

Testing for Network Effects in Field Experiments: Examples from Legislative Studies

Sayali Phadke

Bruce Desmarais

May 15, 2016

Abstract

This is the pre-JSM draft; the draft before edits were made for JSM

Most social processes involve complex interaction and dependence among units in a network. The stable unit treatment value assumption (SUTVA)—the assumption that a unit’s outcome is unaffected by other units’ treatment statuses—is required in conventional approaches to causal inference. When SUTVA is violated, as in networked social interaction, treatment effects spread to control units through the network structure. We evaluate the evidence for spillover effects in data from three field experiments on US state legislatures. Randomized field experiments represent the gold standard in causal inference when studying political elites. It is not possible to bring political elites into the lab, and causal identification with observational data is fraught with problems. We propose new specifications for treatment spillover models, and construct networks through geographical or ideological proximity and co-sponsorship. Considering different combinations of spillover models and networks, we evaluate the robustness of recently developed non-parametric tests for interference. The approaches we illustrate can be applied to any experimental setting in which interference is suspected.

1 Introduction

Networks are integral parts of human interaction and hence social science research. If one unit in a network gets treated, the effect may trickle down throughout network. The currently established framework for causal inference relies on SUTVA (Stable Unit Treatment Value Assumption). It assumes that whether or not one person/unit/node is

treated, does not affect any other unit. However, SUTVA breaks down in a network setting. It is therefore imperative to take the interference structure into account. Rather, in policy planning or designing marketing campaigns, a researcher may be interested in studying the propagation of treatment effect itself.

In field experiments on social groups, interference may be substantial. In this project we intend to study interference models for randomized experiments conducted on social networks and causal inference basis this.

To understand, explain and predict social phenomena, social scientists typically look to individual actors' attributes to explain their behavior (e.g., an increase in an individual's wealth will result in a decrease in that individual's support for government spending on social welfare), or to attributes of the macro context (e.g., an increase in the unemployment rate will lead an individual's support for the party of the president to decrease). However, these two conceptual approaches to explaining individual behavior leave out a potentially powerful class of social dynamics – interpersonal influence. That is, the behavior of one individual may depend upon the behavior of one or more others (e.g., a person may decide to vote due to their friends claiming to have voted (?)). Inferences regarding influence involve the analysis of individual behaviors and the behaviors of those adjacent or nearby in some contact network. However, as in most settings, it is generally not possible to identify the causal effects that map onto the process of social influence in observational data (?). As such, we need experimental methods to identify causal influence effects.

1.1 tasks

- Points about why it is interesting to study propagation. (BD)

- Outline of the paper (SP)

2 Background

- Paragraph on each category of papers that serve as relevant background (SP)
- Interference models (diffusion, propagation) (SP–Review)
- Experiments on networks (applications) (SP–Review)
- Approaches to inference or estimation with propagation (SP–Review)
- Potential outcomes framework (SP – find papers & Review)
- Review of political networks (SP–Review)
- Review of field experiments (SP–Review)
 - (??????)

3 Research Design

We plan to re-analyze data from past field experimental studies to understand how conclusions regarding direct effects and interference effects depend upon the network structure.

Table 1: The Results

	Distance from treated < 5		Distance from treated > 5	
	Linear	Non-linear	Linear	Non-linear
Number of treated neighbors	1	2	5	6
Proportion of treated neighbors	3	4	7	8

3.1 tasks

The key factors that we will consider in building propagation models are:

1. Distance from the nearest treated node (d_i)
2. Number/proportion of treated nodes neighboring each d_i
3. Form of spread (linear or non-linear)

** Think of what should go into this table? Can we put empirical models that fit into each of the cells or examples of networks for which such a spreading model will be suitable?

Tabulates the basic idea of the models we will look at. The linear/non-linear refers to the form taken by the spread of effect. Distance from the nearest treated node is vaguely separated as <5 or >5 for now. Finally, we want to look at the difference in spillovers effects caused due to number of treated neighbors as against proportion of treated neighbors. Potentially, the number/proportion at hops of different lengths can also be considered.

