INFX 576: Problem Set 2 - Graph Level Indices and CUG Tests*

Suchitra Sundararaman

Due: Thursday, January 26, 2017

Collaborators:

Instructions: Before beginning this assignment, please ensure you have access to R and RStudio.

1. Download the `problemset2.Rmd` file from Canvas. You will also need the `problemset2_data.Rdata` file which contains the three different network datasets needed for this assignment.
2. Replace the "Insert Your Name Here" text in the `author:` field with your own full name. Any collaborators must be listed on the top of your assignment.
3. Be sure to include well-documented (e.g. commented) code chucks, figures and clearly written text chunk explanations as necessary. Any figures should be clearly labeled and appropriately referenced within the text.
4. Collaboration on problem sets is acceptable, and even encouraged, but each student must turn in an individual write-up in his or her own words and his or her own work. The names of all collaborators must be listed on each assignment. Do not copy-and-paste from other students' responses or code.
5. When you have completed the assignment and have **checked** that your code both runs in the Console and knits correctly when you click `Knit PDF`, rename the R Markdown file to `YourLastName_YourFirstName_ps2.Rmd`, knit a PDF and submit the PDF file on Canvas.

Setup: In this problem set you will need, at minimum, the following R packages.

```
# Load standard libraries
library(statnet)

# Load data
load("problemset2_data.Rdata")
ls() # Print objects in workspace to see what is available
```

```
## [1] "kaptail.ins" "mids_1993"   "sampson"
```

Problem 1: Graph-Level Indices Consider the Sampson monk data[1]. Sampson collected various relationships between several monks at a monastery. Suppose we divide the types of social ties into positive and negative relationship types as follows:

- Positive Relationships: Esteem, Influence, LikeT1, LikeT2, LikeT3, and Praise
- Negative Relationships: Disesteem, NegInfluence, Dislike, and Blame

Using a vector permutation test, evaluate the questions below.

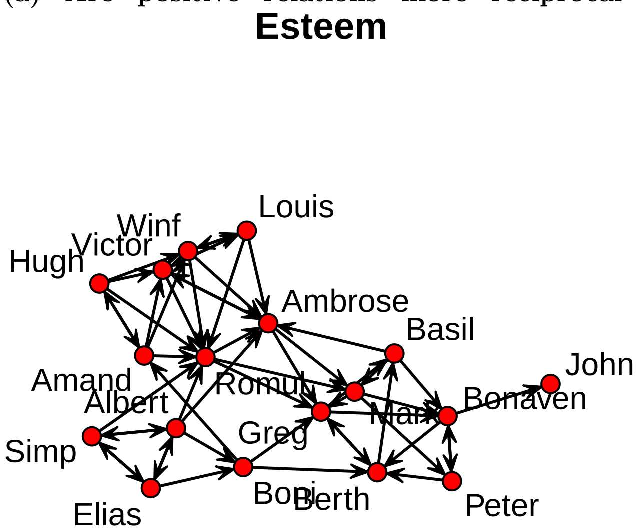
*Problems originally written by C.T. Butts (2009)

[1] F. S. Sampson. A novitiate in a period of change: An experimental and case study of social relationships. PhD thesis, Cornell University. 1968.

```
recip <- log(grecip(sampson,measure="edgewise")/gden(sampson))

#Plotting the first network.
par(mfrow=c(1,1))
gplot(sampson[1], displaylabels = TRUE, main= names(sampson[1]))
```

(a) Are positive relations more reciprocal (relative to density) than negative ones?



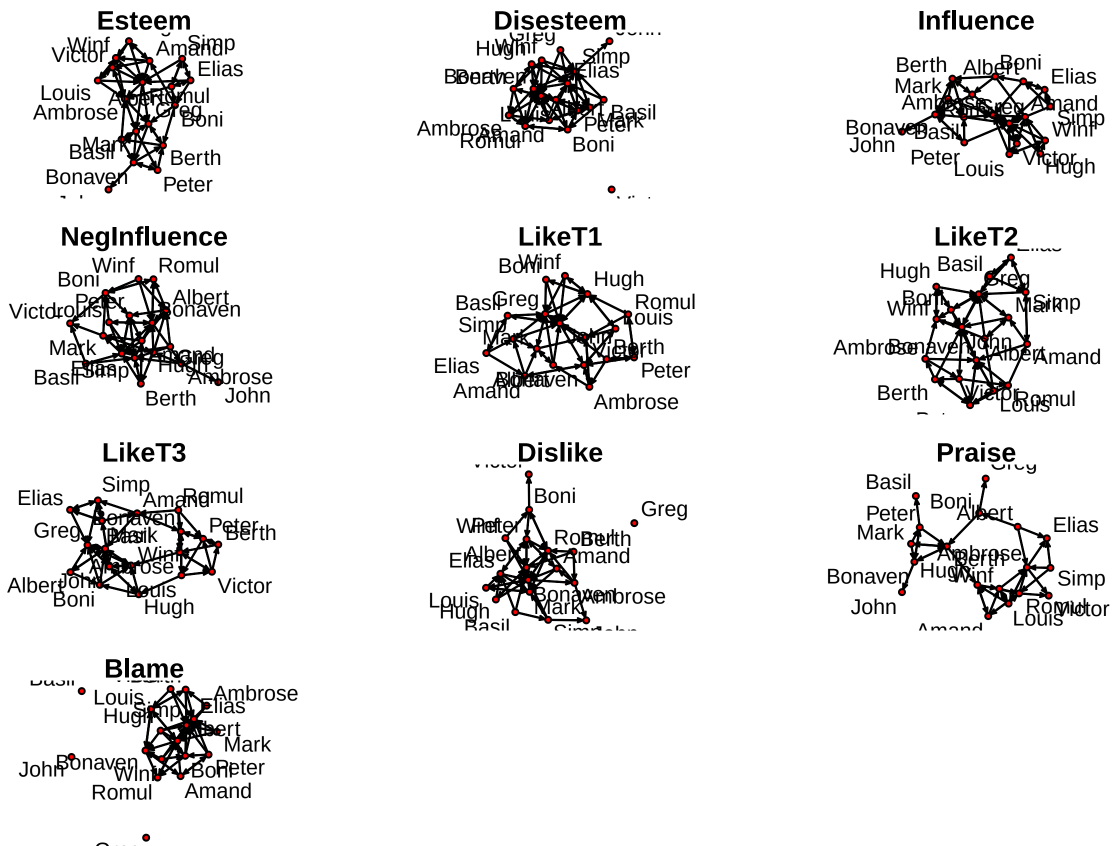

```
par(mar=c(1,1,1,1))
par(mfrow=c(4,3))
for(i in 1:length(sampson))
  gplot(sampson[i],displaylabels = TRUE, main= names(sampson[i]))

positive <- c(TRUE, FALSE, TRUE, FALSE, TRUE, TRUE, TRUE, FALSE, TRUE,FALSE)
obs <- sum(recip[positive])- sum(recip[!positive])


perm.cor.test<-function(x,niter=5000){  #Define a simple test function
  rep <- vector()
  for(i in 1:niter)
  {
    pos_sample <- sample(positive)
    rep[i] <- sum(recip[pos_sample])- sum(recip[!pos_sample])
  }
  cat("Vector Permutation Test:\n\tObserved relationship: ",obs,"\tReplicate quantiles (niter=",niter,")
  cat("\t\tPr(rho>=obs):",mean(rep>=obs),"\n")
  cat("\t\tPr(|rho|>=|obs|):",mean(abs(rep)>=abs(obs)),"\n")
  }

perm.cor.test(recip)
```

```
## Vector Permutation Test:
##  Observed relationship: 3.059769 Replicate quantiles (niter=5000)
##      Pr(rho>=obs): 0.0346
##      Pr(|rho|>=|obs|): 0.0346
```




```
trans <-log(gtrans(sampson)/gden(sampson))
obs <- sum(trans[positive])- sum(trans[!positive])
perm.cor.test<-function(x,niter=5000){  #Define a simple test function
  rep <- vector()
  for(i in 1:niter)
  {
    pos_sample <- sample(positive)
    rep[i] <- sum(trans[pos_sample])- sum(trans[!pos_sample])
  }
  cat("Vector Permutation Test:\n\n\tObserved relationship: ",obs,"\tReplicate quantiles (niter=",niter,"
  cat("\t\tPr(rho>=obs):",mean(rep>=obs),"\n")
  cat("\t\tPr(|rho|>=|obs|):",mean(abs(rep)>=abs(obs)),"\n")
  }

perm.cor.test(trans)
```

