

Modeling the Impact of R Packages Using Dependency and Contributor Networks

Gizem Korkmaz
Virginia Tech
gkorkmaz@vt.edu

Claire Kelling
Penn State University
cek32@psu.edu

Carol Robbins
National Science Foundation
crobbins@nsf.gov

Sallie A. Keller
Virginia Tech
sallie41@vt.edu

Abstract—This project aims to identify the factors that affect the impact of Open Source Software (OSS), measured by number of downloads and citations, with a case study of R packages. We generate the dependency and contributor networks of the packages using data collected from Depsy.org, and develop statistical models that use the network characteristics, as well as author and package attributes. We find that there are common network and package attributes that are important in determining both the number of downloads and citations of a package, including the number of authors, indegree, outdegree, and several other properties.

I. INTRODUCTION

Open Source Software (OSS), defined by Open Source Initiative [1], is a computer software with its source code made available with a license in which the copyright holder provides the rights to study, change, and distribute the software to anyone and for any purpose. OSS is everywhere, both as specialized applications nurtured by devoted user communities, and as digital infrastructure underlying platforms used by millions daily. This type of software is developed, maintained, and extended both within and outside of the private sector, through the contribution of people from universities, government research institutions, nonprofits, and individuals. Examples include Linux, Apache, Python, and R. Despite its ubiquity and extensive use, reliable measures of the scope and impact of this software outside of the business sector are scarce, and the activities around its development are currently missing or not well measured in existing innovation statistics.

While the extent and impact of open source software is currently unknown, recent estimates suggest that its magnitude is significant. The Apache server, developed initially at the National Center for Supercomputing Applications at the University of Illinois, is estimated to be equivalent to between 1.3 and 8.7 % of the stock of prepackaged software currently accounted for in US private fixed investment [2].

Current and comprehensive survey data do not exist for the contributions of OSS, however, as OSS is disseminated online, a wealth of information is available to be scraped, both metadata and information embedded in repositories and in the code and headers of the software programs themselves. Our contribution is to show how these data can be used to develop measures of open source software that are complementary to GDP measures of private investment, shedding light on impact of open source software innovation currently not well understood.

Researchers have started stressing the importance of quantifying coding *impact* in the scientific community [3], [4]. Citations are used to measure the impact of patents and publications which are most frequently used to estimate the research outputs of universities and government institutions. However, it is not a common practice (yet) to cite the software packages that are used in published studies. Howison and Bullard (2016) randomly selected 90 biology papers and found that two-thirds mentioned the software used, but less than half of those papers actually cited the package [5]. Moreover, even though developers write papers that describe their software, researchers may not know which paper to cite because software packages often have multiple articles associated with them [4].

Depsy.org, developed by Impactstory [6], is a website that compiles R and Python packages to measure the impact of software built by academics. Depsy tracks the following: (i) literature reuse via mentions within papers (citations), (ii) software reuse via reverse dependencies, (iii) downloads from software repositories, and calculates an *impact score* as the average of these three measures. This free tool creates a very rich source of data (including information about contributors, commits, downloads, citations) that could be used to study the *development activities, pattern of complex interactions among community members and contributors, and package dependencies* to estimate determinants of impact.

In this paper, we use data collected from Depsy on R packages to generate the dependency and contributor networks of this OSS community, and develop statistical models that use the features of the packages, developers and the structural properties of the networks to estimate their impact, measured by the number of downloads and citations. We find that network centrality measures such as the indegree and outdegree in the dependency network and the closeness and betweenness centralities, and clustering as well as package attributes such as the number of stars, the number of commits, and the number of authors have all significant influence on both measures of impact.

The rest of the paper is organized as follows. The next section gives background information on R. Section III summarizes the related work. In Section IV, we describe the data source used to develop the statistical models, the networks generated using the package and contributor connections, and the statistical models, respectively. In Section V, the model results are presented. Section VI discusses our findings.

II. BACKGROUND

Since its inception, R has been one of the fastest growing programming languages [7]. R has gained ubiquity in work surrounding statistical analysis and mathematical modeling. Not surprisingly, R has gained traction in nearly every industry by firms that want to optimize and grow their businesses. Now that R is so widely used, it is important to understand the economic impact of this programming language.

