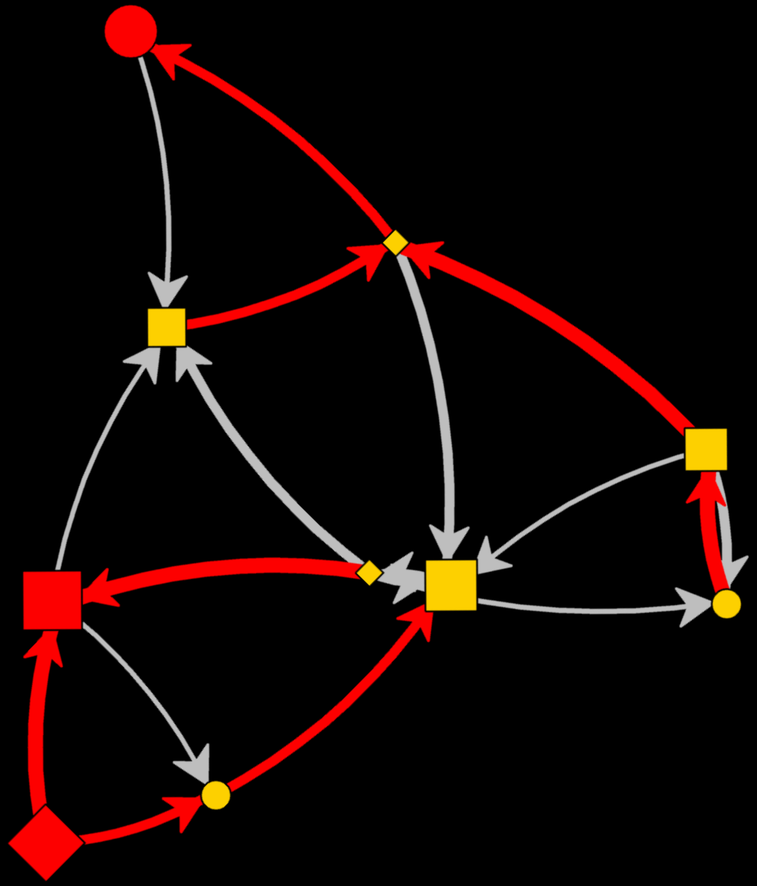
SOCIAL NETWORK ANALYSIS USING STATA



5 Sept 2014, Aarhus

Nordic and Baltic Stata Group Meeting

Thomas Grund
Linköping University

thomas.u.grund@gmail.com
www.grund.co.uk

(with contributions from Peter Hedström, Yvonne Aberg, Lorien Jasny)

http://nwcommands.org

BETA

TESTING

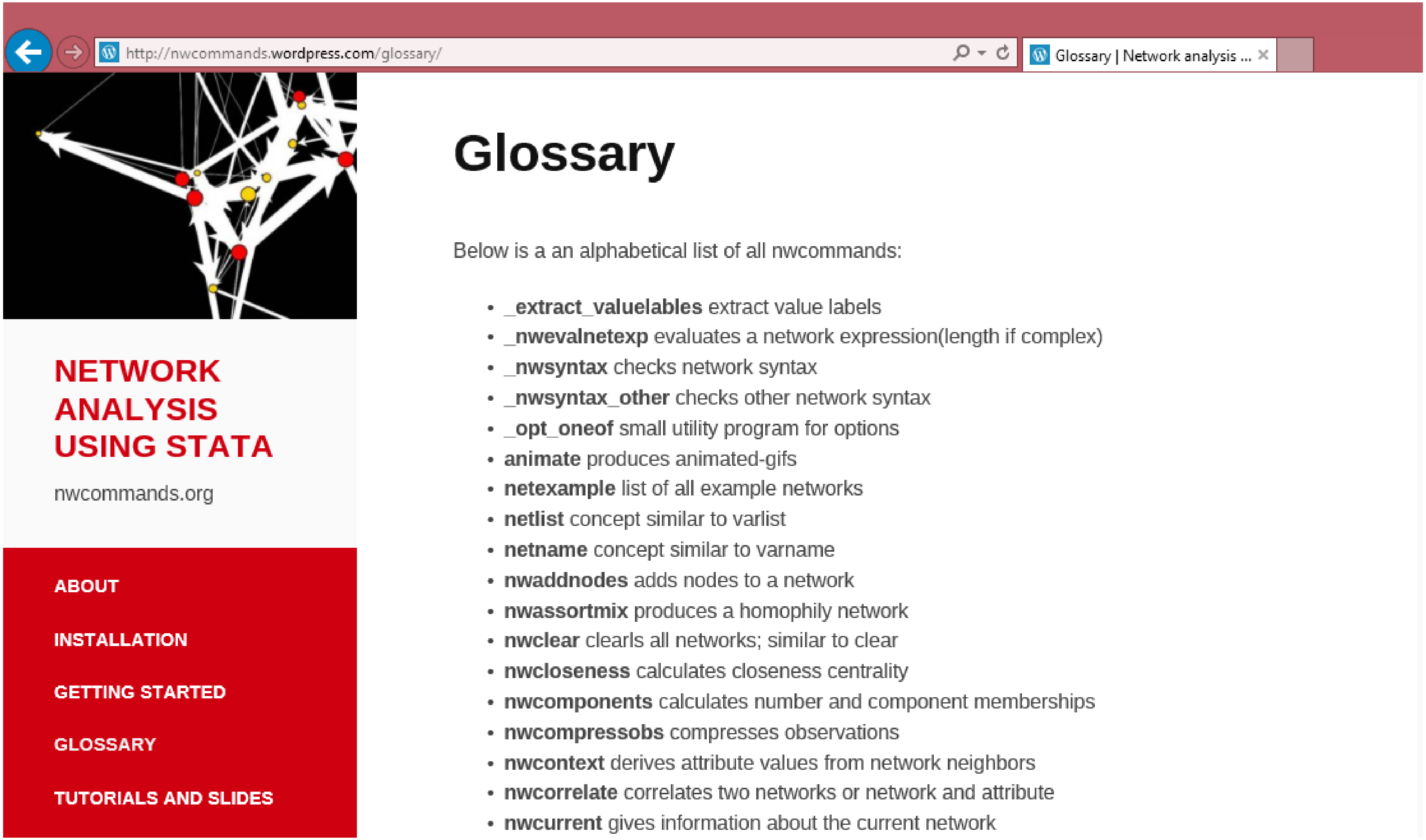
BETA

Email list

https://groups.google.com/forum/#!forum/nwcommands/join

http://nwcommands.org

BETA TESTING BETA


http://nwcommands.wordpress.com/glossary/
Glossary | Network analysis ...

NETWORK ANALYSIS USING STATA

nwcommands.org

ABOUT

INSTALLATION

GETTING STARTED

GLOSSARY

TUTORIALS AND SLIDES

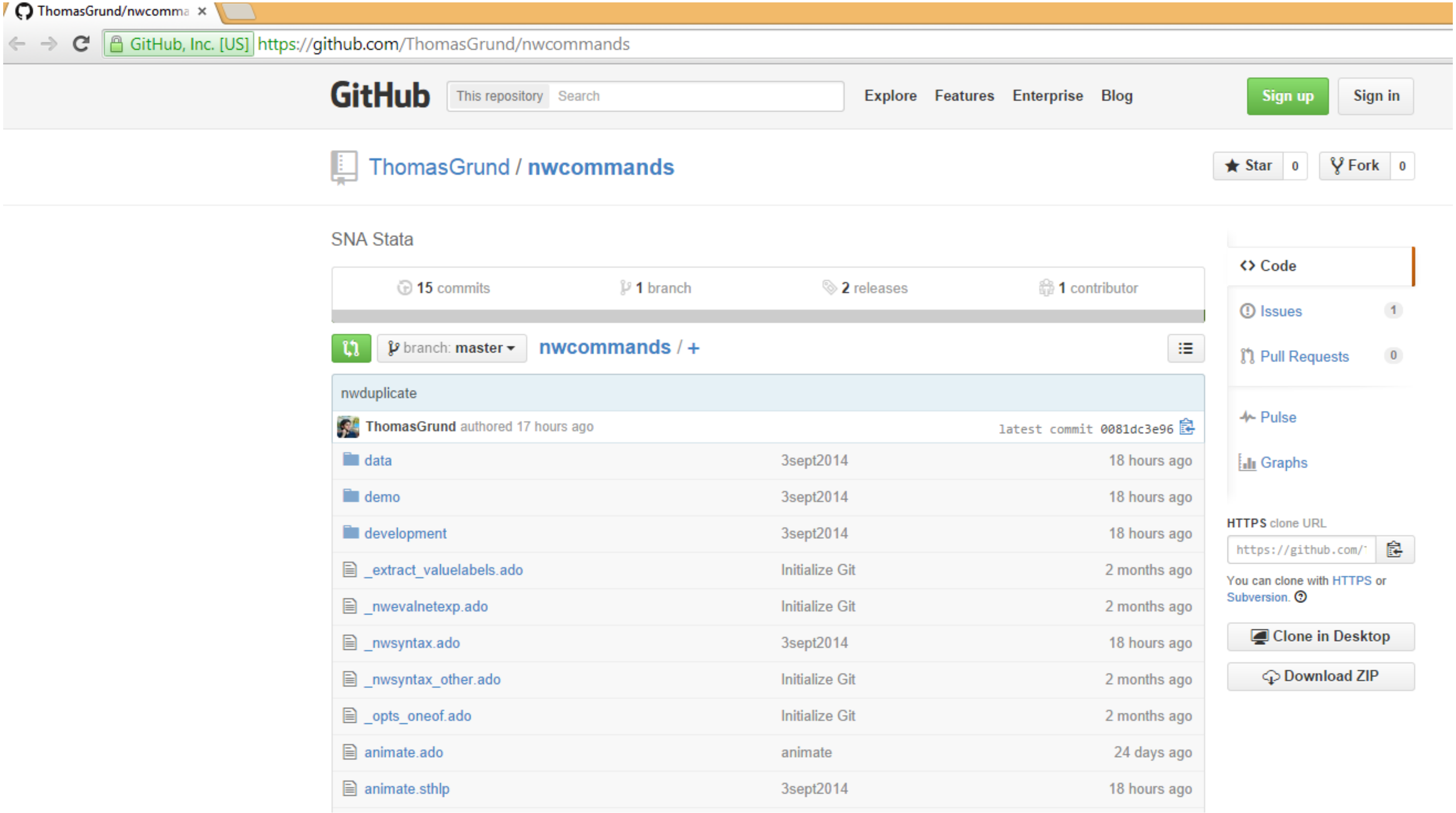
Glossary

Below is a an alphabetical list of all nwcommands:

- **_extract_valuelables** extract value labels
- **_nwevalnetexp** evaluates a network expression(length if complex)
- **_nwsyntax** checks network syntax
- **_nwsyntax_other** checks other network syntax
- **_opt_oneof** small utility program for options
- **animate** produces animated-gifs
- **netexample** list of all example networks
- **netlist** concept similar to varlist
- **netname** concept similar to varname
- **nwaddnodes** adds nodes to a network
- **nwassortmix** produces a homophily network
- **nwclear** clearls all networks; similar to clear
- **nwcloseness** calculates closeness centrality
- **nwcomponents** calculates number and component memberships
- **nwcompressobs** compresses observations
- **nwcontext** derives attribute values from network neighbors
- **nwcorrelate** correlates two networks or network and attribute
- **nwcurrent** gives information about the current network

GitHub

https://github.com/ThomasGrund/nwcommands



ThomasGrund / nwcommands

SNA Stata

15 commits · 1 branch · 2 releases · 1 contributor

branch: master · nwcommands / +

nwduplicate

ThomasGrund authored 17 hours ago · latest commit 0081dc3e96

data	3sept2014	18 hours ago
demo	3sept2014	18 hours ago
development	3sept2014	18 hours ago
_extract_valuelabels.ado	Initialize Git	2 months ago
_nwevalnetexp.ado	Initialize Git	2 months ago
_nwsyntax.ado	3sept2014	18 hours ago
_nwsyntax_other.ado	Initialize Git	2 months ago
_opts_oneof.ado	Initialize Git	2 months ago
animate.ado	animate	24 days ago
animate.sthlp	3sept2014	18 hours ago

HTTPS clone URL

https://github.com/

You can clone with HTTPS or Subversion.