(b) Are positive relations more transitive (relative to density) than negative ones?

```
## Vector Permutation Test:
##  Observed relationship: 3.421488 Replicate quantiles (niter=5000)
##      Pr(rho>=obs): 0.0304
##      Pr(|rho|>=|obs|): 0.0304
```

(c) Discuss the findings from part (a) and part (b). A: Vector permutation test was performed to analyze whether positive relations are more reciprocal than the negative ones. So the null hypothesis considered here is: Positive relations are not more reciprocal than the negative ones. The permutation test gave a p-value of 0.0296 and this implies that there is only a 2.9% chance of positive relations being equal to or less reciprocal than the negative ones(relative to density). This value is less than the significance level considered i.e 0.05. Thus, as the p-value is lesser than the significance level, we can reject the null hpothesis. Thus, we can say that positive relations are more reciprocal than the negative ones.

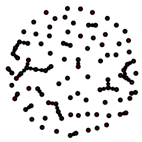
B:Vector permutation test was performed to analyze whether positive relations are more transitive than the negative ones. So the null hypothesis considered here is: Positive relations are not more transitive than the negative ones. The permutation test gave a p-value of 0.0304 and this implies that there is only a 3% chance of positive relations being equal to or less transitive than the negative ones(relative to density). This value is less than the significance level considered i.e 0.05. Thus, as the p-value is lesser than the significance level, we can reject the null hpothesis. Thus,we can say that positive relations are more transitive than the negative ones.

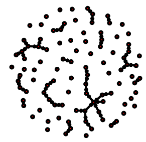
Problem 2: Random Graphs

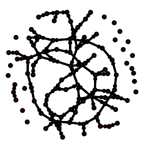
(a) Generating Random Graphs Generate 100-node random directed graphs with expected densities of 0.0025, 0.005, 0.01, 0.015, 0.02, and 0.025, with at least 500 graphs per sample. Remember the `rgraph` function can draw more than one graph at a time. Plot the average Krackhardt connectedness, dyadic reciprocity, and edgewise reciprocity as a function of expected density. Use these to describe the baseline effect of increasing density on network structure.

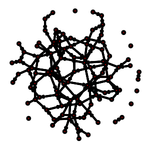
```r
par(mfrow = c(2,2))
den <- c(0.0025, 0.005, 0.01, 0.015, 0.02,0.025)
krack_con <- c(0,0,0,0,0,0)
recip_dyadic <- c(0,0,0,0,0,0)
recip_edge <- c(0,0,0,0,0,0)

for(i in 1:length(den))
{
gplot(rgraph(100,500, tprob = den[i]))
krack_con[i] <- sum(connectedness(rgraph(100, 500, tprob = den[i])))
recip_dyadic[i] <- sum(grecip(rgraph(100,500, tprob = den[i]), measure = "dyadic"))
recip_edge[i] <- sum(grecip(rgraph(100,500, tprob = den[i]), measure = "edgewise"))
}
```









```
#Average Krackhardt Connectedness
krack_con_avg <- krack_con/500.0
krack_con_avg
```

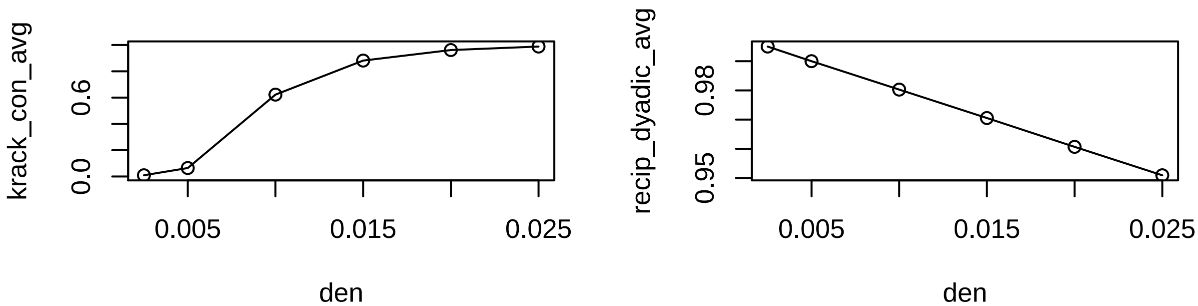
```
## [1] 0.009375354 0.064349899 0.622584646 0.881563636 0.961649697 0.988200808
```

```
 plot(den,krack_con_avg)
lines(den,krack_con_avg)

 #Average dyadic reciprosity
recip_dyadic_avg <- recip_dyadic/500.0
recip_dyadic_avg
```

```
## [1] 0.9950234 0.9900105 0.9802772 0.9705337 0.9606606 0.9509265
```

