R Packages for Social Network Analysis

Ian McCulloh and Alexander Perrone

Applied Physics Laboratory
Johns Hopkins University
11100 Johns Hopkins Road
Laurel, MD 20723
ian.mcculloh@jhuapl.edu and Alexander.Perrone@jhuapl.edu

Synonyms

ergm, igraph, network, network science, R, R-project, RSiena, sna, SocialMediaLab, statnet

Glossary

CLI: Command Line Interface is a software interface that requires users to enter typed commands to perform computational operations.

CRAN: The Comprehensive R Archive Network is a network of FTP sites around the world that stores identical code and documentation for R. The CRAN is maintained by the R Foundation.

GUI: Graphical User Interface is a software interface designed to simplify and standardize the use of computer programs.

INSNA: The International Network of Social Network Analysts is the academic professional society for social network analysis, founded in the 1970s.

Package: A set of computer code, functions, and commands that have been bundled, tested, reviewed, and are available for use in the R programming environment.

R: An open source computer language for statistical computing.

R-Project: An activity supported by the non-profit R Foundation to promote open source statistical computing.

Social Media: An online means of communication, where large groups of people communicate, share information, and develop personal and professional contacts.

SOCNET: A list-serv where social network scholars share knowledge and information pertaining to social network analysis research and application.

Sunbelt Social Networks Conference: The annual international conference of INSNA.

Definition

R is an open source computer language and environment for statistical computing. Many users contribute to the R environment by publishing packages that perform various computational functions. There has been a recent growth in the use of R for social network analysis and with it the introduction of numerous packages. In some cases, these packages provide redundant capabilities and may be incompatible with each other. In other cases, packages provide unique analytics, but may not be widely known. This paper provides a review of many of the common R packages used in social network analysis. Recommendations for users and future R development are provided.

Introduction

Social network analysis (SNA) is an academic methodology to investigate the structure of social interactions among people (Wasserman and Faust, 1994; McCulloh et al, 2013). SNA relies on a wide range of interdisciplinary theories from throughout the social and natural sciences. Social networks consist of actors, represented by nodes (or vertices), and relationships between actors, represented by links (or edges). There are many types of relationships that may be represented, such as friendship, respect, or co-affiliation. There are infinite potential relationships that can be modeled or investigated. SNA is often extended more broadly than networks of people. These extended networks are referred to as meta-networks (Carley and Krackhardt, 1998; Carley, 1999) and can include nodes such as organizations, resources, skills, tasks, events, roles, beliefs, and more.

A key aspect of SNA that differentiates it from other analytic methods is its focus on structure. In most scientific analyses, data are assumed to be independent and identically distributed (i.i.d.). In cases where this does not hold, the data are often transformed so that the i.i.d. assumption is met. This assumption allows the application of powerful statistical methods. In most applications of SNA, the i.i.d. assumption is assumed to be false. The behavior and attributes of one actor are not independent of the other actors within their network and they are not identically distributed. SNA is interested in how these structures form and how they impact the behavior of the nodes and groups as a whole. Interest in network structure is not exclusive to scientists studying social groups. There are many applications of networks, and the methods and measures developed in SNA have found broader application to a range of scientific disciplines.

The broad scope of SNA has led to a wide range of software. R is an open-source language and environment for statistical computing and graphics (Ihaka and Gentleman, 1996), which now includes additional applications such as SNA. Communities of contributors write and maintain packages for use in R. Still others write and share scripts to perform certain analyses. It is common that a package or script intended for one application will be repurposed for another. There are also multiple packages for

R that accomplish some of the same functions. For example, the packages ‘igraph’ and ‘sna’ both support SNA and offer commands to calculate some of the same common structural measures such as centrality (Freeman, 1977; 1978; Freeman et al, 1979) or a triad census (Wasserman, 1977).

Historical Background

There has been tremendous growth in both the use of R and the number of available packages. As of the writing of this paper, there have been over 8,000 R packages published to the Comprehensive R Archive Network (CRAN), the main repository of packages and documentation for R. Figure 1 displays the number of CRAN packages over time. This shows exponential growth in the contributions to R. Figure 2 displays an estimate of the number of user downloads of R packages over time. This figure shows a similar explosion in activity related to R with over half a million package downloads per day.

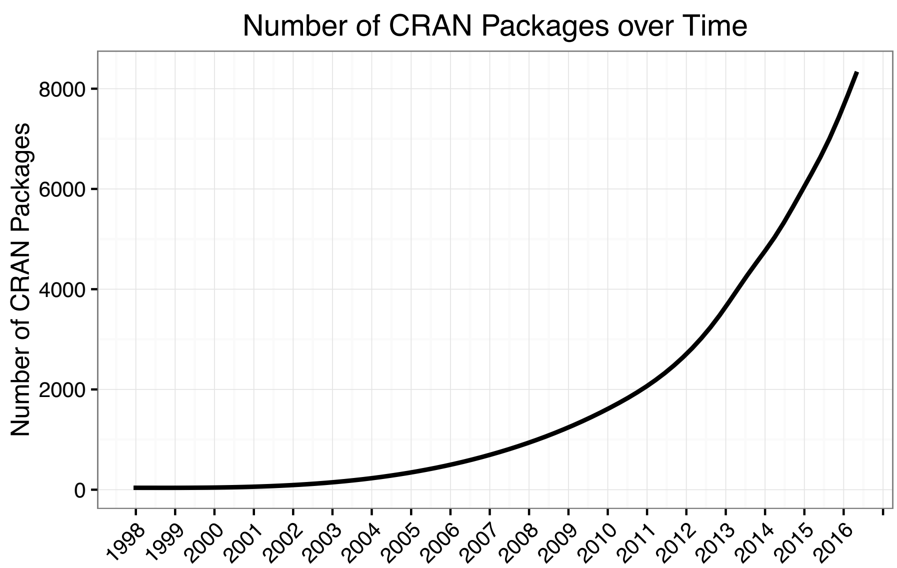


Figure 1. **<u>The number of R packages available over time</u>. The CRAN page for each package was used to provide a data set of packages. The main page for each package often (but not always) provides a link to “old sources” which contains an archive of previous releases. The earliest release date on the archive was used as the initial release date for the package. If no archive existed, the published date on the main page was used as the initial release date, since there was only one version of the package. The initial release dates were then sorted by date from earliest to latest, and a cumulative sum was computed to count the number of packages over time, which is displayed in the plot.**

