Network Modeling

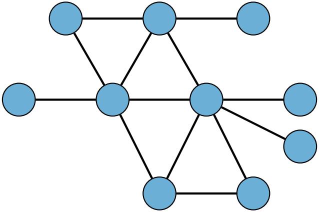
Viviana Amati Jürgen Lerner

Department of Computer &Information Science
University of Konstanz

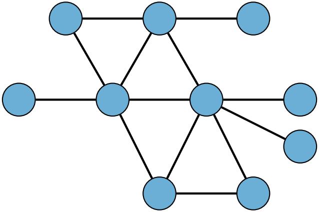
Winter 2014/2015
(version 04 February 2015)

Outline

So far...

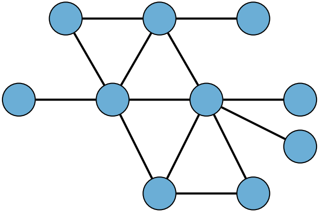


So far...



Model	Main feature	Real data
$\mathcal{G}(n, p)$	ties are independent	tie dependence
Planted partition	intra/inter group density	tie dependence
Preferential attachment	degree distribution	other structural properties
ERGM	class of models	reasonable representation

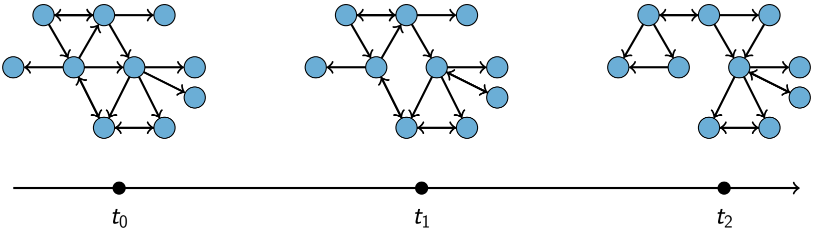
So far...



Model	Main feature	Real data
$\mathcal{G}(n, p)$	ties are independent	tie dependence
Planted partition	intra/inter group density	tie dependence
Preferential attachment	degree distribution	other structural properties
ERGM	class of models	reasonable representation

These are models for **cross-sectional** (directed/undirected) network data

Now...



Network are dynamic by nature:

The observed networks are the result of tie changes over time

How can we model the network evolution over time?

Longitudinal Network Data

(also referred to as network panel data)

- A social network consists of
  - a set of actors $\mathcal{N} = \{1, 2, \ldots, n\}$
  - a relation $\mathcal{R}$
- We can represent a network using
  - a graph: $G(V, E)$
  - an adjacency matrix x such that

$$x_{ij} = \begin{cases} 1 & i \rightarrow j \\ 0 & \textit{otherwise} \end{cases}$$

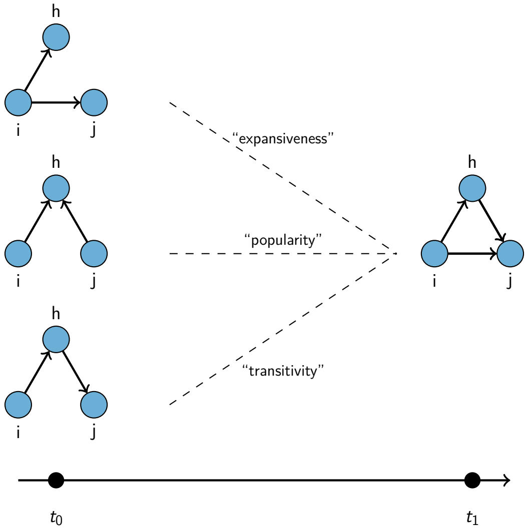
Longitudinal network data

- M+1 repeated observations of a network

$$x(t_0),\ x(t_1),\ \ldots,\ x(t_m),\ \ldots,\ x(t_{M-1}),\ x(t_M)$$

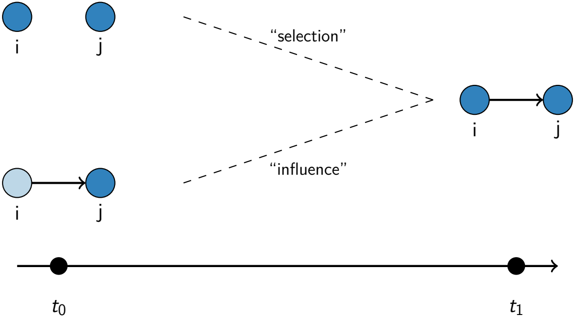
- actor covariates V (gender, age, social status, ...)

Why does time is important?



We can observe a transitive triplets because of several mechanisms

Why does time is important?



We can observe a homophilous dyad because of two processes

Why does time is important?

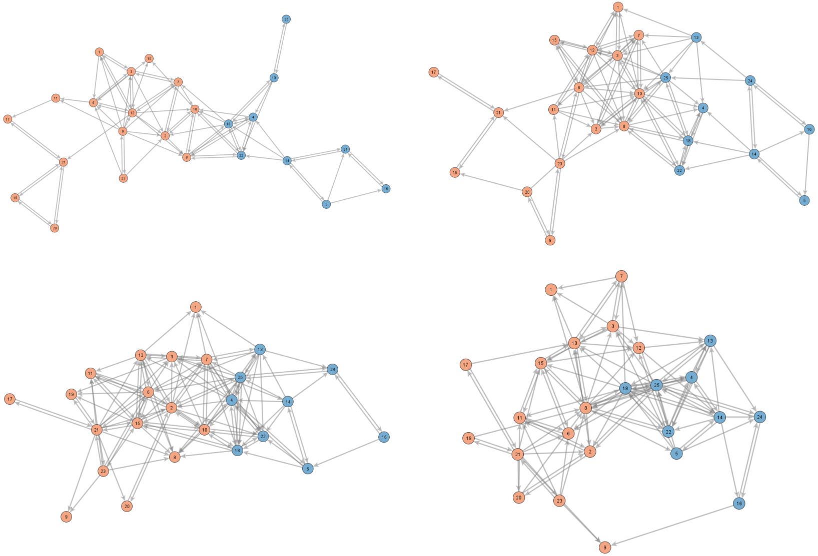
Networks can change over time:
ties can be created, deleted or maintained

Some questions:

1. How frequently do actors change ties?
2. What are the reasons that lead to a tie change?
3. How might appear the network in the future?

An example

A. Knecht (2008): “Friendship Selection and Friends’ Influence”



Four time points in the pupils’ first year at secondary school

Some questions

Is there any tendency in friendship formation ...

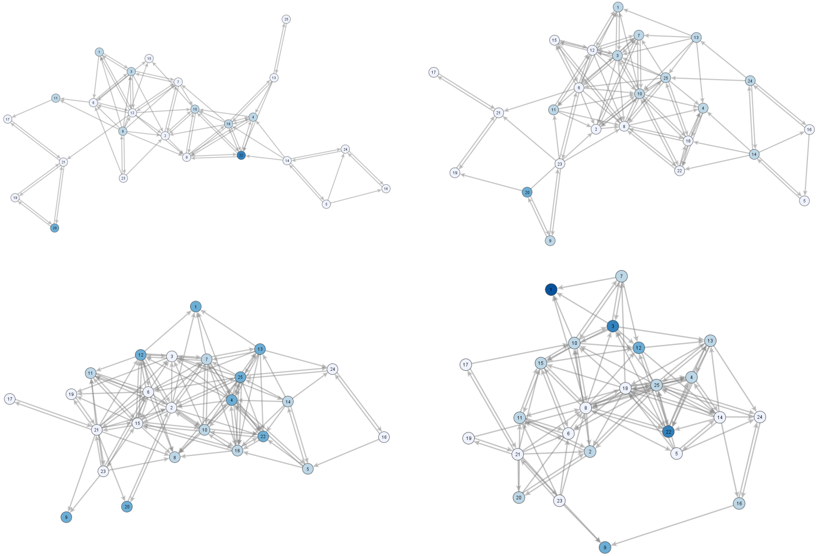
- towards reciprocity?

- towards transitivity?

- towards homphily w.r.t. gender?

An example

A. Knecht (2008): “Friendship Selection and Friends’ Influence”



Four time points in the pupils’ first year at secondary school (color delinquency)

Some questions

- Is there selection?

- Is there influence?

Networks models for longitudinal data

- Stochastic actor-oriented models (SAOMs)
- Temporal exponential random graph models (TERGMs)

Aim

Explain network evolution as a result of:

- endogenous variables:
  structural effects depending on the network only
  (e.g. reciprocity, transitivity, etc.)
- exogenous variables:
  actor-dependent and dyadic-dependent covariates
  (e.g. effect of a covariate on the existence of a tie or on homophily)

simultaneously

Outline

Background: probability space

Definition

A **probability space** is a pair (Ω, P) where

- Ω is a set of possible outcomes of a random experiment
- $P : \Omega \to [0,1]$ is a *probability function* such that:

  1. $P(\omega) \geq 0$
  2. $\sum_{\omega \in \Omega} P(\omega) = 1$

Notation

- $P(\omega)$ is called the probability of $\omega \in \Omega$
- The probability of a subset $\Omega' \subseteq \Omega$ is defined by $P(\Omega') = \sum_{\omega \in \Omega'} P(\omega)$

Background: random variable

Definition

A (real-valued) **random variable** (r.v.) is a function $X : \Omega \to \mathbb{R}$.

The set of values X can take is called **range** and will be denoted by $\mathcal{S}$

Example

Random experiment: throwing two dice

(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)					
	(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(2,6)				
		(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)			
			(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)		
				(5,1)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)	
					(6,1)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)

Ω

Background: random variable

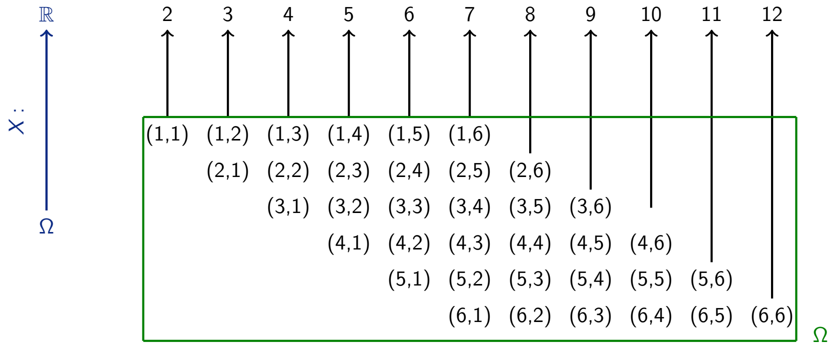
Definition

A (real-valued) **random variable** (r.v.) is a function $X : \Omega \to \mathbb{R}$.

The set of values X can take is called **range** and will be denoted by $\mathcal{S}$

Example

$X :=$ sum of two dice



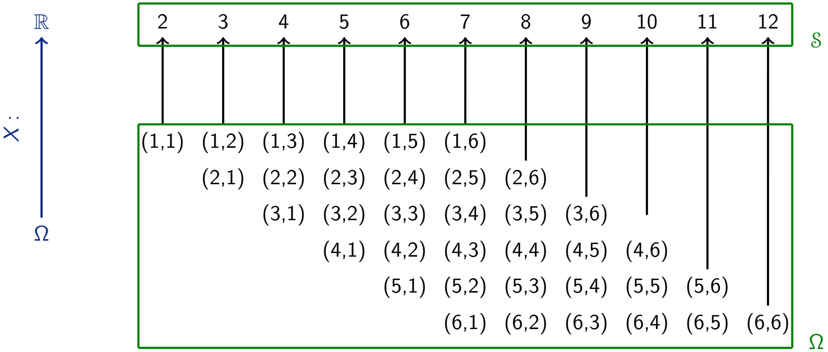
Background: random variable

Definition

A (real-valued) **random variable** (r.v.) is a function $X : \Omega \to \mathbb{R}$.
The set of values X can take is called **range** and will be denoted by $\mathcal{S}$

Example

$X :=$ sum of two dice



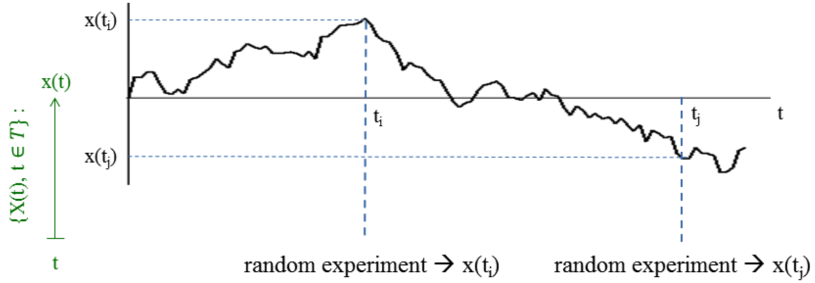
Outline

Background: stochastic (or random) process

Definition

A stochastic process $\{X(t), t \in \mathcal{T}\}$ is a mapping

$$\forall t \in \mathcal{T} \mapsto X(t) : \Omega \to \mathbb{R}$$

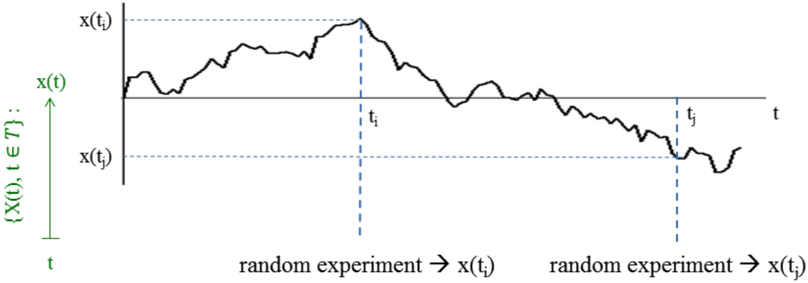


Background: stochastic (or random) process

Definition

A stochastic process $\{X(t), t \in \mathcal{T}\}$ is a mapping

$$\forall t \in \mathcal{T} \mapsto X(t) : \Omega \to \mathbb{R}$$



Notation

- $\mathcal{T}$ is an index set
- $\mathcal{S}$ is the state space of the process
  (i.e. set of values taken by the process)

Background: stochastic process

Different stochastic processes can be defined according to $\mathcal{S}$ and $\mathcal{T}$

$\mathcal{S}$	$\mathcal{T}$	
	Countable (discrete)	**Uncountable** (continuous)
Countable (finite)	discrete-time with finite state space	continuous-time with finite state space
Uncountable (continuous)	discrete-time with continuous state space	continuous-time with continuous state space

Background: stochastic process

Different stochastic processes can be defined according to $\mathcal{S}$ and $\mathcal{T}$

$\mathcal{S}$	$\mathcal{T}$	
	Countable (discrete)	**Uncountable** (continuous)
Countable (finite)	discrete-time with finite state space	**continuous-time with finite state space**
Uncountable (continuous)	discrete-time with continuous state space	continuous-time with continuous state space

Background: continuous-time Markov chain

Definition

A **continuous-time Markov chain** $\{X_t,\ t \geq 0\}$ is a stochastic process

1. with finite state space
2. evolving in continuous-time
3. having the Markovian property

Definition

$\{X(t),\ t \in \mathcal{T}\}$ has the **Markov property** if for all $x \in \mathcal{S}$ and for any pair $t_i < t_j$

$$P(X(t_j) = x(t_j) \mid X(t) = x(t), \forall\ t \leq t_i) = P(X(t_j) = x(t_j) \mid X(t_i) = x(t_i))$$

Intuitively: "the *future* depends on the *past* only through the *present*"

Background: continuous-time Markov chain

Example

$X(t) =$ number of goals that a given soccer player scores by time t
(time played in official matches)

$$\{X(t),\ t \geq 0\} \text{ is a continuous-time Markov chains}$$

Why?

Background: continuous-time Markov chain

Example

$X(t) =$ number of goals that a given soccer player scores by time t
(time played in official matches)

$$\{X(t),\ t \geq 0\} \text{ is a continuous-time Markov chains}$$

Why?

1. state space:

   $\mathcal{S} = \{0, 1, 2, \ldots, A\}$

   A = total number of goals scored during the career

Background: continuous-time Markov chain

Example

$X(t) =$ number of goals that a given soccer player scores by time t
(time played in official matches)

$$\{X(t),\ t \geq 0\} \text{ is a continuous-time Markov chains}$$

Why?

1. state space:

   $\mathcal{S} = \{0, 1, 2, \ldots, A\}$

   A = total number of goals scored during the career

2. the time is continuous:

   [0,B]

   B = time of retirement

Background: continuous-time Markov chain

Example

$X(t) =$ number of goals that a given soccer player scores by time t
(time played in official matches)

$$\{X(t),\ t \geq 0\} \text{ is a continuous-time Markov chains}$$

Why?

1. state space:

   $\mathcal{S} = \{0, 1, 2, \ldots, A\}$

   A = total number of goals scored during the career

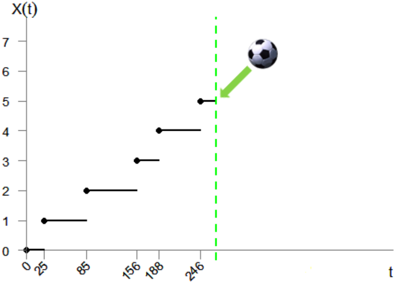
2. the time is continuous:

   [0,B]

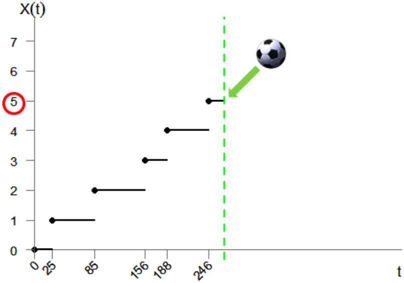
   B = time of retirement

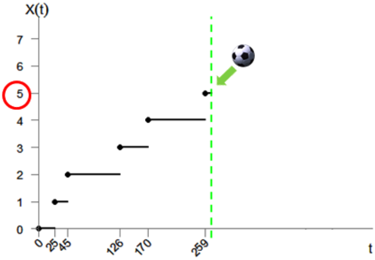
3. the process $\{X(t),\ t \geq 0\}$ has the Markov property

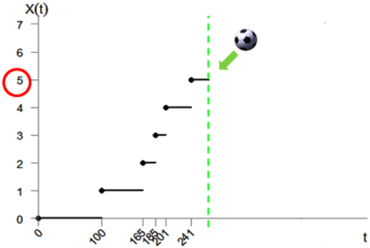
Background: Markov property



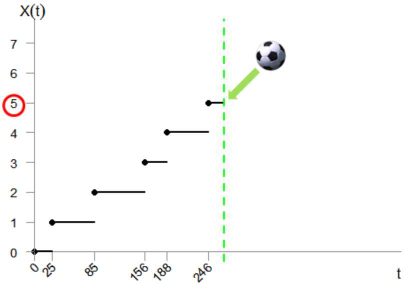
Background: Markov property

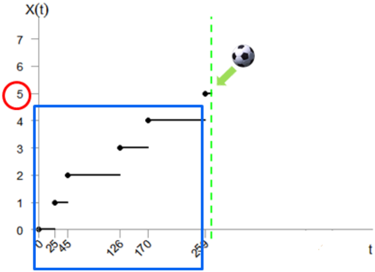


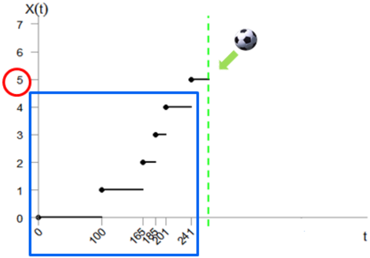




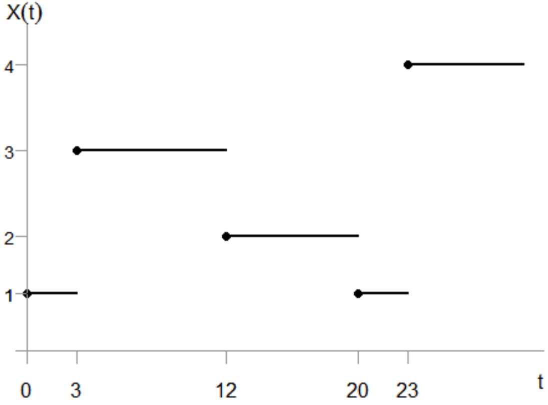
Background: Markov property







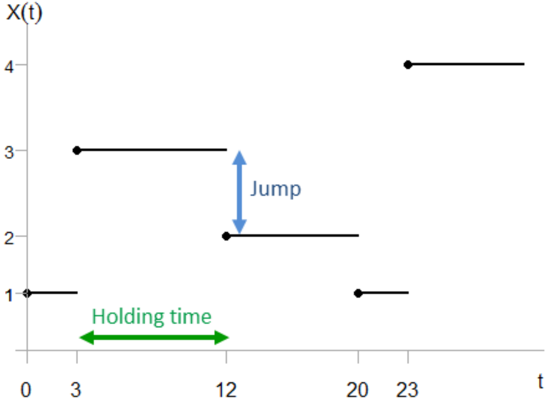
Background: describing a continuous-time Markov chain



We can decompose the process in a series of step defined by:

- the time there is a change
- the new state of the chain

Background: describing a continuous-time Markov chain



We can decompose the process in a series of step defined by:

- the time there is a change
- the new state of the chain

Background: describing a continuous-time Markov chain

Holding time

T_i = amount of time the chain spends in state i

It is assumed that T_i is exponentially distributed with p.d.f.

$$\varphi_T(t) = \lambda_i e^{-\lambda_i t}, \qquad \lambda_i > 0, \quad t > 0$$

where λ_i is called *rate parameter*

Why?

The Exponential r.v. has the *memoryless property*

$$P(T > s + t \mid T > t) = P(T > s) \quad \forall\ s, t > 0$$

Background: describing a continuous-time Markov chain

Jump chain

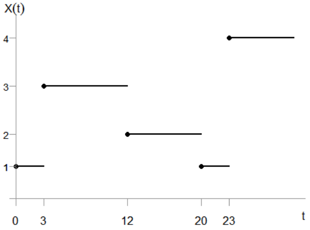
Let $s = |\mathcal{S}|$. The jump chain is described by a **jump matrix**

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1s} \\ p_{21} & p_{22} & \cdots & p_{2s} \\ \cdots & \cdots & \cdots & \cdots \\ p_{s1} & p_{s2} & \cdots & p_{ss} \end{bmatrix}$$

where

$$p_{ij} = P(X(t') = j | X(t) = i, \textit{ the opportunity to leave } i) \qquad p_{ij} \geq 0 \qquad \sum_{j \in \mathcal{S}} p_{ij} = 1$$

Example



$$P = \begin{bmatrix} 0.1 & 0 & 0.6 & 0.3 \\ 0.8 & 0.1 & 0.1 & 0 \\ 0.05 & 0.5 & 0.05 & 0.4 \\ 0.6 & 0.1 & 0.15 & 0.15 \end{bmatrix}$$

Outline

Stochastic Actor Oriented Models (SAOMs)

- Family of models
- Developed by T. Snijders in 1996
  - non-reflexive directed ties
  - ties have a tendency to endure over time (not event!!!)
  - several extensions during the past two decades
    Snijders, van de Bunt, and Steglich,
    Introduction to stochastic actor-based models for network dynamics. Social Networks 32(1):44-60, 2010.
- Aim: describe the evolution of a network over time
- Network evolution is the outcome of a
  continuous-time Markov chain
  ties are formed as a reaction to the existence of other ties

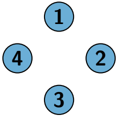
Model definition: continuous-time Markov chain

Finite state space

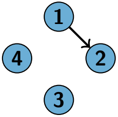
$\mathcal{X}$ is the set of all possible adjacency matrices defined on $\mathcal{N}$

$|\mathcal{X}| = 2^{n(n-1)}$

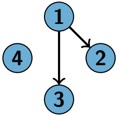
Example



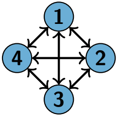
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0



0	1	0	0
0	0	0	0
0	0	0	0
0	0	0	0



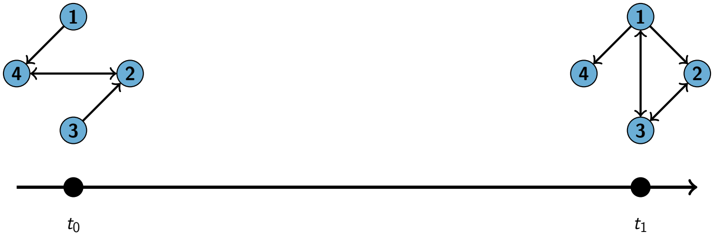
0	1	1	0
0	0	0	0
0	0	0	0
0	0	0	0



0	1	1	1
1	0	1	1
1	1	0	1
1	1	1	0

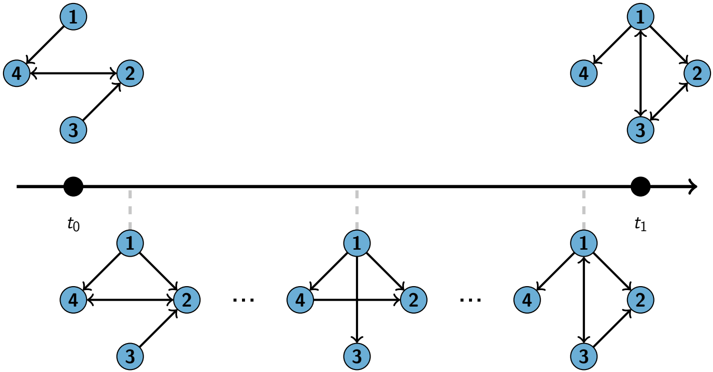
Model definition: continuous-time Markov chain

Continuous-time process



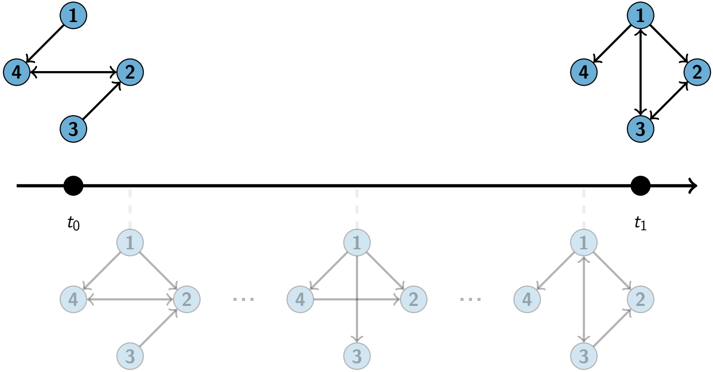
Model definition: continuous-time Markov chain

Continuous-time process



Model definition: continuous-time Markov chain

Continuous-time process



Latent process

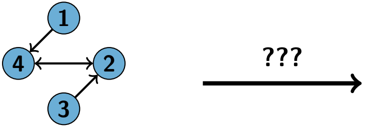
the network evolves in continuous-time but
we observed it only at discrete time points

Model definition: continuous-time Markov chain

Markov property

The current state of the network determines probabilistically its further evolution

- Given the current network (x) what is the next network (x')?



