Social Network Analysis

Network Autocorrelation

Dr. Chun-Hsiang Chan

Department of Geography,
National Taiwan Normal University

Photo credit: midjourney

Outline

– Network Autocorrelation Model
– Recall: Regression Basics
– Modeling Social Influence
– Theories on Social Influence
– Communication vs Comparison
– Adjacent vs Non-adjacent
– Operationalization of Weighting Matrix
– Summary
– R installation

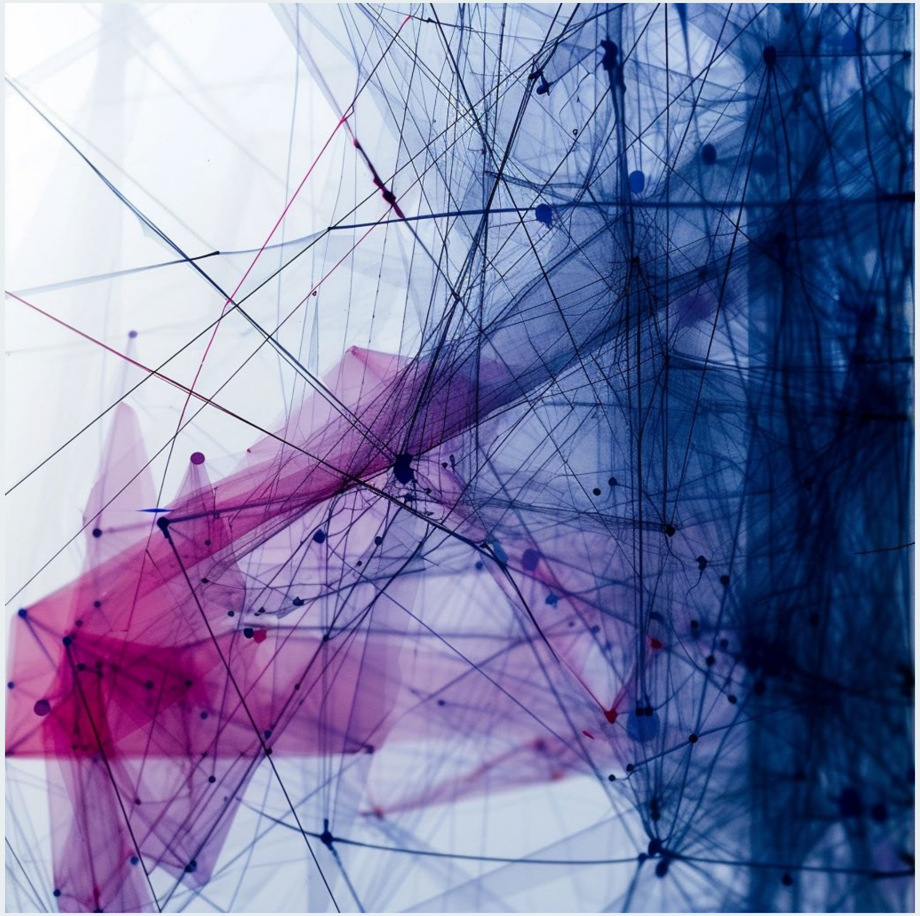


Photo credit: midjourney

Quadratic Assignment Procedure (QAP) Regression

Comparing multiple networks: QAP regression

– The substantive question is how one set of relations (or dyadic attributes) relates to another.

– Are country relations correlated with trade networks?

– Do in-person contacts associated with friendship relations from social media?

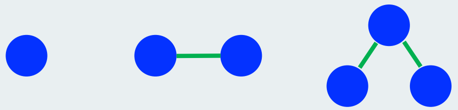
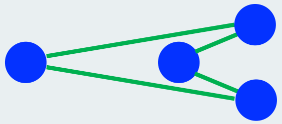
Network Autocorrelation Model

Autocorrelation concept in the social networks.

– Thousands of examples show that every decision we made is affected by our surrounding; in other words, the neighboring effect plays an important role in our daily life.
  – Homophily: people are attracted to similar others
  – Influence: people become more similar to their neighbor

Covariates

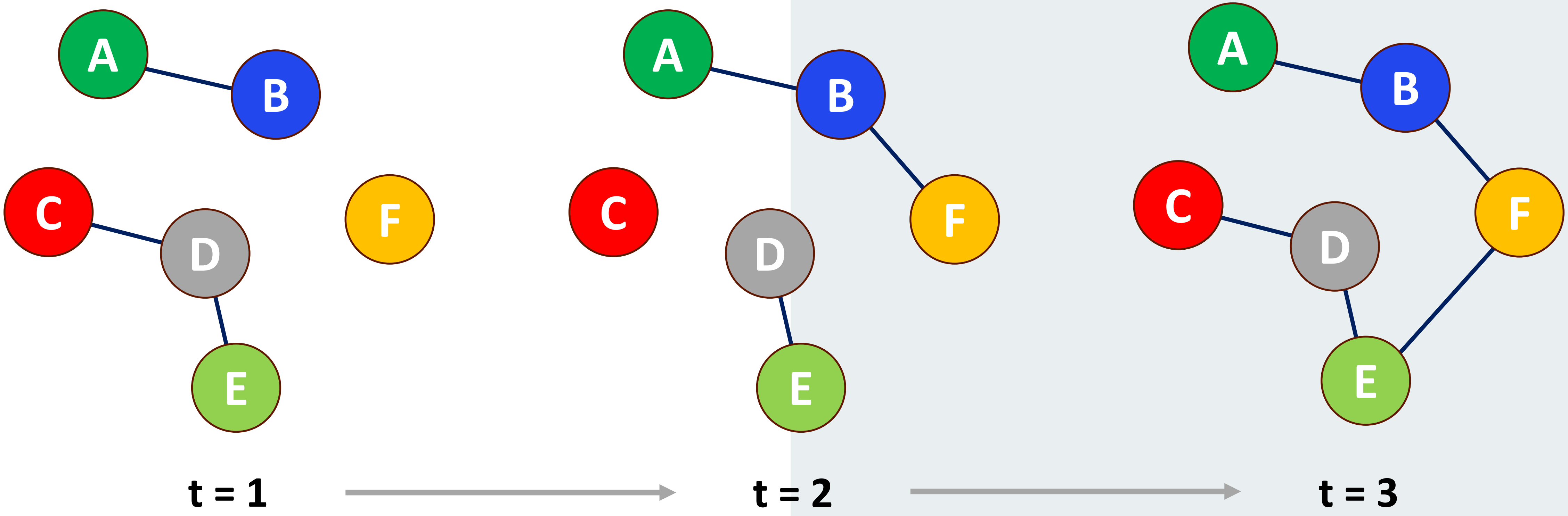
- Modeling social networks parametrically
- What creatres heterogeneity in the probability of a tie being formed?

Attributes of nodes	Heterogeneity by group - average acticity - mixing by group Individual heterogeneity
Attributes of links	Heterogeneity in - duration – types (sex, drug, …)
Configurations	Degree distributions (or stars)  Cycle distributions (2,3,4, etc.)  Shared partner distributions 

Dyad **Independent** Terms

Dyad **Dependent** Terms

Temporal Network Dynamics



Recall: Regression Basics

A statistical technique is employed to analyze the connection between a primary variable of interest (often termed the dependent variable, Y) and one or more explanatory variables (also known as predictors, X).

This method assesses:

– The strength of the relationship.

– The direction of the relationship (whether it's positive, negative, or zero).

– The goodness of fit of the model.

Recall: Regression Basics

This technique enables the calculation of how much the dependent variable changes when a predictor variable shifts by one unit, while keeping all other predictors constant.

Frequently known as Ordinary Least Squares (OLS) regression

– Its objective is to minimize ($\Sigma\left(Y_i - \widehat{Y_i}\right)^2$).
– Regression with a single predictor is termed simple regression
– Regression involving two or more predictors is referred to as multiple regression.
– Similar to correlation, if an explanatory variable **significantly predicts** the dependent variable, **it does not necessarily imply causation**.