Clone in Desktop

Download ZIP

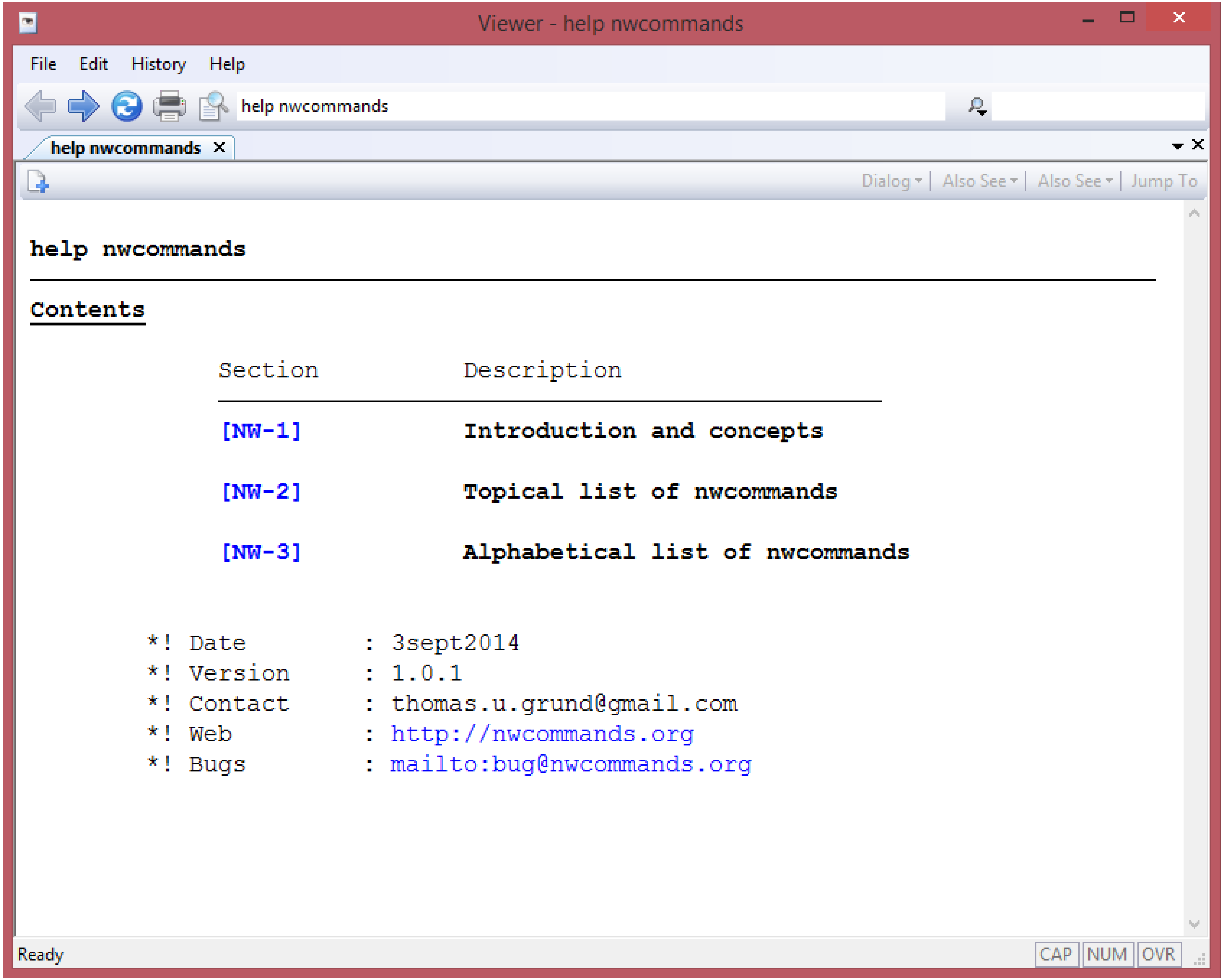
http://nwcommands.org

BETA

TESTING

BETA

Getting started

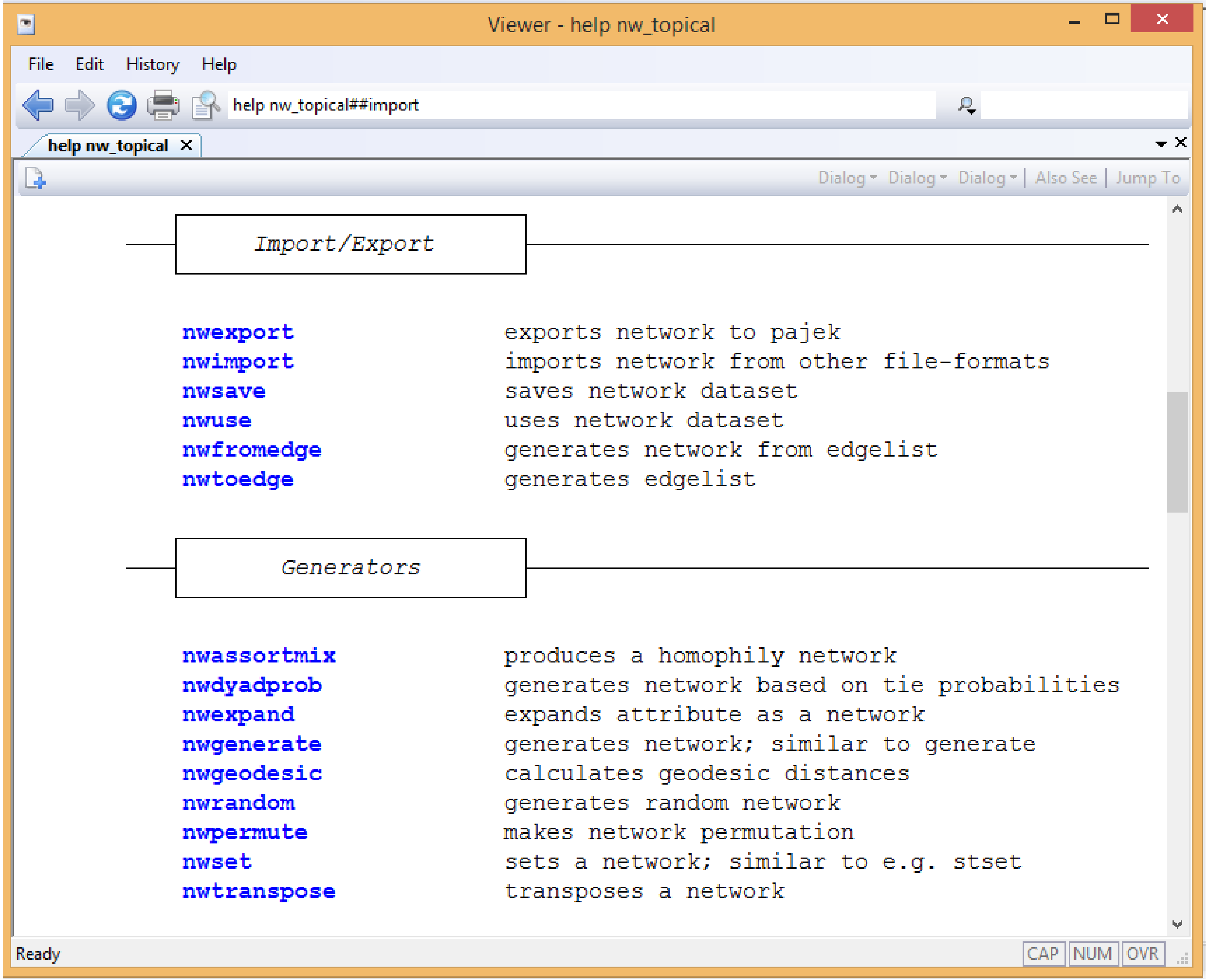


help nwcommands

Contents

Section	Description
[NW-1]	**Introduction and concepts**
[NW-2]	**Topical list of nwcommands**
[NW-3]	**Alphabetical list of nwcommands**

```
*! Date          : 3sept2014
*! Version       : 1.0.1
*! Contact       : thomas.u.grund@gmail.com
*! Web           : http://nwcommands.org
*! Bugs          : mailto:bug@nwcommands.org
```



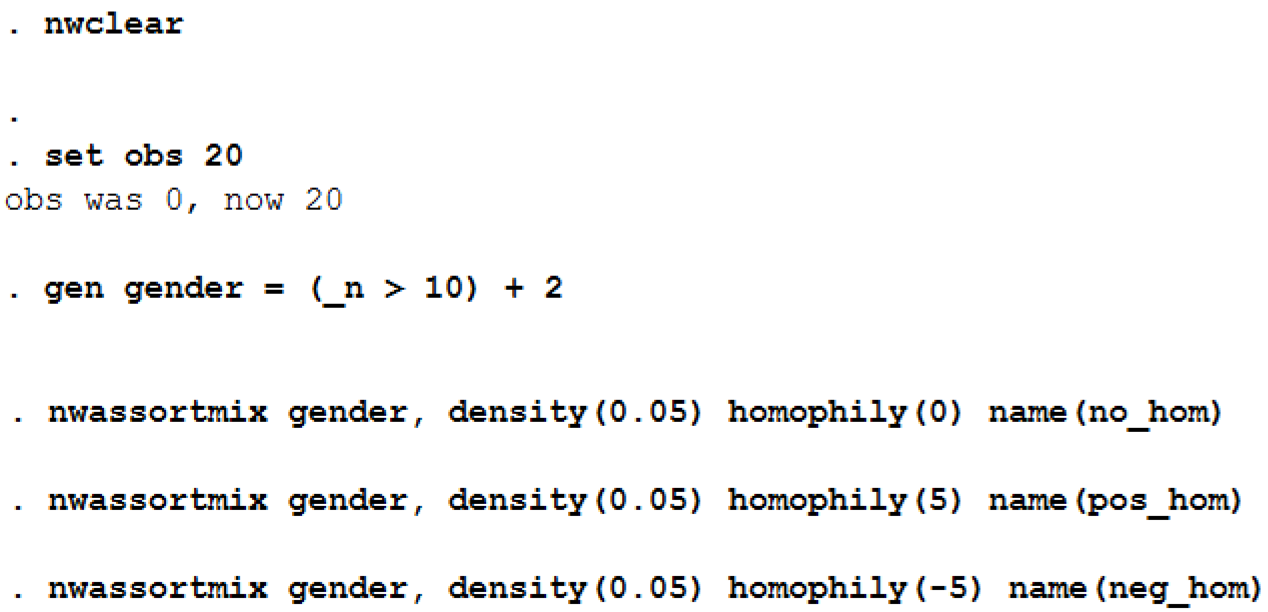
Import/Export

nwexport	exports network to pajek
nwimport	imports network from other file-formats
nwsave	saves network dataset
nwuse	uses network dataset
nwfromedge	generates network from edgelist
nwtoedge	generates edgelist

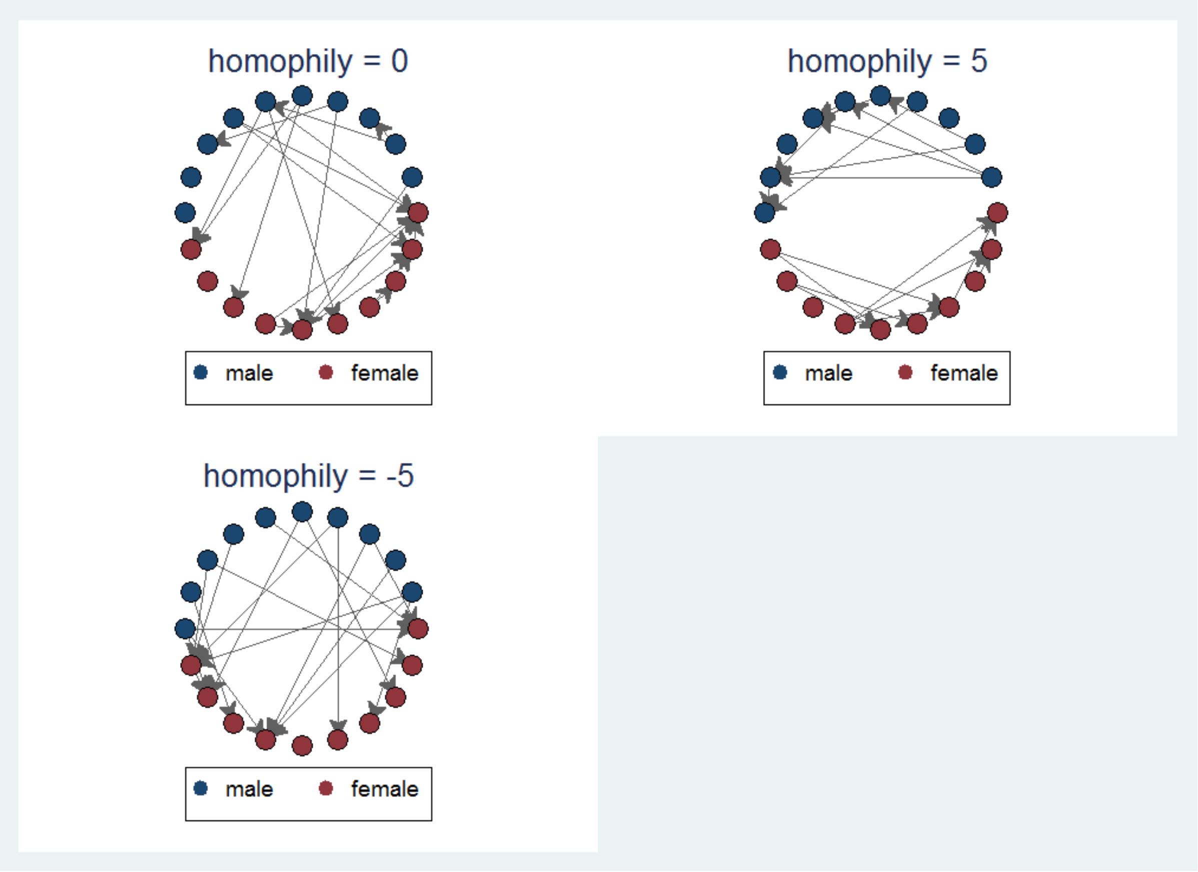
Generators

nwassortmix	produces a homophily network
nwdyadprob	generates network based on tie probabilities
nwexpand	expands attribute as a network
nwgenerate	generates network; similar to generate
nwgeodesic	calculates geodesic distances
nwrandom	generates random network
nwpermute	makes network permutation
nwset	sets a network; similar to e.g. stset
nwtranspose	transposes a network

