

Online Supplement to

Networks, social influence and the choice among competing innovations: Insights from open source software licenses

Explanation of Network Construction

To assess social proximity, we constructed 22 affiliation networks. Each network contains projects assigned to a particular foundry. An affiliation network is a network of actors connected by their participation in common events, while events are connected by common actors (Wasserman and Faust 1994). In our case, the actors are individual developers and the events are projects. Developers have social ties with one another as a result of working together on the same project and projects are related to one another as a result of sharing developers. Data on projects and their developers are represented using an affiliation matrix, where $A = \{a_{ij}\}$, and a_{ij} is “1” if developer i is involved with project j , and “0” otherwise.

Figure 1 provides an illustrative example with five projects (P1, P2, P3, P4, and P5) and five developers (D1, D2, D3, D4, and D5). The network diagram on the left shows developers in ellipses and projects in rectangles. Lines between ellipses and rectangles designate which developers work on which projects. The calendar month associated with each developer-project link designates when the particular developer joined the particular project. The affiliation matrix on the right in Figure 1 reflects the developer-project relationships shown in the network diagram on the left. A value of one for element (D1, P1) and zero for element (D1, P2) indicates that developer D1 contributes to project P1 but not project P2. A bolded “1” indicates the particular developer was the project administrator for the particular project. The month listed at the bottom of each column indicates when the respective developers joined the project.

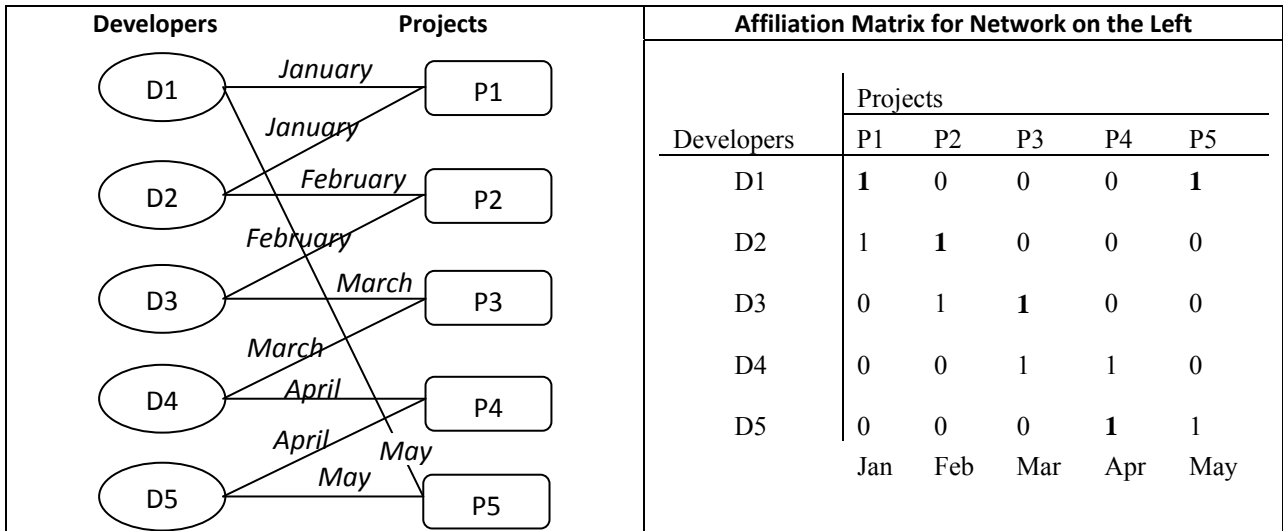


Figure 1. Illustrative Example of an Affiliation Network

Because a project's administrator is responsible for choosing the project's license and is the only member of the project at the time of license choice, the project and administrator are one and the same at the time of license choice. Moreover, our theory suggests an administrator is influenced by his/her social proximity to, and characteristics of, other projects (rather than individual developers). Thus, an administrator's relevant social network at the time of license choice consists of the social ties the administrator-as-project has to previously-established OSS projects. Because the unit of analysis is the project, we assessed social proximity among projects by projecting each developers-by-project affiliation network into its respective unipartite (i.e., one mode) inter-project network. Figure 2 provides the unipartite inter-project network projection of the bipartite network shown in Figure 1. In the network diagram on the left, links between projects correspond to the sharing of two (or more) developers across the projects (as identified in Figure 1). The months associated with each link indicate the month the link was established. The inter-project adjacency matrix on the right reflects the presence and absence of links among projects shown on the left. Given that links are undirected, this matrix is symmetric.

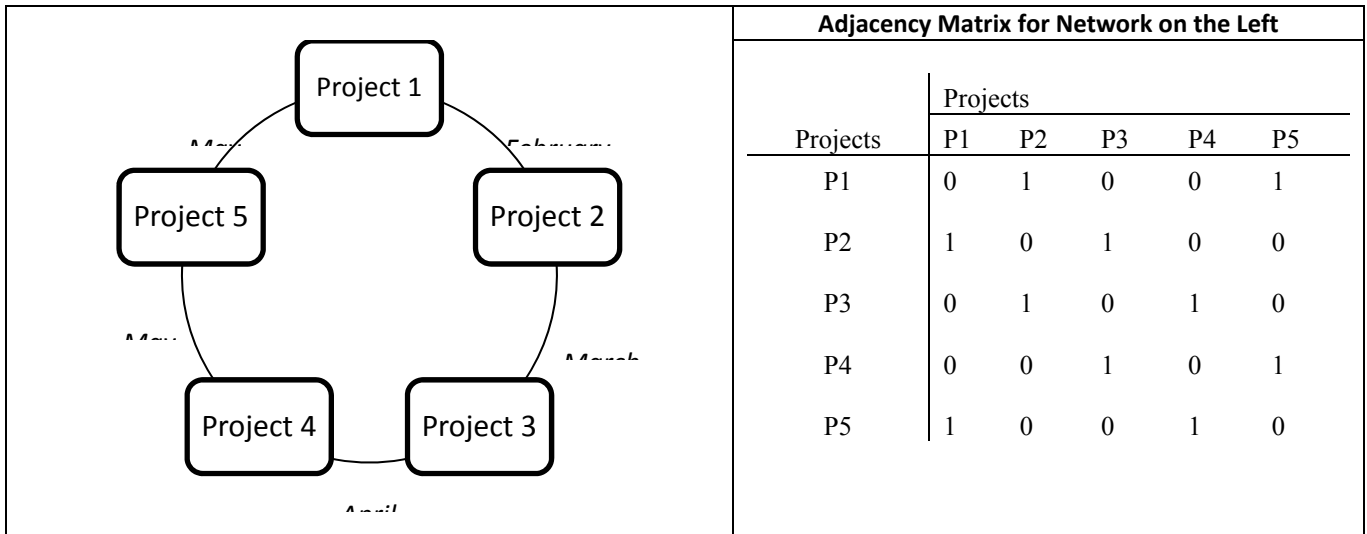


Figure 2. Projection of Bipartite Network in Figure 1 into Unipartite Inter-project Network

