Cohesive Subgroups

October 4, 2006

Based on slides by Steve Borgatti.
Modifications (many sensible) by Rich DeJordy

Before We Start

- Any questions on cohesion?

Why do we care about Cohesive Subgroups

1. They exist!
2. They affect (social) processes we care about.
3. They offer the opportunity for data reduction:
   - Analyze separately
   - Aggregate cohesive subgroups

So, what are they?

- Many formalized definitions for lots of different flavors of cohesive subgroups, but, in general they are:

    Sections of the network in which actors are more closely relate to each other, on the whole, than to those outside that the group.

A Two By Two
(and a half)

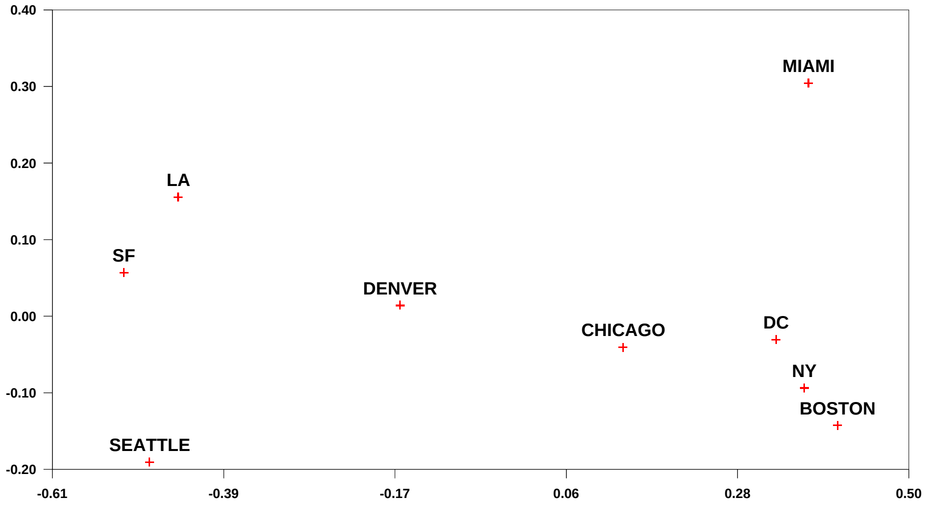
	Group by defined Algorithm	Group by defined Characteristics
Network/ Graph Theory	Newman-Girvan	**Distance**: Component, Clique, n-clique, n-clan, n-club **Density**: Clique, k-core, k-plex, ls-set, lamba set (Core/Periphery)
Proximity /Distance	Hierarchical Clustering MDS K-Means	Factions (Core/Periphery) (Combinatorial Optimization)

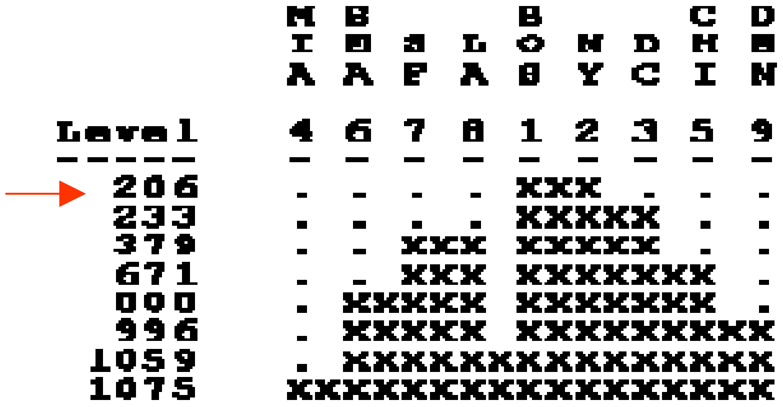
Groups defined by an algorithm based on distances/proximities

	Group by defined Algorithm	Group by defined Characteristics
Network/ Graph Theory	Newman-Girvan	**Distance**: Component, Clique, n-clique, n-clan, n-club **Density**: Clique, k-core, k-plex, ls-set, lamba set (Core/Periphery)
Proximity /Distance	Hierarchical Clustering MDS K-Means	Factions (Core/Periphery) (Combinatorial Optimization)

Johnson’s Hierarchical Clustering

- Output is a set of nested **partitions**, starting with identity partition and ending with the complete partition
  - A “PARTITION” is a vector that associates each node with one and only one “group” (mutually exclusive)
- Different flavors based on how distance from a cluster to outside point/node is defined
  - Single linkage; connectedness; minimum
  - Complete linkage; diameter; maximum
  - Average, median, etc.