Recall: Regression Basics

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots + \beta_n X_n + \varepsilon$$

– y: dependent variables

– $\beta_0, \beta_1, \beta_2, \ldots, \beta_n$: coefficients

– $X_1, X_2, X_3, \ldots, X_n$: explanatory variables

– ε: random error term/ residuals

Recall: Regression Basics

– The hypothesis of residuals in the OLS regression:

(1) The expected value of residuals is zero (zero mean)

$E(u) = 0$

(2) Residuals follow homoskedasticity

$var(u) = \sigma^2, \sigma^2\ is\ a\ constant$

(3) Residuals are non-autocorrelation

$cov(ut, ut - s) = 0, for\ s \neq 0$

(4) The relationship between independent variables and residuals is orthogonality

$cov(x, u) = 0, for\ any\ I$

(5) Residuals follow normality

The residuals follow abovementioned conditions can be regards as independently identical distribution (iid) residuals: $u \sim i.i.d.N(0, \sigma^2)$

Recall: Regression Basics

- Social networks vs spatial networks
- The observations should be independent of each other

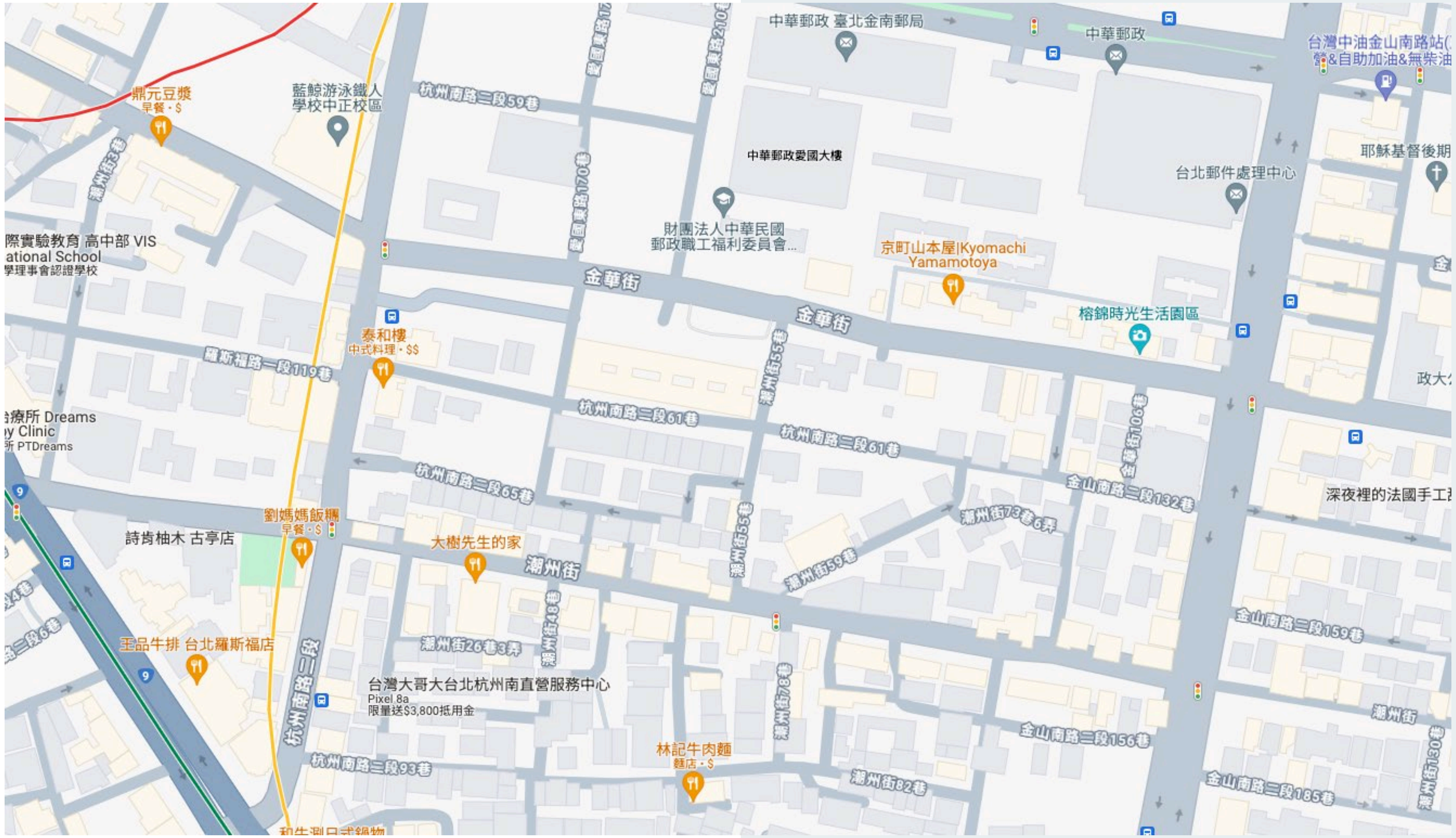
EID	States	Populaton	Area
1	Texas	30,503,301	695,660
2	Iowa	3,190,369	145,746
…	…	…	…
…	…	…	…
50	Kansas	2,937,880	213,100

Recall: Regression Basics

Spatial Regression

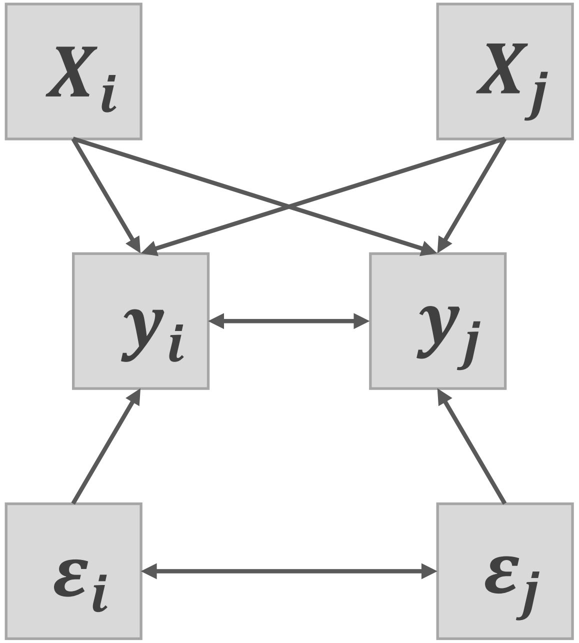
- Spatial autocorrelation is present in a variable when nearby observations in space exhibit correlated values (Tobler's Law).
- One of the assumptions of regression is the **independence of observations**. If this assumption is violated, it leads to inaccurate estimations of the β coefficients, and the error term ε contains spatial dependencies, meaning it holds meaningful information. However, ideally, we want the error term to be indistinguishable from random noise.

Social Networks vs Road Networks



Recall: Regression Basics

Spatial Lag Model vs Spatial Error Model



Spatial Lag Model (SLM)

$$y = \rho W y + X\beta + \varepsilon$$

Spatial Error Model (SEM)

$$y - \lambda W y = X\beta - \lambda W X\beta + e$$
$$(I - \lambda W)y = (I - \lambda W)X\beta + e$$

$$y = X\beta + u$$
$$u = \lambda W u + e$$

Modeling Social Influence

- Determine the occurrence of social influence:
- Assess if it manifests through the autocorrelation of the dependent variable (SLM), the autocorrelation of disturbances (SEM), or a blend of both.
- Decide which mechanism governs social influence: communication or comparison.
- Make choices regarding which elements of $\boldsymbol{W}$ are zero and which are non-zero.
- For non-zero elements of $\boldsymbol{W}$, ascertain the extent of influence exerted.

Modeling Social Influence

- Network effects model
  $y = \rho W y + X\beta + \varepsilon, \varepsilon \sim N(0, \sigma^2 I)$
  e.g., ego's voting behavior may be influenced by discussing matters with friends (interaction), but will often also depend on ego's status, income, education, and so forth (local effects)
- Network disturbances model
  $y = X\beta + \varepsilon, \varepsilon = \rho W \varepsilon + \nu, \nu \sim N(0, \sigma^2 I)$
- Combined network effect and disturbance
  $y = \rho_1 W_1 y + X\beta + \varepsilon, \varepsilon = \rho_2 W_2 \varepsilon + \nu, \nu \sim N(0, \sigma^2 I)$

Modeling Social Influence

- The decision to model contagion by either autocorrelating the dependent term or the disturbance term represents **a theoretical distinction** in understanding **how contagion occurs**.
- Definition: The opinion an actor would demonstrate in **the absence of social influence** is an actor's **intrinsic opinion** (Leenders, 1995).
- $y = \rho_1 W_1 y + \boxed{X\beta} + \varepsilon$
- $\varepsilon = \rho_2 W_2 \varepsilon + \nu$
- $\nu \sim N(0, \sigma^2 I)$

Modeling Social Influence

- Ego forms their opinion based on both their inherent beliefs and the perspectives of those around them.
- For example, ego's political inclinations are **influenced by their socioeconomic status, education, and income** (intrinsic opinion), as well as by the political viewpoints voiced by their **family, neighbors, and colleagues** (contagion).
- These combined influences jointly shape ego's political position.

Modeling Social Influence

- Ego's stance might initially be solely shaped by their inherent opinion; their status, education, and income initially dictate their political outlook. However, as **ego notices their significant others diverging from their intrinsic opinions**, ego adjusts to their deviation.
- Ego might observe their peers leaning more towards either the left or right side of the political spectrum than what their status, education, and income would suggest. Witnessing this discrepancy, **ego feels inclined to shift their own viewpoint towards the corresponding direction.**

Model Selection: Network Disturbances Model

– In a statistical context, the process of **adapting to deviations from intrinsic opinions**, rather than to the opinions themselves, is mirrored by the autocorrelation of residuals.

