Network Diffusion of Innovations in R: Introducing **netdiffuseR**

George G. Vega Yon   Thomas W. Valente

Department of Preventive Medicine
University of Southern California

June 29th, 2016

Acknowledgements

netdiffuseR *was created with the support of grant R01 CA157577 from the National Cancer Institute/National Institutes of Health.*

netdiffuseR *has benefited from input provided by participants of the Center for Applied Network Analysis (CANA), and the Computational Social Science Lab (CSSL) at the University of Southern California.*

netdiffuseR*'s original code was developed by* **Thomas Valente***, improved by* **Stephanie Dyal** *and* **Timothy Hayes***, and extended by* **George Vega Yon***.*

Contents

1. Motivation

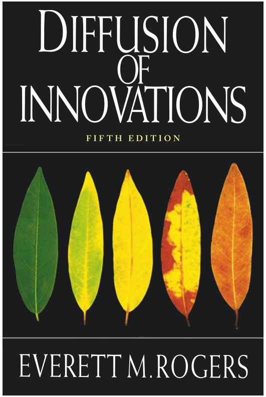
2. **netdiffuseR**: network diffusion of innovations in R

3. **netdiffuseR**: some examples
   - The basics
   - Visualizing Diffusion
   - Simulating Diffusion Process
   - Statistical inference

4. Concluding remarks

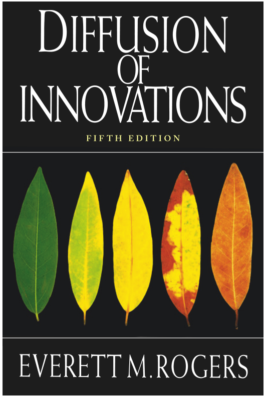
Motivation

Network Diffusion of Innovations

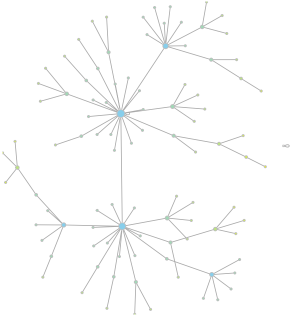


Motivation

Network Diffusion of Innovations

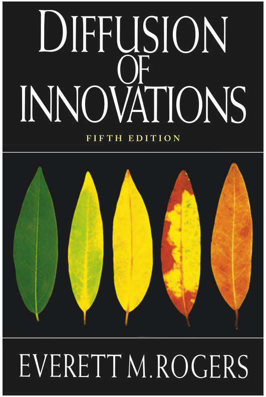


+



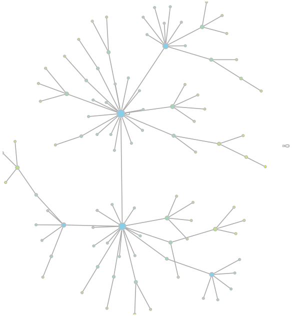
Motivation

Network Diffusion of Innovations

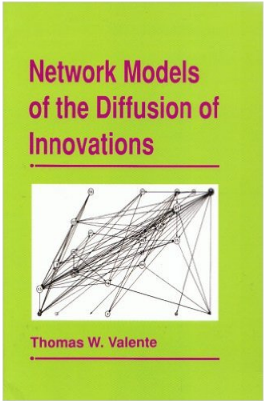


DIFFUSION OF INNOVATIONS
FIFTH EDITION
EVERETT M. ROGERS

+



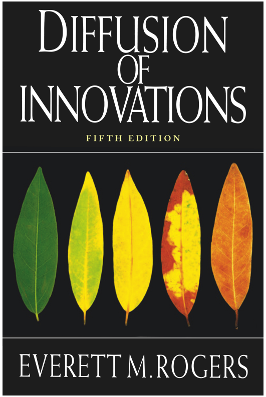
=



Network Models of the Diffusion of Innovations
Thomas W. Valente

Motivation

Network Diffusion of Innovations

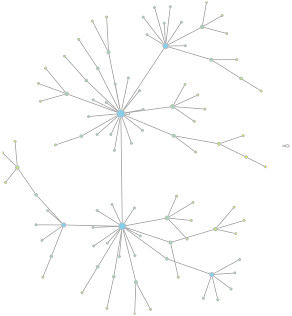


DIFFUSION OF INNOVATIONS

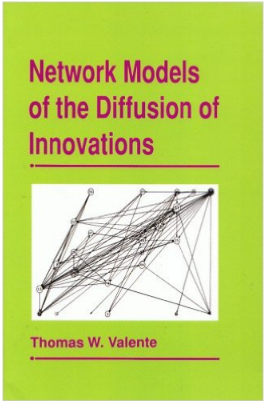
FIFTH EDITION

EVERETT M. ROGERS

+



=



Network Models of the Diffusion of Innovations

Thomas W. Valente

So what is Network Diffusion of Innovations anyway?

