R Code for Social Network Analysis

Erik Gjesfjeld

This R Markdown document contains R code for social network analysis of compositional pottery data from the Kuril Islands. Documentation of this approach can be found in Gjesfjeld and Phillips (2013). Please do not cite without permission from the author (erik.gjesfjeld@gmail.com).

This R code assumes that a matrix specifying network relationships between archaeological sites has already been established. If using geochemical compositional data, this is often through the use of cluster analysis, prinicipal component analysis and Mahalnobis Distance assignment as outlined by Glascock et al. (2004). This also assumes that you have installed the necessary packages into the R environment (sna)

Citations:

Gjesfjeld, Erik and S. Colby Phillips
2013 Evaluating Adaptive Network Strategies with Geochemical Sourcing Data: A Case Study from the Kuril Islands. In *Network Analysis in Archaeology*, edited Carl Knappett, pp. 281-305. Oxford University Press.

Glascock, M., H. Neff and K.J. Vaughn 2004 Instrumental Neutron Activation Analysis and Multivariate Statistics for Pottery Provenanace. *Hyperfine Interactions*. 154(1-4):pp 95-105.

1. *Installing library and setting the working directory*

```
library(sna)
setwd("~/Dropbox/SNA")
```

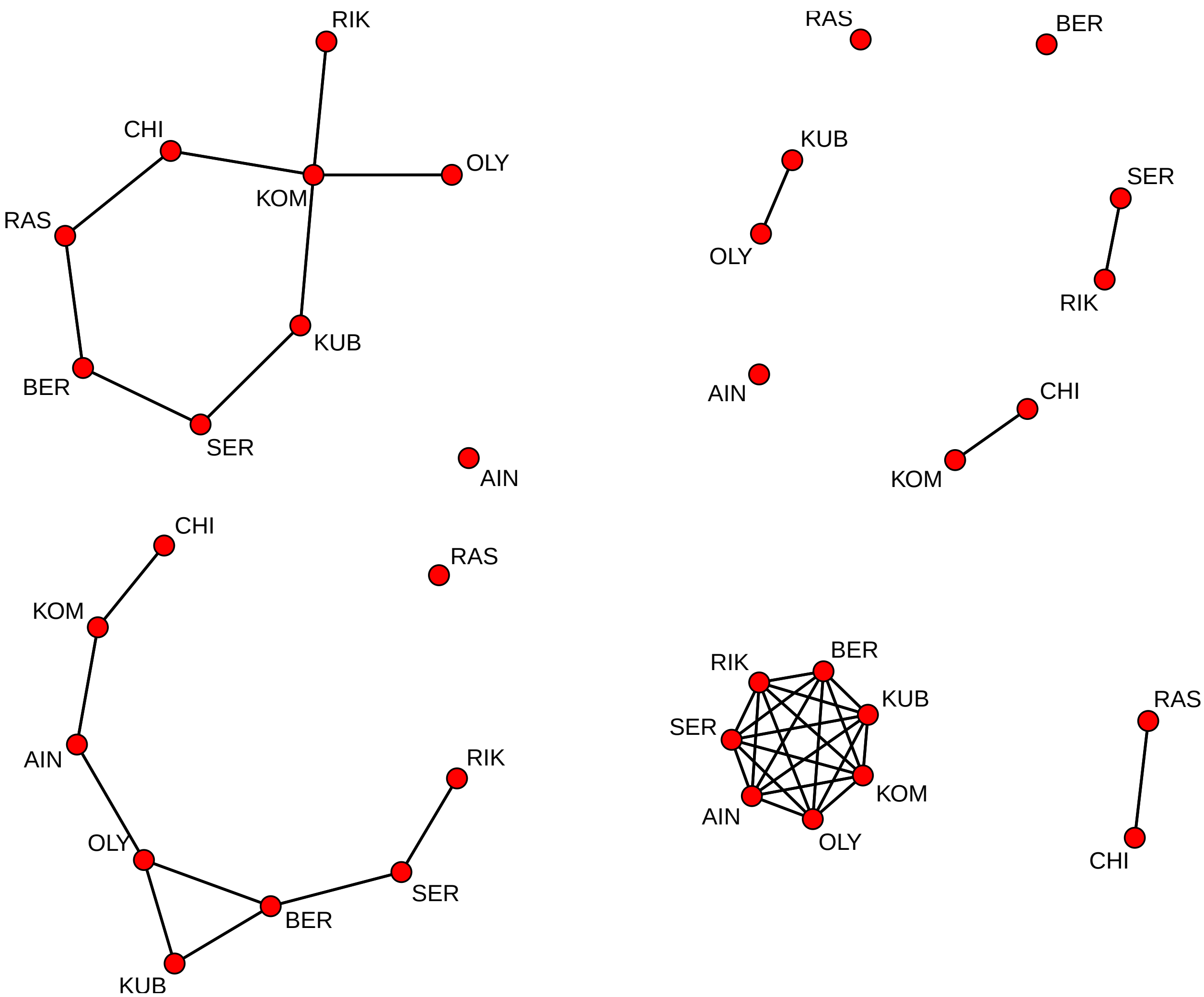
2. *Reading in the raw data from text files* Raw data can be found here on Github

```
#Reading in matrices for the different network graphs
EJ_observed<-read.delim("EJ-observed.txt",header=T,row.names = 1)
EJ_local<-read.delim("EJ-local.txt",header=T,row.names =1)
EJ_recip<-read.delim("EJ-recip.txt",header=T,row.names =1)
EJ_central<-read.delim("EJ-central.txt",header = T, row.names = 1)
Okh_observed<-read.delim("Okh-observed.txt",header=T,row.names = 1)
Okh_local<-read.delim("Okh-local.txt",header=T,row.names =1)
Okh_recip<-read.delim("Okh-recip.txt",header=T,row.names =1)
Okh_central<-read.delim("Okh-central.txt",header = T, row.names = 1)
```