The types of networks we will consider in the analysis are:

1. Geographical proximity
2. Ideological similarity
3. Co-sponsorship

Each of these factors is such, that a treatment such as the message sent through emails in New Hampshire, would possibly spread to untreated units as well. Legislators from adjoining districts may affect each other's opinions through geographic proximity as well as potentially via common issues faced by citizens in their constituencies.

Ideological similarity is a tricky variable because it could be hard to distinguish that from party affiliation. In the New Hampshire paper, there is a need to include this effect in the model since matched pairs are created based on party affiliation. If a Republican candidate receives the treatment, the chances that through various communication channels, he/she will convey the message to the control group candidate from the same party and district, are very high.

Finally, serving on the same committee can also contribute to spreading the effect of a treatment. We must test for any dependence across these three factors before incorporating them into our model. Therefore it is important that we propose and test propagation models that consider the spread of treatment through our network.

****Notes:**

1. Ideological similarity: This should get highest priority in modeling, since I believe, it would be easiest to affect an undecided legislator's vote through similarity in ideas and belief about citizen's issues and how to resolve them. I would propose that we model immediate neighbours to have a 50

2. Geographical proximity: An untreated legislator from adjoining district would be
3. Co-sponsorship: Serving on the same committee increases the chances of

4 Analysis

We begin this section by reviewing prior methodological work. We will look at two spillover specifications and testing frameworks given in Bowers et. al. and Coppock papers. We will extend their analysis by considering various other specifications for spillover effects. Additionally, we will look into tests other than the Kolmogorov-Smirnov (KS) test.

4.1 Review of existing methods

- **Bowers et. al. method:** This paper introduces a Fisherian inference algorithm to test for spillover of treatment effect. The model \mathcal{H} is compared against the observed data. The hypothesized model of interference is specified by the researcher. The steps involved in conducting this test are as follows:

1. Assume the "sharp null hypothesis of no effects" i.e. the treatment assignment has no effect on any unit
2. Specify the causal model which describes the change in potential outcomes when treatment assignment changes from \mathbf{u} to \mathbf{w} ; $\mathcal{H}(y_{i,\mathbf{u}}, \mathbf{w}, \theta)$
3. Potential outcomes from the causal model must be mapped to observed outcomes y_z . Treatment assignment in the experiment (\mathbf{z}) must be mapped to the

uniformity trial ($\mathcal{H}(y_{\mathbf{z}}, \mathbf{0}, \theta) = \mathbf{y}_0$) which is based on the baseline condition of no-treatment assignment i.e. every unit is a control unit

4. Test statistic \mathcal{T} should be a small value when distribution of treated and control outcomes in the adjusted data are similar, and a large value when distributions are dissimilar. We need a sensitive measure to account for similarity on not just the center but also higher-order moments of a distribution. Bowers et. al. recommend the Kolmogorov-Smirnov (KS) test statistic
5. Assume that treatment only spreads through edges and the spillover effect only depends on the number of neighbours treated. Spillover effect modeled using a growth curve $\beta + (1 - \beta)e^{-\tau^2 \mathbf{z}^T \mathbf{s}}$
6. Generate the distribution of test statistic under our hypothesis. The exact distribution is specified by computing $t_k = \mathcal{T}(\mathbf{y}_0, \mathbf{Z}_k)$ for each $\mathbf{Z}_k \in \Omega$. Alternatively, we can use sampling methods and limit theorems to estimate the distribution from data.
7. p-value calculated using $\frac{\sum_{k=1}^{|\Omega|} I(x_i > t_k)}{|\Omega|}$

- **Coppock replication:** Coppock builds upon the New Mexico Legislator experiment conducted by Butler and Nickerson (2011). This paper also works with the idea of sharp null of no effects and uniformity trial. Similarity in ideological scores is used as the adjacency matrix. Procedure is as follows:

- W-NOMINATE ideology score calculated for each legislator using rollcall vote data
- Ideological similarity calculated as $Similarity_{i,j} = \frac{2 - |ideo_i - ideo_j|}{2}$

