Social Network Analysis

QAP Test

Robin Burke

DePaul University

Chicago, IL

Project

- Visualization due today
- Next milestone
  - Feedback to support group

Rest of the quarter

- 5/15
  - Visualization response
  - EC 6
- 5/22
  - Lab 2 (5/15)
  - Draft visualizations
- 5/29
  - Homework 6
- 6/5
  - Final report
  - Scuiz points

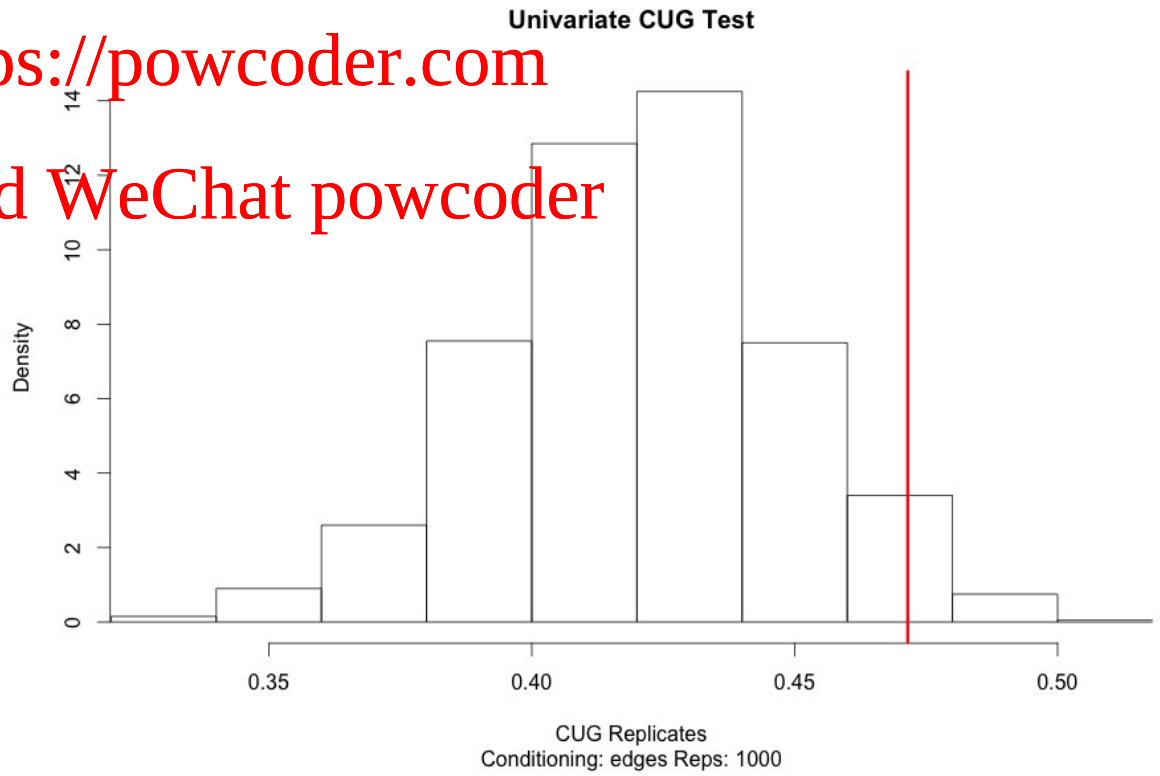
Review: CUG test

- Generate many random graphs
- See if the properties of those graphs
  - match the graph you observed
- If so, you can't reject the null hypothesis

Example: Transitivity

- Pr(X>=Obs): 0.036
- Pr(X<=Obs): 0.964

More transitive than almost all random networks



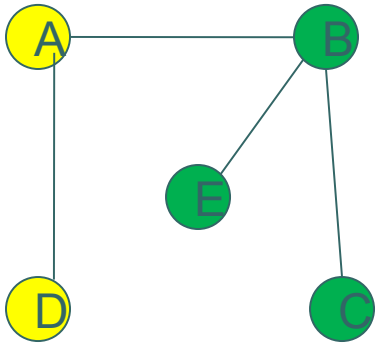
Example

- New data set
  - ps5 social network data set
    - we'll see this again for ERGM
- CUG for
  - assortativity by color
  - transitivity
  - reciprocity

Quadratic Assignment Procedure (QAP)

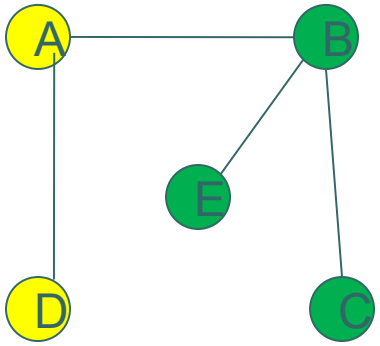
- Like CUG
  - the goal is to get a non-parametric test of a network property
- With QAP
  - the goal is to hold network structure fixed
  - and randomly scramble the vertices