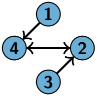
$|\mathcal{X}| = 2^{n(n-1)}$ possibilities

too many!!!

- The model is **actor-oriented**
  - *Opportunity to change*
    at any given moment t one actor has the opportunity to change
  - *Absence of co-occurrence*
    no more than one tie can change at any given moment t
  - *Actor's decision*
    change in ties are made by the actor who sends the ties

Model definition: continuous-time Markov chain

Opportunity to change

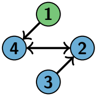


0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x=current state

Model definition: continuous-time Markov chain

Opportunity to change

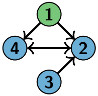


0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x=current state

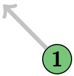
Model definition: continuous-time Markov chain

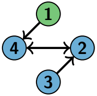
Absence of co-occurrence



0	1	0	1
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 2)$





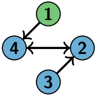
0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x=current state

Notation:
$x(i \rightsquigarrow j)$ denotes the network x where the tie from i to j is turned into its opposite
$x(i \rightsquigarrow i)$ means that i does not change any of his outgoing ties

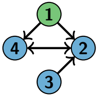
Model definition: continuous-time Markov chain

Absence of co-occurrence



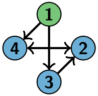
0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x=current state



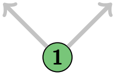
0	1	0	1
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 2)$



0	0	1	1
0	0	0	1
0	1	0	0
0	1	0	0

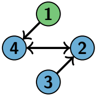
$x(1 \rightsquigarrow 3)$



Notation:
$x(i \rightsquigarrow j)$ denotes the network x where the tie from i to j is turned into its opposite
$x(i \rightsquigarrow i)$ means that i does not change any of his outgoing ties

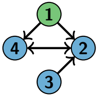
Model definition: continuous-time Markov chain

Absence of co-occurrence

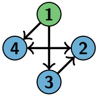


0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x=current state



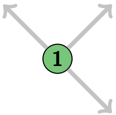
0	**1**	0	1
0	0	0	1
0	1	0	0
0	1	0	0



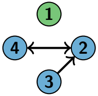
0	0	**1**	1
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 2)$

$x(1 \rightsquigarrow 3)$



$x(1 \rightsquigarrow 4)$



0	0	0	**0**
0	0	0	1
0	1	0	0
0	1	0	0

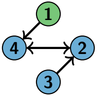
Notation:

$x(i \rightsquigarrow j)$ denotes the network x where the tie from i to j is turned into its opposite

$x(i \rightsquigarrow i)$ means that i does not change any of his outgoing ties

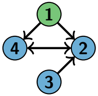
Model definition: continuous-time Markov chain

Absence of co-occurrence



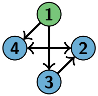
0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x=current state



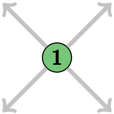
0	1	0	1
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 2)$

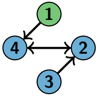


0	0	1	1
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 3)$

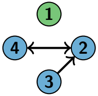


$x(1 \rightsquigarrow 1)$



0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 4)$

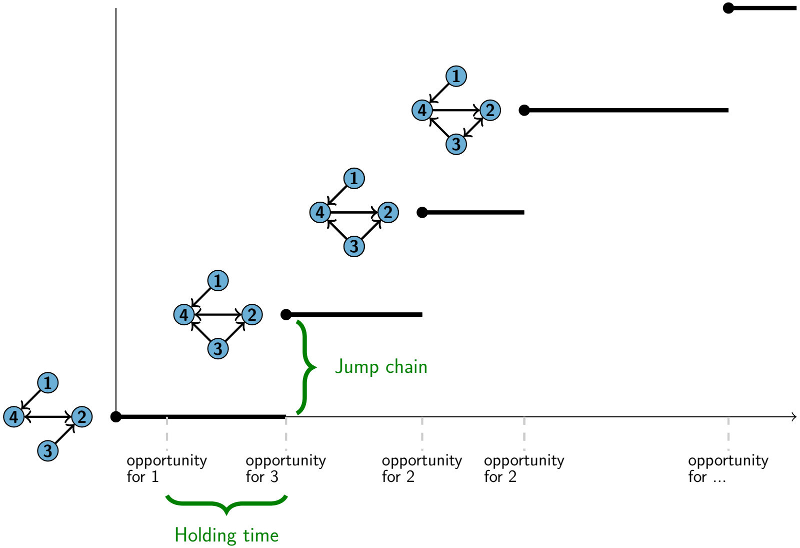


0	0	0	0
0	0	0	1
0	1	0	0
0	1	0	0

Notation:
$x(i \rightsquigarrow j)$ denotes the network x where the tie from i to j is turned into its opposite
$x(i \rightsquigarrow i)$ means that i does not change any of his outgoing ties

Model definition: continuous-time Markov chain

Trajectory



Model definition: continuous-time Markov chain

The evolution process can be decomposed into micro-steps

Micro-step	Continuous-time Markov chain
▸ the time at which i had the opportunity to change	▸ the waiting time until the next opportunity for a change made by an actor i (holding time)
▸ the precise change i made	▸ the probability of changing x_{ij} given that i is allowed to change (jump chain)

Model definition: continuous-time Markov chain

Holding times: rate function

The waiting time between opportunities of change for an actor i is exponentially distributed with parameter λ_i

λ_i is called **rate function**

- *Simplest specification:*
  all actors have the same rate of change λ

$$P(i \textit{ has the opportunity of change}) = \frac{\lambda}{\lambda n} = \frac{1}{n} \quad \forall i \in \mathcal{N}$$

- *More complex specification:*
  actors may change their ties at different frequencies $\lambda_i(\alpha, x, v)$

$$P(i \textit{ has the opportunity of change}) = \frac{\lambda_i(\alpha, x, v)}{\sum_{j=1}^{n} \lambda_j(\alpha, x, v)}$$

Model definition: continuous-time Markov chain

Holding times: rate function

In the following we assume that:

- all actors have the same rate of change

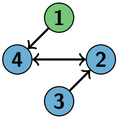
  $\Longrightarrow$ λ is constant over the actors

- the frequencies at which actors have the opportunity to make a change depends on time

  $\Longrightarrow$ λ is not constant over time

Model definition: continuous-time Markov chain

Jump matrix



0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x=current state

0	**1**	0	1
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 2)$ $\quad p_{12} > 0$

0	0	**1**	1
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 3)$ $\quad p_{13} > 0$

0	0	0	**0**
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 4)$ $\quad p_{14} > 0$

0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 1)$ $\quad p_{11} > 0$

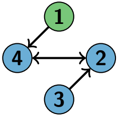
0	0	**1**	1
0	0	0	1
0	**0**	0	0
0	1	0	0

0	0	1	1
0	0	**1**	**0**
0	1	0	0
0	1	0	0

$P(other) = 0$

Model definition: continuous-time Markov chain

Jump matrix



0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x=current state

0	1	0	1
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 2)$ $p_{12} > 0$

0	0	1	1
0	0	0	1
0	1	0	0
0	1	0	0

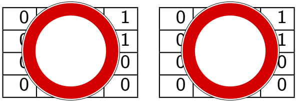
$x(1 \rightsquigarrow 3)$ $p_{13} > 0$

0	0	0	0
0	0	0	1
0	1	0	0
0	1	0	0

$x(1 \rightsquigarrow 4)$ $p_{14} > 0$

0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

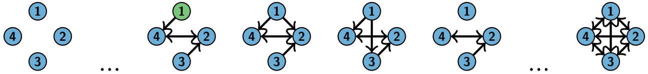
$x(1 \rightsquigarrow 1)$ $p_{11} > 0$

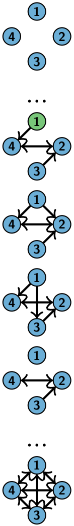


$P(other) = 0$

Model definition: continuous-time Markov chain

Jump matrix





$0 \quad \dots \quad p_{11} \quad p_{12} \quad p_{13} \quad p_{14} \quad \dots \quad 0$

Background: random utility model

Setting

decision makers who face a choice between N-alternatives

Notation:

- i denotes the decision maker
- $J = \{1, \ldots, j, \ldots, N\}$ choice set
  J is **exhaustive** and choices are **mutually exclusive**

Assumption

the decision makers obtain a certain level of profit from each alternative.
The profit is modeled by the *utility function* $U_{ij} : J \to \mathbb{R}$

Decision rule

i chooses the alternative j that assures him the highest profit, i.e.

$$j \; : \; max_{j \in J} \; U_{ij}$$

Background: random utility model

- The researcher does not completely know the decision maker's utility. Therefore, the utility function is decomposed as

$$U_{ij} = F_{ij} + \mathcal{E}_{ij}$$

  - F_{ij} is the deterministic part of the utility (observed!)

$$F_{ij} = \sum_a \gamma_a v_i + \sum_b \delta_b c_j$$

    - v_i variables characterizing the decision maker i
    - c_j variables characterizing the choice j

  - $\mathcal{E}_{ij}$: random term with Gumbel distribution (not observed!)
    The random term are independent and identically distributed

- The probability that i chooses the alternative j is given by

$$p_{ij} = P(U_{ij} > U_{ih},\ \forall\ h \in J) = \frac{e^{F_{ij}}}{\sum_{h=1}^{N} e^{F_{ih}}}$$

Model definition: continuous-time Markov chain

Jump matrix: evaluation function

- Actors change their ties in order to maximize a utility function

$$u_i(\beta, x(i \rightsquigarrow j)) = f_i(\beta, x(i \rightsquigarrow j), v_i, v_j) + \mathcal{E}_{ij}$$

  - $f_i(\beta, x(i \rightsquigarrow j), v_i, v_j)$ is the *evaluation function*
  - $\mathcal{E}_{ij}$ is random term (distributed as a Gumbel r.v.)

- The probability that i changes his outgoing tie towards j is:

$$p_{ij} = \frac{exp\left(f_i(\beta, x(i \rightsquigarrow j), v_i, v_j)\right))}{\sum_{h=1}^{n} exp\left(f_i(\beta, x(i \rightsquigarrow h), v_i, v_j)\right)}$$

- Probability interpretation:
  - p_{ij} is the probability that i changes the tie towards j
  - p_{ii} is the probability of not changing

Model definition: continuous-time Markov chain

Jump matrix: evaluation function

The **evaluation function** is defined as a linear combination

$$f_i(\beta, x(i \rightsquigarrow j), v_i, v_j) = \sum_{k=1}^{K} \beta_k s_{ik}(x(i \rightsquigarrow j), v_i, v_j)$$

- $s_{ik}(x(i \rightsquigarrow j), v_i, v_j)$ is called statistic
- $\beta_k \in \mathbb{R}$ is a statistical parameter

Model definition: continuous-time Markov chain

Jump matrix: evaluation function

The **evaluation function** is defined as a linear combination

$$f_i(\beta, x(i \rightsquigarrow j), v_i, v_j) = \sum_{k=1}^{K} \beta_k s_{ik}(x(i \rightsquigarrow j), v_i, v_j)$$

- $s_{ik}(x(i \rightsquigarrow j), v_i, v_j)$ is called statistic
- $\beta_k \in \mathbb{R}$ is a statistical parameter

N.b.
In the following, we will write:

- x' instead of $x(i \rightsquigarrow j)$
- $s_{ik}(x', v)$ instead of $s_{ik}(x(i \rightsquigarrow j), v_i, v_j)$

to simplify the notation

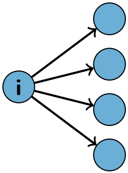
Outline

Evaluation function specification

Endogenous statistics = *dependent on the network structures*

- Outdegree statistic

$$s_{i\_out}(x') = \sum_j x'_{ij}$$

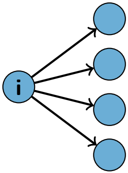


Evaluation function specification

Endogenous statistics = *dependent on the network structures*

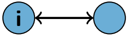
- Outdegree statistic

$$s_{i\_out}(x') = \sum_j x'_{ij}$$



- Reciprocity statistic

$$s_{i\_rec}(x') = \sum_j x'_{ij} x'_{ji}$$

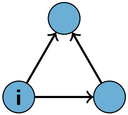


Evaluation function specification

Endogenous statistics = *dependent on the network structures*

- Transitive statistic

$$s_{i\_trans}(x') = \sum_{j,h} x'_{ij} x'_{ih} x'_{jh}$$

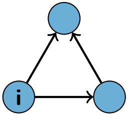


Evaluation function specification

Endogenous statistics = *dependent on the network structures*

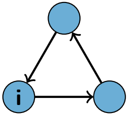
- Transitive statistic

$$s_{i\_trans}(x') = \sum_{j,h} x'_{ij} x'_{ih} x'_{jh}$$



- Three-cycle statistic

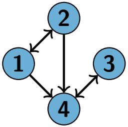
$$s_{i\_cyc}(x') = \sum_{j,h} x'_{ij} x'_{jh} x'_{hi}$$



Evaluation function specification

Example

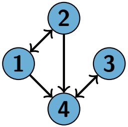
$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

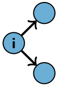
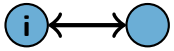
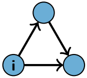


Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

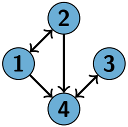


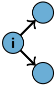
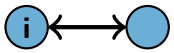
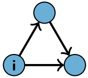
	s_{i_out}	s_{i_rec}	s_{i_trans}
$i \to j$			
$1 \to 1$			
$1 \to 2$			
$1 \to 3$			
$1 \to 4$			

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

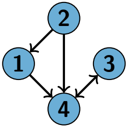


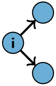
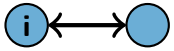
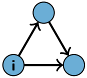
$i \to j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 
$1 \to 1$	2	1	1
$1 \to 2$			
$1 \to 3$			
$1 \to 4$			

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

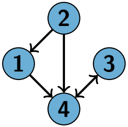


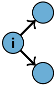
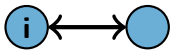
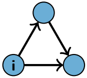
$i \rightarrow j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 
$1 \rightarrow 1$	2	1	1
$1 \rightarrow 2$			
$1 \rightarrow 3$			
$1 \rightarrow 4$			

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

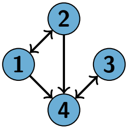


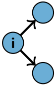
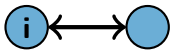
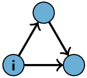
$i \rightarrow j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 
$1 \rightarrow 1$	2	1	1
$1 \rightarrow 2$	1	0	0
$1 \rightarrow 3$			
$1 \rightarrow 4$			

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

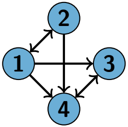


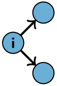
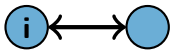
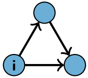
$i \to j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 
$1 \to 1$	2	1	1
$1 \to 2$	1	0	0
$1 \to 3$			
$1 \to 4$			

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

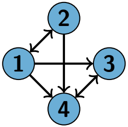


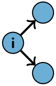
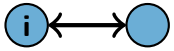
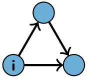
$i \to j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 
$1 \to 1$	2	1	1
$1 \to 2$	1	0	0
$1 \to 3$			
$1 \to 4$			

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

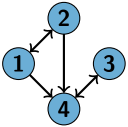


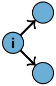
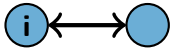
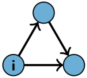
$i \rightarrow j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 
$1 \rightarrow 1$	2	1	1
$1 \rightarrow 2$	1	0	0
$1 \rightarrow 3$	3	1	3
$1 \rightarrow 4$			

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

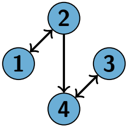


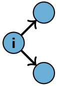
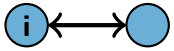
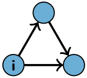
$i \rightarrow j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 
$1 \rightarrow 1$	2	1	1
$1 \rightarrow 2$	1	0	0
$1 \rightarrow 3$	3	1	3
$1 \rightarrow 4$			

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

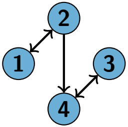


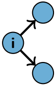
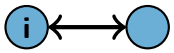
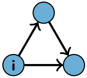
$i \rightarrow j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 
$1 \rightarrow 1$	2	1	1
$1 \rightarrow 2$	1	0	0
$1 \rightarrow 3$	3	1	3
$1 \rightarrow 4$			

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

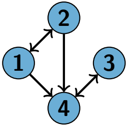


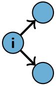
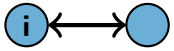
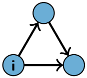
$i \to j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 
$1 \to 1$	2	1	1
$1 \to 2$	1	0	0
$1 \to 3$	3	1	3
$1 \to 4$	1	1	0

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$

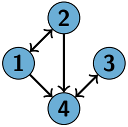


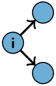
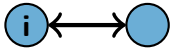
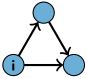
$i \rightarrow j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 	
$1 \rightarrow 1$	2	1	1	-1.75
$1 \rightarrow 2$	1	0	0	-1.00
$1 \rightarrow 3$	3	1	3	-3.25
$1 \rightarrow 4$	1	1	0	-0.50

Evaluation function specification

Example

$$\beta_{out} = -1 \qquad \beta_{rec} = +0.5 \qquad \beta_{trans} = -0.25$$



$i \to j$	s_{i_out} 	s_{i_rec} 	s_{i_trans} 	
$1 \to 1$	2	1	1	-1.75
$1 \to 2$	1	0	0	-1.00
$1 \to 3$	3	1	3	-3.25
$1 \to 4$	1	1	0	-0.50

$$p_{11} = 0.146 \qquad p_{12} = 0.310 \qquad p_{13} = 0.033 \qquad p_{14} = 0.511$$

Evaluation function specification

Exogenous statistics = *related to actor's attributes*

Examples

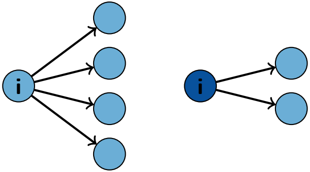
- Friendship among pupils:
  - Smoking: non, occasional, regular
  - Gender: boys, girls
- Trade/Trust (Alliances) among countries:
  - Geographical area: Europe, Asia, North-America,...
  - Worlds: First, Second, Third, Fourth
- Giving advice among employees:
  - seniority
  - office membership

Evaluation function specification

Exogenous statistics *(individual covariate)*

- Covariate-ego statistic

$$s_{i\_cego}(x', v) = v_i \sum_j x'_{ij}$$

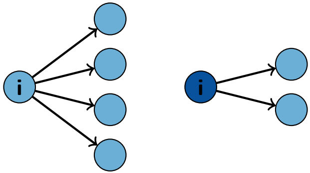


Evaluation function specification

Exogenous statistics *(individual covariate)*

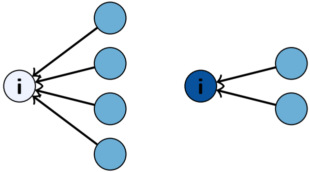
- Covariate-ego statistic

$$s_{i\_cego}(x', v) = v_i \sum_j x'_{ij}$$



- Covariate-alter statistic

$$s_{i\_calt}(x', v) = \sum_j x'_{ij} v_j$$

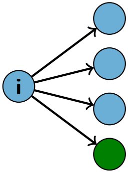


Evaluation function specification

Exogenous statistics *(dyadic covariate)*

- Covariate-related similarity statistic

$$s_{i\_csim}(x', v) = \sum_j x'_{ij}\left(1 - \frac{|v_i - v_j|}{R_V}\right)$$



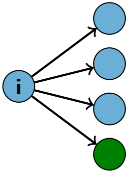
where R_V is the range of V and $\left(1 - \frac{|v_i - v_j|}{R_V}\right)$ is called *similarity score*

Evaluation function specification

Exogenous statistics *(dyadic covariate)*

- Covariate-related similarity statistic

$$s_{i\_csim}(x', v) = \sum_j x'_{ij} \left(1 - \frac{|v_i - v_j|}{R_V}\right)$$



where R_V is the range of V and $\left(1 - \frac{|v_i - v_j|}{R_V}\right)$ is called *similarity score*

Remark:

when V is a binary covariate, the covariate-related similarity can be written in the following way:

$$s_{i\_csim}(x', v) = \sum_j x'_{ij} \mathbb{I}\left\{v_i = v_j\right\}$$

SAOM definition: summary

Model assumptions:

1. Ties have a **tendency to endure over time**
2. The evolution process is a **continuous-time Markov chain**

   2.1 *Waiting time:*
   exponentially distributed with parameter λ

   - constant over the actors
   - period dependent
     i.e. $M + 1$ observations $\Longrightarrow \lambda_1, \cdots \lambda_M$

SAOM definition: summary

Model assumptions:

1. Ties have a **tendency to endure over time**
2. The evolution process is a **continuous-time Markov chain**

   2.2 *Jump chain*

   - At any given moment t one actor has the opportunity to change one of his outgoing ties
   - Actors change their ties in order to maximize a utility function

     $$u_i(\beta, x(i \rightsquigarrow j)) = f_i(\beta, x(i \rightsquigarrow j), v_i, v_j) + \mathcal{E}_{ij}$$

     The probability that i changes his outgoing tie towards j is:

     $$p_{ij} = \frac{exp\left(f_i(\beta, x(i \rightsquigarrow j), v_i, v_j))\right)}{\sum_{h=1}^{n} exp\left(f_i(\beta, x(i \rightsquigarrow h), v_i, v_j)\right)}$$

   - The parameters $\beta_1, \ldots, \beta_k$ are constant over actors and time

SAOM definition: consequences

- Markov property
  - The future configuration of the network depend solely on the current configuration of the network
- At any given moment t one actor has the opportunity to change one of his outgoing ties
  - Simultaneous changes are not allowed
- Actors change their ties in order to maximize a utility function

$$u_i(\beta, x(i \rightsquigarrow j)) = f_i(\beta, x(i \rightsquigarrow j), v_i, v_j) + \mathcal{E}_{ij}$$

  - To compute the evaluation function actors should have full knowledge of the network (existing ties, actors and their attribute)
  - All the actors use the same evaluation function

Evaluation function specification

Which statistics must be included in the evaluation function?

Outdegree and Reciprocity must always be included.
The choice of the other statistics must be determined according to hypotheses derived from theory

Evaluation function specification

Which statistics must be included in the evaluation function?

Outdegree and Reciprocity must always be included.
The choice of the other statistics must be determined according to hypotheses derived from theory

Example
Friendship network

Theory		Statistics
the friend of my friend is also my friend	$\Rightarrow$	transitive effect

Evaluation function specification

Which statistics must be included in the evaluation function?

Outdegree and Reciprocity must always be included.
The choice of the other statistics must be determined according to hypotheses derived from theory

Example
Friendship network

Theory		Statistics
the friend of my friend is also my friend	$\Rightarrow$	transitive effect
girls trust girls boys trust boys	$\Rightarrow$	covariate-related similarity

Outline

Simulating network evolution

Aim: given $x(t_0)$ and fixed parameter values, provide $x^{sim}(t_1)$ according to the process behind the SAOM

$$\Downarrow$$

produce a possible series of micro-steps between t_0 and t_1

Input

$x(t_0) =$ network at time t_0

$\lambda =$ rate parameter

$\beta = (\beta_1, \ldots, \beta_k) =$ evaluation function parameters

Output

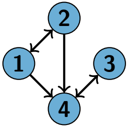
$x^{sim}(t_1) =$ network at time t_1

Simulating network evolution

Algorithm: Network evolution

Input: $x(t_0)$, λ, β, n
Output: $x^{sim}(t_1)$
$t \leftarrow 0$
$x \leftarrow x(t_0)$
while $condition = TRUE$ **do**
  $dt \sim Exp(n\lambda)$
  $i \sim Uniform(1, \ldots, n)$
  $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
  if $i \neq j$ **then**
    $x \leftarrow x(i \rightsquigarrow j)$
  else
    $x \leftarrow x$
  $t \leftarrow t + dt$
$x^{sim}(t_1) \leftarrow x$
return $x^{sim}(t_1)$

t = time
dt = holding time between consecutive opportunities to change
$\sim$ = generated from



$n = 4$

$\lambda = 1.5$

$\beta = (\beta_{out}, \beta_{rec}, \beta_{trans})$
=(-1,0.5,-0.25)

Simulating network evolution

Algorithm: Network evolution

Input: $x(t_0)$, λ, β, n
Output: $x^{sim}(t_1)$
$t \leftarrow 0$
$x \leftarrow x(t_0)$
while $condition = TRUE$ **do**
  $dt \sim Exp(n\lambda)$
  $i \sim Uniform(1, \ldots, n)$
  $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
  if $i \neq j$ **then**
    $x \leftarrow x(i \rightsquigarrow j)$
  else
    $x \leftarrow x$
  $t \leftarrow t + dt$
$x^{sim}(t_1) \leftarrow x$
return $x^{sim}(t_1)$

t = time
dt = holding time between consecutive opportunities to change
$\sim$ = generated from

Generate the time elapsed between t_0 and the first opportunity to change

The more intuitive way to generate dt is:

- generate the waiting time for each actor i

$$t_i \sim Exp(\lambda)$$

- $dt = \min_{1 \leq i \leq n} \{t_i\}$

but this requires the generation of n numbers.