- Raw exposure calculated as $Rawexposure_i = \sum_{j=1}^n Similarity_{i,j} * z_j, j \neq i$. However, this specification introduces a correlation between exposure - direct treatment, exposure - ideology and exposure - other unobservable characteristics. Therefore we further process the raw exposures.
- To calculate expected exposures, we simulate exposures under a large number of randomizations. Each randomization where legislator i is in treatment is indexed as k ($k = 1, 2, \dots, K$) and where legislator i is in control is indexed as l ($l = 1, 2, \dots, L$)

$$Expectedexposure_{i,z_i=1} = \frac{\sum_{k=1}^K \sum_{j=1}^n Similarity_{i,j} * z_{j,k}}{K}, j \neq i, z_{i,k} = 0$$

$$Expectedexposure_{i,z_i=0} = \frac{\sum_{l=1}^L \sum_{j=1}^n Similarity_{i,j} * z_{j,l}}{L}, j \neq i, z_{i,l} = 1$$

- Using ideological similarity scores in the adjacency matrix, Coppock separates out the direct treatment effect and spillover occurring through the network.

4.2 Extensions of Coppock analysis

We consider several extensions of the Coppock method, all applied to the Butler and Nickerson data. We would like to apply the same methods to the Bergan data as well. (The ones in red are planned extensions, in the order of priority)

- Consider k ($=3, 5, 8, 12$) neighbors according to ideological scores to form adjacency matrix. A tie between i and j indicates that legislator j is one of the k -nearest neighbors to i . This generated some ties. We will try the following two ways of

taking care of the ties:

1. Choose, among the ties in the k closest neighbors for i , those nodes for which i is the closest neighbor. To illustrate, if i is equally close to j and h , we ask whether i ranks higher on j 's list of closest neighbors. If yes, then we kick out h . If no, we keep j and h if i ranks equally on j and h 's neighbor list, or kick out j if h ranks i more highly.
 2. Look at the number of nodes for which j is the nearest neighbor and in some way account for the j 's influence on that basis?
1. Look at control unit i
 2. Calculate a composite score that indicates the ranking of i for each of its k neighbors (similar to the ranking idea you have suggested)
 3. Estimate a parameter to see if change in composite score leads to a change in propensity for changing the outcome
 4. We can consider modeling this parameter as something other than linear effect

For example, say we are under $k=3$ condition. If I am a control unit and I am the lowest in the proximity of all of my 3 neighbors, I am less likely to receive a spillover; as compared to if I was in the highest proximity of all three of them (the two extremes). We might have to condition this somehow on the treatment status of my neighbors though.

- Consider committee network instead of ideological network, where a tie between i and j indicates that they have served on two or more committees together
- Use number of shared committees in adjacency matrix and see the results

- Separate out the high and low support district in original data and conduct separate analyses. This would explain the direction of the spillover effect much better. Conduct this analysis with ideological network as well as committee network
- Replicate all of these models (with suitable modifications) to the Bergan (New Hampshire) data
- Include geographical network in both dataset and extend analysis
- Consider different spillover models
- Consider weighted combinations of different networks to estimate a γ
- Explore the idea of using communities to model spread across the network

4.3 tasks

- Replicate Bergan. (SP)
- Find other network data for the New Mexico legislature. (BD)
- Geography and ideology data for New Hampshire (BD)
- Produce cosponsorship, ideology and geography estimates for both Bergan and Nickerson (SP & BD)
- For at least two spreading models (SP & BD)
- Replicate Nyhan (SP)

Records of standing committee membership in the 16 standing committees in place during the 2008 regular session was obtained from the New Mexico Legislative Council Service Librarian.

5 Summary

So far, we have replicated the two papers mentioned earlier; (?) and (?). As of now, we will go ahead with the Coppock method alone. Bowers method is suitable if the outcome (y) is continuous. For binary outcomes, the uniformity trial conversion leads to ties in the sub-vectors we will use to perform KS test. In our datasets, outcome variable is voting decision; essentially binary. The codes are available in the appendix. So far we have worked on the New Mexico legislators dataset from (?) and considered network based on ideological similarity. The plan ahead for this paper is:

- Consider different diffusion models by varying the distance from treated node, number/proportion of treated neighbors and form of spread of treatment
- Consider other legislator networks depending on geographical proximity and co-sponsorship
- Consider additional test statistic such as the Anderson-Darling test and other tests mentioned in (?) (Mann-Whitney U test, Control Median test etc.)