Identifying the timing of the formation of new projects is critical to constructing accurate affiliation networks. These networks evolve over time as new projects are established and new and existing developers join and leave projects. A project administrator is responsible for choosing the project's license. The administrator is the only developer associated with a project when it adopts a specific license. Only relationships and projects that exist prior to a focal project can influence the focal administrator's license decision. Constructing the networks in this way eliminates the potential for a simultaneity bias influencing our results because the choice of license temporally follows the formation of relationships with other projects that previously adopted their own licenses. For each focal project, the relevant affiliation network consists of all projects initiated in the respective SourceForge foundry prior to the start of the focal project. Because developers may join projects anytime after they commence, only developers who joined existing projects within the relevant foundry before the start date of the focal project were used to construct the relevant inter-project network. To ensure social ties actually existed among developers on the same project, we excluded all developers who did not actively contribute to the projects for which they volunteered¹.

¹ To ensure that developers associated with projects actively participated in the projects during the period under consideration, we matched their contribution efforts and period of involvement using CVS log files, project communications, and project documentation, which typically lists all contributors. While the presence of a developer's name on the list of contributors provides conclusive evidence of that developer's involvement, only 39% of projects in our sample had project documentation. For projects lacking documentation, to indicate involvement, we required that a developer had sent at least ten messages in project email archives or contributed at least five times to CVS commits in addition to being a member of the project. These conditions ensured that the developers that we consider to be part of a project were involved in the project during the time period under study.

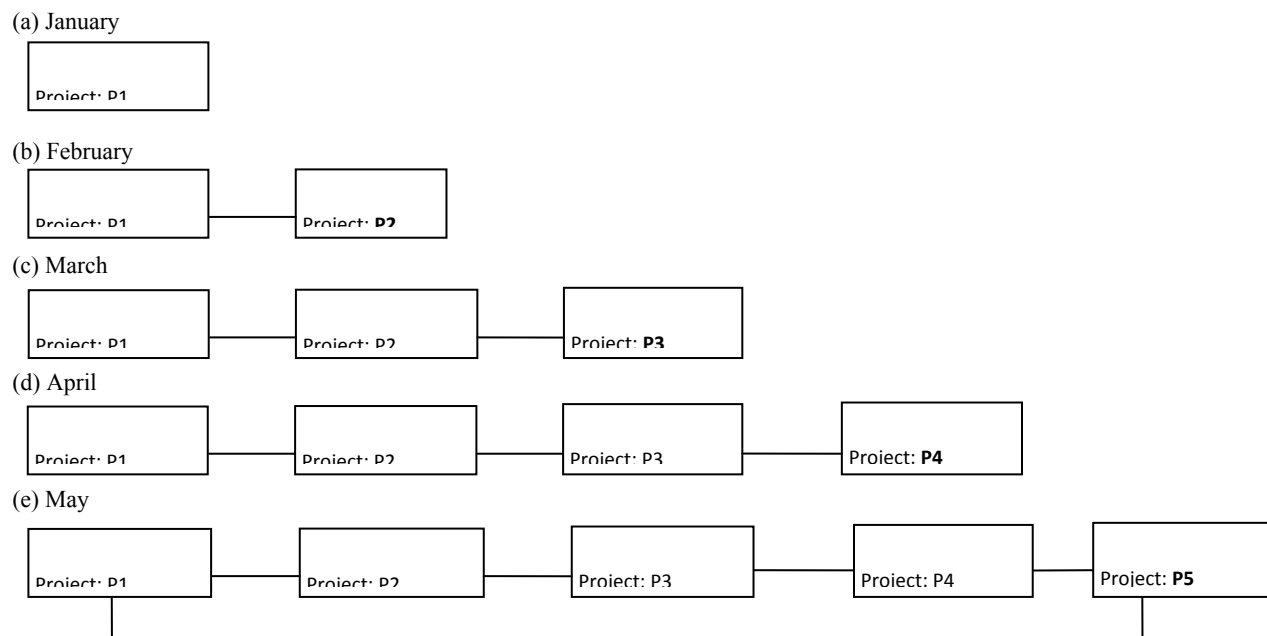


Figure 3. Evolving Sociogram Corresponding to Figure 1
 Licensor and focal projects for each month are represented in bold.

Figure 3 shows how the affiliation network in Figure 1 developed over time. If a developer works on two projects, then there is a link between the projects from sharing a common developer. In January, Developer D1 initiates Project P1 as the project administrator and chooses the projects' license(s), after which D2 joins the project as a developer. In February, Developer D2 initiates Project P2 as administrator and chooses the projects' license(s), after which D3 joins as a developer. As a result of sharing Developer D2, Projects P1 and P2 are connected. In March, Developer D3 initiates Project P3 as administrator and chooses the projects' license(s), after which D4 joins as a developer. As a result of sharing Developer D3, Projects P2 and P3 are connected. In April, Developer D5 initiates Project P4 as administrator and chooses the projects' license(s), after which D4 joins as a developer. As a result of sharing Developer D4, Projects P3 and P4 are connected. Finally, in May, Developer D1 initiates Project P5 as administrator and chooses the projects' license(s), after which D5 joins as a developer. As a result of sharing Developer D5, Projects P4 and P5 are connected, while projects P5 and P1 are connected as a result of sharing Developer D1.