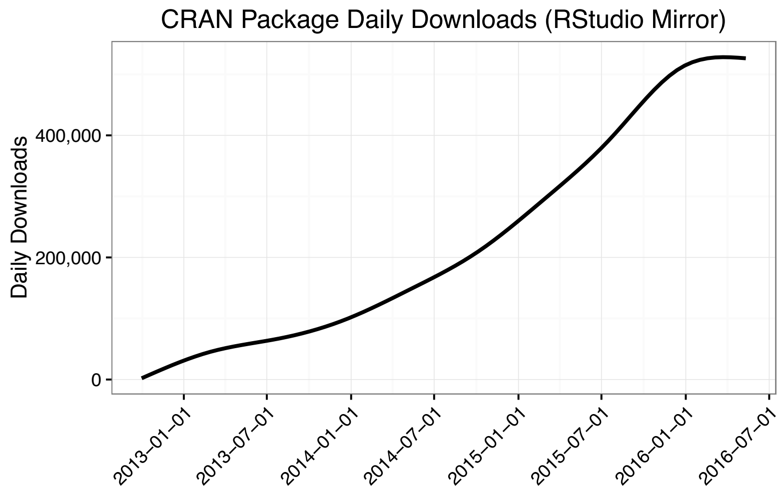


Figure 2. R Package Daily Downloads. Obtained using the 'cran.stats' package in R, which provides functions for scraping and analyzing download logs from RStudio's CRAN mirror. The download logs were obtained from 2012-10-01 (earliest available date) to 2016-05-11. The plot shows the daily downloads smoothed using locally estimated scatterplot smoothing (LOESS) (Cleveland, 1979) with span set to 0.75 to remove daily variability. These data are used as a proxy for R package downloads. These data are therefore sensitive to the popularity of RStudio as an environment for using R, with the proxy becoming better the more RStudio has become popular among R users.

R has also become increasingly popular for SNA. Figure 3 shows the growth in three common R packages for use with social networks: 'igraph', 'sna', and 'tnet'. While SNA use in R is not as dramatic as overall R usage, the trend is generally increasing. Daily download estimates of SNA packages in R exceeds 1,500.

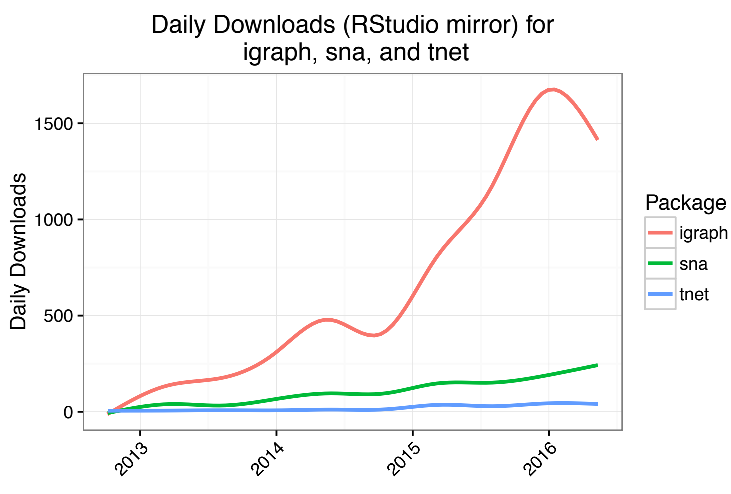


Figure 3. Estimated daily downloads of SNA packages in R over time. The download logs from the RStudio mirror were subset to the packages igraph, sna, and tnet, and their downloads are shown in the plot. The daily estimates were smoothed using LOESS with span set to 0.75.

Key Points

Given the growth of SNA use in R and the explosion in the number of packages, a review of available R packages is necessary to assist users in finding packages with needed functionality. Due to the volume of packages, however, a package by package comparison was deemed infeasible. The remainder of this paper is therefore organized as follows. Advantages and disadvantages of the use of R compared to commercial SNA software are briefly discussed. Definitions that are relevant to the R community are provided. Given the number of R packages in use for SNA and the broader R community, a methodology is introduced that is used to identify and compare key packages for SNA. Data obtained using this methodology are also described. Descriptions and comparisons of key packages are discussed. This includes key packages for SNA as well as packages intended to add niche functionality to other SNA packages. Recommendations are provided for those using R for SNA. Finally, future directions for the use and implementation of SNA in R are proposed.

Advantages and Disadvantages of R

There are advantages and disadvantages for the use of R as an SNA software tool. This is also true of particular packages in R. R is free and open source. This makes the use of R available to anyone, even with no budget. While this free tool does not have direct customer support, there is a growing user community that will provide help and advice. In addition, there are numerous training opportunities such as in-person or online bootcamps, academic conference workshops, online courses, and learning platforms. There are also numerous online Coursera courses using R, including two R specializations by JHU and Duke (Peng et al, 2016; Çetinkaya-Rundel et al, 2016). Many of the organizations offering these training opportunities are also available to offer tailored workshops for organizations wishing to adopt R.

A disadvantage of R is that commands are entered via code at the command line, rather than a point-and-click graphical user interface (GUI). While there are some GUI options available, such as R Commander, users typically need to have some comfort with programming to take full advantage of R. Recently, the Shiny web application framework has become a popular way to create user interfaces using only R. At this point, R users typically need to develop their own user interfaces, although there have been some R packages utilizing Shiny which provide user interfaces for other R packages, such as statnetWeb (for statnet) and ShinyStan (for Stan, JAGS, and related languages). It is likely that these applications will become more common, making the use of R more accessible to a wider range of users. Nevertheless, like the R package landscape, it will likely remain a decentralized array of user interfaces built for particular purposes, rather than a single, general GUI application.

Since R content is generated by a user community, it is possible that the algorithms developed for applications are not tested to the level of rigor expected of commercial software (Wasserman and

Capra, 2007). In addition, customers of commercial software have the expectation that the code base will be maintained, updated, quality controlled and that in some cases support services will be provided. The tradeoff, however, is that novel analytics often appear in R before they are available in commercial software, if they ever appear in commercial software at all. Users also maintain a great deal of flexibility by being able to modify or develop their own analytics in R. This flexibility is valued by users who are often faced with novel questions that cannot be addressed with traditional software packages.

The R Development Environment

R is a computer language and environment for statistical computation and graphics. It is available as free software distributed under the GNU General Public License, version 2. The goal of this license is to ensure that developers maintain the freedoms to run R for any purpose, to access and study its source code, redistribute copies, and redistribute any modified versions of R under the terms of the license. In addition, many R packages are distributed with either a free software license or an open-source license.