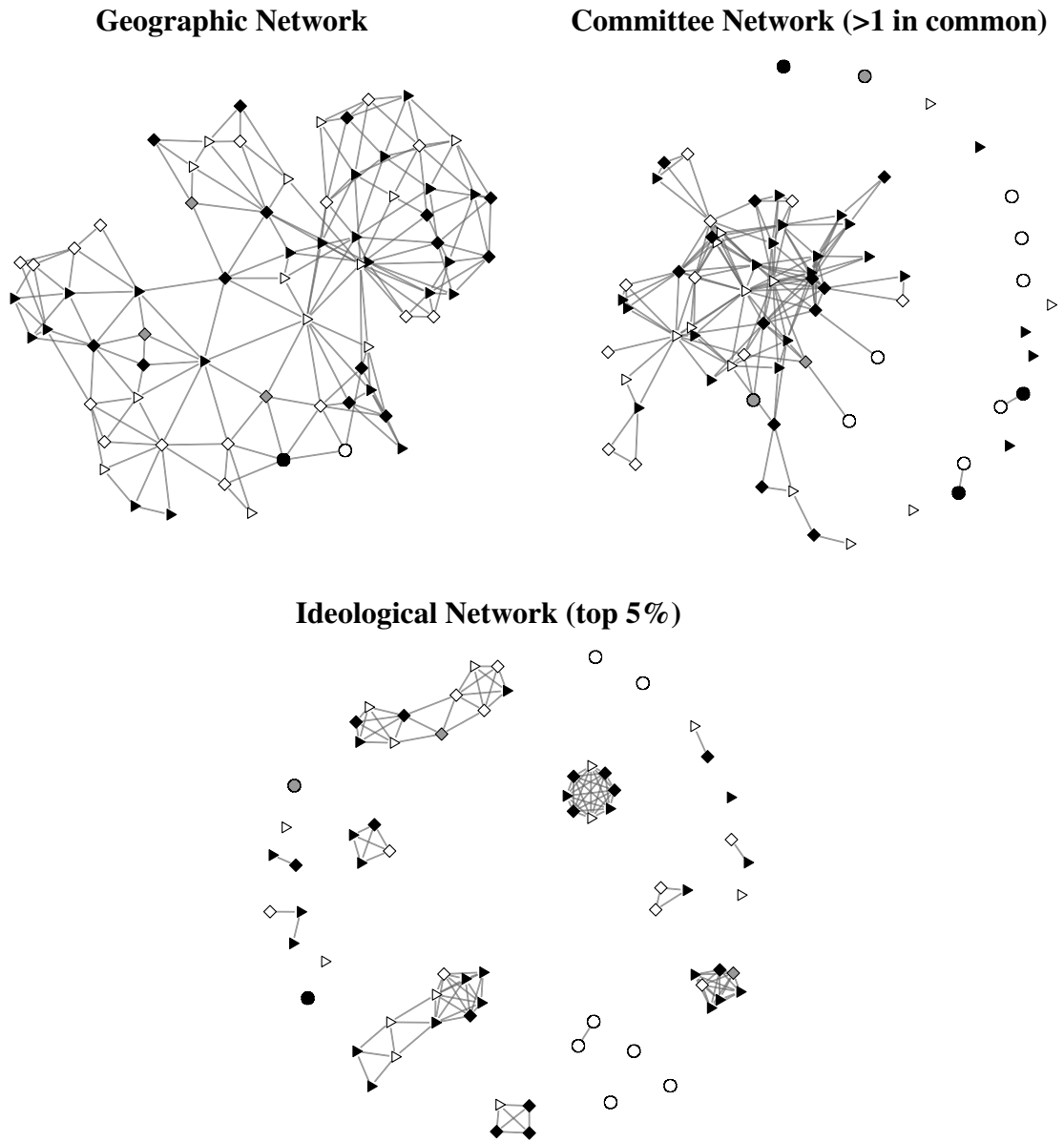


Figure 1: Different networks among New Mexico legislators. Colors denote outcome: black means voted with district, gray means abstained, white means voted against. Shape denotes treatment status. Triangles are treated. Squares are adjacent to treated. Circles are isolated from treatment

6 Appendix

6.1 Appendix 1A

In this section, we will look at user-defined R-functions that replicate the (?) methodology. This contains four steps:

- A function to transform the observed outcomes into potential outcomes for any treatment assignment w
- A function to separate the hypothesized treatment effect
- A function to calculate test statistic
- A function to calculate the p-value.

