Using Aggregated Relational Data to feasibly identify network structure without network data

Online Appendix

Emily Breza
Arun G. Chandrasekhar
Tyler McCormick
Mengjie Pan

Appendix A. Implementation Appendix

A.1. **Cookbook.** The goal of this section is to provide a researcher or policymaker with a practical blueprint for collecting the required data and implementing our latent distance model. We propose this method in situations when the researchers want to estimate social network characteristics but when full social network maps are either infeasible or prohibitively expensive to collect.

In our preferred implementation, the researchers would collect a census of all members of the graph of interest. This approach might be feasible in settings such as a rural village, where typically there is enumeration done and basic demographics are taken for all nodes. However, we recognize that censuses might not be feasible in all settings such as a large urban slum. We include a discussion of such settings in Section A.1.1.

We envision researchers conducting the following steps:

(1) **Design ARD survey questions:** The first step is to choose which traits to use. This choice will depend on the context of the specific empirical setting. But generally-speaking, the traits should satisfy the following criteria:
   - The traits should satisfy the core assumptions of the model: that in a latent space sense they are located predominantly in one region (the distribution of individuals' latent positions is single-peaked). See Section A.2 for a more detailed discussion of this.
   - The traits should likely be observable by others (because eliciting the information in a survey relies on the observations of the respondent) and should not be subject to much measurement error (respondents should not know so many people with the trait that it is difficult for them to recall everyone, for example).
   - The number of traits should not be very long, both to avoid survey fatigue and keep costs low.[1]

(2) **Conduct census survey:** The census survey should include the following parts:
   - The ARD traits: Knowing this allows the researcher to calculate the population share of nodes in the graph with each trait k.
   - An additional set of demographic characteristics denoted by X. The vector of Xs allows for the researcher to predict the latent locations of the nodes not included in the ARD survey sample.

   See Section A.3 for a sample census questionnaire.

(3) **Conduct ARD survey:** The researchers will need to decide what share ψ of households will be surveyed. This is simply a budgetary computation, but we suggest that $\psi \geq 0.2$.

[1] However, recall that the method requires fixing the positions of three groups on the surface. Therefore, the number should be larger than five.

The ARD survey should contain:

- Link enumeration: This step is useful for providing a clear way to define a link, to aid in the interpretation of the ARD counts, and to decrease measurement error in the ARD counts. This step also gives a direct estimate of each node's degree in the sample. If the friend list methodology is not possible, we discuss how the procedure changes in Section A.1.2.
- ARD responses: For every trait in the ARD list, ask the subject to count within the enumerated list of links, how many have each trait.

See Section A.4 for a sample ARD questionnaire.

(4) **Run the ARD estimation procedure using inputs from the surveys:** Online Appendix B details how to download and execute all of the ARD estimation codes in R.

(5) **Estimate the economic parameter of interest:** See Online Appendix B for details on the estimation procedure.

A.1.1. *Census Infeasible.* In this subsection we assume that the researcher does not have access to a census of the population and has a vector of attributes for every unit (e.g., household or individual) in the population. Intuitively, the core difference between this context and the prior context is that the researcher does not have the population share by type from the census itself. This is the case for the Hyderabad urban example in Section III.B.

(3) **ARD survey:** If there is no census, then the researcher should ask every node in the ARD sample whether they have each trait. Then these sampled observations can be used to compute estimates of the population shares.

(4) **ARD estimation procedure:** Without a census, one cannot follow the procedure in Section II.D to estimate the locations of the non-ARD nodes. Instead:

- For the 1-ψ share of non-ARD nodes, draw node locations based on latent trait distributions observed in the ARD sample.
- When drawing graphs, use the estimated latent locations based on the ARD responses for the m ARD survey nodes and from this procedure for the remaining $n-m$ nodes.

A.1.2. *Link enumeration is infeasible.*

(3) **ARD survey:** Ask the subject to reflect on their friends (or links in whatever manner the researcher is trying to collect data).

- This can be recorded by the enumerators. The number of links gives the degree for each ARD node.

- If the number of links is expected to be too large for respondents to reliably count, use a N-Sum like method (see e.g. McCormick et al. (2010)).

(4) **ARD estimation procedure:** The one difference in the estimation procedure is that the expected degree of each node needs to be estimated (see Equations 2 and 3), rather than taken directly from survey responses. The code is built to accommodate this case.

A.2. **Discussion of Question Design.** Here, we discuss how to choose ARD traits to enable us to construct a good image of the network. While we leave a precise characterization of optimal questions to future research, we nonetheless can offer practical insights to aid in ARD survey design.

Conceptually ARD traits are those which, under the model, organize the latent space into regions such that nodes with certain traits are more likely to be towards the centers of those regions. Recognizing that under the model, nodes are linked as a function of their distance in this latent space, nodes are more liked to be linked to other nodes with similar such traits. This gives some insight as to which ARD features may be useful to organize the latent space.

Then when we ask, "how many of your friends have been gored by a bull" or "how many of your friends have multiple wives," those that have a positive count of this are going to have to be located somewhere close to the (latent, unknown) location of the cluster of people with this kind of experience. The reason is because we assume that the network that exists forms from the model in Equation 1, so it is most likely that someone who knows a friend that got gored by a bull and another person who has a friend who got gored by a bull are then likely to be in the same part of the latent space. What this means is that we do not need traits that "drive" the latent space per se, but traits that are informative. So a bad example might be a trait where it is peppered throughout the village. Not everyone does it, but many groups do, and so many people at very different points in the latent space are likely to have known someone who has this trait. As such, both (1) how many friends have ever experienced crop loss due to a drought and (2) how many friends do you know who have twins (in a rural setting where IVF is uncommon) would presumably be uninformative. However, something where a subcommunity engages in a practice (multiple wives) would be a better trait.

In sum, a good way to think about a useful trait, in our view, is one that is "single peaked". It should be a characteristic that is likely to be held by one group, not distributed throughout. Furthermore because traits are used to triangulate the latent space, ones that are not essentially redundant should be chosen. If the traits essentially identify the same set of people (e.g., how many friends are Muslim?; how many friends have ever gone to a mosque?), then clearly they do not add value.

A.3. **Sample Census Questionnaire.**

IDENTIFICATION.

(1) Date of Interview
(2) Surveyor Name
(3) District Name
(4) SubDistrict Name
(5) Village Name
(6) HHID
(7) GPS of the HH (marked automatically)

HOUSEHOLD IDENTIFIERS.

