pr}(\text{an edge leads to a node of type } \nu)$$

- Write $\mathbf{E} = [e_{\mu\nu}]$, $\vec{a} = [a_{\mu}]$, and $\vec{b} = [b_{\nu}]$.
- Requirements:

$$\sum_{\mu}\sum_{\nu} e_{\mu\nu} = 1, \ \sum_{\nu} e_{\mu\nu} = a_{\mu}, \text{ and } \sum_{\mu} e_{\mu\nu} = b_{\nu}.$$

Notes:

- Varying $e_{\mu\nu}$ allows us to move between the following:

Notes:

- Varying $e_{\mu\nu}$ allows us to move between the following:
  1. Perfectly assortative networks where nodes only connect to like nodes, and the network breaks into subnetworks.

Notes:

- Varying $e_{\mu\nu}$ allows us to move between the following:
  1. Perfectly assortative networks where nodes only connect to like nodes, and the network breaks into subnetworks.
     Requires $e_{\mu\nu} = 0$ if $\mu \neq \nu$ and $\sum_{\mu} e_{\mu\mu} = 1$.

Notes:

- Varying $e_{\mu\nu}$ allows us to move between the following:
  1. Perfectly assortative networks where nodes only connect to like nodes, and the network breaks into subnetworks.
     Requires $e_{\mu\nu} = 0$ if $\mu \neq \nu$ and $\sum_{\mu} e_{\mu\mu} = 1$.
  2. Uncorrelated networks (as we have studied so far)

Notes:

- Varying $e_{\mu\nu}$ allows us to move between the following:
  1. Perfectly assortative networks where nodes only connect to like nodes, and the network breaks into subnetworks.
     Requires $e_{\mu\nu} = 0$ if $\mu \neq \nu$ and $\sum_{\mu} e_{\mu\mu} = 1$.
  2. Uncorrelated networks (as we have studied so far)
     For these we must have independence:
     $e_{\mu\nu} = a_{\mu} b_{\nu}$.

Notes:

- Varying $e_{\mu\nu}$ allows us to move between the following:
  1. Perfectly assortative networks where nodes only connect to like nodes, and the network breaks into subnetworks.
     Requires $e_{\mu\nu} = 0$ if $\mu \neq \nu$ and $\sum_{\mu} e_{\mu\mu} = 1$.
  2. Uncorrelated networks (as we have studied so far) For these we must have independence: $e_{\mu\nu} = a_{\mu} b_{\nu}$.
  3. Disassortative networks where nodes connect to nodes distinct from themselves.

Notes:

- Varying $e_{\mu\nu}$ allows us to move between the following:
  1. Perfectly assortative networks where nodes only connect to like nodes, and the network breaks into subnetworks.
     Requires $e_{\mu\nu} = 0$ if $\mu \neq \nu$ and $\sum_{\mu} e_{\mu\mu} = 1$.
  2. Uncorrelated networks (as we have studied so far)
     For these we must have independence:
     $e_{\mu\nu} = a_{\mu} b_{\nu}$.
  3. Disassortative networks where nodes connect to nodes distinct from themselves.
- Disassortative networks can be hard to build and may require constraints on the $e_{\mu\nu}$.

Notes:

- Varying $e_{\mu\nu}$ allows us to move between the following:
  1. Perfectly assortative networks where nodes only connect to like nodes, and the network breaks into subnetworks.
     Requires $e_{\mu\nu} = 0$ if $\mu \neq \nu$ and $\sum_{\mu} e_{\mu\mu} = 1$.
  2. Uncorrelated networks (as we have studied so far)
     For these we must have independence:
     $e_{\mu\nu} = a_{\mu} b_{\nu}$.
  3. Disassortative networks where nodes connect to nodes distinct from themselves.
- Disassortative networks can be hard to build and may require constraints on the $e_{\mu\nu}$.
- Basic story: level of assortativity reflects the degree to which nodes are connected to nodes within their group.

Correlation coefficient:

- Quantify the level of assortativity with the following assortativity coefficient [6]:

$$r = \frac{\sum_{\mu} e_{\mu\mu} - \sum_{\mu} a_{\mu} b_{\mu}}{1 - \sum_{\mu} a_{\mu} b_{\mu}} = \frac{\text{Tr}\,\mathbf{E} - ||E^2||_1}{1 - ||E^2||_1}$$

where $||\cdot||_1$ is the 1-norm = sum of a matrix's entries.

Correlation coefficient:

- Quantify the level of assortativity with the following assortativity coefficient [6]:

$$r = \frac{\sum_\mu e_{\mu\mu} - \sum_\mu a_\mu b_\mu}{1 - \sum_\mu a_\mu b_\mu} = \frac{\mathrm{Tr}\,\mathbf{E} - ||E^2||_1}{1 - ||E^2||_1}$$

  where $||\cdot||_1$ is the 1-norm = sum of a matrix's entries.
- Tr **E** is the fraction of edges that are within groups.

