R Code for Analysis of Small Scale Networks

Erik Gjesfjeld

June 5, 2015

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 2015. Please do not cite without permission from the author (erik.gjesfjeld@gmail.com).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 accomplished through the use of cluster analysis, prinicipal component analysis and Mahalanobis 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 2015 Social Network Analysis of Archaeological Data from Hunter-Gatherers: Methodological Problems and Potential Solutions. *Journal of Archaeological Method and Theory* 22(1): pp. 182-205. Available (with subscription) from Springer here.

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 libraries 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

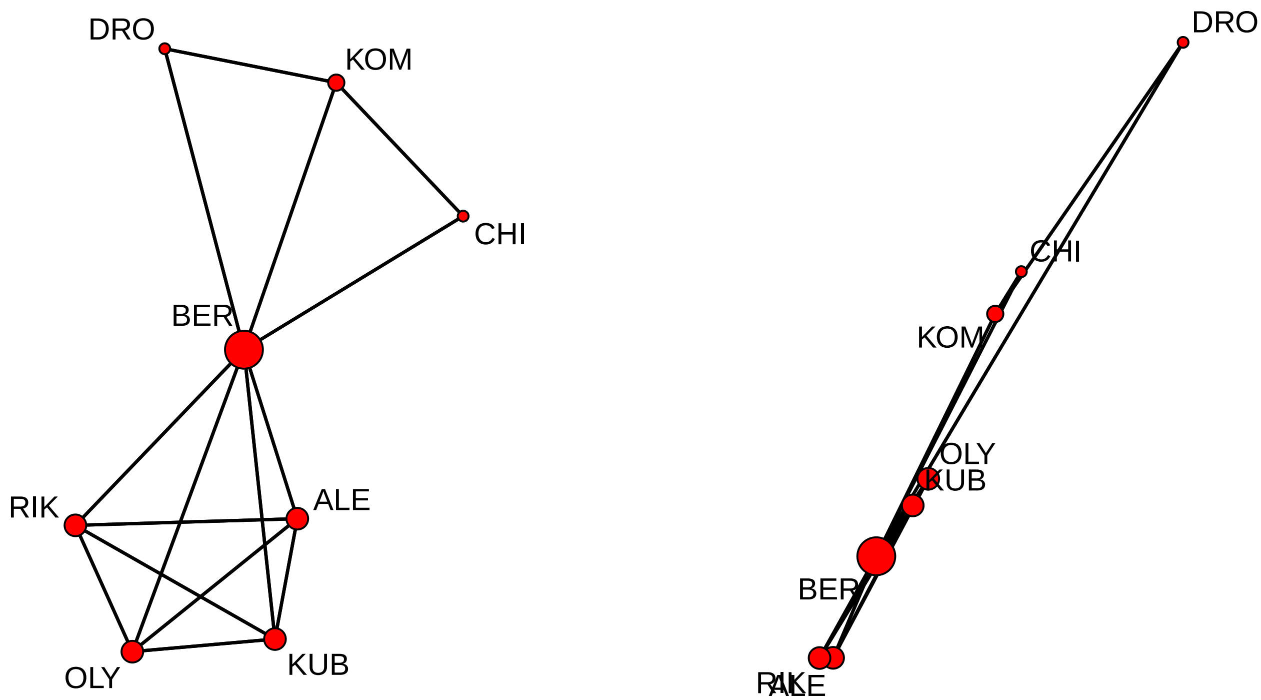
```
ej_net<-read.delim("EJ-net_JAMT.txt",header=T,row.names = 1) #Epi-Jomon
okh_net<-read.delim("Okh-net_JAMT.txt",header=T, row.names = 1) #Okhotsk

#Valued edge data
ej_net_val<-read.delim("EJ-net_JAMT_value.txt",header=T,row.names = 1) #Epi-Jomon
okh_net_val<-read.delim("Okh-net_JAMT_value.txt",header=T,row.names = 1) #Okhotsk

#With isolates removed
ej_net_iso_removed<-read.delim("EJ-net_JAMT_iso_removed.txt",header=T, row.names = 1)
okh_net_iso_removed<-read.delim("Okh-net_JAMT_iso_removed.txt",header=T, row.names = 1)

#Geographic Coordinates of Archaeological Sites
ej_coord<-read.delim("EJ-Coord2.txt",header=F,row.names=1)
okh_coord<-read.delim("Okh-Coord.txt",header=F,row.names=1)
```

