Graph Correlation, QAP, and Network Regression

SOC 280: Analysis of Social Network Data

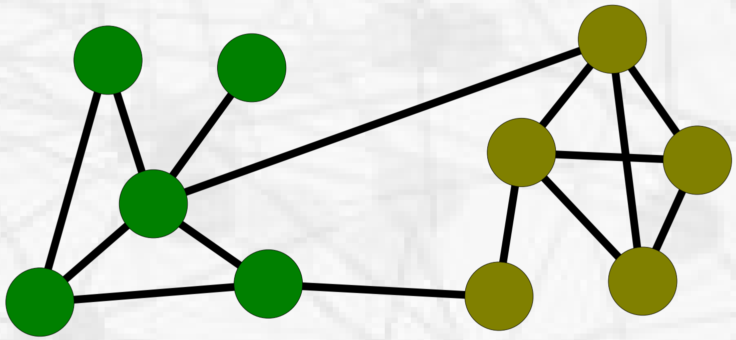
Carter T. Butts
Department of Sociology and
Institute for Mathematical Behavioral Sciences
University of California, Irvine

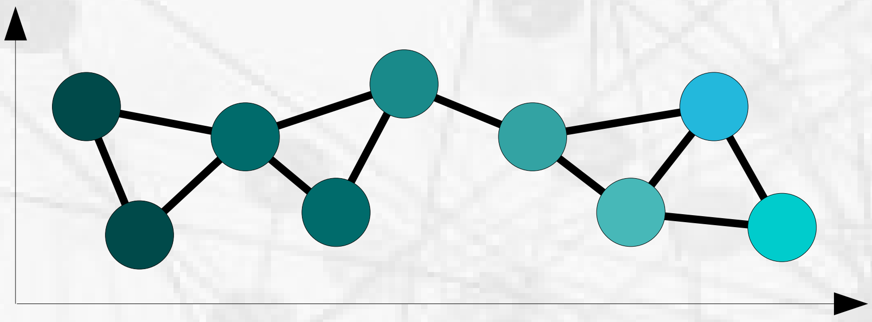
From Description to Modeling

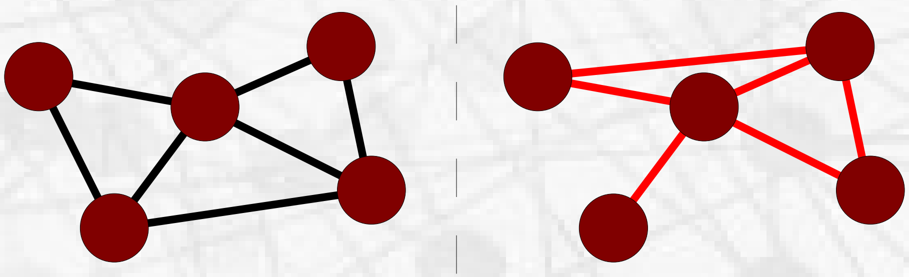
- **Ultimately, want to do more than describe networks**
- **Network modeling: predict the formation and structure of social networks**
- **Have already seen a few simple examples**
  - Baseline models, confirmatory density-based models
- **Today, some simple ideas that bridge the two**
  - Graph correlation, network regression (and associated tests)
  - Often used in semi-descriptive (at least exploratory fashion)
  - Can be used as more serious models; move us towards a more systematic modeling approach
    - Good starting point for notions like permutation models
    - Useful practical tools in many situations

