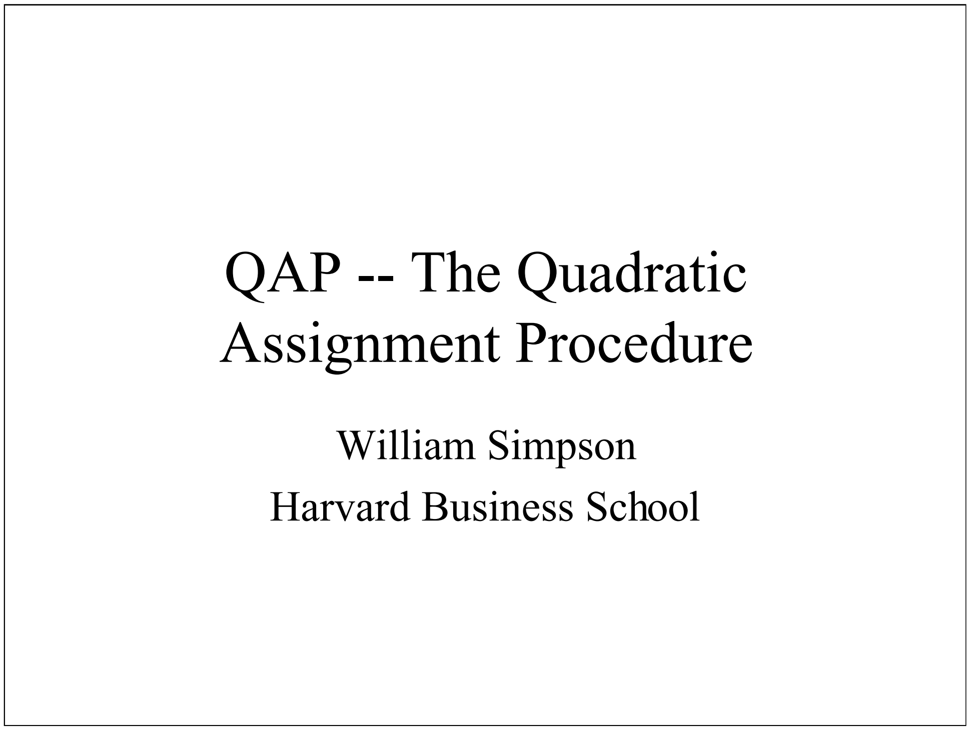


QAP -- The Quadratic Assignment Procedure

William Simpson

Harvard Business School

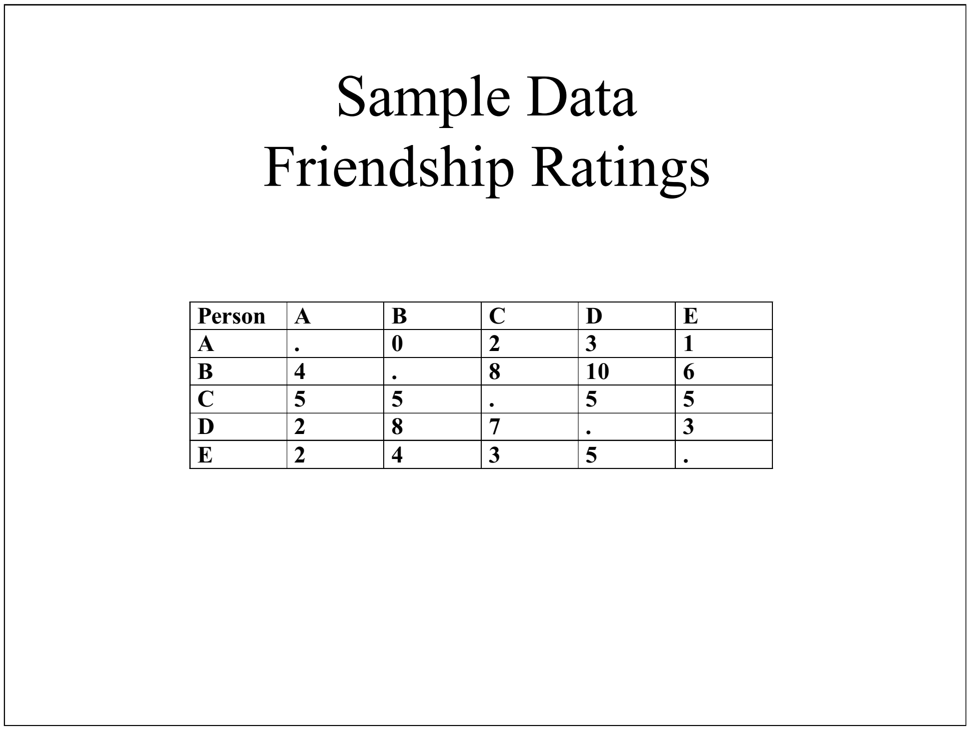
QAP is method that has been used in social network analysis, and is useful for analyzing dyadic data sets, i.e. data sets where pairs of entities are analyzed. Examples include:

How does level of trade between countries vary as a function of language similarity, level of development, and other factors? -- looking at pairs of countries

Do companies with overlapping boards of directors tend to perform similarly in the stock market? -- looking at pairs of companies

Are people more likely to be friends if they share similar characteristics, such as being about the same age, or using the same statistical software?

To explain QAP, I will start with this last example, (leaving out the statistical software part): are people who have similar ages more likely to be friends with each other?