3. Creating Network Graphs

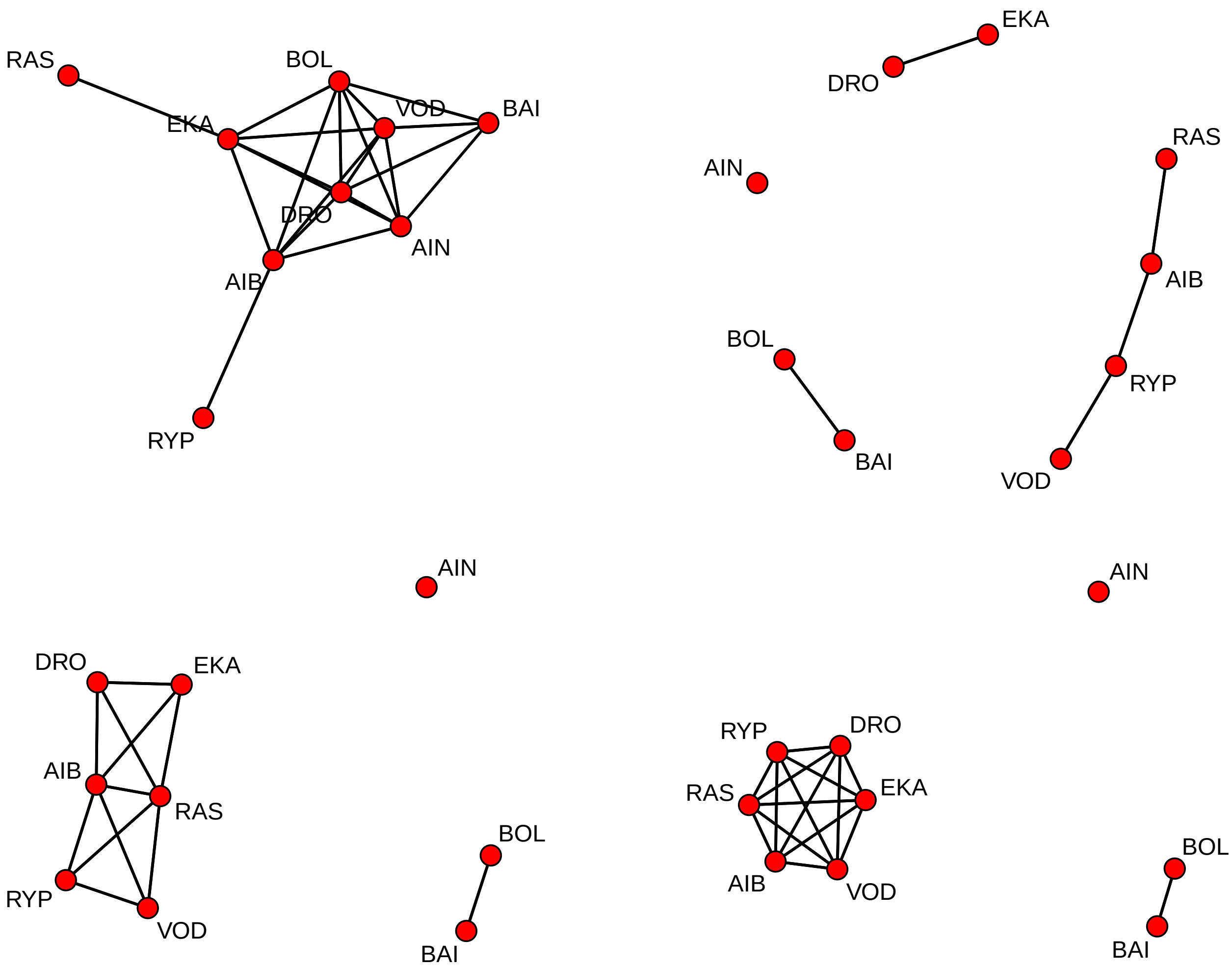
Epi-Jomon Graphs

```
par(mfrow=c(2,2)) #Plotting four graphs together
#par(mfrow=c(1,1)) #One graph at a time
par(mar=c(0.5,0.5,0.5,0.5)) #reducing margins inbetween graphs
gplot(EJ_observed,gmode="graph", displaylabels = TRUE) #geochem data
gplot(EJ_local,gmode="graph",displaylabels = TRUE) #predicted local network
gplot(EJ_recip,gmode="graph",displaylabels = TRUE)  #Predicated reciprocal network
gplot(EJ_central,gmode="graph",displaylabels = TRUE) #predicted central-place network
```