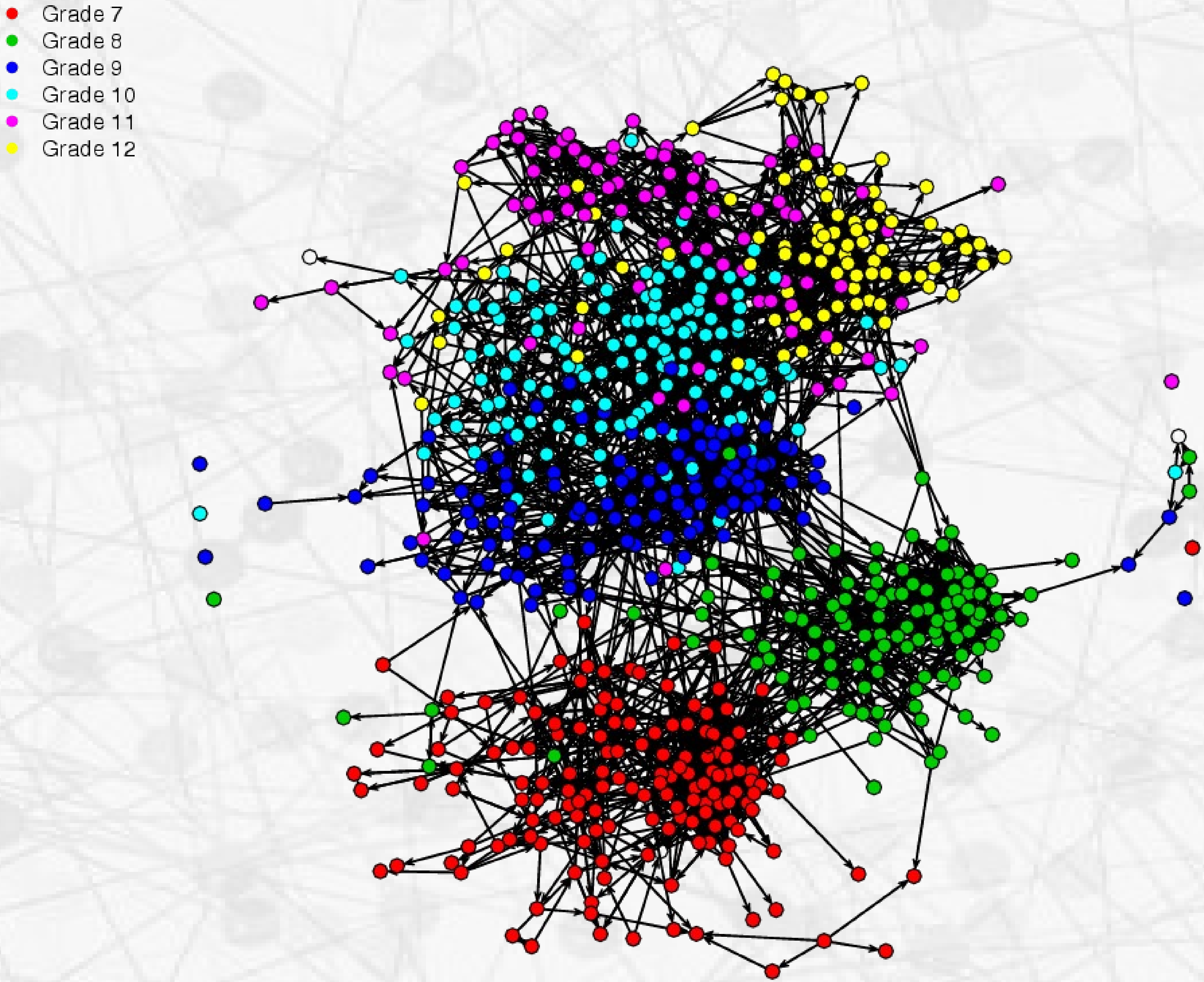
Initial Intuition: Factors in Tie Formation

- **All ties are not equally probable**
  - Chance of an (i,j) edge may depend on properties of i and j
  - Can also depend on other (i,j) relationships
- **Some examples:**
  - Homophily
  - Propinquity
  - Multiplexity