3. Plotting the Network Graphs

a. Epi-Jomon Networks in Relational and Geographic Space

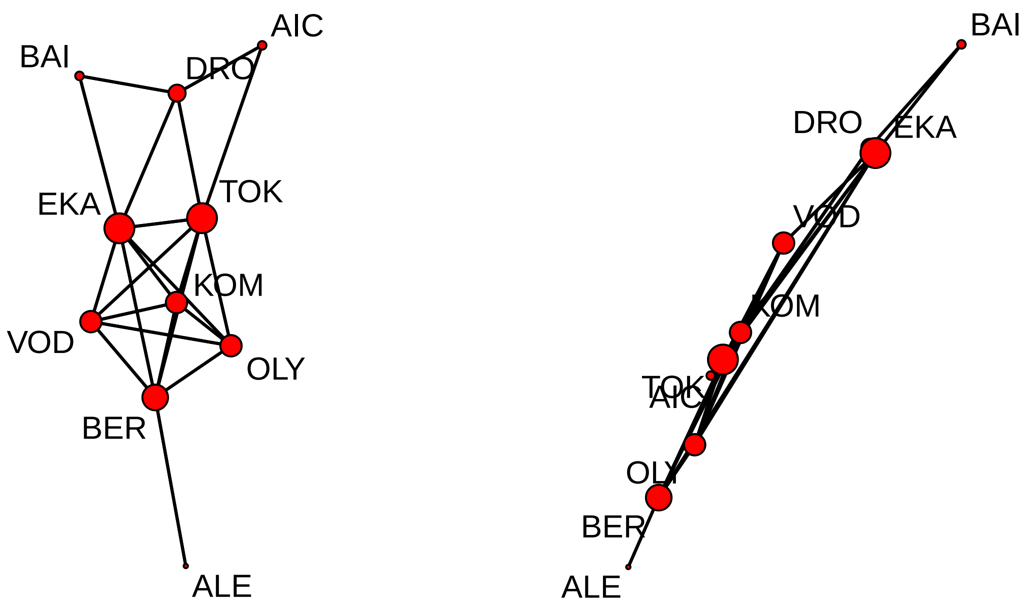
Node size is based on degree centrality and isolates are removed

```
par(mfrow=c(1,2)) #Plotting two across from each other
#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_net,displaylabels = TRUE,gmode="graph",vertex.cex=degree(ej_net)*0.1,
      displayisolates = F)
gplot(ej_net,displaylabels = TRUE,gmode="graph",vertex.cex=degree(ej_net)*0.1,
      displayisolates = F,coord=ej_coord)
```



b. Okhotsk Networks in Relational and Geographic Space

```
par(mfrow=c(1,2)) #Plotting two across from each other
#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_net,displaylabels = TRUE,gmode="graph",vertex.cex=degree(okh_net)*0.1,
      displayisolates = F)
gplot(okh_net,displaylabels = TRUE,gmode="graph",vertex.cex=degree(okh_net)*0.1,
      displayisolates = F,coord=okh_coord)
```



4. Graph Correlation

```
EJ_OKH_cor<-gcor(ej_net,okh_net,mode="graph")
```

Graph Correlation: 0.019 *Note: This value is different from the one published in Gjesfjeld 2015. I am currently unsure why these values differ but the general interpretation of no correlation between the two graphs remains the same.