R, based closely off of the prior statistical language S [8], was developed by two professors at the University of Auckland in the 1990's to teach statistical methods to their students [9]. It was released in 2000 and since then, users around the world have developed packages that are shared with the whole R community. The packages must pass a certain set of standards before being committed to the software [10].

After a few years of development, and well before its official release to the public, the two professors behind R were receiving too many bug complaints from a select group of professors they asked to test the software. So, the Comprehensive R Archive Network (CRAN) was developed at TU Vienna for users to commit their improvements [9]. CRAN is still used today and is growing in importance. This global network holds the most recent releases of R code and documentation.

R is released by the Free Software Foundations General Public License. This has three major implications. First, there are no regulations on how R can be distributed. This lowers the barriers of entry for users with less capital to invest, allowing a larger and stronger user base. Second, any release of R can be acquired for free and redistributed. Third, this license disallows development of proprietary versions of this software, which keeps R true to its original open source nature.

A key reason why R has such a loyal following is the ability to make and share packages. In this context, R packages are, "...fundamental units of reproducible R code. They include reusable R functions, the documentation that describes how to use them, and sample data" [11]. R packages are relatively easy to start developing, however once a user wants to commit their package back to a major repository (CRAN, Bioconductor, Github), they must pass a certain set of standards to maintain re-usability. This has led to the development and distribution of over 10,000 readily available packages, each with a specific purpose. As the R community is consistently receiving more packages, R's power and functionality is also consistently growing, and this leads to a growing user base across a broad range of sectors. This calls for developing methods to understand R's impact in the economy.

III. RELATED WORK

Open source software development exhibits a complex social structure that influence contribution beyond technical merit [12], [13]. Hence, it is closely related to collaborative production in academic research. The recent trend of this topic in many disciplines (e.g., economics, sociology, physics, management, mathematics) is to capture the patterns of collaboration using a network: a web of collaborative interactions in which two researchers are linked if they work together

on a project or coauthor a paper (e.g., [14]–[22]). There exists a growing literature, both empirical and theoretical, on collaboration and coauthorship networks (e.g., [15], [16], [23]–[29]). A review of these studies can be found at [30].

Our work is most closely related to the studies that focus on the relationship between factors of productivity and the network properties. Some of these studies suggest that productivity is affected by both the number of links and network structure due to communication effectiveness and exchange and flow of information [18], [31], [32]. Moreover, the local network structure and the centrality of an individual in the network will affect the outcome [24], [33]–[35].

Empirical studies have found correlation between the centrality measures in coauthorship networks and productivity, research performance of authors, institutions and countries. [36]–[43]. Yan and Ding (2009) [37] used 20 years of data from 16 journals and found that among the centrality measures (closeness, betweenness, degree and pagerank centrality), betweenness centrality had the highest correlation with citation counts. Abbasi et al. (2011) [43] used *g-index*, an extension of widely used *h-index* [44], as a performance measure and found that only normalized degree centrality, efficiency and average ties strength had significant influence on impact. Our paper is consistent with these studies as we find that network centrality measures such as indegree and outdegree in the dependency network and the closeness and betweenness centralities, and clustering have all significant influence on impact (both number of downloads and citation counts).

IV. METHODOLOGY

A. Data

We gathered all of the R packages listed on CRAN (10,926 packages listed as of July 11th, 2017), and scraped the characteristics from the JSON page affiliated with each R package from Depsy. In total, information about 9,810 packages that were on Depsy (last update in Sept. 2015) and around 24,000 affiliated contributors were collected.

The dataset includes several metrics about each package including its contributors, number of commits, number of downloads, number of citations, packages reused by, stars (identifying active development), and tags (only 25% of the packages have tags). Figure 1 illustrates a word cloud of the packages in our dataset, where the size indicates the number of contributors. The package with the highest number of contributors was *xgboost*, with 117, followed closely by *rvowpalwabbit*, *knitr*, *dplyr*, and *ggplot*. The average number of contributors was 2.69, with many packages only having 1 contributor (3,622 packages).