Correlation coefficient:

- Quantify the level of assortativity with the following assortativity coefficient [6]:

$$r = \frac{\sum_\mu e_{\mu\mu} - \sum_\mu a_\mu b_\mu}{1 - \sum_\mu a_\mu b_\mu} = \frac{\mathrm{Tr}\,\mathbf{E} - ||E^2||_1}{1 - ||E^2||_1}$$

where $||\cdot||_1$ is the 1-norm = sum of a matrix's entries.

- $\mathrm{Tr}\,\mathbf{E}$ is the fraction of edges that are within groups.
- $||E^2||_1$ is the fraction of edges that would be within groups if connections were random.

Correlation coefficient:

- Quantify the level of assortativity with the following assortativity coefficient [6]:

$$r = \frac{\sum_\mu e_{\mu\mu} - \sum_\mu a_\mu b_\mu}{1 - \sum_\mu a_\mu b_\mu} = \frac{\text{Tr}\,\mathbf{E} - ||E^2||_1}{1 - ||E^2||_1}$$

where $||\cdot||_1$ is the 1-norm = sum of a matrix's entries.

- Tr **E** is the fraction of edges that are within groups.
- $||E^2||_1$ is the fraction of edges that would be within groups if connections were random.
- $1 - ||E^2||_1$ is a normalization factor so $r_{\text{max}} = 1$.

Correlation coefficient:

- Quantify the level of assortativity with the following assortativity coefficient [6]:

$$r = \frac{\sum_\mu e_{\mu\mu} - \sum_\mu a_\mu b_\mu}{1 - \sum_\mu a_\mu b_\mu} = \frac{\text{Tr}\,\mathbf{E} - ||E^2||_1}{1 - ||E^2||_1}$$

where $||\cdot||_1$ is the 1-norm = sum of a matrix's entries.

- $\text{Tr}\,\mathbf{E}$ is the fraction of edges that are within groups.
- $||E^2||_1$ is the fraction of edges that would be within groups if connections were random.
- $1 - ||E^2||_1$ is a normalization factor so $r_{\text{max}} = 1$.
- When $\text{Tr}\, e_{\mu\mu} = 1$, we have $r = 1$. ✔

Correlation coefficient:

- Quantify the level of assortativity with the following assortativity coefficient [6]:

$$r = \frac{\sum_\mu e_{\mu\mu} - \sum_\mu a_\mu b_\mu}{1 - \sum_\mu a_\mu b_\mu} = \frac{\text{Tr}\,\mathbf{E} - ||E^2||_1}{1 - ||E^2||_1}$$

where $||\cdot||_1$ is the 1-norm = sum of a matrix's entries.

- Tr **E** is the fraction of edges that are within groups.
- $||E^2||_1$ is the fraction of edges that would be within groups if connections were random.
- $1 - ||E^2||_1$ is a normalization factor so $r_{\text{max}} = 1$.
- When Tr $e_{\mu\mu} = 1$, we have $r = 1$. ✔
- When $e_{\mu\mu} = a_\mu b_\mu$, we have $r = 0$. ✔

Correlation coefficient:

Notes:

- $r = -1$ is inaccessible if three or more types are present.

Correlation coefficient:

Notes:

- $r = -1$ is inaccessible if three or more types are present.
- Disassortative networks simply have nodes connected to unlike nodes—no measure of how unlike nodes are.

Correlation coefficient:

Notes:

- $r = -1$ is inaccessible if three or more types are present.
- Disassortative networks simply have nodes connected to unlike nodes—no measure of how unlike nodes are.
- Minimum value of r occurs when all links between non-like nodes: Tr $e_{\mu\mu} = 0$.

Correlation coefficient:

Notes:

- $r = -1$ is inaccessible if three or more types are present.
- Disassortative networks simply have nodes connected to unlike nodes—no measure of how unlike nodes are.
- Minimum value of r occurs when all links between non-like nodes: Tr $e_{\mu\mu} = 0$.
-
$$r_{\text{min}} = \frac{-||E^2||_1}{1 - ||E^2||_1}$$

where $-1 \leq r_{\text{min}} < 0$.

Watch your step

zzzhhhhwoooommmmmm

NuhnuhNuhnuhNuhnuhNuhnuhNuhnuhNuhnuh

...

Scalar quantities

- Now consider nodes defined by a scalar integer quantity.

Scalar quantities

- Now consider nodes defined by a scalar integer quantity.
- Examples: age in years, height in inches, number of friends, ...

Scalar quantities

- Now consider nodes defined by a scalar integer quantity.
- Examples: age in years, height in inches, number of friends, ...
- e_{jk} = **Pr** (a randomly chosen edge connects a node with value j to a node with value k).

Scalar quantities

- Now consider nodes defined by a scalar integer quantity.
- Examples: age in years, height in inches, number of friends, ...
- e_{jk} = **Pr** (a randomly chosen edge connects a node with value j to a node with value k).
- a_j and b_k are defined as before.