The results from the `ks.test` function in R for calculating Kolmogorov-Smirnoff test statistic are verified with that in Footnote 12 of the paper.

Function 1: calculating potential outcomes

```
1 set.seed(132)
2 library(doParallel)
3 library(foreach)
4 library(kSamples)
5 library(network)
6 library(permute)
7
8 ##### Potential outcomes #####
9
10 ##### Transform uniformity trial outcome into observed outcome
```

```

11 unif.to.z <- function(z, S, y.0, beta, tau){
12   # z: observed treatment assignment
13   # S: adjacency matrix
14   # y.0: outcome vector for uniformity trial
15   # beta: growth curve parameter
16   # tau: rate of growth parameter
17
18   scalar <- as.vector(t(z)%*%S)
19   spillover <- rep(NA, n)
20
21   spillover <- beta + ((1-z) * (1-beta) * exp(-tau^2 * scalar))
22
23   # This is equation 4
24   h.y0.z <- spillover*y.0
25 }
26
27 ##### Transform observed outcome into uniformity trial outcome
28 z.to.unif <- function(z, S, y.z, beta, tau){
29   # z: initial treatment assignment
30   # S: adjacency matrix
31   # y.z: observed outcome vector
32   # beta: growth curve parameter
33   # tau: rate of growth parameter
34
35   scalar <- as.vector(t(z)%*%S)
36   spillover <- rep(NA, n)
37
38   # Equation (3)

```

```

39   spillover <- beta + ((1-z) * (1-beta) * exp(-tau^2 * scalar))
40
41   # This is equation 5
42   h.yz.0 <- (1/spillover)*y.z
43 }
44
45 ##### Transform observed outcome into outcome for ANY other assignment w
46 z.to.w <- function(z, S, w, y.z, beta, tau){
47   # z: initial treatment assignment
48   # S: adjacency matrix
49   # w: new treatment assignment
50   # y.z: vector of outcomes for z
51   # beta: growth curve parameter
52   # tau: rate of growth parameter
53
54   scalar.z <- as.vector(t(z)%*%S)
55   scalar.w <- as.vector(t(w)%*%S)
56
57   spillover.z <- rep(NA, n)
58   spillover.z <- beta + ((1-z) * (1-beta) * exp(-tau^2 * scalar.z))
59
60
61   spillover.w <- rep(NA, n)
62   spillover.w <- beta + ((1-w) * (1-beta) * exp(-tau^2 * scalar.w))
63
64
65   # Below is the actual function that transforms observed outcomes into
    potential outcomes

```

```

66 # Equation (6)
67
68 h.z.to.w <- (spillover.w / spillover.z) * y.z
69 }
70
71 ##### Testing and p-value calculation #####
72
73 p.val <- function(z, y.z){
74
75   cl <- makeCluster(4) #Setup for parallel computing
76   registerDoParallel(cl)
77
78   # Calculate the outcome vector after taking away the effect of
       treatment
79   y.0 <- z.to.unif(z=z, S=S, y.z=y.z, beta=beta, tau=tau)
80
81   # Calculate test statistic
82   test.stat <- ks.test(y.0[z==1], y.0[z==0],
83                       alternative = "g")$statistic
84   sign <- noquote(strsplit(names(test.stat), NULL)[[1]])[3]
85   if(sign==""){
86     test.stat <- test.stat
87   } else {
88     test.stat <- test.stat*-1
89   }
90
91   # Calculate a vector of test statistic using permutations
92   results <- foreach (i = 1:perms) %dopar%{

```



```

93   require(permute)
94   perm.z <- z[sample(1:length(z),length(z),rep=F)]
95   perm.test.stat <- ks.test(y.0[perm.z==1], y.0[perm.z==0],
96                           alternative = "g")$statistic
97   sign <- noquote(strsplit(names(perm.test.stat), NULL)[[1]])[3]
98
99   if(sign==""){
100     return(perm.test.stat)
101   } else {
102     return(perm.test.stat*-1)
103   }
104 }
105 stopCluster(cl)
106
107 # A vector of test statistics
108 all.test.stat.vals <- unlist(results)
109
110 # Calculating p-value
111 pval <- sum(all.test.stat.vals > test.stat)/perms
112 return(pval)
113 }