– These residuals, ε_i, encapsulate latent forces that steer an individual's opinion away from their intrinsic standpoint.

$$y = X\beta + \varepsilon \qquad \varepsilon = \rho W\varepsilon + \upsilon \qquad \upsilon \sim N(0, \sigma^2 I)$$

Specification of Influence Matrix (W): Importance

– **Parameter estimation:** you may directly change the W and obtain different estimates. The difference can be substantial.

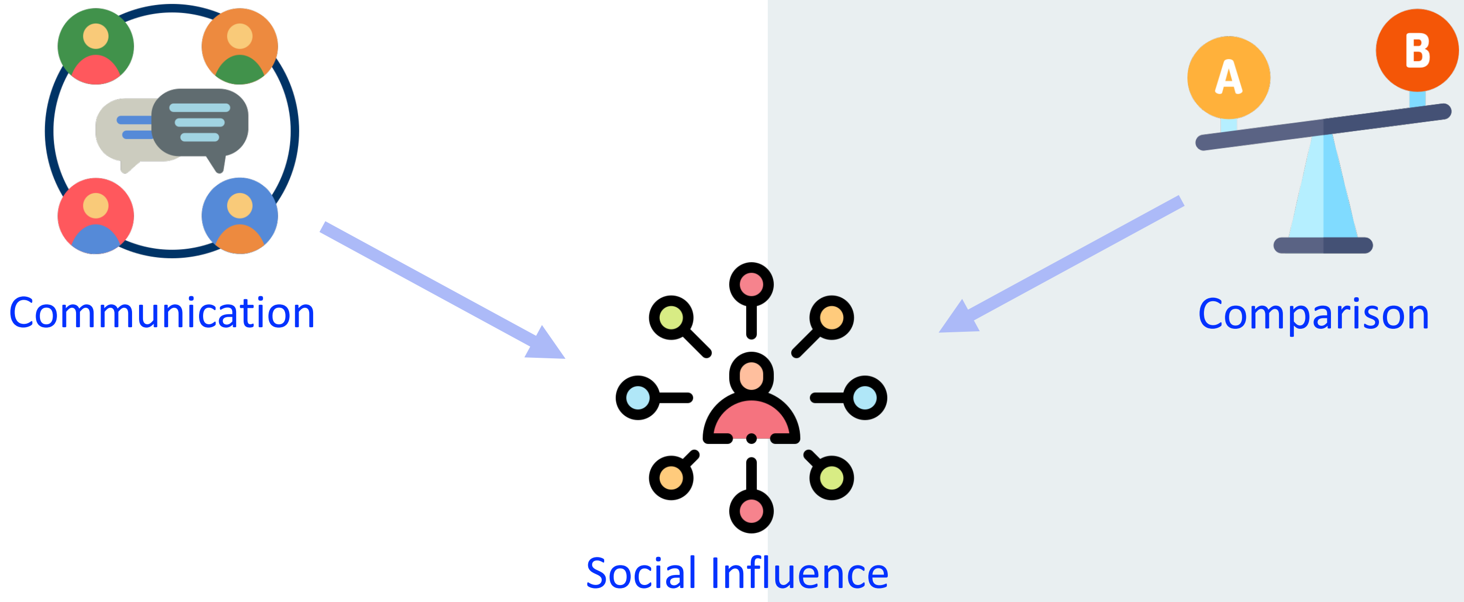
– **Hypothesis test:** you may directly change W and draw a different conclusion

– **Theories of social influence test:** you may directly change one's theory and change W

Theories of Social Influence

- Social influence transpires when an individual adjusts their behavior, attitude, or belief to align with those of others within the social framework.
- In network analysis, social influence is perceived as a dyadic phenomenon: one individual adapts their behavior to match that of another, fostering behavior similarity in their actions.
- The term "contagion" is commonly employed to characterize these processes of social influence.

Theories of Social Influence



– **Communication:** actors use actors with whom they are directly tied as their frame of reference.

– **Comparison:** actors use actors they feel similar to as their frame of reference

Communication vs Comparison

- In the **communication** aspect of contagion, **individuals exchange information** regarding a particular issue, express uncertainties, share past experiences, and learn from each other's errors. This process results in **a convergence of opinions**, attitudes, and beliefs, though not necessarily conformity in behavior.
- On the contrary, **comparison**, which can occur among individuals not directly connected (**non-adjacent actors**), relies explicitly on role modeling and the emulation of behavior.

Communication vs Comparison

- In brief, **communication** tends to foster **similarity in beliefs** without necessarily impacting behavior, while **comparison** tends to induce **similarity in behavior** without necessarily altering underlying beliefs.
- The significance of this disparity lies in the fact that while a researcher can easily observe **similarity in behavior**, detecting **similarity in beliefs** and attitudes is more challenging.

Social Infuence

- The operationalization of a communication process is via **cohesion**.
  - Cohesion encompasses factors such as the quantity, length, and strength of connections between actors, giving rise to notions like cliques, k-plexes, and k-cores.
  - It's examined whether actors belonging to the same subgroup (cliques, k-plexes, and k-cores) exhibit similar behaviors.

Social Influence

- Comparison is commonly operationalized using the notion of **equivalence**. Equivalently positioned actors share similar network embeddings.
  - Actors are deemed structurally equivalent if their ties to and from all actors are identical. They **may not be directly connected** and might not even be aware of each other's existence.

Communication vs Comparison

- The communication perspective links similarity to the direct connections between actors:
  - When adjacent actors are seen to be or become alike, this is attributed to communication.
- On the other hand, the comparison viewpoint links similarity to the manner in which actors are situated within the social structure:
  - If equivalent actors with similar embeddings are observed to be or become alike, this is attributed to comparison.

Communication vs Comparison

– An endeavor is **undertaken to contrast communication with comparison** by **initially assessing equivalence while controlling for cohesion**.

– The rationale behind this approach is that cohesion encompasses communication effects, thus by isolating equivalence, only comparison effects are estimated.

Notice: This is incorrect!

Communication vs Comparison

- **Cohesion** will continue to encapsulate both **direct and indirect communication**, as well as **comparison effects** among **adjacent actors**.
- **Equivalence** will encapsulate *comparison effects and indirect communication among non-adjacent actors*.
- It may be theoretically feasible to argue for one over the other, *empirically testing their relative impact is challenging*.

Adjacency vs Non-adjacent

– Differentiate between contagion among directly connected actors and contagion among actors without direct ties.

Direct Tie

Indirect Tie

Social Influence

Adjacency vs Non-adjacent

- Contagion among **directly connected actors** encompasses both **direct and indirect communication and comparison effects**.
- Contagion among actors **lacking direct ties** encompasses **direct and indirect comparison as well as indirect communication effects**.
- As the ***sociometric distance between actors increases***, the influence of ***indirect communication and comparison diminishes***, thereby **amplifying the direct comparison component**.

Social Influence: among adjacent and non-adjacent

Adjacent Actors	Non-adjacent Actors
Communication Indirect Communication	Indirect Communication
Comparison Indirect Comparison	Comparison Indirect Comparison

Direct
Communcation

– The differentiation between contagion via direct and indirect connections doesn't fully resolve the debate between communication and comparison.

– It does **segregate the impacts of contagion via direct communication**, which represents one of the primary theoretical explanations for contagion.

Operationalizations for Contagion

- Contagion among **adjacent actors** is simply operationalized by examining whether they tend to exhibit similar opinions or behaviors.
- For contagion among **non-adjacent actors**, two distinct operational approaches are employed:
  - The first approach involves assessing **equivalence among unconnected actors**.
  - The second approach involves examining **the quantity of paths of different lengths between actors**.