Scalar quantities

- Now consider nodes defined by a scalar integer quantity.
- Examples: age in years, height in inches, number of friends, ...
- e_{jk} = **Pr** (a randomly chosen edge connects a node with value j to a node with value k).
- a_j and b_k are defined as before.
- Can now measure correlations between nodes based on this scalar quantity using standard Pearson correlation coefficient:

Scalar quantities

- Now consider nodes defined by a scalar integer quantity.
- Examples: age in years, height in inches, number of friends, ...
- e_{jk} = **Pr** (a randomly chosen edge connects a node with value j to a node with value k).
- a_j and b_k are defined as before.
- Can now measure correlations between nodes based on this scalar quantity using standard Pearson correlation coefficient:

$$r = \frac{\sum_{j\,k} j\,k(e_{jk} - a_j b_k)}{\sigma_a\,\sigma_b} = \frac{\langle jk\rangle - \langle j\rangle_a \langle k\rangle_b}{\sqrt{\langle j^2\rangle_a - \langle j\rangle_a^2}\sqrt{\langle k^2\rangle_b - \langle k\rangle_b^2}}$$

Scalar quantities

- Now consider nodes defined by a scalar integer quantity.
- Examples: age in years, height in inches, number of friends, ...
- e_{jk} = **Pr** (a randomly chosen edge connects a node with value j to a node with value k).
- a_j and b_k are defined as before.
- Can now measure correlations between nodes based on this scalar quantity using standard Pearson correlation coefficient:

$$r = \frac{\sum_{j\,k} j\,k(e_{jk} - a_j b_k)}{\sigma_a\,\sigma_b} = \frac{\langle jk\rangle - \langle j\rangle_a \langle k\rangle_b}{\sqrt{\langle j^2\rangle_a - \langle j\rangle_a^2}\sqrt{\langle k^2\rangle_b - \langle k\rangle_b^2}}$$

- This is the observed normalized deviation from randomness in the product jk.

Degree-degree correlations

Degree-degree correlations

- Natural correlation is between the degrees of connected nodes.

Degree-degree correlations

- Natural correlation is between the degrees of connected nodes.
- Now define e_{jk} with a slight twist:

$$e_{jk} = \textbf{Pr}\left(\begin{array}{l}\text{an edge connects a degree } j+1 \text{ node} \\ \text{to a degree } k+1 \text{ node}\end{array}\right)$$

Degree-degree correlations

- Natural correlation is between the degrees of connected nodes.
- Now define e_{jk} with a slight twist:

$$e_{jk} = \mathbf{Pr}\left(\begin{array}{l} \text{an edge connects a degree } j+1 \text{ node} \\ \text{to a degree } k+1 \text{ node} \end{array} \right)$$

$$= \mathbf{Pr}\left(\begin{array}{l} \text{an edge runs between a node of in-degree } j \\ \text{and a node of out-degree } k \end{array} \right)$$

Degree-degree correlations

- Natural correlation is between the degrees of connected nodes.
- Now define e_{jk} with a slight twist:

$$e_{jk} = \textbf{Pr}\left(\begin{array}{l}\text{an edge connects a degree } j+1 \text{ node} \\ \text{to a degree } k+1 \text{ node}\end{array}\right)$$

$$= \textbf{Pr}\left(\begin{array}{l}\text{an edge runs between a node of in-degree } j \\ \text{and a node of out-degree } k\end{array}\right)$$

- Useful for calculations (as per R_k)

Degree-degree correlations

- Natural correlation is between the degrees of connected nodes.
- Now define e_{jk} with a slight twist:

$$e_{jk} = \textbf{Pr}\left(\begin{array}{l}\text{an edge connects a degree } j+1 \text{ node} \\ \text{to a degree } k+1 \text{ node}\end{array}\right)$$

$$= \textbf{Pr}\left(\begin{array}{l}\text{an edge runs between a node of in-degree } j \\ \text{and a node of out-degree } k\end{array}\right)$$

- Useful for calculations (as per R_k)
- Important: Must separately define P_0 as the $\{e_{jk}\}$ contain no information about isolated nodes.

Degree-degree correlations

- Natural correlation is between the degrees of connected nodes.
- Now define e_{jk} with a slight twist:

$$e_{jk} = \textbf{Pr}\left(\begin{array}{l}\text{an edge connects a degree } j+1 \text{ node} \\ \text{to a degree } k+1 \text{ node}\end{array}\right)$$

$$= \textbf{Pr}\left(\begin{array}{l}\text{an edge runs between a node of in-degree } j \\ \text{and a node of out-degree } k\end{array}\right)$$

- Useful for calculations (as per R_k)
- Important: Must separately define P_0 as the $\{e_{jk}\}$ contain no information about isolated nodes.
- Directed networks still fine but we will assume from here on that $e_{jk} = e_{kj}$.

Degree-degree correlations

- Notation reconciliation for undirected networks:

$$r = \frac{\sum_{j\,k} j\,k(e_{jk} - R_j R_k)}{\sigma_R^2}$$

where, as before, R_k is the probability that a randomly chosen edge leads to a node of degree $k+1$, and

$$\sigma_R^2 = \sum_j j^2 R_j - \left[\sum_j j R_j\right]^2.$$

Degree-degree correlations

Error estimate for r:

- Remove edge i and recompute r to obtain r_i.