Simulating network evolution

Algorithm: Network evolution

Input: $x(t_0)$, λ, β, n
Output: $x^{sim}(t_1)$
$t \leftarrow 0$
$x \leftarrow x(t_0)$
while $condition = TRUE$ **do**
    $dt \sim Exp(n\lambda)$
    $i \sim Uniform(1, \ldots, n)$
    $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
    if $i \neq j$ **then**
        $x \leftarrow x(i \rightsquigarrow j)$
    else
        $x \leftarrow x$
    $t \leftarrow t + dt$
$x^{sim}(t_1) \leftarrow x$
return $x^{sim}(t_1)$

t = time
dt = holding time between consecutive opportunities to change
$\sim$ = generated from

Generate the time elapsed between t_0 and the first opportunity to change

To avoid the generation of n numbers, we use the following result:

If

$$T_i \sim Exp(\lambda_i), \quad 1 \leq i \leq n$$

and $T_1, \ldots, T_n$ *are mutually independent, then*

$$DT = \min\{T_1, \ldots, T_n\} \sim Exp(\sum_{i=1}^{n} \lambda_i)$$

e.g. $dt = 0.0027$

Simulating network evolution

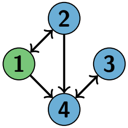
Algorithm: Network evolution

Input: $x(t_0)$, λ, β, n
Output: $x^{sim}(t_1)$
$t \leftarrow 0$
$x \leftarrow x(t_0)$
while $condition = TRUE$ **do**
  $dt \sim Exp(n\lambda)$
  $i \sim Uniform(1, \ldots, n)$
  $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
  if $i \neq j$ **then**
    $x \leftarrow x(i \rightsquigarrow j)$
  else
    $x \leftarrow x$
  $t \leftarrow t + dt$
$x^{sim}(t_1) \leftarrow x$
return $x^{sim}(t_1)$

t = time
dt = holding time between consecutive opportunities to change
$\sim$ = generated from

Select the actor i who has the opportunity to change

e.g. $i = 1$



Simulating network evolution

Algorithm: Network evolution

Input: $x(t_0)$, λ, β, n
Output: $x^{sim}(t_1)$
$t \leftarrow 0$
$x \leftarrow x(t_0)$
while $condition = TRUE$ **do**
    $dt \sim Exp(n\lambda)$
    $i \sim Uniform(1, \ldots, n)$
    $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
    if $i \neq j$ **then**
        $x \leftarrow x(i \rightsquigarrow j)$
    else
        $x \leftarrow x$
    $t \leftarrow t + dt$
$x^{sim}(t_1) \leftarrow x$
return $x^{sim}(t_1)$

t = time
dt = holding time between consecutive opportunities to change
$\sim$ = generated from

Select j, the actor towards i is going to change his outgoing tie

$i \to j$	f_i	p_{ij}
1 $\to$ 1	-1.75	0.15
1 $\to$ 2	-1.00	0.31
1 $\to$ 3	-3.25	0.03
1 $\to$ 4	-0.5	0.51

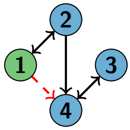
Simulating network evolution

Algorithm: Network evolution

Input: $x(t_0)$, λ, β, n
Output: $x^{sim}(t_1)$
$t \leftarrow 0$
$x \leftarrow x(t_0)$
while $condition = TRUE$ **do**
  $dt \sim Exp(n\lambda)$
  $i \sim Uniform(1, \ldots, n)$
  $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
  if $i \neq j$ **then**
    $x \leftarrow x(i \rightsquigarrow j)$
  else
    $x \leftarrow x$
  $t \leftarrow t + dt$
$x^{sim}(t_1) \leftarrow x$
return $x^{sim}(t_1)$

t = time
dt = holding time between consecutive opportunities to change
$\sim$ = generated from

e.g. $j = 4$



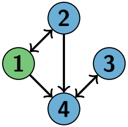
Simulating network evolution

Algorithm: Network evolution

Input: $x(t_0)$, λ, β, n
Output: $x^{sim}(t_1)$
$t \leftarrow 0$
$x \leftarrow x(t_0)$
while $condition = TRUE$ **do**
  $dt \sim Exp(n\lambda)$
  $i \sim Uniform(1, \ldots, n)$
  $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
  if $i \neq j$ **then**
    $x \leftarrow x(i \rightsquigarrow j)$
  else
    $x \leftarrow x$
  $t \leftarrow t + dt$
$x^{sim}(t_1) \leftarrow x$
return $x^{sim}(t_1)$

t = time
dt = holding time between consecutive opportunities to change
$\sim$ = generated from

e.g. $j = 1$



Simulating network evolution

Algorithm: Network evolution

Input: $x(t_0)$, λ, β, n
Output: $x^{sim}(t_1)$
$t \leftarrow 0$
$x \leftarrow x(t_0)$
while $condition = TRUE$ **do**
  $dt \sim Exp(n\lambda)$
  $i \sim Uniform(1, \ldots, n)$
  $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
  if $i \neq j$ **then**
    $x \leftarrow x(i \rightsquigarrow j)$
  else
    $x \leftarrow x$
  $t \leftarrow t + dt$
$x^{sim}(t_1) \leftarrow x$
return $x^{sim}(t_1)$

t = time
dt = holding time between consecutive opportunities to change
$\sim$ = generated from

e.g. $t = 0 + 0.0027$

Simulating network evolution

Two different stopping rules:

1. *Unconditional* simulation:

   the simulation of the network evolution carries on until a predetermined time length has elapsed (usually until $t = 1$)

Simulating network evolution

Two different stopping rules:

1. *Unconditional* simulation:

   the simulation of the network evolution carries on until a predetermined time length has elapsed (usually until $t = 1$)

2. *Conditional* simulation on the observed number of changes:

   the simulation runs on until

$$\sum_{\substack{i,j=1 \\ i \neq j}}^{n} \left| x_{ij}^{obs}(t_1) - x_{ij}(t_0) \right| = \sum_{\substack{i,j=1 \\ i \neq j}}^{n} \left| x_{ij}^{sim}(t_1) - x_{ij}(t_0) \right|$$

   This criterion can be generalized conditioning on any other explanatory variable.

Simulating network evolution

Use of simulations:

- simulating the network evolution between two consecutive time points

  N.b.
  For simulations of 3 or more waves ($M \geq 2$), the simulation for wave $m+1$ starts at the simulated network for wave m.

- provide possible scenarios of the network evolution according to different values of the parameters

- estimate the parameter of the model

- evaluate the goodness of fit of the model

Outline

Estimating the parameter of the SAOM

Issue

Given

$$x(t_0),\ x(t_1), \ldots,\ x(t_M)$$

and a specification of the SAOM, we want to estimate

$$\theta = (\lambda_1,\ \ldots, \lambda_M, \beta_1,\ \ldots, \beta_K)$$

Two estimation methods are implemented in `Rsiena`:

1. Method of Moments
2. Maximum-likelihood estimation

Background: expected value

Definition

Let X be a random variable with probability distribution $\varphi(x;\theta)$

The **expected value** (or **moment**) of X, denoted by $E_\theta[X]$, is:

$$E_\theta[X] = \sum_{x \in \mathcal{S}} x \cdot \varphi(x,\theta)$$

if X is discrete and

$$E_\theta[X] = \int_{x \in \mathcal{S}} x \cdot \varphi(x,\theta) dx$$

if X is continuous

Let $(x_1, \ \ldots, \ x_q)$ a sample of q observations from the r.v. X.

The **sample counterpart** of $E_\theta[X]$, denoted by μ, is defined by:

$$\mu = \frac{1}{q}\sum_{i=1}^{q} x_i$$

Background: Method of Moments (MoM)

Definition

The method of moment estimator for θ is found by equating the expected value $E_\theta[X]$ to its sample counterpart μ

$$E_\theta[X] = \mu$$

and solving the resulting equation for the unknown parameter. The estimate for θ is denoted by $\widehat{\theta}$.

Background: Method of Moments (MoM)

Definition

The method of moment estimator for θ is found by equating the expected value $E_\theta[X]$ to its sample counterpart μ

$$E_\theta[X] = \mu$$

and solving the resulting equation for the unknown parameter.
The estimate for θ is denoted by $\widehat{\theta}$. In practice:

1. Compute the expected value $E_\theta[X]$
2. Compute the sample counterpart $\mu = \frac{1}{q}\sum_{i=1}^{q} x_i$
3. Solve the moment equation $E_\theta[X] = \mu$ for θ

Motivation

One can observe that the expected value of a certain distribution usually depends on the parameter θ

Background: Method of Moments (MoM)

Example

Let T be the r.v. describing the waiting times between two consecutive opportunities for change for an actor in a network evolution process described by the SAOM.
A sample is reported in the following table:

	1	2	3	4	5	6	7	8	9	10
t_i	0.33	0.08	0.06	0.01	0.04	0.11	0.03	0.18	0.02	0.07

Background: Method of Moments (MoM)

Example

Let T be the r.v. describing the waiting times between two consecutive opportunities for change for an actor in a network evolution process described by the SAOM.
A sample is reported in the following table:

	1	2	3	4	5	6	7	8	9	10
t_i	0.33	0.08	0.06	0.01	0.04	0.11	0.03	0.18	0.02	0.07

From the assumptions of the SAOM it follows that $T \sim Exp(\lambda)$

$$\varphi_T(t) = \lambda e^{-\lambda t} \qquad \lambda, t > 0$$

Estimate the rate parameter λ using the MoM

Background: Method of Moments (MoM)

Example

1. Compute the expected value

$$
\begin{aligned}
E_\lambda[T] &= \int_0^{+\infty} t \cdot \varphi_T(t) dt = \int_0^{+\infty} t \cdot \lambda e^{-\lambda t} dt \\
&= \underbrace{\left[-t \cdot e^{-\lambda t}\right]_0^{+\infty} - \int_0^{+\infty} -e^{-\lambda t} dt}_{integration\ \ by\ \ parts} \\
&= 0 - \left[-\frac{1}{\lambda} e^{-\lambda t}\right]_0^{+\infty} = \frac{1}{\lambda}
\end{aligned}
$$

Background: Method of Moments (MoM)

Example

	1	2	3	4	5	6	7	8	9	10
t_i	0.33	0.08	0.06	0.01	0.04	0.11	0.03	0.18	0.02	0.07

2. Compute the sample counterpart:

$$\mu = \frac{1}{10}\sum_{i=1}^{10} t_i = \frac{0.93}{10} = 0.093$$

3. The estimate for λ is the solution of:

$$E_\lambda[T] = \mu$$
$$\frac{1}{\lambda} = \mu$$

   namely

$$\widehat{\lambda} = \frac{1}{\mu} = \frac{1}{0.093} = 10.75$$

Background: Generalizations of MoM

The principle of the MoM can be generalized to any function $s : \mathcal{S} \longmapsto \mathbb{R}$.

1. Expected value of $s(X)$:

$$E_\theta[s(X)] = \sum_{x \in \mathcal{S}} s(x)\varphi(x, \theta)$$

$$E_\theta[s(X)] = \int_{x \in \mathcal{S}} s(x)\varphi(x, \theta) dx$$

2. Corresponding sample moment:

$$\gamma = \frac{1}{q} \sum_{i=1}^{q} s(x_i)$$

3. Moment equation:

$$E_\theta[s(X)] = \gamma$$

The functions $s(X)$ are called *statistics*

Background: Generalizations of MoM

The MoM can be applied also in situations where $\theta = (\theta_1, \ldots, \theta_p)$.

1. Definition of p statistics $(s_1(X), \ldots, s_p(X))$

2. Definition of p moment conditions:

$$E_\theta[s_1(X)] = \gamma_1$$

$$E_\theta[s_2(X)] = \gamma_2$$

$$\ldots$$

$$E_\theta[s_p(X)] = \gamma_p$$

3. Solving the resulting equations for the unknown parameters

Estimating the parameter of the SAOM using MoM

Aim: estimate θ using the MoM

$$\theta = (\lambda_1,\ \ldots,\ \lambda_M,\ \beta_1,\ \ldots,\ \beta_K)$$

Estimating the parameter of the SAOM using MoM

Aim: estimate θ using the MoM

$$\theta = (\lambda_1, \ldots, \lambda_M, \beta_1, \ldots, \beta_K)$$

In practice:

1. find $M + K$ statistics
2. set the theoretical expected value of each statistic equal to its sample counterpart
3. solve the resulting system of equations with respect to θ.

For simplicity, let us assume to have observed a network at two time points t_0 and t_1 and to condition the estimation on the first observation $x(t_0)$

1. Defining the statistics

The rate parameter λ describes the frequency at which changes can potentially happen.

$$s_\lambda(X(t_1), X(t_0) | X(t_0) = x(t_0)) = \sum_{i,j=1}^{n} |X_{ij}(t_1) - X_{ij}(t_0)|$$

1. Defining the statistics

The rate parameter λ describes the frequency at which changes can potentially happen.

$$s_\lambda(X(t_1), X(t_0) | X(t_0) = x(t_0)) = \sum_{i,j=1}^{n} |X_{ij}(t_1) - X_{ij}(t_0)|$$

Reason

	$\lambda = 2$	$\lambda = 3$	$\lambda = 4$
s_λ	94	135	171

$\Rightarrow$ higher values of λ leads to higher values of s_λ

1. Defining the statistics

The parameter β_k quantifies the role played by each effect in the network evolution.

$$s_k(X(t_1)|X(t_0) = x(t_0)) = \sum_{i=1}^{n} s_{ik}(X(t_1))$$

1. Defining the statistics

The parameter β_k quantifies the role played by each effect in the network evolution.

$$s_k(X(t_1)|X(t_0) = x(t_0)) = \sum_{i=1}^{n} s_{ik}(X(t_1))$$

Example

Let us consider the outdegree:

$$s_{out}(X(t_1)|X(t_0) = x(t_0)) = \sum_{i=1}^{n} s_{i\_out}(X(t_1)) = \sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}(t_1)$$

	$\beta_{out} = -2.5$	$\beta_{out} = -2$	$\beta_{out} = -1.5$
s_{out}	195	214	234

$\Rightarrow$ higher values of β_{out} leads to higher values of s_{out}

1. Defining the statistics

Generalizing to M periods:

- Statistics for the rate function parameters

$$s_{\lambda_1}(X(t_1), X(t_0) | X(t_0) = x(t_0)) = \sum_{i,j=1}^{n} |X_{ij}(t_1) - X_{ij}(t_0)|$$

$\dots$

$$s_{\lambda_M}(X(t_M), X(t_{M-1}) | X(t_{M-1}) = x(t_{M-1})) = \sum_{i,j=1}^{n} |X_{ij}(t_M) - X_{ij}(t_{M-1})|$$

1. Defining the statistics

Generalizing to M periods:

- Statistics for the rate function parameters

$$s_{\lambda_1}(X(t_1), X(t_0) | X(t_0) = x(t_0)) = \sum_{i,j=1}^{n} |X_{ij}(t_1) - X_{ij}(t_0)|$$

$\dots$

$$s_{\lambda_M}(X(t_M), X(t_{M-1}) | X(t_{M-1}) = x(t_{M-1})) = \sum_{i,j=1}^{n} |X_{ij}(t_M) - X_{ij}(t_{M-1})|$$

- Statistics for the objective function parameters:

$$\sum_{m=1}^{M} s_{mk}(X(t_m) | X(t_{m-1}) = x(t_{m-1})) = \sum_{m=1}^{M} s_{mk}(X(t_m))$$

2. Setting the moment equations

The MoM estimator for θ is defined as the solution of the system of $M+K$ equations

$$\begin{cases} E_\theta\left[s_{\lambda_m}(X(t_m), X(t_{m-1}) | X(t_{m-1}) = x(t_{m-1}))\right] = s_{\lambda_m}(x(t_m), x(t_{m-1})) \\ \\ E_\theta\left[\sum_{m=1}^{M} s_{mk}(X(t_m) | X(t_{m-1}) = x(t_{m-1}))\right] = \sum_{m=1}^{M} s_{mk}(x(t_m)) \end{cases}$$

with $m = 1, \ldots, M$ and $k = 1, \cdots, K$

3. Solving the moment equations

Simplified notation:

- S: $(M+K)$-dimensional vector of statistics
- s: $(M+K)$-dimensional vector of the observed values of the statistics

Consequently, the system of moment equations can be written as

$$E_\theta[S] = s$$

or equivalently as

$$E_\theta[S - s] = 0$$

3. Solving the moment equations

Simplified notation:

- S: $(M+K)$-dimensional vector of statistics
- s: $(M+K)$-dimensional vector of the observed values of the statistics

Consequently, the system of moment equations can be written as

$$E_\theta[S] = s$$

or equivalently as

$$E_\theta[S - s] = 0$$

Problem: analytical procedures cannot be applied to solve this system

3. Solving the moment equations

Simplified notation:

- S: $(M+K)$-dimensional vector of statistics
- s: $(M+K)$-dimensional vector of the observed values of the statistics

Consequently, the system of moment equations can be written as

$$E_\theta[S] = s$$

or equivalently as

$$E_\theta[S - s] = 0$$

Problem: analytical procedures cannot be applied to solve this system

Solution: stochastic approximation method
i.e. an iterative stochastic algorithm that attempt to find zeros of functions which cannot be analytically computed

3. Solving the moment equations

Stochastic approximation method

Given an initial guess θ_0 for the parameter θ, the procedure can be roughly depicted as follows:

$$
\begin{array}{ccccc}
\theta_0 & \xrightarrow{\textit{approximation}} & E_{\theta_0}[S-s] & \xrightarrow{\textit{update}} & \theta_1 \\
\theta_1 & \xrightarrow{\textit{approximation}} & E_{\theta_1}[S-s] & \xrightarrow{\textit{update}} & \theta_2 \\
\dots & \xrightarrow{\textit{approximation}} & \dots & \xrightarrow{\textit{update}} & \dots \\
\theta_{i-1} & \xrightarrow{\textit{approximation}} & E_{\theta_{i-1}}[S-s] & \xrightarrow{\textit{update}} & \theta_i \\
\dots & \xrightarrow{\textit{approximation}} & \dots & \xrightarrow{\textit{update}} & \dots
\end{array}
$$

until a certain criterion is satisfied

3. Solving the moment equations

Stochastic approximation method

Approximation: Monte Carlo method

1. Given $x(t_0)$ and θ_i, we simulate the network evolution q times

$$x^{(1)}(t_1),\ x^{(1)}(t_2),\ \ldots,\ x^{(1)}(t_M)$$

$$\ldots$$

$$x^{(q)}(t_1),\ x^{(q)}(t_2),\ \ldots,\ \ x^{(q)}(t_M)$$

3. Solving the moment equations

Stochastic approximation method

Approximation: Monte Carlo method

1. Given $x(t_0)$ and θ_i, we simulate the network evolution q times

$$x^{(1)}(t_1),\ x^{(1)}(t_2),\ \ldots,\ x^{(1)}(t_M)$$

$$\ldots$$

$$x^{(q)}(t_1),\ x^{(q)}(t_2),\ \ldots,\ x^{(q)}(t_M)$$

1. For each sequence compute the value $S^{(l)}$ taken by S (footnotesize $l = 1, \ldots, n$)

3. Solving the moment equations

Stochastic approximation method

Approximation: Monte Carlo method

1. Given $x(t_0)$ and θ_i, we simulate the network evolution q times

$$x^{(1)}(t_1),\ x^{(1)}(t_2),\ \ldots,\ x^{(1)}(t_M)$$

$$\ldots$$

$$x^{(q)}(t_1),\ x^{(q)}(t_2),\ \ldots,\ \ x^{(q)}(t_M)$$

1. For each sequence compute the value $S^{(l)}$ taken by S (footnotesize $l = 1, \ldots, n$)

2. Approximate the expected value by

$$\overline{S} = \frac{1}{q} \sum_{l=1}^{q} S^{(l)}$$

3. Solving the moment equations

Stochastic approximation method

Approximation: Monte Carlo method

1. Given $x(t_0)$ and θ_i, we simulate the network evolution q times

$$x^{(1)}(t_1),\ x^{(1)}(t_2),\ \dots,\ x^{(1)}(t_M)$$

$$\dots$$

$$x^{(q)}(t_1),\ x^{(q)}(t_2),\ \dots,\ \ x^{(q)}(t_M)$$

1. For each sequence compute the value $S^{(l)}$ taken by S (footnotesize $l = 1, \dots, n$)

2. Approximate the expected value by

$$\overline{S} = \frac{1}{q} \sum_{l=1}^{q} S^{(l)} \to E_\theta[S]$$

3. Solving the moment equations

Stochastic approximation method

Approximation: Monte Carlo method

Example

1. Given:

   - $x(t_0)$
   - $\theta = (\lambda_1 = 10.69, \lambda_2 = 8.82, \beta_{out} = -2.63, \beta_{rec} = 2.17, \beta_{trans} = 0.46)$

   simulate the network evolution $q = 1000$ times

$$x^{(1)}(t_1),\ x^{(1)}(t_2),\ \ldots,\ x^{(1)}(t_M)$$

$$\ldots$$

$$x^{(q)}(t_1),\ x^{(q)}(t_2),\ \ldots,\ x^{(q)}(t_M)$$

Approximation: Monte Carlo Method

Example

2. Compute the value assumed by S_{out} for each sequence of networks

$$S_{out}^{(l)} = \sum_{m=1}^{M} \sum_{i=1}^{n} \sum_{j=1}^{n} x_{ij}^{(l)}(t_m)$$

sim	1	2	3	4	5	6	7	8	...
Nr. Edges	942	874	1047	881	865	866	999	948	...

Approximation: Monte Carlo Method

Example

3. Approximate the expected value by

$$\overline{S}_{out} = \frac{1}{q}\sum_{i=1}^{q} S_{out}^{(l)}$$

$$\overline{S}_{out} = \frac{942 + 874 + 1047 + 881 + 865 + 866 + 999 + 948 + \ldots}{1000} \approx 912$$

3. Solving the moment equations

Stochastic approximation method

Updating rule: the Robbins-Monro (RM) algorithm

Iterative algorithm to find the solution to

$$E_\theta[S] = s$$

The value of θ is iteratively updated according to:

$$\widehat{\theta}_{i+1} = \widehat{\theta}_i - a_i \widehat{D}^{-1}(S_i - s)$$

where:

- a_i is a series such that

$$\lim_{i\to\infty} a_i = 0 \qquad \sum_{i=1}^{\infty} a_i = \infty \qquad \sum_{i=1}^{\infty} a_i^2 < \infty$$

- $\widehat{D}$ is a diagonal matrix with elements

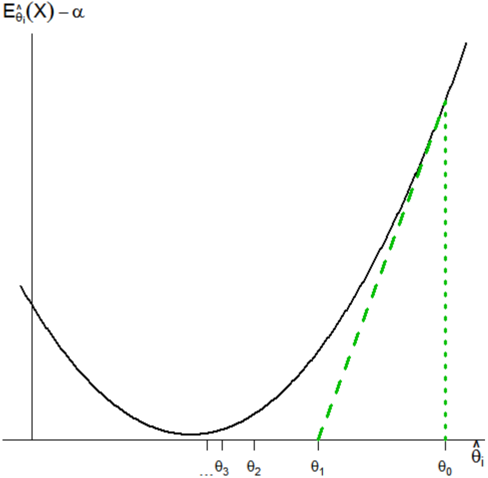
$$\widehat{D} = \frac{\partial}{\partial \widehat{\theta}_i} E_{\widehat{\theta}_i}[S]$$

3. Solving the moment equations

Stochastic approximation method

Updating rule: the RM algorithm

Intuitively:

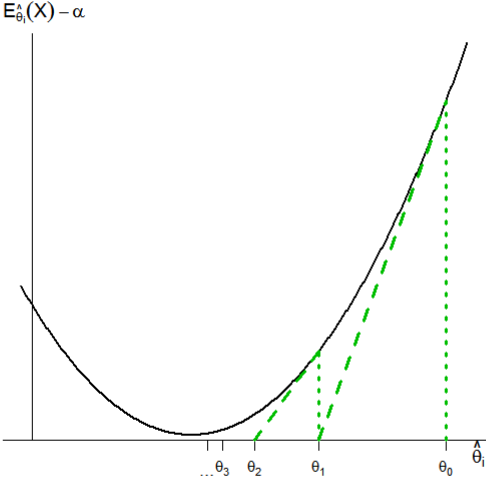


3. Solving the moment equations

Stochastic approximation method

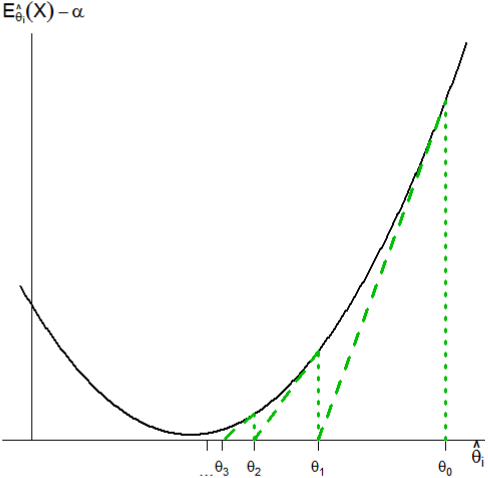
Updating rule: the RM algorithm

Intuitively:



Updating rule: the RM algorithm

Intuitively:

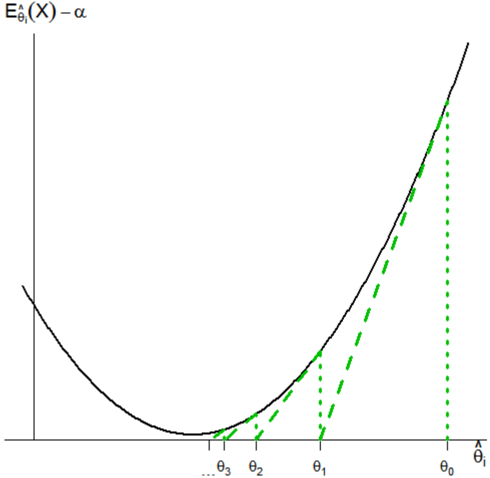


3. Solving the moment equations

Stochastic approximation method

Updating rule: the RM algorithm

Intuitively:



Estimating the parameter of the SAOM

Issue

Given

$$x(t_0),\ x(t_1), \ldots,\ x(t_M)$$

and a specification of the SAOM, we want to estimate

$$\theta = (\lambda_1,\ \ldots, \lambda_M, \beta_1,\ \ldots, \beta_K)$$

Two estimation methods are implemented in `Rsiena`:

1. Method of Moments
2. Maximum-likelihood estimation

Background: the Maximum-likelihood estimation (MLE)

Definition

Suppose that X is a r.v. with probability distribution $\varphi(x, \theta)$, $\theta \in \Theta \subset \mathbb{R}^k$.
Let $x = (x_1, x_2, \ldots, x_q)$ be the observed value of a random sample

The **likelihood function** associated with the observed data is:

$$L(\theta) : \Theta \to \mathbb{R}; \quad \theta \longmapsto P_\theta(x_1, \ldots, x_q)$$

defined as:

$$L(\theta) = \prod_{i=1}^{q} \varphi(x_i, \theta)$$

A parameter vector $\widehat{\theta}$ maximizing L:

$$\widehat{\theta} = arg \max_{\theta \in \Theta} L(\theta)$$

is called a **maximum likelihood estimate** for θ

Background: the Maximum-likelihood estimation (MLE)

In practice, it is easier to compute $\widehat{\theta}$ using the *log-likelihood function*, i.e. $log(L(\theta))$

$$\widehat{\theta} = arg \max_{\theta \in \Theta} log(L(\theta))$$

N.b.

The logarithm is a monotonic increasing function

Background: the Maximum-likelihood estimation (MLE)

Example

Let T be the r.v. describing the waiting times between two consecutive opportunities for change for an actor in a network evolution process described by the SAOM.
A sample is reported in the following table:

	1	2	3	4	5	6	7	8	9	10
t_i	0.33	0.08	0.06	0.01	0.04	0.11	0.03	0.18	0.02	0.07

From the assumptions of the SAOM it follows that $T \sim Exp(\lambda)$

$$\varphi_T(t, \lambda) = \lambda e^{-\lambda t} \qquad \lambda, t > 0$$

Estimate the rate parameter λ according to the MLE.