We also collected information on developers such as number of packages contributing to (both R and Python), tags, and specific packages they support, and percent contribution to the package. We also gathered information on their main language, as identified by Depsy, as well as their roles on their projects, such as author and/or Github owner.



Fig. 1. A word cloud of the R packages in Depsy. The size of the package name indicates the number of contributors.

As mentioned before, we use citations and downloads as our impact metrics. We present some of the top downloaded and top cited packages in our dataset in Table I. The average number of downloads for the R packages in Depsy is approximately 58,000 and the average number of citations is approximately 6.83. Histograms of the distributions of the number of downloads and the number of citations are presented in Figure 2. In both plots, we have cutoff the distribution at a number in the tail to visualize the main section of the distribution. We observe that the distributions of both of these metrics are skewed to the right.

Top Downloaded Packages	Number of Downloads	Top Cited Packages	Number of Citations
Rcpp	6,683,565	vegan	4,275
ggplot2	6,255,500	lme4	4,023
stringr	5,366,703	nlme	2,916
plyr	5,345,308	ggplot2	1,702
digest	5,251,824	gplots	1,307

TABLE I
TOP DOWNLOADED AND CITED PACKAGES IN R

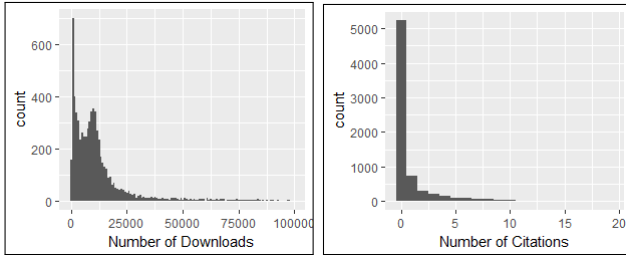


Fig. 2. Histograms of number of downloads (left) and number of citations (right) for all R packages used in our analysis. Both of these metrics are skewed to the right.

B. Dependency Network

Depsy provides information on the packages that are required for the development of other packages (i.e., dependencies). Models developed to measure the value and cost of OSS projects need to take into account these dependencies. We have generated the network of R packages where a directed edge $i \rightarrow j$ indicates that the package j requires i to be installed to function. We obtain a network with 7,389 nodes and 20,235 directed edges.

The definitions of the network measures mentioned throughout are given in Table II, and Table III summarizes the properties of the dependency network. The average degree (in-degree and out-degree) of the dependency network is found as 2.74. The network is composed of 56 weakly connected components (and 7,388 strongly connected components) with the largest component of 7,261 packages.

Network Measure	Definition
Out-degree (in-degree, resp.)	total number of outgoing (incoming) links, respectively
Closeness centrality	the inverse average distance from a given node to all other nodes in the network
Betweenness centrality	the number of shortest paths that pass through a node
Eigencentrality	takes into account the centrality of the neighbors
PageRank	(i) the number of links the node receives (ii) the number of links outgoing from neighbors (iii) the centrality of its neighbors.
Clustering coefficient	the degree to which a node's neighbors are connected.

TABLE II
NETWORK MEASURES AND DEFINITIONS

Nodes	Edges	Avg. Degree	Diameter	Avg. Clustering Coefficient	Avg. Path Length	Connected Components
7,389	20,235	2.74	9	0.07	2.4	56

TABLE III
STRUCTURAL FEATURES OF THE DEPENDENCY NETWORK

Figure 3 illustrates the full network where the size of the node is proportional to the outdegree (number of packages that reuses the package in consideration) and the different colors indicate communities identified using the modularity algorithm [45] implemented in Gephi [46]. For illustrative purposes only the communities that make up of at least 5% of the nodes are shown in the figure (described in the figure legend). The community in purple includes packages that are used for data wrangling, exploration and visualization such as ggplot, dplyr, plyr and data.table. This community with 572 nodes (7.74% of all nodes) and 957 edges (4.73% of all edges) is shown separately in Figure 4. The size of the node indicates the out-degree, and the color represents the closeness centrality (how close a node is to every other node), i.e., closeness centrality increases as it approaches to 1. We observe that although ggplot has the highest outdegree in this community, its closeness centrality is around 0.5. There are many nodes that are closer (on average) to the rest of the nodes in the network, therefore, we use closeness centrality as one of the variables in our models in addition to outdegree, which captures *local* centrality.