Degree-degree correlations

Error estimate for r:

- Remove edge i and recompute r to obtain r_i.
- Repeat for all edges and compute using the jackknife method [3]

$$\sigma_r^2 = \sum_i (r_i - r)^2.$$

Degree-degree correlations

Error estimate for r:

- Remove edge i and recompute r to obtain r_i.
- Repeat for all edges and compute using the jackknife method[3]

$$\sigma_r^2 = \sum_i (r_i - r)^2.$$

- Mildly sneaky as variables need to be independent for us to be truly happy and edges are correlated...

Measurements of degree-degree correlations

	Group	Network	Type	Size n	Assortativity r	Error σ_r
	a	Physics coauthorship	undirected	52 909	0.363	0.002
	a	Biology coauthorship	undirected	1 520 251	0.127	0.0004
	b	Mathematics coauthorship	undirected	253 339	0.120	0.002
Social	c	Film actor collaborations	undirected	449 913	0.208	0.0002
	d	Company directors	undirected	7 673	0.276	0.004
	e	Student relationships	undirected	573	−0.029	0.037
	f	Email address books	directed	16 881	0.092	0.004
	g	Power grid	undirected	4 941	−0.003	0.013
Technological	h	Internet	undirected	10 697	−0.189	0.002
	i	World Wide Web	directed	269 504	−0.067	0.0002
	j	Software dependencies	directed	3 162	−0.016	0.020
	k	Protein interactions	undirected	2 115	−0.156	0.010
	l	Metabolic network	undirected	765	−0.240	0.007
Biological	m	Neural network	directed	307	−0.226	0.016
	n	Marine food web	directed	134	−0.263	0.037
	o	Freshwater food web	directed	92	−0.326	0.031

- Social networks tend to be assortative (homophily)
- Technological and biological networks tend to be disassortative

Hot lava

"I like it"

Outline

Spreading on degree-correlated networks

- Next: Generalize our work for random networks to degree-correlated networks.

Spreading on degree-correlated networks

- Next: Generalize our work for random networks to degree-correlated networks.
- As before, by allowing that a node of degree k is activated by one neighbor with probability B_{k1}, we can handle various problems:

Spreading on degree-correlated networks

- Next: Generalize our work for random networks to degree-correlated networks.
- As before, by allowing that a node of degree k is activated by one neighbor with probability B_{k1}, we can handle various problems:
  1. find the giant component size.

Spreading on degree-correlated networks

- Next: Generalize our work for random networks to degree-correlated networks.
- As before, by allowing that a node of degree k is activated by one neighbor with probability B_{k1}, we can handle various problems:
  1. find the giant component size.
  2. find the probability and extent of spread for simple disease models.

Spreading on degree-correlated networks

- Next: Generalize our work for random networks to degree-correlated networks.
- As before, by allowing that a node of degree k is activated by one neighbor with probability B_{k1}, we can handle various problems:
  1. find the giant component size.
  2. find the probability and extent of spread for simple disease models.
  3. find the probability of spreading for simple threshold models.

Spreading on degree-correlated networks

- Goal: Find $f_{n,j}$ = **Pr** an edge emanating from a degree $j + 1$ node leads to a finite active subcomponent of size n.

Spreading on degree-correlated networks

- Goal: Find $f_{n,j}$ = **Pr** an edge emanating from a degree $j+1$ node leads to a finite active subcomponent of size n.
- Repeat: a node of degree k is in the game with probability B_{k1}.

Spreading on degree-correlated networks

- Goal: Find $f_{n,j}$ = **Pr** an edge emanating from a degree $j+1$ node leads to a finite active subcomponent of size n.
- Repeat: a node of degree k is in the game with probability B_{k1}.
- Define $\vec{B}_1 = [B_{k1}]$.

Spreading on degree-correlated networks

- Goal: Find $f_{n,j}$ = **Pr** an edge emanating from a degree $j+1$ node leads to a finite active subcomponent of size n.
- Repeat: a node of degree k is in the game with probability B_{k1}.
- Define $\vec{B}_1 = [B_{k1}]$.
- Plan: Find the generating function $F_j(x;\vec{B}_1) = \sum_{n=0}^{\infty} f_{n,j} x^n$.

Spreading on degree-correlated networks

- Recursive relationship:

$$F_j(x;\vec{B}_1) = x^0 \sum_{k=0}^{\infty} \frac{e_{jk}}{R_j}(1 - B_{k+1,1}) + x \sum_{k=0}^{\infty} \frac{e_{jk}}{R_j} B_{k+1,1} \left[F_k(x;\vec{B}_1)\right]^k .$$

Spreading on degree-correlated networks

- Recursive relationship:

$$F_j(x;\vec{B}_1) = x^0 \sum_{k=0}^{\infty} \frac{e_{jk}}{R_j}(1 - B_{k+1,1})$$
$$+ x \sum_{k=0}^{\infty} \frac{e_{jk}}{R_j} B_{k+1,1} \left[F_k(x;\vec{B}_1)\right]^k .$$

- First term = **Pr** (that the first node we reach is not in the game).