There are four primary methods that R developers share their code. The Comprehensive R Archive Network (CRAN) is a collection of web servers that offer the R software environment, extensions or packages contributed by the R community, and documentation for R. The master R site is located at https://CRAN.R-project.org. This is the official repository maintained by the R Foundation and a team of volunteers. Developers create packages to extend R functionality and submit them through the CRAN website for consideration to be included on the CRAN repository. Policies for CRAN package development and submission can be found at https://cran.r-project.org/web/packages/policies.html. The policies ensure that the package is free of legal restrictions, conforms to certain standards, and doesn't create major conflicts or problems with existing packages, such as two packages with the same name. It also requires that a single individual maintain the package and that they can easily be contacted by the CRAN team. The CRAN is the primary location to find stable R code for application.

Two other locations for R code are R-Forge and GitHub. These platforms provide version control and aid in collaboration among software developers. It allows two or more developers to modify code at the same time. Others in the community can review and make comments on the code that may offer helpful suggestions or efficiency. There are thousands of packages available on R-Forge and GitHub. A developer may or may not choose to submit their finished code to CRAN, as the CRAN standards are more restrictive than R-Forge or GitHub. In addition, using a package from one of these sites is just as easy as using one from CRAN. For example, a GitHub package can be loaded into the R environment with the following commands:

```
install.packages("devtools")
devtools::install_github ("username/packagename")
```

The disadvantage of R-Forge and GitHub is that the packages may not be maintained and may not meet the compatibility and quality standards required by CRAN. For this reason, we restrict our exploration of SNA packages in R to the CRAN site only.

The fourth method that R developers share code is through scripts. Scripts can be shared as a text file or as files written in R and saved with a .R extension. Scripts may not be under version control and may not be part of a package available on CRAN, R-Forge, or GitHub. Most R users, however, will maintain personal scripts for using R that they might share with students or colleagues. Scripts do not need to be loaded like a package (i.e., using the library command) since they are a collection of commands that can be run directly at the command line.

The R package is the basic unit of shareable R code. An R package includes functions and documentation describing their use and often includes sample data. An R package may also require dependencies. A dependency is another R package that is used by the primary package. A dependency has two levels of strength. An import dependency is a package that must be loaded in order for the primary package to work. A suggest dependency is a package that can be used by the primary package to extend functionality. Import dependencies are automatically loaded with the primary package, whereas suggest dependencies are not.

R does not allow cyclic dependencies. In other words, if package A is required by package B and B is required by package C, then C cannot be required by package A. The dependency relation in R therefore allows the construction of a hierarchical network with no cycles.

This section is by no means comprehensive and such an undertaking is well beyond the scope of this paper. This section is meant to provide a brief orientation to the R development environment and packages. CRAN and dependency relations will be used to identify packages that are available and useful for conducting SNA in R.

Method and Data Collection

Given the large number of packages available in R, it was important to develop a methodology for which SNA relevant packages could be identified. The methodology used an inductive and a segmental approach (Valente, 2012). The inductive approach began with seeking feedback from the SNA community about common R packages and applications. The segmental approach involved scraping all R packages and using a community detection approach to identify categories of related packages that might find application with SNA. This method resulted in 135 R packages that find application in SNA and nine categories of packages.

The inductive approach began with identifying a broad base of SNA professionals. The authors identified the International Network of Social Network Analysts (INSNA). This organization is the academic professional society for SNA, founded in 1977. Each year, INSNA hosts an annual conference known as the International Sunbelt Social Networks Conference. In 2016, this event was attended by over 800 people. The conference offers a number of workshops at the beginning of the conference. INSNA also maintains a list-serv called 'SOCNET' where the SNA community can post questions, share resources, and provide assistance to others in the community.

Once a community of experts was defined, we identified the use of R in the Sunbelt conference workshops. Of the 33 workshops, 16 of the workshops used R. The nine specific packages that were taught include: 'statnet', 'ergm', 'tergm', 'igraph', 'netdiffuseR', 'RSiena', 'BlauNet', 'egonetR', 'SocialMediaLab'. This step was used to identify popular SNA packages in current use by the SNA community.

Step two required a search of the CRAN for the following key words: "network", "matrix", and "graph". This returned 53 additional packages including: 'amen', 'assortnet', 'bipartite', 'blkergm', 'blockmodeling', 'data.table', 'devtools', 'd3Network', 'DCG', 'dils', 'dnc', 'Dominance', 'geomnet', 'GGally', 'ggnetwork', 'influenceR', 'intergraph', 'latentnet', 'keyplayer', 'linkcomm', 'Matrix', 'mcPAFit', 'multiplex', 'ndtv', 'NetCluster', 'netcoh', 'NetComp', 'nets', 'NetSim', 'NetSwan', 'nettools', 'NSUM', 'network', 'networkDynamic', 'networksis', 'networkD3', 'queueing-package', 'sand', 'slam', 'sna', 'SNscan', 'SocialNetworks', 'sparkR', 'SPARQL', 'SparseM', 'spnet', 'statnetWeb', 'statools', 'timeordered', 'tnam', 'tnet', 'tsna', and 'visNetwork'. Upon inspection, it was clear that the primary application for several of these packages was not SNA. We therefore removed the following packages from consideration: 'data.table', 'devtools', 'GGally'. Several additional packages used the term "network" for non-social network application; for example, "neural networks" or "Bayesian networks". These packages were excluded from additional analysis.

The third step involved searching the CRAN for well-known social media platforms. This search returned 12 additional packages. They consisted of 'graphTweets', 'instar', 'RedditExtractoR', 'Rfacebook', 'RKlout', 'Rlinkedin', 'RNewsflow', 'SocialMediaMineR', 'tumblR', 'twitteR', 'WikipediaR' and 'WikiSocio'. While the authors recognize the difference between SNA and social media analysis (SMA), the SMA packages remained in the data set under investigation.

For the fourth step, the consolidated list of R packages was posted to the SOCNET list-serv with the following request: "If you are an R user, I'd appreciate it if you could look at the list [of R packages] and let me know if I am missing any packages." List-serv members were also asked to provide "any thoughts on things [related to R and SNA] that might [be] useful in a review paper on this topic." The list-serv remained active on this topic for approximately one month and one of the authors continued to receive personal emails regarding the use of R for SNA for approximately two months. This resulted in 10 additional packages to include: 'cssTools', 'astsa', 'forecast', 'hergm', 'PCIT', 'RSienaTest', 'mixer', 'dplyr', 'ggplot', and 'rvest'.

The fifth and final step in the induction approach was to scrape the descriptions of all R packages and conduct a key word search for "igraph", "graph", "network", "sna", and "social network analysis". This resulted in identifying 386 packages, some of which were already identified. Many of the 386 packages were not applicable to social network analysis, but may have used a network analysis approach for some other application. The authors made a qualitative judgment on which of the 386 packages should be included. The total list of SNA-related packages consists of 135 and is described below.