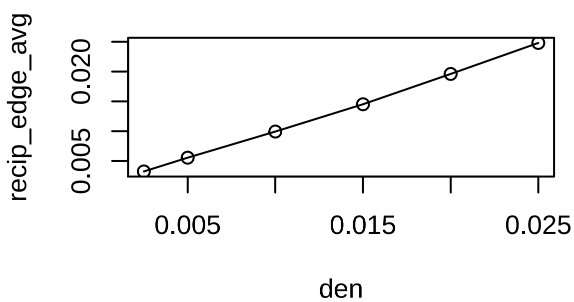
```
plot(den,recip_dyadic_avg)
lines(den,recip_dyadic_avg)
```




```r
#Average edgewise reciprosity
recip_edge_avg <- recip_edge/500.0
recip_edge_avg
```

```
## [1] 0.003244441 0.005551224 0.009935458 0.014515854 0.019601474 0.024803001
```

```r
plot(den,recip_edge_avg)
lines(den,recip_edge_avg)
```



From the plots for average Krackhardt connectedness, average dyadic reciprosity and average edgewise reciprosity, it can be seen that the increasing density has an effect on all these statistics.

As the density increases, krackhardt connectednes increases, which implies that the fraction of weakly connected vertex pairs increases in the network.

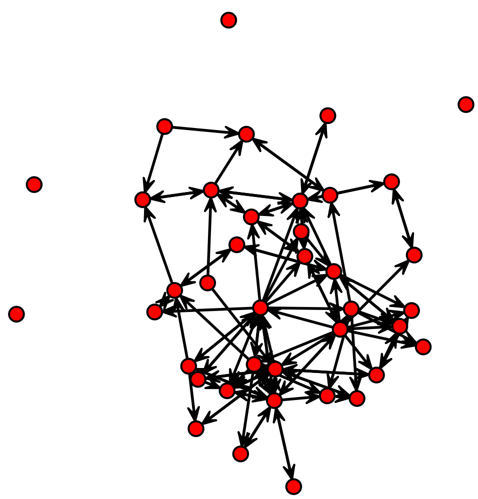
As the density increases, the dyadic reciprosity decreases. This implies that as the fraction of possible edges increases in the network, it increases the probability of a dyad being assymetric.

As the density increases, edgewise reciprosity increases. This implies that as the density increases the probability of a node being reciprocated increases. This also indicates that the probability of a dyad being mutual as compared to null, in symmetric dyad increases.

(b) Comparing GLIs In this problem we will use the well-known social network dataset, collected by Bruce Kapferer in Zambia from June 1965 to August 1965, involves interactions among workers in a tailor shop as observed by Kapferer himself.[2] Here, an interaction is defined by Kapferer as "continuous uninterrupted social activity involving the participation of at least two persons"; only transactions that were relatively frequent are recorded.

Generate 500 random directed graphs whose dyad census is the same as that of `kaptail.ins`. Plot histograms for total degree centralization, betweenness centralization, transitivity, and Krackhardt connectedness from this random sample. On your plot mark the observed values of these statistics (from the `kaptail.ins` data) using a vertial line You mgith find the `abline` function helpful here. Try modifying the `lwd` argument to the plot function to make the vertical line stand out. How do the replicated graphs compare to the observed data.

```
gplot(kaptail.ins)
```



```
par(mfrow=c(2,2))
#Observed statistics of Kaptail.ins
cent_obs <- centralization(kaptail.ins,degree)
bet_obs <- centralization(kaptail.ins, betweenness)
trans_obs <- gtrans(kaptail.ins)
connectedness_obs <- connectedness(kaptail.ins)

dyad.census(kaptail.ins)
```

```
##      Mut Asym Null
## [1,]  33   43  665
```

```
x <- rguman(500, 39, mut = dyad.census(kaptail.ins)[1,1] , asym = dyad.census(kaptail.ins)[1,2], null =

#Histogram for total Degree Centralization.
cent <- c()
for(i in 1:network.size(kaptail.ins))
{
cent[i]<- centralization(x,g=i,degree)
}
cent
```

[2] Kapferer B. (1972). Strategy and transaction in an African factory. Manchester: Manchester University Press.

```
##  [1] 0.06116643 0.10277383 0.06116643 0.07503556 0.06116643 0.08890469
##  [7] 0.06116643 0.10277383 0.11664296 0.11664296 0.10277383 0.14438122
## [13] 0.11664296 0.10277383 0.11664296 0.10277383 0.08890469 0.08890469
## [19] 0.11664296 0.14438122 0.06116643 0.04729730 0.07503556 0.11664296
## [25] 0.10277383 0.07503556 0.08890469 0.10277383 0.04729730 0.08890469
## [31] 0.08890469 0.08890469 0.08890469 0.13051209 0.10277383 0.10277383
## [37] 0.08890469 0.06116643 0.08890469
```

```r
hist(cent, main = "Degree Centralization", xlim = c(0,0.2))
abline(v=cent_obs, lwd=3)

#Histogram for total Betweenness Centralization.
bet <- c()
for(i in 1:network.size(kaptail.ins))
{
bet[i]<- centralization(x,g=i,betweenness)
}
bet
```

```
##  [1] 0.13938816 0.14184741 0.10864098 0.13312322 0.10757896 0.15269461
##  [7] 0.16160291 0.14621039 0.18772451 0.14257753 0.20427113 0.23114071
## [13] 0.14795244 0.13010005 0.15793083 0.11805483 0.16431967 0.10561459
## [19] 0.16503091 0.30146766 0.08251626 0.13139612 0.11436260 0.19471839
## [25] 0.17750163 0.13883798 0.17405908 0.19115026 0.08629270 0.09921590
## [31] 0.11611889 0.14880533 0.19147170 0.15571283 0.16716796 0.16534860
## [37] 0.14402404 0.15152301 0.16289334
```