C. Contributor Network

To capture the influence of the contributors on the impact of a package, we also generate the contributor network where an undirected edge between i and j indicates that user i and j contribute to the same R package. We obtain a network with 12,340 nodes, 90,030 edges, and 1,651 components. Table IV summarizes the structural properties of this network.

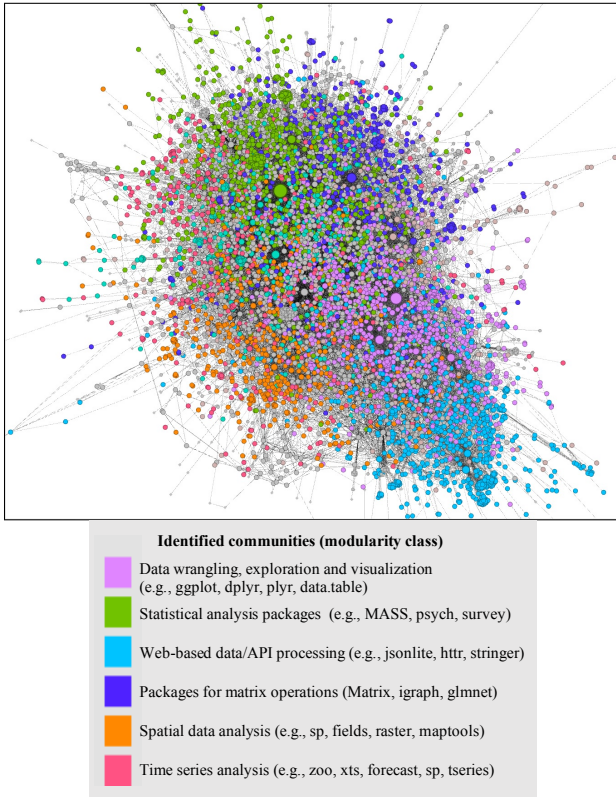


Fig. 3. Dependency network of R packages on Depsy. The network includes 7,547 nodes and 20,641 directed edges. The size of the node is proportional to the out-degree (number of packages that reuses the package) and the different colors indicate communities identified using the modularity algorithm [45]. For illustrative purposes only the communities that make up of at least 5% of the nodes are shown in the figure. The communities correspond to different uses of R packages as described in the legend.

Nodes	Edges	Avg. Weighted Degree	Diameter	Avg. Clustering Coefficient	Avg. Path Length	Connected Components
12,340	90,030	15.4	19	0.84	4.4	1,651

TABLE IV
STRUCTURAL FEATURES OF THE CONTRIBUTOR NETWORK

In Figure 5, we focus on the largest component with 6,502 nodes which makes up of 52.7% all the nodes in the contributor network. For illustrative purposes, we filtered out the nodes that have weighted degree smaller than 90.

As we can see, some of these contributors are people, and some are entities, e.g., Rstudio.

D. Modeling Framework

Our goal is to identify the factors that affect the impact of R packages. We measure impact in two ways: (i) number of downloads, (ii) number of citations. We fit two Quasi-Poisson models for the number of downloads and the number of citations, as dependent the variables, y . We let $E(y) = \mu$ and $Var(y) = \theta\mu$. We assume that $y_i \sim \text{Poisson}(\mu_i, \theta)$ and let the mean μ_i for the i^{th} observation vary as a function of the p covariates as follows: $\mu_i = e^{\beta_0 + \beta_1 x_{1,i} + \dots + \beta_p x_{p,i}}$.