Motivation

Network Diffusion of Innovations

- Tries to explain how new ideas and practices (innovations) spread within and between communities.

Motivation

Network Diffusion of Innovations

- Tries to explain how new ideas and practices (innovations) spread within and between communities.
- While a lot of factors have been shown to influence diffusion (Spatial, Economic, Cultural, Biological, etc.), Social Networks is a prominent one.

Motivation

Network Diffusion of Innovations

- Tries to explain how new ideas and practices (innovations) spread within and between communities.
- While a lot of factors have been shown to influence diffusion (Spatial, Economic, Cultural, Biological, etc.), Social Networks is a prominent one.
- More complex than *contagion* $\Longrightarrow$ a single tie is no longer enough for an innovation to spread across a social system.

Motivation

Network Diffusion of Innovations

- Tries to explain how new ideas and practices (innovations) spread within and between communities.
- While a lot of factors have been shown to influence diffusion (Spatial, Economic, Cultural, Biological, etc.), Social Networks is a prominent one.
- More complex than *contagion* ⟹ a single tie is no longer enough for an innovation to spread across a social system.
- We think of this in terms of adoption thresholds and social exposure.

Motivation

Network Diffusion of Innovations

The basic idea

- Network thresholds (Valente, 1995), τ, are defined as the required proportion or number of neighbors that leads you to adopt a particular behavior (innovation), $a = 1$. In (very) general terms

Motivation

Network Diffusion of Innovations

The basic idea

- Network thresholds (Valente, 1995), τ, are defined as the required proportion or number of neighbors that leads you to adopt a particular behavior (innovation), $a = 1$. In (very) general terms

$$a_i = \begin{cases} 1 & \text{if } \tau_i \leq E_i \\ 0 & \text{Otherwise} \end{cases} \qquad E_i \equiv \frac{\sum_{j\neq i} \mathbf{X}_{ij} a_j}{\sum_{j\neq i} \mathbf{X}_{ij}} \tag{1}$$

Where E_i is i's exposure to the innovation and $\mathbf{X}$ is the adjacency matrix (the network).

Motivation

Network Diffusion of Innovations

The basic idea

- Network thresholds (Valente, 1995), τ, are defined as the required proportion or number of neighbors that leads you to adopt a particular behavior (innovation), $a = 1$. In (very) general terms

$$a_i = \begin{cases} 1 & \text{if } \tau_i \leq E_i \\ 0 & \text{Otherwise} \end{cases} \qquad E_i \equiv \frac{\sum_{j\neq i} \mathbf{X}_{ij} a_j}{\sum_{j\neq i} \mathbf{X}_{ij}} \tag{1}$$

  Where E_i is i’s exposure to the innovation and $\mathbf{X}$ is the adjacency matrix (the network).

- This can be generalized and extended to include covariates and other weighting schemes (that’s what **netdiffuseR** is all about).

Motivation

Network Diffusion of Innovations

The basic idea

- Network thresholds (Valente, 1995), τ, are defined as the required proportion or number of neighbors that leads you to adopt a particular behavior (innovation), $a = 1$. In (very) general terms

$$a_i = \begin{cases} 1 & \text{if } \tau_i \leq E_i \\ 0 & \text{Otherwise} \end{cases} \qquad E_i \equiv \frac{\sum_{j\neq i} \mathbf{X}_{ij} a_j}{\sum_{j\neq i} \mathbf{X}_{ij}} \tag{1}$$

  Where E_i is i's exposure to the innovation and $\mathbf{X}$ is the adjacency matrix (the network).

- This can be generalized and extended to include covariates and other weighting schemes (that's what **netdiffuseR** is all about).

So now, what is **netdiffuseR**?

netdiffuseR: network diffusion of innovations in R

netdiffuseR is an **R** package that

netdiffuseR: network diffusion of innovations in R

netdiffuseR is an **R** package that

- Is designed for Visualizing, Analyzing and Simulating network diffusion data (in general).