The segmental approach intended to identify categories of R packages. Categories for R packages were posted on the INSNA SOCNET list-serv and also hosted on a Google Sheets page, where people in the SNA community were able to nominate categories for SNA packages. The nominations consisted of 1) basic SNA; 2) statistical analysis of networks; 3) network visualization; 4) social media analysis; 5) community detection and subgroup analysis; and 6) a miscellaneous category. Upon further examination of the consolidated list of packages, the authors added three additional categories consisting of 7) simulation; 8) data management; and 9) example data packages. One of the people on the SOCNET list-serv suggested doing a consensus analysis, however, due to the cross-discussion, people's responses were not provided independently from the input of others. Therefore, a key assumption underlying consensus analysis was missing. Ideally, future similar efforts should attempt to conduct a proper consensus analysis with a pile sort to determine categories. For this project, 13 people provided input into the category selection similar to a focus group, but conducted asynchronously, online.

The final step in the segmental approach involved creating a network visualization, where nodes represented R packages and links represented dependency relations as reported on the CRAN package page as of May 5, 2016. Nodes are colored based on the category in which they were grouped. The authors provide qualitative discussion and insights based on the network visualization.

The analysis revealed two primary basic SNA packages used in R, 'igraph' and 'sna'. These packages perform some redundant analytic calculations, they use different data structures, and they create conflicts with each other. Therefore, the authors conducted several performance comparisons between the two packages. Performance criteria included system time and consistency across calculation. Memory as a performance criterion was compared for the two different data structures used by 'igraph' and 'sna'. In order to evaluate the performance, four SNA tasks were selected that could be performed in both 'igraph' and 'sna'. Those tasks included two centrality measures, degree and betweenness, as well as two graph level measures, diameter, and triad census. These were selected to vary the computational complexity of the calculation as well as the focus of measure. In addition, the size and topology of the networks were varied. While the performance of the basic SNA package cannot speak to the efficiency and functionality of the other packages that depend upon it, it may provide some additional insight for selecting which basic SNA package to build upon for new development.

Categorization of R Packages

A full explanation and comparison of the 135 packages used for SNA in R is well beyond the scope of this paper. This section will identify packages used for common SNA functions and provide additional detail for the most popular and widely used packages. The nine categories of packages are: basic SNA, statistical analysis of networks, network visualization, social media analysis, community detection/subgroup analysis, niche SNA, simulation, data management, and data.

Basic SNA

There are two main packages for basic SNA in R: 'igraph' and 'sna'. The package 'igraph' is by far the most popular by download statistics. This may be due, in part, to its application in network analysis that is not necessarily social. Another possible explanation is that 'igraph' is not exclusive to R. It is also available for Python and C/C++. The package 'sna', on the other hand, is primarily used by the package 'statnet' and associated dependencies for the statistical analysis of networks solely within R.

The two packages use different data structures and are not fully compatible. The 'igraph' package uses its own class in R. The command "as.igraph()" will coerce a matrix object into an 'igraph' object. The 'igraph' package also has an "as_adj_list()" command to coerce a data structure into an adjacency list format, which is often more computationally efficient. The 'sna' package, however, uses the 'network' class and relies on the 'network' package.

There is a package 'intergraph' that provides functions to coerce network data between a 'network' object and an 'igraph' object. This package does not make commands interoperable between the 'sna' and 'igraph' however. It will be necessary to load both 'sna' and 'igraph' in addition to 'intergraph' in order to coerce network data from one format to the other. The "library()" command is used to load a package into R. Once the network data has been converted into the desired format, the "detach()" command should be used to remove the unneeded package from the environment. Alternatively, one could also reference functions directly if there is a conflict without using the detach command. For example, the command "igraph::betweenness(N)" will calculate the betweenness centrality of graph N using the 'igraph' package, while the command 'sna::betweenness(N)' will calculate betweenness centrality of graph N using the 'sna' package. Referencing functions without detaching them is necessary if one still needs both packages loaded for some reason. Failure to resolve conflicting packages can result in excessive bugs and errant analysis.

Table 1 contrasts the difference in basic capability and associated command for common SNA routines. It can be seen in the table that each package contains measures that the other does not. Both packages calculate basic centrality and common graph level measures.

Table 1. Comparison of Basic SNA Package Functions

Measure	igraph	sna
Bonacich Centrality	alpha_centrality	---
Assortativity coefficient	assortativity	---
Authority Centrality	authority_score	---
Centralization	centralize	centralization
Centralization, betweenness	centr_betw	centralization

Centralization, closeness	centr_clo	centralization
Centralization, degree	centr_degree	centralization
Centralization, eigenvector	centr_eigen	centralization
Cliques	cliques	clique.census
Centrality, betweenness	betweenness estimate_betweenness	betweenness
Centrality, closeness	closeness	closeness
Centrality, degree	degree	degree
Centrality, eigenvector	eigen_centrality	evcent
Diameter	diameter	~geodist
Dyad Census	dyad_census	dyad.census
Find Ego Network	Ego	ego.extract
Hub Centrality	hub_score	---
Page Rank	page_rank	---
Bonacich Power Centrality	power_centrality	bonpow
Radius	radius	---
Reciprocity	reciprocity	grecip mutuality
Transitivity	transitivity	gtrans
Triad Census	Triad_census	triad.census
Blockmodel	---	blockmodel
Connectedness	---	conectedness
Efficiency	---	efficiency
Density	---	gden
Hierarchy	---	hierarchy
Information Centrality	---	infocent
Least Upper Bound	---	lubness
Triad Classification	---	triad.classify

The two basic SNA packages ‘igraph’ and ‘sna’ were compared for performance differences using simulation. Directed networks were simulated using ‘igraph’s built in simulation functions. Half of the networks were generated as Erdos-Renyi (1959) random networks using the command “erdos.renyi.game()”. The other half of the networks were generated as scale-free networks using a preferential attachment mechanism (Barabási and Albert, 1999) using the command “barabasi.game()”. Half of the networks were small networks, consisting of 30 nodes. The other half of the networks were mid-size networks consisting of 150 nodes. The density of the network was fixed at 0.1. The “*m*” parameter on the scale-free network was used to calibrate the scale-free network generation to achieve a comparable network density. For the 30 node network, $m = 3$, and for the 150 node network, $m = 2$. A total of 1,000 simulations were run for each of the four conditions. For each simulation run, the elapsed time was recorded for degree centrality, betweenness centrality, diameter, and triad census on the generated ‘igraph’ network object. The ‘intergraph’ package was then used to coerce the simulated network from an 'igraph' object into a 'network' object for use with the 'sna' package. The rationale was to compare a computationally fast algorithm, such as degree centrality with a computationally slow algorithm, such as betweenness centrality. The diameter and triad census measures provided graph level measures The object size in memory of the generated network, stored as an 'igraph' object and as a 'network' object were also recorded.