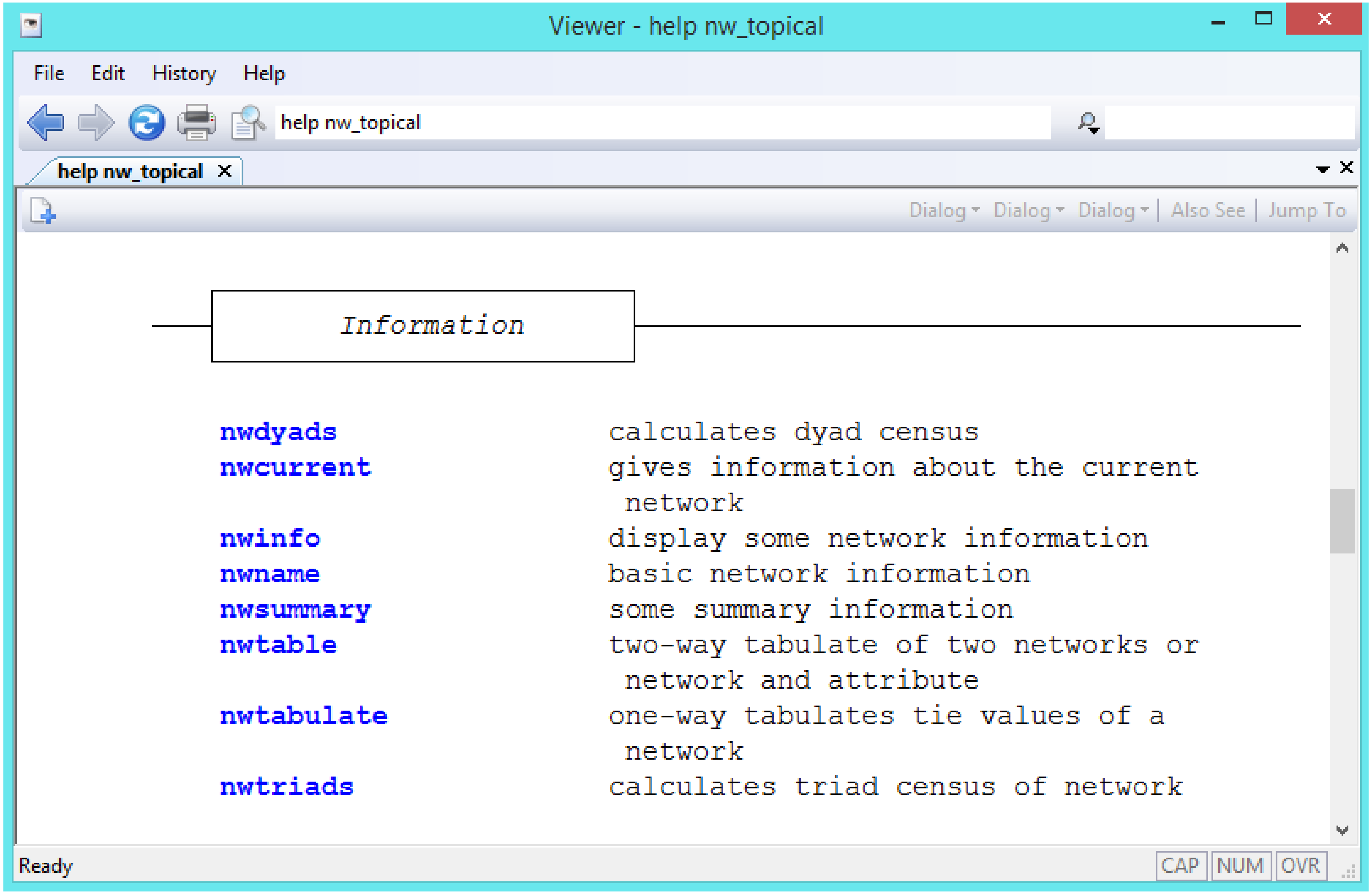
```
. nwuse glasgow

Loading successful
(3 networks)

 1) Stored Network

   Network name: glasgow2
   Directed: true
   Nodes: 50


 2) Stored Network

   Network name: glasgow1
   Directed: true
   Nodes: 50


 3) Current Network

   Network name: glasgow3
   Directed: true
   Nodes: 50

.
```



```
. nwclear

.
. set obs 20
obs was 0, now 20

. gen gender = (_n > 10) + 2


. nwassortmix gender, density(0.05) homophily(0) name(no_hom)

. nwassortmix gender, density(0.05) homophily(5) name(pos_hom)

. nwassortmix gender, density(0.05) homophily(-5) name(neg_hom)
. nwplot no_hom, color(gender) layout(circle) title("homophily = 0") saving(g1)

. nwplot pos_hom, color(gender) layout(circle) title("homophily = 5") saving(g2)

. nwplot neg_hom, color(gender) layout(circle) title("homophily = -5") saving(g3)

. graph combine g1.gph g2.gph g3.gph
```

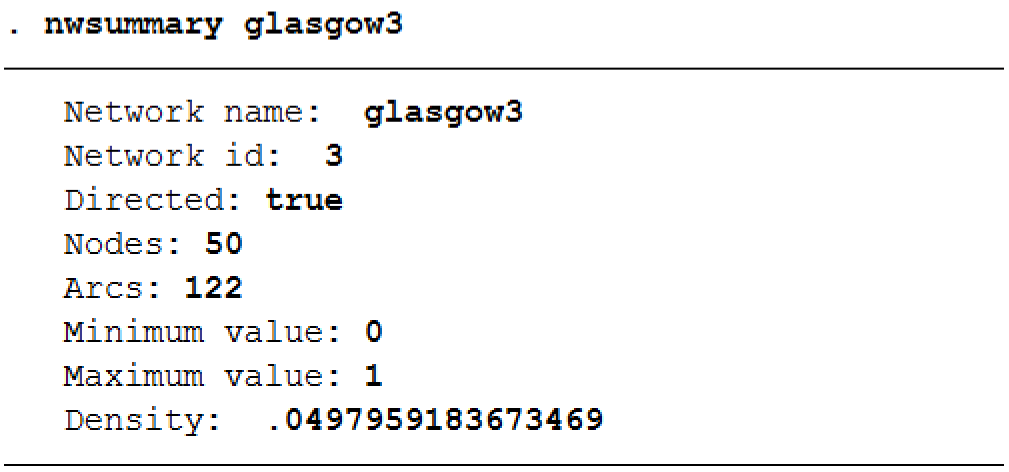


Describe networks

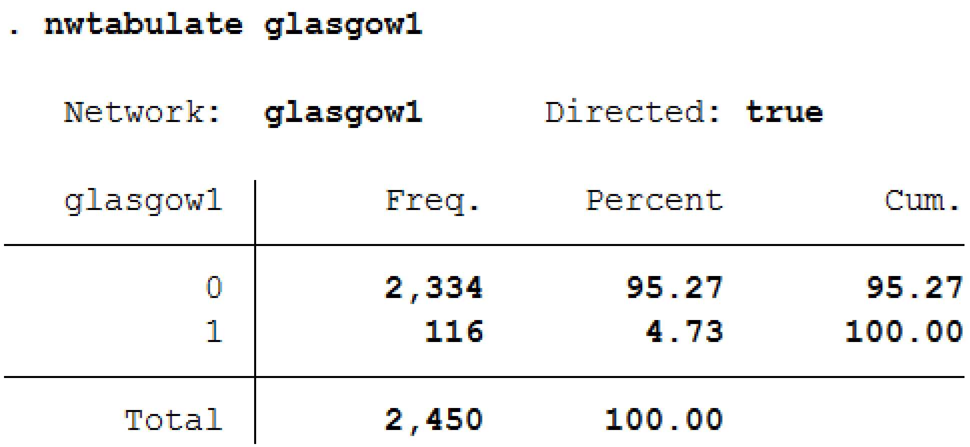


Information

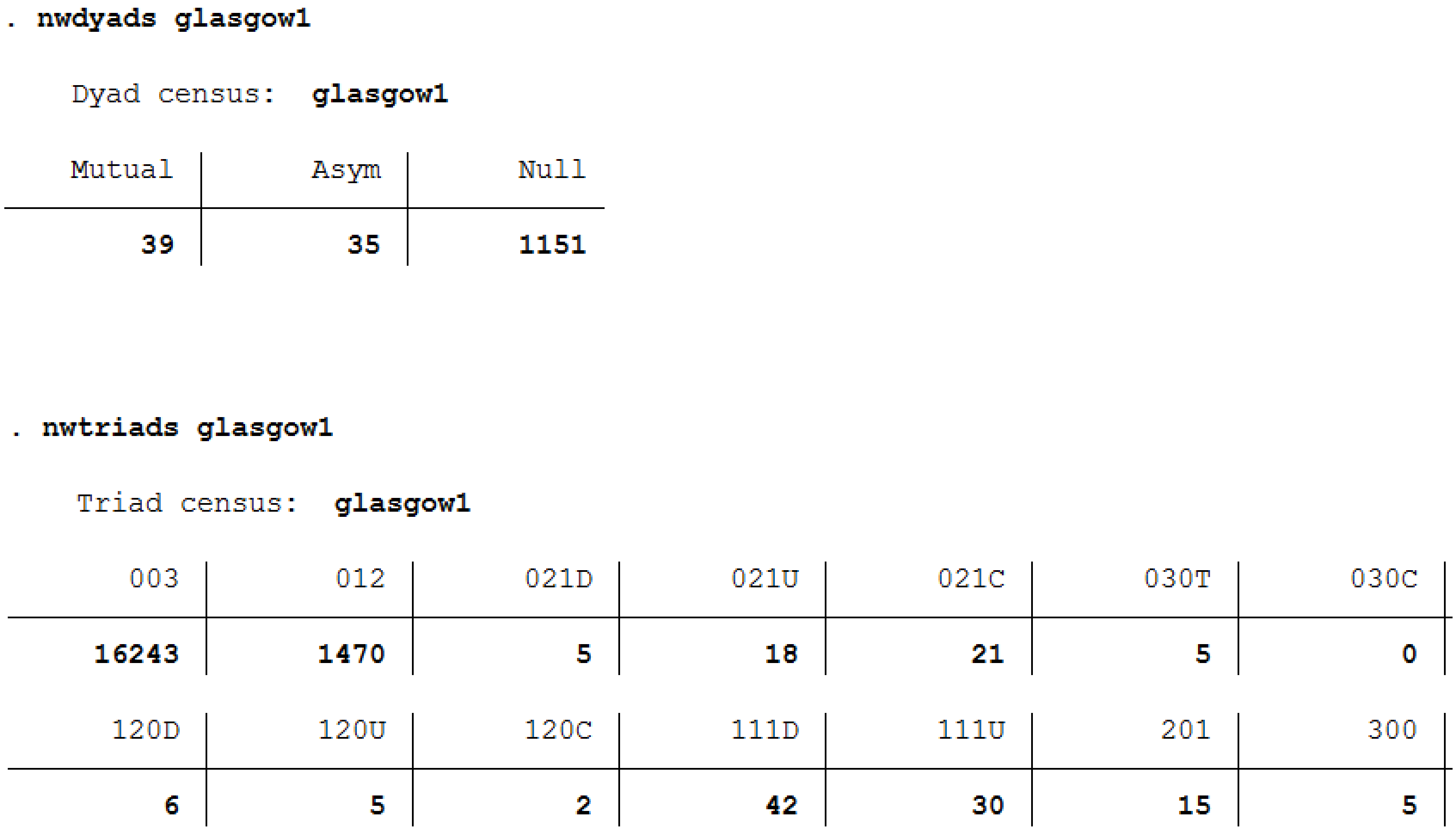
```
nwdyads                 calculates dyad census
nwcurrent               gives information about the current
                         network
nwinfo                  display some network information
nwname                  basic network information
nwsummary               some summary information
nwtable                 two-way tabulate of two networks or
                         network and attribute
nwtabulate              one-way tabulates tie values of a
                         network
nwtriads                calculates triad census of network
```