netdiffuseR: network diffusion of innovations in R

netdiffuseR is an **R** package that

- Is designed for Visualizing, Analyzing and Simulating network diffusion data (in general).
- Depends on some pretty popular packages:

netdiffuseR: network diffusion of innovations in R

netdiffuseR is an **R** package that

- Is designed for Visualizing, Analyzing and Simulating network diffusion data (in general).
- Depends on some pretty popular packages:
  - **RcppArmadillo**: So it’s fast,

netdiffuseR: network diffusion of innovations in R

netdiffuseR is an **R** package that

- Is designed for Visualizing, Analyzing and Simulating network diffusion data (in general).
- Depends on some pretty popular packages:
  - **RcppArmadillo**: So it’s fast,
  - **Matrix**: So it’s big,

netdiffuseR: network diffusion of innovations in R

netdiffuseR is an **R** package that

- Is designed for Visualizing, Analyzing and Simulating network diffusion data (in general).
- Depends on some pretty popular packages:
  - **RcppArmadillo**: So it's fast,
  - **Matrix**: So it's big,
  - **statnet** and **igraph**: So it's not from scratch

netdiffuseR: network diffusion of innovations in R

netdiffuseR is an **R** package that

- Is designed for Visualizing, Analyzing and Simulating network diffusion data (in general).
- Depends on some pretty popular packages:
  - **RcppArmadillo**: So it's fast,
  - **Matrix**: So it's big,
  - **statnet** and **igraph**: So it's not from scratch
- It's roughly 14,000 lines of code (`R/` + `src/` + `tests/`), 100 pages of manual, and 4 vignettes,

netdiffuseR: network diffusion of innovations in R

netdiffuseR is an **R** package that

- Is designed for Visualizing, Analyzing and Simulating network diffusion data (in general).
- Depends on some pretty popular packages:
    - **RcppArmadillo**: So it's fast,
    - **Matrix**: So it's big,
    - **statnet** and **igraph**: So it's not from scratch
- It's roughly 14,000 lines of code (`R/` + `src/` + `tests/`), 100 pages of manual, and 4 vignettes,
- Can handle big graphs, more than 4 billion elements adjacency matrix (PR for RcppArmadillo),

netdiffuseR: network diffusion of innovations in R

netdiffuseR is an **R** package that

- Is designed for Visualizing, Analyzing and Simulating network diffusion data (in general).
- Depends on some pretty popular packages:
  - **RcppArmadillo**: So it's fast,
  - **Matrix**: So it's big,
  - **statnet** and **igraph**: So it's not from scratch
- It's roughly 14,000 lines of code (`R/` + `src/` + `tests/`), 100 pages of manual, and 4 vignettes,
- Can handle big graphs, more than 4 billion elements adjacency matrix (PR for RcppArmadillo),
- Already on CRAN (2 iterations) with 700 downloads since its first version, Feb 2016,

netdiffuseR: network diffusion of innovations in R

netdiffuseR is an **R** package that

- Is designed for Visualizing, Analyzing and Simulating network diffusion data (in general).
- Depends on some pretty popular packages:
  - **RcppArmadillo**: So it's fast,
  - **Matrix**: So it's big,
  - **statnet** and **igraph**: So it's not from scratch
- It's roughly 14,000 lines of code (`R/` + `src/` + `tests/`), 100 pages of manual, and 4 vignettes,
- Can handle big graphs, more than 4 billion elements adjacency matrix (PR for RcppArmadillo),
- Already on CRAN (2 iterations) with 700 downloads since its first version, Feb 2016,
- A lot of features to make it easy to read network (dynamic) data, making it a nice companion of other net packages.

netdiffuseR: network diffusion of innovations in R

netdiffuseR distributes classical diffusion of innovations datasets:

	Medical Innovation	**Brazilian Farmers**	**Korean Family Planning**
Country	USA	Brazil	Korea
N Respondents	125 Doctors	692 Farmers	1,047 Women
N Communities	4	11	25
Innovation	Tetracycline	Hybrid Corn Seed	Family Planning
Time for Diffusion	18 Months	20 Years	11 Years
Year Data Collected	1955-1956	1966	1973
Ave. Time to 0.50	6	16	7
Highest Saturation	0.89	0.98	0.83
Lowest Saturation	0.81	0.29	0.44
Citation	Coleman et al (1966)	Rogers et al (1970)	Rogers & Kincaid (1981)

netdiffuseR: network diffusion of innovations in R

netdiffuseR distributes classical diffusion of innovations datasets:

	Medical Innovation	**Brazilian Farmers**	**Korean Family Planning**
Country	USA	Brazil	Korea
N Respondents	125 Doctors	692 Farmers	1,047 Women
N Communities	4	11	25
Innovation	Tetracycline	Hybrid Corn Seed	Family Planning
Time for Diffusion	18 Months	20 Years	11 Years
Year Data Collected	1955-1956	1966	1973
Ave. Time to 0.50	6	16	7
Highest Saturation	0.89	0.98	0.83
Lowest Saturation	0.81	0.29	0.44
Citation	Coleman et al (1966)	Rogers et al (1970)	Rogers & Kincaid (1981)