(1) What is the name of the respondent ?
(2) What is the name of the Household Head ?
(3) What is the caste of the household head?
(4) What is the sub-caste of the household head?
(5) Does the Household have an electricity connection?
(6) What type of roofing material does the household have?
(7) Does the Household own land ?
(8) Does the Household have a toilet ?

ARD TRAITS.

(1) Does the House have 2 or greater than 2 floors ?
(2) Does the respondent own a kirana shop / tea/ sweets shop/PDS shop?
(3) Has any member in your household migrated to another city for labor or construction work in the last 2 years?
(4) Does any member in your household own a bike?
(5) Does the respondents' house have iron/steel gates?
(6) Has any member in the household passed the 12th Standard?
(7) Does anyone in your household own a goat/hen?
(8) Is any member in the household a government Employee?
(9) Does anyone in your household have a smart phone?
(10) Did any adult in the household have typhoid, malaria, or cholera in the past six months ?
(11) Does the house have 5 or greater than 5 children below the age of 18 ?
(12) Is anyone in your Household a member of religious or cultural committee at the village level ?

A.4. **Sample ARD Questionnaire.**

IDENTIFICATION.

(1) Enter HHID
(2) Village Name
(3) District Name
(4) SubDistrict Name
(5) Gram Panchayat Name
(6) Name of the Respondent

FRIEND LIST ELICITATION. *Instruction to Enumerator: Note down the list of names as given by the respondent. As you note down the names make sure that names that are repeated are marked, so that at the end of the 3 questions, we have a list of unique friends*

(1) Tell the names of the Household Heads of those families in this village whose house you visit or who visit your house frequently or with whom you socialize frequently ?
(2) Tell the names of Household Heads of those families in this village who give you advice/ or to whom you give advice on farming/health/financial issues? *(Ask each part separately)*
(3) If you urgently needed kerosene/charcoal, rice or money, who do you borrow them from or who borrows it from you? *(Ask each part separately)*

ARD. *Instruction to Enumerator:*

- *Inform the respondent of the name and no of friends that have been named in the previous section*
- *Tell the respondent that the questions in this section pertain to the friends named in the previous section*

Out of all the households whose name you took in the previous section, how many have the following traits :

(1) No of floors in the house are greater than or equal to 2?
(2) No of households out of your friend list who own a kirana shop/ tea/ sweets shop/ PDS?
(3) No of households out of your friend lis wherein any member migrated to another city for labor/construction work in the last 2 years?
(4) No of households among your friend list who own a bike?
(5) No of households among your friend list whose house have iron/steel gates?
(6) No of households among your friend list wherein any member has passed the 12th Standard?
(7) No of households among your friend list which own goats/ hen?

(8) No of households among your friend list where in any member is a government Employee?

(9) No of households among your friend list where someone has a smart phone?

(10) No of households among your friend list where any adult has had typhoid, malaria, or cholera in the past six months?

(11) No of households among your friend list which have 5 or greater than 5 children below the age of 18?

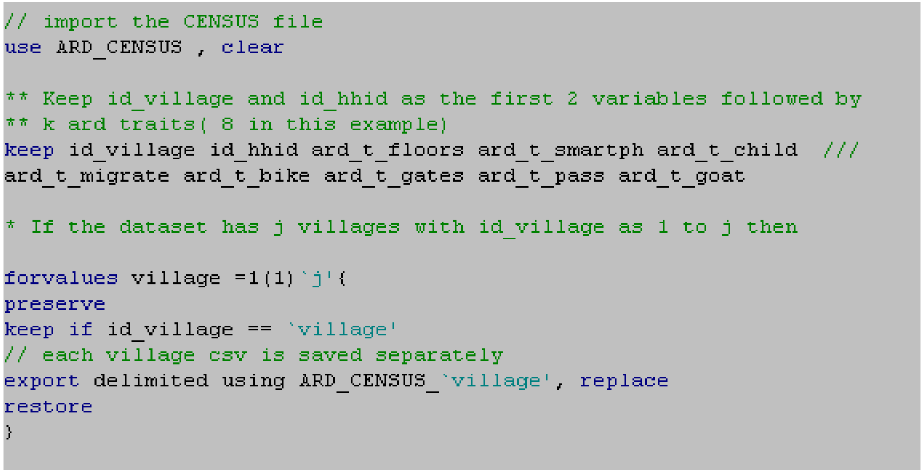
(12) No of households among your friend list where anyone is a member of religious or cultural committee at the village level?

(13) No of households among your friend list who belong to the Scheduled Caste?

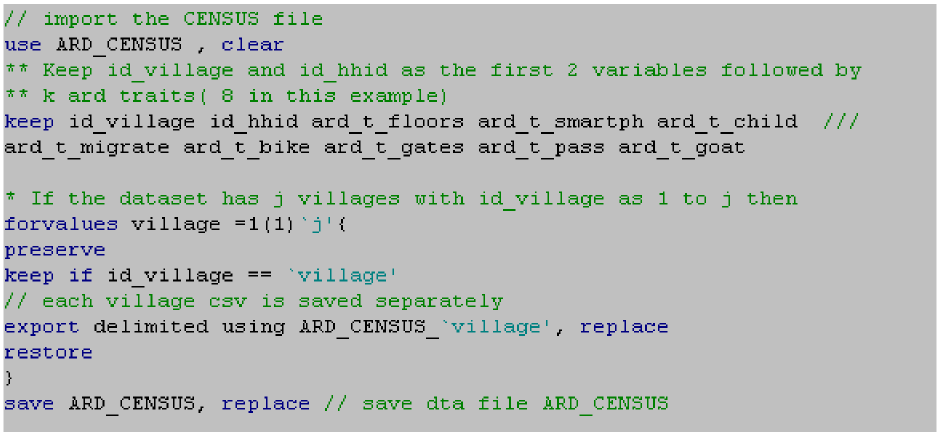
Appendix B. Detailed Estimation Procedure

This section presents the detailed walk-through for the estimation procedure. We assume the researcher has csv or xls data and is familiar with Stata and (to a lesser degree) R (R Core Team, 2018). We walk the researcher step-by-step moving from the raw data, through Stata, through R (with code provided) which outputs csv files, back to Stata in order to conduct estimation of interest. Of course, if the researcher is comfortable with R, the procedure is far more streamlined and immediate.