Spreading on degree-correlated networks

- Recursive relationship:

$$F_j(x;\vec{B}_1) = x^0 \sum_{k=0}^{\infty} \frac{e_{jk}}{R_j}(1 - B_{k+1,1}) + x \sum_{k=0}^{\infty} \frac{e_{jk}}{R_j} B_{k+1,1} \left[F_k(x;\vec{B}_1)\right]^k.$$

- First term = **Pr** (that the first node we reach is not in the game).
- Second term involves **Pr** (we hit an active node which has k outgoing edges).

Spreading on degree-correlated networks

- Recursive relationship:

$$F_j(x;\vec{B}_1) = x^0 \sum_{k=0}^{\infty} \frac{e_{jk}}{R_j}(1 - B_{k+1,1}) + x \sum_{k=0}^{\infty} \frac{e_{jk}}{R_j} B_{k+1,1} \left[F_k(x;\vec{B}_1)\right]^k .$$

- First term = **Pr** (that the first node we reach is not in the game).
- Second term involves **Pr** (we hit an active node which has k outgoing edges).
- Next: find average size of active components reached by following a link from a degree $j+1$ node = $F_j'(1;\vec{B}_1)$.

Spreading on degree-correlated networks

- Differentiate $F_j(x;\vec{B}_1)$, set $x = 1$, and rearrange.

Spreading on degree-correlated networks

- Differentiate $F_j(x;\vec{B}_1)$, set $x = 1$, and rearrange.
- We use $F_k(1;\vec{B}_1) = 1$ which is true when no giant component exists.

Spreading on degree-correlated networks

- Differentiate $F_j(x;\vec{B}_1)$, set $x = 1$, and rearrange.
- We use $F_k(1;\vec{B}_1) = 1$ which is true when no giant component exists. We find:

$$R_j F_j'(1;\vec{B}_1) = \sum_{k=0}^{\infty} e_{jk} B_{k+1,1} + \sum_{k=0}^{\infty} k e_{jk} B_{k+1,1} F_k'(1;\vec{B}_1).$$

Spreading on degree-correlated networks

- Differentiate $F_j(x;\vec{B}_1)$, set $x = 1$, and rearrange.
- We use $F_k(1;\vec{B}_1) = 1$ which is true when no giant component exists. We find:

$$R_j F_j'(1;\vec{B}_1) = \sum_{k=0}^{\infty} e_{jk} B_{k+1,1} + \sum_{k=0}^{\infty} k e_{jk} B_{k+1,1} F_k'(1;\vec{B}_1).$$

- Rearranging and introducing a sneaky δ_{jk}:

$$\sum_{k=0}^{\infty} \left(\delta_{jk} R_k - k B_{k+1,1} e_{jk}\right) F_k'(1;\vec{B}_1) = \sum_{k=0}^{\infty} e_{jk} B_{k+1,1}.$$

Spreading on degree-correlated networks

- In matrix form, we have

$$\mathbf{A}_{\mathbf{E},\vec{B}_1}\vec{F}'(1;\vec{B}_1) = \mathbf{E}\vec{B}_1$$

where

$$\left[\mathbf{A}_{\mathbf{E},\vec{B}_1}\right]_{j+1,k+1} = \delta_{jk}R_k - kB_{k+1,1}e_{jk},$$

$$\left[\vec{F}'(1;\vec{B}_1)\right]_{k+1} = F'_k(1;\vec{B}_1),$$

$$[\mathbf{E}]_{j+1,k+1} = e_{jk}, \text{ and } \left[\vec{B}_1\right]_{k+1} = B_{k+1,1}.$$

Spreading on degree-correlated networks

- So, in principle at least:

$$\vec{F}'(1;\vec{B}_1) = \mathbf{A}_{\mathbf{E},\vec{B}_1}^{-1}\,\mathbf{E}\vec{B}_1 .$$

Spreading on degree-correlated networks

- So, in principle at least:

$$\vec{F}'(1;\vec{B}_1) = \mathbf{A}^{-1}_{\mathbf{E},\vec{B}_1}\,\mathbf{E}\vec{B}_1.$$

- Now: as $\vec{F}'(1;\vec{B}_1)$, the average size of an active component reached along an edge, increases, we move towards a transition to a giant component.

Spreading on degree-correlated networks

- So, in principle at least:

$$\vec{F}'(1;\vec{B}_1) = \mathbf{A}^{-1}_{\mathbf{E},\vec{B}_1}\,\mathbf{E}\vec{B}_1 .$$

- Now: as $\vec{F}'(1;\vec{B}_1)$, the average size of an active component reached along an edge, increases, we move towards a transition to a giant component.
- Right at the transition, the average component size explodes.