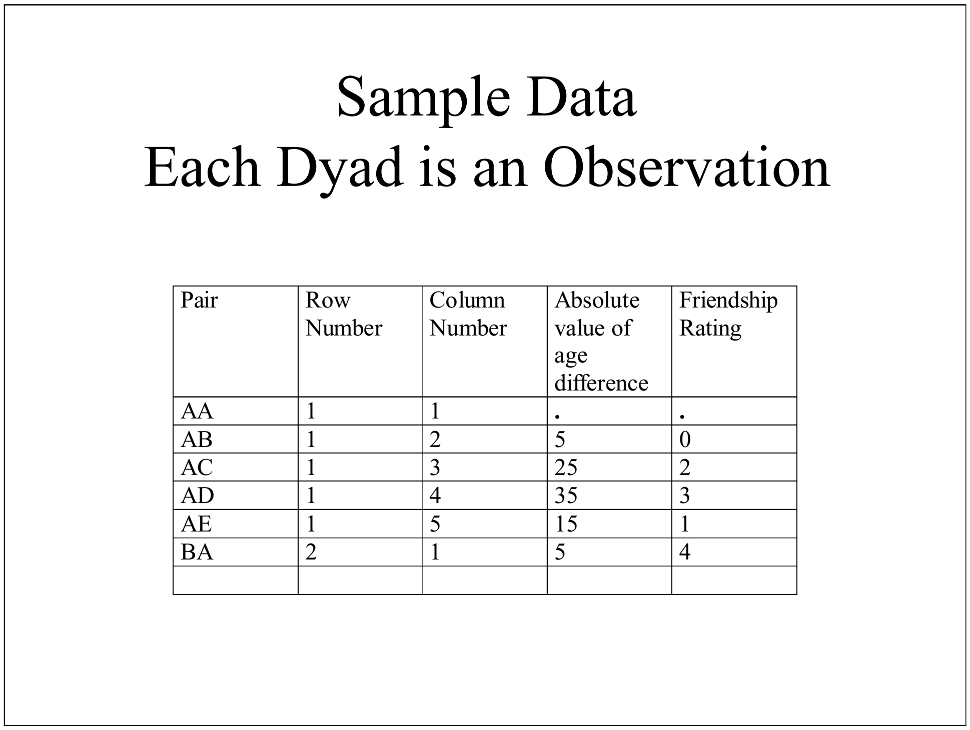
Sample Data
Friendship Ratings

Person	A	B	C	D	E
A	.	0	2	3	1
B	4	.	8	10	6
C	5	5	.	5	5
D	2	8	7	.	3
E	2	4	3	5	.

We set up our data as a square matrix, with each person on a row and also on a column.

We then ask each person to rate each other person as to how good a friend they are, in this case on a 0-10 scale. Notice that we don t collect self-ratings, and it s common in dyadic data sets that the diagonals will have missing values.

We also generate a matrix of absolute values of the age difference for each pair, which will have a similar format.



Sample Data
Each Dyad is an Observation

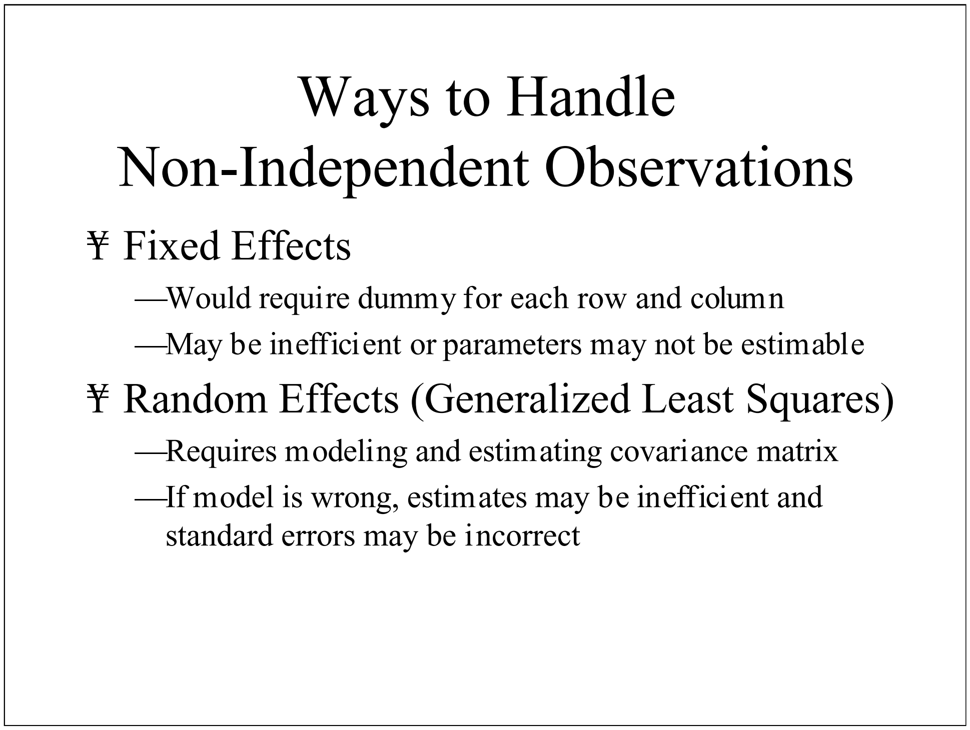
Pair	Row Number	Column Number	Absolute value of age difference	Friendship Rating
AA	1	1	.	.
AB	1	2	5	0
AC	1	3	25	2
AD	1	4	35	3
AE	1	5	15	1
BA	2	1	5	4

To analyze the data, we combine matrices and turn this into a data set where each pair is an observation. I've also added a row and column number, which will be needed for the QAP procedure.

Then we can do an OLS regression of friendship ratings on age difference and get a consistent coefficient estimate.

The problem is, that our standard error is wrong. The error terms in the regression will be correlated across observations. For example, in my sample data, person A gives consistently low friendship ratings, so person A's residuals will tend to be low. Most typically, observations in the same row or column will be positively correlated, and this will make the standard errors be too small and the p-values too optimistic.

Problems of non-independent observations come up in other contexts. A typical example is panel data, and there are a number of solutions.



Ways to Handle Non-Independent Observations

¥ Fixed Effects

—Would require dummy for each row and column

—May be inefficient or parameters may not be estimable

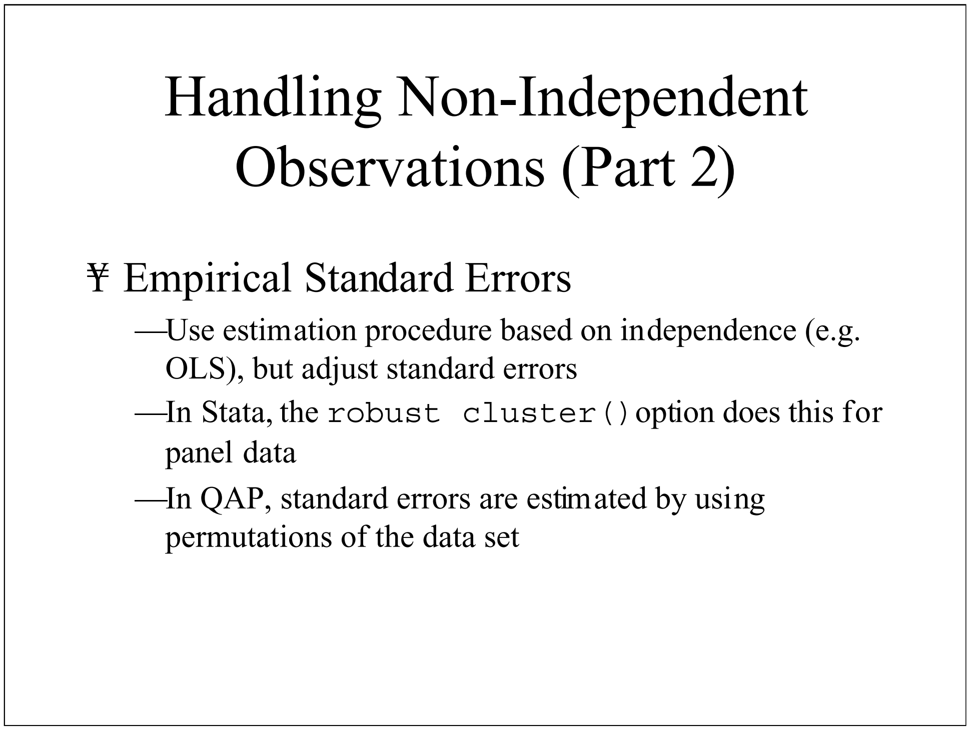
¥ Random Effects (Generalized Least Squares)