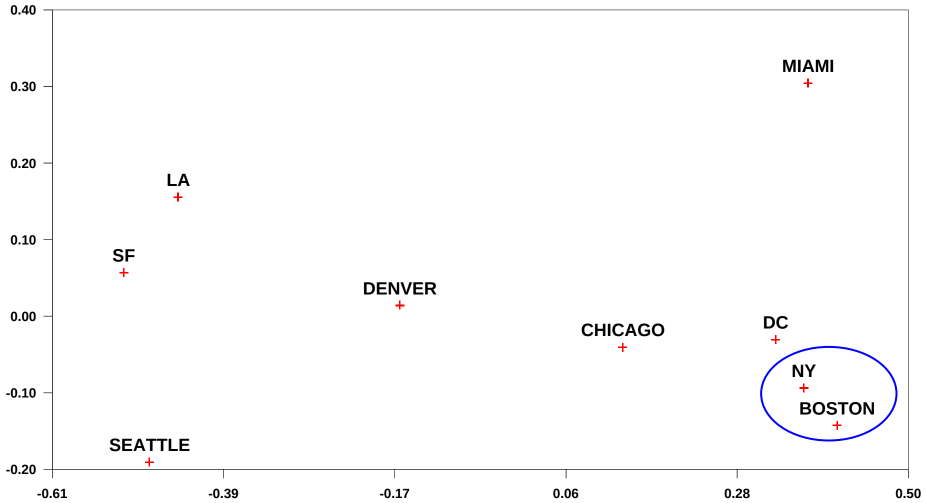


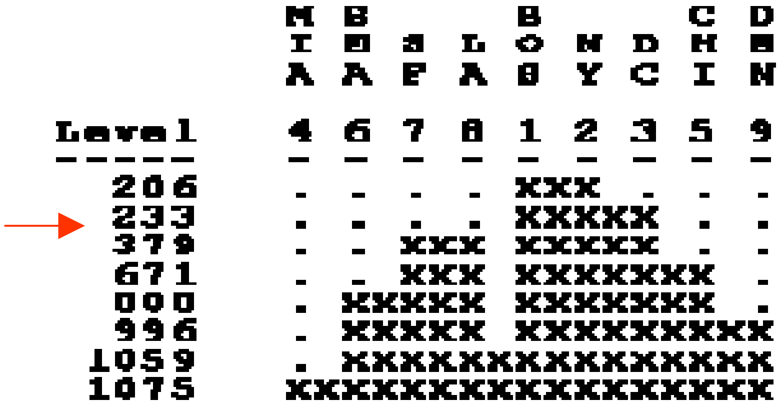


```
         M B       B       C D
         I E S L O N D H E
         A A F A S Y C I N

Level    4 6 7 8 1 2 3 5 9
-----    - - - - - - - - -
  206    . . . . XXX . . .
  233    . . . . XXXXX . .
  379    . . XXX XXXXX . .
  671    . . XXX XXXXXXX .
  000    . XXXXX XXXXXXX .
  996    . XXXXX XXXXXXXXX
 1059    . XXXXXXXXXXXXXXX
 1075    XXXXXXXXXXXXXXXXX
```

	BOS	NY	DC	MIA	CHI	SEA	SF	LA	DEN
BOS	0	206	429	1504	963	2976	3095	2979	1949
NY	206	0	233	1308	802	2815	2934	2786	1771
DC	429	233	0	1075	671	2684	2799	2631	1616
MIA	1504	1308	1075	0	1329	3273	3053	2687	2037
CHI	963	802	671	1329	0	2013	2142	2054	996
SEA	2976	2815	2684	3273	2013	0	808	1131	1307
SF	3095	2934	2799	3053	2142	808	0	379	1235
LA	2979	2786	2631	2687	2054	1131	379	0	1059
DEN	1949	1771	1616	2037	996	1307	1235	1059	0

Closest distance is NY-BOS = 206, so merge these.