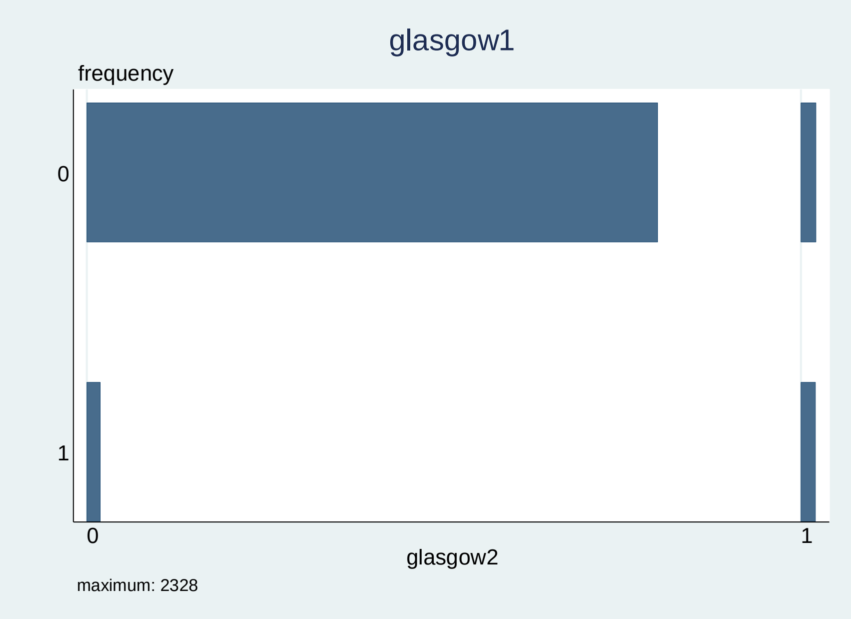
```
. nwsummary glasgow3
--------------------------------------------------------
   Network name:  glasgow3
   Network id:  3
   Directed: true
   Nodes: 50
   Arcs: 122
   Minimum value: 0
   Maximum value: 1
   Density:  .0497959183673469
--------------------------------------------------------
```



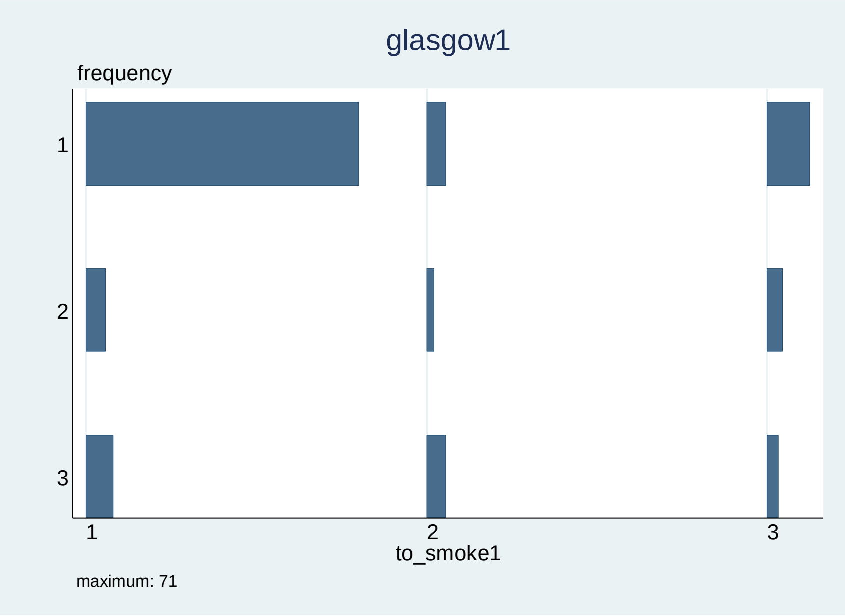
```
. nwtabulate glasgow1

   Network:  glasgow1      Directed: true

   glasgow1 |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |      2,334       95.27       95.27
          1 |        116        4.73      100.00
------------+-----------------------------------
      Total |      2,450      100.00
```



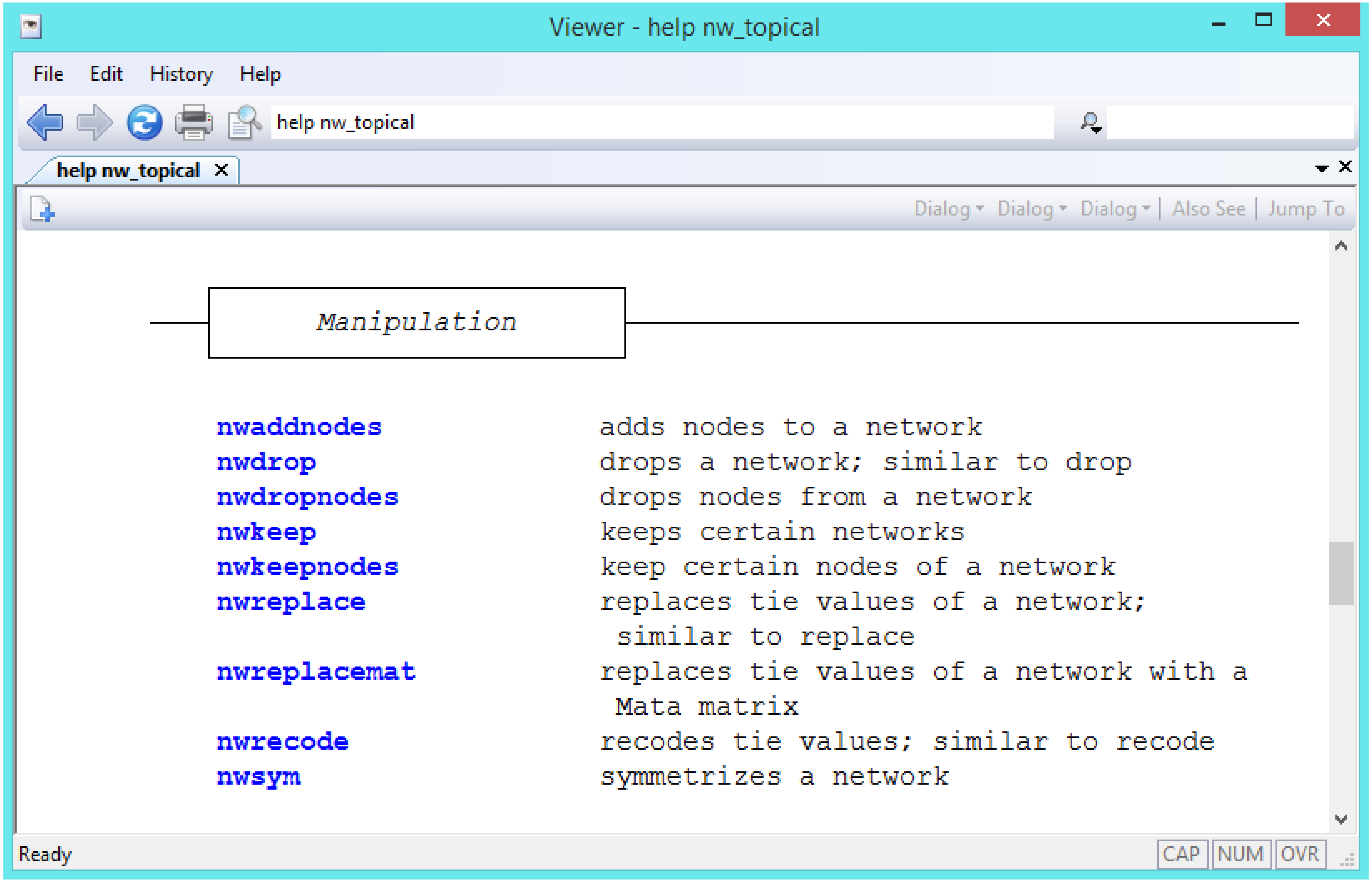
```
. nwdyads glasgow1

   Dyad census:  glasgow1

     Mutual |       Asym |       Null
 -----------+------------+-----------
         39 |         35 |       1151


. nwtriads glasgow1

   Triad census:  glasgow1

        003 |        012 |       021D |       021U |       021C |       030T |       030C |
 -----------+------------+------------+------------+------------+------------+------------+
      16243 |       1470 |          5 |         18 |         21 |          5 |          0 |
       120D |       120U |       120C |       111D |       111U |        201 |        300 |
 -----------+------------+------------+------------+------------+------------+------------+
          6 |          5 |          2 |         42 |         30 |         15 |          5 |
. nwtable glasgow1 glasgow2, plot

   Network 1:  glasgow1    Directed: true
   Network 2:  glasgow2    Directed: true


           │       glasgow2
  glasgow1 │         0          1 │     Total
───────────┼──────────────────────┼──────────
         0 │     2,278         59 │     2,337
         1 │        56         57 │       113
───────────┼──────────────────────┼──────────
     Total │     2,334        116 │     2,450
```



```
. nwtable glasgow1 smoke1, plot

   Network:   glasgow1       Directed: true
   Attribute:   smoke1


from_smoke |            to_smoke1
         1 |         1          2          3 |     Total
-----------+---------------------------------+----------
         1 |        71          5         11 |        87
         2 |         5          2          4 |        11
         3 |         7          5          3 |        15
-----------+---------------------------------+----------
     Total |        83         12         18 |       113
```