—Requires modeling and estimating covariance matrix

—If model is wrong, estimates may be inefficient and standard errors may be incorrect

One option would be to do a fixed effects model. For our dyadic data, that would involve putting in a dummy variable for each row and each column in the original matrix. In some cases, that can lead to inefficiency or to models where the substantive parameters can’t be estimated.

A second option would be to do a generalized least squares, assuming some form for the covariance matrix. This can be done, but the methods have not been as thoroughly worked out as for panel data.



Handling Non-Independent Observations (Part 2)

¥ Empirical Standard Errors

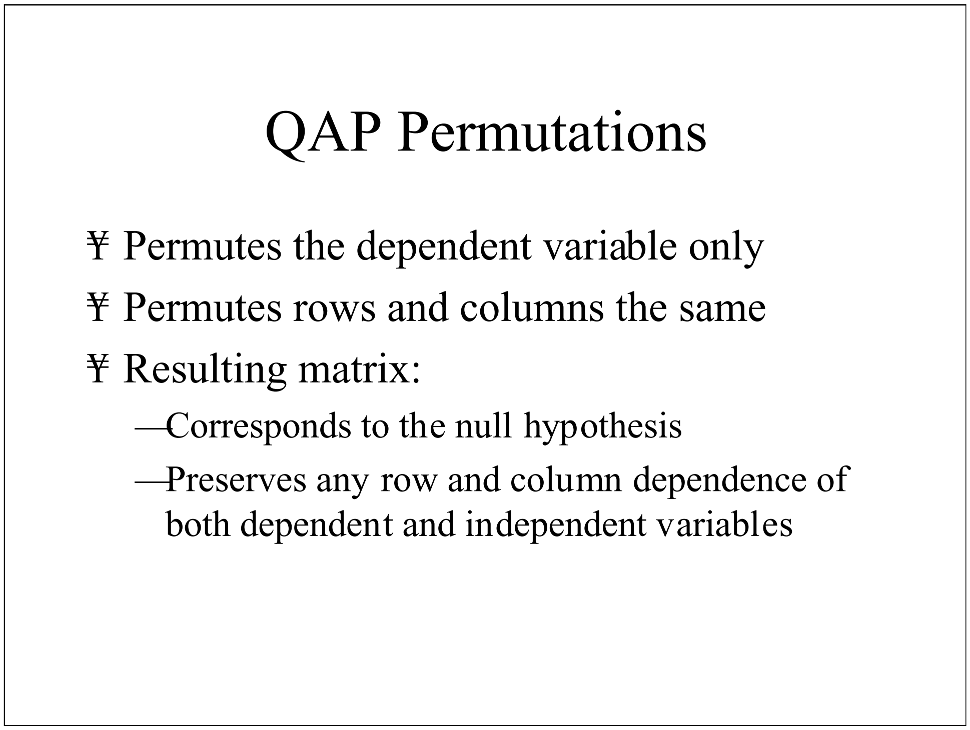
—Use estimation procedure based on independence (e.g. OLS), but adjust standard errors

—In Stata, the `robust cluster()` option does this for panel data

—In QAP, standard errors are estimated by using permutations of the data set

A third option is to stay with an estimation that assumes independence, such as OLS in regular regression, but to adjust the standard errors. For panel data, Stata's robust cluster option does this. But for dyadic data, we have two candidates for clustering on, the rows and the columns, and we need to adjust for both.

QAP is an alternative approach based on simulation; in particular, it uses permutations of the data set.



QAP Permutations

¥ Permutes the dependent variable only

¥ Permutes rows and columns the same

¥ Resulting matrix:

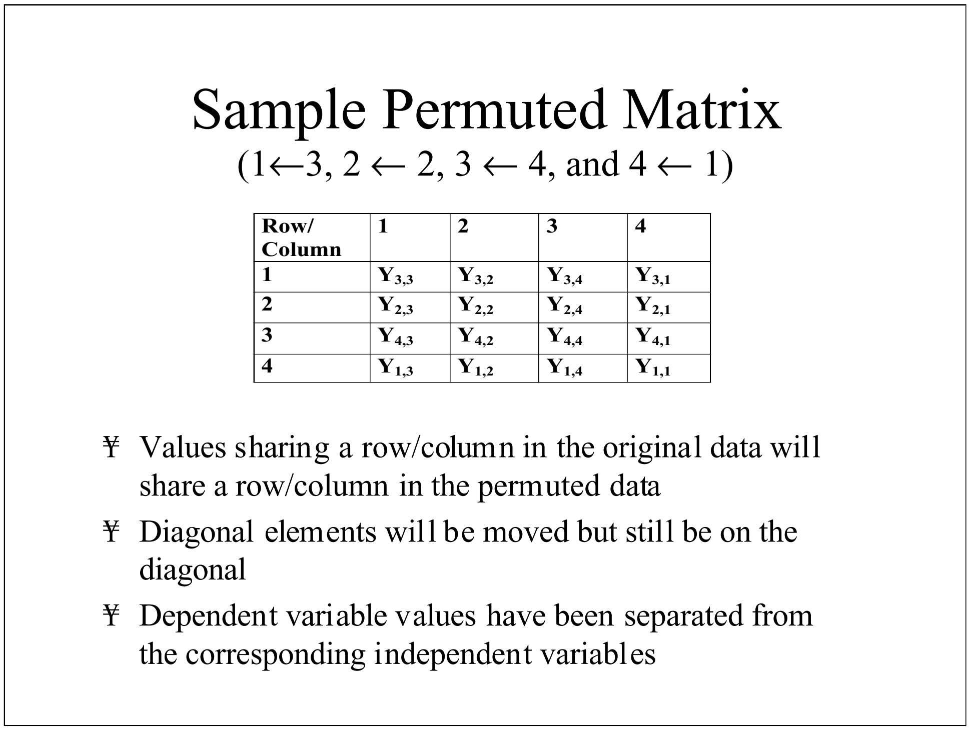
—Corresponds to the null hypothesis

—Preserves any row and column dependence of both dependent and independent variables

Here's how it works:

Suppose that we kept our data set of pairs, but we scrambled the data by randomly reassigning the friendship ratings to new observations, while keeping the age difference in the original observations. After scrambling, we would expect no relationship between age difference and friendship. That is, our new data set would correspond to the null hypothesis. If we ran the regression, we would get a point from the sampling distribution of the coefficient under the null hypothesis. If we repeated the scrambling a number of times, we would get an empirical sampling distribution, and we could then compare our actual coefficient with this empirical distribution, and if it were at an extreme high or low percentile, reject the null hypothesis.