Very nice. . .

netdiffuseR: network diffusion of innovations in R

netdiffuseR distributes classical diffusion of innovations datasets:

	Medical Innovation	**Brazilian Farmers**	**Korean Family Planning**
Country	USA	Brazil	Korea
N Respondents	125 Doctors	692 Farmers	1,047 Women
N Communities	4	11	25
Innovation	Tetracycline	Hybrid Corn Seed	Family Planning
Time for Diffusion	18 Months	20 Years	11 Years
Year Data Collected	1955-1956	1966	1973
Ave. Time to 0.50	6	16	7
Highest Saturation	0.89	0.98	0.83
Lowest Saturation	0.81	0.29	0.44
Citation	Coleman et al (1966)	Rogers et al (1970)	Rogers & Kincaid (1981)

Very nice. . . But it's better to show it with examples!

The basics: Reading data

```r
library(netdiffuseR)
print(head(fakesurveyDyn))

##   id  toa group net1 net2 net3 age gender
## 1  1 1991     1   NA   NA   NA  30      M
## 2  2 1990     1    3    1   NA  35      F
## 3  3 1991     1   NA    2   NA  31      F
## 4  4 1990     1    6    5   NA  30      M
## 5  5 1991     1    4    4    3  40      F
## 6  1 1991     2    3    4    8  29      F
##                                                  note time
## 1                          First wave: No nominations 1990
## 2                           First wave: Nothing weird 1990
## 3                  First wave: Only nominates in net2 1990
## 4 First wave: Nominates someone who wasn't interview 1990
## 5                   First wave: Nominates 4 two times 1990
## 6                First wave: Only nominates outsiders 1990
```

The basics: Reading data

```r
library(netdiffuseR)
print(head(fakesurveyDyn))

##   id  toa group net1 net2 net3 age gender
## 1  1 1991     1   NA   NA   NA  30      M
## 2  2 1990     1    3    1   NA  35      F
## 3  3 1991     1   NA    2   NA  31      F
## 4  4 1990     1    6    5   NA  30      M
## 5  5 1991     1    4    4    3  40      F
## 6  1 1991     2    3    4    8  29      F
##                                                 note time
## 1                         First wave: No nominations 1990
## 2                          First wave: Nothing weird 1990
## 3                 First wave: Only nominates in net2 1990
## 4 First wave: Nominates someone who wasn't interview 1990
## 5                  First wave: Nominates 4 two times 1990
## 6               First wave: Only nominates outsiders 1990
```

We can use the `survey_to_diffnet` function!

The basics: Reading data

```r
mydiffnet <- survey_to_diffnet(
  fakesurveyDyn,
  idvar     = "id",                     # Id of each vertex
  netvars   = c("net1", "net2", "net3"), # Net nominations (e.g. 'name 3 closest friends')
  toavar    = "toa",                    # Time of Adoption
  groupvar = "group",                   # Group (e.g. neighbor, village, etc.)
  timevar   = "time"                    # Timestamp
  )

# Print method
mydiffnet


## Dynamic network of class -diffnet-
##  # of nodes         : 9 (101, 102, 103, 104, 105, 201, 202, 205, ...)
##  # of time periods  : 2 (1990 - 1991)
##  Type               : directed
##  Final prevalence   : 1.00
##  Static attributes  : -
##  Dynamic attributes : group, net1, net2, net3, age, gender, note, time (8)
```

This is a `diffnet` object.

The basics: `diffnet` objects

`diffnet` are

Lists that contains (among other things):

- A list of `dgCMatrix` objects (sparse matrices),
- A set of `data.frames` with vertex attributes,
- An `integer` vector with times of adoption

The basics: `diffnet` objects

`diffnet` are

Lists that contains (among other things):

- A list of `dgCMatrix` objects (sparse matrices),
- A set of `data.frames` with vertex attributes,
- An `integer` vector with times of adoption

Nice objects with a lot of methods

Common methods

`print, summary, plot`	The usual deal
`[[, [[<-`	Attribute access
`[, [<-`	Adjacency matrix access
`%*%, ^, t, c`	Matrix mult and others
`+, -, *, &, |`	Arithmetic

The basics: `diffnet` objects

`diffnet` are

Lists that contains (among other things):

- A list of `dgCMatrix` objects (sparse matrices),
- A set of `data.frames` with vertex attributes,
- An `integer` vector with times of adoption

Nice objects with a lot of methods

Common methods

`print, summary, plot`	The usual deal
`[[, [[<-`	Attribute access
`[, [<-`	Adjacency matrix access
`%*%, ^, t, c`	Matrix mult and others
`+, -, *, &, |`	Arithmetic

Other special functions

`exposure (threshold)`	Basic net-diff stats
`infection, susceptibility`	Some friends from epi
`struct_equiv`	Burt's (1987) structural equivalence
`struct_test`	We'll see this later. . .