```

6.2 Appendix 1B

Below code replicates the (?) results using the framework setup in the Bowers replication code (Version before final corrections by BD)

1

```

2 setwd("~/Dropbox/professional/Research/Active/causalityinnetworks –
  agenda/Interference_in_Field_Experiments/Analysis/coppock_
  replication_data/") # BD
3
4 rm(list=ls())
5 set.seed(312)
6
7 library(doParallel)
8 library(fields)
9 library(foreach)
10 library(kSamples)
11 library(network)
12 library(permute)
13 library(wnominate)
14
15
16 #### Read the original Butler and Nicketson data
17 #### This is the New Mexico dataset
18
19 data <- read.table("nm.replication.tab", sep="\t", header=TRUE)
20
21 z <- data$treatment #observed treatment
22 y.z <- data$sb24 #observed outcome
23 n <- length(y.z) #number of observations
24 t <- length(z[z==1]) #number of treated units
25 perms <- 10000 #number of permutations to use in generating expected
  exposure
26 perms.test <- 1000 #number of permutations used in testing

```

```

27
28
29 ##### Generate Similarity Scores (this code taken from CoppockJEPS_
    datapreparation.R)
30
31 nmhouse2008 <- read.csv("CoppockJEPS_rolldata.csv")
32 bills <- data.frame(nmhouse2008[5:21])
33
34 ## Nominate Scores
35
36 bills_nona <- bills
37 bills_nona[bills_nona==99] <- NA
38 rollcalls <- rollcall(bills_nona)
39 nominate_scores <- wnominat(rollcalls , polarity=c(1, 2), minvotes=10)
40 dwnom_scores <- nominate_scores$legislators$coordID
41
42 get.similarity <- function(x, y){
43   return((2-abs(x-y))/2)
44 }
45
46
47 ## Create an adjacency/similarity matrix using ideology
48 S.ideo <- matrix(NA, ncol=70, nrow=70)
49 for (i in 1:70){
50   for (j in 1:70){
51     S.ideo[i,j] <- get.similarity(dwnom_scores[i], dwnom_scores[j])
52   }
53 }

```

```

54 diag(S.ideo) <- 0
55 S.ideo[is.na(S.ideo)==T] <- 0
56
57
58 ##### Generate expected exposure
59 perm <- replicate(perms, z[sample(1:length(z),length(z),rep=F)])
60
61 expected.exp0 <- rep(0, n)
62 expected.exp1 <- rep(0, n)
63
64 for(p in 1:ncol(perm)){
65   zp <- perm[,p]
66   for(i in 1:n){
67     if (zp[i] == 1){
68       expected.exp1[i] <- expected.exp1[i] + sum(S.ideo[i,-i]*zp[-i])
69     }
70     else{
71       expected.exp0[i] <- expected.exp0[i] + sum(S.ideo[i,]*zp)
72     }
73   }
74 }
75 num_treat <- apply(perm,1,sum)
76 num_control <- apply(1-perm,1,sum)
77 expected.exp1 <- expected.exp1/num_treat
78 expected.exp0 <- expected.exp0/num_control
79
80
81 ##### Generate expected and net exposure

```

```

82 ##### This is the spillover effect model
83
84 indirect.treatment <- function(permutation, adj.mat){ #any treatment
      assignment vector and adjacency matrix can be used
85 # permutation: can be the initial treatment assignment or a
      permutation
86 raw.exp <- rep(NA, n)
87 for (i in 1:n){
88   raw.exp[i] <- sum(adj.mat[i,]*permutation)
89 }
90
91 net.exp <- raw.exp - (permutation*expected.exp1 + (1-permutation)*
      expected.exp0)
92 standard.exp <- (net.exp - mean(net.exp))/sd(net.exp) #this is the
      spillover or indirect effect
93 return(standard.exp)
94 }
95
96
97 ##### We now model the uniformity trial transformation
98
99 z.to.unif <- function(outcome, beta1, beta2, permutation, adj.mat){
100 # outcome: vector of direct treatment outcomes
101 # beta1: direct treatment effect parameter
102 # beta2: indirect treatment effect parameter
103 # permutation: vector of a permutation of z (can be z itself)
104 # adj.mat: adjacency matrix
105