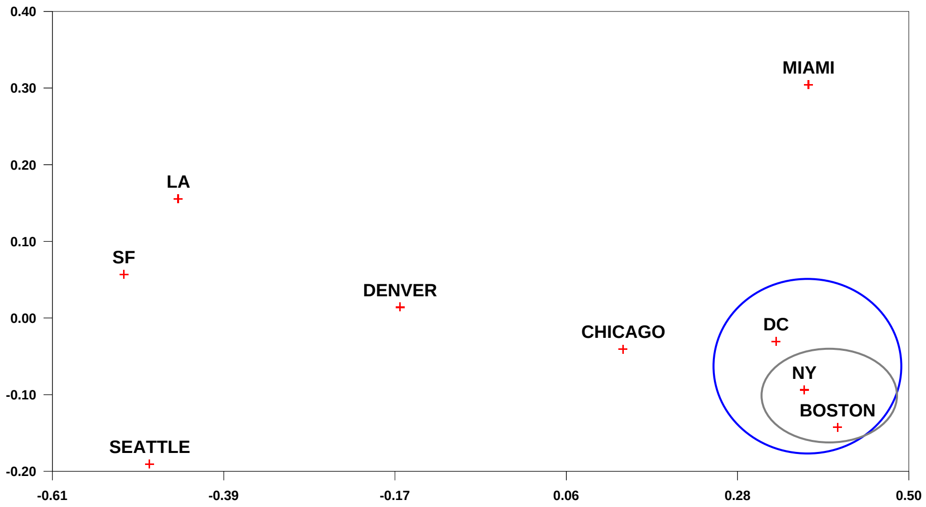


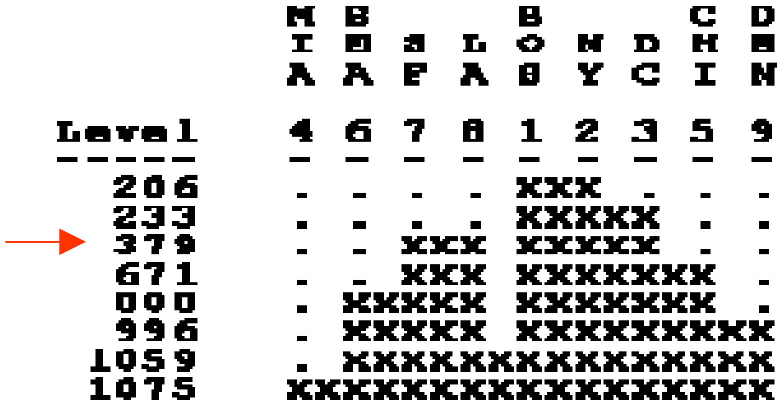


```
               M B       B       C D
               I E S L O N D H E
               A A F A S Y C I N

Level          4 6 7 8 1 2 3 5 9
-----          - - - - - - - - -
  206          . . . . XXX . . .
  233          . . . . XXXXX . .
  379          . . XXX XXXXX . .
  671          . . XXX XXXXXXX .
  000          . XXXXX XXXXXXX .
  996          . XXXXX XXXXXXXXX
 1059          . XXXXXXXXXXXXXXX
 1075          XXXXXXXXXXXXXXXXX
```

	BOSNY	DC	MIA	CHI	SEA	SF	LA	DEN
BOS/ NY	0	233	1308	802	2815	2934	2786	1771
DC	233	0	1075	671	2684	2799	2631	1616
MIA	1308	1075	0	1329	3273	3053	2687	2037
CHI	802	671	1329	0	2013	2142	2054	996
SEA	2815	2684	3273	2013	0	808	1131	1307
SF	2934	2799	3053	2142	808	0	379	1235
LA	2786	2631	2687	2054	1131	379	0	1059
DEN	1771	1616	2037	996	1307	1235	1059	0

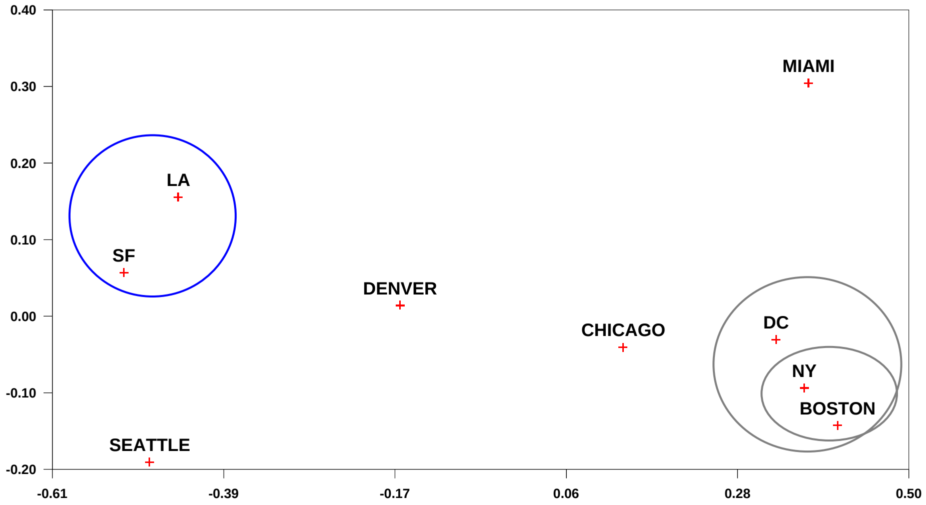
Closest pair is DC to BOSNY combo @ 233. So merge these.