The basics: `diffnet` objects

`diffnet` are

Lists that contains (among other things):

- A list of `dgCMatrix` objects (sparse matrices),
- A set of `data.frames` with vertex attributes,
- An `integer` vector with times of adoption

Nice objects with a lot of methods

Common methods

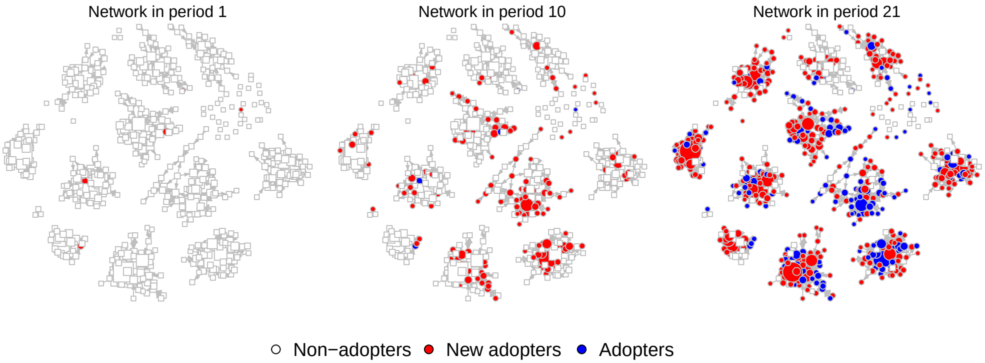
`print, summary, plot`	The usual deal
`[[, [[<-`	Attribute access
`[, [<-`	Adjacency matrix access
`%*%, ^, t, c`	Matrix mult and others
`+, -, *, &, |`	Arithmetic

Other special functions

`exposure (threshold)`	Basic net-diff stats
`infection, susceptibility`	Some friends from epi
`struct_equiv`	Burt's (1987) structural equivalence
`struct_test`	We'll see this later. . .

So, what can we do with `diffnet` objects?

Visualizing Diffusion: `plot_diffnet`



Network in period 1

Network in period 10

Network in period 21

Figure: Diffusion of Hybrid Corn Seed (Brazilian Farmers) - 11 communities/20 years.

Visualizing Diffusion: `plot_diffnet2`

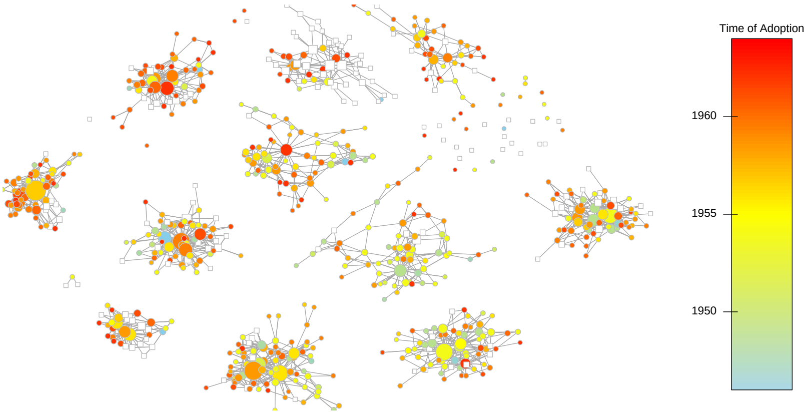


Figure: Diffusion of Hybrid Corn Seed (Brazilian Farmers) - 11 communities/20 years.