Spreading on degree-correlated networks

- So, in principle at least:

$$\vec{F}'(1;\vec{B}_1) = \mathbf{A}^{-1}_{\mathbf{E},\vec{B}_1}\,\mathbf{E}\vec{B}_1.$$

- Now: as $\vec{F}'(1;\vec{B}_1)$, the average size of an active component reached along an edge, increases, we move towards a transition to a giant component.
- Right at the transition, the average component size explodes.
- Exploding inverses of matrices occur when their determinants are 0.

Spreading on degree-correlated networks

- So, in principle at least:

$$\vec{F}'(1;\vec{B}_1) = \mathbf{A}_{\mathbf{E},\vec{B}_1}^{-1}\,\mathbf{E}\vec{B}_1.$$

- Now: as $\vec{F}'(1;\vec{B}_1)$, the average size of an active component reached along an edge, increases, we move towards a transition to a giant component.
- Right at the transition, the average component size explodes.
- Exploding inverses of matrices occur when their determinants are 0.
- The condition is therefore:

$$\det \mathbf{A}_{\mathbf{E},\vec{B}_1} = 0$$

.

Spreading on degree-correlated networks

- General condition details:

$$\det \mathbf{A}_{\mathbf{E},\vec{B}_1} = \det \left[\delta_{jk} R_{k-1} - (k-1) B_{k,1} e_{j-1,k-1} \right] = 0.$$

Spreading on degree-correlated networks

- General condition details:

$$\det \mathbf{A}_{\mathbf{E},\vec{B}_1} = \det\left[\delta_{jk}R_{k-1} - (k-1)B_{k,1}e_{j-1,k-1}\right] = 0.$$

- The above collapses to our standard contagion condition when $e_{jk} = R_j R_k$ (see next slide). [2]

Spreading on degree-correlated networks

- General condition details:

$$\det \mathbf{A}_{\mathbf{E},\vec{B}_1} = \det\left[\delta_{jk} R_{k-1} - (k-1) B_{k,1} e_{j-1,k-1}\right] = 0.$$

- The above collapses to our standard contagion condition when $e_{jk} = R_j R_k$ (see next slide). [2]
- When $\vec{B}_1 = B\vec{1}$, we have the condition for a simple disease model's successful spread

$$\det\left[\delta_{jk} R_{k-1} - B(k-1) e_{j-1,k-1}\right] = 0.$$

Spreading on degree-correlated networks

- General condition details:

$$\det \mathbf{A}_{\mathbf{E},\vec{B}_1} = \det\left[\delta_{jk}R_{k-1} - (k-1)B_{k,1}e_{j-1,k-1}\right] = 0.$$

- The above collapses to our standard contagion condition when $e_{jk} = R_j R_k$ (see next slide). [2]
- When $\vec{B}_1 = B\vec{1}$, we have the condition for a simple disease model's successful spread

$$\det\left[\delta_{jk}R_{k-1} - B(k-1)e_{j-1,k-1}\right] = 0.$$

- When $\vec{B}_1 = \vec{1}$, we have the condition for the existence of a giant component:

$$\det\left[\delta_{jk}R_{k-1} - (k-1)e_{j-1,k-1}\right] = 0.$$

Spreading on degree-correlated networks

- General condition details:

$$\det \mathbf{A}_{\mathbf{E},\vec{B}_1} = \det\left[\delta_{jk}R_{k-1} - (k-1)B_{k,1}e_{j-1,k-1}\right] = 0.$$

- The above collapses to our standard contagion condition when $e_{jk} = R_j R_k$ (see next slide). [2]
- When $\vec{B}_1 = B\vec{1}$, we have the condition for a simple disease model's successful spread

$$\det\left[\delta_{jk}R_{k-1} - B(k-1)e_{j-1,k-1}\right] = 0.$$

- When $\vec{B}_1 = \vec{1}$, we have the condition for the existence of a giant component:

$$\det\left[\delta_{jk}R_{k-1} - (k-1)e_{j-1,k-1}\right] = 0.$$

- Bonusville: We'll find a much better version of this set of conditions later...

Retrieving the cascade condition for uncorrelated networks

Outline

Spreading on degree-correlated networks

We'll next find two more pieces:

1. P_{trig}, the probability of starting a cascade

Spreading on degree-correlated networks

We'll next find two more pieces:

1. P_{trig}, the probability of starting a cascade
2. S, the expected extent of activation given a small seed.

Spreading on degree-correlated networks

We'll next find two more pieces:

1. P_{trig}, the probability of starting a cascade
2. S, the expected extent of activation given a small seed.

Triggering probability:

- Generating function:

$$H(x; \vec{B}_1) = x \sum_{k=0}^{\infty} P_k \left[F_{k-1}(x; \vec{B}_1) \right]^k .$$

Spreading on degree-correlated networks

We'll next find two more pieces:

1. P_{trig}, the probability of starting a cascade
2. S, the expected extent of activation given a small seed.

Triggering probability:

- Generating function:

$$H(x;\vec{B}_1) = x \sum_{k=0}^{\infty} P_k \left[F_{k-1}(x;\vec{B}_1)\right]^k .$$

- Generating function for vulnerable component size is more complicated.