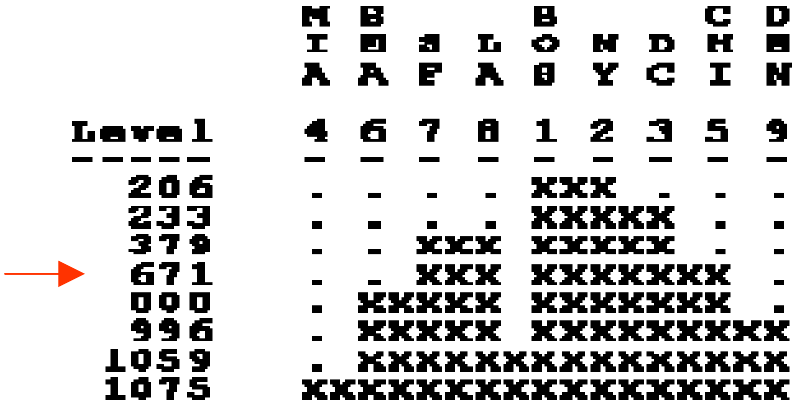




```
            M S     B     C D
            I E S L O N D H E
            A A F A S Y C I N

Level       4 6 7 8 1 2 3 5 9
-----       - - - - - - - - -
  206       . . . . XXX . . .
  233       . . . . XXXXX . .
  379       . . XXX XXXXX . .
  671       . . XXX XXXXXXX .
  000       . XXXXX XXXXXXX .
  996       . XXXXX XXXXXXXXX
 1059       . XXXXXXXXXXXXXXX
 1075       XXXXXXXXXXXXXXXXX
```

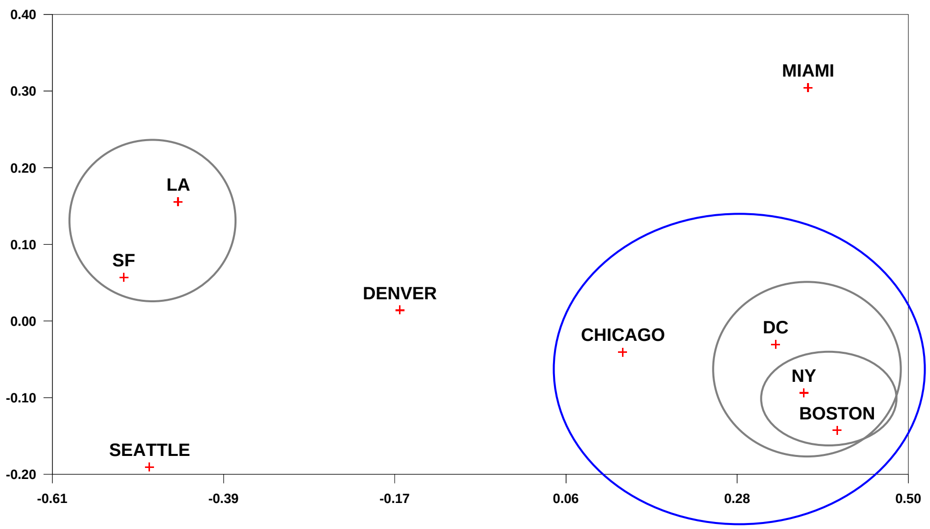
	BOS/ NY/ DC	MIA	CHI	SEA	SF	LA	DEN
BOS/NY DC	0	1075	671	2684	2799	2631	1616
MIA	1075	0	1329	3273	3053	2687	2037
CHI	671	1329	0	2013	2142	2054	996
SEA	2684	3273	2013	0	808	1131	1307
SF	2799	3053	2142	808	0	379	1235
LA	2631	2687	2054	1131	379	0	1059
DEN	1616	2037	996	1307	1235	1059	0