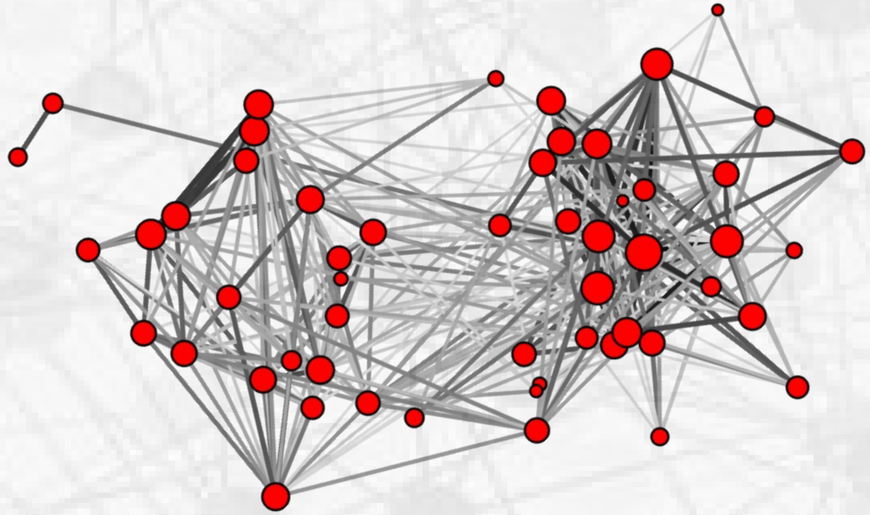








AddHealth Friendship Network, by Grade



Freeman et al.