Background: the Maximum-likelihood estimation (MLE)

Example

Finding an estimate for θ requires:

1. computing the (log-)likelihood of the evolution process
2. maximizing the (log-)likelihood

1. Computing the likelihood of the evolution process

$$L(\lambda) = \prod_{i=1}^{q} f_T(t_i, \lambda) = \prod_{i=1}^{q} \lambda e^{-\lambda t_i} = \lambda^q e^{-\lambda \sum_{i=1}^{q} t_i}$$

$$log(L(\lambda)) = log\left(\lambda^q e^{-\lambda \sum_{i=1}^{q} t_i}\right) = q \cdot log(\lambda) - \lambda \sum_{i=1}^{q} t_i$$

Background: the Maximum-likelihood estimation (MLE)

Example

2. Maximizing the (log-)likelihood

$$\begin{aligned}\frac{\partial}{\partial\lambda} log(L(\lambda)) &= 0 \\ \frac{q}{\lambda} - \sum_{i=1}^{q} t_i &= 0 \Longrightarrow \\ \lambda &= \frac{q}{\sum_{i=1}^{q} t_i} \quad (stationary\ point)\end{aligned}$$

Checking that this stationary point is a maximum

$$\frac{\partial^2}{\partial\lambda^2} log(L(\lambda)) = -\frac{q}{\lambda^2} < 0$$

Therefore, $\widehat{\lambda} = 10.75$

1. Computing the (log-)likelihood of the evolution process

For semplicity, let us consider only two observations $x(t_0)$ and $x(t_1)$

The model assumptions allow to decompose the process in a series of micro-steps:

$$\{(T_r, i_r, j_r), r = 1, \dots, R\}$$

- T_r: time point for an opportunity for change,
- i_r: actor who has the opportunity to change
- j_r: actor towards whom the tie is changed

Given the sequence $\{(T_r, i_r, j_r), r = 1, \dots, R\}$, the likelihood of the evolution process

$$logL(\theta) = log\left(\prod_{r=1}^{R} P_\theta((T_r, i_r, j_r))\right) \propto log\left(\frac{(n\lambda)^R}{R!} e^{-n\lambda} \prod_{r=1}^{R} \frac{1}{n} p_{i_r j_r}(\beta, x(T_r))\right)$$

2. Maximizing the (log-)likelihood

Problem:
we cannot observe the complete data, i.e., the complete series of micro-steps that lead from $x(t_0)$ to $x(t_1)$, from $x(t_1)$ to $x(t_2)$, ...

$$\Downarrow$$

we cannot compute the L of the observed data

$$\Downarrow$$

a stochastic approximation method must be applied.

2. Maximizing the (log-)likelihood

Stochastic approximation method

Given an initial guess θ_0 for the parameter θ, the procedure can be roughly depicted as follows:

$$
\begin{array}{ccccc}
\theta_0 & \xrightarrow{approximation} & \frac{\partial}{\partial\theta} log(L(\theta_0)) & \xrightarrow{update} & \theta_1 \\
\theta_1 & \xrightarrow{approximation} & \frac{\partial}{\partial\theta} log(L(\theta_1)) & \xrightarrow{update} & \theta_2 \\
\dots & \xrightarrow{approximation} & \dots & \xrightarrow{update} & \dots \\
\theta_{i-1} & \xrightarrow{approximation} & \frac{\partial}{\partial\theta} log(L(\theta_{i-1})) & \xrightarrow{update} & \theta_i \\
\dots & \xrightarrow{approximation} & \dots & \xrightarrow{update} & \dots
\end{array}
$$

until a certain criterion is satisfied

2. Maximizing the (log-)likelihood

Stochastic approximation method

Approximation: augmented data method

Definition

The *augmented data* (or *sample path*) consist of the sequence of tie changes that brings the network from $x(t_0)$ to $x(t_1)$

$$(i_1, j_1), \ldots, (i_R, j_R)$$

Formally:

$$\underline{v} = \{(i_1, j_1), \ldots, (i_R, j_R)\} \in \mathcal{V}$$

where $\mathcal{V}$ is the set of all sample paths connecting $x(t_0)$ and $x(t_1)$.

Approximation: augmented data method

Definition

The *augmented data* (or *sample path*) consist of the sequence of tie changes that brings the network from $x(t_0)$ to $x(t_1)$

$$(i_1, j_1), \ldots, (i_R, j_R)$$

Formally:

$$\underline{v} = \{(i_1, j_1), \ldots, (i_R, j_R)\} \in \mathcal{V}$$

where $\mathcal{V}$ is the set of all sample paths connecting $x(t_0)$ and $x(t_1)$.

We can approximate the (log-)likelihood function of the observed data using the probability of $\underline{v}$

$$logP(\underline{v}|x(t_0), x(t_1)) \propto log\left(\frac{(n\lambda)^R}{R!} e^{-n\lambda} \prod_{r=1}^{R} \frac{1}{n} p_{i_r j_r}(\beta, x(T_r))\right)$$

2. Maximizing the (log-)likelihood

Stochastic approximation method

Updating rule

We would like to solve the equation:

$$\frac{\partial}{\partial\theta} log(L(\theta)) = 0$$

Given $\widehat{\theta}_i$ and the corresponding approximation of the score function:

$$\frac{\partial}{\partial\theta} log(L(\widehat{\theta}_i; v_m^{(i)}))$$

we update the parameter estimate using the Robbins-Monro step

$$\theta_{i+1} = \theta_i + a_i D^{-1} \frac{\partial}{\partial\theta} log(L(\widehat{\theta}_i; v_m^{(i)}))$$

where D is a diagonal matrix with elements

$$D^{-1} = \left[\frac{\partial^2}{\partial\theta^2} log(L(\widehat{\theta}_i; v_m^{(i)}))\right]^{-1}$$

Outline

SAOMs

Recall

The evolution process is a **continuous-time Markov chain**:

- At any given moment t one actor has the opportunity to change one of his outgoing ties

- Actors change their ties in order to maximize a utility function

$$u_i(\beta, x(i \rightsquigarrow j)) = f_i(\beta, x(i \rightsquigarrow j), v_i, v_j) + \mathcal{E}_{ij}$$

  The probability that i changes his outgoing tie towards j is:

$$p_{ij} = \frac{exp\left(f_i(\beta, x(i \rightsquigarrow j), v_i, v_j))\right)}{\sum_{h=1}^{n} exp\left(f_i(\beta, x(i \rightsquigarrow h), v_i, v_j)\right)}$$

- The parameters $\beta_1, \ldots, \beta_k$ are constant over actors and time

SAOMs

Parameter interpretation

Actor-oriented perspective:

the evaluation function can be regarded as the "attractiveness" of a network

Let:

x the current state of the network
x^+ the network x with $x_{ij} = 1$
x^- the network x with $x_{ij} = 0$

then the difference in the utility is

$$u(\beta, x^+) - u(\beta, x^-) = \sum_k \beta_k (s_{ik}(x^+) - s_{ik}(x^-))$$

- $\beta_k > 0$: $s_{ik}(x)$ is positively evaluated
- $\beta_k < 0$: $s_{ik}(x)$ is negatively evaluated
- $\beta_k = 0$: $s_{ik}(x)$ is not important

Creating and terminating ties

Given $x(t_0)$ and $x(t_1)$ four possible tie changes are possible:

$x(t_0)$	$x(t_1)$	
$i \qquad j$	$i \longrightarrow j$	creation of a tie
$i \longrightarrow j$	$i \longrightarrow j$	maintenance of a tie
$i \longrightarrow j$	$i \qquad j$	termination of a tie
$i \qquad j$	$i \qquad j$	maintenance of a “no-tie”

Creating and terminating ties

Given $x(t_0)$ and $x(t_1)$ four possible tie changes are possible:

$x(t_0)$	$x(t_1)$	
$i \quad j$	$i \rightarrow j$	creation of a tie
$i \rightarrow j$	$i \rightarrow j$	maintenance of a tie
$i \rightarrow j$	$i \quad j$	termination of a tie
$i \quad j$	$i \quad j$	maintenance of a “no-tie”

The evaluation function models the presence of ties regardless they were created or maintained...

but maintaining (terminating) a tie is not always the opposite of creating a tie

Creating and terminating ties

To account for the creation and the termination of ties a more complex utility function is needed

Next to the evaluation function

1. the *creation function* $c_i(\delta, x')$

   and

2. the *endowment function* $e_i(\eta, x')$

are included in the utility function

$$u_i(x') = f_i(\beta, x') + \underbrace{c_i(\delta, x')}_{=0 \text{ tie termination}} + \underbrace{e_i(\eta, x')}_{=0 \text{ tie creation}} + \epsilon_i(t, x', j)$$

where $x' = x(i \rightsquigarrow j)$

Creating ties

Creation function

Models the gain in satisfaction incurred when a network tie is created:

$$c_i(\delta, x') = \sum_a \delta_a s_{ia}(x')$$

where

- δ_a are parameters
- $s_{ia}(x')$ are the effects whose strength is different in creating and terminating ties

The utility function for an actor i when he creates a new tie is

$$u_i(x') = f_i(\beta, x') + c_i(\delta, x') + \epsilon_i(t, x', j)$$

Creating ties

Parameter interpretation

A positive (negative) δ_a implies that the creation of a tie increasing $s_{ia}(x)$ is more attractive, i.e. the tie is more (less) likely to be created (given that $\beta_k > 0$)

In fact the difference in the utility functions is

$$
\begin{aligned}
u_i(x^+) - u_i(x^-) &= (f_i(\beta, x^+) + c_i(\delta, x^+)) + (f_i(\beta, x^-) + c_i(\delta, x^-)) \\
&= \sum_k \beta_k (s_{ik}(x^+) - s_{ik}(x^-)) + \sum_a \delta_a (s_{ia}(x^+) - s_{ia}(x^-))
\end{aligned}
$$

Terminating a tie

Endowment function

Models the loss in satisfaction incurred when a network tie is deleted

$$e_i(\eta, x') = \sum_b \eta_b s_{ib}(x')$$

where

- η_b are parameters
- $s_{ib}(x')$ are the effects whose strength is different in creating and terminating ties

The utility function for an actor i when he deletes a tie is

$$u_i(x') = f_i(\beta, x') + e_i(\eta, x') + \epsilon_i(t, x', j)$$

Terminating a tie

Parameter interpretation

A positive (negative) η_b implies that the maintenance of a tie is more attractive, i.e. the tie is less (more) likely to be terminated (given that $\beta_k > 0$)

In fact the contribution in the utility functions

$$\begin{aligned}
u_i(x^+) - u_i(x^-) &= exp((f_i(\beta, x^+) + e_i(\eta, x^+)) + (f_i(\beta, x^-) + e_i(\eta, x^-))) \\
&= \sum_k \beta_k (s_{ik}(x^+) - s_{ik}(x^-)) + \sum_b \eta_a (s_{ib}(x^+) - s_{ib}(x^-))
\end{aligned}$$

Creating and terminating ties

Remarks

- Using only the evaluation effect assumes that the effect has the same impact in both tie creation and tie termination

  a model with only evaluation effects leads to the same network dynamics as a specification where these effects are turned into creation and endowment effects, with the same parameters

- An effect can appear as components of one or two of these functions in a single model, but never in all three

- Practical point of view:
  - start modeling with evaluation effects
  - specify the endowment and the creation function given
    a clear idea about the available data and
    how tie creation and endowment may be different
    in the analysed data set

Creating and terminating ties

R code

The list of all effects available for a certain data set is provided by

```
effectsDocumentation(effects = myeff)
```

row	name	effectName	shortName	type	inter1	inter2	parm	interactionType
1	friendship	constant friendship rate (period 1)	Rate	rate			0	
2	friendship	constant friendship rate (period 2)	Rate	rate			0	
3	friendship	constant friendship rate (period 3)	Rate	rate			0	
4	friendship	outdegree effect on rate friendship	outRate	rate			0	
5	friendship	indegree effect on rate friendship	inRate	rate			0	
6	friendship	reciprocity effect on rate friendship	recipRate	rate			0	
7	friendship	effect 1/outdegree on rate friendship	outRateInv	rate			0	
8	friendship	effect gender on rate	RateX	rate	gender		0	
9	friendship	effect delinquency on rate	RateX	rate	delinquency		0	
10	friendship	outdegree (density)	density	eval			0	dyadic
11	friendship	outdegree (density)	density	endow			0	dyadic
12	friendship	outdegree (density)	density	creation			0	dyadic
13	friendship	reciprocity	recip	eval			0	dyadic
14	friendship	reciprocity	recip	endow			0	dyadic
15	friendship	reciprocity	recip	creation			0	dyadic
16	friendship	transitive triplets	transTrip	eval			0	
17	friendship	transitive triplets	transTrip	endow			0	
18	friendship	transitive triplets	transTrip	creation			0	

Creating and terminating ties

R code

Effects for the creation and the endowment function are specified using the argument type

- 'rate' = rate function
- 'eval' = evaluation function (default)
- 'creation' = creation function
- 'endow' = endowment function

Example

```
myeff <- includeEffects(myeff,recip,type='endow')\\
myeff <- includeEffects(myeff,transTrip,type='creation')
```

While the reciprocity effect specifies the endowment function, the transitive triplets effect specifies the creation function

Outline

Non-directed relations

For directed relation we assumed that:

1. an actor gets the opportunity to make a change
2. he decided for the change that assures him the highest payoff

Are these assumptions still reliable when we consider undirected relations such as: collaboration, trade, strategic alliance?

Non-directed relations

For directed relation we assumed that:

1. an actor gets the opportunity to make a change
2. he decided for the change that assures him the highest payoff

Are these assumptions still reliable when we consider undirected relations such as: collaboration, trade, strategic alliance?

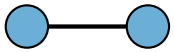
Yes AND No!!!

Non-directed relations

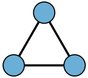
Extending the SAOM

Notation

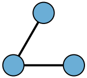
- x is the current state of the network
  Since relations are non-directed $x_{ij} = x_{ji}$
  Therefore, from now on, x_{ij} denotes the tie between i and j
  (and not the tie from i to j!!!)
- $x^{(+ij)}$ denotes the network where the tie between i and j is present
- $x^{(-ij)}$ denotes the network where the tie between i and j is absent
- x' denotes the next state of the network according to the evolution process
- The evaluation function is defined as: $f_i(x, \beta) = \sum_k \beta_k s_{ik}(x)(x)$
  where $s_{ik}(x)$ are the statistics for a non-directed network



edges



triangles



2-stars

For semplicity we always write $f_i(x)$ instead of $f_i(x, \beta)$

Non-directed relations

Extending the SAOM

Some preliminary remarks:

- necessity of making reasonable assumptions
  about the negotiation or coordination of the actors
  involved in the maintenance, creation or termination of a tie
- Several SAOMs can be defined
  (i.e. there is not only a single formulation,
  and several cases must be considered!)
- The distinction among the SAOMs concernes both
  the change opportunity process (i.e. the *rate function*)
  and the change determination process (i.e. the *evaluation function*)

Non-directed relations

Extending the SAOM: assumptions

Assumptions that are maintained:

- continuos-time
  while the observation schedule is in discrete time,
  the underlying evolution process takes place in continuous time
- Markov assumption
  The future configuration of the network depends only on
  the current configuration
- At each point in time only one tie can change
  Given x the next state of the network x' is
  either $x' = x^{(+ij)}$ or $x' = x^{(-ij)}$, shortly $x' = x^{(\pm ij)}$

The other assumptions depend on the change opportunity process and the change determination process

Non-directed relations

Extending the SAOM: assumptions

Two options are available for the change opportunity process:

1. One-sided initiative
   one actor *i* gets the opportunity to propose a change
2. Two-sided initiative
   a pair of actors (i,j) is selected and
   gets the opportunity to change the tie between them

Non-directed relations

Extending the SAOM: assumptions

Two options are available for the change opportunity process:

1. One-sided initiative
   one actor *i* gets the opportunity to propose a change
2. Two-sided initiative
   a pair of actors (i,j) is selected and
   gets the opportunity to change the tie between them

Three options are available for the change determination process:

a. Dictatorial choice
   one actor imposes a decision

b. Mutual choice
   one actor suggests a change and the other has to agree

c. Compensatory choice
   actors decide on the base of their combined interests

Non-directed relations

Extending the SAOM: one-sided initiative

The change opportunity process follows the same formulation of the SAOMs for directed ties

(Recall)
The waiting time between opportunities of change for an actor i is exponentially distributed with parameter $\lambda_i(\alpha, x, v)$

- all actors have the same rate of change λ

$$P(i \textit{ has the opportunity of change}) = \frac{\lambda}{\lambda n} = \frac{1}{n} \quad \forall i \in \mathcal{N}$$

- actors may change their ties at different frequencies $\lambda_i(\alpha, x, v)$

$$P(i \textit{ has the opportunity of change}) = \frac{\lambda_i(\alpha, x, v)}{\sum_{j=1}^{n} \lambda_j(\alpha, x, v)}$$

Non-directed relations

Extending the SAOM: one-sided initiative

Given the change opportunity process we can considered the change determination process.

Two options are available:

a. Dictatorial choice:
   i chooses his action and imposes his decision to j

$$\Downarrow$$

   The formulation of the model is equal to that of the SAOM for directed ties

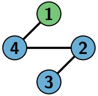
b. Mutual agreement:
   i suggests a tie and j has to agree

c. Compensatory:
   actors decide on the base of their combined interests
   This is quite artificial and not considered!

Non-directed relations

Extending the SAOM: one-sided initiative and mutual choice

E.g. actor 1 gets the opportunity to change



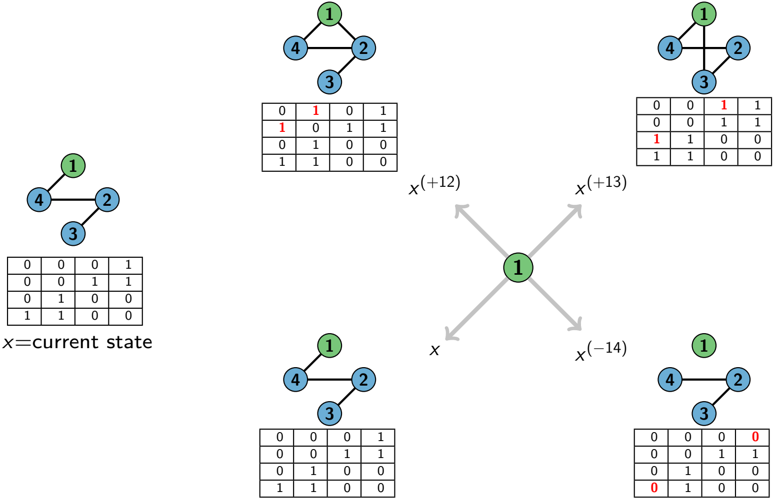
0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state

Non-directed relations

Extending the SAOM: one-sided initiative and mutual choice

E.g. actor 1 evaluates the alternatives



0	1	0	1
1	0	1	1
0	1	0	0
1	1	0	0

0	0	1	1
0	0	1	1
1	1	0	0
1	1	0	0

0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state

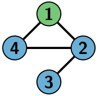
0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

0	0	0	0
0	0	1	1
0	1	0	0
0	1	0	0

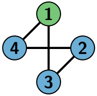
Non-directed relations

Extending the SAOM: one-sided initiative and mutual choice

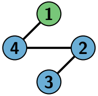
E.g. the best choice of actor 1 is to delete the tie between himself and 4



0	**1**	0	1
1	0	1	1
0	1	0	0
1	1	0	0

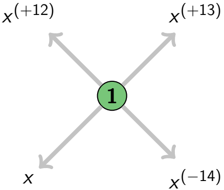


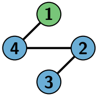
0	0	**1**	1
0	0	1	1
1	1	0	0
1	1	0	0



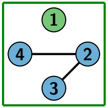
0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state





0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0



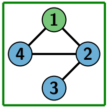
0	0	0	**0**
0	0	1	1
0	1	0	0
0	1	0	0

The tie is terminated!!!

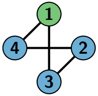
Non-directed relations

Extending the SAOM: one-sided initiative and mutual choice

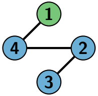
E.g. actor 1 suggests to actor 2 to create the tie between them



0	1	0	1
1	0	1	1
0	1	0	0
1	1	0	0

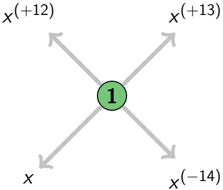


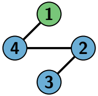
0	0	1	1
0	0	1	1
1	1	0	0
1	1	0	0



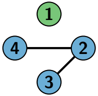
0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state





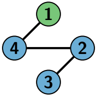
0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0



0	0	0	0
0	0	1	1
0	1	0	0
0	1	0	0

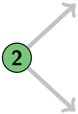
Non-directed relations

E.g. actor 2 evaluates the proposal of actor 1

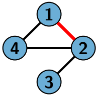


0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state

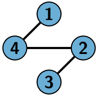


$x^{(+12)}$



0	**1**	0	1
1	0	1	1
0	1	0	0
1	1	0	0

$x^{(-12)}$



0	**0**	0	1
0	0	1	1
0	1	0	0
1	1	0	0

Non-directed relations

Extending the SAOM: one-sided initiative and mutual choice

- Actor i is selected and has the opportunity to make a change
- Actor i selects the best possible choice with probabilities

$$p_{i(\pm ij)} = \frac{exp(f_i(x^{(\pm ij)}))}{\sum_h exp(f_i(x^{(\pm ih)}))}$$

- If the best choice for i is to terminate or do not create x_{ij}, the proposal is put into effect, i.e. $x' = x^{(-ij)}$
- If the best choice for i is to create or maintain x_{ij} i.e. $x'_{ij} = x_{ij}^{(+ij)}$, this is proposed to j who accepts with probability

$$p_{j(+ij)} = \frac{exp(f_j(x^{(+ij)}))}{exp(f_j(x^{(-ij)})) + exp(f_j(x^{(+x_{ij})}))}$$

From now on, $p_{i(\cdot)}$ denotes the probability that i chooses $(\cdot)$

Non-directed relations

Extending the SAOM: one-sided initiative and mutual choice

Jointly these rules lead to the following transition probability:

- $$p_{x'} = \frac{exp(f_i(x^{(\pm ij)}))}{\sum_h exp(f_i(x^{(\pm ih)}))}$$

  when $x' = x^{(-ij)}$

- $$p_{x'} = \frac{exp(f_i(x'))}{\sum_h exp(f_i(x^{(\pm ih)}))} \left(\frac{exp(f_j(x^{(+ij)}))}{exp(f_j(x)) + exp(f_j(x^{(+x_{ij})}))} \right)$$

  when $x' = x^{(+ij)}$

Non-directed relations

Extending the SAOM: assumptions

Two options are available for the change opportunity process:

1. One-sided initiative
   one actor i gets the opportunity to propose a change
2. Two-sided initiative
   a pair of actors (i,j) is selected and
   gets the opportunity to change the tie between them

Three options are available for the change determination process:

a. Dictatorial choice
   one actor imposes a decision

b. Mutual choice
   one actor suggests a change and the other has to agree

c. Compensatory choice
   actors decide on the base of their combined interests

Non-directed relations

Extending the SAOM: two-sided initiative

The change opportunity process models the frequency at which
a couple (i,j) gets the opportunity to change the tie between them

The waiting time between opportunities of change for a couple (i,j) is exponentially distributed with parameter $\lambda_{ij}(\alpha, x, v)$

- all the couples have the same rate of change λ

$$P((i,j)\ has\ the\ opportunity\ of\ change) = \frac{2\lambda}{\lambda n(n-1)} = \frac{2}{n(n-1)} \quad \forall i,j \in \mathcal{N}$$

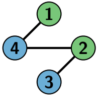
- couples may change at different frequencies $\lambda_{ij}(\alpha, x, v)$

$$P((i,j)\ has\ the\ opportunity\ of\ change) = \frac{\lambda_{ij}(\alpha, x, v)}{\sum_{i,j=1}^{n} \lambda_{ij}(\alpha, x, v)}$$

Non-directed relations

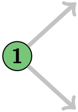
Extending the SAOM: two-sided initiative and dictatorial choice

E.g.
The couple (1,2) is selected and actor 1 imposed his decision on 2

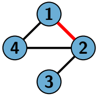


0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state

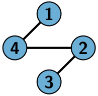


$x^{(+12)}$



0	**1**	0	1
1	0	1	1
0	1	0	0
1	1	0	0

$x^{(-12)}$



0	**0**	0	1
0	0	1	1
0	1	0	0
1	1	0	0

Non-directed relations

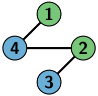
Extending the SAOM: two-sided initiative and dictatorial choice

- Actor i and j are selected and have the opportunity to change the tie between them
- Actor i imposes the decision about the existence of the tie x_{ij} on j

$$p_{i(\pm ij)} = \frac{exp(f_i(x^{\pm ij}))}{exp(f_i(x^{+ij})) + exp(f_i(x^{-ij}))} = p_{x'}$$

Non-directed relations

Extending the SAOM: two-sided initiative and mutual choice

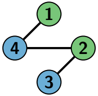


0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state

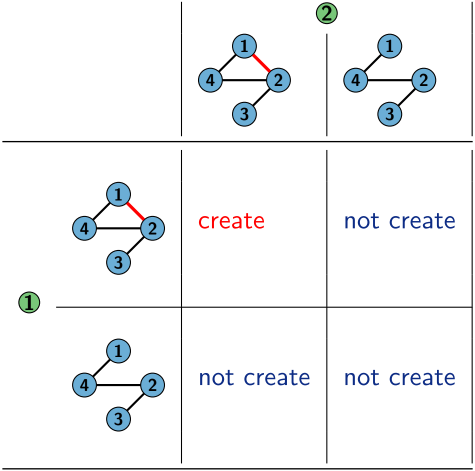
Non-directed relations

Extending the SAOM: two-sided initiative and mutual choice



0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state



Non-directed relations

Extending the SAOM: two-sided initiative and mutual choice

- Actor i and j are selected and have the opportunity to change the tie between them
- Actor i proposes his choice with probability

$$p_{i(\pm ij)} = \frac{exp(f_i(x^{(\pm ij)}))}{exp(f_i(x^{(+ij)})) + exp(f_i(x^{(-ij)}))}$$

- Actor j proposes his choice with probability

$$p_{j(\pm ij)} = \frac{exp(f_j(x^{\pm ij}))}{exp(f_j(x^{+ij})) + exp(f_j(x^{-ij}))}$$

Non-directed relations

Extending the SAOM: two-sided initiative and mutual choice

Jointly these rules lead to the following transition probability:

- $x' = x^{(+ij)}$

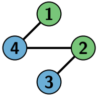
$$p_{x'} = \frac{exp(f_i(x^{(+ij)}))}{exp(f_i(x^{(+ij)})) + exp(f_i(x^{(-ij)}))} \frac{exp(f_j(x^{+ij}))}{exp(f_j(x^{+ij})) + exp(f_j(x^{-ij}))}$$

- $x' = x^{(-ij)}$

$$p_{x'} = 1 - \frac{exp(f_i(x^{(+ij)}))}{exp(f_i(x^{(+ij)})) + exp(f_i(x^{(-ij)}))} \frac{exp(f_j(x^{+ij}))}{exp(f_j(x^{+ij})) + exp(f_j(x^{-ij}))}$$