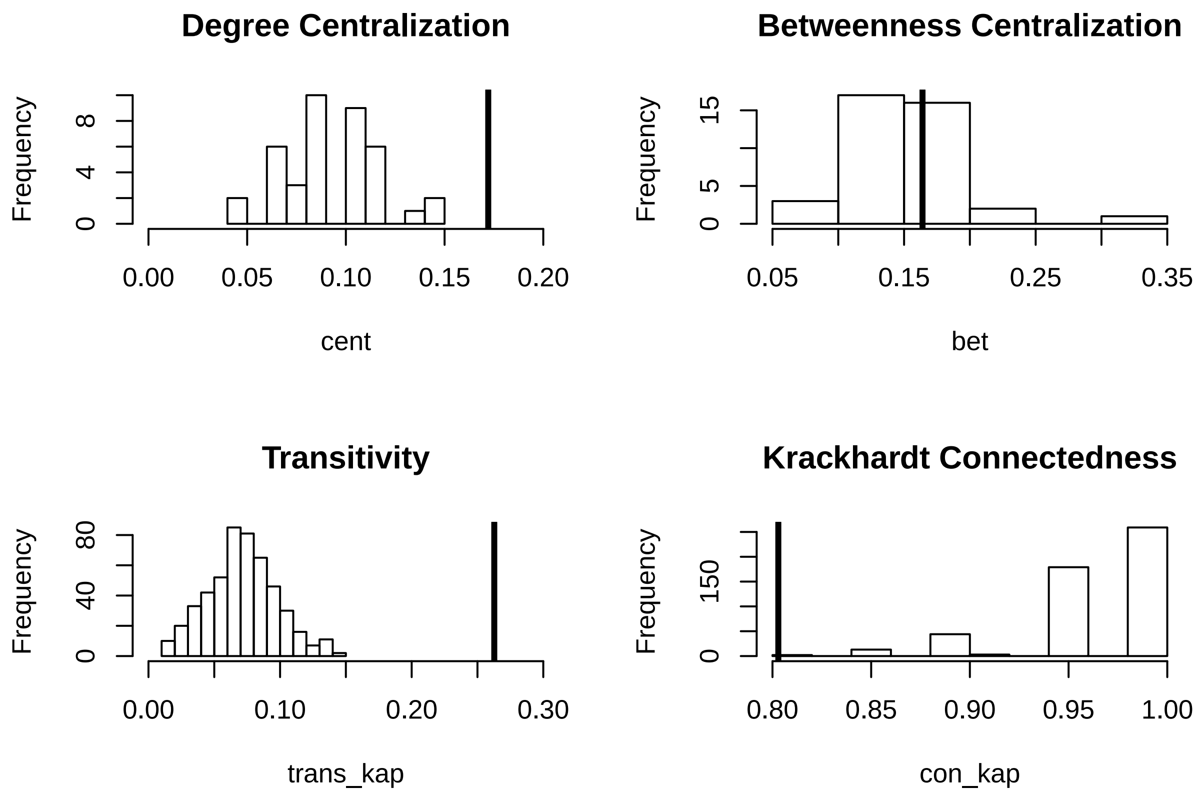
```r
summary(bet)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.08252 0.13070 0.14880 0.15230 0.16630 0.30150
```

```r
hist(bet, main = "Betweenness Centralization")
abline(v= bet_obs, lwd=3)

# Histogram for Transitivity
trans_kap <- gtrans(x)
hist(trans_kap, main= "Transitivity", xlim = c(0,0.3))
abline(v= trans_obs, lwd=3)

#Histogram for Krackhardt Connectedness
con_kap <- connectedness(x)
hist(con_kap, main="Krackhardt Connectedness", xlim = c(0.8,1.00))
abline(v= connectedness_obs, lwd=3)
```



Degree Centralization: From the plot, it can be observed that the degree centralization of the observed data is higher than any of the random graph which is generated. As we are conditioning the random graphs on the dyad census of the Kaptail.ins data, the number of mutual, null and assymetric dyads remain the same for the generated graphs. This may indicate that the observed data has centrality concentrated on fewer vertices, whereas the centrality meaures are more distributed in the generated random graphs.

Betweenness Centralization: The median of the random graph distribution(0.1465) is slightly lesser than the observed betwenness centralization of Kaptail.ins data(0.164). This may indicate that the vertices in the observed data are more closely connected as compared to the random graphs.

Transitivity: From the plot, it can be observed that the transitivity of the observed data is much higher than the transitivity of the generated random graphs. Transitivity might indicate the presence of stronger ties in the observed data as compared to the generated graphs.

Krackhardt Connectedness:The Krackhardt connectedness of the observed data is much smaller than that of the generated graphs.This may imply that the fraction of weakly connected vertices in the observed data is lesser as compared to the random graphs.

Problem 3: Testing Structural Hypotheses Consider the following set of propositions, which may or may not be true of given dataset. For each, do the following:

1. Identify a statistic (e.g. GLI) whose value should deviate from a random baseline if the proposition is true.
2. Identify the approporate baseline distribution to which the statistic should be compared.
3. Determine whether the proposition implies that the statistic should be greater or lower than its baseline distribution would indicate.
4. Conduct a conditional uniform graph test based on your conclusions in 1-3. In reporting your results, include appropriate summary output from the `cug.test` function as well as the resulting distributional plots. Based on the results, indicate whether the data appears to support or undermine the proposition in question. Be sure to justify your conclusion.

```r
# Statistic : Edgewise Reciprosity
# Baseline Distribution : size and edges.
# Higher
summary(mids_1993)
```