(1) Download ARD code: `https://github.com/MengjiePan/BCMP`
(2) Format survey data in the following manner:
  - Create a dataset(csv,xls) that is m ARD nodes by K ARD responses for each village and save each file as ARD_SURVEY_i.csv
  - Create a dataset that is n nodes by the K ARD-trait covariates from the census for each village and save each file as ARD_CENSUS_i.csv
  - Create a dataset that is m ARD nodes by the L covariates from the census (e.g., GPS, household identifiers). Create another dataset that is $n-m$ Non ARD nodes by the L covariates from the census(same covariates as used for ARD Nodes). Use L covariates of these two datasets in a distance function to create a $n-m$ by m dataset. This will be used in k-nearest neighbours algorithm. Save each file as distance_i.csv

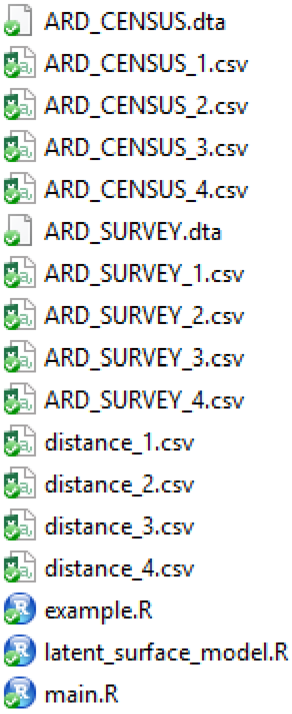


```
// import the CENSUS file
use ARD_CENSUS , clear

** Keep id_village and id_hhid as the first 2 variables followed by
** k ard traits( 8 in this example)
keep id_village id_hhid ard_t_floors ard_t_smartph ard_t_child  ///
ard_t_migrate ard_t_bike ard_t_gates ard_t_pass ard_t_goat

* If the dataset has j villages with id_village as 1 to j then

forvalues village =1(1)`j'{
preserve
keep if id_village == `village'
// each village csv is saved separately
export delimited using ARD_CENSUS_`village', replace
restore
}
```



```
// import the CENSUS file
use ARD_CENSUS , clear
** Keep id_village and id_hhid as the first 2 variables followed by
** k ard traits( 8 in this example)
keep id_village id_hhid ard_t_floors ard_t_smartph ard_t_child  ///
ard_t_migrate ard_t_bike ard_t_gates ard_t_pass ard_t_goat

* If the dataset has j villages with id_village as 1 to j then
forvalues village =1(1)`j'{
preserve
keep if id_village == `village'
// each village csv is saved separately
export delimited using ARD_CENSUS_`village', replace
restore
}
save ARD_CENSUS, replace // save dta file ARD_CENSUS
```

(3) Copy the downloaded R files in the same folder. The folder structure should be as shown in the figure below(for 4 villages)

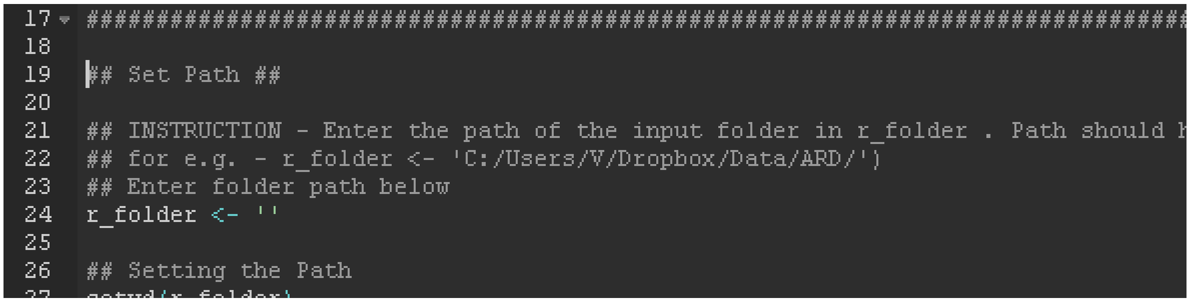


ARD_CENSUS.dta
ARD_CENSUS_1.csv
ARD_CENSUS_2.csv
ARD_CENSUS_3.csv
ARD_CENSUS_4.csv
ARD_SURVEY.dta
ARD_SURVEY_1.csv
ARD_SURVEY_2.csv
ARD_SURVEY_3.csv
ARD_SURVEY_4.csv
distance_1.csv
distance_2.csv
distance_3.csv
distance_4.csv
example.R
latent_surface_model.R
main.R

(4) Open the file example.R

(5) Download R Packages - `igraph`(Csardi and Nepusz, 2006) , `movMF`(Hornik and Grün, 2014), `xlsx`(Dragulescu et al., 2018) (if the datasets are in xls), `readstata13`(Garbuszus and Jeworutzki, 2018) (if the datasets are in Stata 13,14) [example.R downloads these packages]

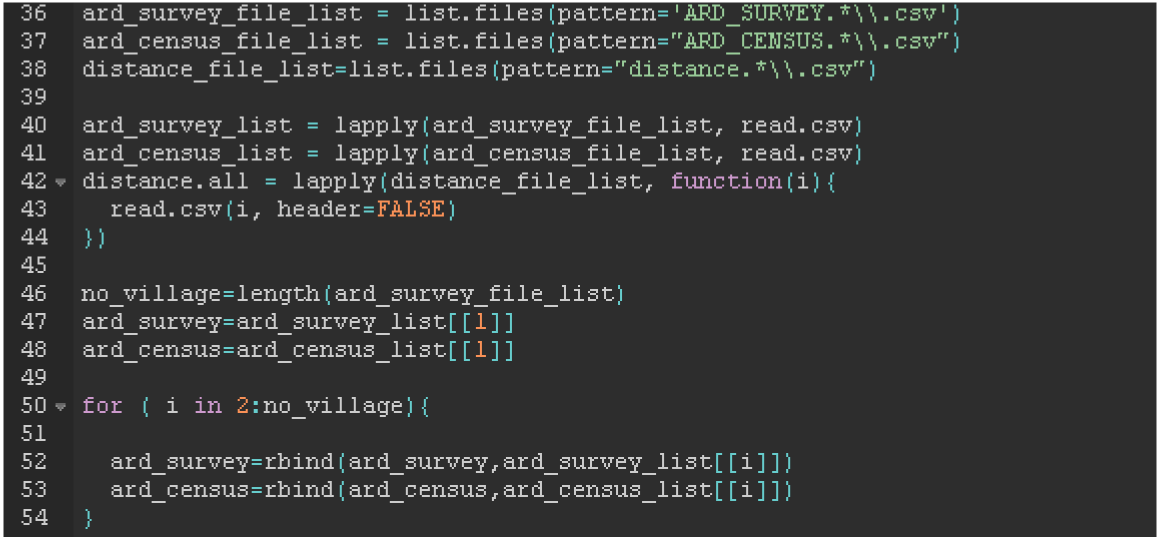
(6) Enter the path to the folder in variable `r_folder` (Line 24). **Running the R Script example.R now** should generate the ARD Output in the Folder `OUT` in the current folder. The steps given next explain the process in detail through code snippets.