Well, not quite! The problem is that when we scrambled the data, we separated the data values belonging to the rows and columns. E.g., person A's values are no longer together. So what QAP does is to permute the rows and columns of the matrix, and use the same permutation for the rows as for the columns.



Sample Permuted Matrix

(1←3, 2 ← 2, 3 ← 4, and 4 ← 1)

Row/ Column	**1**	**2**	**3**	**4**
1	$Y_{3,3}$	$Y_{3,2}$	$Y_{3,4}$	$Y_{3,1}$
2	$Y_{2,3}$	$Y_{2,2}$	$Y_{2,4}$	$Y_{2,1}$
3	$Y_{4,3}$	$Y_{4,2}$	$Y_{4,4}$	$Y_{4,1}$
4	$Y_{1,3}$	$Y_{1,2}$	$Y_{1,4}$	$Y_{1,1}$

¥ Values sharing a row/column in the original data will share a row/column in the permuted data

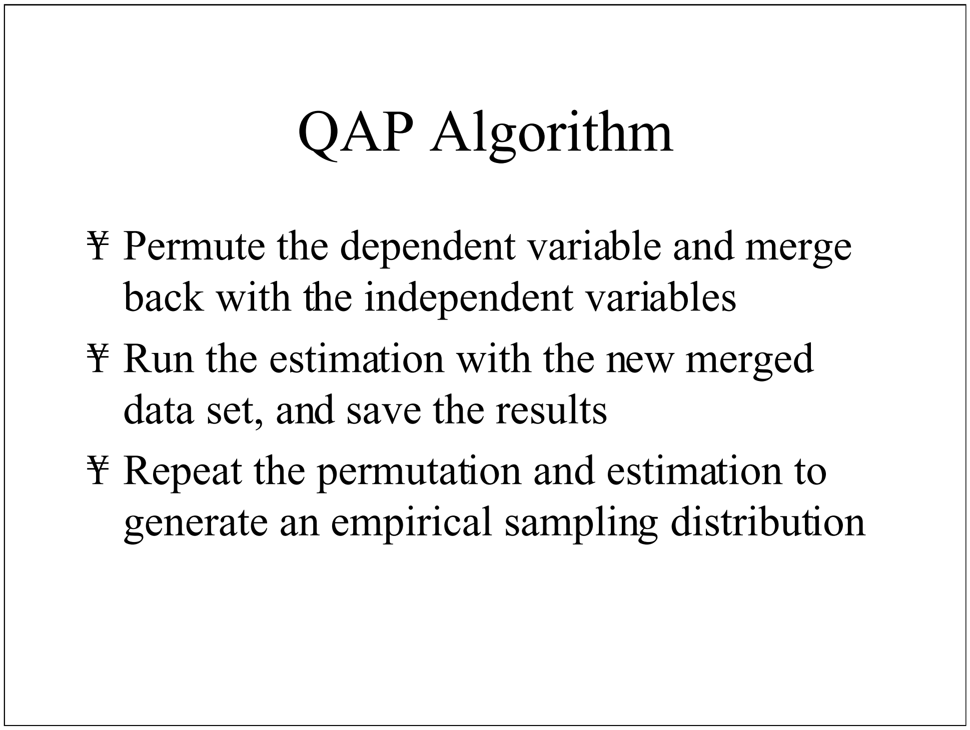
¥ Diagonal elements will be moved but still be on the diagonal

¥ Dependent variable values have been separated from the corresponding independent variables

For example, suppose that we have just 4 rows and columns, and suppose that we permuted them so that row and column 1 contain data originally in row/column 3, row and column 2 stay in place, row/column 3 comes from 4, and row/column 4 comes from 1. The permuted matrix is shown.

Notice that for each row, the first subscript is shared by all the entries in the row. For each column, the second subscript is shared. And each diagonal element moves to a new position but remains a diagonal element.

We have preserved any dependence among elements of the same row or column, but have eliminated any relationship between the dependent variable and the independent variable.



QAP Algorithm

¥ Permute the dependent variable and merge back with the independent variables

¥ Run the estimation with the new merged data set, and save the results

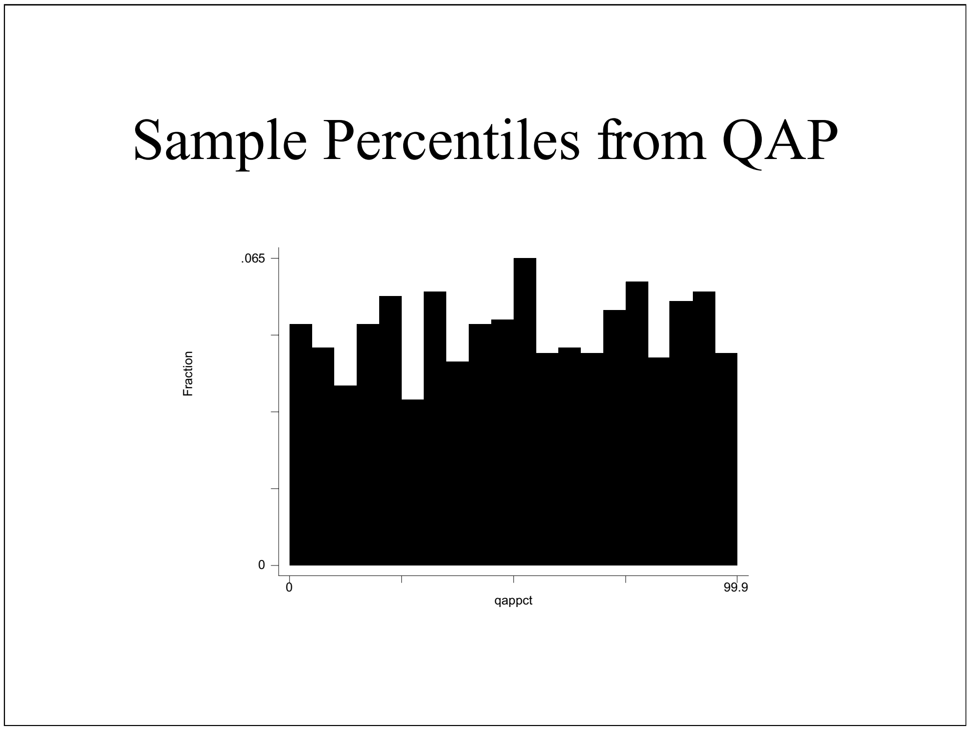
¥ Repeat the permutation and estimation to generate an empirical sampling distribution

The permuted data set corresponds to the null hypothesis. If we now run our estimation command, the coefficients and statistics will be values from the empirical sampling distribution under the null hypothesis, but the sampling distribution correctly takes into account the correlation among observations.