Visualizing Diffusion: `plot_infectsuscept`

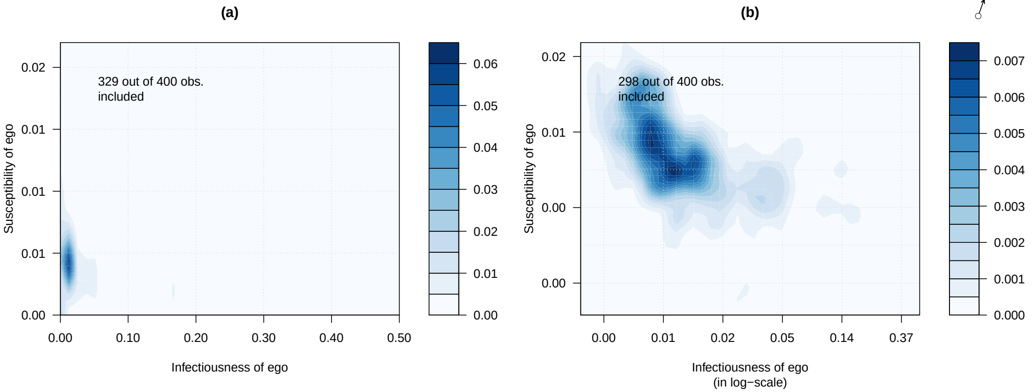


Figure: Joint distribution of Infectiousness and Susceptibility in a random bernoulli diffusion network: (b) shows (a) with log-scale.

Visualizing Diffusion: `plot_threshold` (a classic)

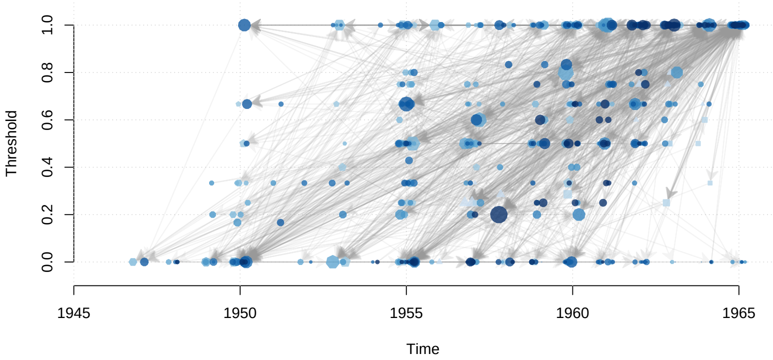


Figure: Adoption thresholds and times of adoption for the Brazilian Farmers: Colors and shapes represent villages. Size is scaled accordingly to each vertex degree.

Visualizing Diffusion: `classify_adopters` (also a classic)

	Very Low Thresh.	Low Thresh.	High Thresh.	Very High Thresh.
Early Adopters	14.04	8.40	0.57	0.29
Early Majority	5.64	11.65	5.54	2.58
Late Majority	1.34	5.06	6.21	2.96
Laggards	1.53	0.00	0.00	34.19

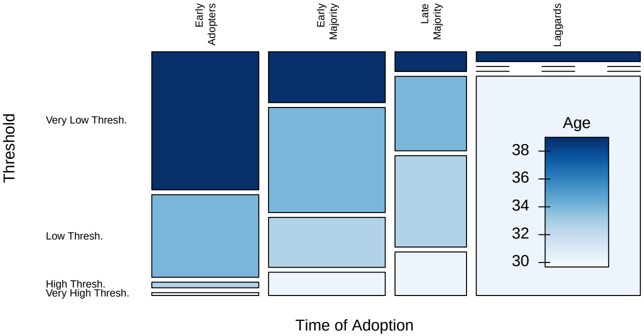


Figure: Adopters classification in the Korean Family data: From the mosaic we can see that in general low threshold levels and early adoption seems to be positively correlated with age.

What about **netdiffuseR** being big??

Visualizing Diffusion: `plot_diffnet2` w/ big graph

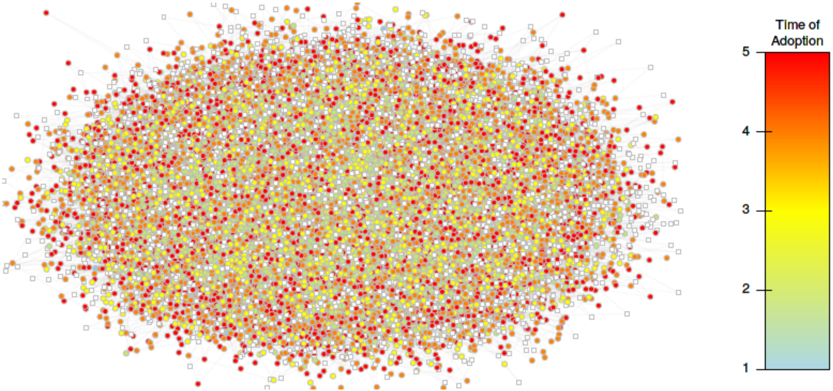


Figure: Random scale-free diffusion network: With 5e4 vertices, diffusion started at the central nodes.