Correcting for the Reflection Problem

We implemented an approach developed by (Bramoulle et al. 2009) to identify the true effect of social influence by accounting for the reflection problem. To understand this approach, consider the following example. Assume there are

four projects: A, B, C and D. A is an established project while B, C and D are new projects that are socially connected to A. The reflection problem implies that B, C and D are connected to A because they share the same *unobserved* preferences for a particular license type. The probability that B, C and D choose a particular license (or not) is a function of the license A has chosen and B, C and D's respective unobserved license preferences. Because B shares the same license preference as A, as do C and D, we can infer by transitivity that B, C and D share the same license preference as each other. We can therefore conclude the probability that C and D choose a particular license is a function of the license A has chosen and B's unobserved license preference. This implies there is a common unobserved effect (i.e., B's unobserved license preference) across three observations. Following (Chamberlain 1980) method, this unobserved effect can be treated as a fixed effect in a logistic model (in which the outcome variable is binary) and it can be conditioned out of the model by appropriate differencing using the three observations. Estimation can then proceed using a standard maximum likelihood logit approach. The identification of the parameter estimates in this conditional logit approach requires within group variation in the outcome variable of interest. In our example, this means that all projects (A-D) cannot have chosen the same license. Bramoulle et al. (2009) build on this result and argue that actors in a network component share the same unobserved effect. A component is a part of a social network in which each actor can reach, either directly or indirectly, every other actor. Pairs of actors in a component that do not choose the same outcome (e.g., same license) provide the necessary variance to condition out the unobserved effect. In implementing this approach, for every month we observe new project license choice, we construct all network components that consist of new projects. We then form all pairwise combinations of new projects in each component that chose different licenses. For example, if one chose all highly restrictive, then the other project will have chosen not all highly restrictive. Following Bramoulle et al. (2009), we assume these pairs have the same unobserved preference for a license type. We then use Chamberlain's differencing method to condition out the component-specific fixed effects.

Table A1. OSI Approved Open Source Software Licenses

License Name	Restrictive?	Highly Restrictive?	Observations in the Sample
Academic Free License (AFL)	N	N	951
Adaptive Public License	Y	N	89
Apache License V2.0	N	N	761
Apache Software License	N	N	1124
Apple Public Source License	Y	N	74
Artistic License	N	N	1355
Attribution Assurance License	N	N	22
BSD License	N	N	5180
Common Development and Distribution License	Y	N	177
Common Public License	Y	N	721
Computer Associates Trusted Open Source License	Y	N	23
CUA Office Public License V1.0	Y	N	5
Eclipse Public License	Y	N	222
Educational Community License	Y	N	135
Eiffel Forum License	Y	N	15
Eiffel Forum License V2.0	Y	N	21
Entessa Public License	N	N	8
EU Data Grid Software License	N	N	14
Fair License	N	N	88
Frameworkx Open License	Y	N	22
GNU General Public License (GPL)	Y	Y	49083
Lesser General Public License (LGPL)	Y	N	7726
IBM Public License	Y	N	119
Intel Open Source License	N	N	45
Jabber Open Source License	Y	N	50
Lucent Public License (Plan9)	N	N	10

Lucent Public License Version 1.02	N	N	8
MIT License	N	N	1315
MITRE Collaborative Virtual Workspace License (CVW)	Y	Y	6
Motosoto License	Y	N	1
Mozilla Public License 1.0 (MPL)	Y	N	315
Mozilla Public License 1.1 (MPL 1.1)	Y	N	835
Nethack General Public License	Y	N	27
Nokia Open Source License	Y	N	19
OCLC Research Public License 2.0	Y	N	2
Open Group Test Suite License	N	N	18
Open Software License	Y	N	678
PHP License	N	N	453
Python License (CNRI Python License)	N	N	186
Python Software Foundation License	N	N	137
Qt Public License (QPL)	Y	N	253
Real Networks Public Source License V1.0	Y	N	8
Reciprocal Public License	Y	N	27
Ricoh Source Code Public License	Y	N	8
Sleepycat License	Y	N	17
Sun Industry Standards Source License (SISSL)	N	N	46
Sun Public License	Y	N	125
Sybase Open Watcom Public License	Y	N	3
University of Illinois/NCSA Open Source License	N	N	40
Vovida Software License 1.0	N	N	8
W3C License	N	N	63
X.Net License	N	N	16
Zlib/libpng License	N	N	340
Zope Public License	N	N	41

A license is coded Restrictive if it includes the "copyleft" clause.

A license is coded as Highly Restrictive if it includes both the "viral" and the "copyleft" clauses.

Table A2. Correlation Statistics for Model 1 (All Licenses Highly Restrictive?) in Table 3.

Variables	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 ALLHR	1														
2 Social Cohesion	0.21	1													
3 Technical Equivalence	0.26	0.11	1												
4 <i>EXP</i>	-0.07	0.10	-0.01	1											
5 <i>RELSIZE</i>	-0.08	0.19	0.05	0.12	1										
6 <i>TENURE</i>	-0.09	0.13	0.09	0.19	0.14	1									
7 <i>DIVERSEXP</i>	0.01	0.05	0.08	0.17	0.09	0.10	1								
8 Social Cohesion X <i>EXP</i>	-0.15	0.24	-0.02	0.22	-0.02	0.06	0.03	1							
9 Social Cohesion X <i>DIVERSEXP</i>	-0.27	0.19	0.05	-0.01	0.01	0.21	0.24	0.17	1						
10 Social Cohesion X <i>RELSIZE</i>	-0.22	0.16	0	-0.02	0.13	0.04	0.07	0.11	0.12	1					
11 Social Cohesion X <i>TENURE</i>	-0.05	0.21	0.04	0.05	0.03	0.14	0.08	0.09	0.07	0.18	1				
12 Technical Equivalence X <i>EXP</i>	-0.07	-0.09	0.15	0.28	0.02	0.03	-0.02	0.15	-0.03	0.08	0.09	1			
13 Technical Equivalence X <i>RELSIZE</i>	-0.04	-0.11	0.17	-0.02	0.29	0.05	-0.04	0.07	0.04	0.17	0.04	0.22	1		
14 Technical Equivalence X <i>TENURE</i>	-0.09	0.12	0.21	0.10	-0.05	0.14	-0.01	0.08	0.06	-0.07	0.21	0.19	0.17	1	
15 Technical Equivalence X <i>DIVERSEXP</i>	0.07	0.08	0.13	-0.06	-0.08	-0.01	0.15	-0.04	0.11	0.02	-0.09	0.23	0.10	0.16	1

Table A3. Correlation Statistics for Model 2 (Some Licenses Highly Restrictive?) in Table 3.