Non-directed relations

Extending the SAOM: two-sided initiative and compensatory choice

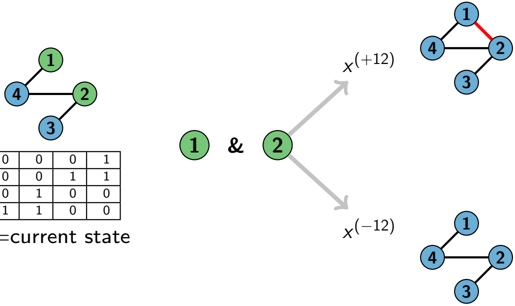


0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state

Non-directed relations

Extending the SAOM: two-sided initiative and compensatory choice



0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state

$x^{(+12)}$:

0	1	0	1
1	0	1	1
0	1	0	0
1	1	0	0

$x^{(-12)}$:

0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

Non-directed relations

Extending the SAOM: two-sided initiative and compensatory choice

- Actor i and j are selected and have the opportunity to change the tie between them
- Actor i and j choose their action with probability

$$p_{ij(\pm ij)} = \frac{exp(f_i(x^{(\pm ij)}) + f_j(x^{(\pm ij)}))}{exp(f_i(x^{(+ij)}) + f_j(x^{(+ij)})) + exp(f_i(x^{(-ij)}) + f_j(x^{(-ij)}))} = p_{x'}$$

where $p_{ij(\cdot)}$ denotes the probability that i and j choose $(\cdot)$

Non-directed relations

Stochastic tie-oriented model

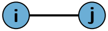
The focus is entirely on dyads:

- two-side opportunity process

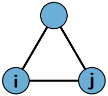
- the utility function is computed with respect to the couple

$$f_{(i,j)}(\beta, x) = \sum_k \beta_k s_{(i,j)k}(x))$$

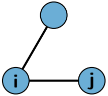
where $s_{(i,j)k}(x)$ is the statistic computed from the point of view of both i and j (or equivalently from the point of view of the tie x_{ij}!)



edges



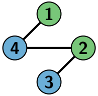
triangles



2-stars

Non-directed relations

Stochastic tie-oriented model

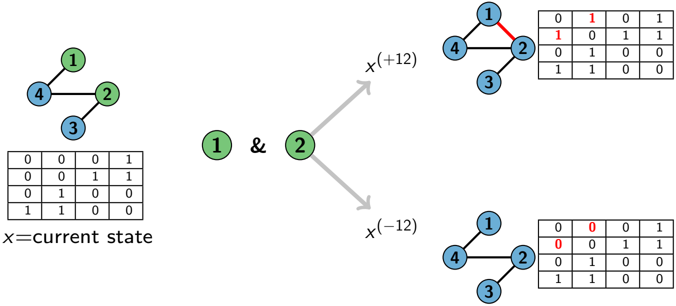


0	0	0	1
0	0	1	1
0	1	0	0
1	1	0	0

x=current state

Non-directed relations

Stochastic tie-oriented model



$$x = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

x=current state

$$x^{(+12)} = \begin{bmatrix} 0 & \mathbf{1} & 0 & 1 \\ \mathbf{1} & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

$$x^{(-12)} = \begin{bmatrix} 0 & \mathbf{0} & 0 & 1 \\ \mathbf{0} & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

Non-directed relations

Stochastic tie-oriented model

- Actor i and j are selected and have the opportunity to change the tie between them
- Actor i and j choose their action with probability

$$p_{ij(\pm ij)} = \frac{exp(f_{ij}(x^{(\pm ij)}))}{exp(f_{ij}(x^{(+ij)})) + exp(f_{ij}(x^{(-ij)})} = p_{x'}$$

Outline

ERGMs

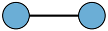
Recall

ERGMs are models for cross-sectional data:

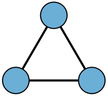
they return the probability of an observed graph (network) $G \in \mathcal{G}$
as a function of statistics $s_i(G)$ and statistical parameters θ_i

$$P_\theta(G) = \frac{1}{\kappa(\theta)} exp\left(\sum_{i=1}^{k} \theta_i \cdot s_i(G)\right)$$

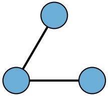
Examples of statistics $s_i(G)$ are:



edges



triangles



2-stars

ERGMs

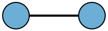
Recall

ERGMs are models for cross-sectional data:

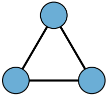
they return the probability of an observed graph (network) $G \in \mathcal{G}$
as a function of statistics $s_i(G)$ and statistical parameters θ_i

$$P_\theta(G) = \frac{1}{\kappa(\theta)} exp\left(\sum_{i=1}^{k} \theta_i \cdot s_i(G)\right)$$

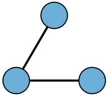
Examples of statistics $s_i(G)$ are:



edges



triangles



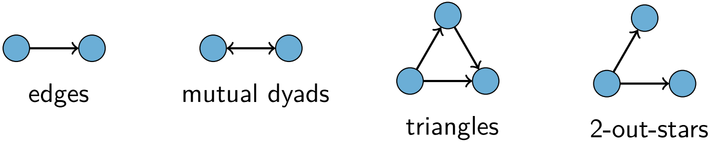
2-stars

ERGMs

Recall

ERGMs are also defined for directed graphs:
the mathematical formulation is the same but the effects take into account the direction of ties

Examples of statistics $s_i(G)$ are:



SAOMs

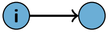
Recall

SAOMs are models for longitudinal data:
they explain the evolution of the network over time, assuming that network changes happen according to a continuous-time Markov chain modeled by:

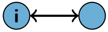
- the rate function λ
- the evaluation function

$$f_i(\beta, x(i \rightsquigarrow j), v_i, v_j) = \sum_{k=1}^{K} \beta_k s_{ik}(x(i \rightsquigarrow j))$$

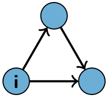
where examples of the statistics $s_{ik}(x(i \rightsquigarrow j))$ are:



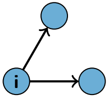
edges



mutual dyads



triangles



2-out-stars

SAOMs

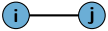
Recall

SAOMs can be also defined for non-directed ties:

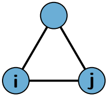
- according to the assumptions related to the change opportunity process and the change determination different models can be define
- the evaluation function is still computed as a linear combination of parameters and statistics from the point of view of either an actor *i* or a couple of actors (i,j)

$$f_{(\cdot)}(\beta, x(i \rightsquigarrow j), v_i, v_j) = \sum_{k=1}^{K} \beta_k s_{(\cdot)k}(x(i \rightsquigarrow j))$$

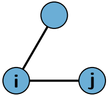
Examples of statistics $s_{(\cdot)k}(x(i \rightsquigarrow j))$ are:



edges



triangles



2-stars

SAOMs and ERGMs

Although ERGMs and SAOMs have different aims and require different data, the same statistics are used as explanatory variables in both models.

This might suggest the existence of a "statistical" relation between ERGMs and SAOMs

SAOMs and ERGMs

Although ERGMs and SAOMs have different aims and require different data, the same statistics are used as explanatory variables in both models.

This might suggest the existence of a "statistical" relation between ERGMs and SAOMs

We are going to prove that:

1. ERGMs are the limiting distribution of the process described by a certain specification of SAOMs when ties are directed
2. ERGMs are the limiting distribution of a particular formulation of the SAOMs when ties are undirected

Background: intensity matrix

Definition

Let $\{X(t), t \in \mathcal{T}\}$ be a continuous-time Markov chain whose transition probabilities are defined by:

$$P(X(t_j) = x' | X(t) = x(t), \forall t \leq t_i) = P(X(t_j) = x' | X(t_i) = x) \quad \forall x, x' \in \mathcal{S}$$

and holding time modelled by the rate λ
There exists a function $q : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ such that

$$\begin{cases} q(x, x') = \lim_{dt \to 0} \frac{P(X(t+dt)=x'|X(t)=x)}{dt} = \lambda P(X(t_j) = x' | X(t_i) = x) \\ \\ q(x, x) = \lim_{dt \to 0} \frac{P(X(t+dt)=x'|X(t)=x)-1}{dt} = \lambda P(X(t_j) = x | X(t_i) = x) \end{cases}$$

The function q is called **intensity matrix** of the process.

The element $q(x, \widetilde{x})$ is referred to as the rate at which the process in state x tends to change into $\widetilde{x}$

Background: limiting distribution

Definition

The **limiting distribution** P of a continuous-time Markov chain $\{X(t), t \in \mathcal{T}\}$ is defined as

$$P_{x'} = \lim_{t \to \infty} P(X(t_j) = x' | X(t_i) = x)$$

Therefore, the limiting distribution of $\{X(t), t \in \mathcal{T}\}$ is the distribution that describes the probability of jumping from x to x' in the long run behaviour of the process.

$P_{x'}$ is also the stationary distribution of the process

Irreducible aperiodic Markov chain and limiting distribution

Definition

A continuous-time Markov chain is **irreducible** if there is a path between any states x and x'

A continuous-time Markov chain is **aperiodic** if greatest common divisor of the length of all cycles equals one.

Irreducible aperiodic Markov chain and limiting distribution

Definition

A continuous-time Markov chain is **irreducible** if there is a path between any states x and x'

A continuous-time Markov chain is **aperiodic** if greatest common divisor of the length of all cycles equals one.

Theorem

If $\{X(t), t \in \mathcal{T}\}$ is an irreducible and aperiodic continuous-time Markov chain and the detailed balance condition holds

$$P_{x'} \cdot q(x', x) = P_x \cdot q(x, x')$$

then P_x is the unique limiting (stationary) distribution of $\{X(t), t \in \mathcal{T}\}$

ERGMs and SAOMs

Directed ties

Let us now consider a SAOM specified by the following functions:

- rate function

$$\lambda_i = \sum_{h=1}^{n} exp\left(\beta' s(x(i \rightsquigarrow h))\right)$$

  i.e., actors for whom changed relations have a higher value, will indeed change their relation more quickly.

- evaluation function

$$f_i(\beta, x(i \rightsquigarrow j)) = \sum_{i=1}^{K} \beta_k s_k(x(i \rightsquigarrow j) = \beta' s(x(i \rightsquigarrow j))$$

  i.e. actors take their decision considering the global configuration of the network

ERGMs and SAOMs

Directed ties

The rate and the objective functions define a continuous-time Markov chain on the set $\mathcal{X}$.

The associated intensity matrix q of the process is:

$$\begin{cases} q(x, x(i \rightsquigarrow j)) = \lambda_i p_{ij} = exp(\beta' s(x(i \rightsquigarrow j)) \\ \\ q(x, x) = \lambda_i p_{ij} = exp(\beta' s(x(i \rightsquigarrow i)) \end{cases}$$

ERGMs and SAOMs

Directed ties

The rate and the objective functions define a continuous-time Markov chain on the set $\mathcal{X}$.

The associated intensity matrix q of the process is:

$$\begin{cases} q(x, x(i \rightsquigarrow j)) = \lambda_i p_{ij} = exp(\beta' s(x(i \rightsquigarrow j)) \\ \\ q(x, x) = \lambda_i p_{ij} = exp(\beta' s(x(i \rightsquigarrow i)) \end{cases}$$

We can prove that ERGMs

$$P(X = x) = \frac{exp\left(\sum_{i=1}^{K} \beta_k s_k(x)\right)}{\kappa(\theta)} = \frac{exp(\beta' s(x))}{\kappa(\theta)}$$

are the unique stationary distribution of the SAOM defined before

Computing the limiting distribution

Directed ties

Proof

1. Existence of a unique invariant distribution

The continuous-time Markov chain described by the SAOM is:

- irriducible:
  each network configuration can be reached from any other network configuration in a finite number of steps
- aperiodic:
  at each time point t an actor i has the opportunity not to change anything and, thus, the period of each state is equal to 1

Computing the limiting distribution

Directed ties

Proof (continue)

2. ERGMs are the stationary distribution

Given two states x and $x(i \rightsquigarrow j)$ of $\{X(t), t \in \mathcal{T}\}$ the balance equation holds when ERGMs is the stationary distribution:

$$\begin{aligned}
P_{x(i \rightsquigarrow j)} \cdot q(x(i \rightsquigarrow j), x) &= \frac{exp(\beta' s(x(i \rightsquigarrow j))}{\kappa(\theta)} \cdot exp(\beta' s(x)) \\
&= \frac{exp(\beta' s(x))}{\kappa(\theta)} \cdot exp(\beta' s(x(i \rightsquigarrow j)) \\
&= P_x \cdot q(x, x(i \rightsquigarrow j))
\end{aligned}$$

Tie-based model

Unirected ties

We assume that

- each dyad (i,j) can be selected with the same rate λ
- the objective function is:

$$f_{(i,j)}(\beta,x) = \sum_k \beta_k s_{(i,j)k}(x))$$

  where $s_{(i,j)k}(x)$ is the statistic computed from the point of view of both i and j
- The transition probability is

$$p_{ij(\pm ij)} = \frac{exp(f_{ij}(x^{(\pm ij)}))}{exp(f_{ij}(x^{(+ij)})) + exp(f_{ij}(x^{(-ij)})}$$

Tie-based model

Unirected ties

The intensity matrix of the process is:

$$\begin{cases} q(x, x^{(+ij)}) = \lambda p_{ij(+ij)} = \lambda \frac{exp(f_{ij}(x^{(+ij)}))}{exp(f_{ij}(x^{(+ij)})) + exp(f_{ij}(x^{(-ij)})} \\ q(x, x^{(-ij)}) = \lambda p_{ij(-ij)} = \lambda \frac{exp(f_{ij}(x^{(-ij)}))}{exp(f_{ij}(x^{(+ij)})) + exp(f_{ij}(x^{(-ij)})} \end{cases}$$

The limiting distribution of such a model is again an ERGM

Proof

1. Existence of a unique invariant distribution

The continuous-time Markov chain defined by the tie based model is

- irriducible:
  each network configuration can be reached from any other network configuration in a finite number of steps
- aperiodic:
  at each time point t a pair (i, j) has the opportunity not to change anything and, thus, the period of each state is equal to 1

Computing the limiting distribution

Tie-based model

Proof (continue)

2. ERGMs are the stationary distribution

Given the two states x^{-ij} and x^{+ij} of $\{X(t), t \in \mathcal{T}\}$ the balance equation holds when ERGMs is the stationary distribution:

$$
\begin{aligned}
P_{x^{-ij}} q(x^{-ij}, x^{+ij}) &= \frac{e^{\beta' s(x^{-ij})}}{\kappa(\theta)} \cdot \lambda \cdot \frac{e^{\beta' s_{ij}(x^{+ij})}}{e^{\beta' s_{ij}(x^{+ij})} + e^{\beta' s_{ij}(x^{-ij})}} \\
&= \frac{e^{\beta' s(x^{-ij}) - \beta' s(x^{+ij}) + \beta' s(x^{+ij})}}{\kappa(\theta)} \cdot \frac{\lambda}{1 + e^{(\beta' s_{ij}(x^{-ij}) - \beta' s_{ij}(x^{+ij}))}} \\
&= \frac{e^{\beta' s(x^{+ij})}}{\kappa(\theta)} \cdot \lambda \cdot \frac{e^{\beta' s(x^{-ij}) - \beta' s(x^{+ij})}}{1 + e^{\beta' s_{ij}(x^{-ij}) - \beta' s_{ij}(x^{+ij})}} \\
&\underset{(*)}{=} \frac{e^{\beta' s(x^{+ij})}}{\kappa(\theta)} \cdot \lambda \cdot \frac{e^{\beta' s_{ij}(x^{-ij})}}{e^{\beta' s_{ij}(x^{+ij})} + e^{\beta' s_{ij}(x^{-ij})}} \\
&= P_{x^{+ij}} \cdot q(x^{+ij}, x^{-ij})
\end{aligned}
$$

$(*) \quad \beta' s(x^{-ij}) - \beta' s(x^{+ij}) = \beta' s_{ij}(x^{-ij}) - \beta' s_{ij}(x^{+ij})$

Outline

Goodness of fit

Evaluate the performance of the SAOM

Analysis of the network evolution:

1. Specification of the model:
   Which effects should be used to specify the rate and the evaluation function?
2. Estimation of the parameters of the model:
   using the software
3. Interpretation of the results:
   What can be concluded about the network evolution?

Fundamental question before "selling" our results are:
Is the specified model a "good" model? How well is it performing?

As for the ERGMs, we need to analyse the goodness of fit of the model!

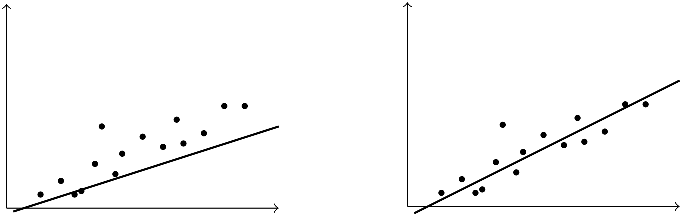
gof: goodness of fit

Goodness of fit

Evaluate the performance of the SAOM

When we consider a simple model, e.g. regression analysis, evaluating the gof is very simple:

1. compute the values of the dependent variables predicted by the model
2. compare the observed values with the predicted values



poor gof   good gof

This can be generalized also to model for longitudinal data ...
but what if the dependent variable is a series of networks?

Goodness of fit

Evaluate the performance of the SAOM

The main question is:
How to compare networks?

Heuristic gof:

1. simulate the series of M networks a large number of times
2. compute the distribution of a statistic that is not directly fitted by the model
   (e.g. the indegree distribution)
3. if the observed value of the statistic is not extreme in the distribution, then the statistic is well fitted by the model

The statistic that is not directly fitted by the model is called **auxiliary statistic**. We will denote it as s^{aux}.

Repeating this procedure for several auxiliary statistics provides information on the gof of the model

Goodness of fit

Evaluate the performance of the SAOM

We need a statistical test to decide if

$$H_0\text{: good gof}$$

should be rejected in favour of

$$H_1\text{: poor gof}$$

Logic of the test:

- we can compare the simulated values of the auxiliary statistics with the observed values
  (e.g. the simulated and the observed indegree distributions)
- if the values are similar our model has a good gof
- if the values are far away than the model has a poor gof

Goodness of fit

Evaluate the performance of the SAOM

Let

- $s^{aux} = (s_1^{aux}(x), \ldots,\ s_h^{aux}(x), \ldots,\ s_H^{aux}(x))$
  the vector of H auxiliary statistics
- $\bar{s}^{aux} = (\bar{s}_1^{aux}(x), \ldots,\ \bar{s}_h^{aux}(x), \ldots,\ \bar{s}_H^{aux}(x))$
  the Monte Carlo approximation of s^{aux}
- $s^{obs} = (s_1^{obs}(x), \ldots,\ s_h^{obs}(x), \ldots,\ s_H^{obs}(x))$
  the observed values of the auxiliary statistics

The test statistic is

$$D = \sqrt{\left(\bar{s}_h^{aux} - s_h^{obs}\right)' \left(\Sigma_{s^{aux}}\right)^{-1} \left(\bar{s}_h^{aux} - s_h^{obs}\right)}$$

where $\Sigma_{s^{aux}}$ is the covariace matrix of the auxiliary statistics

Goodness of fit

Evaluate the performance of the SAOM

The test statistic is

$$D = \sqrt{\left(\bar{s}_h^{aux} - s_h^{obs}\right)' \left(\Sigma_{s^{aux}}\right)^{-1} \left(\bar{s}_h^{aux} - s_h^{obs}\right)} \sim \chi_h^2$$

where $\Sigma_{s^{aux}}$ is the covariace matrix of the auxiliary statistics

Interpretation:

- higher values of D (p-values$<$0.05) provides evidence against H_0
- lower values of D (p-values$>$0.05) provides evidence to H_0

Goodness of fit

Evaluate the performance of the SAOM: an example

- s50 data:
  an excerpt of the data and part of "*Teenage Friends and Lyfestyle Study*" available at
  `http://www.stats.ox.ac.uk/~snijders/siena/`
- 3 observations of a cohort of pupils in a Scottish school
  over a 3 year period
- actors: 50 boys
- relation: friendship
- SAOM: edges, reciprocity, transitive triplets
- gof is evaluated with the `sienaGOF` function
  see the R script "gof.R" on the webapge of the course

Goodness of fit

Evaluate the performance of the SAOM: an example

For each auxiliary statistic the `sienaGOF` allows to analyse the gof of a SAOM using two instruments

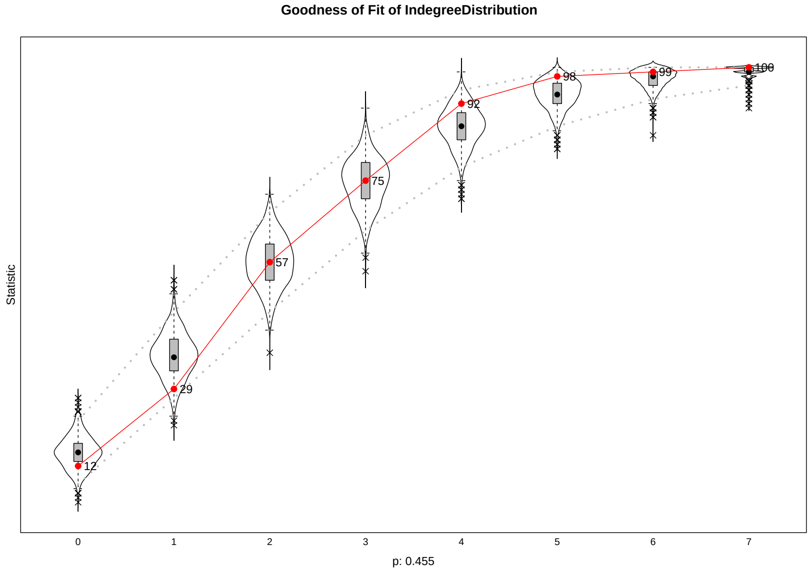
- statistical test

  based on Mahalanobis distance

- violin plots:
  box-plot+density plot

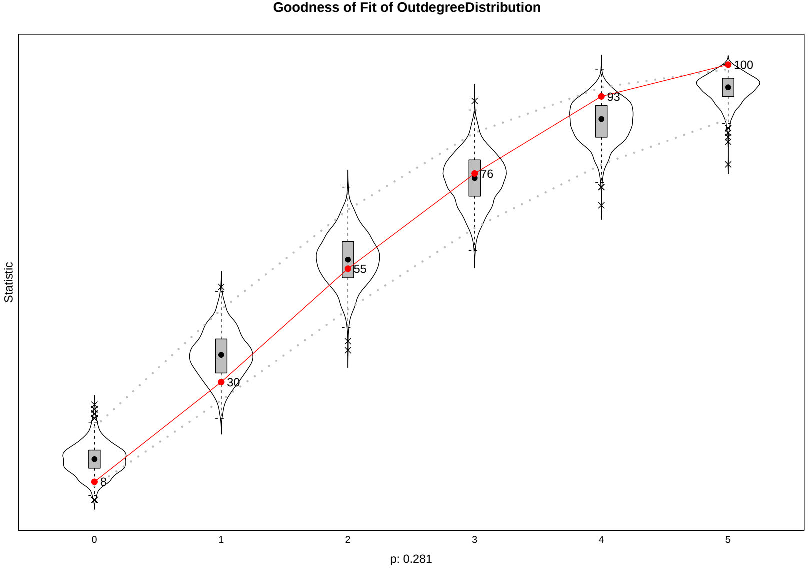
Goodness of fit

Evaluate the performance of the SAOM: an example



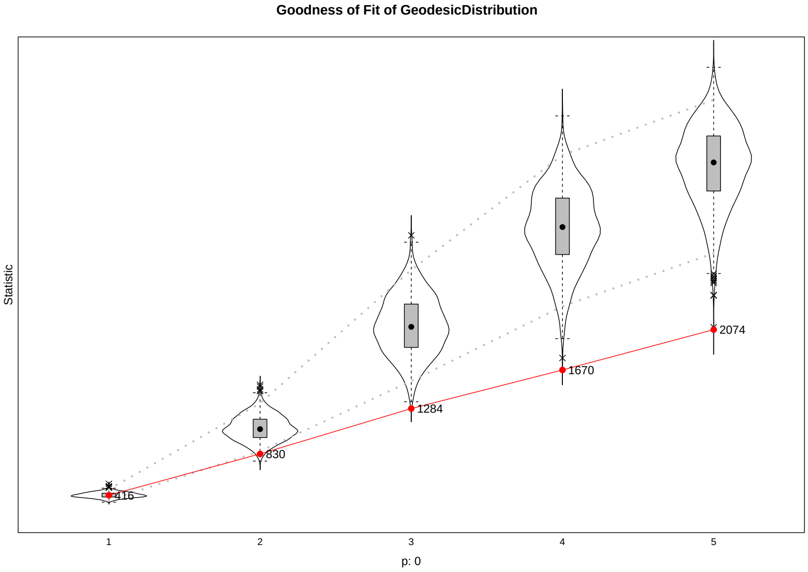
Goodness of fit

Evaluate the performance of the SAOM: an example



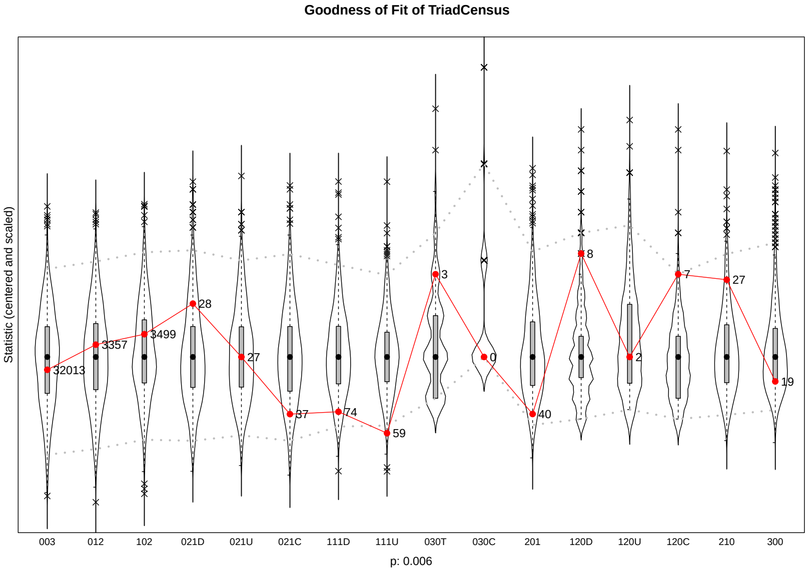
Goodness of fit

Evaluate the performance of the SAOM: an example



Goodness of fit

Evaluate the performance of the SAOM: an example



Goodness of fit

Evaluate the performance of the SAOM: an example

The previous graphs show that:

- good fit for the indegree and the outdegree distribution
- poor fit for the geodesic distance and the triadic census

Why do we get a poor fit?

1. Some assumptions of the SAOM are not valid
   (e.g. there is time heterogeneity)
2. The model is missspecified
   (i.e. not all the statistics explaining the network evolution are included)

Time heterogeneity

Are the parameters of the evaluation function constant over time?

Why do we usually neglect time heterogeneity?