We consider the following variables as p covariates, i.e., determinants of impact: number of authors, number of commits,

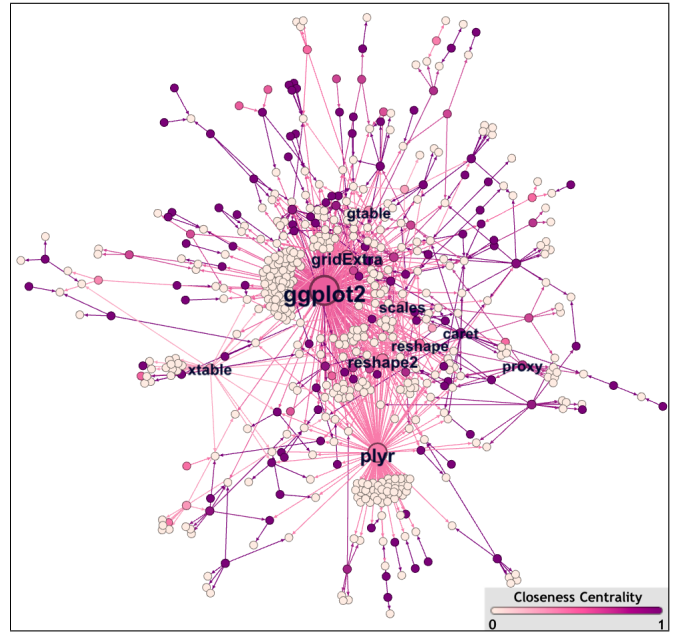


Fig. 4. A subgraph of the dependency network of R packages on Depsy. The community in purple from Fig. 3 is illustrated. The network includes 572 nodes (7.74%). The size of the node indicates the out-degree (the top nodes are labeled), and the color represents the closeness centrality (measures the inverse average distance from a given node to all other nodes in the network): the darker nodes have higher closeness centrality.

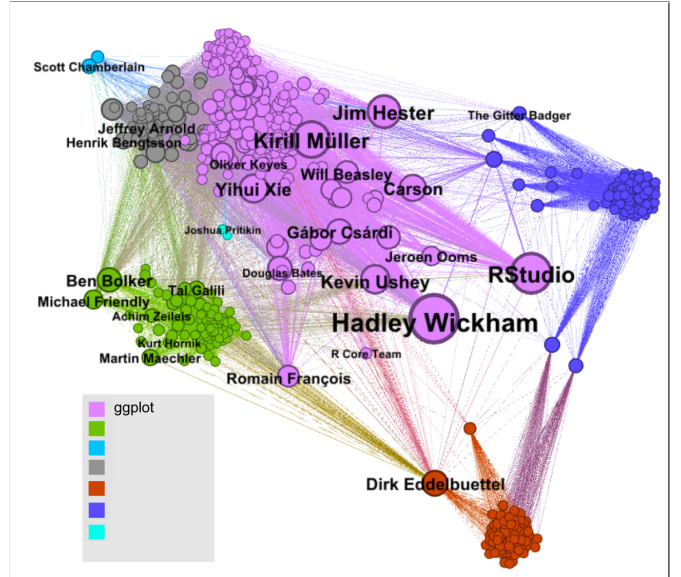


Fig. 5. the legend will be filled out - needs some investigation to these people & packages A snapshot of a subgraph of the contributor network of R packages on Depsy. We focus on the component with the largest size (6,502 nodes (52.7%)) and illustrate nodes with weighted degree larger than 90. The colors correspond to communities identified using the modularity algorithm [45] and the size of the node indicates the weighted degree (the top nodes for each community are labeled).

tag indicator, number of stars are used as package features, the network measures used are given in Table II which

include characteristics of the packages and developers¹. For each package, we calculate weighted average of the degrees and centralities of the authors based on their percent contributions to the package (given in Depsy).

V. FINDINGS

As mentioned above, we develop two Quasi-Poisson models with the number of downloads and the number of citations as the dependent variables. This is because of a concern for overdispersion in our response variable, which is also a count. The Quasi-Poisson model is a common method for dealing with overdispersed count data [47]. We check for overdispersion by comparing the values of the fitted means against the fitted variance. If the slope is 1, then there is little concern for overdispersion, and we can use a normal Poisson model. If the slope is linear, but not equal to one, then we have reason to think a Quasi-Poisson model would be a better fit for our data. This can be seen in the model assumptions, where we let $E(y) = \mu$ and $Var(y) = \theta\mu$, or that the variance is proportional to the mean.