Example



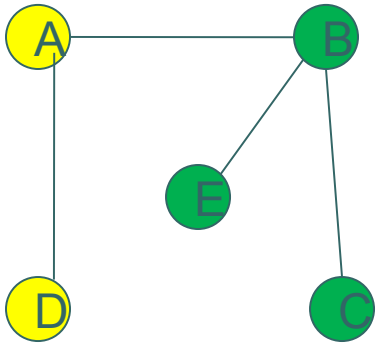
- Is this graph assortative?
  - (yes)
  - more than one would expect?
- We could take all rearrangements of these vertices
  - look at the assortativity values of all these graphs
- Where does the assortativity of the observed graph
  - lie relative to all of the permutations

Null hypothesis



- The assortativity of the graph could have arisen through random placement of nodes within the network structure
- Combination of
  - distribution of node types
  - particular network configuration

Example



- In this case
  - you could try all possibilities
    - only 10 (because order doesn't matter and only need to choose two positions for A and D)
- In a real network
  - need to sample from a very large number of possible permutations

In R

```
myqaptest(gr,
    assortativity.nominal,
    types=V(gr)$type,
    directed=FALSE)
```

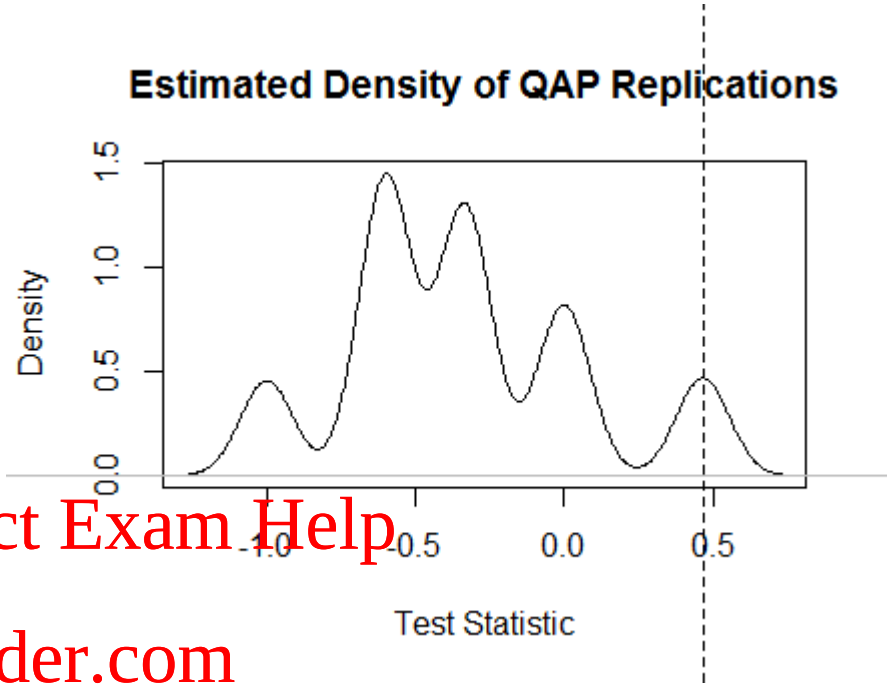
network

function

function args

Note: assortativity.nominal doesn't accept types = 0. Dreaded invalid "types" vector error. Or R crashes totally

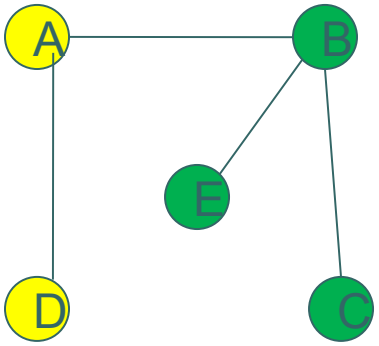
Result



- print.qaptest()

```
QAP Test Results
Estimated p-values:
p(f(perm) >= f(d)): 0.101
p(f(perm) <= f(d)): 1
> summary.qaptest()
QAP Test Results
Estimated p-values:
        p(f(perm) >= f(d)): 0.101
        p(f(perm) <= f(d)): 1
Test Diagnostics:
        Test Value (f(d)): 0.4666667
        Replications: 1000
        Distribution Summary:...
```

Result



- The assortativity of this network is highly associated with the placement of types in the structure
  - not with the number of nodes of each type
  - or the structure of the network
- This is the only configuration (out of 10) that has assortativity this high

QAP Test

- Allows us to look for effects
  - network structure vs distribution of node attributes
- We can ask
  - is the value associated with this particular network organization "unique" or "rare"
- Cannot use this for network properties
  - like transitivity
  - all vertex permutations will yield the same value

Example

- gr3
  - transitivity?
- QAP test is meaningless
  - transitivity is a structural property
  - rearranging the vertices doesn't change it
- Can look at assortativity

Results

- These particular configurations of individuals within the network
  - somewhat disassociative with the color groups
- 84% of random configurations showed greater association
  - `Estimated p-values:`
  - `p(f(perm) >= f(d)): 0.839`
  - `p(f(perm) <= f(d)): 0.161`

probability random configuration will have a test stat >= actual value

Conclusion

- QAP test useful to test configuration properties
  - prime example: associativity
  - also if you have multiple networks
- Do all re-labelings of nodes show the same properties?
  - network structure fixed
- A “harder” test than CUG
  - we know networks aren’t random

Example