- onerous and time consuming including more parameters when time heterogeneity is not part of the research question
- it is unknown under which circumstances omitting time heterogeneity leads to erroneous conclusions

Consequences of neglecting time heterogeneity in SAOMs:

- Estimates that average over heterogeneity
  but some statistics might not be relevant at the beginning
- Some statistics might turn to be not significant (when they are!)
  if a statistic plays a role only between two consecutive observations,
  it might turn not to be significant over the entire period
- poor gof
  estimates will not be able to reproduce the observed value of the statistics
  between the pair of observations

Time heterogeneity

How to detect it?

Utilities deriving from the choice of the actors are driven by the evaluation function

$$f_i(x,\beta) = \sum_k \beta_k s_{ik}(x) \tag{1}$$

but the rules regulating the choice may have changed over time. This suggests reformulating (1) to account for time heterogeneity

$$f_i(x,\beta) = \sum_k (\beta_k + \delta_k^{(m)} h_k^{(m)}) s_{ik}(x) \tag{2}$$

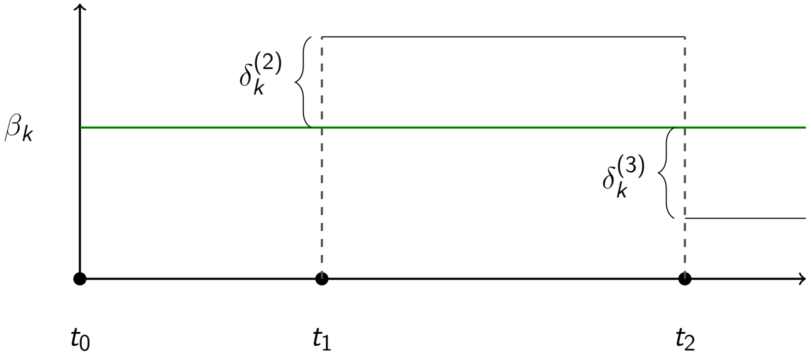
where $\delta_k^{(m)}$ are period-specific parameters and

$$h_k^{(m)} = \begin{cases} 1 & \textit{for period } [t_{m-1}, t_m] \\ 0 & \textit{otherwise} \end{cases}$$

time heterogeneity

How to detect it?

Intuitively



Example

β_{rec} is the average contribution of reciprocity

$\delta_{rec}^{(m)}$ added contribution of reciprocity between t_{m-1} and t_m

Time heterogeneity

Statistical test

Testing time heterogeneity corresponds to test

$$H_0 : \delta_k^{(m)} = 0 \quad \textit{for all } k, m$$

$$H_1 : \delta_k^{(m)} \neq 0 \quad \textit{for some } k, m$$

How can we test this?

1. Task 3, assignment 10 (see discussion in the tutorial)
2. Use simulation
   - estimate the model under H_0 so that we have an estimate $\widehat{\beta}_k$ for β_k
   - compute the differences

     $$E_{\widehat{\beta}_k}[S_{mk} - s_{mk}] \quad \forall m, k$$

   - If this differences are large, then $\widehat{\beta}_k$ is not a good estimate

Time heterogeneity

Statistical test

This is formally tested using the test statistic

$$B = g(E_{\widehat{\beta}_k}[S_{mk} - s_{mk}])'\Sigma_g^{-1} g(E_{\widehat{\beta}_k}[S_{mk} - s_{mk}]) \sim \chi_k^2$$

where

- $g : \mathbb{R} \to \mathbb{R}$ is a function
- Σ_g is a covariance matrix of $g(E_{\widehat{\beta}_k}[S_{mk} - s_{mk}])$

Interpretation:

- higher values of B (p-values < 0.05) provides evidence against H_0
- lower values of B (highp-value$>$0.05) provides evidence to H_0

Time heterogeneity

Statistical test

If H_0 is rejected, i.e. there is time heterogeneity

- a researcher can estimate different SAOMs based the observations of the network for which there is time-homogeneity

  drawback: we have several models

- we can specify a new evaluation function:

$$f_i(x, \beta) = \sum_k \beta_k s_{ik}(x) + \delta_k^{(m)} h_k^{(m)} s_{ik}(x)$$

  comprising of the time-dependent statistics $h_k^{(m)} s_{ik}(x)$ so that we can estimate $\delta_k^{(m)}$

  This results in one model with more parameters

Time heterogeneity

Example

Testing if the poor gof of the SAOM on the s50 data is due to time heterogeneity

This is done using the command `sienaTimeTest`
(see the R script gof.R)

```
Joint significance test of time heterogeneity:
chi-squared = 3.59, d.f.  = 3, p= 0.3091,
where H0:  The following parameters are zero:
(1) (*)Dummy2:outdegree (density)
(2) (*)Dummy2:reciprocity
(3) (*)Dummy2:transitive triplets
```

No effect of time heterogeneity

Model specification

How to specify SAOMs?

- theory should always guide model selection,
  but a data driven approach can also help!
- it is recommended to use a forward approach
  - start from a simple model
  - include more complex effect step-by-step

We follow this approach in order to improve the gof of the SAOM for the s50 data

Model specification

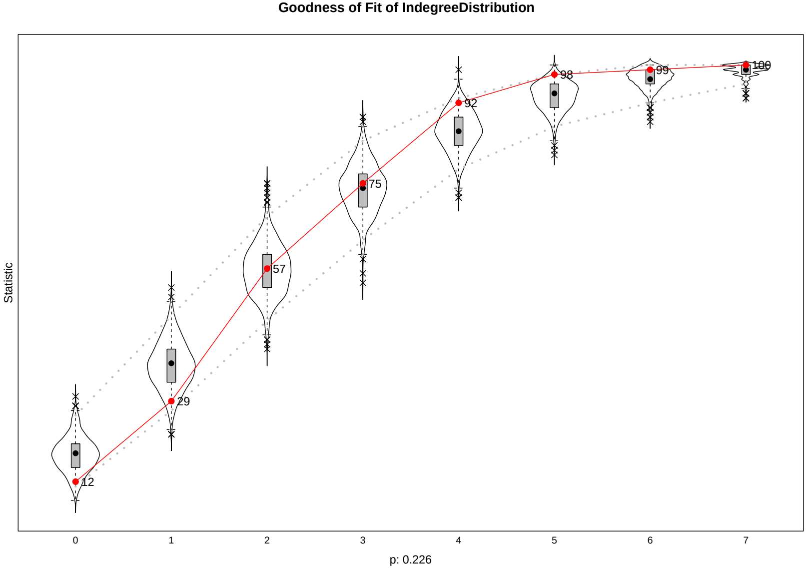
How to specify SAOMs?

- Theory guided approach
  - the tendency to transitive closure might depend less strongly on the number of indirect connections than represented by the transitive triplets effect. Good alternatives might be:
    - the transitive ties effect
    - the geometrically weighted edgewise shared partner effect
  - 3-cycle effect may be important as an inverse indication of local hierarchy
  - the interaction between reciprocity and transitivity may be important

We specify a model including the statistics corresponding to these effects apart from the geometrically weighted edgewise shared partner effect

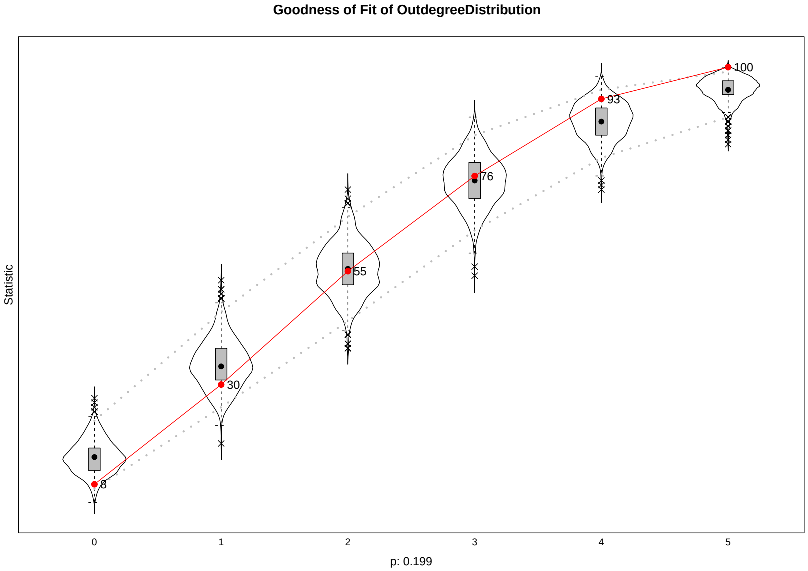
Model specification

How to specify SAOMs?



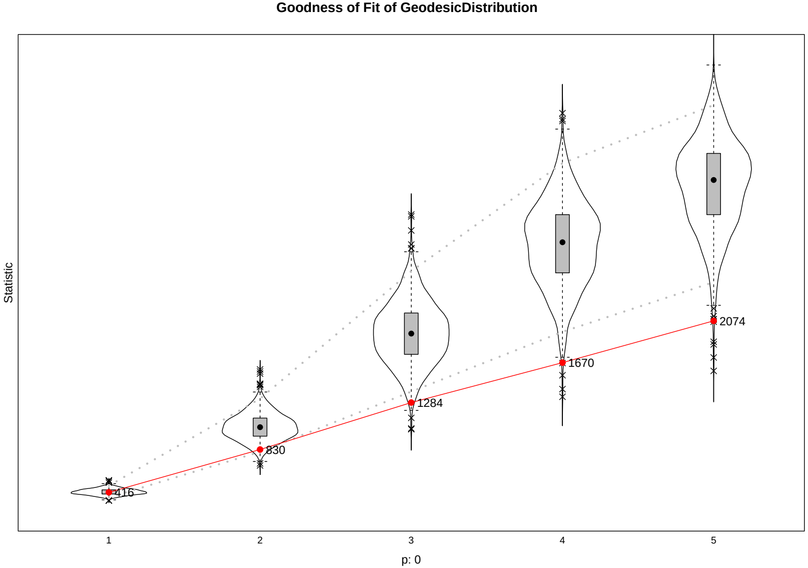
Model specification

How to specify SAOMs?



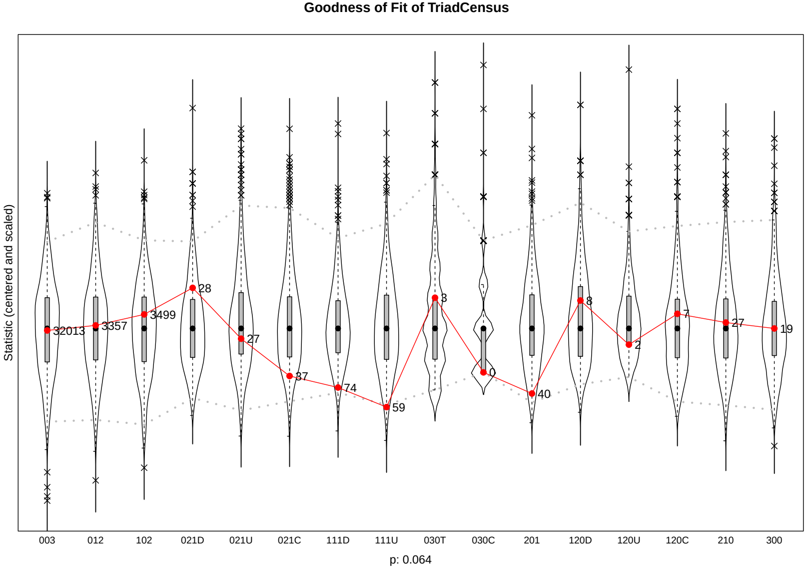
Model specification

How to specify SAOMs?



Model specification

How to specify SAOMs?



Model specification

How to specify SAOMs?

Testing if the poor gof of the new SAOM on the s50 data is due to time heterogeneity

This is done using the command `sienaTimeTest`
(see the R script gof.R)

```
Joint significance test of time heterogeneity:
chi-squared = 6.57, d.f.  = 6, p= 0.3627,
where H0:  The following parameters are zero:
(1) (*)Dummy2:outdegree (density)
(2) (*)Dummy2:reciprocity
(3) (*)Dummy2:transitive triplets
(4) (*)Dummy2:transitive reciprocated triplets
(5) (*)Dummy2:3-cycles
(6) (*)Dummy2:transitive ties
```

No effect of time heterogeneity

Model specification

How to specify SAOMs?

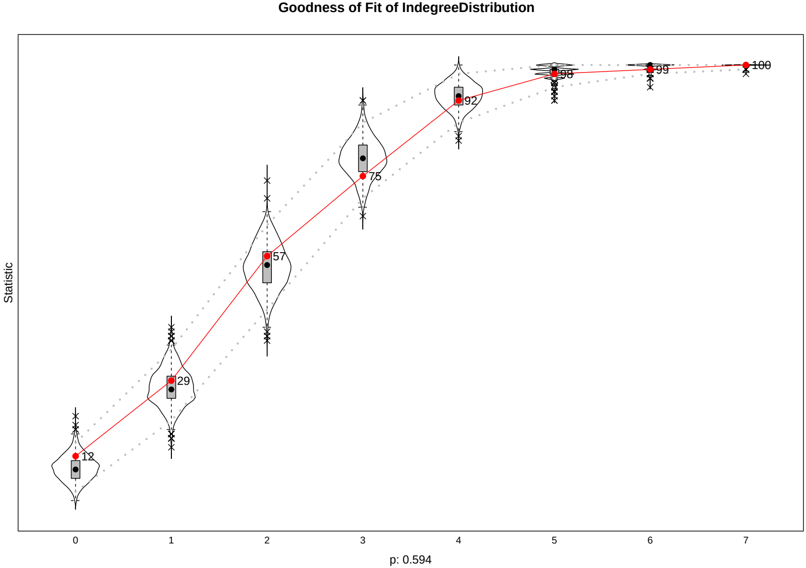
2. Data driven approach
    - We need also effects to improve the outdegree distribution e.g. outdegree activity and outdegree popularity

      (and these effects are also supported by theory...data driven approach could help us if we have forgotten something)

We include them in the previous model

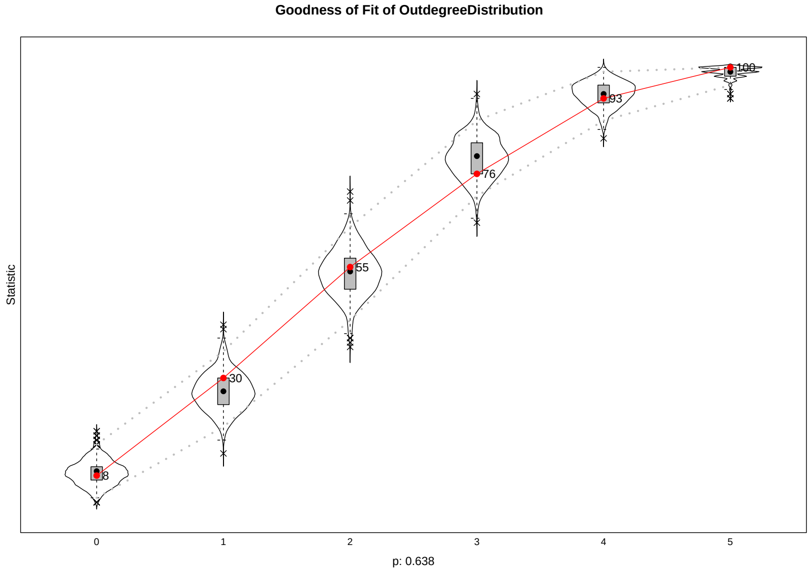
Model specification

How to specify SAOMs?



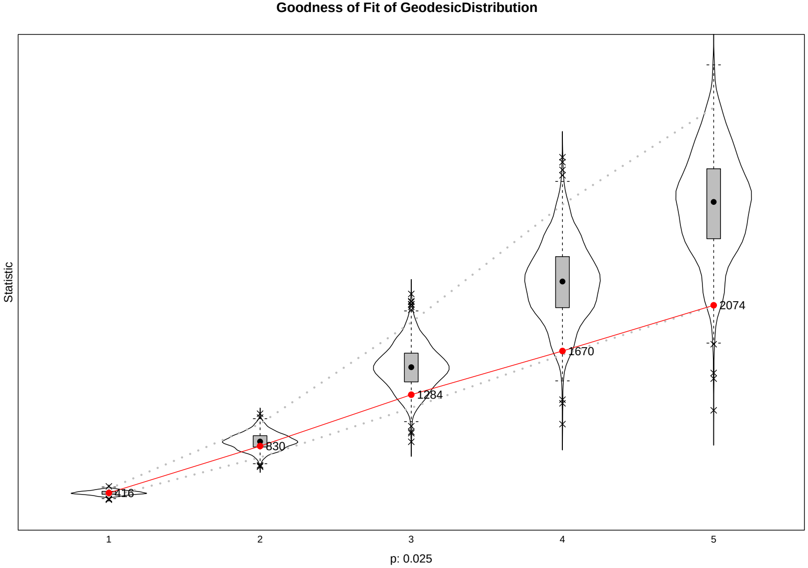
Model specification

How to specify SAOMs?



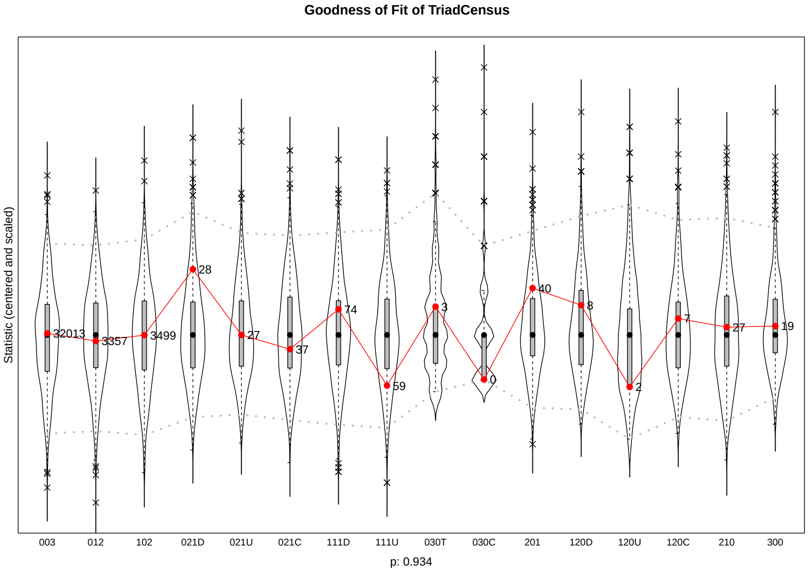
Model specification

How to specify SAOMs?



Model specification

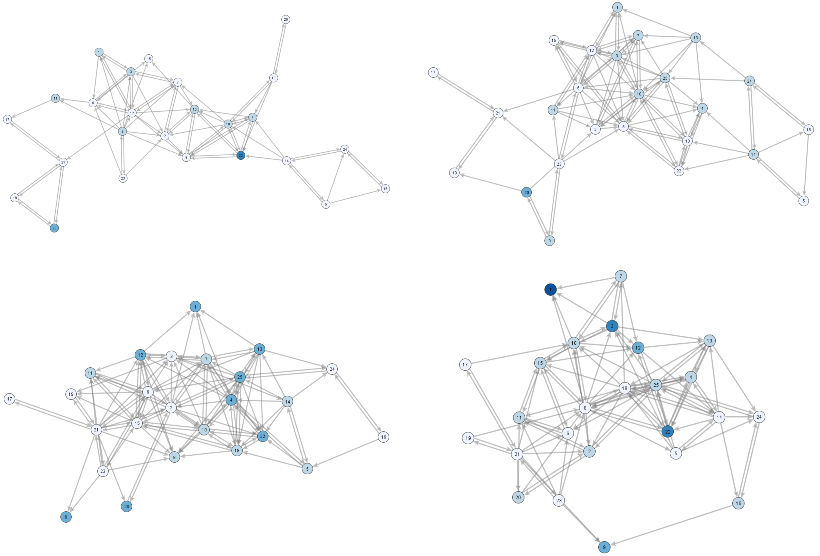
How to specify SAOMs?



Outline

Networks are dynamic by nature: a real example

A. Knecht (2008): “Friendship Selection and Friends’ Influence”



Four time points in the pupils’ first year at secondary school (color delinquency)

Motivation

1. Social network dynamics can depend on actors' characteristics

Selection process:
partners are selected according to their characteristics

Example

Homophily:
the formation of relations based on the similarity of two actors

E.g. delinquency behaviour

t_0 t_1

pupils with the same delinquent behaviour tend to become friends

Motivation

2. Changeable actors' characteristics can depend on the social network

E.g.: opinions, attitudes, intentions, etc.

Changeable actors' characteristics are called **behaviour**

Influence process:
actors adjust their characteristics according to the characteristics of other actors to whom they are tied

Example

Assimilation/contagion:
connected actors become increasingly similar over time

E.g. delinquency behaviour

pupils adjust they delinquent behaviour to that of their friends

Competing explanatory stories

Homophily and assimilation give rise to the same outcome
(similarity of connected individuals)

$$\Downarrow$$

study of influence requires the consideration of selection and vice versa

Fundamental question:
is the similarity of connected individuals caused mainly by influence or mainly by selection?

Competing explanatory stories

Homophily and assimilation give rise to the same outcome
(similarity of connected individuals)

$$\Downarrow$$

study of influence requires the consideration of selection and vice versa

Fundamental question:
is the similarity of connected individuals caused mainly by influence or mainly by selection?

Extending the SAOM to analise the
co-evolution of networks and behaviours

Competing explanatory stories

Example

Similarity in delinquency:

Selection:
a “delinquent” pupil may tend to have “delinquent” friends because of the balance theory

Competing explanatory stories

Example

Similarity in delinquency:

Selection:
a “delinquent” pupil may tend to have “delinquent” friends because of the balance theory

Influence:
the friendship with a “delinquent” pupil may have made an actor adopting a delinquent behaviour in the first place

Longitudinal network-behaviour panel data

1. a network x represented by its adjacency matrix
2. a series of actors' attributes:
   - H constant covariates $V_1, \ldots, V_H$
   - L behaviour covariates $Z_1, \ldots, Z_L$
     behaviour variables are ordinal categorical variables

Longitudinal network-behaviour panel data

1. a network x represented by its adjacency matrix
2. a series of actors' attributes:
   - H constant covariates $V_1, \ldots, V_H$
   - L behaviour covariates $Z_1, \ldots, Z_L$
     behaviour variables are ordinal categorical variables

Longitudinal network-behaviour panel data:
networks and behaviours observed at $M \geq 2$ time points $t_1, \cdots, t_M$

$$(x,z)(t_0),\ (x,z)(t_1), \cdots,\ (x,z)(t_M)$$

and the constant covariates $V_1, \ldots, V_H$

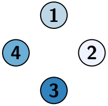
Assumptions

1. *Distribution of the process: continuous-time Markov chain*
   - *State space $\mathcal{C}$:* all the possible configurations arising from the combination of network and behaviours

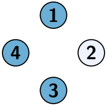
$$|C| = 2^{n(n-1)} \times B^n$$

where B is the number of categories for the behaviour variable

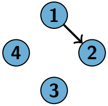
Example



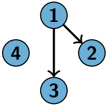
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0



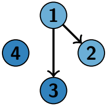
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0



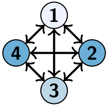
0	1	0	0
0	0	0	0
0	0	0	0
0	0	0	0



0	1	1	0
0	0	0	0
0	0	0	0
0	0	0	0



0	1	1	0
0	0	0	0
0	0	0	0
0	0	0	0



0	1	1	1
1	0	1	1
1	1	0	1
1	1	1	0

Assumptions

1. *Distribution of the process: continuous-time Markov chain*
   - *State space $\mathcal{C}$:* all the possible configurations arising from the combination of network and behaviours

$$|C| = 2^{n(n-1)} \times B^n$$

   where B is the number of categories for the behaviour variable.

   - *Markovian assumption:*
     changes actors make are assumed to depend only on the current state of the network

Assumptions

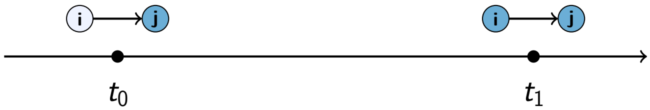
1. *Distribution of the process: continuous-time Markov chain*
   - *State space $\mathcal{C}$:* all the possible configurations arising from the combination of network and behaviours

$$|C| = 2^{n(n-1)} \times B^n$$

   where B is the number of categories for the behaviour variable.

   - *Markovian assumption:*
   changes actors make are assumed to depend only on the current state of the network

   - *Continuous-time:*

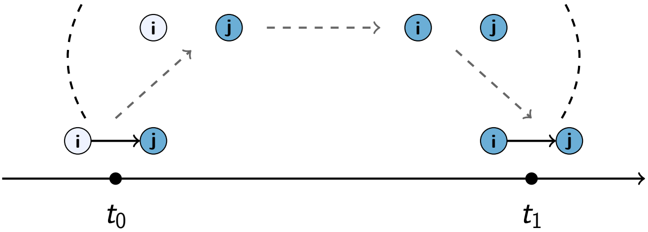


Assumptions

1. *Distribution of the process: continuous-time Markov chain*
   - *State space $\mathcal{C}$:* all the possible configurations arising from the combination of network and behaviours

$$|C| = 2^{n(n-1)} \times B^n$$

     where B is the number of categories for the behaviour variable.
   - *Markovian assumption:*
     changes actors make are assumed to depend only on the current state of the network
   - *Continuous-time:*



Assumptions

1. *Distribution of the process: continuous-time Markov chain*

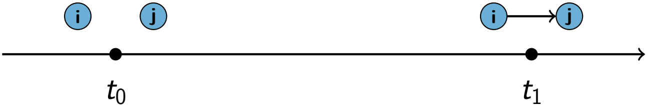
   - *State space $\mathcal{C}$:* all the possible configurations arising from the combination of network and behaviours

   $$|C| = 2^{n(n-1)} \times B^n$$

   where B is the number of categories for the behaviour variable.

   - *Markovian assumption:*
   changes actors make are assumed to depend only on the current state of the network

   - *Continuous-time:*



Assumptions

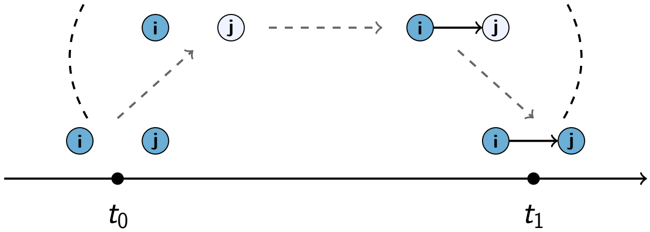
1. *Distribution of the process: continuous-time Markov chain*
   - *State space $\mathcal{C}$:* all the possible configurations arising from the combination of network and behaviours

$$|C| = 2^{n(n-1)} \times B^n$$

     where B is the number of categories for the behaviour variable.