Okhotsk Graphs

```
par(mfrow=c(2,2)) #Plotting four graphs together
#par(mfrow=c(1,1)) #One graph at a time
par(mar=c(0.5,0.5,0.5,0.5)) #reducing margins inbetween graphs
gplot(Okh_observed,gmode="graph", displaylabels = TRUE) #geochem data
gplot(Okh_local,gmode="graph",displaylabels = TRUE) #predicted local network
gplot(Okh_recip,gmode="graph",displaylabels = TRUE)  #Predicated reciprocal network
gplot(Okh_central,gmode="graph",displaylabels = TRUE) #predicted central-place network
```



4. Getting Centrality Scores for Observed Network

Epi-Jomon Centrality Scores

```
EJ_deg<-round(degree(EJ_observed,gmode="graph"),digits = 3) #Degree centrality
EJ_between<-round(betweenness(EJ_observed,gmode="graph"),digits = 3)
EJ_evcent<-round(evcent(EJ_observed,gmode="graph"), digits = 3)
```

Site	Degree	Betweenness	Eigenvector
AIN	0	0	0
BER	2	2	0.29
CHI	2	5	0.44
KOM	4	13	0.49
KUB	2	5	0.44
OLY	1	0	0.29
RAS	2	3	0.24
RIK	1	0	0.29
SER	2	3	0.24

Okhotsk Centrality Scores

```
Okh_deg<-round(degree(Okh_observed,gmode="graph"),digits = 3) #Degree centrality
Okh_between<-round(betweenness(Okh_observed,gmode="graph"),digits = 3)
Okh_evcent<-round(evcent(Okh_observed,gmode="graph"), digits = 3)
```

Site	Degree	Betweenness	Eigenvector
AIB	6	7	0.37
AIN	6	0.5	0.36
BAI	3	0	0.29
BOL	6	1.33	0.41
DRO	6	1.33	0.41
EKA	6	7	0.37
RAS	1	0	0.07
RYP	1	0	0.07
VOD	6	1.33	0.41

5. Graph Density Scores for All Networks

Epi-Jomon

```
EJ_obs_gden<-round(gden(EJ_observed,mode="graph"),digits=3)
EJ_local_gden<-round(gden(EJ_local,mode="graph"),digits=3)
EJ_recip_gden<-round(gden(EJ_recip,mode="graph"),digits=3)
EJ_central_gden<-round(gden(EJ_central,mode="graph"),digits=3)
```

Okhotsk

```
Okh_obs_gden<-round(gden(Okh_observed,mode="graph"),digits=3)
Okh_local_gden<-round(gden(Okh_local,mode="graph"),digits=3)
Okh_recip_gden<-round(gden(Okh_recip,mode="graph"),digits=3)
Okh_central_gden<-round(gden(Okh_central,mode="graph"),digits=3)
```

	Observed	Local	Reciprocal	Central-Place
Epi-Jomon - Graph Density	0.22	0.08	0.22	0.61
Okhotsk - Graph Density	0.57	0.14	0.33	0.44