In Figure 6, we plot the fitted mean against the fitted variance for our fitted models. We see that for both the number of downloads and the number of citations, the fitted variance is proportional to the fitted mean, but the slope of the line is not equal to one. This is especially true for the number of downloads. Therefore, we proceed by presenting the results of our Quasi-Poisson fitted model.

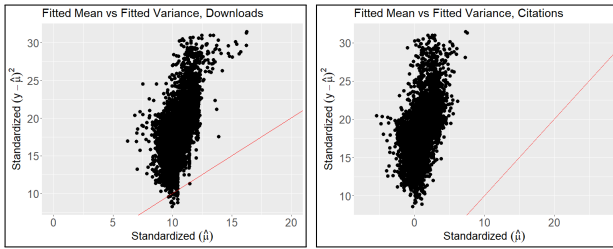


Fig. 6. Plots of fitted mean vs fitted variance for the number of downloads and citations

In Table V, we see the coefficient estimates and the p-values for all covariates that were included in the Quasi-Poisson models for the number of downloads and the number of citations as our response variables. There are a number of variables that are significant for both the number of downloads and the number of citations. Namely, the tag indicator variable (that shows the package has been tagged), number of stars, the number of commits, number of authors, indegree, outdegree, the closeness and betweenness centrality, and clustering are all significant for both number of downloads and the number of citations.

VI. CONCLUSION

We find that package features such as number of commit and authors positive relationship with impact. This is intuitive since

¹Other network characteristics, such as harmonic closeness centrality, and eccentricity were also considered but were removed from our final model because of their high correlations with measures included in the models.

	Downloads		Citations	
	Estimate	Pr(> t)	Estimate	Pr(> t)
(Intercept)	10.35	0.00	1.67	0.00
tag_ind	0.96	0.00	1.39	0.00
num_stars	-0.00	0.00	-0.00	0.00
num_commits	0.00	0.00	0.00	0.00
num_authors	0.04	0.00	0.03	0.00
indegree	-0.09	0.00	0.17	0.00
outdegree	0.00	0.00	0.00	0.00
closenesscentrality	0.77	0.00	1.68	0.00
betweennesscentrality	0.00	0.00	0.00	0.00
eigencentrality	-2.55	0.02	-2.11	0.18
pageranks	-602.29	0.63	-12471.87	0.00
clustering	-1.79	0.00	-3.39	0.00
w_auth_cent	-0.50	0.00	-0.21	0.41
w_auth_degree	0.01	0.00	-0.00	0.33

TABLE V
MODEL COEFFICIENTS AND P-VALUES FOR NUMBER OF DOWNLOADS

a package with more authors is more likely to be published and cited in more papers or it may be more widely used because of the social network inherent in the open source community. Similarly, number of more commits might indicate better quality and a more advanced package or relevant to more fields than one. It might have multiple functions or features that are more widely used, and therefore have a large impact. Moreover, when we consider the network features, we find that not only degree as a local centrality metric, but also measures such as betweenness and closeness centrality are significant in our analysis. This shows that there is a lot to learn by studying the complex interactions in the open source community.

ACKNOWLEDGMENTS

This material is based on work supported by U.S. Department of Agriculture (58-3AEU-7-0074) and the National Science Foundation under IGERT Grant DGE-1144860, Big Data Social Science. We thank Social and Decision Analytics Laboratory members, especially Stephanie Shipp and Daniel Chen, for feedback and insights. We acknowledge the Data Science for the Public Good Program participants Sayali Phadke and Ronnie Fecso, as well as our undergraduate researchers Alex Gagliano, John Higgins, and Romcholo Macatula.

REFERENCES

- [1] Open Source Initiative. (1998) <https://opensource.org/osd>. Accessed: 2018-02-01.
- [2] S. Greenstein and F. Nagle, "Digital dark matter and the economic contribution of Apache," *Research Policy*, vol. 43, no. 4, pp. 623–631, 2014.
- [3] J. Howison, E. Deelman, M. J. McLennan, R. Ferreira da Silva, and J. D. Herbsleb, "Understanding the scientific software ecosystem and its impact: Current and future measures," *Research Evaluation*, vol. 24, no. 4, pp. 454–470, 2015. [Online]. Available: <http://dx.doi.org/10.1093/reseval/rvv014>
- [4] D. Singh Chawla, "The unsung heroes of scientific software," *Nature News*, vol. 529, no. 7584, p. 115, 2016.
- [5] J. Howison and J. Bullard, "Software in the scientific literature: Problems with seeing, finding, and using software mentioned in the biology literature," *Journal of the Association for Information Science and Technology*, vol. 67, no. 9, pp. 2137–2155, 2016.
- [6] Impact Story. (2012) <https://impactstory.org>. Accessed: 2018-04-26.