Variables	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 SHR	1														
2 Social Cohesion	0.25	1													
3 Technical Equivalence	0.26	0.06	1												
4 <i>EXP</i>	-0.04	0.11	-0.02	1											
5 <i>RELSIZE</i>	-0.05	0.17	0.04	0.12	1										
6 <i>TENURE</i>	-0.07	0.15	0.09	0.19	0.14	1									
7 <i>DIVERSEXP</i>	0.03	0.03	0.09	0.17	0.09	0.10	1								
8 Social Cohesion X <i>EXP</i>	-0.17	0.22	-0.07	0.24	-0.03	0.07	0.03	1							
9 Social Cohesion X <i>DIVERSEXP</i>	-0.21	0.17	0.05	-0.04	0.02	0.21	0.21	0.21	1						
10 Social Cohesion X <i>RELSIZE</i>	-0.18	0.18	0.02	0.08	0.11	0.05	0.04	0.16	0.19	1					
11 Social Cohesion X <i>TENURE</i>	-0.15	0.11	0.04	0.06	0.04	0.17	0.09	0.12	0.08	0.19	1				
12 Technical Equivalence X <i>EXP</i>	-0.11	-0.05	0.14	0.24	0.05	0.03	-0.03	0.15	-0.02	0.04	0.07	1			
13 Technical Equivalence X <i>RELSIZE</i>	-0.07	-0.07	0.19	-0.03	0.31	0.06	-0.05	0.05	0.08	0.15	0.06	0.27	1		
14 Technical Equivalence X <i>TENURE</i>	-0.12	0.08	0.22	0.06	-0.04	0.18	0.02	0.07	0.09	-0.05	0.19	0.21	0.22	1	
15 Technical Equivalence X <i>DIVERSEXP</i>	-0.02	0.07	0.21	-0.04	-0.09	-0.05	0.17	-0.03	0.11	-0.03	-0.04	0.17	0.10	0.17	1

Table A4. Correlation Statistics for Model 3 (All Licenses Restrictive?) in Table 3.

Variables	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 ALLR	1														
2 Social Cohesion	0.35	1													
3 Technical Equivalence	0.22	0.07	1												
4 <i>EXP</i>	-0.1	0.11	-0.02	1											
5 <i>RELSIZE</i>	-0.08	0.18	0.04	0.12	1										
6 <i>TENURE</i>	-0.08	0.19	0.09	0.19	0.14	1									
7 <i>DIVERSEXP</i>	0.07	0.05	0.09	0.17	0.09	0.10	1								
8 Social Cohesion X <i>EXP</i>	-0.14	0.24	-0.05	0.22	-0.05	0.05	0.05	1							
9 Social Cohesion X <i>DIVERSEXP</i>	-0.11	0.15	0.07	-0.06	0.04	0.19	0.22	0.22	1						
10 Social Cohesion X <i>RELSIZE</i>	-0.15	0.19	0.04	0.07	0.15	0.06	0.05	0.15	0.21	1					
11 Social Cohesion X <i>TENURE</i>	-0.09	0.10	0.05	0.04	0.03	0.21	0.07	0.13	0.16	0.19	1				
12 Technical Equivalence X <i>EXP</i>	-0.21	-0.06	0.17	0.26	0.07	0.05	-0.06	0.17	-0.04	0.03	0.05	1			
13 Technical Equivalence X <i>RELSIZE</i>	-0.10	-0.08	0.20	-0.05	0.24	0.07	-0.05	0.05	0.08	0.17	0.06	0.23	1		
14 Technical Equivalence X <i>TENURE</i>	-0.17	0.05	0.21	0.09	-0.05	0.18	0.02	0.06	0.07	-0.05	0.15	0.24	0.21	1	
15 Technical Equivalence X <i>DIVERSEXP</i>	-0.11	0.02	0.25	0.05	-0.02	-0.04	0.13	-0.05	0.13	-0.04	-0.03	0.16	0.13	0.21	1

Table A5. Correlation Statistics for Model 4 (Some Licenses Restrictive?) in Table 3.

Variables	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 SR	1														
2 Social Cohesion	0.29	1													
3 Technical Equivalence	0.19	0.08	1												
4 <i>EXP</i>	-0.07	0.09	-0.04	1											
5 <i>RELSIZE</i>	-0.08	0.16	0.03	0.12	1										
6 <i>TENURE</i>	-0.07	0.11	0.10	0.19	0.14	1									
7 <i>DIVERSEXP</i>	0.04	0.07	0.11	0.17	0.09	0.10	1								
8 Social Cohesion X <i>EXP</i>	-0.10	0.21	-0.07	0.24	-0.04	0.06	0.05	1							
9 Social Cohesion X <i>DIVERSEXP</i>	-0.12	0.19	0.08	-0.05	0.03	0.21	0.21	0.19	1						
10 Social Cohesion X <i>RELSIZE</i>	-0.11	0.24	0.07	0.08	0.17	0.05	0.07	0.14	0.22	1					
11 Social Cohesion X <i>TENURE</i>	-0.09	0.15	0.09	0.06	0.05	0.22	0.04	0.11	0.17	0.23	1				
12 Technical Equivalence X <i>EXP</i>	-0.19	-0.09	0.18	0.23	0.10	0.05	-0.06	0.17	-0.04	0.04	0.06	1			
13 Technical Equivalence X <i>RELSIZE</i>	-0.12	-0.07	0.21	-0.06	0.21	0.06	-0.06	0.07	0.06	0.18	0.03	0.25	1		
14 Technical Equivalence X <i>TENURE</i>	-0.11	0.09	0.20	0.10	-0.03	0.20	0.01	0.03	0.05	-0.03	0.19	0.21	0.19	1	
15 Technical Equivalence X <i>DIVERSEXP</i>	-0.09	0.12	0.25	0.07	-0.04	-0.03	0.14	-0.09	0.14	-0.07	-0.03	0.19	0.23	0.20	1

Likelihood Estimation:

From Section 4 in the paper, we have:

$$p(y_i = 1) = F(\beta_i X_i + \gamma Z_i + \tau_i D_i), \text{ and}$$

$$\begin{bmatrix} \beta_i \\ \tau_i \end{bmatrix} = MVN(\boldsymbol{\rho}, \Sigma_{\beta\tau}),$$

where $\boldsymbol{\rho}$ is a vector that corresponds to the mean of $\begin{bmatrix} \beta_i \\ \tau_i \end{bmatrix}$, and $\Sigma_{\beta\tau}$ is the corresponding covariance matrix. To account for the Reflection problem, the above specified model is modified as:

$$p(y_i = 1) = F(\beta_i X_i + \gamma Z_i + \tau_i D_i + \alpha_C).$$

Here, C is the component to which project p belongs, and α_C is the unobservable effect common to all projects in component C . Further, α_C is allowed to be correlated with D , X and Z . Then following Chamberlain (1980), the component specific unobserved effect can be eliminated by conditioning on the event $\{y_i + y_k = 1\}$, where i and k belong to the same component. This leads to the following probabilities:

$$\begin{aligned} p(y_i = 1 | y_i + y_k = 1) &= 1 - F((\beta_i X_i + \gamma Z_i + \tau_i D_i) - (\beta_k X_k + \gamma Z_k + \tau_k D_k)), \\ p(y_j = 1 | y_i + y_k = 1) &= F((\beta_i X_i + \gamma Z_i + \tau_i D_i) - (\beta_k X_k + \gamma Z_k + \tau_k D_k)). \end{aligned}$$