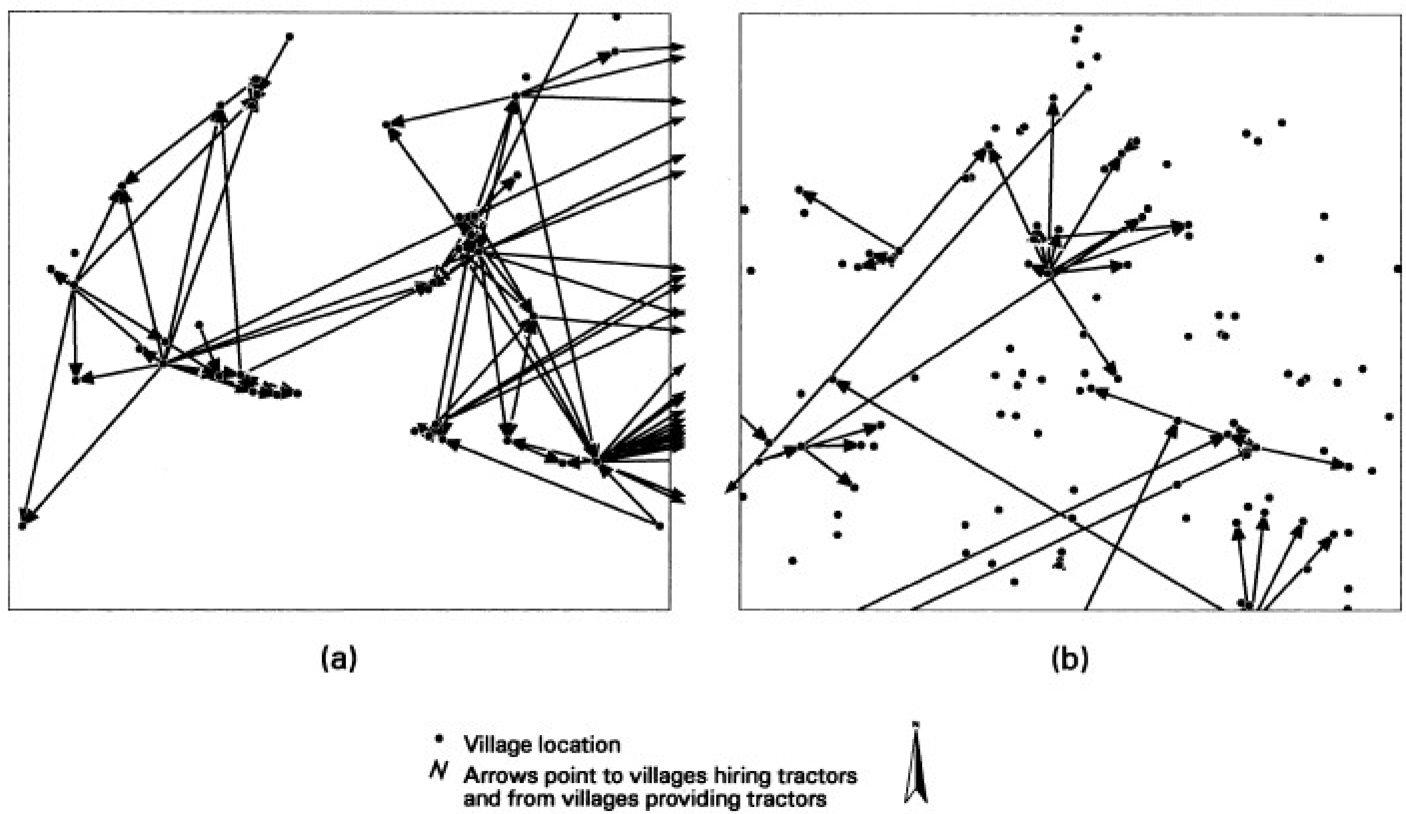
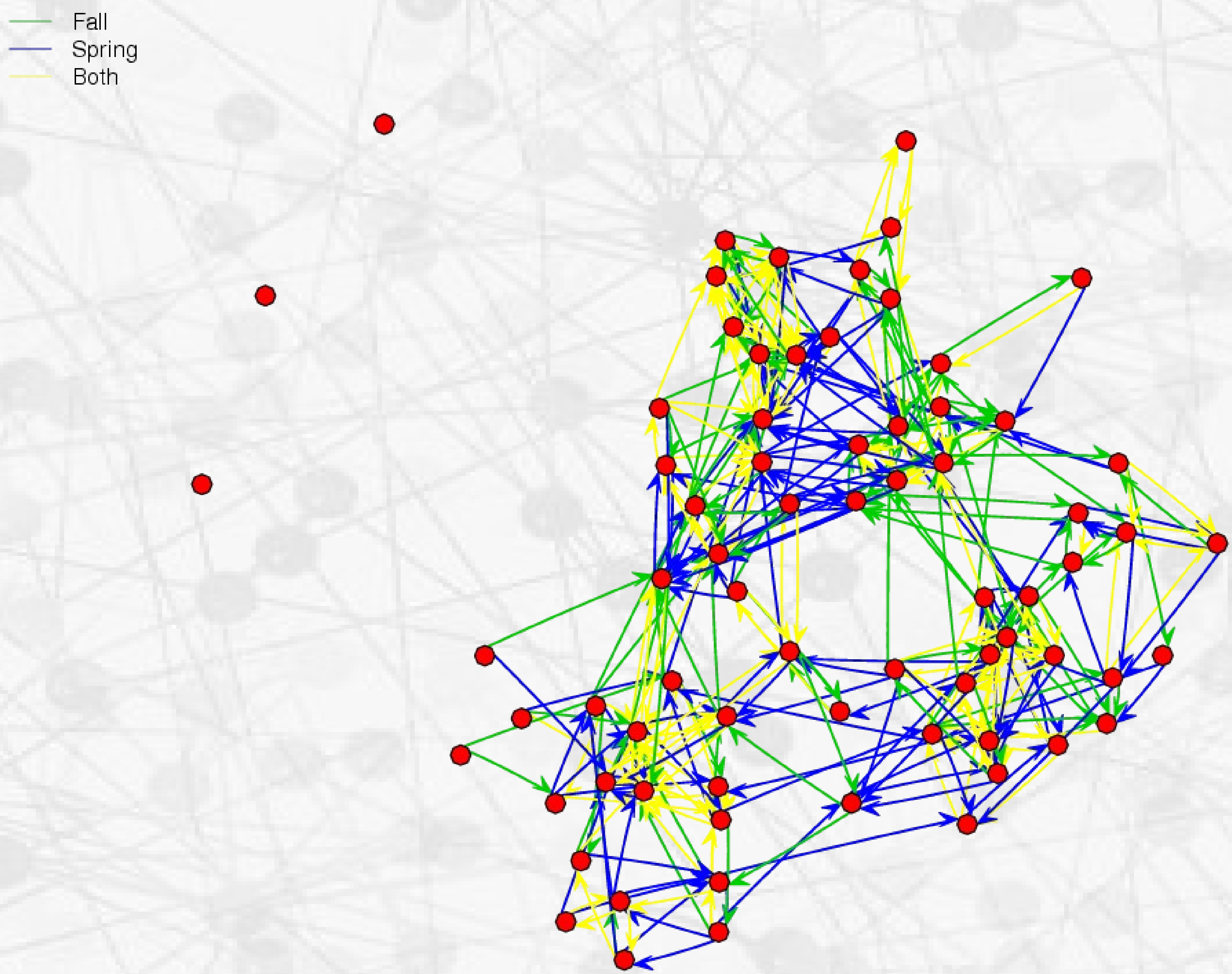


Fig. 3. Tractor hiring network in two regions of Nang Rong.