5. Centrality Scores

a. Epi-Jomon (EJ) and Okhotsk (Okh)

```
EJ_deg<-round(degree(ej_net,gmode="digraph"),digits = 3) #Degree centrality
EJ_between<-round(betweenness(ej_net,gmode="digraph"),digits = 3) #Betweenness
EJ_evcent<-round(evcent(ej_net,gmode="digraph"), digits = 3) #Eigenvector
Okh_deg<-round(degree(okh_net,gmode="digraph"),digits = 3) #Degree centrality
Okh_between<-round(betweenness(okh_net,gmode="digraph"),digits = 3) #Betweenness
Okh_evcent<-round(evcent(okh_net,gmode="digraph"), digits = 3) #Eigenvector
```

Site Name	Site Abbreviation	Centrality Measures		
		Degree	Eigenvector	Betweenness
Epi-Jomon				
Berezovka	BER	14	0.50	25
Alekinha	ALE	8	0.40	0
Kuybshevskaya	KUB	8	0.40	0
Olya	OLY	8	0.40	0
Rikorda	RIK	8	0.40	0
Kompaniskii	KOM	6	0.19	1
Chirpoi	CHI	4	0.16	0
Drobnye	DRO	0	0.00	0
Ainu Bay	AIB	0	0.00	0
Ainu Creek	AIC	0	0.00	0
Rasshua	RAS	0	0.00	0
Sernovodskoe	SER	0	0.00	0
Zapadnaya	ZAP	0	0.00	0
Okhotsk				
Ekarma	EKA	14	0.42	16
Tokotan	TOK	14	0.42	16
Berezovka	BER	12	0.38	16
Kompaniskii	KOM	10	0.37	0
Olya	OLY	10	0.37	0
Vodopodnaya	VOD	10	0.37	0
Drobnye	DRO	8	0.20	4
Ainu Creek	AIN	4	0.12	0
Baikova	BAI	4	0.12	0
Alekinha	ALE	2	0.07	0
Ainu Bay	AIB	0	0.00	0
Bolshoy	BOL	0	0.00	0
Chrinkotan	CHR	0	0.00	0
Lake Lazournye	LAL	0	0.00	0
Rasshua	RAS	0	0.00	0
Ryponkicha	RYP	0	0.00	0
Zapadnaya	ZAP	0	0.00	0

6. Bootstrap Correlation of Centrality Measurements

a. Adjust Epi-Jomon and Okhotks Networks to a Matrix class

```
EJ.mat<-as.matrix(ej_net)
Okh.mat<-as.matrix(okh_net)
```

b. Set up the bootstrap functions

Just the functions for Epi-Jomon networks are provided. In order to run functions for Okhotsk networks (or your own networks) simply replace the EJ with Okh (or your own label)

Degree

```
EJ.deg.NetBoot<-function(x){
  EJ.deg.cor=c()
  for (i in 1:1000) {
    EJ.sampled.nodes <- sort(sample(1:nrow(EJ.mat),size=x,replace=F))
    EJ.removed.nodes <- setdiff(1:nrow(EJ.mat),EJ.sampled.nodes)
    EJ.samp<-EJ.mat[EJ.sampled.nodes,EJ.sampled.nodes]
    EJ.deg.samp<-degree(EJ.samp)
    EJ.orig.deg.samp<-degree(EJ.mat)
    EJ.rev.deg.samp<-EJ.orig.deg.samp[c(-EJ.removed.nodes)]
    EJ.deg.cor[i]=cor(EJ.deg.samp,EJ.rev.deg.samp)
  }
  print(mean(EJ.deg.cor,na.rm=T))
  print(sd(EJ.deg.cor,na.rm=T))
}
```

Betweenness

```
EJ.betweenness.NetBoot<-function(x){
  EJ.betweenness.cor=c()
  for (i in 1:1000) {
    EJ.sampled.nodes <- sort(sample(1:nrow(EJ.mat),size=x,replace=F))
    EJ.removed.nodes <- setdiff(1:nrow(EJ.mat),EJ.sampled.nodes)
    EJ.samp<-EJ.mat[EJ.sampled.nodes,EJ.sampled.nodes]
    EJ.betweenness.samp<-betweenness(EJ.samp)
    EJ.orig.betweenness.samp<-betweenness(EJ.mat)
    EJ.rev.betweenness.samp<-EJ.orig.betweenness.samp[c(-EJ.removed.nodes)]
    EJ.betweenness.cor[i]=cor(EJ.betweenness.samp,EJ.rev.betweenness.samp)
  }
  print(mean(EJ.betweenness.cor,na.rm=T))
  print(sd(EJ.betweenness.cor,na.rm=T))
}
```