Spreading on degree-correlated networks

- Want probability of not reaching a finite component.

$$P_{\text{trig}} = S_{\text{trig}} = 1 - H(1; \vec{B}_1)$$
$$= 1 - \sum_{k=0}^{\infty} P_k \left[F_{k-1}(1; \vec{B}_1)\right]^k .$$

Spreading on degree-correlated networks

- Want probability of not reaching a finite component.

$$P_{\text{trig}} = S_{\text{trig}} = 1 - H(1; \vec{B}_1)$$
$$= 1 - \sum_{k=0}^{\infty} P_k \left[F_{k-1}(1; \vec{B}_1) \right]^k .$$

- Last piece: we have to compute $F_{k-1}(1; \vec{B}_1)$.

Spreading on degree-correlated networks

- Want probability of not reaching a finite component.

$$P_{\text{trig}} = S_{\text{trig}} = 1 - H(1; \vec{B}_1)$$
$$= 1 - \sum_{k=0}^{\infty} P_k \left[F_{k-1}(1; \vec{B}_1)\right]^k .$$

- Last piece: we have to compute $F_{k-1}(1; \vec{B}_1)$.
- Nastier (nonlinear)—we have to solve the recursive expression we started with when $x = 1$:
$F_j(1; \vec{B}_1) = \sum_{k=0}^{\infty} \frac{e_{jk}}{R_j}(1 - B_{k+1,1}) +$
$\sum_{k=0}^{\infty} \frac{e_{jk}}{R_j} B_{k+1,1} \left[F_k(1; \vec{B}_1)\right]^k .$

Spreading on degree-correlated networks

- Want probability of not reaching a finite component.

$$P_{\text{trig}} = S_{\text{trig}} = 1 - H(1; \vec{B}_1)$$
$$= 1 - \sum_{k=0}^{\infty} P_k \left[F_{k-1}(1; \vec{B}_1) \right]^k .$$

- Last piece: we have to compute $F_{k-1}(1; \vec{B}_1)$.
- Nastier (nonlinear)—we have to solve the recursive expression we started with when $x = 1$:
  $F_j(1; \vec{B}_1) = \sum_{k=0}^{\infty} \frac{e_{jk}}{R_j}(1 - B_{k+1,1}) +$
  $\sum_{k=0}^{\infty} \frac{e_{jk}}{R_j} B_{k+1,1} \left[F_k(1; \vec{B}_1) \right]^k .$
- Iterative methods should work here.

Outline

Spreading on degree-correlated networks

- Truly final piece: Find final size using approach of Gleeson [4], a generalization of that used for uncorrelated random networks.

Spreading on degree-correlated networks

- Truly final piece: Find final size using approach of Gleeson [4], a generalization of that used for uncorrelated random networks.
- Need to compute $\theta_{j,t}$, the probability that an edge leading to a degree j node is infected at time t.

Spreading on degree-correlated networks

- Truly final piece: Find final size using approach of Gleeson [4], a generalization of that used for uncorrelated random networks.
- Need to compute $\theta_{j,t}$, the probability that an edge leading to a degree j node is infected at time t.
- Evolution of edge activity probability:

$$\theta_{j,t+1} = G_j(\vec{\theta}_t) = \phi_0 + (1-\phi_0)\times$$

$$\sum_{k=1}^{\infty} \frac{e_{j-1,k-1}}{R_{j-1}} \sum_{i=0}^{k-1} \binom{k-1}{i} \theta_{k,t}^{i} (1-\theta_{k,t})^{k-1-i} B_{ki}.$$

Spreading on degree-correlated networks

- Truly final piece: Find final size using approach of Gleeson [4], a generalization of that used for uncorrelated random networks.
- Need to compute $\theta_{j,t}$, the probability that an edge leading to a degree j node is infected at time t.
- Evolution of edge activity probability:

$$\theta_{j,t+1} = G_j(\vec{\theta}_t) = \phi_0 + (1-\phi_0)\times$$

$$\sum_{k=1}^{\infty} \frac{e_{j-1,k-1}}{R_{j-1}} \sum_{i=0}^{k-1} \binom{k-1}{i} \theta_{k,t}^{i} (1-\theta_{k,t})^{k-1-i} B_{ki}.$$

- Overall active fraction's evolution:

$$\phi_{t+1} = \phi_0 + (1-\phi_0) \sum_{k=0}^{\infty} P_k \sum_{i=0}^{k} \binom{k}{i} \theta_{k,t}^{i} (1-\theta_{k,t})^{k-i} B_{ki}.$$

Spreading on degree-correlated networks

- As before, these equations give the actual evolution of ϕ_t for synchronous updates.

Spreading on degree-correlated networks

- As before, these equations give the actual evolution of ϕ_t for synchronous updates.
- Contagion condition follows from $\vec{\theta}_{t+1} = \vec{G}(\vec{\theta}_t)$.