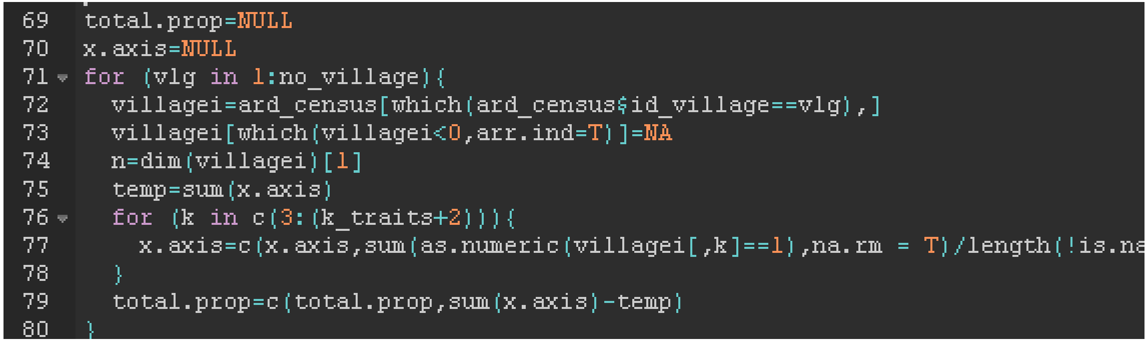
```
17  ################################################################################
18
19  ## Set Path ##
20
21  ## INSTRUCTION - Enter the path of the input folder in r_folder . Path should b
22  ## for e.g. - r_folder <- 'C:/Users/V/Dropbox/Data/ARD/')
23  ## Enter folder path below
24  r_folder <- ''
25
26  ## Setting the Path
27  setwd(r_folder)
```

(7) Preparing the datasets for constructing ARD :

- The datasets created in Step 2 are imported(Line 36-54) and are named `ard_survey`, `ard_census` and `distance.all` respectively
- Calculate the value of variable `total.prop` - *fraction of ties in the network that are made with members of group k, summed over K groups* using `example.R` (Line no 69-80). The variable `villagei` stores the `ard_census` traits.



```
36  ard_survey_file_list = list.files(pattern='ARD_SURVEY.*\\.csv')
37  ard_census_file_list = list.files(pattern="ARD_CENSUS.*\\.csv")
38  distance_file_list=list.files(pattern="distance.*\\.csv")
39
40  ard_survey_list = lapply(ard_survey_file_list, read.csv)
41  ard_census_list = lapply(ard_census_file_list, read.csv)
42  distance.all = lapply(distance_file_list, function(i){
43    read.csv(i, header=FALSE)
44  })
45
46  no_village=length(ard_survey_file_list)
47  ard_survey=ard_survey_list[[1]]
48  ard_census=ard_census_list[[1]]
49
50  for ( i in 2:no_village){
51
52    ard_survey=rbind(ard_survey,ard_survey_list[[i]])
53    ard_census=rbind(ard_census,ard_census_list[[i]])
54  }
```

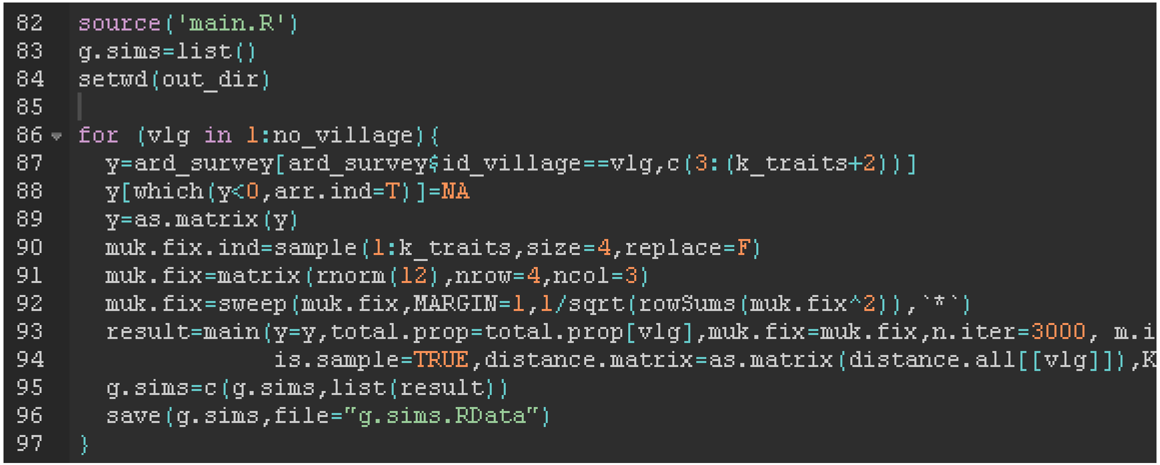


```
69  total.prop=NULL
70  x.axis=NULL
71  for (vlg in 1:no_village){
72    villagei=ard_census[which(ard_census$id_village==vlg),]
73    villagei[which(villagei<0,arr.ind=T)]=NA
74    n=dim(villagei)[1]
75    temp=sum(x.axis)
76    for (k in c(3:(k_traits+2))){
77      x.axis=c(x.axis,sum(as.numeric(villagei[,k]==1),na.rm = T)/length(!is.na
78    }
79    total.prop=c(total.prop,sum(x.axis)-temp)
80  }
```

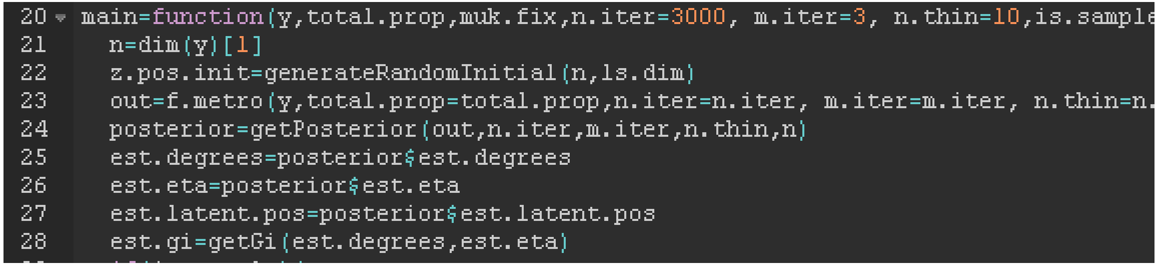
(8) Estimate the parameters of the model: $(\nu_i, z_i)_{i=1}^m$ for the m ARD households, ζ, $(\upsilon_k, \eta_k)_{k=1}^m$ (the latent trait distribution location and concentration parameters).