Manipulate networks

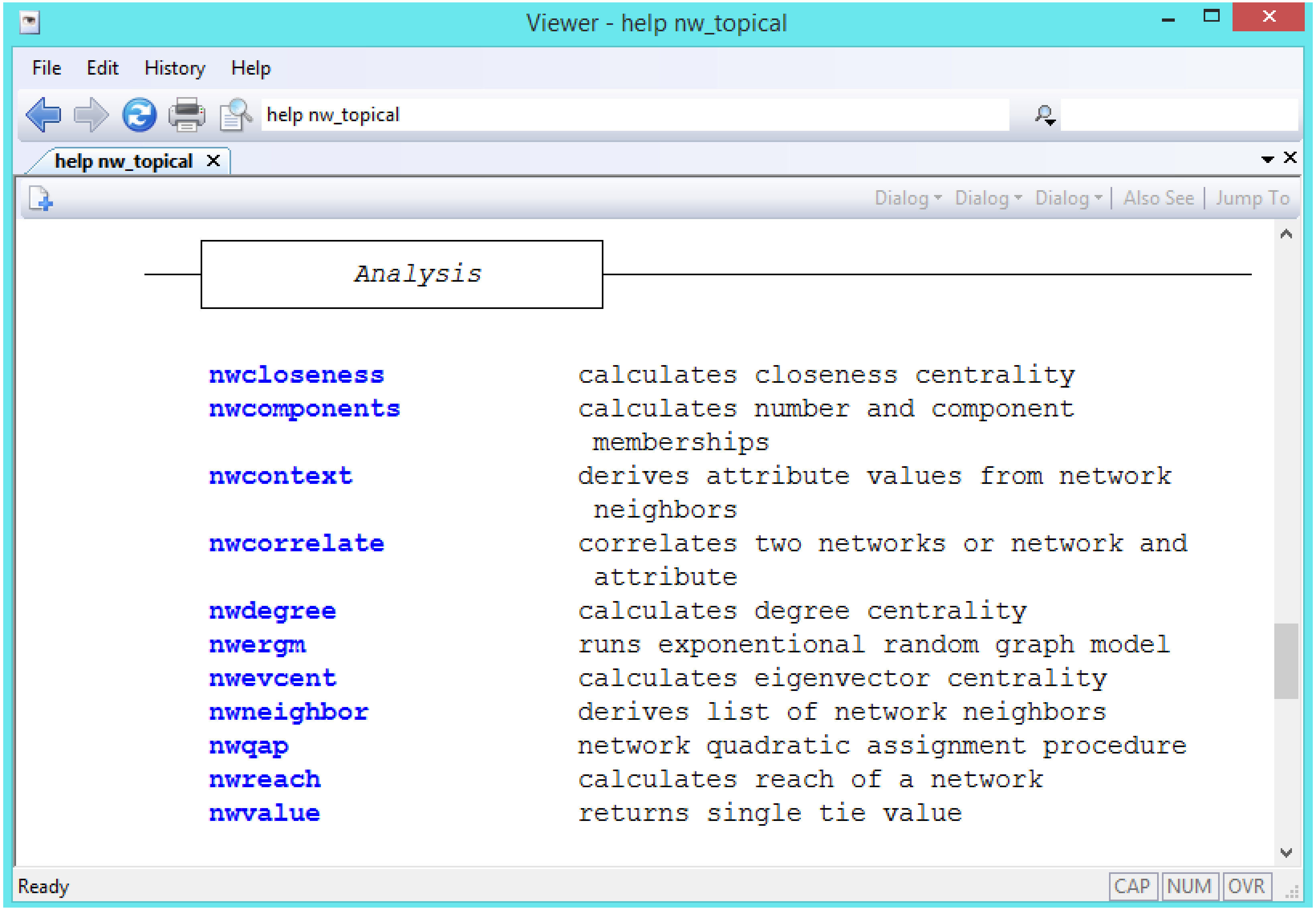


Manipulation

nwaddnodes	adds nodes to a network
nwdrop	drops a network; similar to drop
nwdropnodes	drops nodes from a network
nwkeep	keeps certain networks
nwkeepnodes	keep certain nodes of a network
nwreplace	replaces tie values of a network; similar to replace
nwreplacemat	replaces tie values of a network with a Mata matrix
nwrecode	recodes tie values; similar to recode
nwsym	symmetrizes a network

```
. nwclear

. nwrandom 7, density(.2) name(first)
. nwrandom 7, density(.3) name(second)
. nwrandom 7, density(.3) name(third)
. gen attr= _n * 2


// replacing networks
. nwreplace first = 1
. nwreplace first = 2 in 3/5
. nwreplace first = exp(second) * attr if first == 1


// replacing subnetworks
. nwreplace first[(2::6),(1::5)] = 55
. nwreplace first[(1::4),(1::4)] = second * 7 if third != 1


// replacing with temporary networks
. nwreplace first =  99 * (_nwrandom 7, prob(.3))
```

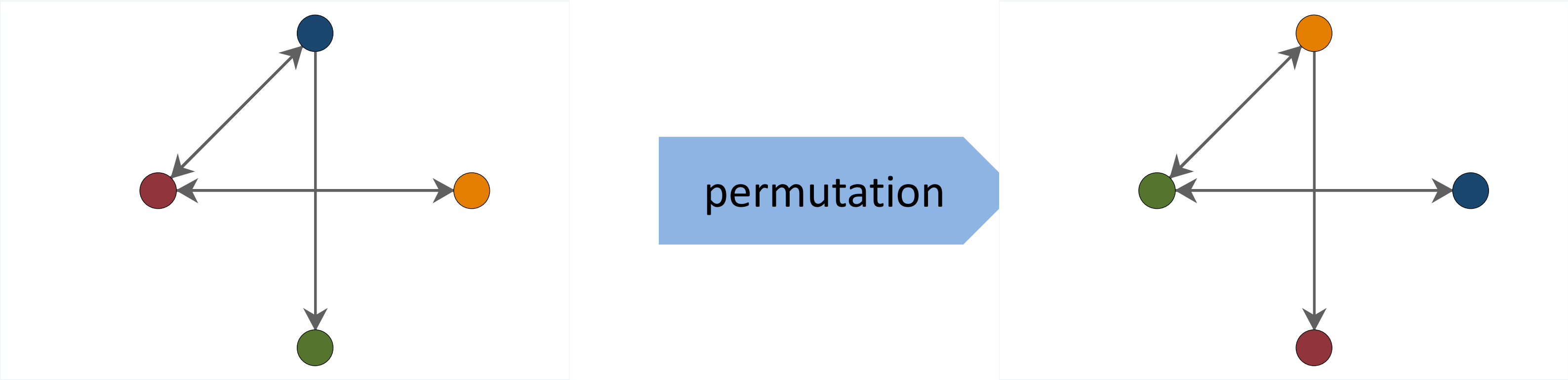
Analyze networks



Analysis

nwcloseness	calculates closeness centrality
nwcomponents	calculates number and component memberships
nwcontext	derives attribute values from network neighbors
nwcorrelate	correlates two networks or network and attribute
nwdegree	calculates degree centrality
nwergm	runs exponentional random graph model
nwevcent	calculates eigenvector centrality
nwneighbor	derives list of network neighbors
nwqap	network quadratic assignment procedure
nwreach	calculates reach of a network
nwvalue	returns single tie value

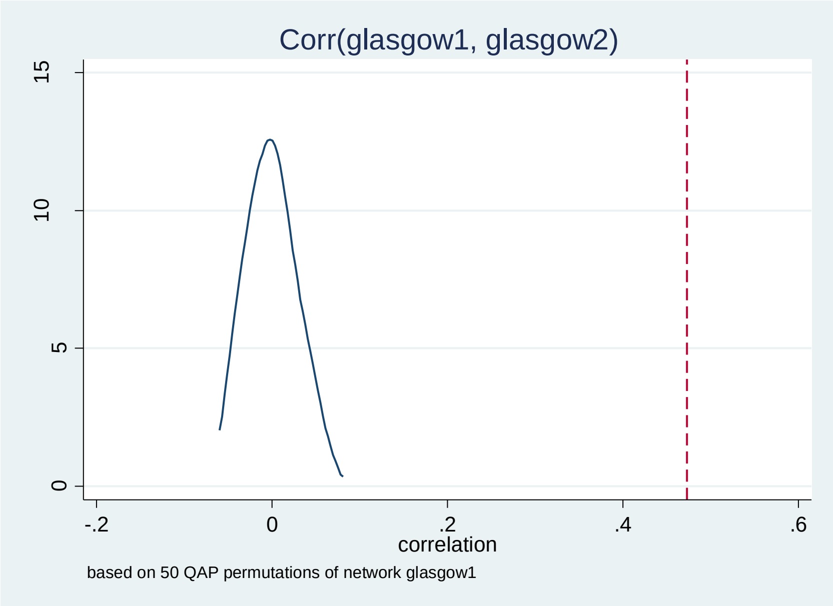
Network permutation



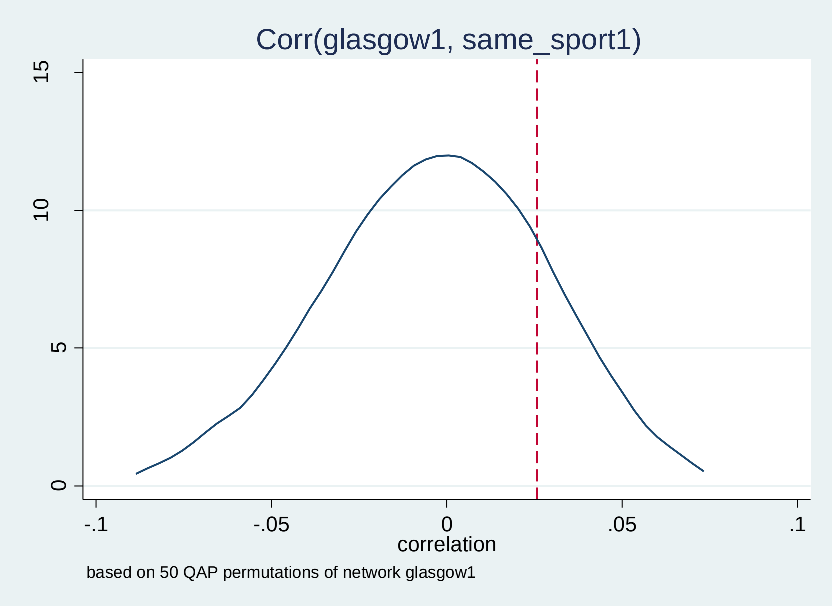
-	1	1	0
1	-	0	1
1	0	-	0
0	0	0	-

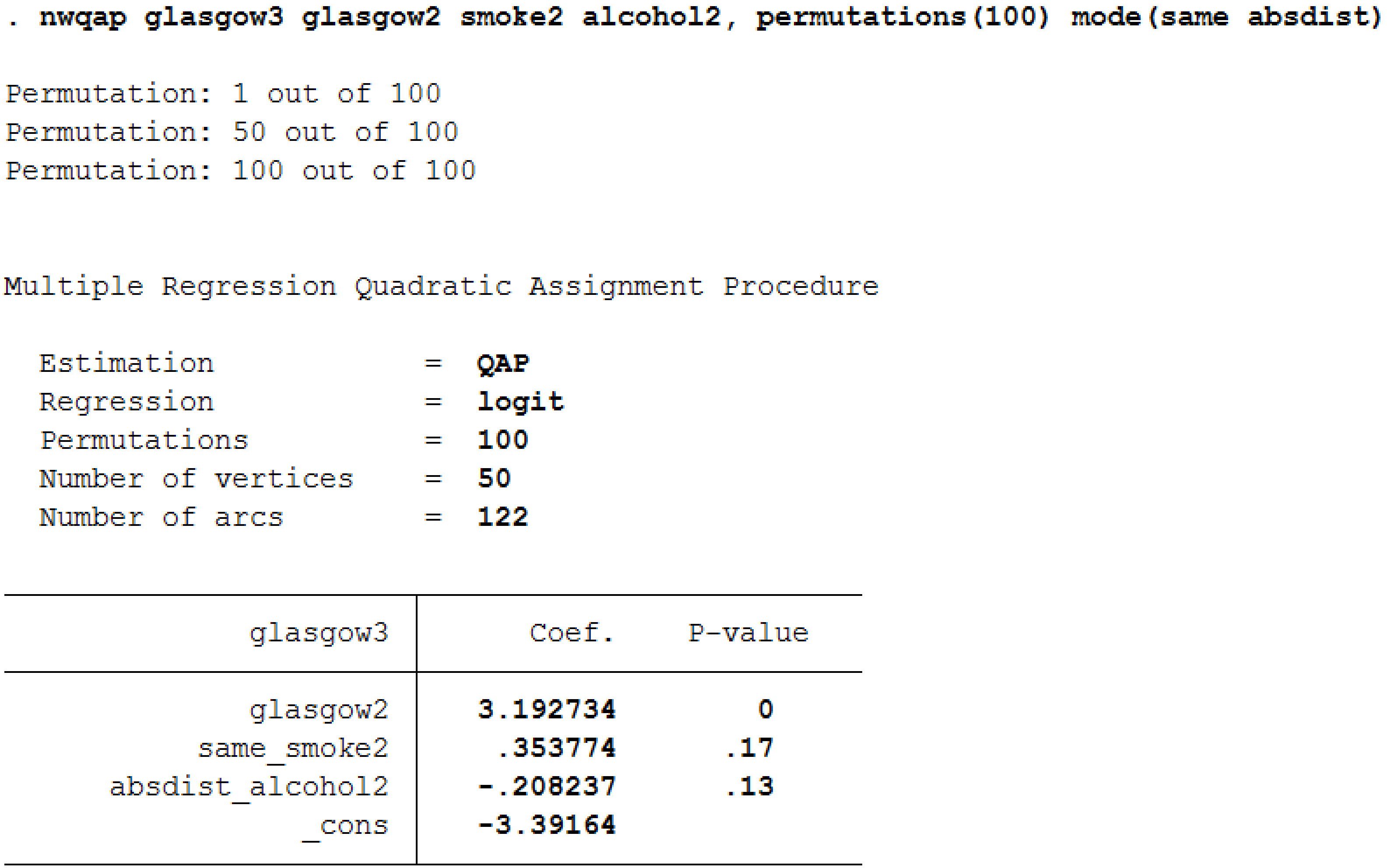
-	0	0	0
0	-	0	1
1	0	-	1
0	1	1	-

```
. nwcorrelate glasgow1 glasgow2, permutation(50)
.4732457209617567

. return list

scalars:
                 r(lb) =  -.0490217059850693
                 r(ub) =  .0426041558384895
             r(pvalue) =  0
               r(corr) =  .4732457209617567
               r(id_2) =  1
               r(id_1) =  2

macros:
             r(name_1) : "glasgow1"
             r(name_2) : "glasgow2"
```



```
. nwcorrelate glasgow1, attribute(sport1) permutation(50)
.025768556436961

. return list

scalars:
                 r(lb) =  -.0450700744986534
                 r(ub) =  .0454459525644779
             r(pvalue) =  .12
               r(corr) =  .025768556436961
               r(id_2) =  4
               r(id_1) =  2

macros:
             r(name_1) : "glasgow1"
             r(name_2) : "same_sport1"
```





```
. nwqap glasgow3 glasgow2 smoke2 alcohol2, permutations(100) mode(same absdist)

Permutation: 1 out of 100
Permutation: 50 out of 100
Permutation: 100 out of 100


Multiple Regression Quadratic Assignment Procedure

  Estimation              =  QAP
  Regression              =  logit
  Permutations            =  100
  Number of vertices      =  50
  Number of arcs          =  122

------------------------------------------------
             glasgow3 |      Coef.     P-value
----------------------+-------------------------
             glasgow2 |   3.192734         0
          same_smoke2 |    .353774       .17
     absdist_alcohol2 |   -.208237       .13
                _cons |   -3.39164
------------------------------------------------
```