Visualizing Diffusion: `plot_diffnet2 + diffmap` w/ big graph

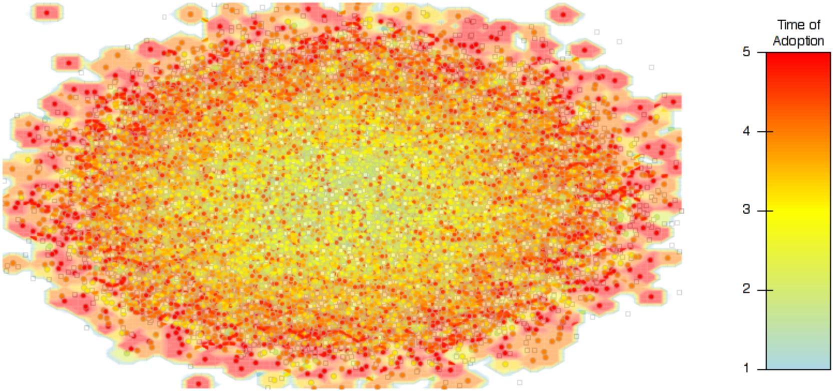


Figure: Random scale-free diffusion network: With 5e4 vertices, diffusion started at the central nodes.

Visualizing Diffusion: `plot_diffnet2 + diffmap` w/ big graph

But. . . be careful with this!

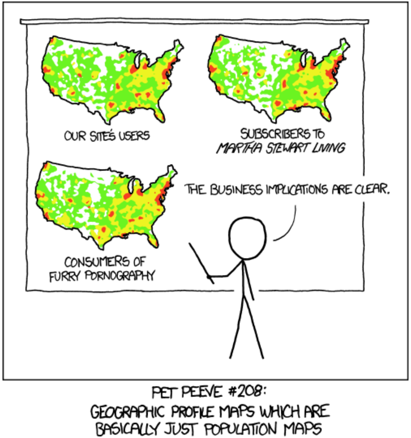

OUR SITE'S USERS
SUBSCRIBERS TO MARTHA STEWART LIVING
CONSUMERS OF FURRY PORNOGRAPHY
THE BUSINESS IMPLICATIONS ARE CLEAR.
PET PEEVE #208:
GEOGRAPHIC PROFILE MAPS WHICH ARE BASICALLY JUST POPULATION MAPS

Figure: The heatmaps pitfall (source `xkcd.com/1138`)

Simulating Diffusion Process: `rdiffnet`

Simulations are a big thing in **netdiffuseR**. The `rdiffnet` function features:

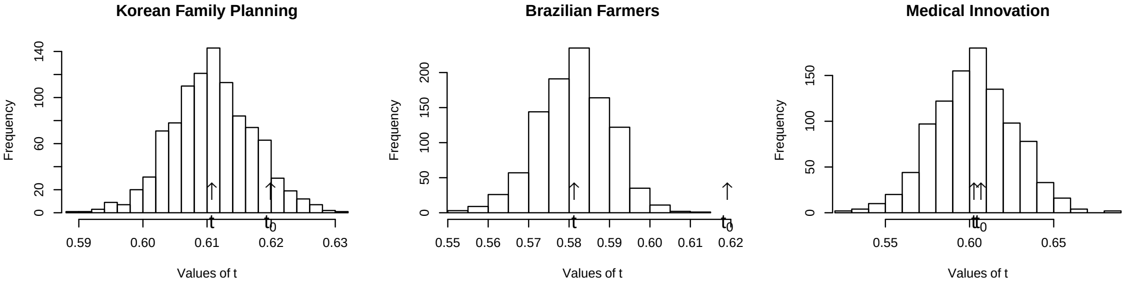
```r
set.seed(887)
mydn <- rdiffnet(
  n=1e3, t=5, # Number of vertices and time points
  seed.nodes     = "random", # Set of initial adopters
  seed.p.adopt   = .15, # Proportion of initial adopters
  seed.graph     = "small-world", # Baseline graph
  rgraph.args    = list(p=4), # Arguments for the rgraph call
  rewire.args    = list(algorithm="swap", p=5), # Rewiring args after time 1
  threshold.dist = function(x) runif(1, .4, .8), # Distribution of thresholds
  exposure.args  = list(normalized=TRUE) # Args for computing exposures
)
```

Statistical inference: `struct_test`

$$H_0 : \mathcal{G} \perp \text{Time of Adoption (random time of adoption)}$$

$$H_a : \mathcal{G} \not\perp \text{Time of Adoption (not random time of adoption)}$$

	Korean Family	Brazilian Farmers	Medical Innovation
p-val	0.1440	0.0000	0.8440
Obs. Avg. threshold t_0	0.6199	0.6191	0.6067
Sim Avg. threshold $\bar{t}$	0.6107	0.5813	0.6026