Now, if our original coefficient is at an extreme percentile of the distribution under the null, we can reject the null hypothesis.

Note that unlike the bootstrap, the empirical confidence interval is around the null, not around the sample value, and so this test corresponds specifically to the situation in classical hypothesis testing.

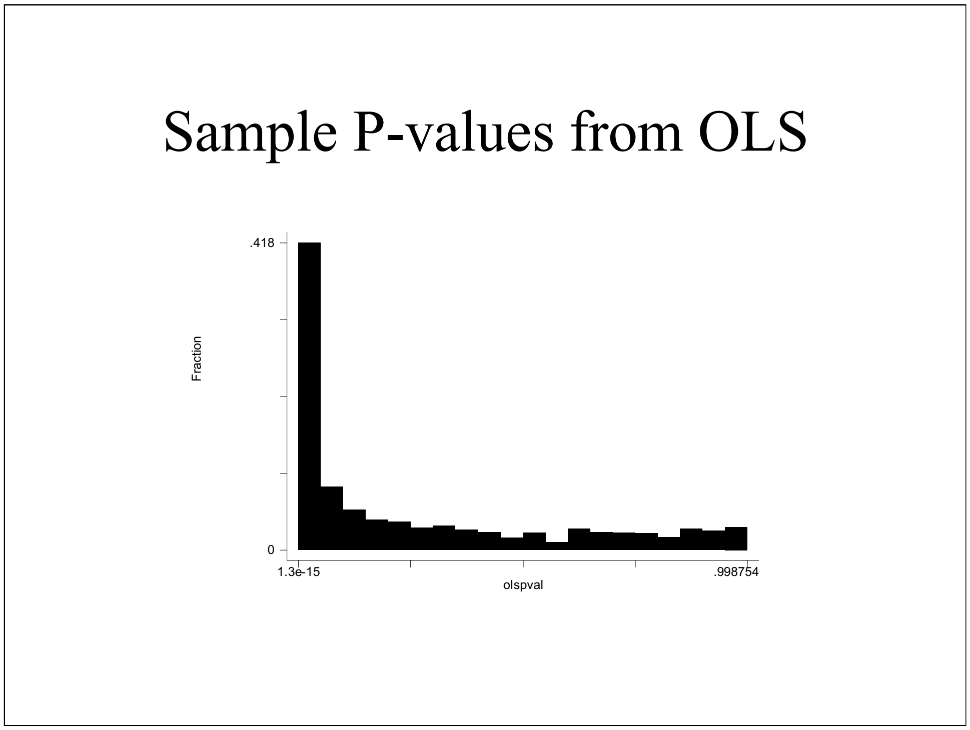
How well does this work? Here s an example. I generated some simulated data of a square matrix with 25 rows and columns, where there was no relationship between X and Y, but where there was a relatively high degree of correlation within rows and columns.



Sample Percentiles from QAP

Here's a graph of the percentiles as calculated by QAP.

Under the null hypothesis, you would expect a uniform distribution from 0 to 1, and clearly QAP is approximating this. This is within sampling error of a uniform distribution.

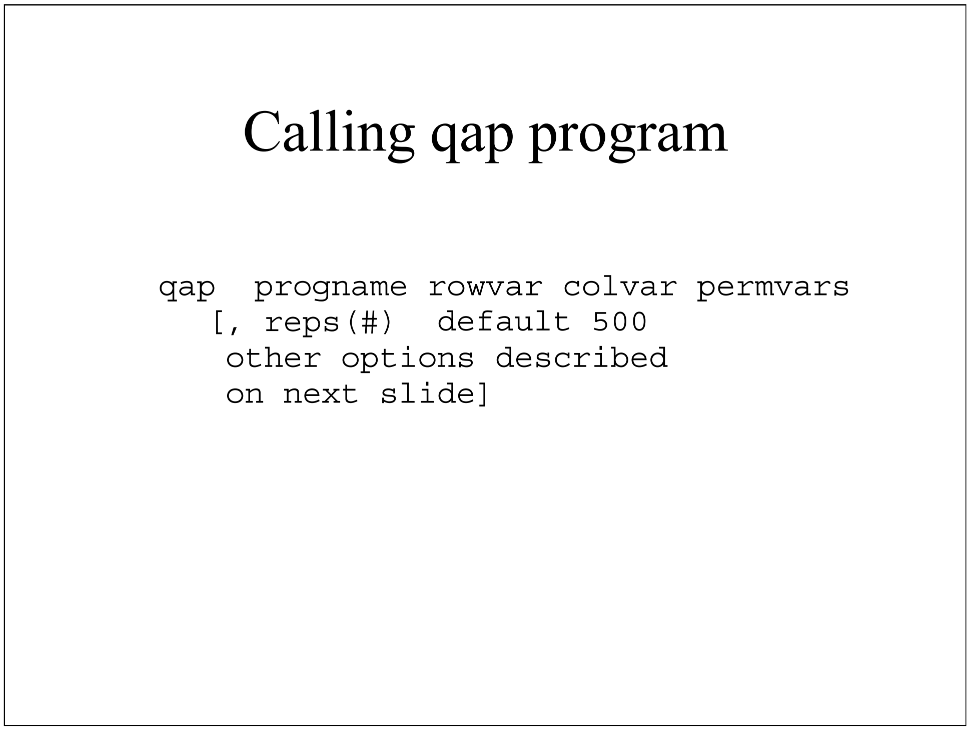


Sample P-values from OLS

Here’s the p-values from OLS.

About 42% of the p-values would be found significant at the .05 level.