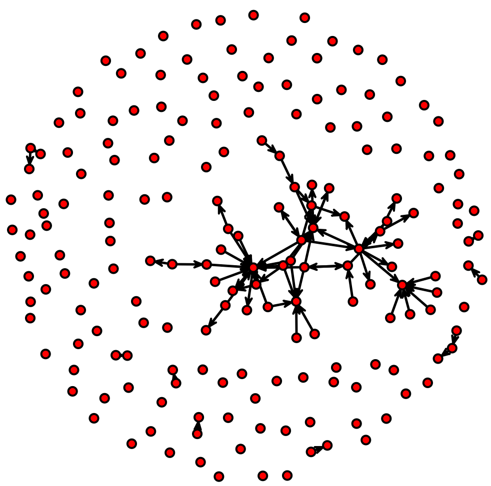
(a) In militarized interstate disputes, hostile acts are disproportionately likely to be responded to in kind.

```
## Network attributes:
##   vertices = 186
##   directed = TRUE
##   hyper = FALSE
##   loops = FALSE
##   multiple = FALSE
##   bipartite = FALSE
##  total edges = 70
##    missing edges = 0
##    non-missing edges = 70
##  density = 0.002034292
##
## Vertex attributes:
##
##  State.Abb:
##    character valued attribute
##    attribute summary:
##    the 10 most common values are:
## AAB AFG ALB ALG AND ANG ARG ARM AUL AUS
##   1   1   1   1   1   1   1   1   1   1
##
##  State.Num:
##    character valued attribute
##    attribute summary:
##    the 10 most common values are:
## 100 101 110 115 130 135 140 145 150 155
##   1   1   1   1   1   1   1   1   1   1
##   vertex.names:
##    character valued attribute
##    186 valid vertex names
##
## Edge attributes:
##
##  Count:
##    numeric valued attribute
##    attribute summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   1.000   1.057   1.000   3.000
##
## Network edgelist matrix:
##       [,1] [,2]
##  [1,]  158    1
##  [2,]    1    5
##  [3,]    2    5
```

```
##  [4,]   34    5
##  [5,]   36    5
##  [6,]   38    5
##  [7,]   41    5
##  [8,]   17   18
##  [9,]   19   20
## [10,]  105   39
## [11,]   68   49
## [12,]   57   58
## [13,]   41   59
## [14,]   60   59
## [15,]    1   60
## [16,]   36   60
## [17,]   38   60
## [18,]   39   60
## [19,]   41   60
## [20,]   45   60
## [21,]   48   60
## [22,]   54   60
## [23,]   57   60
## [24,]   63   60
## [25,]  135   60
## [26,]   59   61
## [27,]   60   61
## [28,]   54   62
## [29,]  135   64
## [30,]   68   66
## [31,]   60   67
## [32,]   75   68
## [33,]  135   68
## [34,]  155   68
## [35,]  134   74
## [36,]  135   74
## [37,]   68   76
## [38,]   74   76
## [39,]   91   87
## [40,]  103   91
## [41,]  100   92
## [42,]   97   98
## [43,]  105  104
## [44,]  137  134
## [45,]   64  135
## [46,]    1  136
## [47,]   36  136
## [48,]   41  136
## [49,]   68  136
## [50,]  134  136
## [51,]  144  136
## [52,]  133  137
## [53,]  141  138
## [54,]  141  139
## [55,]  136  142
## [56,]  136  144
## [57,]   68  149
```

```
## [58,]  150  149
## [59,]  151  149
## [60,]  152  149
## [61,]  153  149
## [62,]  154  149
## [63,]  158  155
## [64,]  155  157
## [65,]    1  158
## [66,]  159  158
## [67,]   68  160
## [68,]  163  161
## [69,]  155  172
## [70,]  179  182
```

```r
gplot(mids_1993)
```



```r
dyad.census(mids_1993)
```

```
##      Mut Asym  Null
## [1,]   3   64 17138
```

```r
grecip(mids_1993)
```

```
##       Mut
## 0.9962802
```

```r
cug1_size <- cug.test(mids_1993, grecip, cmode = c("size"), FUN.args = (measure="edgewise"))
cug1_size
```

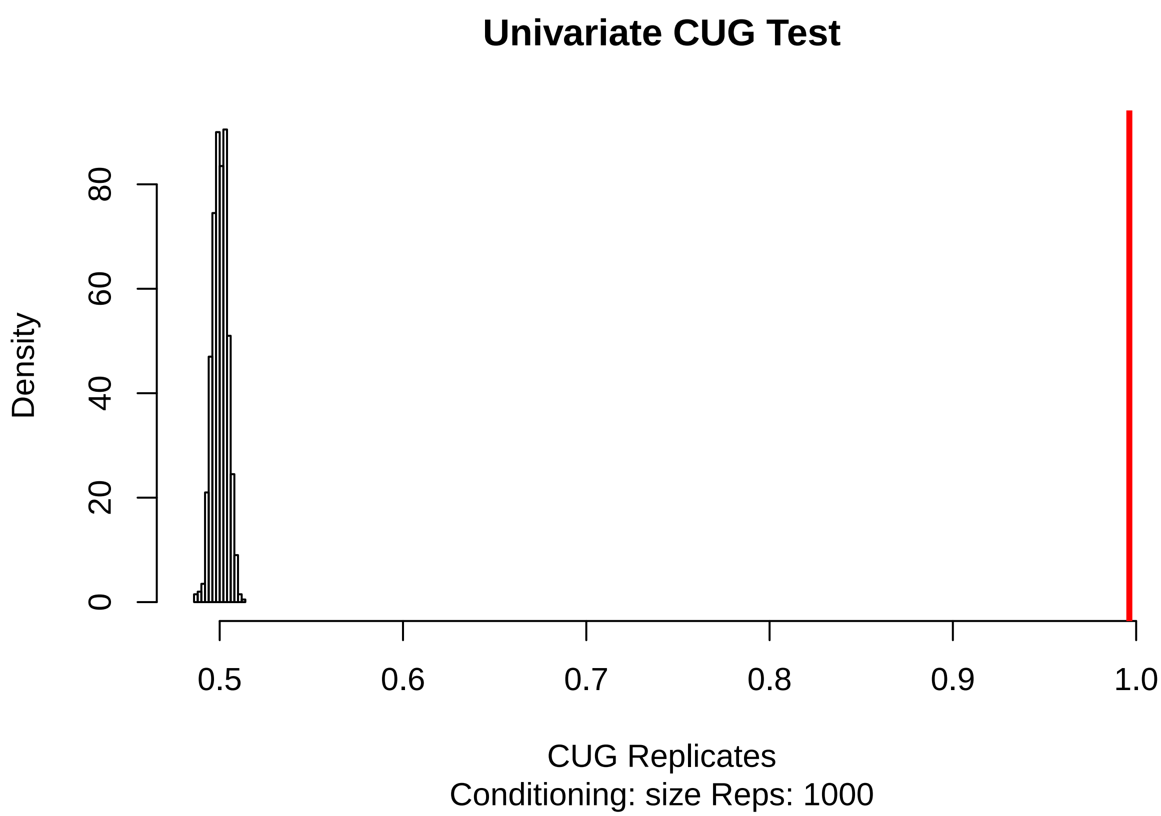
```
##
## Univariate Conditional Uniform Graph Test
##
## Conditioning Method: size
## Graph Type: digraph
## Diagonal Used: FALSE
```

```
## Replications: 1000
##
## Observed Value: 0.9962802
## Pr(X>=Obs): 0
## Pr(X<=Obs): 1
```

```r
summary(cug1_size)
```

```
##           Length Class  Mode
## obs.stat     1   -none- numeric
## rep.stat  1000   -none- numeric
## mode         1   -none- character
## diag         1   -none- logical
## cmode        1   -none- character
## plteobs      1   -none- numeric
## pgteobs      1   -none- numeric
## reps         1   -none- numeric
```

```r
plot.cug.test(cug1_size)
```




```r
cug1_edge <- cug.test(mids_1993, grecip, cmode = c("edges"), FUN.args = (measure="edgewise"))
cug1_edge
```