Faust et al.



Boy's School Friendship Network, Coleman 1964

Graph Correlation

- **Simple way of comparing graphs on same vertex set: *graph correlation***
  - Start with *graph mean* – grand mean of adjacency matrix
  - *Graph covariance*: elementwise covariance of adjacency matrices
    - *Graph variance*: covariance of graph with itself
  - *Graph correlation*: elementwise correlation of adjacency matrices
- **Easily interpretable, works with valued data, etc.**

$$\mathbf{X} = \begin{bmatrix} - & 1 & 1 \\ 0 & - & 1 \\ 0 & 0 & - \end{bmatrix}, \mathbf{Y} = \begin{bmatrix} - & 1 & 0 \\ 0 & - & 1 \\ 0 & 0 & - \end{bmatrix}$$

$$\overline{\mathbf{X}} = \frac{\sum_{(i,j)} \mathbf{X}_{\mathbf{ij}}}{N(N-1)} = \frac{1}{2,} \; \overline{\mathbf{Y}} = \frac{\sum_{(i,j)} \mathbf{Y}_{\mathbf{ij}}}{N(N-1)} = \frac{1}{3}$$

$$\mathrm{Cov}(\mathbf{X}, \mathbf{Y}) = \frac{\sum_{(i,j)} (\mathbf{X}_{\mathbf{ij}} - \overline{\mathbf{X}})(\mathbf{Y}_{\mathbf{ij}} - \overline{\mathbf{Y}})}{N(N-1)-1}$$

$$= 0.2$$

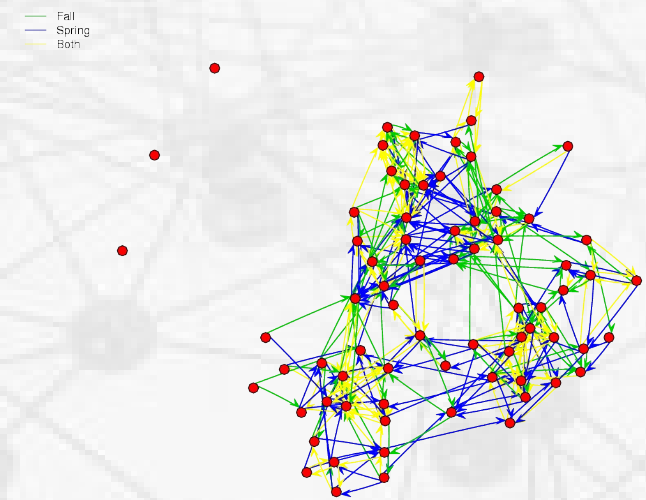
$$\mathrm{Var}(\mathbf{X}) = \mathrm{Cov}(\mathbf{X}, \mathbf{X}) = 0.3$$