For brevity, let us represent $F((\beta_i X_i + \gamma Z_i + \tau_i D_i) - (\beta_k X_k + \gamma Z_k + \tau_k D_k)) = F(\Delta G_{ik} \varphi_{ik})$. Then the conditional likelihood of the above model is equivalent to a logit estimator of the dependent variable $1(\Delta y_{ik} = 1)$ on the independent variables $\{X_i, Z_i, D_i, X_k, Z_k, D_k\}$ for the subsample of observations satisfying $\{y_i + y_k = 1\}$. The conditional log-likelihood estimator is given as:

$$\ln L(\Delta y_{ik}) = \sum_{i,k \in \{i,k: y_i + y_k = 1\}} [w_{ik} \ln F(\Delta G_{ik} \varphi_{ik}) + (1 - w_{ik}) \ln(1 - F(\Delta G_{ik} \varphi_{ik}))],$$

where $w_{ik} = 1$ if $y_i = 1$ and 0 otherwise.

Estimation Procedure.

Step 1: The Selection model is estimated as a Probit regression. After the estimation inverse Mills ratio is calculated for each project that enters Step 2 estimations.

Step 2: Step 2 involves a hierarchical Bayes estimation procedure. We have two types of parameters: (1) vary across licensors $\{\tau_i, \beta_i\}$, (2) constant across licensors $\{\gamma\}$. For brevity, let the coefficient for inverse mills ratio be absorbed in

γ and the inverse mills ratio be absorbed in ΔG_{ik} . Let $\boldsymbol{\theta}_i = \begin{bmatrix} \beta_i \\ \tau_i \end{bmatrix} \sim MVN(\boldsymbol{\rho}, \boldsymbol{\Sigma}_\theta)$, represents the set of parameters that vary across licensors (random coefficients).

The model is estimated using a standard MCMC hierarchical Bayes estimation procedure, using a Gibbs Sampler and the Metropolis Hastings algorithm coded in Matlab (Rossi et al. 2005). In the hierarchical Bayes procedure, the first 100,000 observations were used as burn-in and the last 25,000 were used to calculate the conditional posterior distributions. The MCMC works as follows. MCMC recursively generates draw from the conditional distribution of the model's parameters:

$$\begin{aligned} & \{\boldsymbol{\theta}_i\} | \Delta y_{ik}, \Delta G_{ik}, \boldsymbol{\rho}, \gamma, \boldsymbol{\Sigma}_\theta \\ & \boldsymbol{\rho} | \{\boldsymbol{\theta}_i\}, \boldsymbol{\Sigma}_\theta \\ & \boldsymbol{\Sigma}_\theta | \{\boldsymbol{\theta}_i\}, \boldsymbol{\rho}, \\ & \gamma | \{\boldsymbol{\theta}_i\}, \Delta y_{ik}, \Delta G_{ik}. \end{aligned}$$

Step 1.

Generate $\{\boldsymbol{\theta}_i\}$

$$f(\{\boldsymbol{\theta}_i\} | \Delta y_{ik}, \Delta G_{ik}, \boldsymbol{\rho}, \gamma, \boldsymbol{\Sigma}_\theta) \propto N(\{\boldsymbol{\theta}_i\} | \boldsymbol{\rho}, \gamma, \boldsymbol{\Sigma}_\theta) L(\Delta y_{ik}) \propto |\boldsymbol{\Sigma}_\theta|^{-1/2} \exp\left(-1/2 (\boldsymbol{\theta}_i - \boldsymbol{\rho})' \boldsymbol{\Sigma}_\theta^{-1} (\boldsymbol{\theta}_i - \boldsymbol{\rho})\right) L(\Delta y_{ik})$$

Because this equation does not have a close form, Metropolis Hastings algorithm is used to draw from the conditional distribution of $\boldsymbol{\theta}_i$. To reduce the autocorrelation between draws of the Metropolis-Hastings algorithm and to improve the mixing of the MCMC we used an adaptive Metropolis adjusted Langevin algorithm (Atchade 2006). The acceptance rate for MCMC draws was 19%.

Step 2.

Generate $\boldsymbol{\rho}$

$$vec(\boldsymbol{\rho}') | \{\boldsymbol{\theta}_i\}, \boldsymbol{\Sigma}_\theta = MVN(u_n, V_n)$$

where, $u_n = V_n \left((S' \otimes \boldsymbol{\Sigma}_\theta^{-1}) vec(\boldsymbol{\Theta}') + V_0^{-1} u_0 \right)$,

$$V_n = \left((S'S \otimes \boldsymbol{\Sigma}_\theta^{-1}) + V_0^{-1} \right)^{-1}$$

u_0 and V_0 are prior hyper-parameters. We use diffuse prior for both these hyper-parameters. u_0 is set to $n\theta X1$ vector of zeros and $V_0 = 100\mathbf{I}_{n\theta}$

$S = (S'_1 \dots S'_N)$ is an $N \times 1$ matrix of 1.

$\Theta = (\theta'_1 \dots \theta'_N)$ is an $N \times n\theta$ matrix that stacks $\{\theta_i\}$. $n\theta$ is the dimension of θ_i matrix of covariates.

Step 3.

Generate Σ_θ

$$\Sigma_\theta | \{\theta_i\}, S, \theta \propto IW_{n\theta}(G_0^{-1} + \sum_{i=1}^N (\theta_i - \rho)'(\theta_i - \rho), f_0 + N)$$

where IW is inverse Wishart Distribution, f_0 and G_0 are prior hyper-parameters. We use diffuse prior for both these hyper-parameters. $f_0 = n\theta + 5$, and $G_0 = \mathbf{I}_{n\theta}$

Step 4.