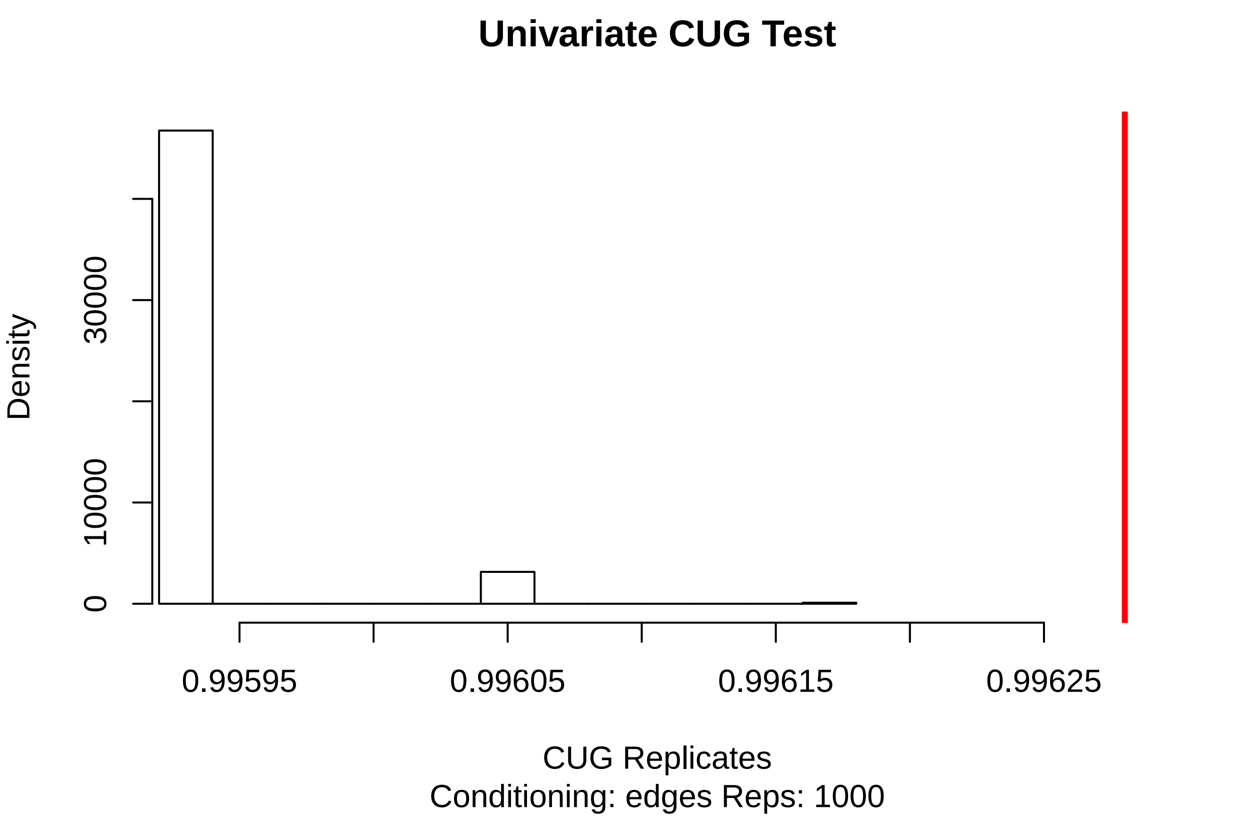
```
##
## Univariate Conditional Uniform Graph Test
##
```

```
## Conditioning Method: edges
## Graph Type: digraph
## Diagonal Used: FALSE
## Replications: 1000
##
## Observed Value: 0.9962802
## Pr(X>=Obs): 0
## Pr(X<=Obs): 1
```

```
summary(cug1_edge)
```

```
##          Length Class  Mode
## obs.stat    1   -none- numeric
## rep.stat 1000   -none- numeric
## mode        1   -none- character
## diag        1   -none- logical
## cmode       1   -none- character
## plteobs     1   -none- numeric
## pgteobs     1   -none- numeric
## reps        1   -none- numeric
```

```
plot.cug.test(cug1_edge)
```



If the proposition is true, the statistic whose value might deviate from a random baseline: Edgewise Reciprosity Baseline Distribution considered: size and edges. If the proposition is true, statistic would be greater than the baseline.