Eigenvector Centrality

```
EJ.evcent.NetBoot<-function(x){
  EJ.evcent.cor=c()
  for (i in 1:1000) {
    EJ.sampled.nodes <- sort(sample(1:nrow(EJ.mat),size=x,replace=F))
    EJ.removed.nodes <- setdiff(1:nrow(EJ.mat),EJ.sampled.nodes)
    EJ.samp<-EJ.mat[EJ.sampled.nodes,EJ.sampled.nodes]
```

```
    EJ.evcent.samp<-evcent(EJ.samp)
    EJ.orig.evcent.samp<-evcent(EJ.mat)
    EJ.rev.evcent.samp<-EJ.orig.evcent.samp[c(-EJ.removed.nodes)]
    EJ.evcent.cor[i]=cor(EJ.evcent.samp,EJ.rev.evcent.samp)
  }
  print(mean(EJ.evcent.cor,na.rm=T))
  print(sd(EJ.evcent.cor,na.rm=T))
}
```

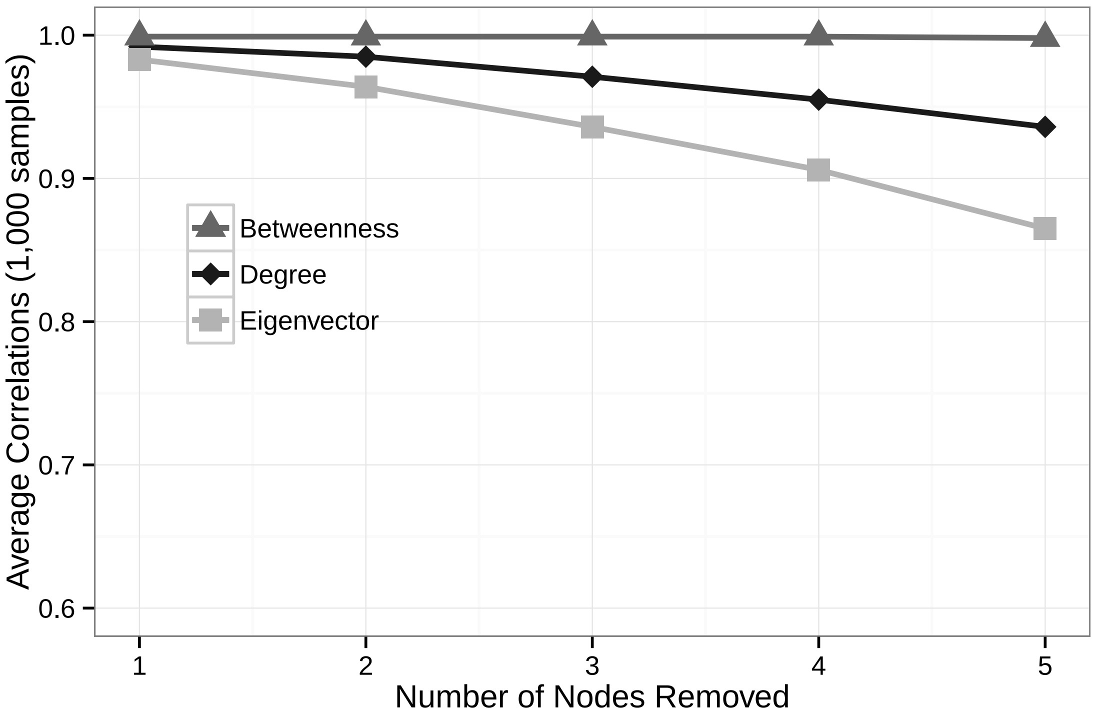
Run the functions

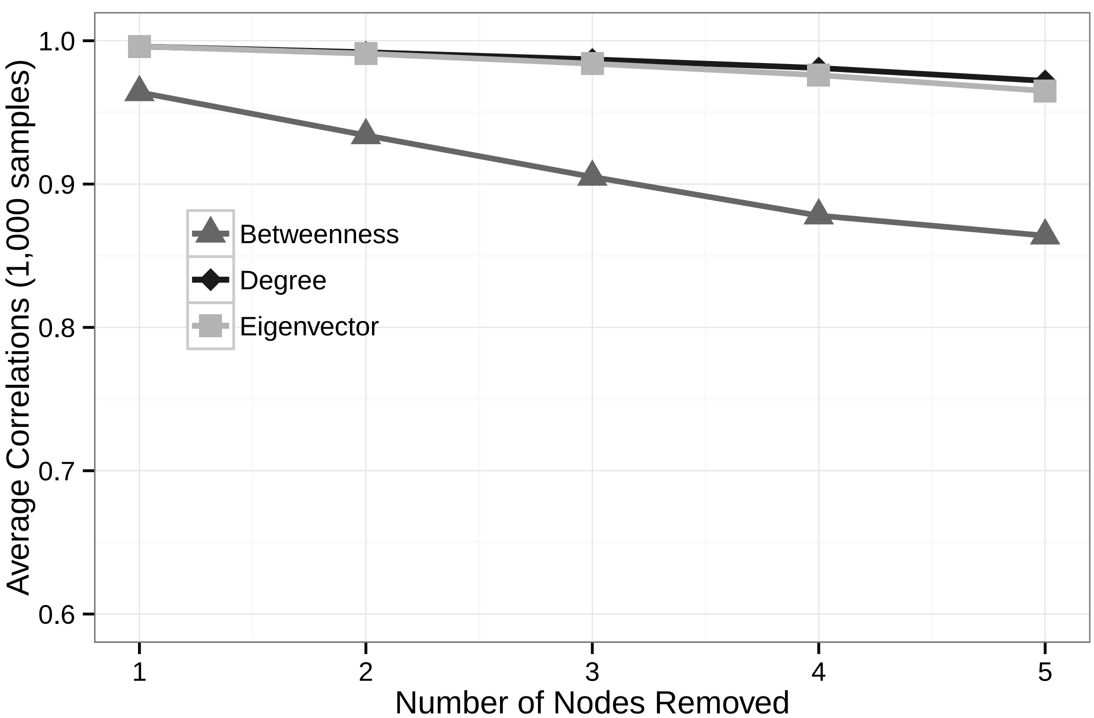
Use differing numbers of nodes progressively removing 1 node from the total network size. Total network size for Epi-Jomon is 13 and Okhotsk is 17. Function may fail if too many nodes are unable to be recovered.

```
#Degree - Epi-Jomon
EJ.deg.NetBoot(12) #1 node removed
EJ.deg.NetBoot(11) #2 nodes removed
EJ.deg.NetBoot(10) #3 nodes removed
EJ.deg.NetBoot(9) #4 nodes removed
EJ.deg.NetBoot(8) #5 nodes removed
#Betweenness - Epi-Jomon
EJ.betweenness.NetBoot(12) #1 node removed
EJ.betweenness.NetBoot(11) #2 nodes removed
EJ.betweenness.NetBoot(10) #3 nodes removed
EJ.betweenness.NetBoot(9) #4 nodes removed
EJ.betweenness.NetBoot(8) #5 nodes removed
#Eigenvector - Epi-Jomon
EJ.evcent.NetBoot(12) #1 node removed
EJ.evcent.NetBoot(11) #2 nodes removed
EJ.evcent.NetBoot(10) #3 nodes removed
EJ.evcent.NetBoot(9) #4 nodes removed
EJ.evcent.NetBoot(8) #5 nodes removed
```

Given the random sampling procedure used in the function, results will differ slightly on reproducution, and may differ significantly with different networks