Generate τ

$$\gamma | \{\theta_i\}, \Delta y_{ik}, \Delta G_{ik}.$$

$$\propto N(\gamma_0, V_{\gamma_0})L(\Delta y)$$

$$\propto |V_{\gamma_0}|^{-1/2} \exp\left(-1/2 (\gamma - \gamma_0)' V_{\gamma_0}^{-1} (\gamma - \gamma_0)\right) L(\Delta y)$$

γ_0 and V_{γ_0} are prior hyper-parameters. We use diffuse prior for both these hyper-parameters. $\gamma_0 = n\gamma X1$ vector of zeros ($n\gamma$ is the dimension of γ) and $V_{\gamma_0} = 50\mathbf{I}_{n\gamma}$ Metropolis Hastings algorithm is used to draw from the conditional distribution of γ . An acceptance probability of 23% is achieved.

Convergence Check: We follow the method suggested by (Gelman and Rubin 1992) to check whether convergence has been achieved. The within to between variance for each parameter estimated across multiple chains was compared. Across five parallel chains, the scale reduction estimate for all parameters estimated was lower than 1.1 which indicated that the convergence was achieved.

Robustness Checks

We performed several robustness checks. We reran the analysis by including all developers associated with

a project, regardless of whether their active involvement could be established. We also used an alternative measure of relative project success using a project's SourceForge percentile rank. The results from these analyses were consistent with the results in Tables 3 and 4. Next, we used two alternative constructions of our dependent variable. First, we considered each of the 47 different OSS license to be at risk of adoption and modeled the likelihood a new project adopted a particular license using multinomial logit regression. We treated new projects that adopted multiple licenses as multiple, individual projects to model the choice of each specific license. Accordingly, we allowed for correlation in the errors associated with projects that adopt multiple licenses. We recomputed all explanatory variables to reflect a licensor's exposure to individual licenses rather than to categories of licenses. For 35 licenses, the signs and significance of coefficients for explanatory variables were consistent with those reported in Table 3. For the remaining licenses, coefficients were in expected directions but largely insignificant, most likely because of the much smaller number of projects that adopted these licenses. Second, we treated the DV as three mutually exclusive and exhaustive *types* of licenses and modeled the likelihood a new project adopted a particular license type using multinomial logit regression. We used the most restrictive license a multi-license project adopted because our explanatory variables reflect a licensor's experience with and exposure to licenses based on their degree of restrictiveness. Using the least restrictive license type a new project chose, which may be less restrictive than those adopted by other socially proximate projects, would restrict the range of adoption behavior we could observe and potentially bias the results. Results for the explanatory variables in this analysis were consistent with those in Tables 3-4 in terms of sign and statistical significance.

Finally, we checked if our results were biased by unobserved heterogeneity in social influence. Some licensors may have worked on their new projects with developers prior to registering the projects on SF, resulting in unobserved social influence on these licensors' choices and potentially biased estimates for our social influence variables. In addition to using random parameters to accommodate such unobserved heterogeneity, we reconstructed the inter-project networks by assuming that developers who joined within one month *after* a project's registration on SF had worked on the project prior to its registration. Results for the explanatory variables were consistent with those reported in Tables 3-4.

Table A6. Heckman Probit Regression Results for **All Licenses Highly Restrictive**. (Selection equation sample includes all projects that started at Sourceforge between Jan 1, 2002 and December 31, 2003. License choice equation sample includes only those projects that have at least one social tie of distance 2). License Choice model is Hierarchical Bayes Logit License Choice model which includes inverse Mills ratio as one of the covariates. The standardized coefficients are reported.

Variables		Selection Equation	License Choice Equation	
		Posterior Coefficient Mean	Posterior Coefficient Mean	Posterior Coefficient Variance.
Selection Instrument	Want to Learn	-0.171***		
Selection Instrument	Network Density	2.457***		
Social Proximity	Social Cohesion		1.394***	(0.32)
	Technical Equivalence		0.966***	(0.34)
Licensor Attractiveness	Licensor's Experience of OSS projects (EXP)		-0.009*	(0.01)
	Relative Size of Licensor's Prior Projects (RELSIZE)		-0.016**	(0.01)
	Licensor's tenure at SF (TENURE)		-0.512***	(0.35)
	Diversity of Licensor's experience at SF (DIVERSEXP)		-0.012	(0.12)
Susceptibility	Social Cohesion X EXP		-0.192***	(0.22)
	Social Cohesion X RELSIZE		-0.160***	(0.11)
	Social Cohesion X TENURE		-0.291*	(0.04)
	Social Cohesion X DIVERSEXP		-0.077*	(0.03)
	Technical Equivalence X EXP		-0.034*	(0.01)
	Technical Equivalence X RELSIZE		-0.145**	(0.03)
	Technical Equivalence X TENURE		-0.044*	(0.13)
	Technical Equivalence X DIVERSEXP		-0.119***	(0.07)
	Structural Equivalence		0.177**	(0.20)
Software Attractiveness	Intended Audience	End users	-0.107	0.041*
		Developers	0.117***	-0.165***
		System Administrators	0.020	0.103***
	Operating Systems	POSIX	-0.123***	0.027
		BSD Platforms	0.030	-0.031
		SGI IRIX	-0.079	-0.073
		Microsoft	0.022	-0.031
		OS/2	-0.051	0.061
		PalmOS	0.128	0.084**
		MacOS	0.169***	-0.067*
		Embedded	-0.122	0.002
	OS independent	0.162***	-0.021	
	Translations	English	0.034	0.031
		Chinese	-0.069	-0.011
		Spanish	0.234***	0.032**
German		-0.041	0.042***	
Russian		0.044	0.047	
Japanese		0.160	-0.041	

	French	-0.028	0.142***
	Swedish	0.040	0.001
	Portuguese	0.048	-0.015
	Polish	-0.150	-0.049
User interface	Curses/Ncurses	0.065	0.037
	MS Windows	-0.111***	-0.055
	X11	0.159***	0.025*
	KDE	0.184**	0.014
	Gnome	-0.342***	0.008
	Web Based	0.062***	0.008
	Daemon	0.148***	-0.036
	Cocoa(MacOS)	0.242***	0.035*
	Handheld	0.125	-0.001
Topic	Communications	0.067**	0.003
	Database	0.060	0.054***
	Desktop Environment	0.225**	-0.031
	Education	-0.013	0.044
	Games	0.084	0.056***
	Internet	0.011	-0.017
	Multimedia	0.006	0.031**
	Financial	0.069	-0.001
	Printing	0.207	0.019
	Scientific	-0.003	0.004
	Security	0.062	0.006
	Software Development	0.190***	-0.010
	Text Editors	0.030	-0.001
	Office Business	0.027	0.003
	Terminals	-0.048	-0.041*
	Inverse Mills Ratio		0.127*
	Number of Projects	21220	5307

All models include Time dummy control variables. Social Cohesion accounts for infectiousness based on size. *** implies that the 99% confidence interval does not include zero. ** implies that the 95% confidence interval does not include zero; * implies that the 90% confidence interval does not include zero.