:



Calling qap program

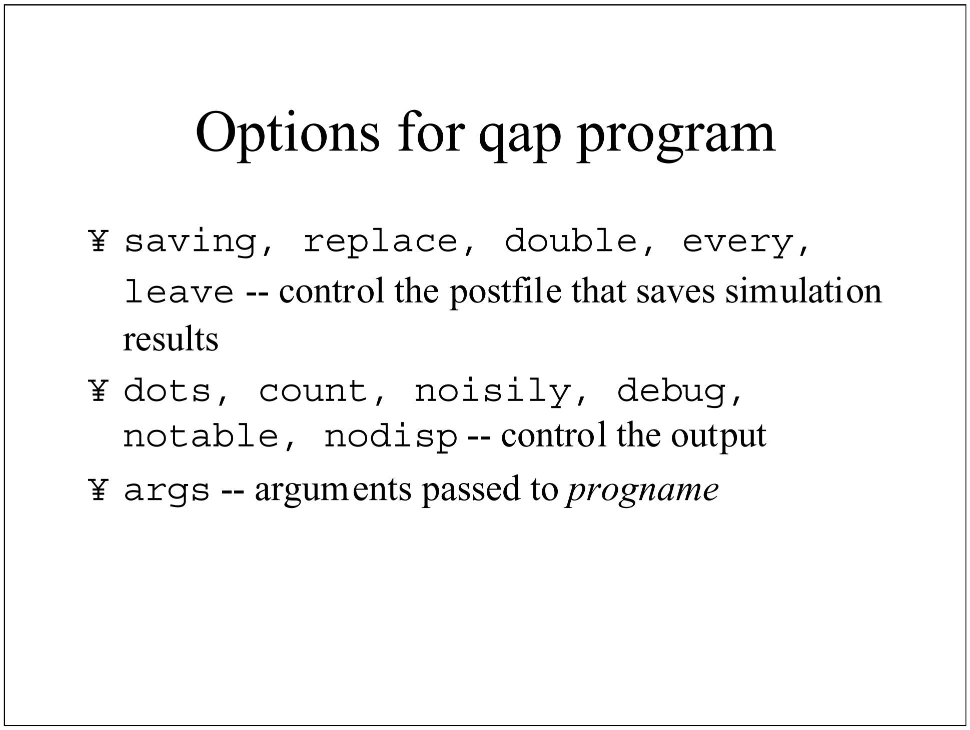
```
qap  progname rowvar colvar permvars
   [, reps(#)  default 500
    other options described
    on next slide]
```

The -qap- command implements the QAP procedure. It is modelled on the -bstrap- command and has many of the same options. The syntax for calling it is shown.

The QAP program generates QAP samples, calls progname for each sample, and generates a postfile of results. Progname should call an estimation command and save one or more statistics in a postfile. After reps iterations, QAP reads the postfile and displays results.

Rowvar and colvar are variable names for the row and column identifiers. Permvars is a list of variables that will be permuted in each QAP iteration. All the other variables will remain stationary. Most typically, permvars will just be the dependent variable. However, other variables can also be permuted. For example, if there are weights, they would normally be associated with the dependent variable and also permuted.

The program progname has to have the same structure as the program called by -bstrap- or -simul-. That is, it is called the first time with a question mark as a parameter, and has to set a global macro variable S_1 with the variable names that will be saved. For every other call, the first parameter is the identifier for the postfile, and the program must issue a post command with the values to be saved.



Options for qap program

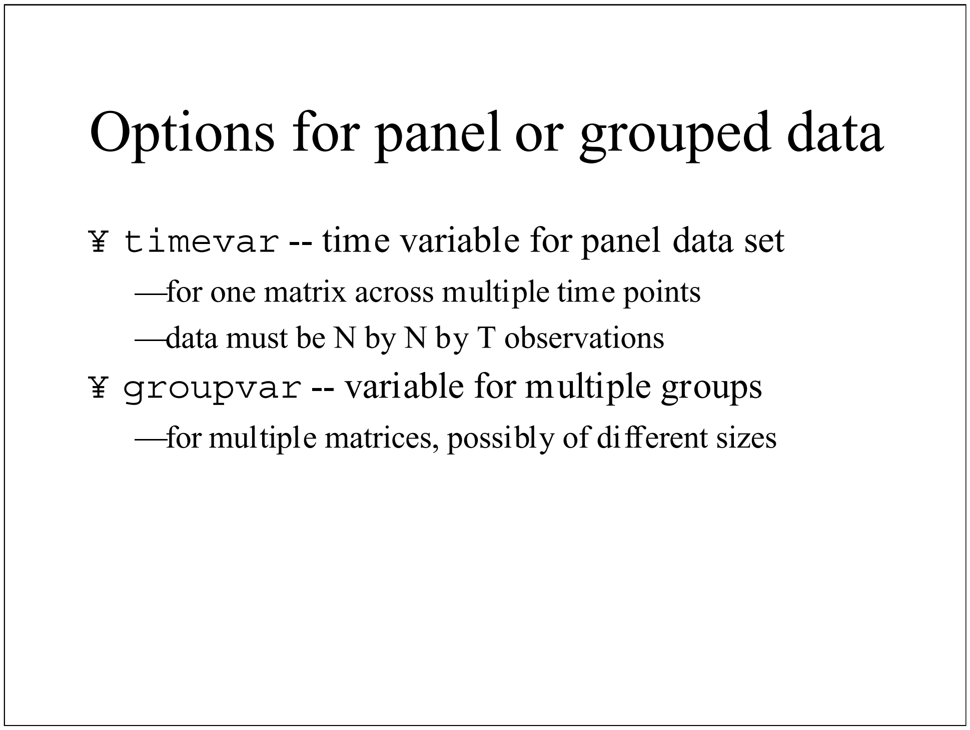
¥ `saving, replace, double, every, leave` -- control the postfile that saves simulation results

¥ `dots, count, noisily, debug, notable, nodisp` -- control the output

¥ `args` -- arguments passed to *progname*

Many of the options are the same as for -bstrap- and -simul-. Reps specifies the number of simulated samples. Double, every(), and replace are options for -postfile-. The saving option specifies a permanent file name for the QAP samples. If not given, a temporary file is used. If the leave option is used, the QAP results are left in memory, replacing the original data file. Dots displays a dot for each iteration. If the iterations are particularly slow, you can use the count option, which displays the iteration count for each QAP iteration. Noisily causes the estimation results to be displayed for each iteration, and debug generates a very large amount of debugging output.

Normally, after the iterations, the program displays the results of summarize, detail for each statistic, and then an output giving the value of the statistic for the actual data and the percentile in the QAP sample. The notable option suppresses the results of summarize, detail, and the nodisp option suppresses the display of actual data and percentile.