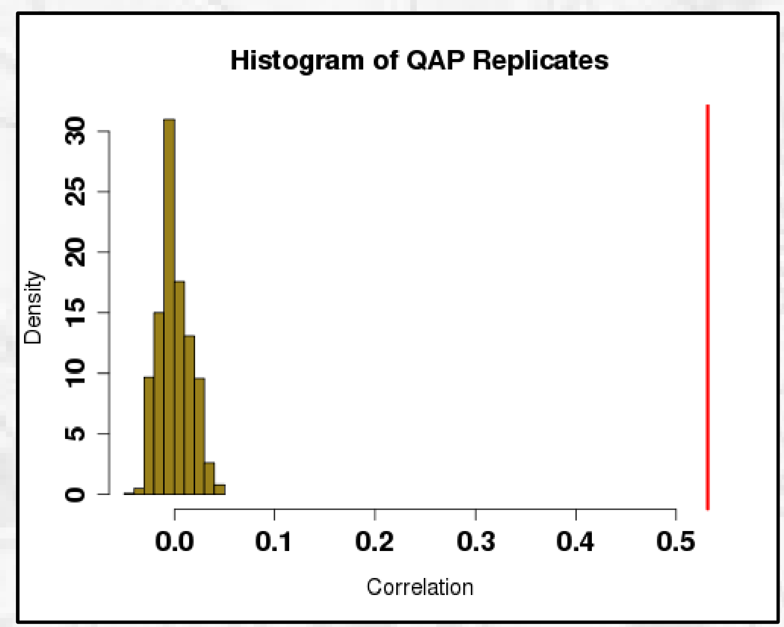
$$\mathrm{Var}(\mathbf{Y}) = \mathrm{Cov}(\mathbf{Y}, \mathbf{Y}) = 0.27$$

$$\rho(\mathbf{X}, \mathbf{Y}) = \frac{\mathrm{Cov}(\mathbf{X}, \mathbf{Y})}{\sqrt{\mathrm{Var}(\mathbf{X})\,\mathrm{Var}(\mathbf{Y})}} = 0.71$$

Hubert's QAP

- **How to tell if our observed correlation is “large”?**
  - Due to autocorrelation, large excursions possible
- **Hubert's QAP**
  - Fix one matrix, repeatedly permute the other
  - Compare observed correlation w/permutation distribution
  - As usual, look to the quantiles of the observed correlation to determine *p*-values
  - Interpretation: CUG test w/<u>all</u> unlabeled properties fixed





Network Regression

- **Simple family of models for predicting social ties**
  - Special case of standard OLS regression
  - Dependent variable is a network adjacency matrix
- **Model form:**

$$\mathbf{E}Y_{ij} = \beta_0 + \beta_1 X_{1\mathrm{ij}} + \beta_2 X_{2\mathrm{ijk}} + \ldots + \beta_p X_{pij}$$

  - where $\mathbf{E}$ is the expectation operator (analogous to "mean" or "average"), Y_{ij} is the value of the edge from i to j on the dependent relation with adjacency matrix $\mathbf{Y}$, X_{kij} is the value of the kth predictor for the (i,j) ordered pair, and $\beta_0,\ldots,\beta_p$ are coefficients

Dependent Variable

- **From previous, dependent variable is an adjacency matrix**
  - Standard case: dichotomous data
    - Interpretation: model predicts tie probability (maybe not well)
  - Valued case
    - Interpretation: model predicts tie strength
- **To prepare data, just code network into adjacency matrix form**
  - No special tactics required for one-mode data
  - For two-mode data, either treat as one-mode or use projection matrix

Independent Variable(s)

- **For independent variables (X), may need to prepare data**
  - Always take matrix form, but may be based on vector data
- **Several examples:**
  - Simple adjacency matrices
  - Sender/receiver effects
  - Attribute differences
  - Elements held in common

Types of Predictors: Adjacency Matrices

- **Simplest predictor: standard adjacency matrix**
- **When to use it?**
  - When you think that adjacency in one relationship (or tie strength in that relationship) might affect tie probability/strength in the dependent relation
- **What does it mean?**
  - Unvalued: if $X_{ij}=1$, predicted probability of $Y_{ij}=1$ increases by β
  - Valued: unit change in X_{ij} predicted to increase Y_{ij} by β
- **Examples**
  - Friendship might increase chance of collaboration
  - Direct reporting might increase chance of advice-seeking
- **In R**
  - Given response *myY* and predictor *myX*
  - *ymod<-netlm(myY,myX)*
  - No special tricks required, in general

Types of Predictors: Sender/Receiver Effects

- **Fancier idea: predictors for outdegree/indegree**
- **When to use it?**
  - When you expect that some actors will send/attract more ties than others
- **What does it mean?**
  - Unvalued: each unit change in X_i (X_j) increases all outgoing (incoming) tie probabilities by β
  - Valued: as above, but changes are in tie strength
- **Examples**
  - More senior personnel may give more advice
  - Federal organizations may receive more communications during disasters
- **In R**
  - Given predictor vector *x* (*x[i]* is effect for *i*th vertex)
    - Receiver: *myX<-sapply(x,rep,length(x))*
    - Sender: *myX<-t(sapply(x,rep,length(x)))*
  - Use *myX* normally in *netlm*