Korean Family Planning

Brazilian Farmers

Medical Innovation

Concluding remarks

So, to conclude, **netdiffuseR**

- Is an R package focused on network diffusion of innovations

Concluding remarks

So, to conclude, **netdiffuseR**

- Is an R package focused on network diffusion ~~of innovations~~ (in general)

Concluding remarks

So, to conclude, **netdiffuseR**

- Is an R package focused on network diffusion ~~of innovations~~ (in general).
    - Health behavior diffusion (tobacco, drinking, etc.),
    - Memes on social media,
    - Country-level spill over effects (FCTC)
    - ... social contagion (human or not) in general.[1]

[1] Social learning in animals, see Franz and Nunn (2009)

Concluding remarks

So, to conclude, **netdiffuseR**

- Is an R package focused on network diffusion ~~of innovations~~ (in general).
    - Health behavior diffusion (tobacco, drinking, etc.),
    - Memes on social media,
    - Country-level spill over effects (FCTC)
    - ... social contagion (human or not) in general.[1]
- Provides out-of-the-box tools for network diffusion analysis (both classic and new),

[1] Social learning in animals, see Franz and Nunn (2009)

Concluding remarks

So, to conclude, **netdiffuseR**

- Is an R package focused on network diffusion ~~of innovations~~ (in general).
  - Health behavior diffusion (tobacco, drinking, etc.),
  - Memes on social media,
  - Country-level spill over effects (FCTC)
  - ... social contagion (human or not) in general.[1]
- Provides out-of-the-box tools for network diffusion analysis (both classic and new),
- Complements your favorite SNA R package–**igraph**, **statnet**, **RSiena**, etc. (and further, work is been done to extend it to others *s.a.* **spdep**, **spatialprobit**)

[1] Social learning in animals, see Franz and Nunn (2009)

Concluding remarks

So, to conclude, **netdiffuseR**

- Is an R package focused on network diffusion ~~of innovations~~ (in general).
  - Health behavior diffusion (tobacco, drinking, etc.),
  - Memes on social media,
  - Country-level spill over effects (FCTC)
  - ... social contagion (human or not) in general.[1]
- Provides out-of-the-box tools for network diffusion analysis (both classic and new),
- Complements your favorite SNA R package–**igraph**, **statnet**, **RSiena**, etc. (and further, work is been done to extend it to others *s.a.* **spdep**, **spatialprobit**)
- Is on active development, so stay tuned for more to come!

[1] Social learning in animals, see Franz and Nunn (2009)

Thank you!

More info

https://github.com/USCCANA/netdiffuseR

http://cana.usc.edu

contact

vegayon@usc.edu

@gvegayon

Sesssion Info

version	R version 3.3.1 (2016-06-21)
system	x86_64, darwin15.5.0
ui	unknown
language	(EN)
collate	en_US.UTF-8
tz	America/Los_Angeles
date	2016-06-29

package	*	version	date	source
boot		1.3-18	2016-02-23	CRAN (R 3.3.1)
devtools		1.12.0	2016-06-24	CRAN (R 3.3.1)
digest		0.6.9	2016-01-08	CRAN (R 3.3.1)
evaluate		0.9	2016-04-29	CRAN (R 3.3.1)
formatR		1.4	2016-05-09	CRAN (R 3.3.1)
highr		0.6	2016-05-09	CRAN (R 3.3.1)
igraph		1.0.1	2015-06-26	CRAN (R 3.3.1)
knitr	*	1.13	2016-05-09	CRAN (R 3.3.1)
lattice		0.20-33	2015-07-14	CRAN (R 3.3.1)
magrittr		1.5	2014-11-22	CRAN (R 3.3.1)
MASS		7.3-45	2016-04-21	CRAN (R 3.3.1)
Matrix		1.2-6	2016-05-02	CRAN (R 3.3.1)
memoise		1.0.0	2016-01-29	CRAN (R 3.3.1)
netdiffuseR	*	1.16.6	2016-06-27	local
Rcpp		0.12.5	2016-05-14	CRAN (R 3.3.1)
rstudioapi		0.5	2016-01-24	CRAN (R 3.3.1)
sna		2.3-2	2014-01-14	CRAN (R 3.3.1)
SparseM		1.7	2015-08-15	CRAN (R 3.3.1)
stringi		1.1.1	2016-05-27	CRAN (R 3.3.1)
stringr		1.0.0	2015-04-30	CRAN (R 3.3.1)
withr		1.0.2	2016-06-20	CRAN (R 3.3.1)
xtable	*	1.8-2	2016-02-05	CRAN (R 3.3.1)