   - *Markovian assumption:*
     changes actors make are assumed to depend only on the current state of the network

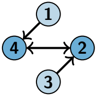
   - *Continuous-time:*



Assumptions

2. Opportunity to change

At any given moment one probabilistically selected actor has the opportunity to change one of his outgoing ties OR his behaviour



0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x

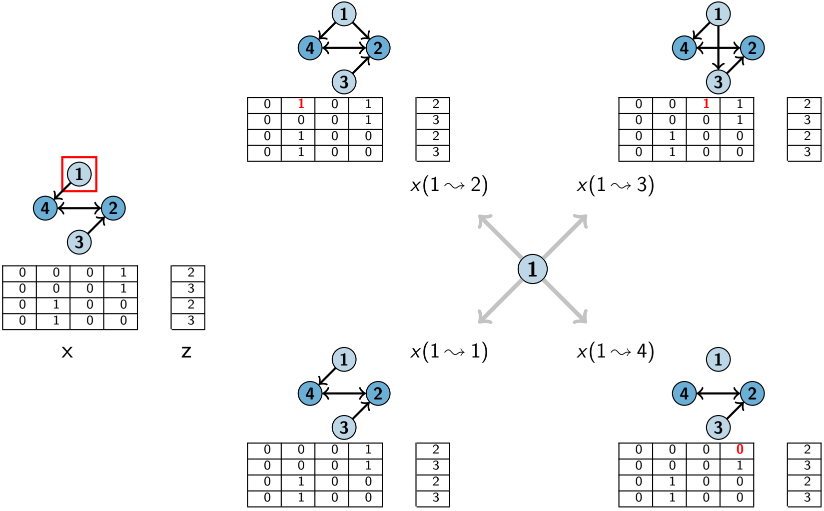
2
3
2
3

z

Assumptions

2. Opportunity to change

At any given moment one probabilistically selected actor has the opportunity to change one of his outgoing ties OR his behaviour



x:

0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

z:

2
3
2
3

$x(1 \rightsquigarrow 2)$:

0	**1**	0	1
0	0	0	1
0	1	0	0
0	1	0	0

2
3
2
3

$x(1 \rightsquigarrow 3)$:

0	0	**1**	1
0	0	0	1
0	1	0	0
0	1	0	0

2
3
2
3

$x(1 \rightsquigarrow 1)$:

0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

2
3
2
3

$x(1 \rightsquigarrow 4)$:

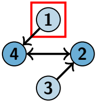
0	0	0	**0**
0	0	0	1
0	1	0	0
0	1	0	0

2
3
2
3

Assumptions

2. Opportunity to change

At any given moment one probabilistically selected actor has the opportunity to change one of his outgoing ties OR his behaviour



0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x

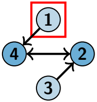
2
3
2
3

z

Assumptions

2. Opportunity to change

At any given moment one probabilistically selected actor has the opportunity to change one of his outgoing ties OR his behaviour

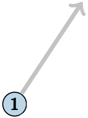


0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

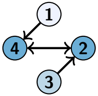
x

2
3
2
3

z



$z(2 \rightsquigarrow 1)$



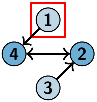
0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

1
3
2
3

Assumptions

2. Opportunity to change

At any given moment one probabilistically selected actor has the opportunity to change one of his outgoing ties OR his behaviour

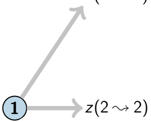


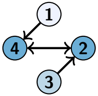
0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

x

2
3
2
3

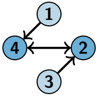
z





0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

1
3
2
3



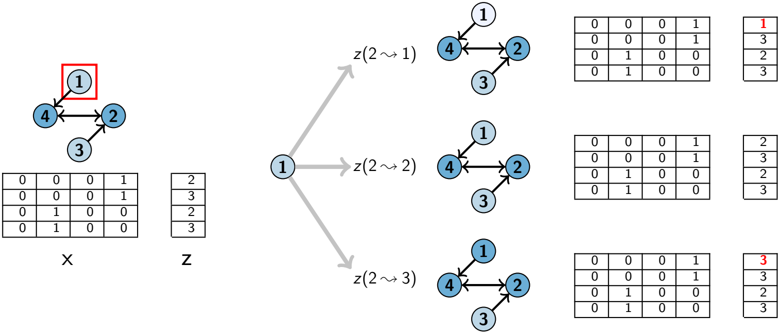
0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

2
3
2
3

Assumptions

2. Opportunity to change

At any given moment one probabilistically selected actor has the opportunity to change one of his outgoing ties OR his behaviour



x:

0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

z:

2
3
2
3

$z(2 \rightsquigarrow 1)$:

0	0	0	1		1
0	0	0	1		3
0	1	0	0		2
0	1	0	0		3

$z(2 \rightsquigarrow 2)$:

0	0	0	1		2
0	0	0	1		3
0	1	0	0		2
0	1	0	0		3

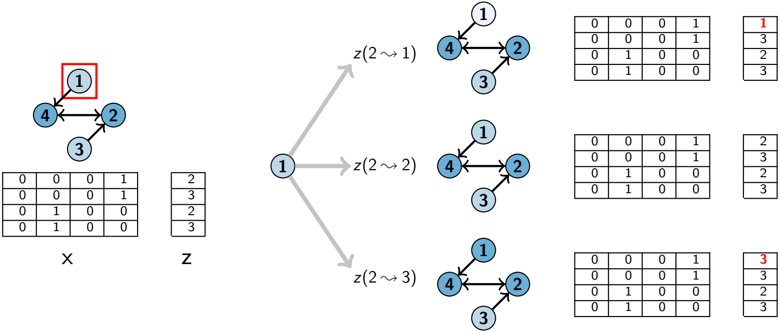
$z(2 \rightsquigarrow 3)$:

0	0	0	1		3
0	0	0	1		3
0	1	0	0		2
0	1	0	0		3

Assumptions

2. Opportunity to change

At any given moment one probabilistically selected actor has the opportunity to change one of his outgoing ties OR his behaviour



Notation:

$z(l \rightsquigarrow l+1)$ denotes the change in the behaviour L when an actor i increases the level of his behaviour by one unit

$z(l \rightsquigarrow l-1)$ denotes the change in the behaviour L when an actor i decreases the level of his behaviour by one unit

$z(l \rightsquigarrow l)$ denotes that an actor i does not change the level of the behaviour

Assumptions

3. *Absence of co-occurrence*
At each instant t, only one actor has the opportunity to change
(one of his outgoing ties or his behaviour)

Assumptions

3. *Absence of co-occurrence*

At each instant t, only one actor has the opportunity to change (one of his outgoing ties or his behaviour)

4. *Actor-oriented perspective*

Actors control their outgoing ties as well as their own behaviour.

- the actor decides to change one of his outgoing ties or his behaviour trying to maximize *a utility function*
- two distinct objective functions: one for the network and one for the behavioural change
- actors have complete knowledge about the network and the behaviours of all the the other actors
- the maximization is based on immediate returns (myopic actors)

Outline

Model definition

The co-evolution process is decomposed into a series of micro-steps:

- **network micro-step:**
  the opportunity of changing one network tie and
  the corresponding tie changed

- **behaviour micro-step:**
  the opportunity of changing a behaviour and
  the corresponding unit changed in behaviour

Model definition

There are two type of micro-steps:

- network micro-steps
- behavioural micro-steps

	Occurrence	*Preference*
Network changes	Network rate function	Network evaluation function
behavioural changes	Behavioural rate function	Behavioural evaluation function

N.b.
In the literature the evaluation function is also called objective function

The rate functions

The frequency by which actors have the opportunity to make a change is modelled by the *rate functions*, one for each type of change.

Why must we specify two different rate functions?

The rate functions

The frequency by which actors have the opportunity to make a change is modelled by the *rate functions*, one for each type of change.

Why must we specify two different rate functions?

Practically always, one type of decision will be made more frequently than the other

The rate functions

The frequency by which actors have the opportunity to make a change is modelled by the *rate functions*, one for each type of change.

Why must we specify two different rate functions?

Practically always, one type of decision will be made more frequently than the other

Example

In a joint study of friendship and smoking behaviour at high school, we would expect more frequent changes in the network than in behaviour

(what about friendship and delinquency???)

The rate functions

Network rate function

$T_i^{net} =$ waiting time until i gets the opportunity to make a network change

$$T_i^{net} \sim Exp(\lambda_i^{net})$$

Behaviour rate function

$T_i^{beh} =$ waiting time until i gets the opportunity to make a behavioural change

$$T_i^{beh} \sim Exp(\lambda_i^{beh})$$

The rate functions

Network rate function

T_i^{net} = waiting time until i gets the opportunity to make a network change

$$T_i^{net} \sim Exp(\lambda_i^{net})$$

Behaviour rate function

T_i^{beh} = waiting time until i gets the opportunity to make a behavioural change

$$T_i^{beh} \sim Exp(\lambda_i^{beh})$$

Waiting time for the next micro-step

$T_i^{net \vee beh}$ = waiting time until i gets the opportunity to make any change

$$T_i^{net \vee beh} \sim Exp(\lambda_i^{net} + \lambda_i^{beh})$$

The rate functions (simplest specification)

Network rate function
$T_i^{net} =$ waiting time until i gets the opportunity to make a network change

$$T_i^{net} \sim Exp(\lambda^{net})$$

behaviour rate function
$T_i^{beh} =$ waiting time until i gets the opportunity to make a behavioural change

$$T_i^{beh} \sim Exp(\lambda^{beh})$$

The rate functions (simplest specification)

Network rate function

T_i^{net} = waiting time until i gets the opportunity to make a network change

$$T_i^{net} \sim Exp(\lambda^{net})$$

behaviour rate function

T_i^{beh} = waiting time until i gets the opportunity to make a behavioural change

$$T_i^{beh} \sim Exp(\lambda^{beh})$$

Waiting time for the next micro-step

$T_i^{net \vee beh}$ = the waiting time until i gets the opportunity to make any change

$$T_i^{net \vee beh} \sim Exp(\lambda^{net} + \lambda^{beh})$$

The rate functions (simplest specification)

Probabilities for an actor to make a micro-step

$$P(i\ can\ make\ a\ network\ micro-step|opportunity) = \frac{\lambda^{net}}{\lambda^{net} + \lambda^{beh}}$$

$$P(i\ can\ make\ a\ behavioural\ micro-step|opportunity) = \frac{\lambda^{beh}}{\lambda^{net} + \lambda^{beh}}$$

The rate functions (simplest specification)

Probabilities for an actor to make a micro-step

$$P(i\ can\ make\ a\ network\ micro-step|opportunity) = \frac{\lambda^{net}}{\lambda^{net} + \lambda^{beh}}$$

$$P(i\ can\ make\ a\ behavioural\ micro-step|opportunity) = \frac{\lambda^{beh}}{\lambda^{net} + \lambda^{beh}}$$

Probabilities for a micro-step

$$P(network\ micro-step) = \frac{n\lambda^{net}}{n(\lambda^{net} + \lambda^{beh})} = \frac{\lambda^{net}}{\lambda^{net} + \lambda^{beh}}$$

$$P(behavioural\ micro-step) = \frac{n\lambda^{beh}}{n(\lambda^{net} + \lambda^{beh})} = \frac{\lambda^{beh}}{\lambda^{net} + \lambda^{beh}}$$

The evaluation functions

Why must we specify two different evaluation functions?

The evaluation functions

Why must we specify two different evaluation functions?

- The network evaluation function represents
  how likely it is for *i* to change one of his outgoing ties
- The behavioural evaluation function represents
  how likely it is for the actor *i* the current level of his behaviour

The evaluation functions

Why must we specify two different evaluation functions?

- The network evaluation function represents
  how likely it is for i to change one of his outgoing ties
- The behavioural evaluation function represents
  how likely it is for the actor i the current level of his behaviour

Network utility function: we know it!

$$
\begin{aligned}
u_i^{net}(\beta, x(i \rightsquigarrow j), z, v) &= f_i^{net}(\beta, x(i \rightsquigarrow j), z, v) + \mathcal{E}_{ij} \\
&= \sum_{k=1}^{K} \beta_k s_{ik}^{net}(x, z, v) + \mathcal{E}_{ij}
\end{aligned}
$$

The behavioural evaluation function

$$
\begin{aligned}
u_i^{beh}(\gamma, z(l \rightsquigarrow l'), x, v) &= f_i^{beh}(\gamma, z(l \rightsquigarrow l'), x, v) + \mathcal{E}_{ll'} \\
&= \sum_{w=1}^{W} \gamma_w s_{iw}^{beh}(x, z(l \rightsquigarrow l'), v) + \mathcal{E}_{ll'}
\end{aligned}
$$

where

- $s_{iw}^{beh}(x, z(l \rightsquigarrow l'), v)$ are statistics
- γ_w are statistical parameters
- $\mathcal{E}_{ll'}$ is a random term (Gumbel distributed)

The behavioural evaluation function

$$
\begin{aligned}
u_i^{beh}(\gamma, z(l \rightsquigarrow l'), x, v) &= f_i^{beh}(\gamma, z(l \rightsquigarrow l'), x, v) + \mathcal{E}_{ll'} \\
&= \sum_{w=1}^{W} \gamma_w s_{iw}^{beh}(x, z(l \rightsquigarrow l'), v) + \mathcal{E}_{ll'}
\end{aligned}
$$

where

- $s_{iw}^{beh}(x, z(l \rightsquigarrow l'), v)$ are statistics
- γ_w are statistical parameters
- $\mathcal{E}_{ll'}$ is a random term (Gumbel distributed)

The probability that an actor i changes his own behaviour by one unit is:

$$
p_{ll'}(i) = \frac{exp\left(f_i^{beh}(\gamma, z(l \rightsquigarrow l'), x, v)\right)}{\sum_{l'' \in \{l+1, l-1, l\}} exp\left(f_i^{beh}(\gamma, z(l \rightsquigarrow l''), x, v)\right)}
$$

$p_{ll}(i)$ is the probability that i does not change his behaviour.

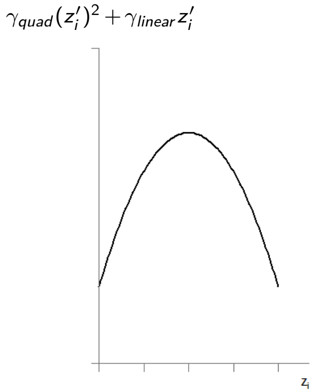
N.b. In the following we will write z' instead of $z(l \rightsquigarrow l')$

The behavioural evaluation function

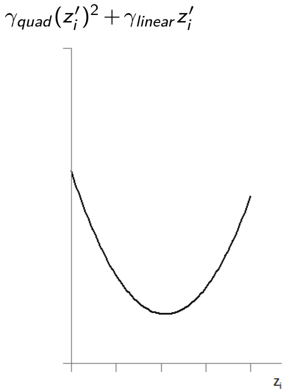
Basic shape effects

$$s^{beh}_{i\_linear}(x, z', v) = z'_i \qquad s^{beh}_{i\_quadratic}(x, z', v) = (z'_i)^2$$

The basic shape effects must be always included in the model specification



$\gamma_{quad} < 0$



$\gamma_{quad} > 0$

The behavioural evaluation function

Classical influence effects

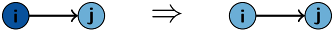
1. The *average similarity effect*

$$s^{beh}_{i\_avsim}(x, z', v) = \frac{1}{\left(\sum_{j=1}^{n} x_{ij}\right)} \sum_{j=1}^{n} x_{ij} \left(1 - \frac{\left|z'_i - z'_j\right|}{R_z}\right)$$

R_z is the range of the behaviour z

2. The *total similarity effect*

$$s^{beh}_{i\_totsim}(x, z', v) = \sum_{j=1}^{n} x_{ij} \left(1 - \frac{\left|z'_i - z'_j\right|}{R_z}\right)$$



Interpretation:

$\gamma_{avsim>(<)0}$: evidence towards (against) influence

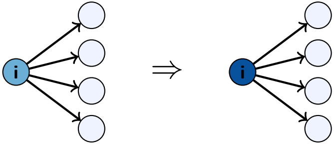
The behavioural evaluation function

Position-dependent influence effects

Network position could also have an effect on the dynamics of the behaviour

1. *Outdegree effect*

$$s_{i\_out}^{beh}(x, z', v) = z_i' \sum_{j=1}^{n} x_{ij}$$



Interpretation: $\gamma_{out} > (<) 0$: active actors tend to increase (decrease) their level of the behaviour

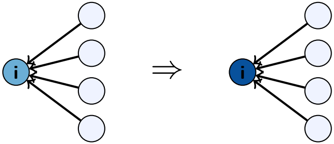
The behavioural evaluation function

Position-dependent influence effects

Network position could also have an effect on the dynamics of the behaviour

2. *Indegree effect*

$$s^{beh}_{i\_ind}(x, z', v) = z'_i \sum_{j=1}^{n} x_{ji}$$

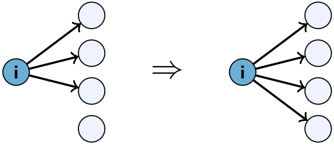
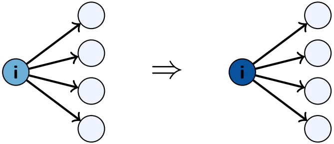
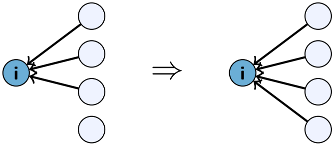
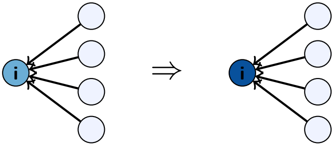


Interpretation:
$\gamma_{ind} > (<) 0$: popular actors tend to increase (decrease) their level of the behaviour

Effects of other actor variables

For each actor's attribute a main effect on the behaviour can be included in the model

Effects: distinguishing selection from influence

Selection	Influence
Covariate-ego $s_{i\_cego}(x', v) = v_i \sum_j x'_{ij}$	Outdegree $s^{beh}_{i\_out}(x, z', v) = z'_i \sum_j x_{ij}$
	
Covariate-alter $s_{i\_calt}(x', v) = \sum_j x'_{ij} v_j$	Indegree $s^{beh}_{i\_ind}(x, z', v) = z'_i \sum_j x_{ji}$
	

Effects: distinguishing selection from influence

Selection	Influence
Covariate-related similarity	Total similarity
$s_{i\_csim}(x', v) = \sum_j x'_{ij} \left(1 - \frac{\lvert v_i - v_j \rvert}{R_V}\right)$	$s^{beh}_{i\_totsim}(x, z', v) = \sum_{j=1}^{n} x_{ij} \left(1 - \frac{\lvert z'_i - z'_j \rvert}{R_z}\right)$
$i \quad j \quad \Rightarrow \quad i \longrightarrow j$	$i \longrightarrow j \quad \Rightarrow \quad i \longrightarrow j$

They differ in the dependent variable!

Outline

Simulating the co-evolution of networks and behaviour

Aim: given $(x, z)(t_0)$ and fixed parameter values,
provide $(x, z)^{sim}(t_1)$ according to the process behind the SAOM

$$\Downarrow$$

reproduce a possible series of network and behavioural micro-steps between t_0 and t_1

Input

$n =$ number of actors

$\lambda^{net} =$ network rate parameter (given)

$\lambda^{beh} =$ behaviour rate parameter (given)

$\beta = (\beta_1, \ldots, \beta_K) =$ evaluation function parameters (given)

$\gamma = (\gamma_1, \ldots, \gamma_W) =$ evaluation function parameters (given)

$(x, z)(t_0) =$ network and behaviour at time t_0 (given)

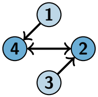
Output

$(x, z)^{sim}(t_1) =$ network and behaviour at time t_1

Simulating the co-evolution of networks and behaviour

Algorithm: Network-behaviour co-evolution

Input: $x(t_0)$, $z(t_0)$, λ^{net}, λ^{beh}, β, γ, n
Output: $x^{sim}(t_1)$, $z^{sim}(t_1)$
$t \leftarrow 0$; $x \leftarrow x(t_0)$; $z \leftarrow z(t_0)$
while *condition*=*TRUE* **do**
    $dt^{net} \sim Exp(n\lambda^{net})$; $dt^{beh} \sim Exp(n\lambda^{beh})$
    if $\min\{dt^{net}, dt^{beh}\} = dt^{net}$ **then**
        $i \sim Uniform(1, \ldots, n)$,
        $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
        if $i \neq j$ **then**
            $x \leftarrow x(i \rightsquigarrow j)$
        $t \leftarrow t + dt^{net}$
    else
        $i \sim Uniform(1, \ldots, n)$,
        $l' \sim Multinomial(p_{l(l-1)}, p_{ll'}, p_{l(l+1)})$
        if $l \neq l'$ **then**
            $z \leftarrow z(l \rightsquigarrow l')$
        $t \leftarrow t + dt^{beh}$
$x^{sim}(t_1) \leftarrow x$; $z^{sim}(t_1) \leftarrow z$
return $x^{sim}(t_1)$, $z^{sim}(t_1)$



0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

2
3
2
3

$(x, z)(t_0)$

$n = 4$

$\lambda^{net} = 1.5$

$\lambda^{beh} = 1$

$\beta = (\beta_{out}, \beta_{rec}, \beta_{trans})$
=(-1,0.5,-0.25)

$\gamma = (\gamma_{linear}, \gamma_{quadratic})$
=(-2,1)

Simulating the co-evolution of networks and behaviour

Algorithm: Network-behaviour co-evolution

Input: $x(t_0)$, $z(t_0)$, λ^{net}, λ^{beh}, β, γ, n
Output: $x^{sim}(t_1)$, $z^{sim}(t_1)$
$t \leftarrow 0$; $x \leftarrow x(t_0)$; $z \leftarrow z(t_0)$
while *condition*=*TRUE* **do**
    $dt^{net} \sim Exp(n\lambda^{net})$; $dt^{beh} \sim Exp(n\lambda^{beh})$
    if $\min\{dt^{net}, dt^{beh}\} = dt^{net}$ **then**
        $i \sim Uniform(1, \ldots, n)$
        $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
        if $i \neq j$ **then**
            $x \leftarrow x(i \rightsquigarrow j)$
        $t \leftarrow t + dt^{net}$
else
    $i \sim Uniform(1, \ldots, n)$
    $l' \sim Multinomial(p_{l(l-1)}, p_{ll'}, p_{l(l+1)})$
    if $l \neq l'$ **then**
        $z \leftarrow z(l \rightsquigarrow l')$
    $t \leftarrow t + dt^{beh}$
$x^{sim}(t_1) \leftarrow x$; $z^{sim}(t_1) \leftarrow z$
return $x^{sim}(t_1)$, $z^{sim}(t_1)$

Generating the waiting time:

- dt^{net} for a tie change
- dt^{beh} for a behaviour change

Simulating the co-evolution of networks and behaviour

Algorithm: Network-behaviour co-evolution

Input: $x(t_0)$, $z(t_0)$, λ^{net}, λ^{beh}, β, γ, n
Output: $x^{sim}(t_1)$, $z^{sim}(t_1)$
$t \leftarrow 0$; $x \leftarrow x(t_0)$; $z \leftarrow z(t_0)$
while *condition*=*TRUE* **do**
  $dt^{net} \sim Exp(n\lambda^{net})$; $dt^{beh} \sim Exp(n\lambda^{beh})$
  if $\min\{dt^{net}, dt^{beh}\} = dt^{net}$ **then**
    $i \sim Uniform(1, \ldots, n)$
    $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
    if $i \neq j$ **then**
      $x \leftarrow x(i \rightsquigarrow j)$
    $t \leftarrow t + dt^{net}$
  else
    $i \sim Uniform(1, \ldots, n)$
    $l' \sim Multinomial(p_{l(l-1)}, p_{ll'}, p_{l(l+1)})$
    if $l \neq l'$ **then**
      $z \leftarrow z(l \rightsquigarrow l')$
    $t \leftarrow t + dt^{beh}$
$x^{sim}(t_1) \leftarrow x$; $z^{sim}(t_1) \leftarrow z$
return $x^{sim}(t_1)$, $z^{sim}(t_1)$

Which micro-step is going to happen?

If

$$dt^{net} < dt^{beh}$$

then a network micro-step takes place

The following steps are the same of those in the algorithm for the network evolution

Simulating the co-evolution of networks and behaviour

Algorithm: Network-behaviour co-evolution

Input: $x(t_0)$, $z(t_0)$, λ^{net}, λ^{beh}, β, γ, n
Output: $x^{sim}(t_1)$, $z^{sim}(t_1)$
$t \leftarrow 0$; $x \leftarrow x(t_0)$; $z \leftarrow z(t_0)$
while *condition=TRUE* **do**
    $dt^{net} \sim Exp(n\lambda^{net})$; $dt^{beh} \sim Exp(n\lambda^{beh})$
    if $\min\{dt^{net}, dt^{beh}\} = dt^{net}$ **then**
        $i \sim Uniform(1, \ldots, n)$
        $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
        if $i \neq j$ **then**
            $x \leftarrow x(i \rightsquigarrow j)$
        $t \leftarrow t + dt^{net}$
    else
        $i \sim Uniform(1, \ldots, n)$
        $l' \sim Multinomial(p_{l(l-1)}, p_{ll'}, p_{l(l+1)})$
        if $l \neq l'$ **then**
            $z \leftarrow z(l \rightsquigarrow l')$
        $t \leftarrow t + dt^{beh}$
$x^{sim}(t_1) \leftarrow x$; $z^{sim}(t_1) \leftarrow z$
return $x^{sim}(t_1)$, $z^{sim}(t_1)$

Which micro-step is going to happen?