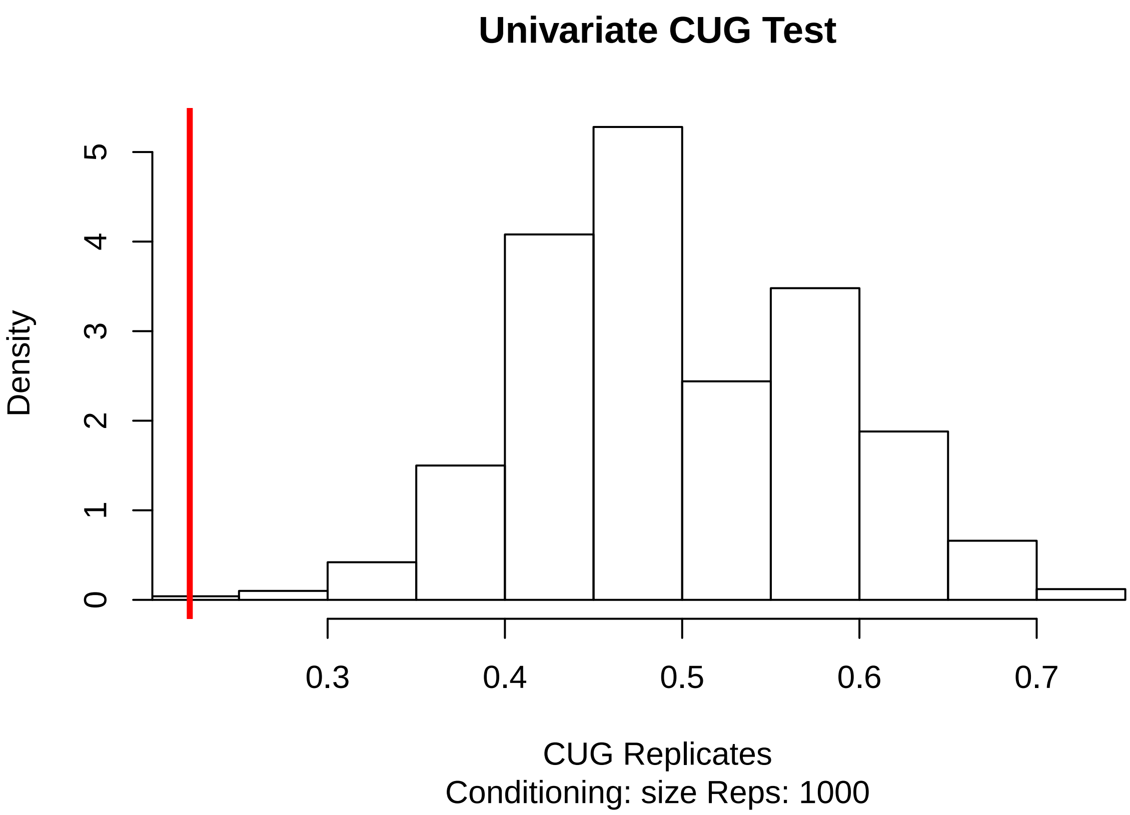
6. Conditional Uniform Graph Test

Epi-Jomon

```
EJ_cug_results<-cug.test(EJ_observed,gden,mode="graph")
EJ_cug_results
```

```
##
## Univariate Conditional Uniform Graph Test
##
## Conditioning Method: size
## Graph Type: graph
## Diagonal Used: FALSE
## Replications: 1000
##
## Observed Value: 0.2222222
## Pr(X>=Obs): 1
## Pr(X<=Obs): 0.001
```

```
plot(EJ_cug_results)
```