Exponential random graph models

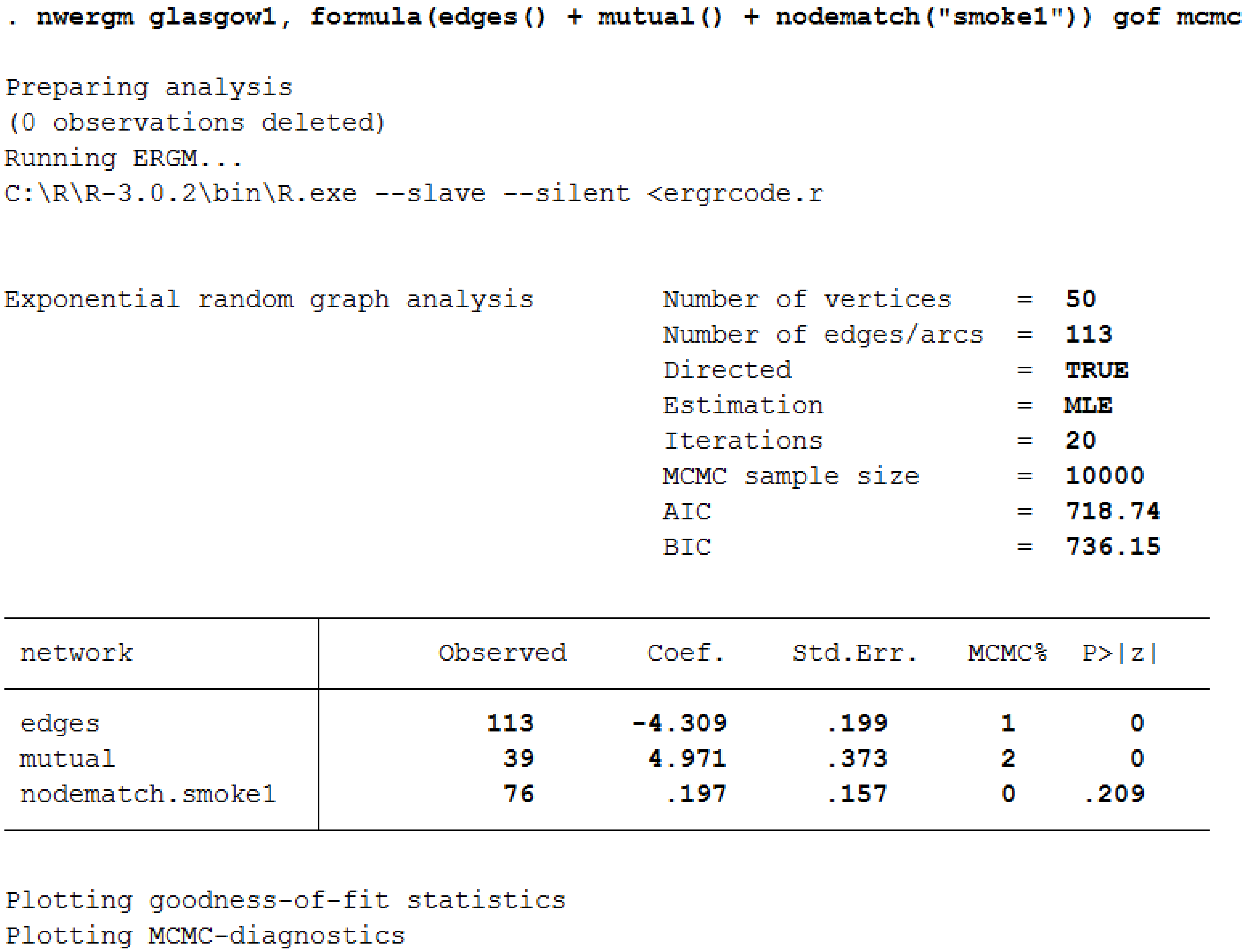
Y_{ij}^{c} = all dyads other than Y_{ij}

Amount by which the feature $s_k(y)$ changes when Y_{ij} is toggled from 0 to 1.

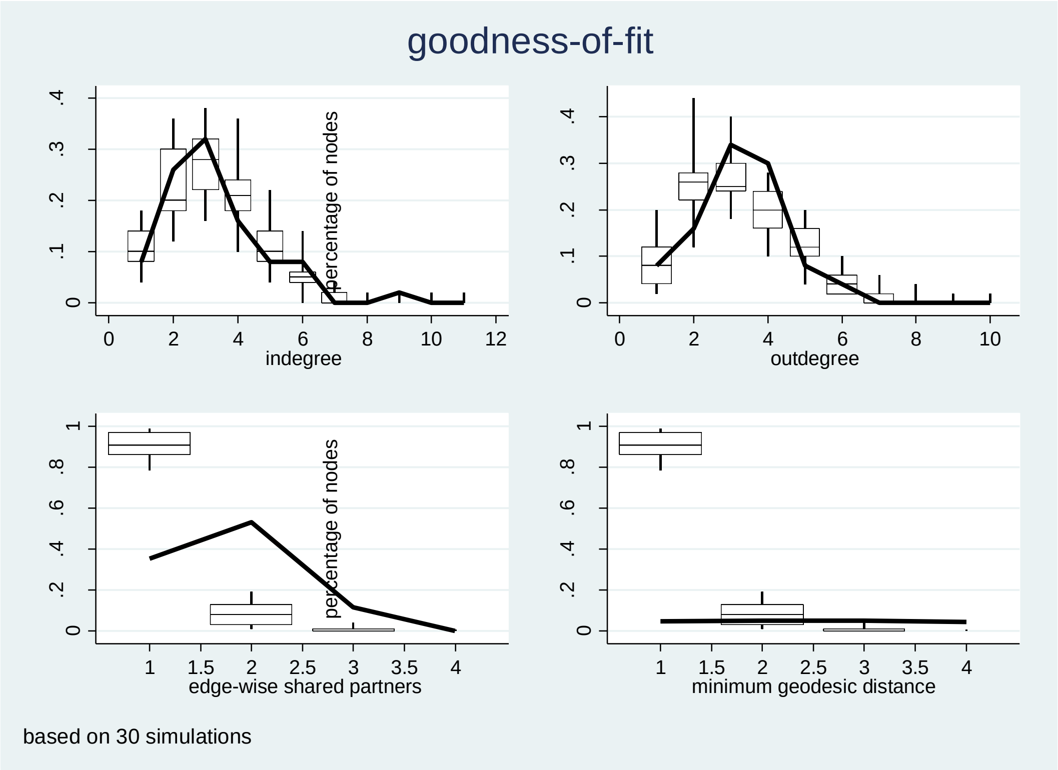
$$\operatorname{logit}\left[P\left(Y_{ij}=1 \middle| n\ actors, Y_{ij}^{c}\ \right)\right]=\sum_{k=1}^{K}\theta_k \delta s_k(\boldsymbol{y})$$

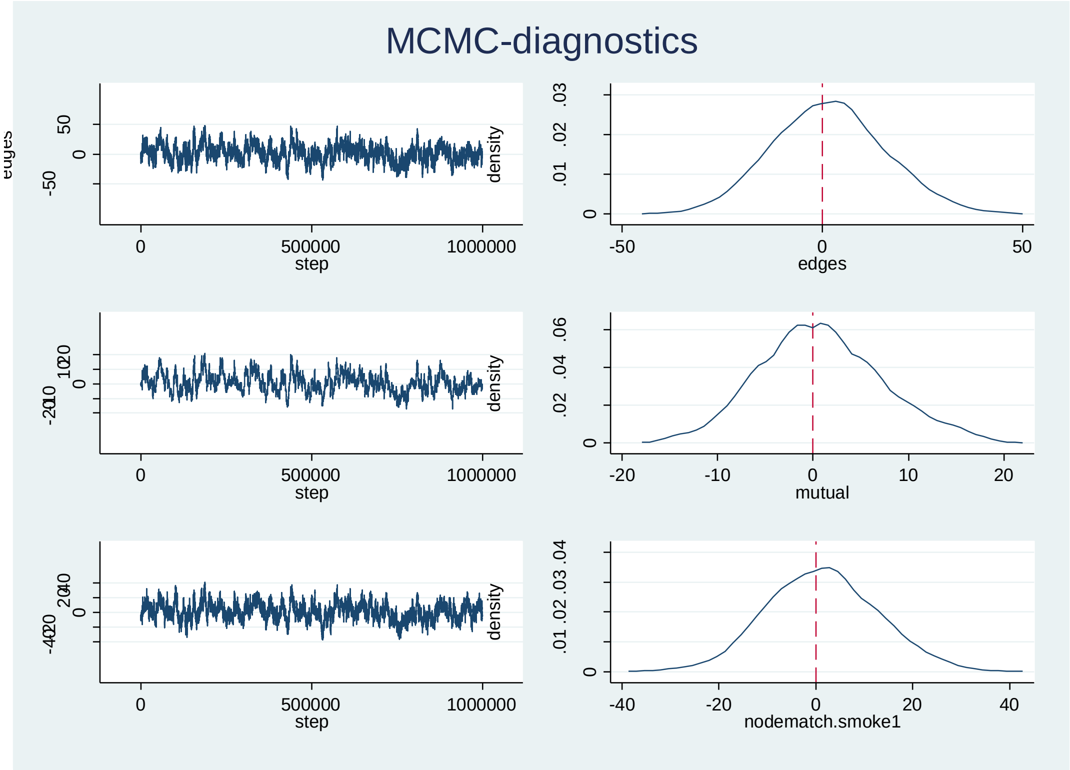
Probability that there is a tie from *i* to *j*.

Given, *n* actors AND the rest of the network, excluding the dyad in question!