If

$$dt^{beh} < dt^{net}$$

then a behaviour micro-step takes place

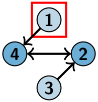
Simulating the co-evolution of networks and behaviours

Algorithm: Network-behaviour co-evolution

Input: $x(t_0)$, $z(t_0)$, λ^{net}, λ^{beh}, β, γ, n
Output: $x^{sim}(t_1)$, $z^{sim}(t_1)$
$t \leftarrow 0$; $x \leftarrow x(t_0)$; $z \leftarrow z(t_0)$
while *condition=TRUE* **do**
  $dt^{net} \sim Exp(n\lambda^{net})$; $dt^{beh} \sim Exp(n\lambda^{beh})$
  if $\min\{dt^{net}, dt^{beh}\} = dt^{net}$ **then**
    $i \sim Uniform(1, \ldots, n)$
    $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
    if $i \neq j$ **then**
      $x \leftarrow x(i \rightsquigarrow j)$
    $t \leftarrow t + dt^{net}$
  else
    $i \sim Uniform(1, \ldots, n)$
    $l' \sim Multinomial(p_{l(l-1)}, p_{ll'}, p_{l(l+1)})$
    if $l \neq l'$ **then**
      $z \leftarrow z(l \rightsquigarrow l')$
    $t \leftarrow t + dt^{beh}$
$x^{sim}(t_1) \leftarrow x$; $z^{sim}(t_1) \leftarrow z$
return $x^{sim}(t_1)$, $z^{sim}(t_1)$

Select the actor i who has the opportunity to change his behaviour

e.g. i=*1*



0	0	0	1
0	0	0	1
0	1	0	0
0	1	0	0

2
3
2
3

$(x, z)(t_0)$

Simulating the co-evolution of networks and behaviours

Algorithm: Network-behaviour co-evolution

Input: $x(t_0)$, $z(t_0)$, λ^{net}, λ^{beh}, β, γ, n
Output: $x^{sim}(t_1)$, $z^{sim}(t_1)$
$t \leftarrow 0$; $x \leftarrow x(t_0)$; $z \leftarrow z(t_0)$
while *condition=TRUE* **do**
    $dt^{net} \sim Exp(n\lambda^{net})$; $dt^{beh} \sim Exp(n\lambda^{beh})$
    if $\min\{dt^{net}, dt^{beh}\} = dt^{net}$ **then**
        $i \sim Uniform(1, \ldots, n)$
        $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
        if $i \neq j$ **then**
            $x \leftarrow x(i \rightsquigarrow j)$
        $t \leftarrow t + dt^{net}$
    else
        $i \sim Uniform(1, \ldots, n)$;
        $l' \sim Multinomial(p_{l(l-1)}, p_{ll'}, p_{l(l+1)})$
        if $l \neq l'$ **then**
            $z \leftarrow z(l \rightsquigarrow l')$
        $t \leftarrow t + dt^{beh}$
$x^{sim}(t_1) \leftarrow x$; $z^{sim}(t_1) \leftarrow z$
return $x^{sim}(t_1)$, $z^{sim}(t_1)$

Select the level l' towards i is going to adjust his behaviour

$l \rightarrow l'$	f_i^{beh}	$p_{ll'}$
$2 \rightarrow 1$	-1	0.017
$2 \rightarrow 2$	0	0.047
$3 \rightarrow 3$	3	0.936

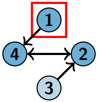
Simulating the co-evolution of networks and behaviours

Algorithm: Network-behaviour co-evolution

Input: $x(t_0)$, $z(t_0)$, λ^{net}, λ^{beh}, β, γ, n
Output: $x^{sim}(t_1)$, $z^{sim}(t_1)$
$t \leftarrow 0$; $x \leftarrow x(t_0)$; $z \leftarrow z(t_0)$
while *condition*=*TRUE* **do**
    $dt^{net} \sim Exp(n\lambda^{net})$; $dt^{beh} \sim Exp(n\lambda^{beh})$
    if $\min\{dt^{net}, dt^{beh}\} = dt^{net}$ **then**
        $i \sim Uniform(1, \dots, n)$
        $j \sim Multinomial(p_{i1}, \dots, p_{in})$
        if $i \neq j$ **then**
            $x \leftarrow x(i \rightsquigarrow j)$
        $t \leftarrow t + dt^{net}$
    else
        $i \sim Uniform(1, \dots, n)$
        $l' \sim Multinomial(p_{l(l-1)}, p_{ll'}, p_{l(l+1)})$
        if $l \neq l'$ **then**
            $z \leftarrow z(l \rightsquigarrow l')$
        $t \leftarrow t + dt^{beh}$
$x^{sim}(t_1) \leftarrow x$; $z^{sim}(t_1) \leftarrow z$
return $x^{sim}(t_1)$, $z^{sim}(t_1)$

Select the level l' towards i is going to adjust his behaviour

e.g. l'=*3*



$$\begin{array}{|c|c|c|c|}\hline 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 1 \\ \hline 0 & 1 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 \\ \hline \end{array} \qquad \begin{array}{|c|}\hline 3 \\ \hline 3 \\ \hline 2 \\ \hline 3 \\ \hline \end{array}$$

$(x, z(l \rightarrow l'))$

Simulating the co-evolution of networks and behaviours

Algorithm: Network-behaviour co-evolution

Input: $x(t_0)$, $z(t_0)$, λ^{net}, λ^{beh}, β, γ, n
Output: $x^{sim}(t_1)$, $z^{sim}(t_1)$
$t \leftarrow 0$; $x \leftarrow x(t_0)$; $z \leftarrow z(t_0)$
while *condition=TRUE* **do**
  $dt^{net} \sim Exp(n\lambda^{net})$; $dt^{beh} \sim Exp(n\lambda^{beh})$
  if $\min\{dt^{net}, dt^{beh}\} = dt^{net}$ **then**
    $i \sim Uniform(1, \ldots, n)$
    $j \sim Multinomial(p_{i1}, \ldots, p_{in})$
    if $i \neq j$ **then**
      $x \leftarrow x(i \rightsquigarrow j)$
    $t \leftarrow t + dt^{net}$
  else
    $i \sim Uniform(1, \ldots, n)$
    $l' \sim Multinomial(p_{l(l-1)}, p_{ll'}, p_{l(l+1)})$
    if $l \neq l'$ **then**
      $z \leftarrow z(l \rightsquigarrow l')$
    $t \leftarrow t + dt^{beh}$

$x^{sim}(t_1) \leftarrow x$; $z^{sim}(t_1) \leftarrow z$
return $x^{sim}(t_1)$, $z^{sim}(t_1)$

Simulating the co-evolution of networks and behaviour

1. *Unconditional* simulation:

   simulation carries on until a predetermined time length has elapsed (usually until $t = 1$).

Simulating the co-evolution of networks and behaviour

1. *Unconditional* simulation:

   simulation carries on until a predetermined time length has elapsed (usually until $t = 1$).

2. *Conditional* simulation on the observed number of changes:
   - simulation runs on until

$$\sum_{\substack{i,j=1 \\ i \neq j}}^{n} \left| X_{ij}^{obs}(t_1) - X_{ij}(t_0) \right| = \sum_{i,j=1}^{n} \left| X_{ij}^{sim}(t_1) - X_{ij}(t_0) \right|$$

Simulating the co-evolution of networks and behaviour

1. *Unconditional* simulation:

   simulation carries on until a predetermined time length has elapsed (usually until $t = 1$).

2. *Conditional* simulation on the observed number of changes:
   - simulation runs on until

     $$\sum_{\substack{i,j=1 \\ i \neq j}}^{n} \left| X_{ij}^{obs}(t_1) - X_{ij}(t_0) \right| = \sum_{i,j=1}^{n} \left| X_{ij}^{sim}(t_1) - X_{ij}(t_0) \right|$$

   - or until

     $$\sum_{i=1}^{n} \left| z_i^{obs}(t_1) - z_i(t_0) \right| = \sum_{i=1}^{n} \left| z_i^{sim}(t_1) - z_i(t_0) \right|$$

Outline

Parameter estimation

Aim: given the longitudinal data

$$(x,z)(t_0),\ldots,(x,z)(t_M) \qquad V_1,\ldots,V_H$$

estimate the parameters for the co-evolution model

- M rate parameters for the network rate function

$$\lambda_1^{net},\ \ldots,\ \lambda_M^{net}$$

- M rate parameters for the behaviour rate function

$$\lambda_1^{beh},\ \ldots,\ \lambda_M^{beh}$$

- K and W parameters for the network evaluation function and the behaviour evaluation function, respectively

$$f_i^{net}(\beta,x',z,v) = \sum_{k=1}^{K} \beta_k s_{ik}^{net}(x',z,v)$$
$$f_i^{beh}(\gamma,x',z,v) = \sum_{w=1}^{W} \gamma_w s_{iw}^{beh}(x,z',v)$$

Parameter estimation

Issue

Given

$$(x,z)(t_0),\ldots,(x,z)(t_M) \qquad V_1,\ldots,V_H$$

and a specification of the SAOM, we want to estimate

$$\theta = (\lambda_1^{net},\ \ldots,\lambda_M^{net},\lambda_1^{beh},\ \ldots,\lambda_M^{beh},\beta_1,\ \ldots,\beta_K,\gamma_1,\ \ldots,\gamma_W)$$

Two estimation methods are implemented in `Rsiena`:

1. Method of Moments
2. Maximum-likelihood estimation

Parameter estimation (MoM)

We can estimate the $2M + K + W$-dimensional parameter θ using the MoM

Parameter estimation (MoM)

We can estimate the $2M + K + W$-dimensional parameter θ using the MoM

In practice:

1. find $2M + K + W$ statistics
2. set the theoretical expected value of each statistic equal to its sample counterpart
3. solve the resulting system of equations

$$E_\theta[S - s] = 0$$

   with respect to θ

Parameter estimation (MoM)

Statistics:

- Network rate parameters for the period m

$$s^{net}_{\lambda_m}(X(t_m), X(t_{m-1})|X(t_{m-1})) = \sum_{i,j=1}^{n} |X_{ij}(t_m) - X_{ij}(t_{m-1})|$$

- behaviour rate parameters for the period m

$$s^{beh}_{\lambda_m}(Z(t_m), Z(t_{m-1})|Z(t_{m-1})) = \sum_{i=1}^{n} |Z_i(t_m) - Z_i(t_{m-1})|$$

$m = 1, \ldots, M$

Parameter estimation (MoM)

Statistics:

- Network evaluation function effects

$$\sum_{m=1}^{M} s_{mk}^{net}\left(X(t_m)|(Z,V)(t_{m-1})\right)$$

- behaviour evaluation function effects

$$\sum_{m=1}^{M} s_{mw}^{beh}\left(Z(t_m)|(X,V)(t_{m-1})\right)$$

Parameter estimation (MoM)

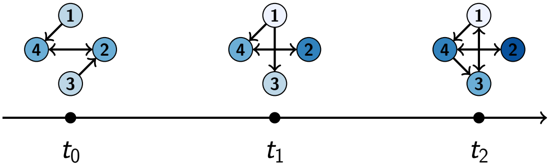
Consequently the MoM estimator for θ is provided by the solution of:

$$\begin{cases} E_\theta\left[s^{net}_{\lambda_m}(X(t_M), X(t_{m-1})|X(t_{m-1}))\right] = s^{net}_{\lambda_m}(x(t_m), x(t_{m-1})) & m = 1, \dots, M \\ E_\theta\left[s^{beh}_{\lambda_m}(Z(t_m), Z(t_{m-1})|Z(t_{m-1}))\right] = s^{beh}_{\lambda_m}(z(t_m), z(t_{m-1})) & m = 1, \dots, M \\ E_\theta\left[\sum_{m=1}^{M} s^{net}_{mk}(X(t_m)|(X, Z, V)(t_{m-1}))\right] = \sum_{m=1}^{M} s^{net}_{mk}(x(t_m)|(x, z, v)(t_{m-1})) & k = 1, \dots, K \\ E_\theta\left[\sum_{m=1}^{M} s^{beh}_{mw}(Z(t_m)|(X, Z, V)(t_{m-1}))\right] = \sum_{m=1}^{M} s^{beh}_{mw}(z(t_m)|(x, z, v)(t_{m-1})) & w = 1, \dots, W \end{cases}$$

Parameter estimation (MoM)

Example

Let us assume to have observed a network at $M = 3$ time points



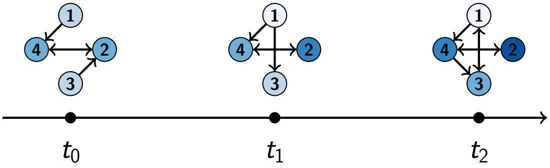
$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 2 \\ 3 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 4 \\ 2 \\ 3 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 5 \\ 3 \\ 4 \end{bmatrix}$$

We want to model the network evolution according to the outdegree, the reciprocity, the linear shape and the quadratic shape effects

Parameter estimation (MoM)

Example

Let us assume to have observed a network at $M = 3$ time points



$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 2 \\ 3 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 4 \\ 2 \\ 3 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 5 \\ 3 \\ 4 \end{bmatrix}$$

We want to model the network evolution according to the outdegree, the reciprocity, the linear shape and the quadratic shape effects

$$\theta = (\lambda_1^{net}, \lambda_2^{net}, \lambda_1^{beh}, \lambda_2^{beh}, \beta_{out}, \beta_{rec}, \gamma_{linear}, \gamma_{quadratic})$$

Parameter estimation (MoM)

Example

Statistics for the network evolution:

$$s_{\lambda_1^{net}}(X(t_1), X(t_0)|X(t_0) = x(t_0)) = \sum_{i,j=1}^{4} |X_{ij}(t_1) - X_{ij}(t_0)|$$

$$s_{\lambda_2^{net}}(X(t_2), X(t_1)|X(t_1) = x(t_1)) = \sum_{i,j=1}^{4} |X_{ij}(t_2) - X_{ij}(t_1)|$$

$$\sum_{m=1}^{M-1} s_{out}\left(X(t_m)|X(t_{m-1}) = x(t_{m-1})\right) = \sum_{m=1}^{2} \sum_{i,j=1}^{4} X_{ij}(t_m)$$

$$\sum_{m=1}^{M-1} s_{rec}\left(X(t_m)|X(t_{m-1}) = x(t_{m-1})\right) = \sum_{m=1}^{2} \sum_{i,j=1}^{4} X_{ij}(t_m) X_{ji}(t_m)$$

Parameter estimation (MoM)

Example

Statistics for the behaviour evolution:

$$s_{\lambda_1^{beh}}(Z(t_1), Z(t_0)|Z(t_0) = z(t_0)) = \sum_{i=1}^{4} |Z_i(t_1) - Z_i(t_0)|$$

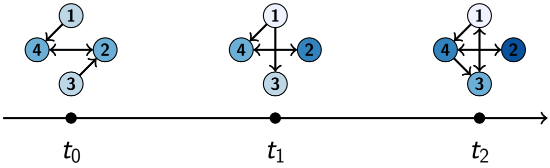
$$s_{\lambda_2^{beh}}(Z(t_2), Z(t_1)|Z(t_1) = z(t_1)) = \sum_{i=1}^{4} |Z_i(t_2) - Z_i(t_1)|$$

$$\sum_{m=1}^{M} s_{linear}\,(Z(t_m)|Z(t_{m-1}) = z(t_{m-1})) = \sum_{m=1}^{2}\sum_{i=1}^{4} z_i(t_m)$$

$$\sum_{m=1}^{M} s_{quadratic}\,(Z(t_m)|Z(t_{m-1}) = z(t_{m-1})) = \sum_{m=1}^{2}\sum_{i=1}^{4} z_i^2(t_m)$$

Parameter estimation (MoM)

Example



$$\begin{bmatrix}0&0&0&1\\0&0&0&1\\0&1&0&0\\0&1&0&0\end{bmatrix}\begin{bmatrix}2\\3\\2\\3\end{bmatrix}\qquad\begin{bmatrix}0&0&0&1\\0&0&0&1\\0&1&0&0\\0&1&0&0\end{bmatrix}\begin{bmatrix}1\\4\\2\\3\end{bmatrix}\qquad\begin{bmatrix}0&0&1&1\\0&0&0&1\\0&0&0&0\\0&1&1&0\end{bmatrix}\begin{bmatrix}1\\5\\3\\4\end{bmatrix}$$

$s_{\lambda_1^{net}} = 2$ $\qquad$ $s_{\lambda_2^{net}} = 2$

$s_{\lambda_1^{beh}} = 2$ $\qquad$ $s_{\lambda_2^{beh}} = 3$

$s_{out} = 4 + 6 = 10$ $\qquad$ $s_{rec} = 2 + 4 = 6$

$s_{linear} = 10 + 13 = 23$ $\qquad$ $s_{quadratic} = 30 + 51 = 81$

The parameter estimation (MoM)

Example

We look for the value of θ that satisfies the system:

$$
\begin{cases}
E_\theta\left[S_{\lambda_1^{net}}\right] = 2 \\
E_\theta\left[S_{\lambda_2^{net}}\right] = 2 \\
E_\theta\left[S_{\lambda_1^{beh}}\right] = 2 \\
E_\theta\left[S_{\lambda_2^{beh}}\right] = 3 \\
E_\theta[S_{out}] = 10 \\
E_\theta[S_{rec}] = 6 \\
E_\theta[S_{linear}] = 23 \\
E_\theta[S_{quadratic}] = 51
\end{cases}
$$

Parameter estimation (MoM)

In a more compact notation, we look for the value of θ that satisfies the system:

$$E_\theta[S - s] = 0$$

but we know that we cannot solve it analytically.

The soultion is again provided by the Robbins-Monro algorithm.

Parameter estimation (MoM)

The Robbins-Monro algorithm

Given an initial guess θ_0 for the parameter θ, the procedure can be roughly depicted as follows:

$$
\begin{array}{ccccc}
\theta_0 & \xrightarrow{\textit{approximation}} & E_{\theta_0}[S-s] & \xrightarrow{\textit{update}} & \theta_1 \\
\theta_1 & \xrightarrow{\textit{approximation}} & E_{\theta_1}[S-s] & \xrightarrow{\textit{update}} & \theta_2 \\
\ldots & \xrightarrow{\textit{approximation}} & \ldots & \xrightarrow{\textit{update}} & \ldots \\
\theta_{i-1} & \xrightarrow{\textit{approximation}} & E_{\theta_{i-1}}[S-s] & \xrightarrow{\textit{update}} & \theta_i \\
\ldots & \xrightarrow{\textit{approximation}} & \ldots & \xrightarrow{\textit{update}} & \ldots
\end{array}
$$

until a certain criterion is satisfied

Parameter estimation (MoM)

The Robbins-Monro algorithm

- The expected value is approximated using the Monte Carlo method:
  - the evolution process is simulated q times according to θ_i
  - the statistics are computed for each simulation
  - $E_\theta[S]$ is approximated by the average of the simulated values of the statistics

- The updating rule is based on the Robbins-Monro step

$$\widehat{\theta}_{i+1} = \widehat{\theta}_i - a_i \widehat{D}^{-1} (S_i - s)$$

where $\widehat{D}$ is a diagonal matrix of first order derivatives

$$\widehat{D} = \frac{\partial}{\partial \widehat{\theta}_i} E_{\widehat{\theta}_i}[S]$$

Parameter estimation (MoM)

The Robbins-Monro algorithm

Phase 1

To compute the Robbins-Monro step we need some "ingredients":

- the initial value of θ denoted by θ_0

  $$\theta_0 = (\lambda_1^{net}, \ldots, \lambda_M^{net}, \lambda_1^{beh}, \ldots, \lambda_M^{beh}, \beta_{out}, \beta_k = 0, \gamma_{linear}, \gamma_w = 0)$$

  where the choice of
  - $\lambda_1^{\cdot}, \ldots, \lambda_M^{\cdot}$ is based on the number of (network/behavioral) changes
  - β_{dens} is based on the density of the observed networks
  - $\gamma_{linear}, \gamma_{quad}$ is based on the observed distribution of the behaviour
- an estimate of $\widehat{D}$
  given θ_0, the network evolution is simulated and the derivatives are estimated using the Monte Carlo method

Parameter estimation (MoM)

The Robbins-Monro algorithm

Phase 2

- Comprising of 4 sub-phases.
- Each sub-phase consists of the following steps
  1. Generate the network evolution from the current value of θ_i
  2. Compute the value of the simulated statistics and update the parameter value
  $$\widehat{\theta}_{i+1} = \widehat{\theta}_i - a_i \widehat{D}^{-1} (S_i - s)$$
  3. 1. and 2. are repeated n_{2sub} time or until $(S_i - s)(S_{i-1} - s) < 0$
  4. The new value for θ is the average of $\widehat{\theta}_i$ over the sub-phase
- The average of $\widehat{\theta}_i$ over the last sub-phase is the (eventual) estimate for θ

Phase 3

Check the convergence of the algorithm and compute the covariance matrix of the estimator

Outline

Creation and Endowment function

Behavioural evaluation function

Given $x(t_0)$ and $x(t_1)$ three possible behavioural changes are possible:

$x(t_0)$	$x(t_1)$	
i	i	increase of the behavioural level
i	i	decrease of the behavioural level
i	i	maintenance of the behavioural level

Creation and Endowment function

Behavioural evaluation function

Given $x(t_0)$ and $x(t_1)$ three possible behavioural changes are possible:

$x(t_0)$	$x(t_1)$	
i	i	increase of the behavioural level
i	i	decrease of the behavioural level
i	i	maintenance of the behavioural level

The behavioural evaluation function models the level of a behaviour in a network regardless the level of a behaviour was increased or decreased...

but increasing the level of a behaviour is not always the opposite of decreasing it
(e.g. use of addictive substances)

Behavioural creation and endowment function

- Creation function
  - models the gain in the utility function when a behavioural level is increased
  - The effects are the same as those given for the behavioural evaluation function...
  - but they enter calculation only when the actor considers increasing his behavioural score by one unit

- Endowment function
  - models the gain in the utility function when a behavioural level is decreased (opposite of maintained)
  - The effects are the same as those given for the behavioural evaluation function...
  - but they enter calculation only when the actor considers decreasing his behavioural score by one unit

Outline

Selection and influence: ERGMs

Selection:

actors' attribute may affect the presence or the absence of network ties (actors may select one another as network partners, depending on the attributes that they have)

Influence:

the presence of a tie may alter the attribute of the actors (individuals may be influenced by their network partners to change their behaviours)

Dependent	Independent	
Network x	Behaviour z	Selection
Behaviour z	Network x	Influence

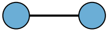
ERG selection models

In ERGMs

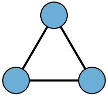
$$P_\theta(G) = \frac{1}{\kappa(\theta)} exp\left(\sum_{i=1}^{k} \theta_i s_i(G)\right)$$

the existence of ties are explained by

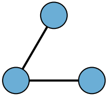
- the existence of other ties (network statistics)



edges

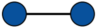


triangles

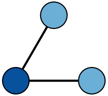


2-stars

- the attributes of the actors (covariate-related statistics)



homophily



covariate-related activity

ERG selection models

Let

- X be the adjacency matrix
- V be the actor-attribute
- Z be the behaviour

associated to a certain graph G

In ERGMs the dependent variable is the network, so that

$$P_\theta(G) = \frac{1}{\kappa(\theta)} exp\left(\sum_{i=1}^{k} \theta_i s_i(G)\right)$$

is equivalent to write

$$P_\theta(X|V,Z) = \frac{1}{\kappa(\theta)} exp\left(\sum \theta_P s_P(x) + \sum \theta_A s_A(x,v,z)\right)$$

ERG selection models

- aim: explain how a particular network structure may be a product of endogenous network processes (clustering, transitivity, popularity) and exogenous nodal and dyadic factors (gender, membership)
- If the attributes are possibly changeable, we are still treating them as predictors of networks ties
  implicit assumption: attribute are not changed by ties
- We should be careful when making inferences
  if we see a significant attribute parameter, we have evidence for an association between attributes and network ties, but we CANNOT make causal inferences

Example

If $\theta_{homophily} > 0$

- we can say that ties between actors having the same attribute are more likely
- we CANNOT say that actors having the same attribute tends to create ties among themselves

ERG influence model

- aim: how individual behaviours may be constrained by position in a network and by behaviours of other actors in the network
- implicit assumption: network ties are not changed by the attributes

In ERG influence model the dependent variable is the behaviour

$$P_\theta(Z|X,V) = \frac{1}{\kappa(\theta)} exp \left(\sum \theta_P s_P(x) + \sum \theta_I s_I(x,z) \sum \theta_C s_C(x,v) \right)$$

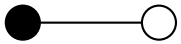
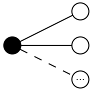
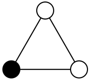
where

- $s_P(x)$ statistic accounting for the network position
- $s_I(x)$ statistic accounting for the influence of other actors
- $s_C(x)$ statistic accounting for actors' covariates

Dependence assumptions should be formulated to define these statistics using the Hammersley-Clifford theorem

Network position statistics

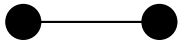
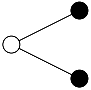
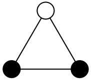
Dependence among the behaviour and the ties

Statistics			Dependence
Attribute density	$\sum_i z_i$		Independence
Actor activity	$\sum_i z_i \sum_j x_{ij}$		Z_i depends on X_{hj} if $\{i\} \cap \{h,j\} \neq \emptyset$
Actor k-star	$\sum_i z_i \binom{\sum_j x_{ij}}{k}$		
Actor triangle	$\sum_i z_i x_{ij} x_{ih} x_{hj}$		Z_i depends on X_{hj} if $x_{ij} = 1$ and $x_{jh} = 1$
...	...	...	...

The statistics comprise only the attribute of a focal actor (black node) and his ties to others, regardless of the attributes of those others (white nodes)

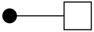
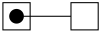
Network influence statistics

Behaviour dependence among connected actors

Statistics			Dependence
Partner attribute	$\sum_i z_i z_j x_{ij}$		Z_i depends on Z_j if $x_{ij} = 1$
Indirect partner attribute	$\sum_{i<h} z_i z_h \sum_j x_{ij} x_{jh}$		Z_i depends on Z_h if $x_{ij} = 1$ and $x_{jh} = 1$
Partner attribute triangle	$\sum_i z_i z_j x_{ij} x_{ih} x_{hj}$		
...	...	...	...

Network influence statistics

Dependence among the behaviour and actors covariates

Statistics			Dependence
Attribute covariate	$\sum_i z_i v_i$		Z_i depends only on V_i
Partner covariate attribute	$\sum_{ij} z_i v_j x_{ij}$		Z_i depends on V_i and V_j if $x_{ij} = 1$
Same partner covariate triangle	$\sum_i z_i \mathbb{I}\left\{v_i = v_j\right\} x_{ij}$		
...	...	...	...

The behaviour Z is represented by the circle and the actor attribute V is represented by a square

ERG influence models

- We should be careful when making inferences
  if we see a significant network/covariate statistics, we have evidence for an association between the behaviour and the network ties or the actors covariates, but we CANNOT make causal inferences

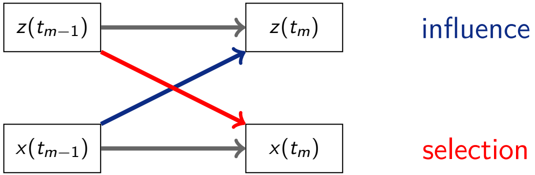
  Example

  If $\theta_{partner\ attribute} > 0$

  - we can say that connected actors are more likely to show the same behaviour
  - we CANNOT say that connected actors adjust their behaviour according to the behaviour of those they are connected to

Selection and influence: ERGMs

We cannot distinguish influence and selection in cross-selectional data! We need to collect longitudinal network data.



With longitudinal network data, we know whether the attribute leads to the tie, or vice versa, the tie leads to a certain value of the attribute

- In principle TERGMs can be used to distinguish selection and influence processes
- Proper statistics should be defined and implemented

Outline

A few words on...

...topics that are not treated in the course

- Missing data
  unit non-response vs. item non-response
- Change in composition
  actors can leave or join the network
- Multi-relational network
  interest in analysing more than one relation
- Multilevel analysis of networks
  a same relation is observed on several groups
  (e.g. friendship in several school classes)
- Multilevel networks analysis
  there is a hierarchy in the nodes
  (e.g. cooperation within a firm and between firms)
- Event network models, models for two-mode networks etc.