- [7] B. Muenchen, "Rs growth continues to accelerate," 2017. [Online]. Available: <http://r4stats.com/2016/04/19/rs-growth-continues-to-accelerate/>
- [8] B. Venables, D. Smith, R. Gentleman, and R. Ihaka, *Notes on R: A programming environment for data analysis and graphics*. University of Auckland, 1998.
- [9] R. Ihaka, "The R project: A brief history and thoughts about the future," 2017. [Online]. Available: <https://www.stat.auckland.ac.nz/~ihaka/downloads/Massey.pdf>
- [10] CRAN, "The comprehensive R archive network," 1997. [Online]. Available: <https://cran.r-project.org/>
- [11] H. Wickham, "R packages: Organize, test, document, and share your code," 2015. [Online]. Available: <http://r-pkgs.had.co.nz/>
- [12] J. Tsay, L. Dabbish, and J. Herbsleb, "Influence of social and technical factors for evaluating contribution in github," in *Proceedings of the 36th international conference on Software engineering*. ACM, 2014, pp. 356–366.
- [13] N. Ducheneaut, "Socialization in an open source software community: A socio-technical analysis," *Computer Supported Cooperative Work (CSCW)*, vol. 14, no. 4, pp. 323–368, 2005.
- [14] S. Goyal, M. J. Van Der Leij, and J. L. Moraga-González, "Economics: An emerging small world," *Journal of political economy*, vol. 114, no. 2, pp. 403–412, 2006.
- [15] M. E. Newman, "Scientific collaboration networks. i. Network construction and fundamental results," *Physical review E*, vol. 64, no. 1, p. 016131, 2001.
- [16] C. Bosquet and P.-P. Combes, "Do large departments make academics more productive? Agglomeration and peer effects in research," 2013.
- [17] K. A. Anderson, M. Crespi, and E. C. Sayre, "Linking behavior in the physics education research coauthorship network," *Physical Review Physics Education Research*, vol. 13, no. 1, p. 010121, 2017.
- [18] J. Moody, "The structure of a social science collaboration network: Disciplinary cohesion from 1963 to 1999," *American sociological review*, vol. 69, no. 2, pp. 213–238, 2004.
- [19] B. Uzzi and J. Spiro, "Collaboration and creativity: The small world problem," *American journal of sociology*, vol. 111, no. 2, pp. 447–504, 2005.
- [20] F. J. Acedo, C. Barroso, C. Casanueva, and J. L. Galán, "Co-authorship in management and organizational studies: An empirical and network analysis," *Journal of Management Studies*, vol. 43, no. 5, pp. 957–983, 2006.
- [21] W. W. Powell, K. W. Koput, and L. Smith-Doerr, "Interorganizational collaboration and the locus of innovation: Networks of learning in biotechnology," *Administrative science quarterly*, pp. 116–145, 1996.
- [22] J. W. Grossman, "The evolution of the mathematical research collaboration graph," *Congressus Numerantium*, pp. 201–212, 2002.
- [23] L. Ductor, "Does co-authorship lead to higher academic productivity?" *Oxford Bulletin of Economics and Statistics*, vol. 77, no. 3, pp. 385–407, 2015.
- [24] L. Ductor, M. Fafchamps, S. Goyal, and M. J. van der Leij, "Social networks and research output," *Review of Economics and Statistics*, vol. 96, no. 5, pp. 936–948, 2014.
- [25] M. E. Newman, "The structure of scientific collaboration networks," *Proceedings of the national academy of sciences*, vol. 98, no. 2, pp. 404–409, 2001.
- [26] —, "Scientific collaboration networks. ii. Shortest paths, weighted networks, and centrality," *Physical review E*, vol. 64, no. 1, p. 016132, 2001.
- [27] —, "Coauthorship networks and patterns of scientific collaboration," *Proceedings of the national academy of sciences*, vol. 101, no. suppl 1, pp. 5200–5205, 2004.