Spreading on degree-correlated networks

- As before, these equations give the actual evolution of ϕ_t for synchronous updates.
- Contagion condition follows from $\vec{\theta}_{t+1} = \vec{G}(\vec{\theta}_t)$.
- Expand $\vec{G}$ around $\vec{\theta}_0 = \vec{0}$.

Spreading on degree-correlated networks

- As before, these equations give the actual evolution of ϕ_t for synchronous updates.
- Contagion condition follows from $\vec{\theta}_{t+1} = \vec{G}(\vec{\theta}_t)$.
- Expand $\vec{G}$ around $\vec{\theta}_0 = \vec{0}$.

$$\theta_{j,t+1} = G_j(\vec{0}) + \sum_{k=1}^{\infty} \frac{\partial G_j(\vec{0})}{\partial \theta_{k,t}} \theta_{k,t} + \frac{1}{2!} \sum_{k=1}^{\infty} \frac{\partial^2 G_j(\vec{0})}{\partial \theta_{k,t}^2} \theta_{k,t}^2 + \ldots$$

Spreading on degree-correlated networks

- As before, these equations give the actual evolution of ϕ_t for synchronous updates.
- Contagion condition follows from $\vec{\theta}_{t+1} = \vec{G}(\vec{\theta}_t)$.
- Expand $\vec{G}$ around $\vec{\theta}_0 = \vec{0}$.

$$\theta_{j,t+1} = G_j(\vec{0}) + \sum_{k=1}^{\infty} \frac{\partial G_j(\vec{0})}{\partial \theta_{k,t}} \theta_{k,t} + \frac{1}{2!} \sum_{k=1}^{\infty} \frac{\partial^2 G_j(\vec{0})}{\partial \theta_{k,t}^2} \theta_{k,t}^2 + \ldots$$

- If $G_j(\vec{0}) \neq 0$ for at least one j, always have some infection.
- If $G_j(\vec{0}) = 0 \,\forall\, j$, want largest eigenvalue $\left[\frac{\partial G_j(\vec{0})}{\partial \theta_{k,t}}\right] > 1$.

Spreading on degree-correlated networks

- As before, these equations give the actual evolution of ϕ_t for synchronous updates.
- Contagion condition follows from $\vec{\theta}_{t+1} = \vec{G}(\vec{\theta}_t)$.
- Expand $\vec{G}$ around $\vec{\theta}_0 = \vec{0}$.

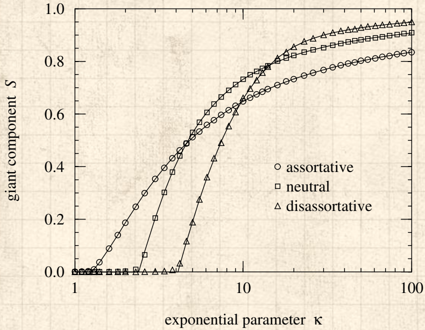
$$\theta_{j,t+1} = G_j(\vec{0}) + \sum_{k=1}^{\infty} \frac{\partial G_j(\vec{0})}{\partial \theta_{k,t}} \theta_{k,t} + \frac{1}{2!} \sum_{k=1}^{\infty} \frac{\partial^2 G_j(\vec{0})}{\partial \theta_{k,t}^2} \theta_{k,t}^2 + \ldots$$

- If $G_j(\vec{0}) \neq 0$ for at least one j, always have some infection.
- If $G_j(\vec{0}) = 0 \, \forall \, j$, want largest eigenvalue $\left[\frac{\partial G_j(\vec{0})}{\partial \theta_{k,t}}\right] > 1$.
- Condition for spreading is therefore dependent on eigenvalues of this matrix:

$$\frac{\partial G_j(\vec{0})}{\partial \theta_{k,t}} = \frac{e_{j-1,k-1}}{R_{j-1}} (k-1) B_{k1}$$

Insert question from assignment 9

How the giant component changes with assortativity:



from Newman, 2002 [5]

- More assortative networks percolate for lower average degrees
- But disassortative networks end up with higher extents of spreading.

Toy guns don't pretend blow up things ...

Splsshht

Robust-yet-Fragileness of the Death Star

References I

[1] M. Boguñá and M. Ángeles Serrano.
Generalized percolation in random directed networks.
Phys. Rev. E, 72:016106, 2005. pdf

[2] P. S. Dodds and J. L. Payne.
Analysis of a threshold model of social contagion on degree-correlated networks.
Phys. Rev. E, 79:066115, 2009. pdf

[3] B. Efron and C. Stein.
The jackknife estimate of variance.
The Annals of Statistics, 9:586–596, 1981. pdf

[4] J. P. Gleeson.
Cascades on correlated and modular random networks.
Phys. Rev. E, 77:046117, 2008. pdf

References II

[5] M. Newman.
Assortative mixing in networks.
Phys. Rev. Lett., 89:208701, 2002. pdf

[6] M. E. J. Newman.
Mixing patterns in networks.
Phys. Rev. E, 67:026126, 2003. pdf