- Use `example.R` to call(Line 93) `main.R`, which calls(Line 23) function `f.metro` in `latent_surface_model.R`.
- The call to function `main` of `main.R` on Line 93 requires 4 input variables

  - `y` - use the `ard_survey` dataset that has been imported
  - `total.prop` - Calculated in Step 4
  - `muk.fix` - the positions of fixed variables calculated in Line 126-127 of `example.R`
  - `distance.matrix` - use the `distance.all` dataset that has been imported
- The Output of the call to `f.metro` is stored in variable `posterior` of `main.R`



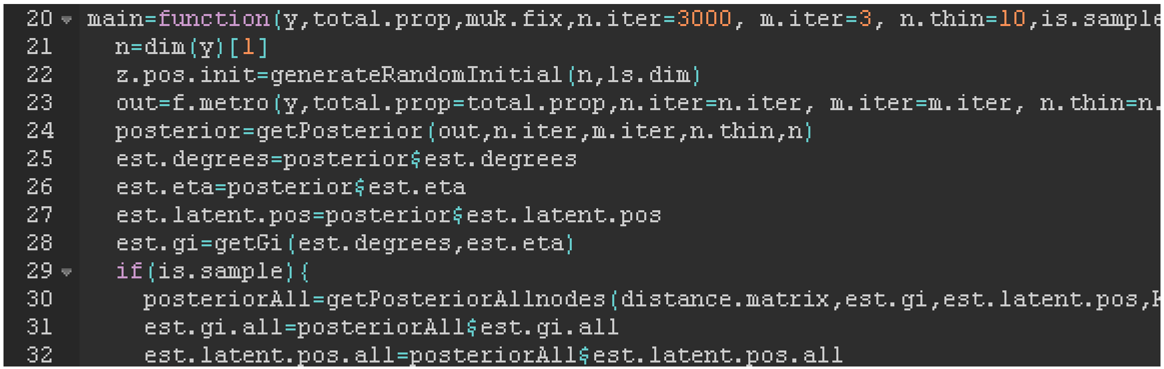
```
82  source('main.R')
83  g.sims=list()
84  setwd(out_dir)
85  
86  for (vlg in 1:no_village){
87    y=ard_survey[ard_survey$id_village==vlg,c(3:(k_traits+2))]
88    y[which(y<0,arr.ind=T)]=NA
89    y=as.matrix(y)
90    muk.fix.ind=sample(1:k_traits,size=4,replace=F)
91    muk.fix=matrix(rnorm(12),nrow=4,ncol=3)
92    muk.fix=sweep(muk.fix,MARGIN=1,1/sqrt(rowSums(muk.fix^2)),`*`)
93    result=main(y=y,total.prop=total.prop[vlg],muk.fix=muk.fix,n.iter=3000, m.i
94                is.sample=TRUE,distance.matrix=as.matrix(distance.all[[vlg]]),K
95    g.sims=c(g.sims,list(result))
96    save(g.sims,file="g.sims.RData")
97  }
```



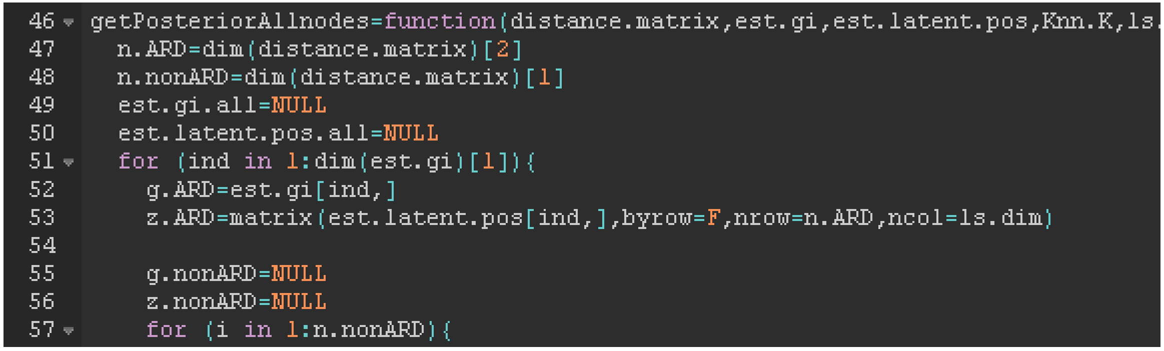
```
20  main=function(y,total.prop,muk.fix,n.iter=3000, m.iter=3, n.thin=10,is.sample
21    n=dim(y)[1]
22    z.pos.init=generateRandomInitial(n,ls.dim)
23    out=f.metro(y,total.prop=total.prop,n.iter=n.iter, m.iter=m.iter, n.thin=n.
24    posterior=getPosterior(out,n.iter,m.iter,n.thin,n)
25    est.degrees=posterior$est.degrees
26    est.eta=posterior$est.eta
27    est.latent.pos=posterior$est.latent.pos
28    est.gi=getGi(est.degrees,est.eta)
```

(9) Estimate ν_i and z_i for the $n - m$ nodes that are in the census but not the ARD sample.