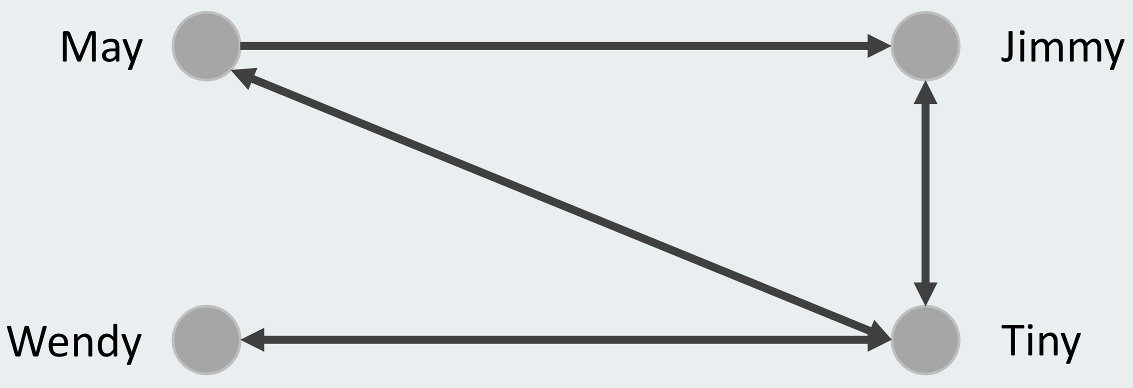
Operationalization of Weighting Matrix W

- **Steps**
  - Row versus column normalization
  - Influence between adjacent actors: cohesion
  - Influence between non-adjacent actors: equivalence
  - Influence between non-adjacent actors: indirect paths
  - Other specifications

Row versus Column Normalization

j

		May	Jimmy	Tiny	Wendy
	May	---	1	1	0
i	**Jimmy**	0	---	1	0
	Tiny	1	1	---	1
	Wendy	0	0	1	---



Row Normalization

$$w_{ij}^{[1]} = \frac{a_{ij}}{a_{i.}}$$

If actor i has **three outgoing ties**, each of his alters will have weight **1/3**.
An actor with only one outgoing tie will be fully influenced by this one alter.

Column Normalization

$$w_{ij}^{[2]} = \frac{a_{ij}}{a_{.j}}$$

The strength of influence actor j has over actor i now depends on the number of actors influenced by j.

Row versus Column Normalization

Row Normalization	Column Normalization
Each **outgoing** contact has equal influence for each actor	Each **ingoing** onact has equal influence for each other
Weight proportional to out degree	Weight proportional to indegree
Total amount of accepted influence equal for all actors	Total amount of exerted influence equal for all actors
Deals with accepted/ received influence	Deals with exerted/exected influence

Influence between Adjacent Actors: Cohesion

Cohesion

$$w_{ij}^{[3]} = \frac{a_{ij}}{a_{i.} + 1}, 1\ is\ the\ resistance$$

Weighting (Resource of an actor)

$$w_{ij}^{[4]} = \frac{r_j a_{ij}}{r_i + \sum_{k=1, k\neq i}^{g} r_k a_{ik}}$$

Influence between Adjacent Actors: Equivalence

$$w_{ij}^{[6]} = \frac{l_{ji}}{1 - l_{ii}},$$

$$where\ l_{ij} = \frac{\left(d_j - d_{ij}\right)^v}{\sum_{q=1}^{g}\left(d_j - d_{qj}\right)^v}$$

v: a constant

w_{ii}: is set to zero and $d_j = \max_i\left(d_{ij}\right)$

- Burt and Doreian utilized weight to depict the transformation of scientists' expressed interests in a scientific journal into a journal norm.
- This norm signifies the expected level of interest in the journal by scientists due to their **structural position** within "**invisible colleges**." It quantifies the degree to which actor j is perceived by actor i as their sole structural peer within the college.

Influence between Non-adjacent actors: Indirect Paths

- To model influence between **non-adjacent actors**, the analysis begins with exploring the **indirect connections** between them.
- **Contagion** is subsequently perceived to propagate from alter to ego through interactions with **third actors**.
- The adjacency matrix is subjected to squaring or cubing, following which **entries corresponding to adjacent actors are nullified**.

Other specifications

- For the study of policy and power networks,

$$w_{ij}^{[7]} = \frac{r_{i(j)}^{\alpha}}{d_{ij}^{\beta}}$$

r_i: as (political) resources available to actor i.

d_{ij}: represents how difficult it is for j to claim i's resources

Statistical Tests on the Weight Structure

- The researcher has narrowed down the potential influence structures to one and seeks to test their theory, as embodied by a specific weight matrix, against alternative influence structures denoted by different $\boldsymbol{W}$ matrices.
- **Scenario 1**: The researcher regards **one structure as the null hypothesis** and aims to compare it against alternative structures.
- **Scenario 2**: The researcher entertains **multiple influence structures** that are equally plausible from a theoretical perspective. The ultimate selection among them relies on statistical analysis.

Situation 1: When a Null Hypothesis is Available

– Formula

$$H_0: y = \rho_0 W_0 y + X_0 \beta_0 + \varepsilon_0$$
$$H_1: y = \rho_1 W_1 y + X_1 \beta_1 + \varepsilon_1$$

– Augmented regressions

$$y = (1 - \alpha)(\rho_0 W_0 y + X_0 \beta_0) + \alpha\left(\widehat{\rho_1} W_1 y + X_1 \widehat{\beta_1}\right) + \upsilon$$

– If H_0 is true, then the true value of nesting parameter α is equal to 0.

Situation 2: When No Hypothesis is Available

– Akaike information criterion (AIC)

$$AIC = -2L\left(\hat{\rho}, \hat{\beta}\right) + q(k)$$

Where $L\left(\hat{\rho}, \hat{\beta}\right)$ is the log-likelihood at the maximum and $q(k)$ is a penalty function of the number of unknown parameters in the model.

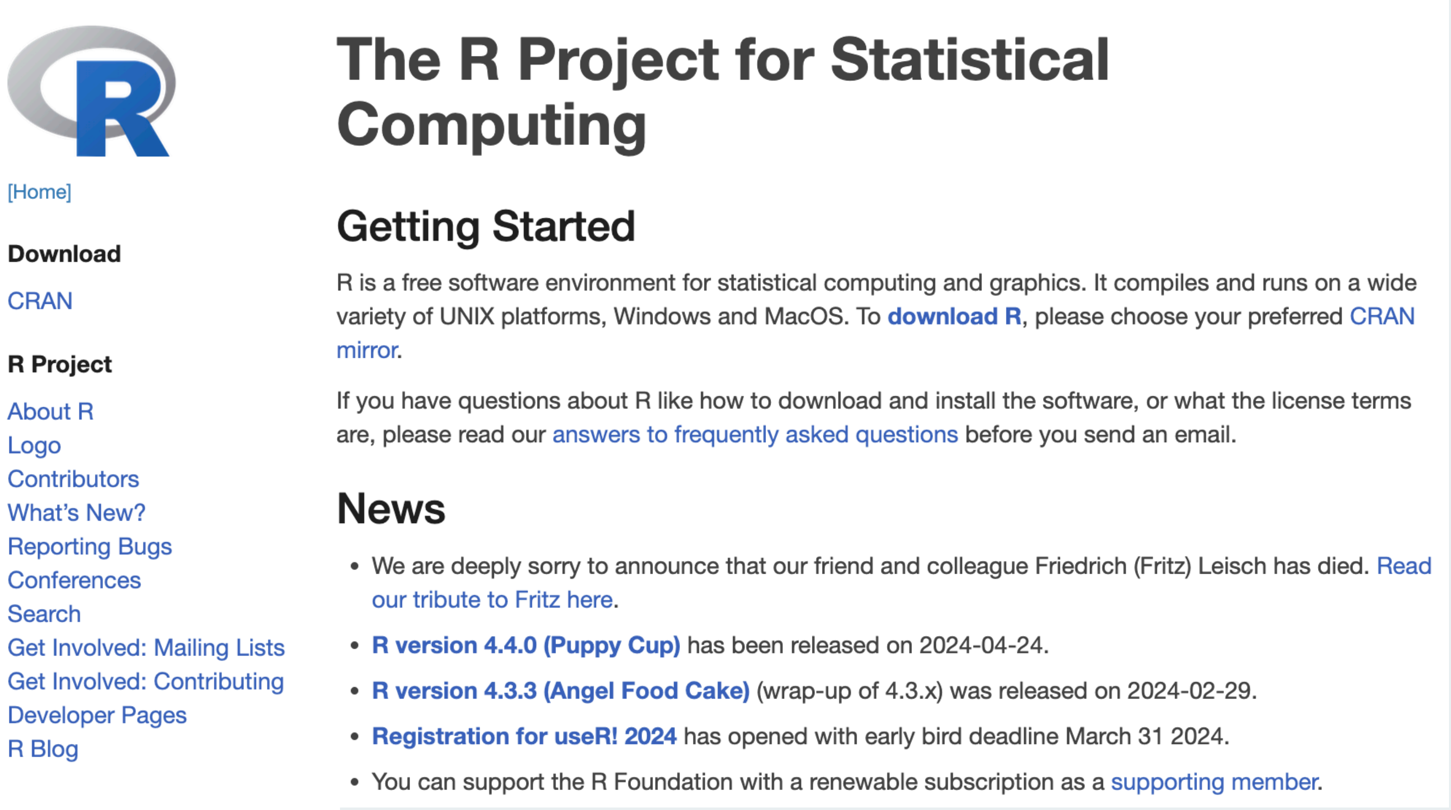
By comparing the AIC values across all models, the model with the lowest AIC indicates the best fit.