Options for panel or grouped data

¥ `timevar` -- time variable for panel data set

—for one matrix across multiple time points

—data must be N by N by T observations

¥ `groupvar` -- variable for multiple groups

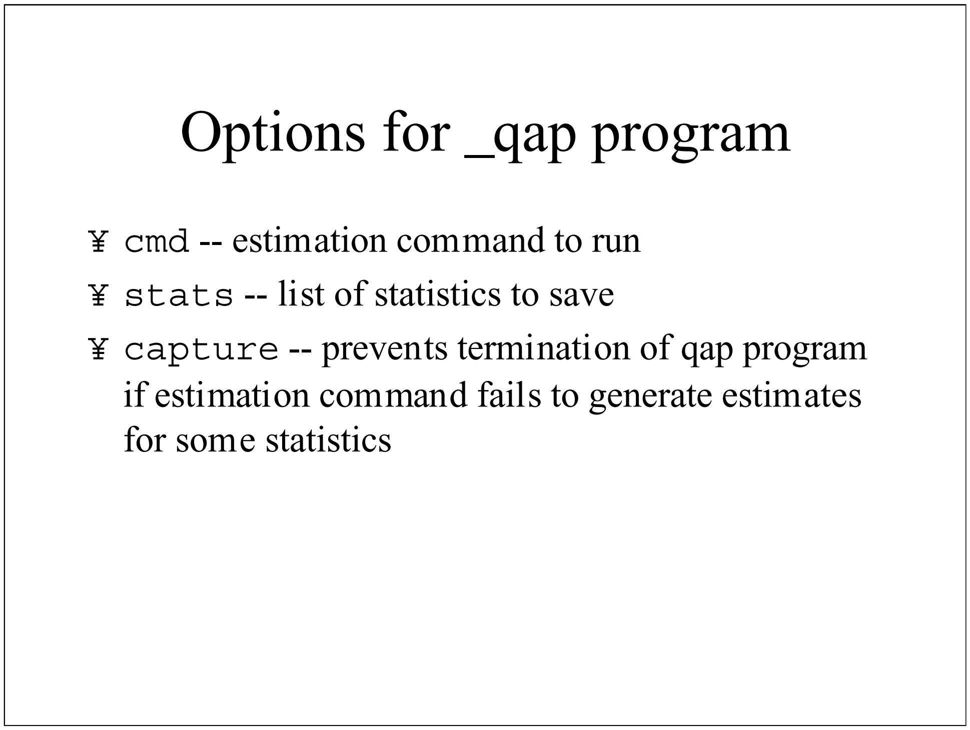
—for multiple matrices, possibly of different sizes

There are two options that are designed to work with more complex data sets.

The timevar option allows analysis of panel data sets, that is cross sectional time series data, where at each time period there is a dyadic matrix of observations. The structure of the data set must be N by N by T. The matrices at each time point must be the same size, and the number of time periods must be the same for each observation. However, missing values can be present in some observations. In this case, the QAP program uses the same permutation for the matrices at each time point, and it preserves the order in time.

The groupvr option allows analyzing multiple groups, where each group is a separate dyadic matrix, and the matrices can be of different sizes. In this case, each matrix is scrambled using a different permutation.

These add flexibility to the program, but should be used with caution, because there are potential pitfalls. For example, using timevar, any cross-sectional variation in the scrambled data corresponds to the null hypothesis, but any systematic relationship between X and Y across time will not be eliminated. So this should not be used for hypothesis testing for time-varying variables. Using groupvar, if there are systematic group differences, these will be preserved in the scrambled data, so hypotheses can not be tested for variables that change systematically by group.



Options for _qap program

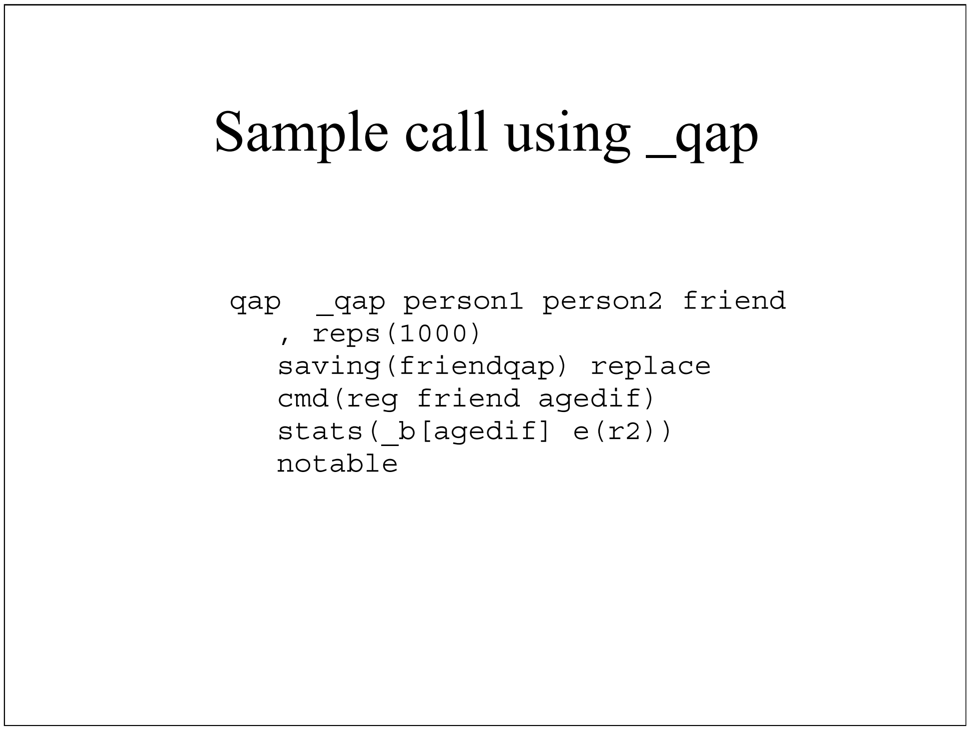
¥ `cmd` -- estimation command to run

¥ `stats` -- list of statistics to save

¥ `capture` -- prevents termination of qap program if estimation command fails to generate estimates for some statistics

For users who don’t want to write the program called by QAP, there is a built-in program, _qap, which can be specified for progname. The user specifies the estimation command using the cmd option and the statistics to save using the stats option, and _qap takes care of running the command and posting the results.

The _qap program can call any estimation command, and there may be occasions when the estimation command fails to generate all the statistics specified in the stats list. For example, if the estimation is a logistic regression, one independent variable may perfectly predict success or failure. If the dependent variable is mostly 1’s or mostly 0’s, then just by chance in the scrambled data set, an independent variable may end up a perfect predictor. If so, Stata will display a message and drop the variable and the affected cases. If after estimation, the _qap program attempts to access the corresponding coefficient, an error is generated. The capture option prevents the _qap program (and hence the entire QAP simulation) from failing due to this error. Instead, any unavailable coefficients are set to missing in the postfile. It is up to the user to decide how to interpret the QAP results if this happens. Note that if the estimation itself returns an error, or if a coefficient is not able to be calculated with the original data, then the QAP program exits.