The plot shows significant departure from the baseline distribution.The p value calculated is less than the significance value of 0.05. Therefore, we reject the null hypothesis. Thus,in militarized interstate disputes, hostile acts are disproportionately likely to be responded to in kind.

```r
#Statistic: Transitivity
#Baseline Model: Edges
#Direction of Deviation : Lower
gtrans(mids_1993)
```

(b) When engaging in disputes, nations behave in accordance with the notion that "the enemy of my enemy is not my enemy".

```
## [1] 0.02409639
```

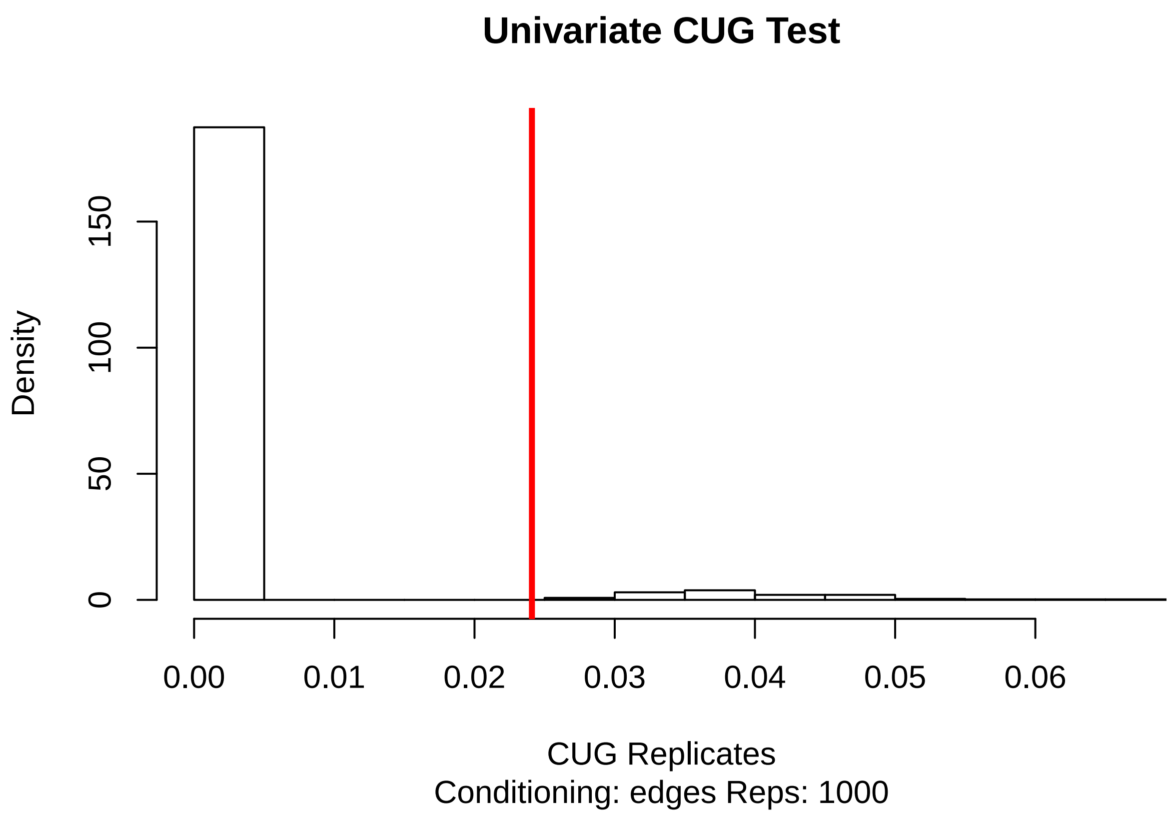
```r
cug2_edge <- cug.test(mids_1993, gtrans, cmode = c("edges"))
cug2_edge
```

```
##
## Univariate Conditional Uniform Graph Test
##
## Conditioning Method: edges
## Graph Type: digraph
## Diagonal Used: FALSE
## Replications: 1000
##
## Observed Value: 0.02409639
## Pr(X>=Obs): 0.063
## Pr(X<=Obs): 0.937
```

```r
summary(cug2_edge)
```

```
##          Length Class  Mode
## obs.stat    1   -none- numeric
## rep.stat 1000   -none- numeric
## mode        1   -none- character
## diag        1   -none- logical
## cmode       1   -none- character
## plteobs     1   -none- numeric
## pgteobs     1   -none- numeric
## reps        1   -none- numeric
```

```r
plot.cug.test(cug2_edge)
```



If the proposition is true, the statistic whose value might deviate from a random baseline: Transitivity Baseline Distribution considered: Edges. If the proposition is true, statistic would be lower than the baseline.

The plot shows that there is not a noteworthy departure from the baseline model. The p value calculated is greater than the significance value of 0.05 . Therefore, we do not have suuficient evidence to reject the null hypothesis. Thus, when engaging in disputes, nations do not behave in accordance with the notion that "the enemy of my enemy is not my enemy".

```
#Statistic: Indegree Centralization
#Baseline Model: Edges and sizes
#Direction of Deviation : Higher
cug3_size <- cug.test(mids_1993, grecip, cmode = c("size"), FUN.args = (measure="edgewise"))
cug3_size
```

(c) Given the number of disputes at any given time, as small number of nations will receive a disproportionate share of aggressive acts.