Summary

- **The Mechanism of Influence:** Communication, Comparison, Adjacency, or Non-Adjacency.
- **The Source of Influence:** Do actors replicate alter's behavior or opinion directly (network effect), or do they mirror the adaptation of alter's behavior or opinion? (network disturbance)
- **Normalization:** Whether actors have a restricted capacity to influence others (row normalization) or a limited readiness to accept influence (column normalization).

R: Installation

- Download R and Install: https://www.r-project.org/
- Click ”**download R**”
- Select your OS


[Home]

Download

CRAN

R Project

About R
Logo
Contributors
What’s New?
Reporting Bugs
Conferences
Search
Get Involved: Mailing Lists
Get Involved: Contributing
Developer Pages
R Blog

The R Project for Statistical Computing

Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To **download R**, please choose your preferred CRAN mirror.

If you have questions about R like how to download and install the software, or what the license terms are, please read our answers to frequently asked questions before you send an email.

News

- We are deeply sorry to announce that our friend and colleague Friedrich (Fritz) Leisch has died. Read our tribute to Fritz here.
- **R version 4.4.0 (Puppy Cup)** has been released on 2024-04-24.
- **R version 4.3.3 (Angel Food Cake)** (wrap-up of 4.3.x) was released on 2024-02-29.
- **Registration for useR! 2024** has opened with early bird deadline March 31 2024.
- You can support the R Foundation with a renewable subscription as a supporting member.

R: Installation

CRAN Mirrors

The Comprehensive R Archive Network is available at the following URLs, please choose a location close to you. Some statistics on the status of the mirrors can be found here: main page, windows release, windows old release.

If you want to host a new mirror at your institution, please have a look at the CRAN Mirror HOWTO.

0-Cloud	
https://cloud.r-project.org/	Automatic redirection to servers worldwide, currently sponsored by Posit
Argentina	
http://mirror.fcaglp.unlp.edu.ar/CRAN/	Universidad Nacional de La Plata
Australia	
https://cran.csiro.au/	CSIRO
https://mirror.aarnet.edu.au/pub/CRAN/	AARNET
https://cran.ms.unimelb.edu.au/	School of Mathematics and Statistics, University of Melbourne
Austria	
https://cran.wu.ac.at/	Wirtschaftsuniversität Wien
Belgium	
https://www.freestatistics.org/cran/	Patrick Wessa
https://ftp.belnet.be/mirror/CRAN/	Belnet, the Belgian research and education network
Brazil	
https://cran-r.c3sl.ufpr.br/	Universidade Federal do Parana
https://cran.fiocruz.br/	Oswaldo Cruz Foundation, Rio de Janeiro
https://vps.fmvz.usp.br/CRAN/	University of Sao Paulo, Sao Paulo
https://brieger.esalq.usp.br/CRAN/	University of Sao Paulo, Piracicaba
Bulgaria	
https://ftp.uni-sofia.bg/CRAN/	Sofia University
Canada	
https://mirror.rcg.sfu.ca/mirror/CRAN/	Simon Fraser University, Burnaby
https://muug.ca/mirror/cran/	Manitoba Unix User Group
https://mirror.csclub.uwaterloo.ca/CRAN/	University of Waterloo
https://cran.mirror.rafal.ca/	Rafal Rzeczkowski
Taiwan	
https://cran.csie.ntu.edu.tw/	National Taiwan University, Taipei

https://cran.csie.ntu.edu.tw/

R: Installation

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- Download R for Linux (Debian, Fedora/Redhat, Ubuntu)
- Download R for macOS
- Download R for Windows

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

R: Installation

R for macOS

This directory contains binaries for the base distribution and of R and packages to run on macOS. R and package binaries for R versions older than 4.0.0 are only available from the CRAN archive so users of such versions should adjust the CRAN mirror setting (`https://cran-archive.r-project.org`) accordingly.

Note: Although we take precautions when assembling binaries, please use the normal precautions with downloaded executables.

R 4.4.0 "Puppy Cup" released on 2024/04/24

Please check the integrity of the downloaded package by checking the signature:
`pkgutil --check-signature R-4.4.0-arm64.pkg`
in the *Terminal* application. If Apple tools are not avaiable you can check the SHA1 checksum of the downloaded image:
`openssl sha1 R-4.4.0-arm64.pkg`

Latest release:

For Apple silicon (M1-3) Macs:
R-4.4.0-arm64.pkg
SHA1-hash: 45e08d760f10c939b1a341223562bf0ac7051332
(ca. 94MB, notarized and signed)

For older Intel Macs:
R-4.4.0-x86_64.pkg
SHA1-hash: 13c1ae112666425ddc9bbb6327b66d2bcb26eba8
(ca. 96MB, notarized and signed)

R 4.4.0 binary for macOS 11 (**Big Sur**) and higher, signed and notarized packages.

Contains R 4.4.0 framework, R.app GUI 1.80, Tcl/Tk 8.6.12 X11 libraries and Texinfo 6.8. The latter two components are optional and can be ommitted when choosing "custom install", they are only needed if you want to use the `tcltk` R package or build package documentation from sources.

macOS Ventura users: there is a known bug in Ventura preventing installations from some locations without a prompt. If the installation fails, move the downloaded file away from the *Downloads* folder (e.g., to your home or Desktop).

Note: the use of X11 (including `tcltk`) requires XQuartz (version 2.8.5 or later). Always re-install XQuartz when upgrading your macOS to a new major version.

This release uses Xcode 14.2/14.3 and GNU Fortran 12.2. If you wish to compile R packages which contain Fortran code, you may need to download the corresponding GNU Fortran compiler from https://mac.R-project.org/tools. Any external libraries and tools are expected to live in `/opt/R/arm64` (Apple silicon) or `/opt/R/x86_64` (Intel).

Mac OS

R: Installation

R for Windows

Subdirectories:

base	Binaries for base distribution. This is what you want to **install R for the first time**.
contrib	Binaries of contributed CRAN packages (for R >= 4.0.x).
old contrib	Binaries of contributed CRAN packages for outdated versions of R (for R < 4.0.x).
Rtools	Tools to build R and R packages. This is what you want to build your own packages on Windows, or to build R itself.

Please do not submit binaries to CRAN. Package developers might want to contact Uwe Ligges directly in case of questions / suggestions related to Windows binaries.

You may also want to read the R FAQ and R for Windows FAQ.

Note: CRAN does some checks on these binaries for viruses, but cannot give guarantees. Use the normal precautions with downloaded executables.

R Studio: Installation

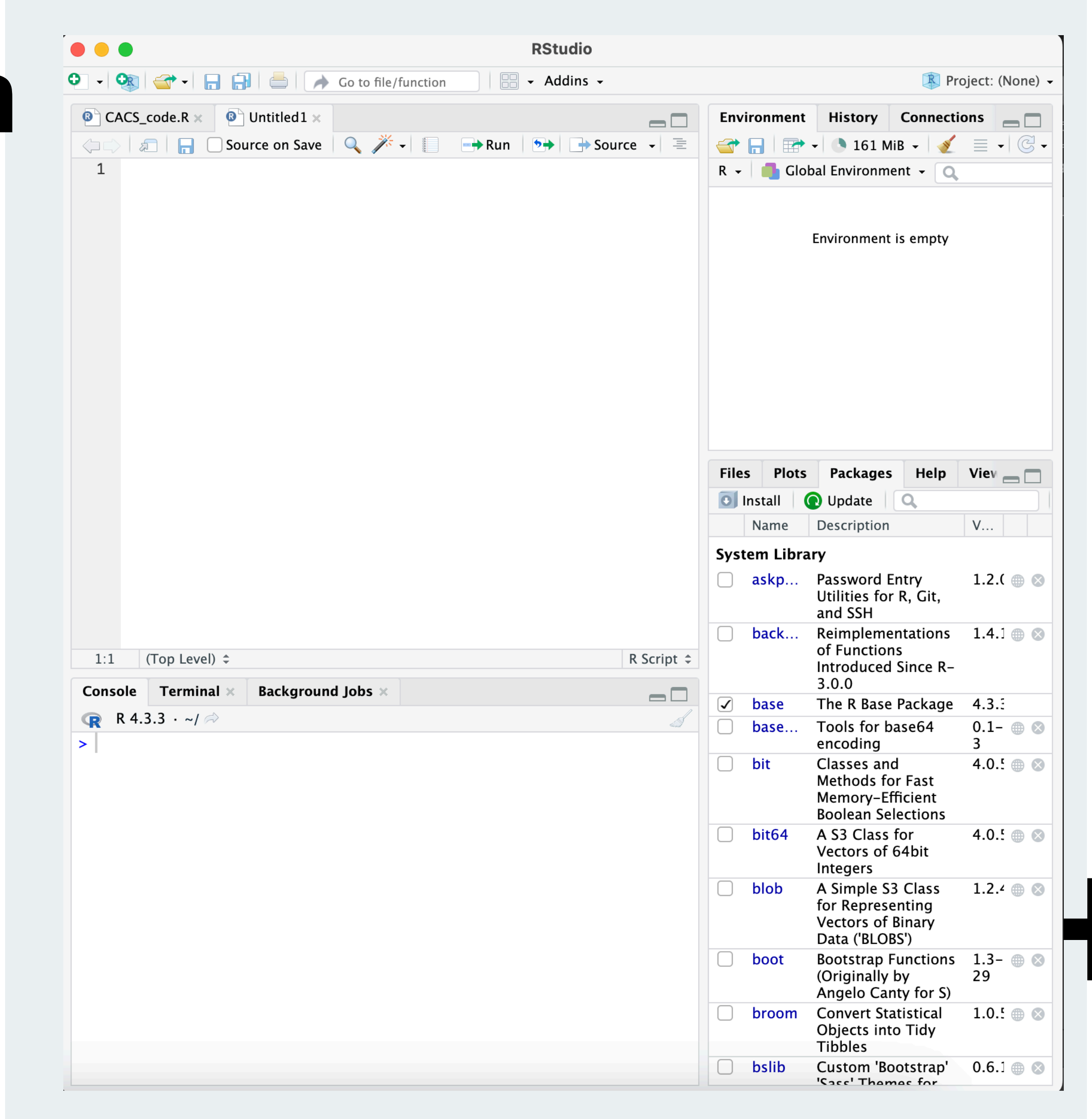
https://posit.co/download/rstudio-desktop/