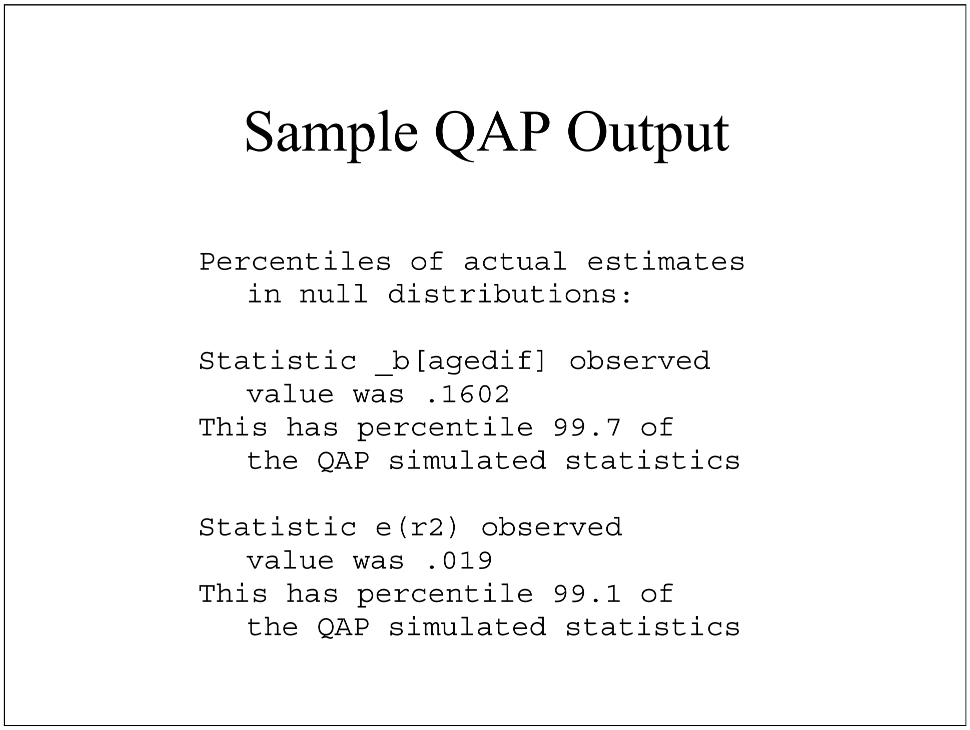


Sample call using _qap

```
qap  _qap person1 person2 friend
   , reps(1000)
   saving(friendqap) replace
   cmd(reg friend agedif)
   stats(_b[agedif] e(r2))
   notable
```

This is a call to qap to analyze a simulated data set from the problem we started with, looking at friendship as a function of age difference.

In this case, I simulated data where greater age difference predicted greater friendship, although with a small effect. The matrices were 25 by 25.



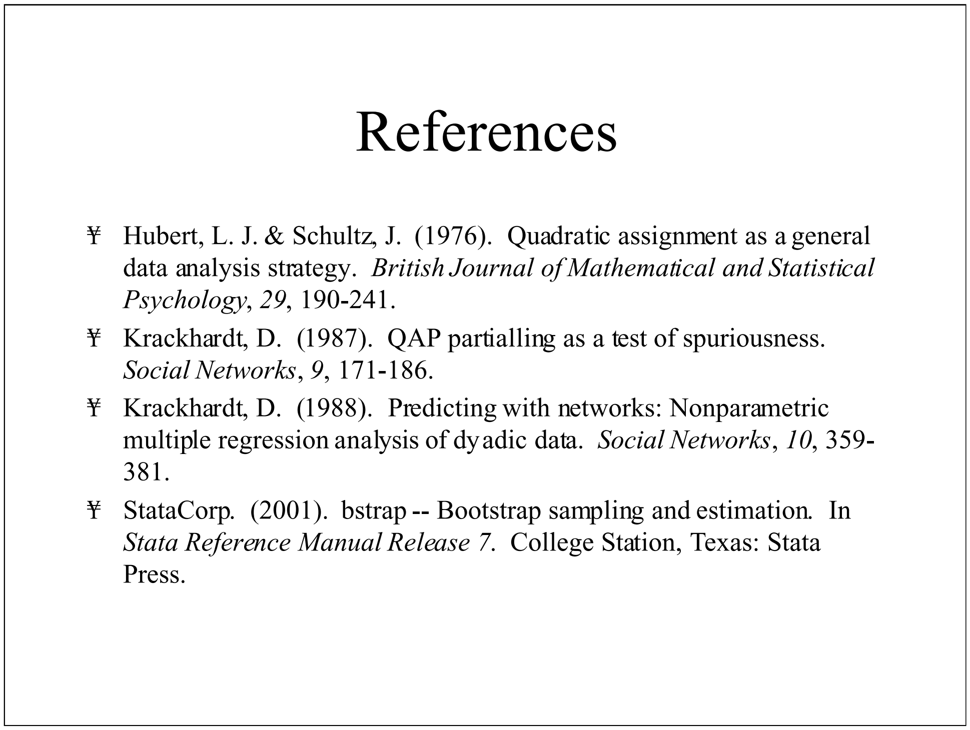
Sample QAP Output

```
Percentiles of actual estimates
   in null distributions:

Statistic _b[agedif] observed
   value was .1602
This has percentile 99.7 of
   the QAP simulated statistics

Statistic e(r2) observed
   value was .019
This has percentile 99.1 of
   the QAP simulated statistics
```

(By default, you get a summarize, detail for each statistic. I suppressed that with the notable option.)

The QAP output gives the observed value for each statistic in the original data set. It then reports the percentile of the simulated statistics where the observed value occurs. In this case, 99.7% of the simulated values were below the observed value, so this corresponds to a one-sided p-value of .003.

The R-squared statistic will be large if the coefficient is either highly positive or highly negative. So the 99.1% result corresponds to a two-sided p-value of .009.



References

¥ Hubert, L. J. & Schultz, J. (1976). Quadratic assignment as a general data analysis strategy. *British Journal of Mathematical and Statistical Psychology*, *29*, 190-241.

¥ Krackhardt, D. (1987). QAP partialling as a test of spuriousness. *Social Networks*, *9*, 171-186.

¥ Krackhardt, D. (1988). Predicting with networks: Nonparametric multiple regression analysis of dyadic data. *Social Networks*, *10*, 359-381.

¥ StataCorp. (2001). bstrap -- Bootstrap sampling and estimation. In *Stata Reference Manual Release 7*. College Station, Texas: Stata Press.

Many of the QAP options are the same as for -bstrap-, so you can use the -bstrap- reference to get more details.

The Hubert and Shultz reference shows the roots of the QAP procedure, but the QAP procedure as currently used in social network analysis and as implemented here is rather different.

The Krackhardt references give a clear description of the method and some simulation results.