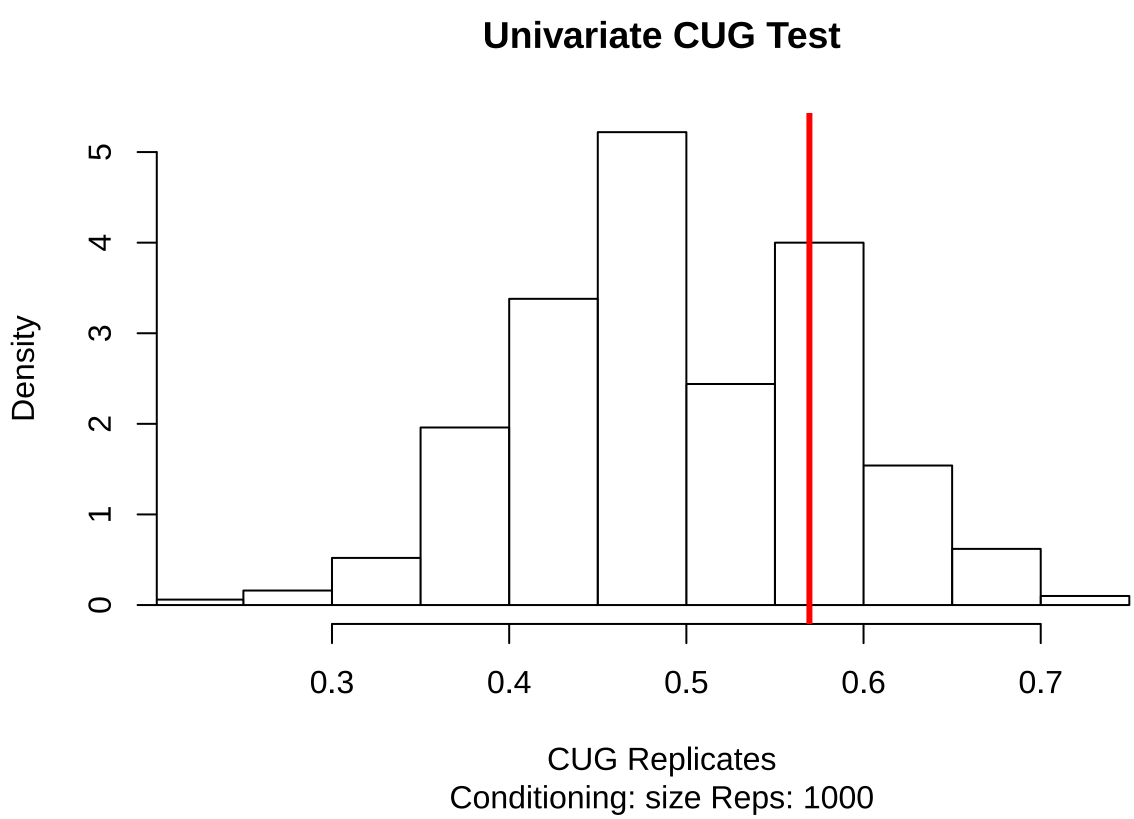


Okhotsk

```
Okh_cug_results<-cug.test(Okh_observed,gden,mode="graph")
Okh_cug_results
```

```
##
## Univariate Conditional Uniform Graph Test
##
## Conditioning Method: size
## Graph Type: graph
## Diagonal Used: FALSE
## Replications: 1000
##
## Observed Value: 0.5694444
## Pr(X>=Obs): 0.193
## Pr(X<=Obs): 0.807
```

```
plot(Okh_cug_results)
```



7. Network Regression

Given that network regression utilizes a quadratic assignment procedure (qap), reproduced values will not be exact but should be fairly similar to results provided in the table.

Epi-Jomon

```
EJ_local_observed_netreg<-netlm(EJ_local,EJ_observed,nullhyp = "qap")
EJ_recip_observed_netreg<-netlm(EJ_recip,EJ_observed,nullhyp = "qap")
EJ_central_observed_netreg<-netlm(EJ_central,EJ_observed,nullhyp = "qap")
```

Epi-Jomon	Estimate	Pr(<=b)	Pr(>=b)	Pr(>=|b|)
Observed ∼ Local				
intercept	0.071	0.34	0.46	0.46
x	0.053	0.79	0.26	0.72
Observed ∼ Reciprocal				
intercept	0.214	0.71	0.29	0.29
x	0.035	0.75	0.25	0.67
Observed ∼ Central				
intercept	0.571	0.91	0.09	0.09
x	0.178	0.72	0.28	0.37

Okhotsk

```
Okh_local_observed_netreg<-netlm(Okh_local,Okh_observed,nullhyp = "qap")
Okh_recip_observed_netreg<-netlm(Okh_recip,Okh_observed,nullhyp = "qap")
Okh_central_observed_netreg<-netlm(Okh_central,Okh_observed,nullhyp = "qap")
```

Epi-Jomon	Estimate	Pr(<=b)	Pr(>=b)	Pr(>=|b|)
Observed ∼ Local				
intercept	0.129	0.82	0.18	0.19
x	0.017	0.45	0.55	0.92
Observed ∼ Reciprocal				
intercept	0.322	0.90	0.10	0.10
x	0.018	0.47	0.53	1.0
Observed ∼ Central				
intercept	0.451	0.96	0.04	0.04
x	-0.012	0.53	0.47	0.98