2: Install RStudio

DOWNLOAD RSTUDIO DESKTOP FOR MACOS 12+

This version of RStudio is only supported on macOS 12 and higher. For earlier macOS environments, please download a previous version.

Size: 566.51 MB | SHA-256: A5EDA699 | Version: 2024.04.1+748 | Released: 2024-05-11



Lab Practice

$$y = \rho_1 \boxed{W_1} y + X\beta + \varepsilon$$
$$\varepsilon = \rho_2 \boxed{W_2} \varepsilon + \nu$$
$$\nu \sim N(0, \sigma^2 I)$$

sna (version 2.7-2)

lnam: Fit a Linear Network Autocorrelation Model

Description

`lnam` is used to fit linear network autocorrelation models. These include standard OLS as a special case, although `lm` is to be preferred for such analyses.

Usage

```
lnam(y, x = NULL, W1 = NULL, W2 = NULL, theta.seed = NULL,
    null.model = c("meanstd", "mean", "std", "none"), method = "BFGS",
    control = list(), tol=1e-10)
```

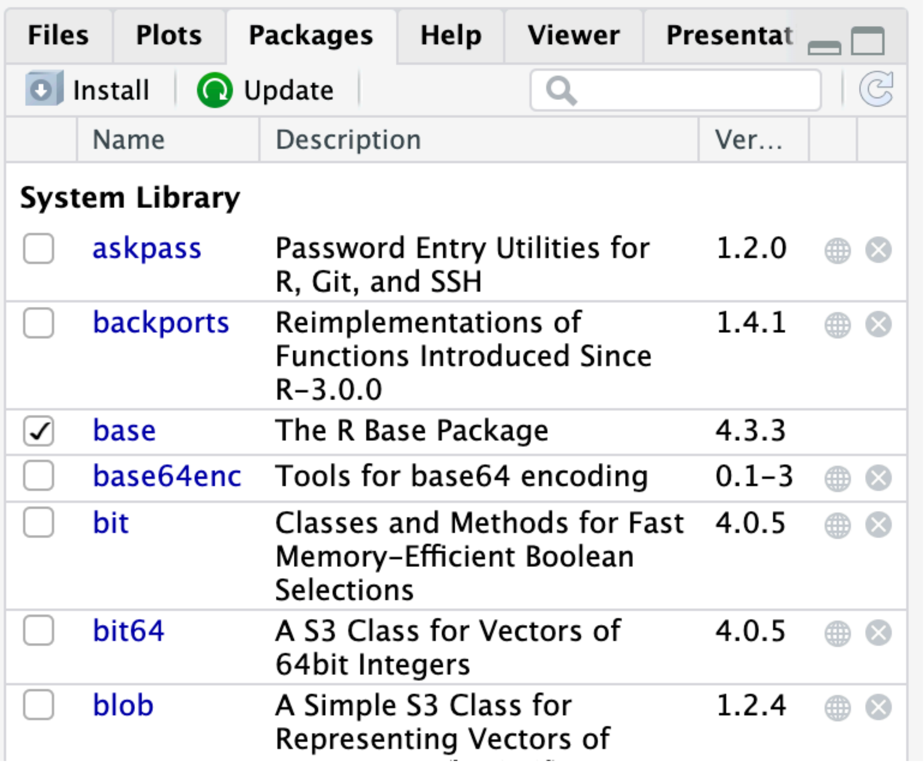

```
lnam(y, x = NULL, W1 = NULL, W2 = NULL, theta.seed = NULL,
    null.model = c("meanstd", "mean", "std", "none"), method = "BFGS",
    control = list(), tol=1e-10)
```

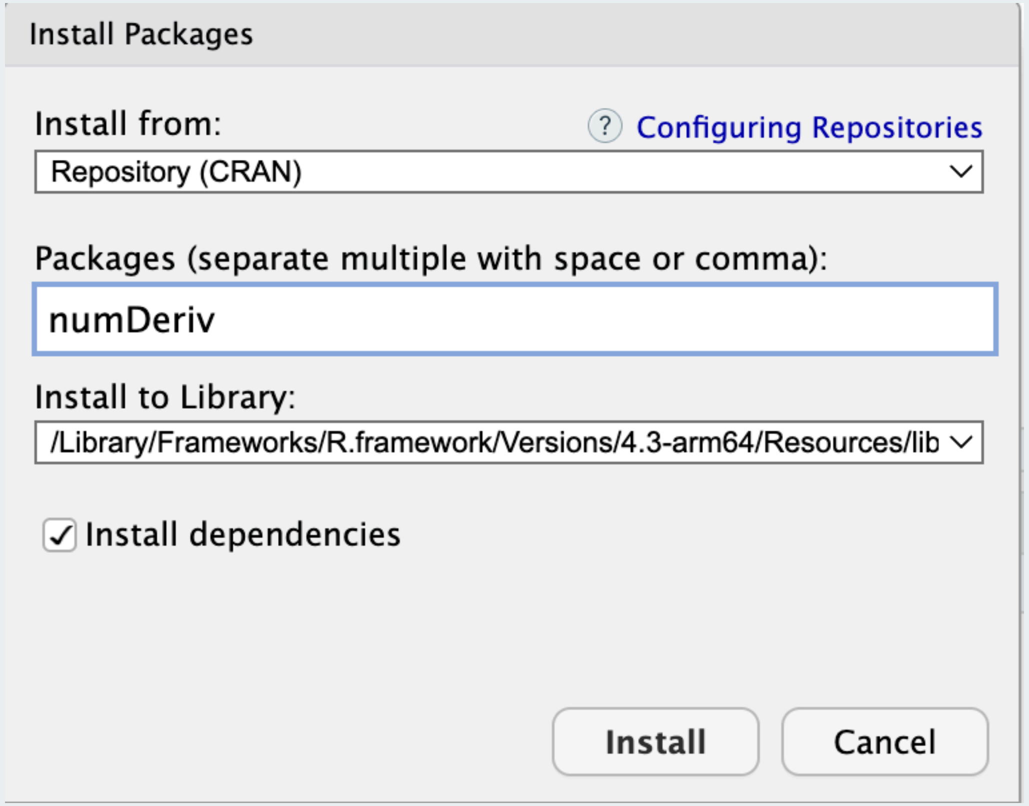

Value

An object of class `"lnam"` containing the following elements:

Lab Practice

– Package installation





Lab Practice

```r
# load libraries
library(reshape)
library(igraph)
library(sna)
library(numDeriv)

# load data
load('~/Documents/python/social_network_analysis/studentnets.peerinfl.rda')
# how does a student (std_id) in the first semester (sem_id) like this subject (sub)?
# show the first 10 rows
head(attitudes, n=10)
```

Semester

Subject Preference (1-5)