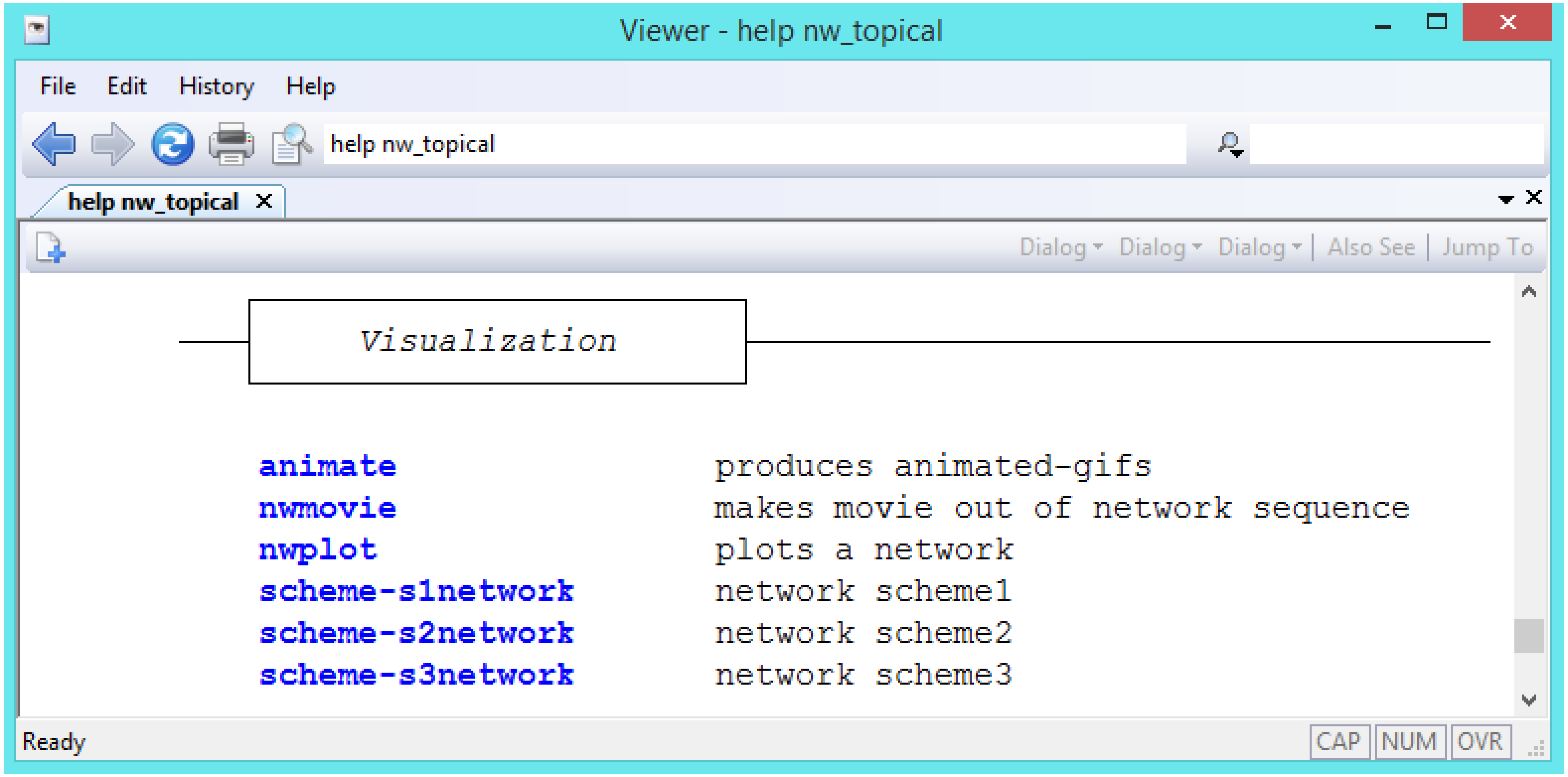


```
. nwergm glasgow1, formula(edges() + mutual() + nodematch("smoke1")) gof mcmc

Preparing analysis
(0 observations deleted)
Running ERGM...
C:\R\R-3.0.2\bin\R.exe --slave --silent <ergrcode.r


Exponential random graph analysis         Number of vertices     =  50
                                          Number of edges/arcs   =  113
                                          Directed               =  TRUE
                                          Estimation             =  MLE
                                          Iterations             =  20
                                          MCMC sample size       =  10000
                                          AIC                    =  718.74
                                          BIC                    =  736.15

------------------------------------------------------------------------------
 network           |      Observed     Coef.     Std.Err.    MCMC%   P>|z|
-------------------+----------------------------------------------------------
 edges             |           113    -4.309        .199        1        0
 mutual            |            39     4.971        .373        2        0
 nodematch.smoke1  |            76      .197        .157        0     .209
------------------------------------------------------------------------------

Plotting goodness-of-fit statistics
Plotting MCMC-diagnostics
```


goodness-of-fit
percentage of nodes
0
.1
.2
.3
.4
0
2
4
6
8
10
12
indegree
0
.1
.2
.3
.4
0
2
4
6
8
10
outdegree
percentage of nodes
0
.2
.4
.6
.8
1
1
1.5
2
2.5
3
3.5
4
edge-wise shared partners
0
.2
.4
.6
.8
1
1
1.5
2
2.5
3
3.5
4
minimum geodesic distance
based on 30 simulations


MCMC-diagnostics
edges
-50
0
50
0
500000
1000000
step
density
0
.01
.02
.03
-50
0
50
edges
-20
-10
0
10
20
0
500000
1000000
step
density
0
.02
.04
.06
-20
-10
0
10
20
mutual
-40
-20
0
20
40
0
500000
1000000
step
density
0
.01
.02
.03
.04
-40
-20
0
20
40
nodematch.smoke1

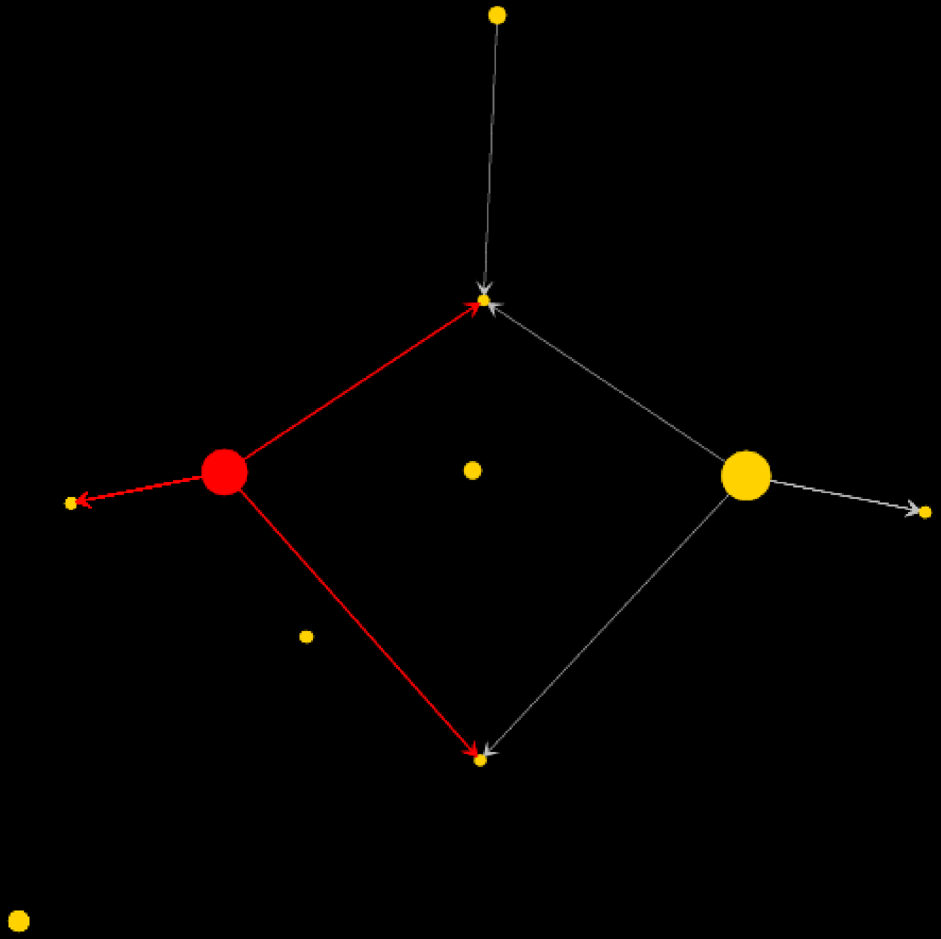
Visualize networks



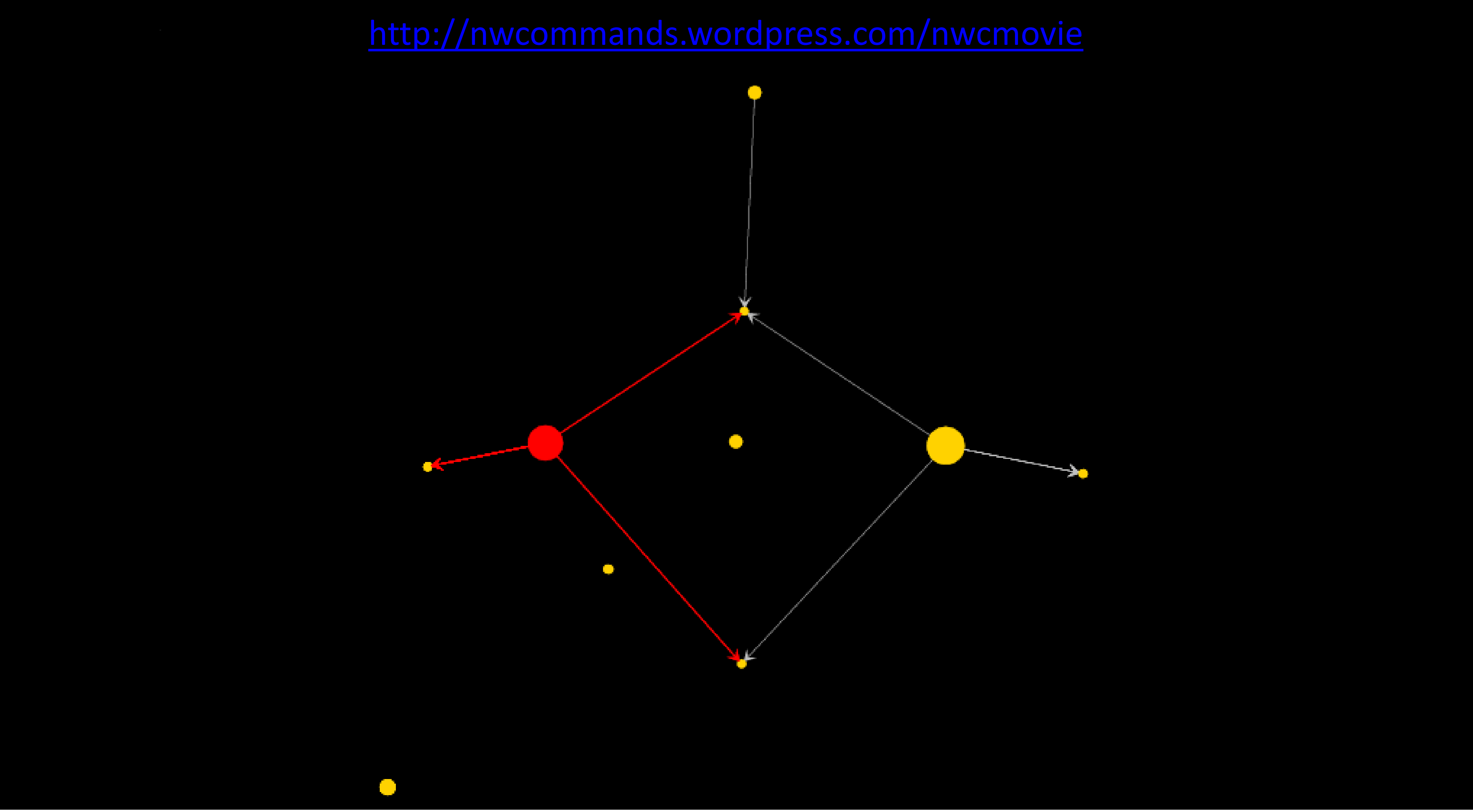
help nw_topical

```
                 Visualization

  animate                  produces animated-gifs
  nwmovie                  makes movie out of network sequence
  nwplot                   plots a network
  scheme-s1network         network scheme1
  scheme-s2network         network scheme2
  scheme-s3network         network scheme3
```