Table A7. Heckman Probit Regression Results for **Some Licenses Highly Restrictive**. (Selection equation sample includes all projects that started at Sourceforge between Jan 1, 2002 and December 31, 2003. License choice equation sample includes only those projects that have at least one social tie of distance 2). License Choice model is Hierarchical Bayes Logit License Choice model which includes inverse Mills ratio as one of the covariates. The standardized coefficients are reported.

		Selection Equation	License Choice Equation	
Variables		Posterior Coefficient Mean	Posterior Coefficient Mean	Posterior Coefficient Variance
Selection Instrument	Want to Learn	-0.171***		
Selection Instrument	Network Density	2.457***		
Social Proximity	Social Cohesion		1.739***	(0.35)
	Technical Equivalence		1.122***	(0.21)
Licensor Attractiveness	Licensor's Experience of OSS projects (EXP)		-0.014*	(0.02)
	Relative Size of Licensor's Prior Projects (RELSIZE)		-0.039***	(0.01)
	Licensor's tenure at SF (TENURE)		-0.492***	(0.23)
	Diversity of Licensor's experience at SF (DIVERSEXP)		-0.074	(0.16)
Susceptibility	Social Cohesion X EXP		-0.117***	(0.21)
	Social Cohesion X RELSIZE		-0.134*	(0.05)
	Social Cohesion X TENURE		-0.216*	(0.03)
	Social Cohesion X DIVERSEXP		-0.065*	(0.03)
	Technical Equivalence X EXP		-0.038*	(0.02)
	Technical Equivalence X RELSIZE		-0.097**	(0.01)
	Technical Equivalence X TENURE		-0.079*	(0.01)
	Technical Equivalence X DIVERSEXP		-0.104***	(0.05)
	Structural Equivalence		0.284**	(0.29)
Software Attractiveness	Intended Audience	Endusers	-0.107	0.057***
		Developers	0.117***	-0.158***
		System Administrators	0.020	0.088***
	Operating Systems	POSIX	-0.123***	0.074*
		BSD Platforms	0.031	-0.015
		SGI IRIX	-0.083	-0.061
		Microsoft	0.022	-0.009
		OS/2	-0.053	0.051
		PalmOS	0.126	0.118**
		MacOS	0.169***	-0.065***
		Embedded	-0.123	0.044
	OS independent	0.163***	-0.074**	
	Translations	English	0.034	0.014
		Chinese	-0.069	-0.021
Spanish		0.233***	0.073***	
German		-0.041	0.057**	
Russian		0.044	0.049*	

	Japanese	0.159	-0.047
	French	-0.029	0.022
	Swedish	0.040	-0.013
	Portuguese	0.049	0.001
	Polish	-0.150	-0.011
User interface	Curses/Ncurses	0.065	0.017
	MS Windows	-0.111***	-0.001
	X11	0.159***	0.036**
	KDE	0.185**	0.017
	Gnome	-0.341***	0.001
	Web Based	0.062***	0.024**
	Daemon	0.148***	-0.006
	Cocoa(MacOS)	0.242***	0.006
	Handheld	0.127	-0.001
Topic	Communications	0.067**	0.009
	Database	0.060	0.042***
	Desktop Environment	0.226**	-0.036
	Education	-0.013	0.009
	Games	0.084	0.049***
	Internet	0.011	-0.022
	Multimedia	0.005	0.039***
	Financial	0.069	0.000
	Printing	0.208	0.012
	Scientific	-0.003	0.076
	Security	0.062	0.013
	Software Development	0.191***	-0.032***
	Text Editors	0.030	-0.007
	Office Business	0.026	0.005
	Terminals	-0.048	-0.038
	Inverse Mills Ratio		0.141**
	Number of Projects	21220	5307

All models include Time dummy control variables. Social Cohesion accounts for infectiousness based on size. *** implies that the 99% confidence interval does not include zero. ** implies that the 95% confidence interval does not include zero; * implies that the 90% confidence interval does not include zero.

Table A8. Heckman Regression Results for **All Licenses Restrictive**. Selection equation is a Probit Regression. License Choice model is Hierarchical Bayes Logit Regression which includes inverse Mills ratio as one of the covariates. The standardized coefficients are reported. (Selection equation sample includes all projects that started at Sourceforge between Jan 1, 2002 and December 31, 2003. License choice equation sample includes only those projects that have at least one social tie of distance 2).

		Selection Equation	License Choice Equation	
Variables		Posterior Coefficient Mean	Posterior Coefficient Mean	Posterior Coefficient Variance
Selection Instrument	Want to Learn	-0.171***		
Selection Instrument	Network Density	2.457***		
Social Proximity	Social Cohesion		2.144***	(0.42)
	Technical Equivalence		1.446***	(0.15)
Licensor Attractiveness	Licensor's Experience of OSS projects (EXP)		-0.055**	(0.01)
	Relative Size of Licensor's Prior Projects (RELSIZE)		-0.027***	(0.01)
	Licensor's tenure at SF (TENURE)		-0.447***	(0.14)
	Diversity of Licensor's experience at SF (DIVERSEXP)		-0.069	(0.15)
Susceptibility	Social Cohesion X EXP		-0.103***	(0.32)
	Social Cohesion X RELSIZE		-0.138***	(0.10)
	Social Cohesion X TENURE		-0.383**	(0.05)
	Social Cohesion X DIVERSEXP		-0.123**	(0.10)
	Technical Equivalence X EXP		-0.026	(0.02)
	Technical Equivalence X RELSIZE		-0.101**	(0.02)
	Technical Equivalence X TENURE		-0.117***	(0.02)
	Technical Equivalence X DIVERSEXP		-0.181***	(0.06)
	Structural Equivalence		0.177**	(0.20)
Software Attractiveness	Intended Audience	Endusers	-0.107	0.080
		Developers	0.117***	-0.065***
		System Administrators	0.020	0.026
	Operating Systems	POSIX	-0.123***	0.086***
		BSD Platforms	0.031	-0.119**
		SGI IRIX	-0.083	-0.039
		Microsoft	0.022	-0.011
		OS/2	-0.053	0.091*
		PalmOS	0.126	0.059
		MacOS	0.169***	-0.027
		Embedded	-0.123	0.025
	OS independent	0.163***	0.015	
	Translations	English	0.034	0.061
		Chinese	-0.069	-0.014
		Spanish	0.233***	0.064*
German		-0.041	0.027***	
Russian		0.044	0.019	