For each measure a two-sample t-test and a pairwise distance t-test were performed to evaluate differences. In all cases, elapsed time for 'igraph' was subtracted from elapsed time for 'sna', so a positive value of the t statistic indicates that 'igraph' is faster, while a negative t statistic indicates that 'sna' is faster. For memory the same convention was applied, meaning that a positive value for the t statistic indicates 'network' objects are more memory intensive, while a negative value indicates 'igraph' objects are more memory intensive. Table 2 provides the t-statistics for the different simulation conditions. All tests are significant at the p<0.0001 level.

Table 2. Comparison t-Statistics for 'igraph' versus 'sna' basic SNA packages

		2-Sample t-Tests					Pairwise t-Test				
Nodes	Topology	Deg	Btw	Dia	Triad	Size	Deg	Btw	Dia	Triad	Size
30	Random	-61.5	-24.7	105.3	162.7	118.3	-58.8	-26.7	103.5	175.5	123.2
30	Scale-free	-68.7	-49.6	115.1	139.5	22410	-63.8	-50.9	111.2	142.4	22410
150	Random	180.6	414.6	390.8	539.7	560.6	183.6	423.6	397.4	566.2	583.7
150	Scale-free	34.2	59.0	437.1	302.7	58272	33.9	68.9	447.1	301.3	58272

Neither package is uniformly superior to the other. The negative t-statistics in Table 2 demonstrate that the 'sna' package is computationally faster for calculating centrality measures on small networks around 30 nodes. In all other cases, however, 'igraph' outperforms 'sna' in terms of elapsed time and the object size in memory. In the range of network size and topology where 'sna' outperforms 'igraph', however, the networks are small enough that the user is unlikely to notice the difference in performance. While the difference is statistically significant, the effect size of two to four ten thousandth of a second difference is undetectable to the user.

Another basic SNA package is 'tnet'. This package is designed to analyze weighted networks, two-mode networks, and longitudinal networks. It provides a method of analysis to prevent information loss when these networks are simplified for dichotomous, single-mode, and static networks. This package depends upon igraph and uses the igraph class as its data structure.

For the remaining sections on categories of packages, each table of packages will reference the basic SNA package that is listed as a dependency for each package in the category. In the event that a package does not have a basic SNA package listed as a dependency, then "n/a" will be recorded in the table.

Statistical Analysis of Networks

There are 35 identified packages for the statistical analysis of social networks in R as shown in Table 3. Most use the 'network' class and rely on the 'sna' basic SNA package. A number of packages simply work with a 'Matrix' class object. In the authors' opinion, 'sna' provides access to a wider range of useful statistical analysis functions for use with network data.

Table 3. Statistical Analysis Packages

Package	SNA Pkg	Application
amen	sna	Analysis of dyadic relations using additive and multiplicative effects (AME)
asnipe	igraph, sna	Animal Social Network Inference and Permutations
Bergm	sna	Tools for Bayesian exponential random graph models
blkergm	sna	Fitting block ERGM given the block structure on social networks
bootnet	n/a	Bootstrap Methods for Various Network Estimation Routines
btergm	sna	Temporal ergm by Bootstrapped Pseudolikelihood
degreenet	igraph, sna	Models for Skewed Count Distributions Relevant to Networks
ergm	sna	Fit, Simulate and Diagnose Exponential-Family Models for Networks
ergm.count	sna	Fit, Simulate and Diagnose ergm with Count Edges
ergm.ego	sna	Fit, Simulate and Diagnose ergm to Egocentrically Sampled Network Data
ergm.graphlets	sna	ERG Modeling Based on Graphlet Properties
ergm.rank	sna	Fit, Simulate and Diagnose ergm for Rank-Order Relational Data
ergm.userterms	sna	User-specified terms for the statnet suite of packages
GERGM	igraph	Tools for generalized exponential random graph models
hergm	sna	Hierarchical exponential random graph models with local dependence
HLSM	n/a	Hierarchical Latent Space Network Model (HLSM)
JRF	n/a	Joint Random Forest (JRF) for the Simultaneous Estimation of Multiple Related Networks
latentnet	sna	Latent position and cluster models for networks
lvm4net	igraph, sna	Latent Variable Models for Networks
netcoh	n/a	Statistical models incorporating network cohesion as an effect
netdiffuseR	igraph	Tools for estimating models of the diffusion of innovations
nets	igraph	Network estimation for time series
PCIT	n/a	Identifies meaningful correlations to define edges in a weighted network
Rambo	sna	The Random Subgraph Model
rem	sna	Relational event models
RSiena	n/a	Fits stochastic actor oriented models to longitudinal network data
statnet	sna	An umbrella package for many network analysis packages
statnet.common	sna	Non-statistical utilities used by statnet and related packages
statnetWeb	sna	A web-based interface for using the statnet package
staTools	n/a	Tools for modeling the statistical distribution of a vector of observations
tergm	sna	Temporal ergm for dynamic networks over time

timeordered	igraph	Time ordered and time aggregated network analysis
tnam	sna	Temporal network autocorrelation models
tsna	sna	Temporal SNA tools for continuous and discrete time networks over time
xergm.common	sna	Extensions of exponential random graph models

Network Visualization

There are several approaches to network visualization. There are basic visualization tools that provide the code necessary to plot static networks. Enhanced visualizations are often made possible by JavaScript. JavaScript can be accessed through a package such as 'networkD3' or 'visNetwork' which provide different libraries for visualization. Other packages, such as 'plotly', translate network layouts into an interactive, web-based version. In the authors' subjective opinions, the 'igraph' basic SNA package provides access to better visualization layouts, routines, and downstream packages than the 'sna' basic SNA package. There are 17 visualization packages identified as shown in Table 4.

Table 4. Network Visualization Packages

Package	SNA Pkg	Application
bipartite	igraph,sna	Visualize bipartite networks
d3Network	n/a	Tools for D3 JavaScript – *Active development is done in networkD3*
diagram	n/a	Functions for visualizing simple graphs, plotting flow diagrams
edgebundleR	igraph	Circle Plot with Bundled Edges
geomnet	sna	Single geometry approach to network visualization with ggplot2
ggnetwork	sna	Geometries to plot networks with ggplot2
ggplot2	n/a	Implementation of the grammar of graphics
HiveR	sna	2D and 3D Hive Plots for R
igraphinshiny	igraph	Use shiny to Demo igraph
ndtv	sna	Tools for animated visualizations of networks
networkD3	igraph	D3 JavaScript network graphs from R
networkDynamic	sna	Support for dynamic, temporal networks
plotly	n/a	Translate ggplot2 into an interactive web-based version
SPARQL	n/a	Select or update queries
spnet	igraph	Plotting social networks on maps
threejs	igraph	Interactive 3D Scatter Plots, Networks and Globes
visNetwork	igraph	Uses vis.js library for network visualizations

Social Media Analysis

There were 13 packages for analyzing social media identified in R and shown in Table 5. With the exception of 'SocialMediaLab', all tools focused on simply extracting data from an application program interface and storing the data as a list, data.frame, data.table, or some other general class. The data can then be coerced into either a 'network' or 'igraph' object for further analysis.