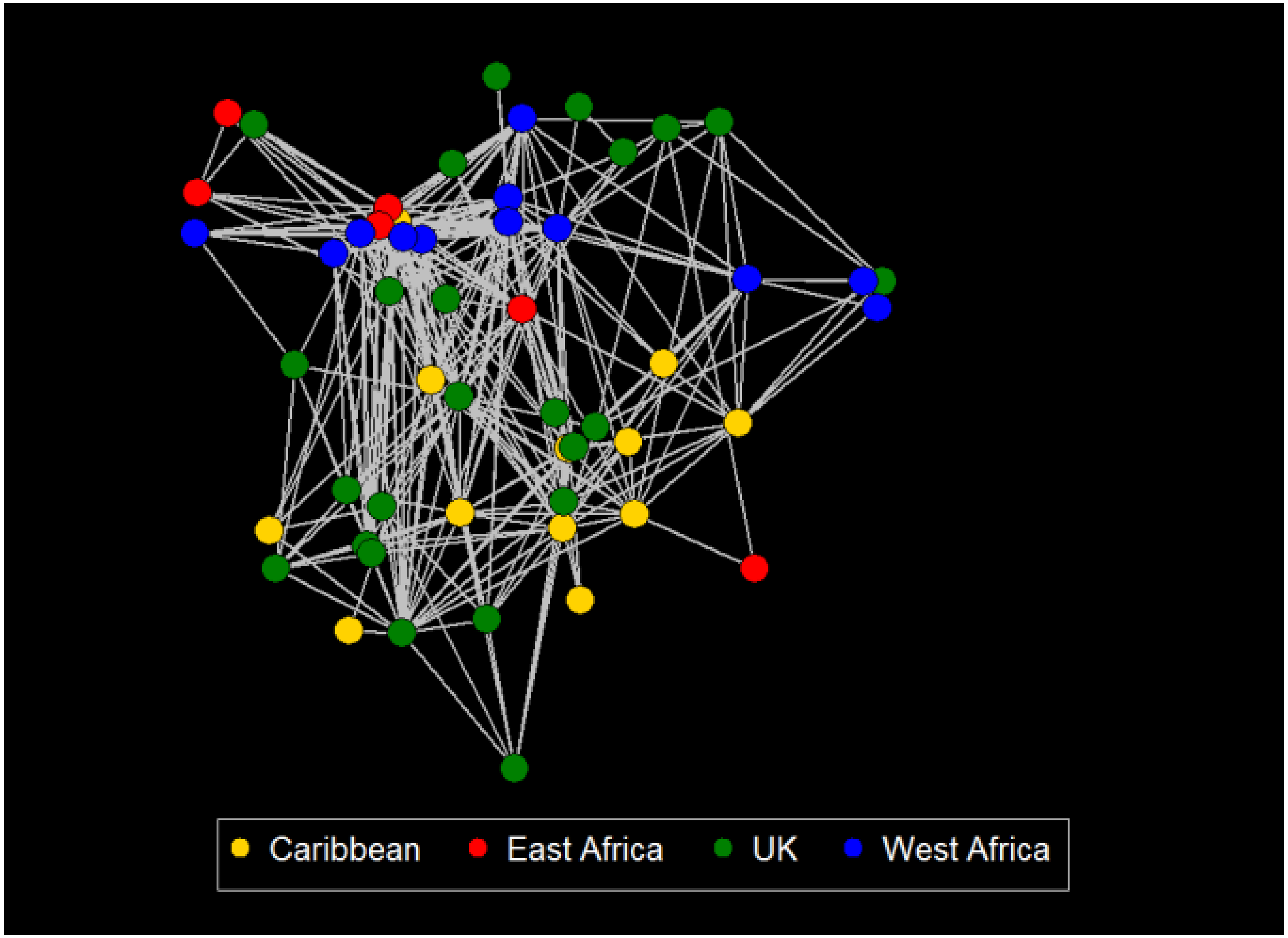
http://nwcommands.wordpress.com/nwcmovie



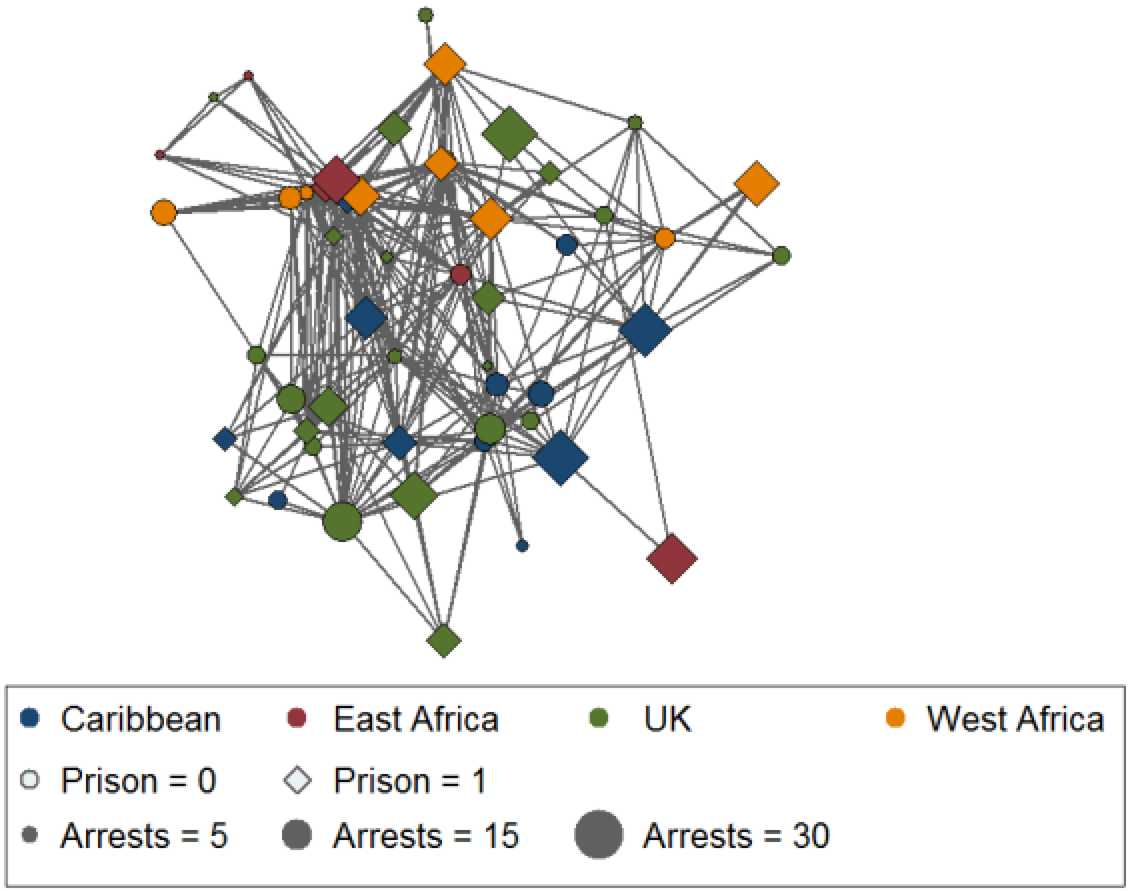
http://nwcommands.wordpress.com/nwcmovie



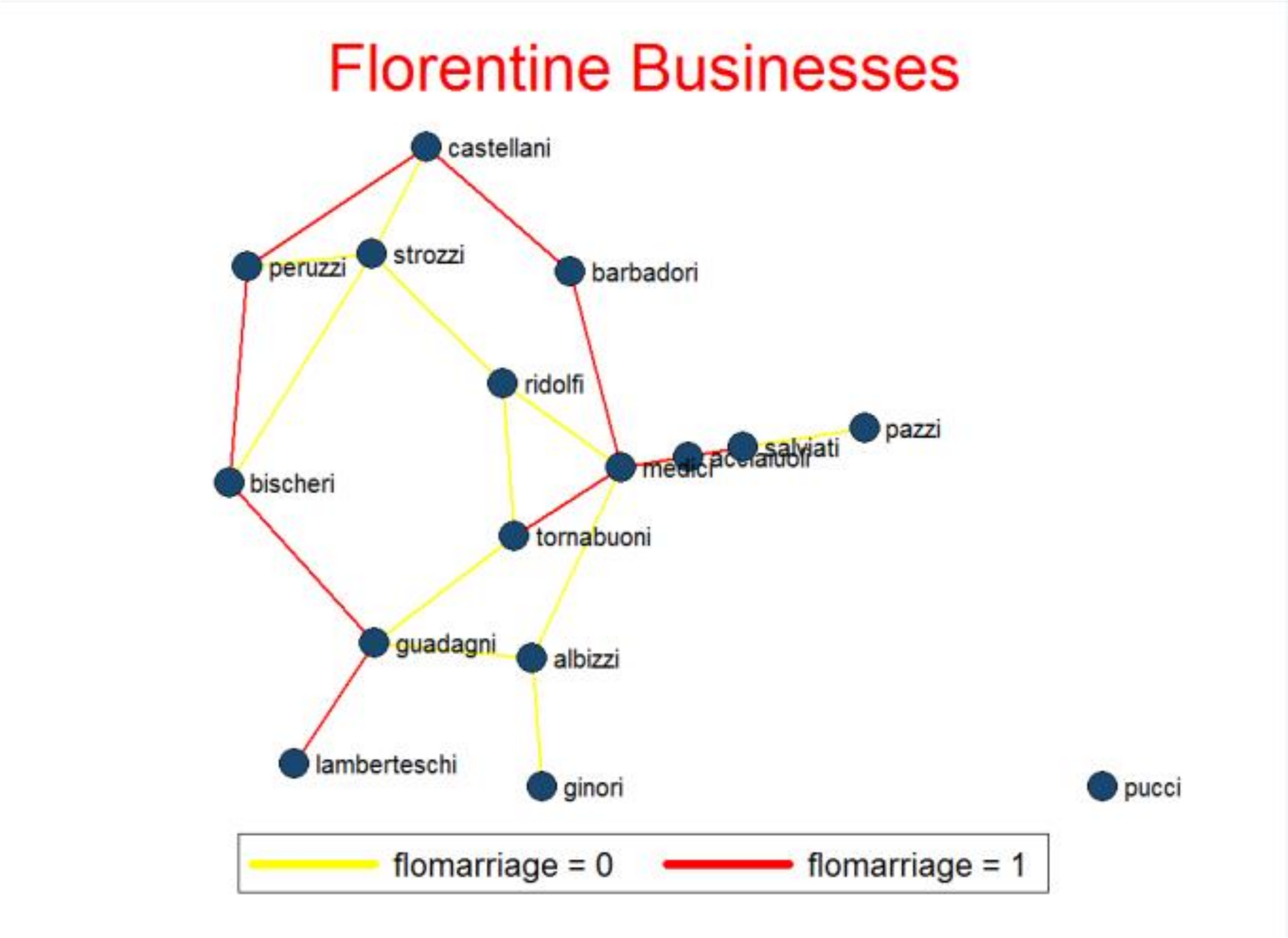
```
. nwuse gomery, nwclear
. nwmovie _all, color(col_t*) scheme(s2network)
```



```
. nwuse gang, nwclear
. nwplot, color(Birthplace) scheme(s2network)
```

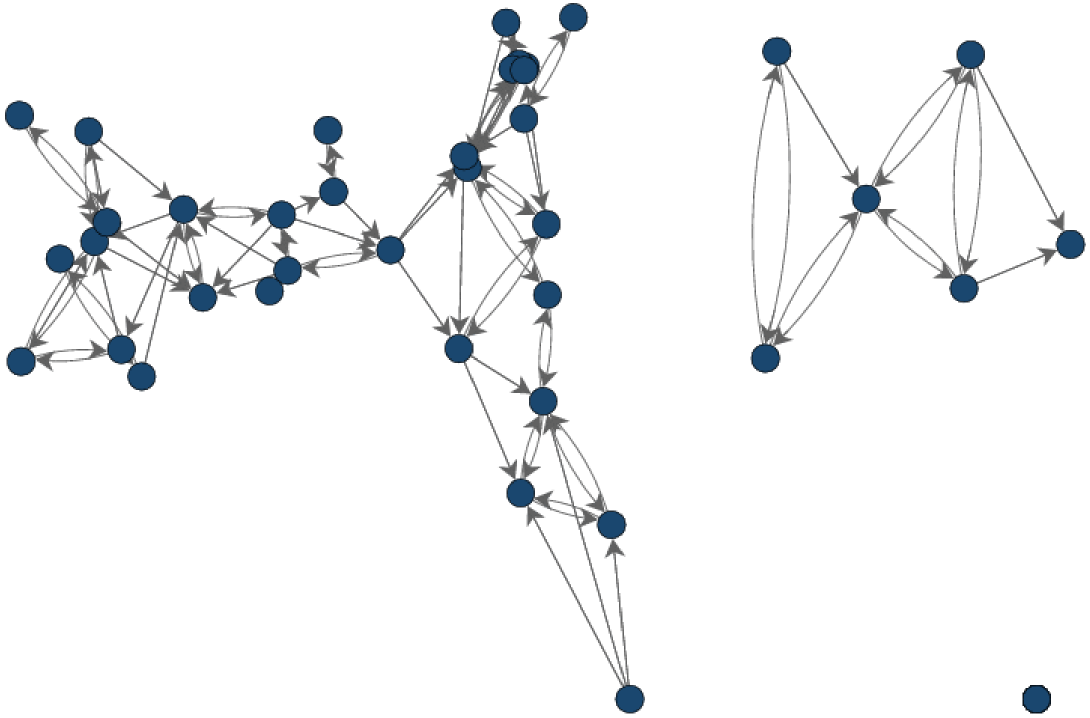


```
. nwplot, size(Arrests, forcekeys(5 15 30)) color(Birthplace)
   symbol(Prison)
```



```
. nwuse florentine, nwclear
. nwplot flobusiness, label(_label) edgecolor(flomarriage)
   edgecolorpalette(yellow red) title(“Florentine
   Businesses”, color(red) size(huge))
```

http://nwcommands.wordpress.com/demo_nwplot



```
. nwuse glasgow, nwclear
. nwmovie _all
```

http://nwcommands.org

BETA

TESTING

BETA

Thomas Grund
Linköping University

thomas.u.grund@gmail.com
www.grund.co.uk