Types of Predictors: Attribute Differences

- **Also useful: differences in attributes**
- **When to use it?**
  - When ties are predicted to be less probable/weaker between individuals who are more/less similar
- **What does it mean?**
  - Unvalued: each unit of absolute difference in X_{ij} increases tie probability by β
  - Valued: same, but change is in tie strength
- **Examples**
  - Children of differing genders may be less likely to be friends
  - Organizations with the same scale of operations may be more likely to collaborate
- **In R**
  - Given attribute vector *x* (*x[i]* is value for *i*th vertex)
    - Numeric: *myX<-abs(outer(x,x,"-"))*
    - Categorical: *myX<-outer(x,x,"!=")*
  - Now, use myX normally

Types of Predictors: Elements Held in Common

- **Less widely used: common elements**
- **When to use it?**
  - When individuals are associated with events or other elements which might promote/inhibit interaction
- **What does it mean?**
  - Unvalued: each additional common event increases tie probability by β
  - Valued: each common event increases tie strength by β
- **Examples**
  - People attending more events together may be more likely to know each other
  - Organizations with more shared tasks may be more likely to collaborate
- **In R**
  - Given a two-mode matrix of vertices by events, *x*
    - *myX<-x%*%t(x)*
  - As before, now use normally

Fitting the Model

- **Given Y and X, want to estimate β**
  - Estimates tell us how X is related to Y
    - Interpret coefficients per previous slides
  - Also use null hypothesis tests to compare observed results to random ("chance") baseline
    - Generally, use so-called “MRQAP” procedure – essentially, QAP applied to residuals after semi-partialling
    - Assess via permuted *t* statstic

- **In R, we do this with *netlm***
  - Analogous to *lm* (the R multiple regression function)
  - Basic syntax (see *?netlm*)
    - *myfit<-netlm(y,x)*
      - *y* should be an *nxn* matrix
      - *x* can be an *nxn* matrix, a *pxnxn* array, or a list of *nxn* matrices
    - Some extra arguments
      - *mode="graph"* (undirected data)
      - *diag=TRUE* (diagonals count)
      - *reps=250* (faster fitting)

Example: Cheyenne EMON

- **Dependent variable: reported communication**
  - 4 point frequency scale
- **Independent variables**
  - Command rank (receiver effect)
  - Sponsorship (difference)
- **R code**
  - Setup (extracting from *emon* data)
    - *data(emon)*
    - *Y<-as.sociomatrix(emon[[1]], "Frequency")*
    - *Y[Y>0]<-5-Y[Y>0]*
- **R code, cont.**
    - *crk<-emon[[1]]%v% "Command.Rank.Score"*
    - *spon<-emon[[1]]%v%"Sponsorship"*
  - Preparing X
    - *Xcr<-sapply(crk,rep,length(crk))*
    - *Xsp<-outer(spon,spon,"!=")*
  - Fitting model
    - *cmfit<-netlm(Y,list(Xcr,Xsp))*
    - Can examine with *print(cmfit)*, *summary(cmfit)*

Results

- **Summary for cmfit:**

```
OLS Network Model

Residuals:
        0%        25%        50%        75%       100%
-3.0330379 -1.2000632 -0.9677143  1.4832930  3.0322857

Coefficients:
            Estimate    Pr(<=b) Pr(>=b) Pr(>=|b|)
(intercept)  1.45400112 1.000   0.000   0.000
x1           0.05163309 1.000   0.000   0.000
x2          -0.48628683 0.104   0.896   0.197

Residual standard error: 1.676 on 179 degrees of freedom
Multiple R-squared: 0.1233      Adjusted R-squared: 0.1135
F-statistic: 12.59 on 2 and 179 degrees of freedom, p-value: 7.692e-06
```