- [28] —, "Who is the best connected scientist? A study of scientific coauthorship networks," in *Complex networks*. Springer, 2004, pp. 337–370.
- [29] M. König, X. Liu, and Y. Zenou, "R&d networks: Theory, empirics and policy implications," 2014.
- [30] S. Kumar, "Co-authorship networks: a review of the literature," *Aslib Journal of Information Management*, vol. 67, no. 1, pp. 55–73, 2015.
- [31] R. S. Burt, "Structural holes versus network closure as social capital," in *Social capital*. Routledge, 2017, pp. 31–56.
- [32] M. O. Jackson and L. Yariv, "Diffusion of behavior and equilibrium properties in network games," *American Economic Review*, vol. 97, no. 2, pp. 92–98, 2007.
- [33] H. Menzel and E. Katz, "Social relations and innovation in the medical profession: The epidemiology of a new drug," *Public Opinion Quarterly*, vol. 19, no. 4, pp. 337–352, 1955.
- [34] A. Calvó-Armengol, "Job contact networks," *Journal of economic Theory*, vol. 115, no. 1, pp. 191–206, 2004.
- [35] A. Banerjee, A. G. Chandrasekhar, E. Duflo, and M. O. Jackson, "The diffusion of microfinance," *Science*, vol. 341, no. 6144, p. 1236498, 2013.
- [36] S. Lee and B. Bozeman, "The impact of research collaboration on scientific productivity," *Social studies of science*, vol. 35, no. 5, pp. 673–702, 2005.
- [37] E. Yan and Y. Ding, "Applying centrality measures to impact analysis: A coauthorship network analysis," *Journal of the Association for Information Science and Technology*, vol. 60, no. 10, pp. 2107–2118, 2009.
- [38] E. Yan, Y. Ding, and Q. Zhu, "Mapping library and information science in china: A coauthorship network analysis," *Scientometrics*, vol. 83, no. 1, pp. 115–131, 2010.
- [39] Q. Ye, T. Li, and R. Law, "A coauthorship network analysis of tourism and hospitality research collaboration," *Journal of Hospitality & Tourism Research*, vol. 37, no. 1, pp. 51–76, 2013.
- [40] S. Uddin, L. Hossain, and K. Rasmussen, "Network effects on scientific collaborations," *PloS one*, vol. 8, no. 2, p. e57546, 2013.
- [41] S. Kumar and J. M. Jan, "Mapping research collaborations in the business and management field in Malaysia, 1980–2010," *Scientometrics*, vol. 97, no. 3, pp. 491–517, 2013.
- [42] —, "Research collaboration networks of two oic nations: Comparative study between turkey and malaysia in the field of 'energy fuels', 2009–2011," *Scientometrics*, vol. 98, no. 1, pp. 387–414, 2014.
- [43] A. Abbasi, J. Altmann, and L. Hossain, "Identifying the effects of co-authorship networks on the performance of scholars: A correlation and regression analysis of performance measures and social network analysis measures," *Journal of Informetrics*, vol. 5, no. 4, pp. 594–607, 2011.
- [44] J. E. Hirsch, "An index to quantify an individual's scientific research output," *Proceedings of the National academy of Sciences of the United States of America*, vol. 102, no. 46, p. 16569, 2005.
- [45] R. Lambiotte, J.-C. Delvenne, and M. Barahona, "Laplacian dynamics and multiscale modular structure in networks," *arXiv preprint arXiv:0812.1770*, 2008.
- [46] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: An open source software for exploring and manipulating networks," 2009. [Online]. Available: <http://www.aiai.org/ocs/index.php/ICWSM/09/paper/view/154>
- [47] J. M. Ver Hoef and P. L. Boveng, "Quasi-poisson vs. negative binomial regression: How should we model overdispersed count data?" *Ecology*, vol. 88, no. 11, pp. 2766–2772, 2007.