	Japanese	0.159	-0.074
	French	-0.029	0.080***
	Swedish	0.040	0.043
	Portuguese	0.049	-0.039
	Polish	-0.150	0.023
User interface	Curses/Ncurses	0.065	0.015
	MS Windows	-0.111***	0.002
	X11	0.159***	0.023**
	KDE	0.185**	0.047***
	Gnome	-0.341***	0.018
	Web Based	0.062***	0.007
	Daemon	0.148***	-0.018
	Cocoa(MacOS)	0.242***	0.027
	Handheld	0.127	-0.018
	Topic	Communications	0.067**
Database		0.060	0.038***
Desktop Environment		0.226**	0.014
Education		-0.013	0.034
Games		0.084	0.129***
Internet		0.011	-0.037
Multimedia		0.005	0.097**
Financial		0.069	0.101**
Printing		0.208	-0.024
Scientific		-0.003	-0.018
Security		0.062	0.004
Software Development		0.191***	-0.030
Text Editors		0.030	0.023
Office Business		0.026	0.093*
Terminals		-0.048	-0.009
	Inverse Mills Ratio		0.096**
	Number of Projects	21220	5307

All models include Time dummy control variables. Social Cohesion accounts for infectiousness based on size. *** implies that the 99% confidence interval does not include zero. ** implies that the 95% confidence interval does not include zero; * implies that the 90% confidence interval does not include zero.

Table A9. Heckman Regression Results for **Some Licenses Restrictive**. Selection equation is a Probit Regression. License Choice model is Hierarchical Bayes Logit Regression which includes inverse Mills ratio as one of the covariates. The standardized coefficients are reported. (Selection equation sample includes all projects that started at Sourceforge between Jan 1, 2002 and December 31, 2003. License choice equation sample includes only those projects that have at least one social tie of distance 2).

		Selection Equation	License Choice Equation	
Variables		Posterior Coefficient Mean	Posterior Coefficient Mean	Posterior Coefficient Variance
Selection Instrument	Want to Learn	-0.171***		
Selection Instrument	Network Density	2.457***		
Social Proximity	Social Cohesion		1.627***	(0.32)
	Technical Equivalence		1.163***	(0.23)
Licensor Attractiveness	Licensor's Experience of OSS projects (EXP)		-0.024**	(0.01)
	Relative Size of Licensor's Prior Projects (RELSIZE)		-0.051***	(0.02)
	Licensor's tenure at SF (TENURE)		-0.455***	(0.06)
	Diversity of Licensor's experience at SF (DIVERSEXP)		-0.023	(0.16)
Susceptibility	Social Cohesion X EXP		-0.105***	(0.22)
	Social Cohesion X RELSIZE		-0.097*	(0.10)
	Social Cohesion X TENURE		-0.192*	(0.09)
	Social Cohesion X DIVERSEXP		-0.111**	(0.06)
	Technical Equivalence X EXP		-0.029*	(0.01)
	Technical Equivalence X RELSIZE		-0.077*	(0.03)
	Technical Equivalence X TENURE		-0.094**	(0.00)
	Technical Equivalence X DIVERSEXP		-0.204***	(0.12)
	Structural Equivalence		0.177**	(0.20)
Software Attractiveness	Intended Audience	Endusers	-0.107	0.018
		Developers	0.117***	-0.015***
		System Administrators	0.020	0.009
	Operating Systems	POSIX	-0.123***	0.021***
		BSD Platforms	0.031	-0.089***
		SGI IRIX	-0.083	-0.006
		Microsoft	0.022	-0.001
		OS/2	-0.053	0.095***
		PalmOS	0.126	0.018*
		MacOS	0.169***	-0.013*
		Embedded	-0.123	0.073***
	OS independent	0.163***	0.015*	
	Translations	English	0.034	0.001
		Chinese	-0.069	-0.002
Spanish		0.233***	0.029***	
German		-0.041	0.014**	
Russian		0.044	0.032***	

	Japanese	0.159	-0.013
	French	-0.029	0.041***
	Swedish	0.040	0.048*
	Portuguese	0.049	-0.012
	Polish	-0.150	-0.011
User interface	Curses/Ncurses	0.065	0.028*
	MS Windows	-0.111***	0.006
	X11	0.159***	0.016
	KDE	0.185**	0.027***
	Gnome	-0.341***	0.004
	Web Based	0.062***	0.014***
	Daemon	0.148***	-0.001
	Cocoa(MacOS)	0.242***	0.007
	Handheld	0.127	-0.004
	Topic	Communications	0.067**
Database		0.060	0.051***
Desktop Environment		0.226**	0.125**
Education		-0.013	0.008
Games		0.084	0.131***
Internet		0.011	0.001
Multimedia		0.005	0.005
Financial		0.069	0.001
Printing		0.208	-0.001
Scientific		-0.003	0.023**
Security		0.062	0.005
Software Development		0.191***	-0.006*
Text Editors		0.030	0.011**
Office Business		0.026	0.004
Terminals		-0.048	-0.002
	Inverse Mills Ratio		0.091**
	Number of Projects	21220	5307

All models include Time dummy control variables. Social Cohesion measures account for infectiousness based on size. *** implies that the 99% confidence interval does not include zero. ** implies that the 95% confidence interval does not include zero; * implies that the 90% confidence interval does not include zero.

References

- Atchade, Y. 2006. An adaptive version for the metropolis adjusted langevin algorithm with a truncated drift. *Methodology and Computing in Applied Probability*. **8**(2) 235-254.
- Bramoulle, Y., Djebbari, H., Fortin., B. 2009. Identification of peer effect social networks. *Journal of Econometrics*. **150** 41-55.
- Chamberlain, G. 1980. Analysis of covariance with qualitative data. *The Review of Economic Studies*. **47**(1) 225-238.

- Gelman, A., Rubin, D. 1992. Inference from iterative simulation using multiple sequences. *Statistical Science*. **7**(4) 457-472.
- Rossi, P., Allenby, G., McCulloch, R. 2005. *Bayesian statistics and marketing*. John Wiley & Sons.
- Wasserman, S., Faust, K. 1994. *Social network analysis*. Cambridge University Press, London.