Table 5. Social Media Analysis Packages

Package	SNA Pkg	Application
graphTweets	igraph	Visualise Twitter Interactions
instar	n/a	Tools for Instagram data
RedditExtractoR	n/a	Tools for Reddit
Rfacebook	n/a	Tools for Facebook data
RKlout	n/a	Fetch Klout Scores for Twitter Users
Rlinkedin	n/a	Tools for LinkedIn data
RNewsflow	igraph	Tools for Analyzing Content Homogeneity and News Diffusion
SocialMediaLab	igraph	Tools for collecting and constructing networks from social media data
SocialMediaMineR	n/a	Investigates popularity and reach of URL(s) on social media
tumblR	n/a	Access to Tumblr v2 API
twitteR	n/a	Tools for Twitter data
WikipediaR	n/a	Wikipedia web API. Login not required
WikiSocio	igraph	store wiki archives to R object - data-frame, lists, vector and variables

Community Detection/Subgroup Analysis

There are several clustering and community detection packages available in R. Some packages are intended for use with 'igraph', others with 'sna', and still others use the 'Matrix' object. Based on the investigation described in this paper, community detection and subgroup analysis remains limited. There is more functionality to visualize clusters and compare the quality of different clustering solutions than there exists to perform clustering and community detection. A total of 18 packages were identified for community detection and subgroup analysis, which are shown in Table 6.

Table 6. Community Detection and Subgroup Analysis Packages

Package	SNA Pkg	Application
blockmodeling	sna	Generalized blockmodeling for valued networks
blockmodels	n/a	Latent and stochastic blockmodel estimation
cvxbiclustr	igraph	Convex Biclustering Algorithm
cvxclustr	igraph	Splitting methods for convex clustering
DCG	n/a	Random walks to find community structure in networks
dnc	igraph	Community detection for dynamic networks (repeated measures)
fcd	n/a	Fused Community Detection
linkcomm	igraph	Reveals nested and overlapping structure in networks
MCL	n/a	Markov Cluster Algorithm
mixer	n/a	Random graph clustering
modMax	igraph	Community Structure Detection via Modularity Maximization
NetCluster	sna	Evaluate clustering solution
rnetcarto	igraph	Fast Network Modularity and Roles by Simulated Annealing
sbmSDP	n/a	Semidefinite Programming for Fitting Block Models of Equal Block Sizes
SNscan	igraph	Test network clustering patterns using scan statistics

SOMbrero	igraph	SOM Bound to Realize Euclidean and Relational Outputs
VBLPCM	sna	Variational Bayes Latent Position Cluster Model for Networks
wfg	igraph	Weighted Fast Greedy Algorithm

Niche SNA

The following table lists 30 R packages that support important SNA capability, but do not seem to fit within the previous categories. Most of these packages are intended for use with 'igraph'. These packages are shown in Table 7.

Table 7. Niche or Uncategorized Network Analysis Packages

Package	SNA Pkg	Application
agop	igraph	Aggregation operators for scientometrics and bibliometrics
assortnet	n/a	Calculate assortitivity
backShift	igraph	An algorithm to estimate the connectivity matrix of a directed (possibly cyclic) graph with hidden variables.
bingat	sna	Binary Graph Analysis Tools
Blaunet	sna	Calculate and Analyze Blau Status for Measuring Social Distance
CITAN	igraph	CITation ANalysis Toolpack
crimelinkage	igraph	Statistical Methods for Crime Series Linkage
cssTools	sna	Estimate networks from random sample of cognitive social structures
dils	igraph	Data informed link strength: combines multiple networks into one
disparityfilter	igraph	Disparity Filter Algorithm for Weighted Networks
Dominance	igraph	Calculate dominance indices
ebdbNet	igraph	Infer the adjacency matrix using an empirical Bayes estimation
egonet	sna	Analysis of ego-centric network data
influenceR	igraph	Node centrality focused at identifying opinion leaders
interactiveIGraph	igraph	Extends igraph to interactively work with igraph objects
keyplayer	igraph,sna	Calculates group centrality scores
mcPAFit	n/a	Estimate preferential attachment from a single networks using MCMC
NetSwan	igraph	Analysis of network robustness, resilience, and vulnerability
nettools	igraph	Network comparison
networkreporting	n/a	Functions for producing estimates from data collected using network reporting techniques like network scale-up, indirect sampling, network reporting, and sibling history
optrees	igraph	Finds optimal trees in weighted graphs
PAFit	n/a	Nonparametric Estimation of Preferential Attachment and Node Fitness in Temporal Complex Networks
qgraph	igraph, sna	Used to visualize data as networks as well as provides an interface for visualizing weighted graphical models
QuACN	igraph	Topological measures for structural analysis of complex networks
SDDE	igraph	Compares network X to an augmented network Y by counting the number of Shortcuts, Detours, Dead Ends (SDDE), equal paths and disconnected nodes.

SocialNetworks	n/a	Generates social networks based on node distance
spatgraphs	n/a	Graph component calculations for spatial locations
spatialnbda	n/a	Network based diffusion analysis (NBDA) allows inference on the asocial and social transmission of information
SSrat	sna	Sociometric status determination with rating scales
VertexSimilarity	igraph	Vertex Similarity matrix of an undirected graph

Simulation

A great deal of research in SNA requires the simulation of networks. For example, comparing the performance of 'igraph' and 'sna' required simulation of networks. The following table lists R packages that can be used to simulate network data under different conditions and topologies. There were seven packages identified as shown in Table 8.

Table 8. Network Simulation Packages

Package	SNA Pkg	Application
CCMnet	sna	Simulate Congruence Class Model for Networks
CIDnetworks	igraph	Generative Models for Complex Networks with Conditionally Independent Dyadic Structure
NetComp	n/a	Network generation and comparison
NetSim	igraph	Social network simulation tool
networksis	sna	Tools to simulate bipartite networks/graphs with the degrees of the nodes fixed and specified
snowboot	igraph	Network simulation non-parametric methods of analysis center around snowball and bootstrap sampling
SpaDES	igraph	Implement a variety of simulation models, with a focus on spatially explicit agent based models

Data Management

Some R packages are used just to manipulate network data or coerce one data object into another. While it might have been helpful to include general purpose data management packages such as 'data.table', or 'dplyr', we restrict our list of packages to those that are often listed as dependencies for network packages. There are ten packages in the list, which are shown in Table 9.