```

```

106 exposure <- indirect.treatment(permutation, adj.mat)
107 # This is equation 5
108 h.yz.0 <- outcome - (beta1*permutation) - (beta2*exposure)
109 return(h.yz.0)
110 }
111
112
113 ##### Testing and p-value calculation
114
115 beta1s <- seq(from=-.7, to=0.2, by=.025)
116 beta2s <- seq(from=-.7, to=0.2, by=.025)
117
118 pvals <- matrix(NA, length(beta1s), length(beta2s))
119
120 cl <- makeCluster(4) #Setup for parallel computing
121 registerDoParallel(cl)
122
123 pvalues.ideo <- foreach (i = 1:length(beta1s)) %do% {
124   abc <- foreach (j = 1:length(beta2s)) %do% {
125
126     # Calculate the outcome vector after taking away the effect of
127     treatment
128
129     y.0 <- z.to.unif(outcome = y.z, beta1 = beta1s[i], beta2 = beta2s[j]
130 ], permutation = z, adj.mat = S.ideo)
131
132     # Calculate observed test statistic
133
134     exposure <- indirect.treatment(permutation = z, adj.mat = S.ideo)

```

```

131   test.stat <- sum((lm(y.0 ~ z + exposure, na.action = na.omit)$resid
132 )^2)
133
134   # Calculate a vector of test statistic using permutations
135
136   results <- foreach (k = 1:perms.test) %dopar% {
137     require(permute)
138     perm.z <- z[sample(1:length(z),length(z),rep=F)] #Each time we
139     sample a permutation of z
140     perm.y.0 <- z.to.unif(outcome = y.z, beta1 = beta1s[i], beta2 =
141     beta2s[j], permutation = perm.z, adj.mat = S.ideo)
142     perm.exposure <- indirect.treatment(permutation = perm.z, adj.mat
143     = S.ideo)
144
145     perm.test.stat <- sum((lm(perm.y.0 ~ perm.z + perm.exposure, na.
146     action = na.omit)$resid)^2)
147   }
148
149   # A vector of test statistics
150   all.test.stat.vals <- as.numeric(unlist(results))
151
152   # Calculating p-value
153   pval <- sum(all.test.stat.vals < test.stat)/perms.test
154 }
155 as.numeric(unlist(abc))
156 }
157

```

```

154 stopCluster(cl)
155
156 for (i in 1:length(beta1s)){
157   pvals[i,] <- unlist(pvalues.ideology[i])
158 }
159
160 pvals
161
162
163 ## Creating a plot
164 image.plot(beta1s, beta2s, pvals,
165            main = "Plot of p-values", xlab = "Direct effects", ylab = "
            Indirect effects")
166
167 pdf("pvalues_figure.pdf")
168 image.plot(beta1s, beta2s, pvals,
169            main = "Plot of p-values",
170            xlab = "Direct effects", ylab = "Indirect effects")
171 dev.off()

```

(Conventional approaches to causal inference rely on the Stable Unit Treatment Value Unit Assumption (SUTVA) i.e. they assume that the outcome of a unit is unaffected by the treatment statuses of other units. They rely on individual-level micro attributes or context-dependent macro attributes to estimate treatment effect. However, most social processes involve complex interaction and dependence among units connected through a network. In this situation, interpersonal interactions influence behavior leading to propagation of treatment effect through network to control units and SUTVA break down. Various factors

impact how and how much the treatment effect spreads. It is important to take the interference structure into account in order to estimate the effect of the treatment effect correctly. In policy planning or marketing strategy, understanding the propagation of treatment itself can be of interest. It is generally not possible to identify the causal effects that map onto the process of social influence in observational data (Shalizi and Thomas 2011). Randomized experiments may not always be feasible. Therefore we look at methods suitable for identifying and estimating causal effect using field experiments.)

References