- `main.R` (Line no 30) calls function `getPosteriorAllnodes` in `main.R`. The call to the function takes variable `distance.matrix` as an input(which had been passed to function `main` from `example.R` in Step 5)
- Output is stored in variable `posteriorAll`. The estimated latent positions z_i are stored as an attribute of `posteriorAll` as `est.latent.pos.all`
- `getPosteriorAllnodes` estimates ν_i and z_i using k-means from `distance.matrix` variable. This variable has been calculated using the $K + L$ covariates for the m nodes in the ARD sample and $n - m$ Non-ARD nodes



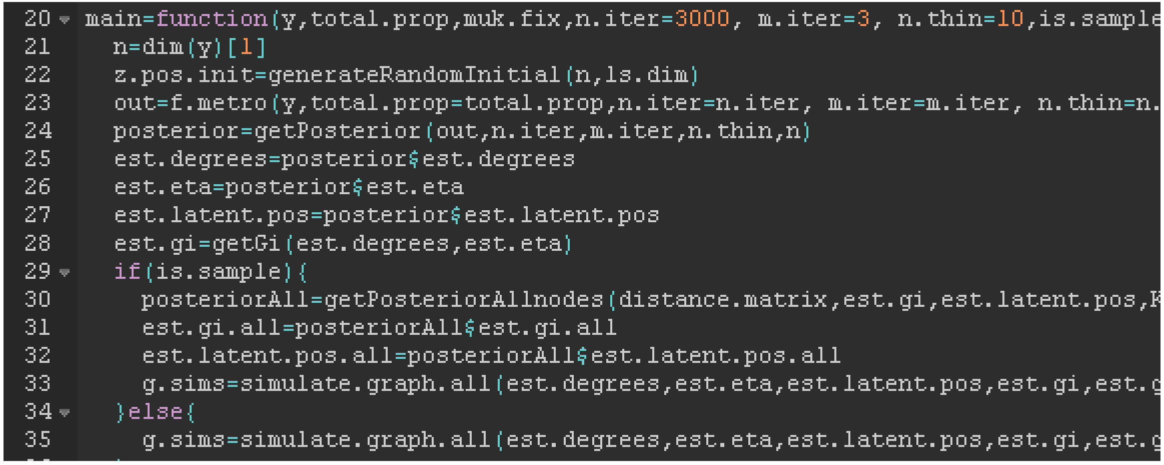
```
20  main=function(y,total.prop,muk.fix,n.iter=3000, m.iter=3, n.thin=10,is.sample
21    n=dim(y)[1]
22    z.pos.init=generateRandomInitial(n,ls.dim)
23    out=f.metro(y,total.prop=total.prop,n.iter=n.iter, m.iter=m.iter, n.thin=n.
24    posterior=getPosterior(out,n.iter,m.iter,n.thin,n)
25    est.degrees=posterior$est.degrees
26    est.eta=posterior$est.eta
27    est.latent.pos=posterior$est.latent.pos
28    est.gi=getGi(est.degrees,est.eta)
29    if(is.sample){
30      posteriorAll=getPosteriorAllnodes(distance.matrix,est.gi,est.latent.pos,K
31      est.gi.all=posteriorAll$est.gi.all
32      est.latent.pos.all=posteriorAll$est.latent.pos.all
```



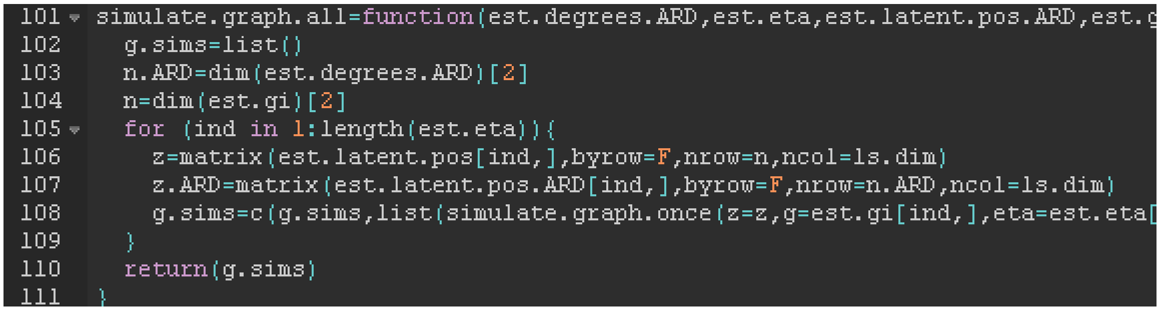
```
46  getPosteriorAllnodes=function(distance.matrix,est.gi,est.latent.pos,Knn.K,ls.
47    n.ARD=dim(distance.matrix)[2]
48    n.nonARD=dim(distance.matrix)[1]
49    est.gi.all=NULL
50    est.latent.pos.all=NULL
51    for (ind in 1:dim(est.gi)[1]){
52      g.ARD=est.gi[ind,]
53      z.ARD=matrix(est.latent.pos[ind,],byrow=F,nrow=n.ARD,ncol=ls.dim)
54
55      g.nonARD=NULL
56      z.nonARD=NULL
57      for (i in 1:n.nonARD){
```

(10) Draw a set of $b = 1, \ldots, B$ draws from the network formation probability model (now with estimated parameters for all nodes) from the posterior distribution.

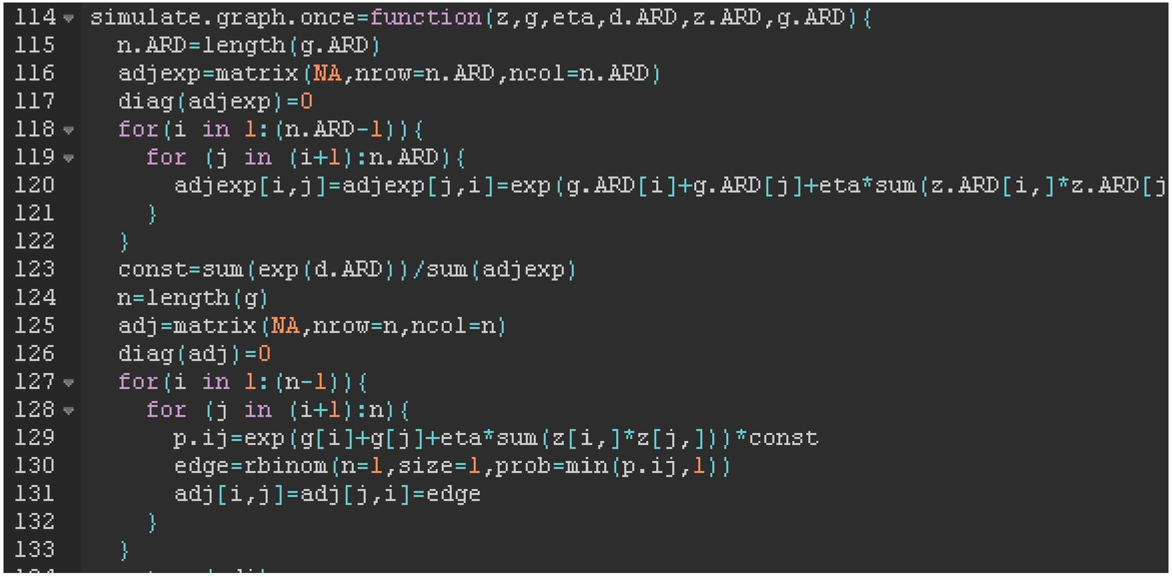
- Use `main.R` (Line no 33) to call function `simulate.graph.all` . The output is stored in variable `g.sims` . `simulate.graph.all` calls(Line 108) `simulate.graph.once` for each run.
- Draw a parameter vector θ (all the above parameters) from the posterior.
- Draw a graph g_b given θ_b. (Line 130 - function `simulate.graph.once`)



```
20  main=function(y,total.prop,muk.fix,n.iter=3000, m.iter=3, n.thin=10,is.sample
21    n=dim(y)[1]
22    z.pos.init=generateRandomInitial(n,ls.dim)
23    out=f.metro(y,total.prop=total.prop,n.iter=n.iter, m.iter=m.iter, n.thin=n.
24    posterior=getPosterior(out,n.iter,m.iter,n.thin,n)
25    est.degrees=posterior$est.degrees
26    est.eta=posterior$est.eta
27    est.latent.pos=posterior$est.latent.pos
28    est.gi=getGi(est.degrees,est.eta)
29    if(is.sample){
30      posteriorAll=getPosteriorAllnodes(distance.matrix,est.gi,est.latent.pos,K
31      est.gi.all=posteriorAll$est.gi.all
32      est.latent.pos.all=posteriorAll$est.latent.pos.all
33      g.sims=simulate.graph.all(est.degrees,est.eta,est.latent.pos,est.gi,est.g
34    }else{
35      g.sims=simulate.graph.all(est.degrees,est.eta,est.latent.pos,est.gi,est.g
```