Results

- **Summary for cmfit:**

```
OLS Network Model

Residuals:
        0%        25%        50%        75%       100%
-3.0330379 -1.2000632 -0.9677143  1.4832930  3.0322857

Coefficients:
            Estimate    Pr(<=b) Pr(>=b) Pr(>=|b|)
(intercept)  1.45400112 1.000   0.000   0.000
x1           0.05163309 1.000   0.000   0.000
x2          -0.48628683 0.104   0.896   0.197

Residual standard error: 1.676 on 179 degrees of freedom
Multiple R-squared: 0.1233      Adjusted R-squared: 0.1135
F-statistic: 12.59 on 2 and 179 degrees of freedom, p-value: 7.692e-06
```

Results

- **Summary for cmfit:**

```
OLS Network Model

Residuals:
        0%        25%        50%        75%       100%
-3.0330379 -1.2000632 -0.9677143  1.4832930  3.0322857

Coefficients:
            Estimate    Pr(<=b) Pr(>=b) Pr(>=|b|)
(intercept)  1.45400112 1.000   0.000   0.000
x1           0.05163309 1.000   0.000   0.000
x2          -0.48628683 0.104   0.896   0.197

Residual standard error: 1.676 on 179 degrees of freedom
Multiple R-squared: 0.1233      Adjusted R-squared: 0.1135
F-statistic: 12.59 on 2 and 179 degrees of freedom, p-value: 7.692e-06
```

Results

- **Summary for cmfit:**

```
OLS Network Model

Residuals:
        0%        25%        50%        75%       100%
-3.0330379 -1.2000632 -0.9677143  1.4832930  3.0322857

Coefficients:
            Estimate    Pr(<=b) Pr(>=b) Pr(>=|b|)
(intercept)  1.45400112 1.000   0.000   0.000
x1           0.05163309 1.000   0.000   0.000
x2          -0.48628683 0.104   0.896   0.197

Residual standard error: 1.676 on 179 degrees of freedom
Multiple R-squared: 0.1233      Adjusted R-squared: 0.1135
F-statistic: 12.59 on 2 and 179 degrees of freedom, p-value: 7.692e-06
```

Density

Command Rank

Sponsorship

Results

- **Summary for cmfit:**

```
OLS Network Model

Residuals:
        0%        25%        50%        75%       100%
-3.0330379 -1.2000632 -0.9677143  1.4832930  3.0322857

Coefficients:
            Estimate    Pr(<=b) Pr(>=b) Pr(>=|b|)
(intercept)  1.45400112 1.000   0.000   0.000
x1           0.05163309 1.000   0.000   0.000
x2          -0.48628683 0.104   0.896   0.197

Residual standard error: 1.676 on 179 degrees of freedom
Multiple R-squared: 0.1233      Adjusted R-squared: 0.1135
F-statistic: 12.59 on 2 and 179 degrees of freedom, p-value: 7.692e-06
```

Density

Command Rank

Sponsorship

Results

- **Summary for cmfit:**

```
OLS Network Model

Residuals:
        0%        25%        50%        75%       100%
-3.0330379 -1.2000632 -0.9677143  1.4832930  3.0322857

Coefficients:
            Estimate    Pr(<=b) Pr(>=b) Pr(>=|b|)
(intercept)  1.45400112 1.000   0.000   0.000        Density
x1           0.05163309 1.000   0.000   0.000        Command Rank
x2          -0.48628683 0.104   0.896   0.197        Sponsorship

Residual standard error: 1.676 on 179 degrees of freedom
Multiple R-squared: 0.1233      Adjusted R-squared: 0.1135
F-statistic: 12.59 on 2 and 179 degrees of freedom, p-value: 7.692e-06
```

Interpretation of Cheyenne EMON Model

- **Specific influences on communication**
  - Organizations viewed as having greater command/control function much more likely to receive ties
    - Command rank score varies from 0 to 40, so an effect of 0.05 is fairly large – could add up to 2 units of interaction
  - No significant effect for sponsorship
    - In this case, no clear tendency for organizations of different types to interact less often
- **Overall fit shows room for improvement**
  - Much left unexplained – other covariates may help