Number of Removed Nodes	Degree		Eigenvector		Betweenness	
	Mean	SD	Mean	SD	Mean	SD
Epi-Jomon						
1	0.992	0.066	0.983	0.022	0.999	0
2	0.985	0.015	0.964	0.044	0.999	0
3	0.971	0.030	0.936	0.124	0.999	0
4	0.955	0.050	0.906	0.172	0.999	0.011
5	0.936	0.086	0.865	0.234	0.998	0.024
Okhotsk						
1	0.996	0.004	0.996	0.005	0.964	0.010
2	0.992	0.007	0.991	0.011	0.934	0.082
3	0.987	0.011	0.984	0.022	0.905	0.107
4	0.981	0.016	0.976	0.040	0.878	0.131
5	0.972	0.027	0.965	0.051	0.864	0.141

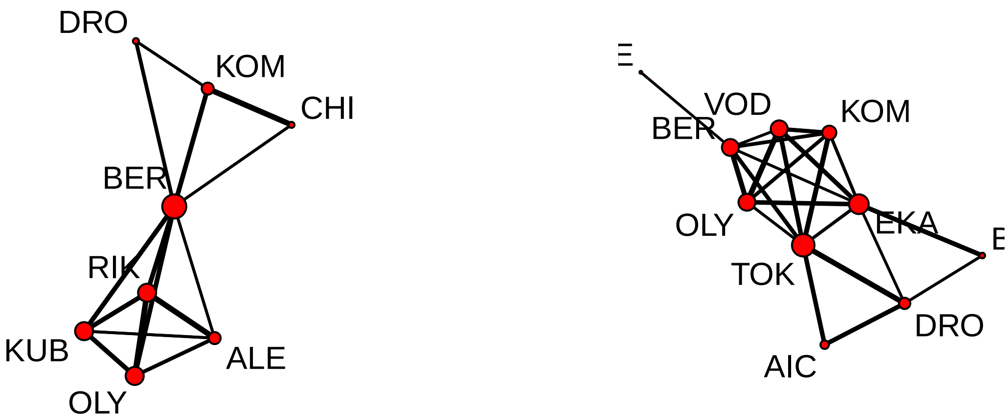

Average Correlations (1,000 samples)
1.0
0.9
0.8
0.7
0.6
Betweenness
Degree
Eigenvector
1
2
3
4
5
Number of Nodes Removed


Average Correlations (1,000 samples)
1.0
0.9
0.8
0.7
0.6
Betweenness
Degree
Eigenvector
1
2
3
4
5
Number of Nodes Removed

7.Sensitivity Analysis

a. Network graphs with valued ties

```
par(mfrow=c(1,2))
gplot(ej_net_val,gmode="graph",edge.lwd  = degree(ej_net_val)*0.3,
      displaylabels = T,vertex.cex=degree(ej_net_val)*0.1, displayisolates = F)

gplot(okh_net_val,gmode="graph",edge.lwd  = degree(okh_net_val)*0.3,
      displaylabels = T, vertex.cex=degree(okh_net_val)*0.1, displayisolates = F)
```



b. Baseline graph-level indices

Epi-Jomon

```
ej_gden_val<-gden(ej_net_val)
ej_cent_deg_val<-centralization(ej_net_val,degree)
ej_cent_evcent_val<-centralization(ej_net_val,evcent)
ej_cent_between_val<-centralization(ej_net_val,betweenness)
```

Okhotsk

```
okh_gden_val<-gden(okh_net_val)
okh_cent_deg_val<-centralization(okh_net_val,degree)
okh_cent_evcent_val<-centralization(okh_net_val,evcent)
okh_cent_between_val<-centralization(okh_net_val,betweenness)
```

	Graph	Centralization		
	Density	Degree	Eigenvector	Betweenness
Epi-Jomon	0.243	0.500	0.295	0.189
Okhotsk	0.176	0.367	0.276	0.057