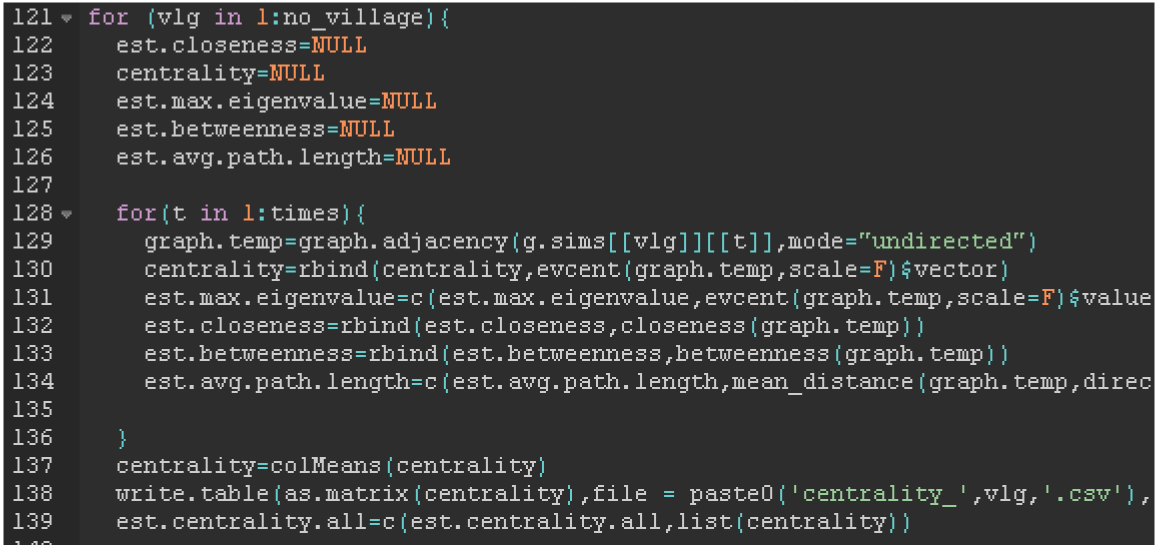
```r
101 simulate.graph.all=function(est.degrees.ARD,est.eta,est.latent.pos.ARD,est.g
102   g.sims=list()
103   n.ARD=dim(est.degrees.ARD)[2]
104   n=dim(est.gi)[2]
105   for (ind in 1:length(est.eta)){
106     z=matrix(est.latent.pos[ind,],byrow=F,nrow=n,ncol=ls.dim)
107     z.ARD=matrix(est.latent.pos.ARD[ind,],byrow=F,nrow=n.ARD,ncol=ls.dim)
108     g.sims=c(g.sims,list(simulate.graph.once(z=z,g=est.gi[ind,],eta=est.eta[
109   }
110   return(g.sims)
111 }
```



```r
114 simulate.graph.once=function(z,g,eta,d.ARD,z.ARD,g.ARD){
115   n.ARD=length(g.ARD)
116   adjexp=matrix(NA,nrow=n.ARD,ncol=n.ARD)
117   diag(adjexp)=0
118   for(i in 1:(n.ARD-1)){
119     for (j in (i+1):n.ARD){
120       adjexp[i,j]=adjexp[j,i]=exp(g.ARD[i]+g.ARD[j]+eta*sum(z.ARD[i,]*z.ARD[j
121     }
122   }
123   const=sum(exp(d.ARD))/sum(adjexp)
124   n=length(g)
125   adj=matrix(NA,nrow=n,ncol=n)
126   diag(adj)=0
127   for(i in 1:(n-1)){
128     for (j in (i+1):n){
129       p.ij=exp(g[i]+g[j]+eta*sum(z[i,]*z[j,]))*const
130       edge=rbinom(n=1,size=1,prob=min(p.ij,1))
131       adj[i,j]=adj[j,i]=edge
132     }
133   }
```