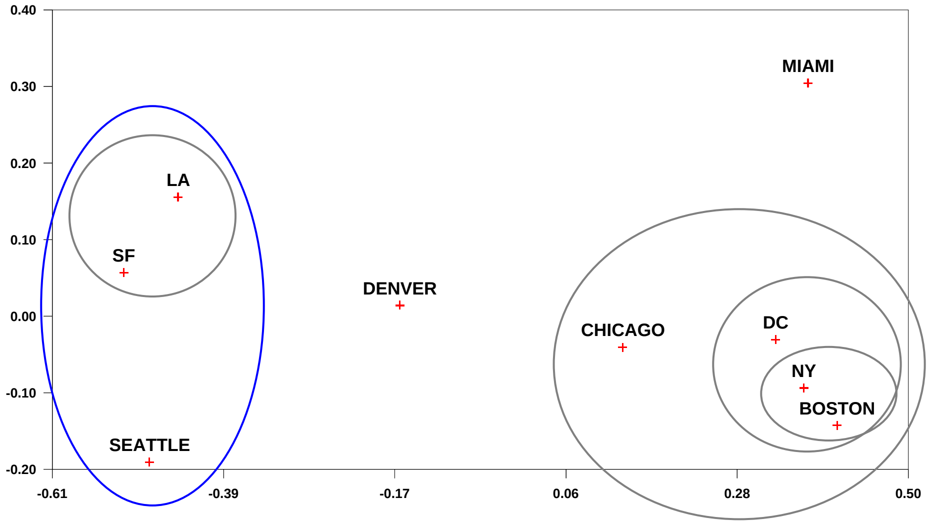
```
              M B       B       C D
              I E S L O N D H E
              A A F A S Y C I N

 Level        4 6 7 8 1 2 3 5 9
 -----        - - - - - - - - -
   206        . . . . XXX . . .
   233        . . . . XXXXX . .
   379        . . XXX XXXXX . .
   671        . . XXX XXXXXXX .
   000        . XXXXX XXXXXXX .
   996        . XXXXX XXXXXXXXX
  1059        . XXXXXXXXXXXXXXX
  1075        XXXXXXXXXXXXXXXXX
```

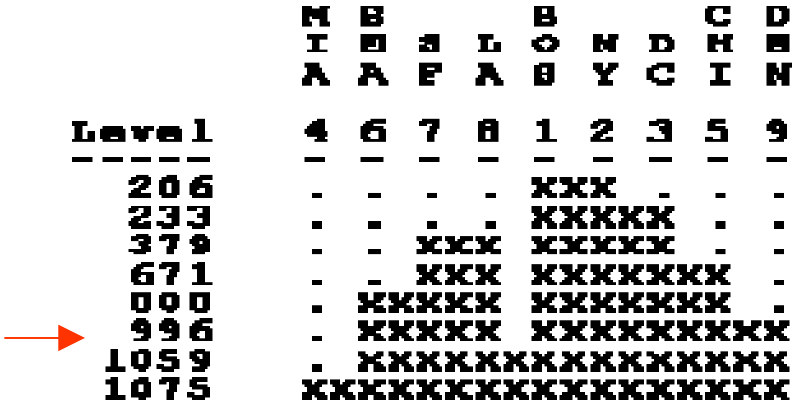
	BOS/ NY/DC	MIA	CHI	SEA	SF/LA	DEN
BOS/NY/DC	0	1075	671	2684	2631	1616
MIA	1075	0	1329	3273	2687	2037
CHI	671	1329	0	2013	2054	996
SEA	2684	3273	2013	0	808	1307
SF/LA	2631	2687	2054	808	0	1059
DEN	1616	2037	996	1307	1059	0



```
                    M B       B       C D
                    I E S L O N D H E
                    A A F A S Y C I N

Level               4 6 7 8 1 2 3 5 9
-----               - - - - - - - - -
  206               . . . . XXX . . .
  233               . . . . XXXXX . .
  379               . . XXX XXXXX . .
  671               . . XXX XXXXXXX .
  000               . XXXXX XXXXXXX .
  996               . XXXXX XXXXXXXXX
 1059               . XXXXXXXXXXXXXXX
 1075               XXXXXXXXXXXXXXXXX
```