c. Assessing difference in graph-level indices with removal of individual noded

Note: I am only showing an example for graph density of the Epi-Jomon network. Centralization values can be examined be replacing gden(EJ.mat.val) with centralization(EJ.mat.val,degree/evcent/betweenness)

```
EJ.mat.val<-ej_net_val
EJ.gden.sens={
  EJ.nodes=c()
  EJ.sampled.gden=c()
  EJ.gden.absdiff=c()
for (i in 1:nrow(EJ.mat.val)) {
  EJ.nodes <- (1:nrow(EJ.mat.val))
  EJ.sampled.nodes<-EJ.nodes[-i]
  EJ.samp.val<-EJ.mat.val[EJ.sampled.nodes,EJ.sampled.nodes]
  EJ.gden.samp.val<-gden(EJ.samp.val)
  EJ.gden.mat.val<-gden(EJ.mat.val)
  EJ.nodes[i]=i
  EJ.sampled.gden[i]=EJ.gden.samp.val
  EJ.gden.absdiff[i]=abs(EJ.gden.samp.val-EJ.gden.mat.val)
}
EJ.sd<-c(sd(EJ.sampled.gden),sd(EJ.sampled.gden)*2)
EJ.gden.list<-data.frame(Nodes=row.names(EJ.mat.val),
                         Graph_Density_with_Node=round(gden(EJ.mat.val),digits=3),
                         Graph_Density_without_Node=round(EJ.sampled.gden,digits=3),
                         Difference_in_Graph_Density=round(EJ.gden.absdiff,digits=3))
}
EJ.gden.list
```

Nodes	Graph Density with Node	Graph Density without Node	Difference in Graph Density
AIB	0.244	0.288	0.044
AIC	0.244	0.288	0.044
ALE	0.244	0.227	0.016
BER	0.244	0.167	0.077
CHI	0.244	0.258	0.014
DRO	0.244	0.258	0.014
KOM	0.244	0.227	0.016
KUB	0.244	0.197	0.047
OLY	0.244	0.197	0.047
RAS	0.244	0.288	0.044
RIK	0.244	0.197	0.047
SER	0.244	0.288	0.044
ZAP	0.244	0.288	0.044

Graph Density	1 SD	2 SD
Epi-Jomon	0.044	0.088

c. Results of Sensitivity Analysis

Table showing results of senstivity analysis. High is more than a 2 standard deviation difference from the mean of all without node graph density / centralization measures. Medium is greater than 1 SD but less than 2 and low is less than 1 SD difference.

Site Name	Site Abbreviation	Degree	Graph	Centralization Measures		
		(Valued)	Density	Degree	Eigenvector	Betweenness
Epi-Jomon						
Berezovka	BER	16	Medium	High	High	High
Kuybshevskaya	KUB	12	Medium	Low	Medium	Low
Olya	OLY	12	Medium	Low	Medium	Low
Rikorda	RIK	12	Medium	Low	Medium	Low
Alekinha	ALE	8	Low	Low	Low	Low
Kompaniskii	KOM	8	Low	Medium	Low	Low
Chirpoi	CHI	4	Low	Low	Low	Low
Drobnye	DRO	4	Low	Low	Low	Low
Ainu Bay	AIB	0	Medium	Low	Low	Low
Ainu Creek	AIC	0	Medium	Low	Low	Low
Rasshua	RAS	0	Medium	Low	Low	Low
Sernovodskoe	SER	0	Medium	Low	Low	Low
Zapadnaya	ZAP	0	Medium	Low	Low	Low
Okhotsk						
Tokotan	TOK	16	Medium	High	Medium	High
Ekarma	EKA	14	Medium	Low	Medium	High
Berezovka	BER	12	Medium	Low	Medium	Low
Olya	OLY	12	Medium	Low	High	Low
Vodopodnaya	VOD	12	Medium	Low	High	Low
Kompaniskii	KOM	10	Low	Low	Medium	Low
Drobnye	DRO	8	Low	Medium	Low	Low
Ainu Creek	AIN	6	Low	Medium	Low	Low
Baikova	BAI	4	Low	Medium	Low	Low
Alekinha	ALE	2	Low	Medium	Low	Low
Ainu Bay	AIB	0	Low	Low	Low	Low
Bolshoy	BOL	0	Low	Low	Low	Low
Chrinkotan	CHR	0	Low	Low	Low	Low
Lake Lazournye	LAL	0	Low	Low	Low	Low
Rasshua	RAS	0	Low	Low	Low	Low
Ryponkicha	RYP	0	Low	Low	Low	Low
Zapadnaya	ZAP	0	Low	Low	Low	Low