Student ID

```
> head(attitudes, n=10)
```

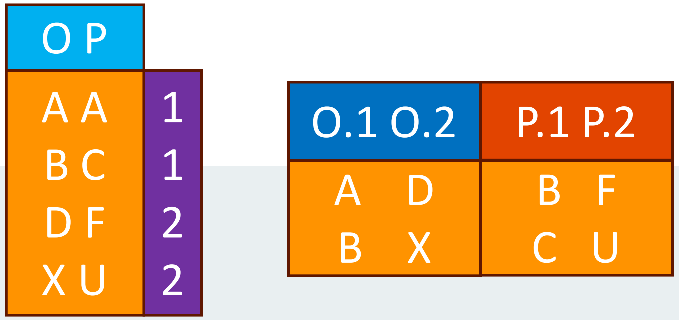
	std_id	sem_id	cls_id	tlks	tlkt	like_c	imp	egrd	jgrd	sub	tot	frn	cmt	tch	voln	call	misb	sanc	sftch	tfstd
1	16144	1	851	4	2	NA	NA	2	-1	2	2	3	2	2	NA	NA	NA	NA	NA	NA
2	16181	1	851	4	2	2	NA	3	0	2	3	4	3	2	NA	NA	NA	NA	NA	NA
3	16247	1	851	4	4	2	NA	4	0	2	2	4	3	2	NA	NA	NA	NA	NA	NA
4	16399	1	851	4	2	3	NA	4	1	3	2	3	1	2	NA	NA	NA	NA	NA	NA
5	15469	1	901	2	2	3	NA	2	0	3	4	4	4	4	NA	NA	NA	NA	NA	NA
6	15947	1	901	3	2	3	NA	3	0	3	4	4	3	4	NA	NA	NA	NA	NA	NA
7	15973	1	901	3	2	3	NA	4	0	3	4	4	3	4	NA	NA	NA	NA	NA	NA
8	16073	1	901	4	3	3	NA	3	0	3	3	4	3	4	NA	NA	NA	NA	NA	NA
9	16177	1	901	2	2	3	NA	3	0	3	3	3	2	4	NA	NA	NA	NA	NA	NA
10	16034	1	901	3	3	4	NA	3	0	3	4	4	3	4	NA	NA	NA	NA	NA	NA

Lab Practice

```r
# the friendship in the first semester
# show the first 10 rows
head(sem1, n=10)
```

```
> head(sem1, n=10)
   std_id alter_id timea ona oca acta nacta wrka nbra otha wkna bfa lva cls_id
2  149824   119516     5   1   0    0     0    0    0    0    0   0   0    251
3  149824   122634     1   1   0    0     0    0    0    0    0   0   0    251
4  149824   114679     1   0   1    0     0    0    0    0    0   0   0    251
7   16868    17142     0   1   0    1     0    0    0    0    0   0   0    851
8   16877    16681     1   0   1    1     0    0    0    0    0   0   0    851
9   16877    16769     1   0   1    1     0    0    0    0    0   0   0    851
10  16877    16851     3   1   0    0     0    0    1    0    0   0   0    851
11  16877    16868     7   0   1    0     0    0    1    0    0   0   0    851
13  16902    16690     0   0   1    0     0    0    0    0    0   0   0    851
14  16902    16868     0   0   1    1     0    0    0    0    0   0   0    851
```

Friendship network: ego is std_id and alter for each ego is alter_id



Lab Practice

```r
# split the semester data into two parts from a long table to a wide table
attitudesw = reshape(attitudes, idvar="std_id", timevar="sem_id", direction="wide")
# show the first 10 rows
head(attitudesw, n=10)
```

```
> head(attitudesw, n=10)
```

	std_id	cls_id.1	tlks.1	tlkt.1	like_c.1	imp.1	egrd.1	jgrd.1	sub.1	tot.1	frn.1	cmt.1	tch.1	voln.1	call.1
1	16144	851	4	2	NA	NA	2	-1	2	2	3	2	2	NA	NA
2	16181	851	4	2	2	NA	3	0	2	3	4	3	2	NA	NA
3	16247	851	4	4	2	NA	4	0	2	2	4	3	2	NA	NA
4	16399	851	4	2	3	NA	4	1	3	2	3	1	2	NA	NA
5	15469	901	2	2	3	NA	2	0	3	4	4	4	4	NA	NA
6	15947	901	3	2	3	NA	3	0	3	4	4	3	4	NA	NA
7	15973	901	3	2	3	NA	4	0	3	4	4	3	4	NA	NA
8	16073	901	4	3	3	NA	3	0	3	3	4	3	4	NA	NA
9	16177	901	2	2	3	NA	3	0	3	3	3	2	4	NA	NA
10	16034	901	3	3	4	NA	3	0	3	4	4	3	4	NA	NA

	misb.1	sanc.1	sftch.1	tfstd.1	tfros.1	chal.1	prev.1	cls_id.2	tlks.2	tlkt.2	like_c.2	imp.2	egrd.2	jgrd.2
1	NA	NA	NA	NA	NA	NA	0	851	4	2	3	3	1	-1
2	NA	NA	NA	NA	NA	3	0	851	3	2	2	2	3	1
3	NA	NA	NA	NA	NA	2	0	726	4	1	3	3	4	1
4	NA	NA	NA	NA	NA	2	0	851	4	2	2	3	2	-1
5	NA	NA	NA	NA	NA	1	0	NA	NA	NA	NA	NA	NA	NA
6	NA	NA	NA	NA	NA	3	0	NA	NA	NA	NA	NA	NA	NA
7	NA	NA	NA	NA	NA	3	0	NA	NA	NA	NA	NA	NA	NA
8	NA	NA	NA	NA	NA	3	0	692	2	3	2	3	2	0
9	NA	NA	NA	NA	NA	3	0	692	3	2	2	3	3	0
10	NA	NA	NA	NA	NA	3	1	NA	NA	NA	NA	NA	NA	NA

	sub.2	tot.2	frn.2	cmt.2	tch.2	voln.2	call.2	misb.2	sanc.2	sftch.2	tfstd.2	tfros.2	chal.2	prev.2
1	2	1	2	1	2	1	1	1	1	0	0	1	NA	NA
2	1	3	3	2	2	3	2	1	1	0	0	0	NA	NA
3	4	2	4	3	3	2	2	3	1	2	2	2	NA	NA
4	3	1	2	2	1	1	1	1	1	0	-1	0	NA	NA
5	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
6	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
7	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
8	2	2	4	3	2	2	3	2	2	0	1	1	NA	NA
9	2	3	4	3	2	1	2	1	1	0	1	1	NA	NA
10	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA

Lab Practice

```r
# calculate peer influences
# create an empty array
attitudesw$mfrsub.1 = numeric(length(attitudesw$sub.1))
# calculate peer influence for each student
for(i in 1:length(attitudesw$std_id)){
  # first get alters of student i:
  altrs = sem1$alter_id[sem1$std_id == attitudesw$std_id[i]]
  # then get alters' attitudes
  altatts = attitudesw$sub.1[attitudesw$std_id %in% altrs]
  # now count how many friends like the class more than "3"
  attitudesw$nfrsubgt3[i] = length(which(altatts > 3))
  # then take the mean, ignoring NAs:
  attitudesw$mfrsub.1[i] = mean(altatts, na.rm = TRUE)}
# show the first 10 rows
head(attitudesw, n=10)
```

```
> head(attitudesw, n=10)
   std_id cls_id.1 tlks.1 tlkt.1 like_c.1 imp.1 egrd.1 jgrd.1 sub.1 tot.1 frn.1 cmt.1 tch.1 voln.1 call.1
1   16144      851      4      2       NA    NA      2     -1     2     2     3     2     2     NA     NA
2   16181      851      4      2        2    NA      3      0     2     3     4     3     2     NA     NA
3   16247      851      4      4        2    NA      4      0     2     2     4     3     2     NA     NA
4   16399      851      4      2        3    NA      4      1     3     2     3     1     2     NA     NA
5   15469      901      2      2        3    NA      2      0     3     4     4     4     4     NA     NA
6   15947      901      3      2        3    NA      3      0     3     4     4     3     4     NA     NA
7   15973      901      3      2        3    NA      4      0     3     4     4     3     4     NA     NA
8   16073      901      4      3        3    NA      3      0     3     3     4     3     4     NA     NA
9   16177      901      2      2        3    NA      3      0     3     3     3     2     4     NA     NA
10  16034      901      3      3        4    NA      3      0     3     4     4     3     4     NA     NA
   misb.1 sanc.1 sftch.1 tfstd.1 tfros.1 chal.1 prev.1 cls_id.2 tlks.2 tlkt.2 like_c.2 imp.2 egrd.2 jgrd.2
1      NA     NA      NA      NA      NA     NA      0      851      4      2        3     3      1     -1
2      NA     NA      NA      NA      NA      3      0      851      3      2        2     2      3      1
3      NA     NA      NA      NA      NA      2      0      726      4      1        3     3      4      1
4      NA     NA      NA      NA      NA      2      0      851      4      2        2     3      2     -1
5      NA     NA      NA      NA      NA      1      0       NA     NA     NA       NA    NA     NA     NA
6      NA     NA      NA      NA      NA      3      0       NA     NA     NA       NA    NA     NA     NA
7      NA     NA      NA      NA      NA      3      0       NA     NA     NA       NA    NA     NA     NA
8      NA     NA      NA      NA      NA      3      0      692      2      3        2     3      2      0
9      NA     NA      NA      NA      NA      3      0      692      3      2        2     3      3      0
10     NA     NA      NA      NA      NA      3      1       NA     NA     NA       NA    NA     NA     NA
   sub.2 tot.2 frn.2 cmt.2 tch.2 voln.2 call.2 misb.2 sanc.2 sftch.2 tfstd.2 tfros.2 chal.2 prev.2 mfrsub.1
1      2     1     2     1     2      1      1      1      1       0       0       1     NA     NA 2.400000
2      1     3     3     2     2      3      2      1      1       0       0       0     NA     NA 2.916667
3      4     2     4     3     3      2      2      3      1       2       2       2     NA     NA 2.600000
4      3     1     2     2     1      1      1      1      1       0      -1       0     NA     NA      NaN
5     NA    NA    NA    NA    NA     NA     NA     NA     NA      NA      NA      NA     NA     NA 3.000000
6     NA    NA    NA    NA    NA     NA     NA     NA     NA      NA      NA      NA     NA     NA 3.000000
7     NA    NA    NA    NA    NA     NA     NA     NA     NA      NA      NA      NA     NA     NA 3.333333
8      2     2     4     3     2      2      3      2      2       0       1       1     NA     NA 2.428571
9      2     3     4     3     2      1      2      1      1       0       1       1     NA     NA 3.125000
10    NA    NA    NA    NA    NA     NA     NA     NA     NA      NA      NA      NA     NA     NA 3.500000
   nfrsubgt3
1          0
2          4
3          1
4          0
5          0
6          2
7          1
8          0
9          2
10         1
```