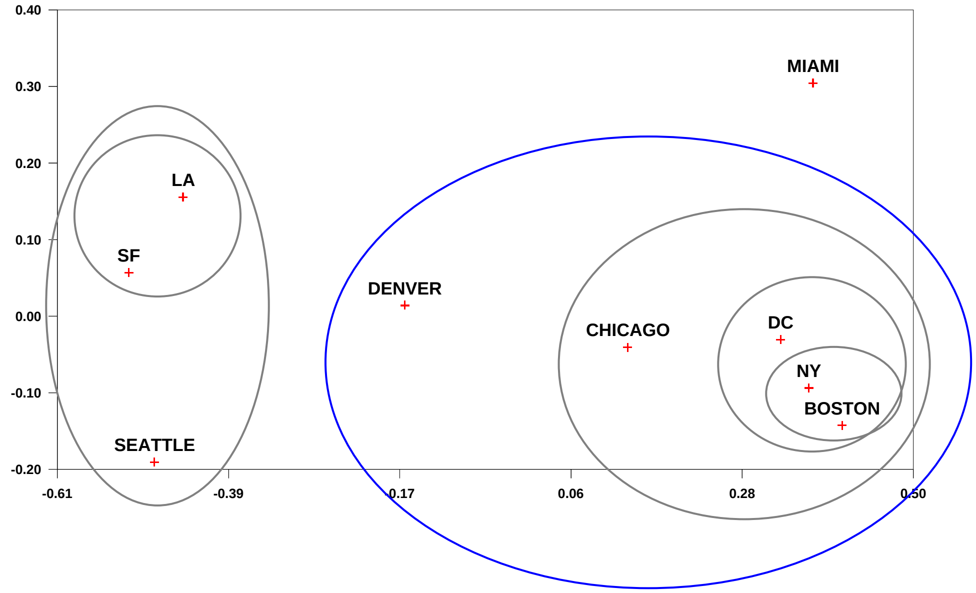
	BOS/ NY/D C/ CHI	MIA	SEA	SF/L A	DEN
BOS/NY/DC/C HI	0	1075	2013	2054	996
MIA	1075	0	3273	2687	2037
SEA	2013	3273	0	808	1307
SF/LA	2054	2687	808	0	1059
DEN	996	2037	1307	1059	0

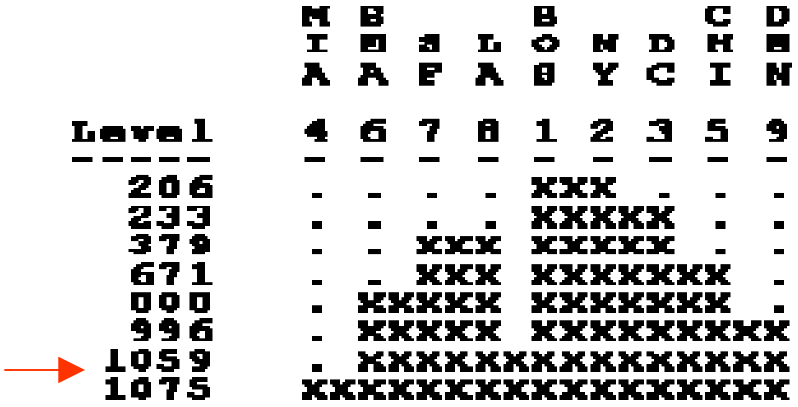




```
            M S       B         C D
            I E S L O N D H E
            A A F A S Y C I N

Level       4 6 7 8 1 2 3 5 9
-----       - - - - - - - - -
  206       . . . . XXX . . .
  233       . . . . XXXXX . .
  379       . . XXX XXXXX . .
  671       . . XXX XXXXXXX .
  000       . XXXXX XXXXXXX .
  996       . XXXXX XXXXXXXXX
 1059       . XXXXXXXXXXXXXXX
 1075       XXXXXXXXXXXXXXXXX
```