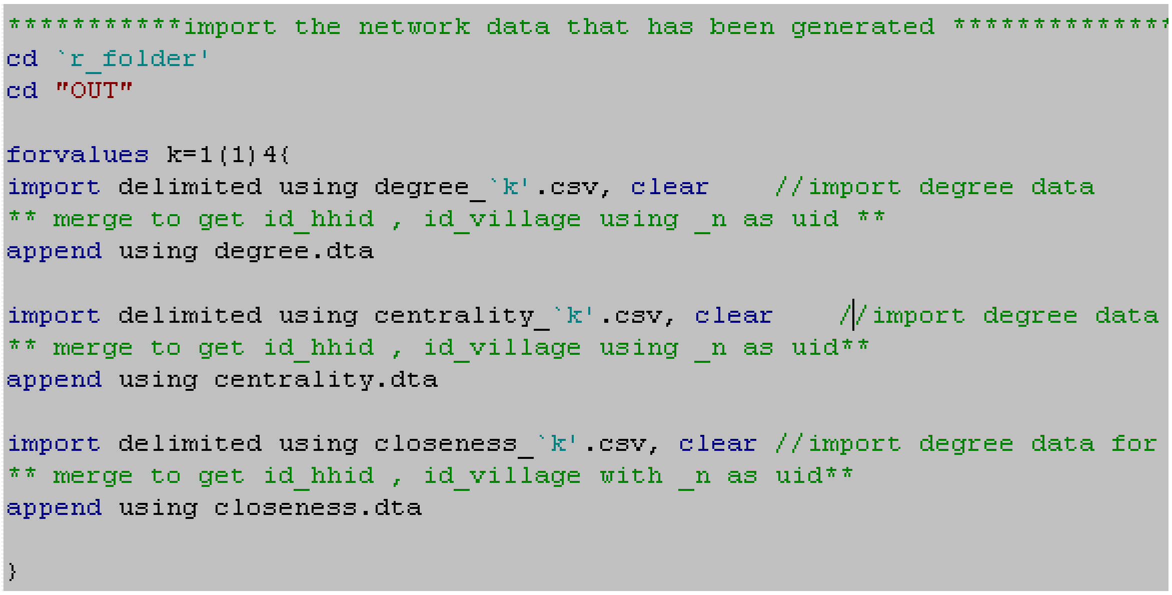
(11) Compute network statistics of interest $S(g_b)$ for each draw g_b for $b = 1, ..., B$.

- Construct your own desired functions
- Or use a suggested code `example.R` (Line no 115-144)



```r
121 for (vlg in 1:no_village){
122   est.closeness=NULL
123   centrality=NULL
124   est.max.eigenvalue=NULL
125   est.betweenness=NULL
126   est.avg.path.length=NULL
127 
128   for(t in 1:times){
129     graph.temp=graph.adjacency(g.sims[[vlg]][[t]],mode="undirected")
130     centrality=rbind(centrality,evcent(graph.temp,scale=F)$vector)
131     est.max.eigenvalue=c(est.max.eigenvalue,evcent(graph.temp,scale=F)$value
132     est.closeness=rbind(est.closeness,closeness(graph.temp))
133     est.betweenness=rbind(est.betweenness,betweenness(graph.temp))
134     est.avg.path.length=c(est.avg.path.length,mean_distance(graph.temp,direc
135 
136   }
137   centrality=colMeans(centrality)
138   write.table(as.matrix(centrality),file = paste0('centrality_',vlg,'.csv'),
139   est.centrality.all=c(est.centrality.all,list(centrality))
```

(12) Import the network characteristics that have been generated in folder `OUT`



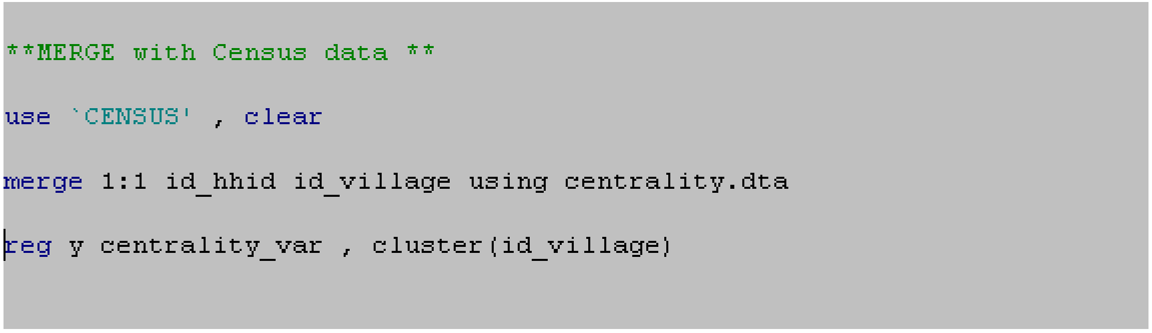
```
***********import the network data that has been generated **************
cd `r_folder'
cd "OUT"

forvalues k=1(1)4{
import delimited using degree_`k'.csv, clear    //import degree data
** merge to get id_hhid , id_village using _n as uid **
append using degree.dta

import delimited using centrality_`k'.csv, clear    //import degree data
** merge to get id_hhid , id_village using _n as uid**
append using centrality.dta

import delimited using closeness_`k'.csv, clear //import degree data for
** merge to get id_hhid , id_village with _n as uid**
append using closeness.dta

}
```

(13) Import the graph simulations that have been generated from folder `OUT/SIMULATION`

(14) Conduct economic estimation of interest. For instance,

$$y_{iv} = \alpha + \beta \frac{1}{B} \sum_{b=1}^{B} S(g)_{iv,b} + \epsilon_{iv},$$

to estimate β, which is the parameter of interest in this example, where i is a node and v is the independent network for $v = 1, ..., V$ networks in the sample.



```

**MERGE with Census data **

use `CENSUS' , clear

merge 1:1 id_hhid id_village using centrality.dta

reg y centrality_var , cluster(id_village)
```

Appendix C. ARD Questions from Banerjee, Breza, Duflo, and Kinnan (2016)

This section presents the ARD questions used in Banerjee, Breza, Duflo, and Kinnan (2016) that we use in Section III.B.

How many other households do you know in your neighborhood ...

(1) where a woman has ever given birth to twins?
(2) where there is a permanent government employee?
(3) where there are 5 or more children?
(4) where any child has studied past 10th standard?
(5) where any adult has had typhoid, malaria, or cholera in the past six months?
(6) where any adult has been arrested by the police?
(7) where at least one woman has had a second marriage?
(8) where at least one man currently has more than one wife?

References

Banerjee, A., E. Breza, E. Duflo, and C. Kinnan (2016): "Do credit constraints limit entrepreneurship: Heterogeneity in the returns to microfinance," *Working Paper*. C

Csardi, G. and T. Nepusz (2006): "The igraph software package for complex network research," *InterJournal, Complex Systems*, 1695, 1–9. 5

Dragulescu, A. A., M. A. A. Dragulescu, and R. Provide (2018): "Package xlsx," *Cell*, 9, 1. 5

Garbuszus, J. M. and S. Jeworutzki (2018): *readstata13: Import 'Stata' Data Files*, r package version 0.9.2. 5

Hornik, K. and B. Grün (2014): "movMF: an R package for fitting mixtures of von Mises-Fisher distributions," *Journal of Statistical Software*, 58, 1–31. 5

McCormick, T. H., M. J. Salganik, and T. Zheng (2010): "How many people do you know?: Efficiently estimating personal network size," *Journal of the American Statistical Association*, 105, 59–70. A.1.2

R Core Team (2018): *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. B