Lab Practice

```r
# delete all na values and obtain the average of altatts
attitudesw$mfrsub.1[i] = mean(altatts, na.rm = TRUE)
# show the first 10 rows
head(attitudesw, n=10)

# transform from a dataframe to a matrix
sem1mat <- as.matrix(sem1[1:2])

# transform from a integer to a character
sem1mat[,1]=as.character(sem1mat[,1])
sem1mat[,2]=as.character(sem1mat[,2])

# transform edgelist to a igraph's graph
sem1graph <- igraph::graph_from_edgelist(sem1mat)
# obtain adjacent matrix
sem1matrix = igraph::get.adjacency(sem1graph)
# as a matrix
sem1matrix = as.matrix(sem1matrix)
# extract the dataframe with conditions
attitudesw = attitudesw[match(row.names(sem1matrix), attitudesw$std_id),]
# drop na values
atts = attitudesw[!is.na(attitudesw$sub.2),]
atts = atts[!is.na(atts$sub.1),]
atts = atts[!is.na(atts$mfrsub.1),]
```

1. Do average
2. Integer to Character
3. Edgelist to igraph
4. Drop na

Lab Practice

```r
# define a weight matrix
W = sem1matrix
# make sure the rows and columns in W are in the same order as atts:
W = W[match(atts$std_id,row.names(W)), match(atts$std_id,colnames(W))]
# make sure (this will return 0 if we did it right):
which(rownames(W) != atts$std_id)
# Linear Network Autocorrelation Model.
pim1 <- lnam(atts$sub.2,cbind(atts$sub.1, atts$mfrsub.1), W2 = W)
# show the results
summary(pim1)
```

```
Call:
lnam(y = atts$sub.2, x = cbind(atts$sub.1, atts$mfrsub.1), W2 = W)

Residuals:
    Min      1Q  Median      3Q     Max
-2.4678 -0.4678  0.1391  0.4455  2.0064

Coefficients:
       Estimate Std. Error Z value Pr(>|z|)
X1      0.60696    0.04019  15.103  < 2e-16 ***
X2      0.34667    0.04232   8.191 2.22e-16 ***
rho2.1  0.02887    0.01691   1.707   0.0878 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      Estimate Std. Error
Sigma   0.7247      0.001

Goodness-of-Fit:
        Residual standard error: 0.7322 on 342 degrees of freedom (w/o Sigma)
        Multiple R-Squared: 0.4136, Adjusted R-Squared: 0.4085
        Model log likelihood: -379 on 341 degrees of freedom (w/Sigma)
        AIC: 766.1 BIC: 781.4

        Null model: meanstd
        Null log likelihood: -476.3 on 343 degrees of freedom
        AIC: 956.5 BIC: 964.2
        AIC difference (model versus null): 190.4
        Heuristic Log Bayes Factor (model versus null):  182.8
```

References

- Prof. Tzai-Hung Wen: Tutorials on Network Data Analysis and Models
- McFarland, Daniel, et.al. 2010. "Social Network Analysis Labs in R." Stanford University.

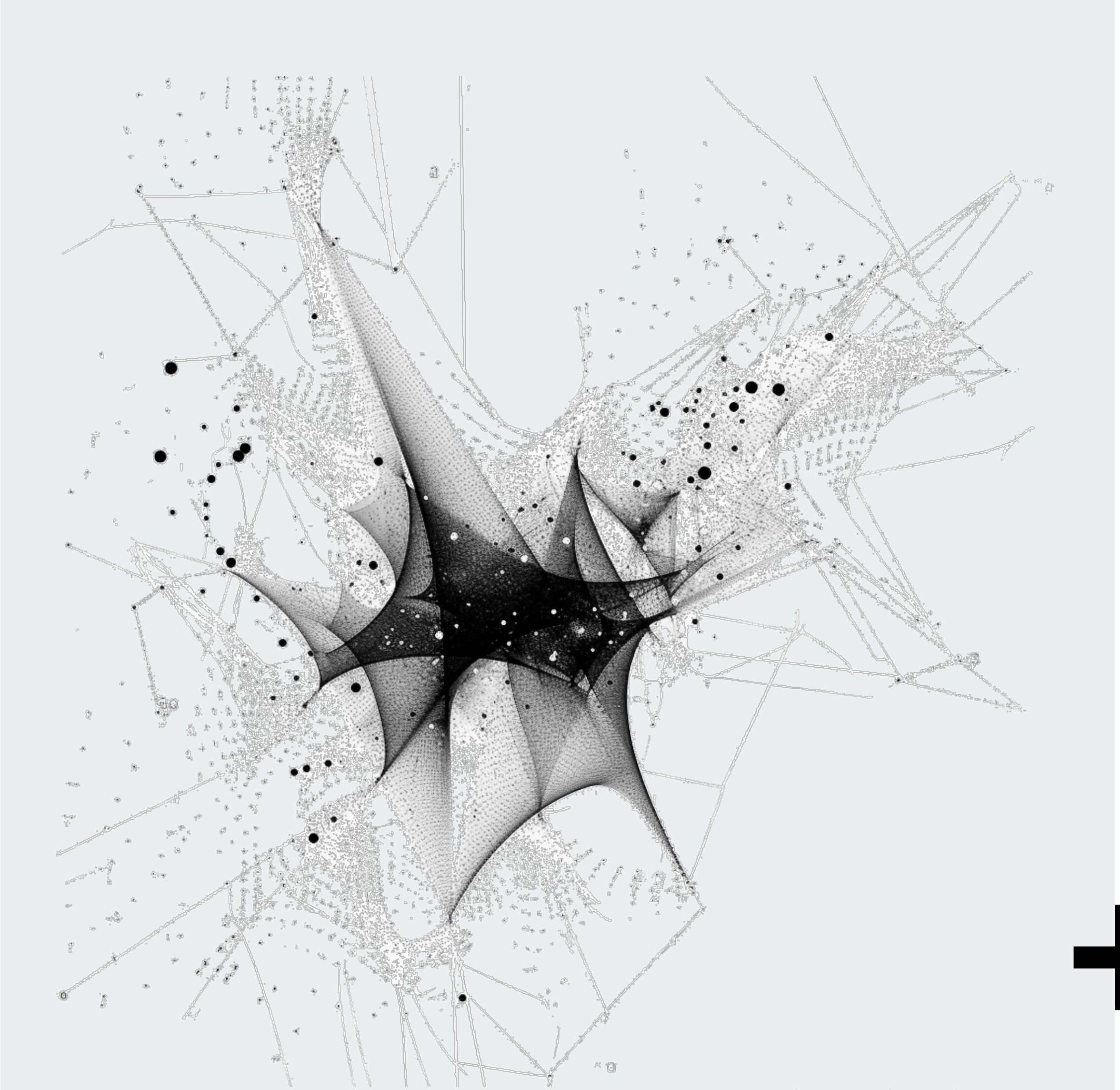


Photo credit: midjourney

The End

Thank you for your attention!

Email: chchan@ntnu.edu.tw
Website: toodou.github.io

Photo credit: midjourney