Table 9. Data Management Packgages

Package	SNA Pkg	Application
igraphtosonia	igraph	Exports igraph objects to the SoNIA file format
intergraph	igraph,sna	Converts data objects between igraph and network classes
Matrix	n/a	Methods and operations for dense and sparse matrices
multiplex	n/a	Relational algebra tools for the analysis of multiple social networks
network	sna	Tools to create and modify network objects

rgexf	igraph	Create, read and write GEXF (Graph Exchange XML Format) files used in Gephi and other programs
slam	n/a	Data structures and algorithms for sparse arrays and matrices
SNFtool	n/a	Takes multiple views of a network and fuses them together to construct an overall status matrix
spam	n/a	Set of functions for sparse matrix algebra
SparseM	n/a	Basic linear algebra functionality for sparse matrices

Four packages included in this section have broader application than SNA. They are 'Matrix', 'slam', 'spam', and 'SparseM'. These are different packages used for storing and manipulating sparse matrices. When the density of a matrix is approximately greater than 0.5, an adjacency matrix is most efficient in terms of memory. When the density is smaller than 0.5 the matrix becomes sparse and an edge list, or adjacency list becomes more efficient.

It is important to note that the SNA basic packages will use their own data object, either 'igraph' or 'network'. Therefore, once the data is encoded in the package specific data object, the sparse matrix packages will not be useful unless data is converted back into a matrix format for operation. The sparse matrix packages are used for gross data manipulation and not for SNA specific calculations such as centrality values. Performance comparisons of the matrix packages are outside the scope of this paper.

Datasets

The final category of packages contains packages that primarily provide sample datasets. These packages are often used to support training and education of the R community. They are shown in Table 10.

Table 10. Packages with Sample Data Sets

Package	SNA Pkg	Application
historydata	n/a	Sample data sets intended for historians learning R to include network analysis with R
igraphdata	igraph	Datasets for use with igraph
networkDynamicData	sna	A collection of dynamic network data sets from various sources and authors represented as networkDynamic formatted objects
sand	igraph	Datasets for use with igraph for statistics application

Network Analysis of R SNA Packages

A simple network analysis of R packages shows two clusters surrounding the two basic SNA packages. Figure 4 shows a network visualization of the above R packages. Nodes represent packages and links represent dependency relations between them. Isolate nodes, representing packages with no dependencies to other listed packages, are omitted from the visualization. The node color indicates the category of R package. It is important to note that 'igraph' is both an SNA package and a network data object, while 'sna' is an SNA package that uses the 'network' network data object. Therefore, 'sna' and 'network' would need to be merged into a single node for a more equitable comparison. It is clear to see the two clusters present in Figure 4. The packages that serve as bridges between the two clusters are 'qgraph', 'keyplayer', 'degreenet', 'lvm4net', 'bipartite', and 'intergraph'. It should be noted that the purpose of 'intergraph' is to convert data objects between 'igraph' and 'network'. The majority of the packages in the sna-network cluster are statistical analysis packages with several visualization packages that are specifically intended for use with 'statnet' and 'sna'. By contrast, the 'igraph' cluster contains more community detection, subgroup analysis packages as well as most of the niche SNA packages. All of the social media analysis packages reside in the 'igraph' cluster.

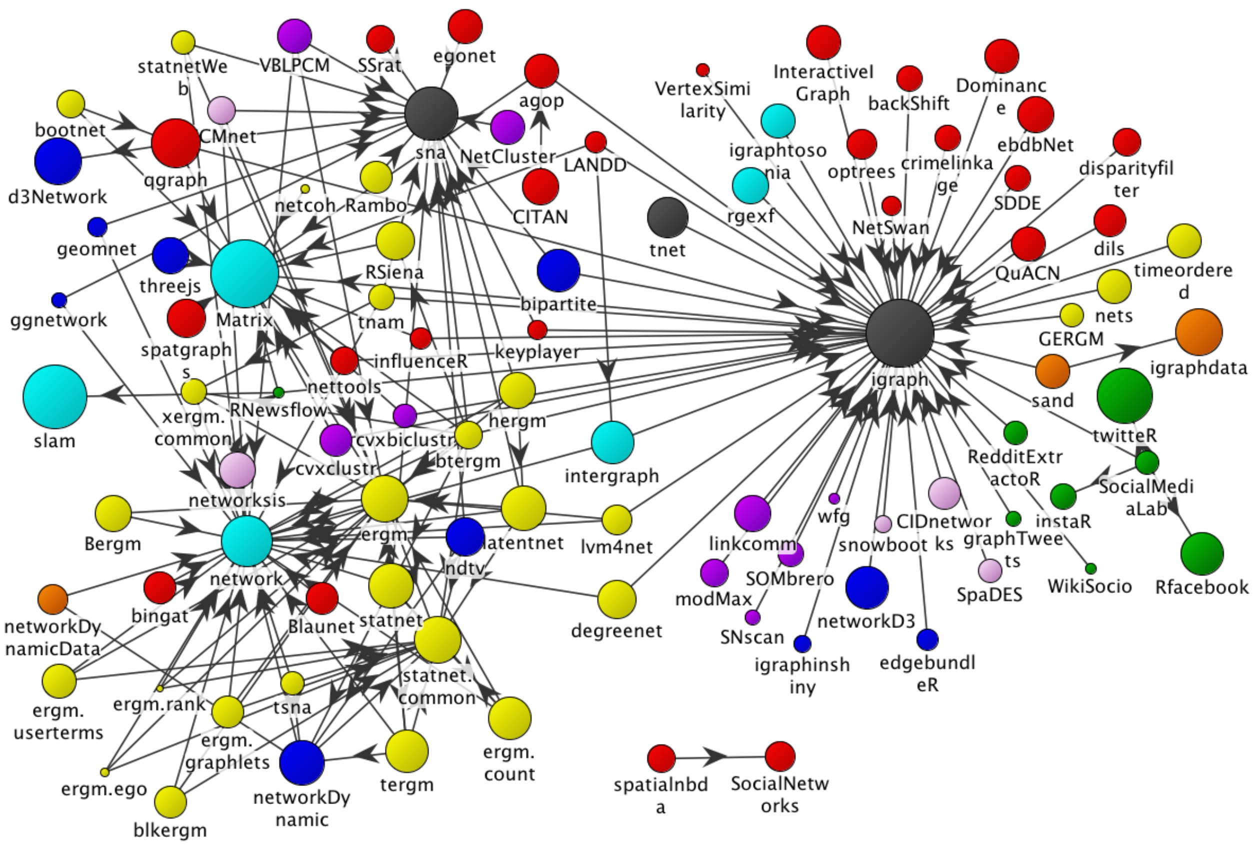
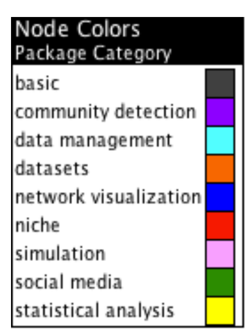


Figure 4. SNA packages in R. A directed edge between package A and package B means package A depends on or imports packages B. Nodes size represents log of downloads according to the RStudio CRAN mirror, and node color represents the category of the package.