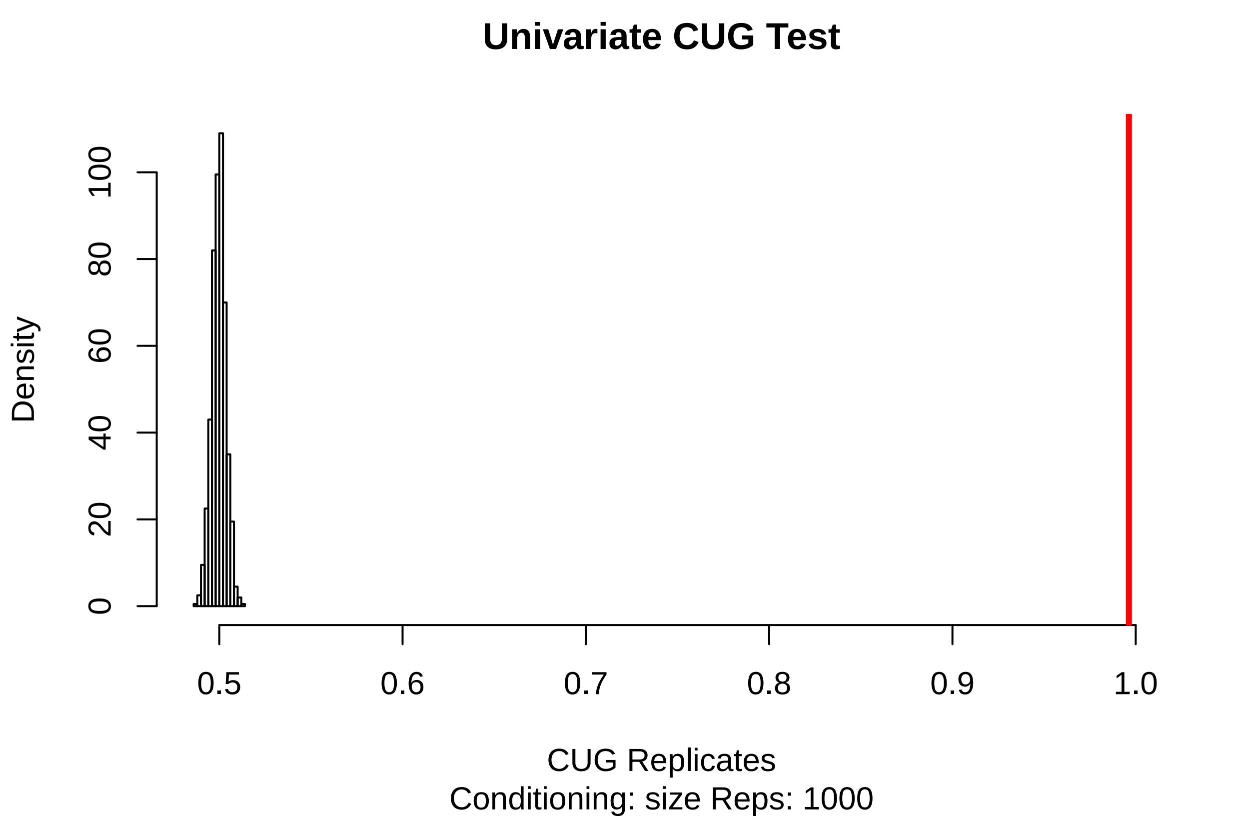
```
##
## Univariate Conditional Uniform Graph Test
##
## Conditioning Method: size
## Graph Type: digraph
## Diagonal Used: FALSE
## Replications: 1000
##
## Observed Value: 0.9962802
```

```
## Pr(X>=Obs): 0
## Pr(X<=Obs): 1
```

```
summary(cug3_size)
```

```
##          Length Class  Mode
## obs.stat    1   -none- numeric
## rep.stat 1000   -none- numeric
## mode        1   -none- character
## diag        1   -none- logical
## cmode       1   -none- character
## plteobs     1   -none- numeric
## pgteobs     1   -none- numeric
## reps        1   -none- numeric
```

```
plot.cug.test(cug3_size)
```

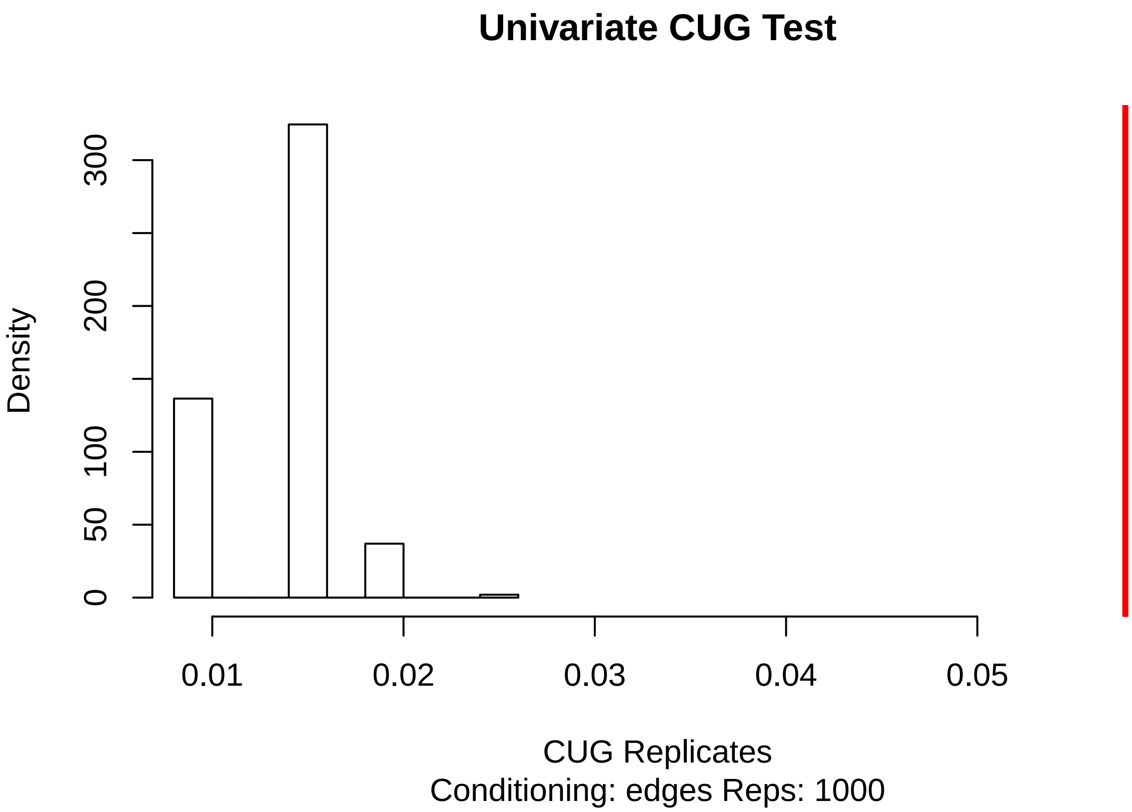



```
cug3_edge <- cug.test(mids_1993, centralization, cmode = c("edges"), FUN.args = list(FUN=degree, cmode=
summary(cug3_edge)
```

```
##          Length Class  Mode
## obs.stat    1   -none- numeric
## rep.stat 1000   -none- numeric
## mode        1   -none- character
## diag        1   -none- logical
## cmode       1   -none- character
```

```
## plteobs      1    -none- numeric
## pgteobs      1    -none- numeric
## reps         1    -none- numeric
```

```r
plot.cug.test(cug3_edge)
```



If the proposition is true, the statistic whose value might deviate from a random baseline: Indegree Centralization. Baseline Distribution considered: Edges and Size. If the proposition is true, statistic would be greater than the baseline.

The plot shows significant departure from the baseline distribution.The p value calculated is less than the significance value of 0.05. Therefore, we reject the null hypothesis. Thus for a given number of disputes at any given time, a small number of nations will receive a disproportionate share of aggressive acts.