It is important to recognize that just because a particular package might depend upon other packages in the 'sna'-'network' or 'igraph' clusters does not mean it cannot be used with both. In some cases packages perform operations on matrix objects that are neither 'igraph' nor 'network' objects. In other cases 'intergraph' can be used to convert between data objects. The purpose of the network analysis of packages is to identify the intended purpose and availability of packages in R. There are well over 100 packages of value for SNA application. Those packages that tend to rely on 'sna' and 'network' are typically used for the statistical analysis of networks. Those packages that tend to rely on 'igraph' include community detection and subgroup analysis, better network visualization features, or contain niche functions. While social media analysis packages tend to rely on 'igraph', they are more concerned with importing data from a particular social media application program interface (API) than with any particular analytic package. Data from these packages are primarily used with 'igraph'. However, they can be readily converted to a network data object for use with 'sna' and the related statistical analysis packages.

Key Applications

R is a language for statistical computing and can be used for most social network analysis computation and visualization. R provides access to the latest advances in the statistical analysis of networks, such as exponential random graph models (ergm) and stochastic actor oriented models (also known as SIENA models). R can also take advantage of advanced visualization packages. The availability of such a diverse and complete set of packages and functions make R an attractive environment for conducting social network analysis.

Future Directions

R is a powerful tool for social network analysis. While its command-line interface has a steeper learning curve, it maintains a great deal of flexibility for the competent analyst. Perhaps the biggest challenge for an analyst is to know where to look for existing code. R hosts certain sites called "CRAN Task Views." These sites provide a general description of a broad class of analytic problems and identify all of the R packages that support that class of analytic problems. As of the writing of this paper, there are 33 CRAN Task View pages, but none devoted to social network analysis, network analysis, or network science. A CRAN Task View, possibly organized based on this paper, would be useful to the community of R users investigating SNA. In the interim, this paper provides an overview of the types of packages available for use in R.

Many R packages provide redundant capability, leaving users a difficult choice on deciding which set of packages they should use. In terms of basic performance, 'igraph' tends to outperform 'sna' for most of the common SNA functions. The 'igraph' package has greater compatibility with a range of other packages, especially for working with social media data or community detection packages. The chief strength of the 'sna' package is compatibility with the majority of packages for statistical analysis of networks. Given the availability of the 'intergraph' package, it may be advantageous for a user to primarily use 'igraph' and convert the 'igraph' objects to 'network' objects for statistical analysis, but otherwise rely on 'igraph' for basic network analysis, visualization, and community detection.

Another issue that analysts may have with using R as an SNA tool is the availability of user support and help files. A quick search of the internet, using search terms such as "network", "R", and "tutorial" will reveal a number of tutorials posted by the user community. An analyst that is new to R, however, is unlikely to notice the differences between 'igraph' and 'sna'. They may also have difficulty becoming aware of less popular packages. Perhaps the INSNA website will host a web page that provides social network analysts a link to an SNA CRAN Task View or a similar page that identifies SNA related R packages and tutorials, possibly patterned after this paper. This service would allow R developers to communicate the tools that they offer the community. It would allow social network analysts a more direct way to stay abreast of new packages and advanced analytics that are available to them.

Those analysts that use R must be careful to understand the dependencies and conflicts that may occur when loading R packages. As of the writing of this paper, igraph and statnet/sna are not compatible. Users should consider detaching unneeded packages to reduce the risk of conflicts.

Institutions of higher education should consider R methods courses to empower students to use and develop R. It can be frustrating for a student to learn powerful analytic methods, only to find that the tools for accessing them are cost prohibitive after their course ends. With R, analytic tools are free. Students that become proficient in R can maintain their analytic capability for use in other courses, independent research, and in support of clients.

Cross References

Actor-Based Models for Longitudinal Networks

Exponential Random Graph Models

Flickr and Twitter Data Analysis

Network Analysis

Siena: Statistical Modeling of Longitudinal Network Data

Acknowledgements

This work was supported, in part, by the U.S. Army Research Laboratory. The authors would like to thank the many social network professionals who contributed comments and advice via the SOCNET list-serv, personal email, and during the 2016 International Sunbelt Social Networks Conference.

References

Barabási, A. L., & Albert, R. (1999). Emergence of scaling in random networks. *science*, *286*(5439), 509-512.

Carley, K. M. (1999). On the evolution of social and organizational networks. *Research in the Sociology of Organizations*, *16*(0).

Carley, K., & Krackhardt, D. (1998). A PCANS model of structure in organizations. In *Symposium on Command and Control Research and Technology* (pp. 113-119).

Çetinkaya-Rundel, M., Banks, D., Rundel, C., Clyde, M., (2016, August 26). *Statistics with R Specialization.* Retrieved from https://www.coursera.org/specializations/statistics

Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association*, *74*(368), 829-836.
Freeman, L. C. (1977). A set of measures of centrality based on betweenness. *Sociometry*, 35-41.

Erdös, P., & Rényi, A. (1959). On random graphs, I. *Publicationes Mathematicae (Debrecen)*, *6*, 290-297.

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social networks*, *1*(3), 215-239.

Freeman, L. C., Roeder, D., & Mulholland, R. R. (1979). Centrality in social networks: II. Experimental results. *Social networks*, *2*(2), 119-141.

Ihaka, R., & Gentleman, R. (1996). R: a language for data analysis and graphics. *Journal of computational and graphical statistics*, *5*(3), 299-314.

McCulloh, I., Armstrong, H., & Johnson, A. (2013). *Social network analysis with applications*. John Wiley & Sons.

Peng, R., Caffo, B., Leek, J., (2016, August 26). *Data Science Specialization.* Retrieved from https://www.coursera.org/specializations/jhu-data-science

Valente, T. W. (2012). Network interventions. *Science*, *337*(6090), 49-53.

Wasserman, S. S. (1977). Random directed graph distributions and the triad census in social networks†. *Journal of Mathematical Sociology*, *5*(1), 61-86.

Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications* (Vol. 8). Cambridge university press.

Wasserman, T., & Capra, E. (2007). Evaluating software engineering processes in commercial and community open source projects.