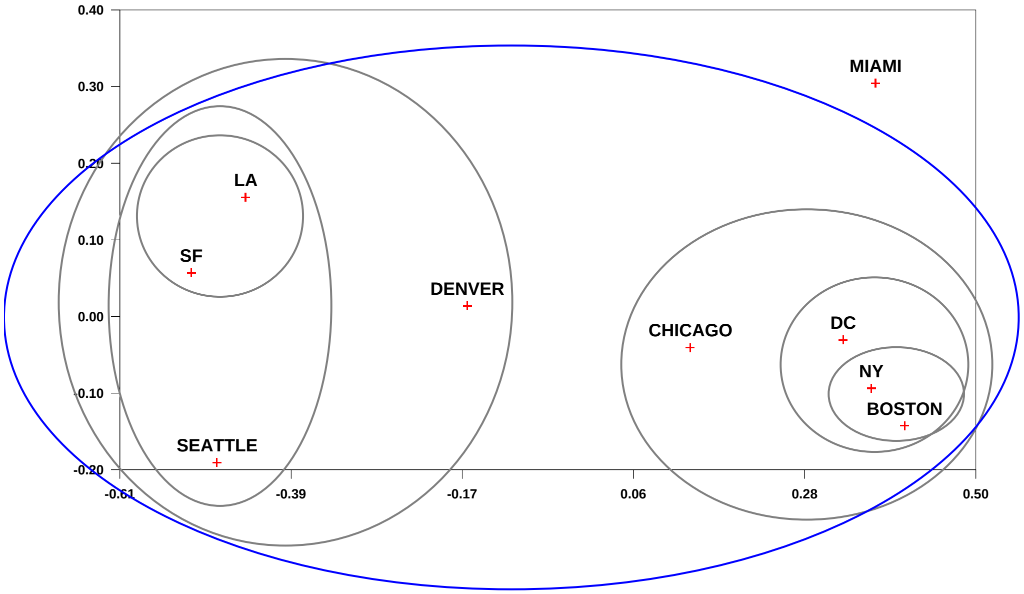
	BOS/NY/DC/CHI	MIA	SF/LA/SEA	DEN
BOS/NY/DC/CHI	0	1075	2013	996
MIA	1075	0	2687	2037
SF/LA/SEA	2054	2687	0	1059
DEN	996	2037	1059	0

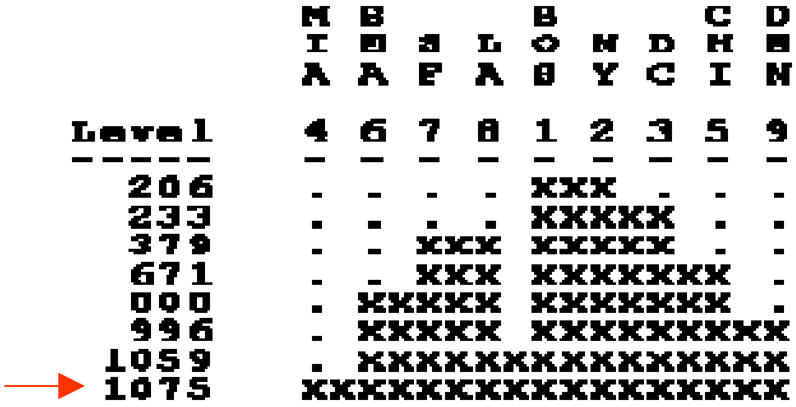




```
       M B     B     C D
       I E S L O N D H E
       A A F A S Y C I N

Level  4 6 7 8 1 2 3 5 9
-----  - - - - - - - - -
  206  . . . . XXX . . .
  233  . . . . XXXXX . .
  379  . . XXX XXXXX . .
  671  . . XXX XXXXXXX .
  000  . XXXXX XXXXXXX .
  996  . XXXXX XXXXXXXXX
 1059  . XXXXXXXXXXXXXXX
 1075  XXXXXXXXXXXXXXXXX
```

	BOS/NY/DC/CHI/DEN	MIA	SF/LA/SEA
BOS/NY/DC/CHI/DEN	0	1075	1059
MIA	1075	0	2687
SF/LA/SEA	1059	2687	0





```
                  M B       B       C D
                  I E S L O N D H E
                  A A F A S Y C I N

Level             4 6 7 8 1 2 3 5 9
-----             - - - - - - - - -
  206             . . . . XXX . . .
  233             . . . . XXXXX . .
  379             . . XXX XXXXX . .
  671             . . XXX XXXXXXX .
  000             . XXXXX XXXXXXX .
  996             . XXXXX XXXXXXXXX
 1059             . XXXXXXXXXXXXXXX
 1075             XXXXXXXXXXXXXXXXX
```

	BOS/NY/DC/CHI/DEN/SF/LA/SEA	MIA
BOS/NY/DC/CHI/DEN/SF/LA/SEA	0	1075
MIA	1075	0

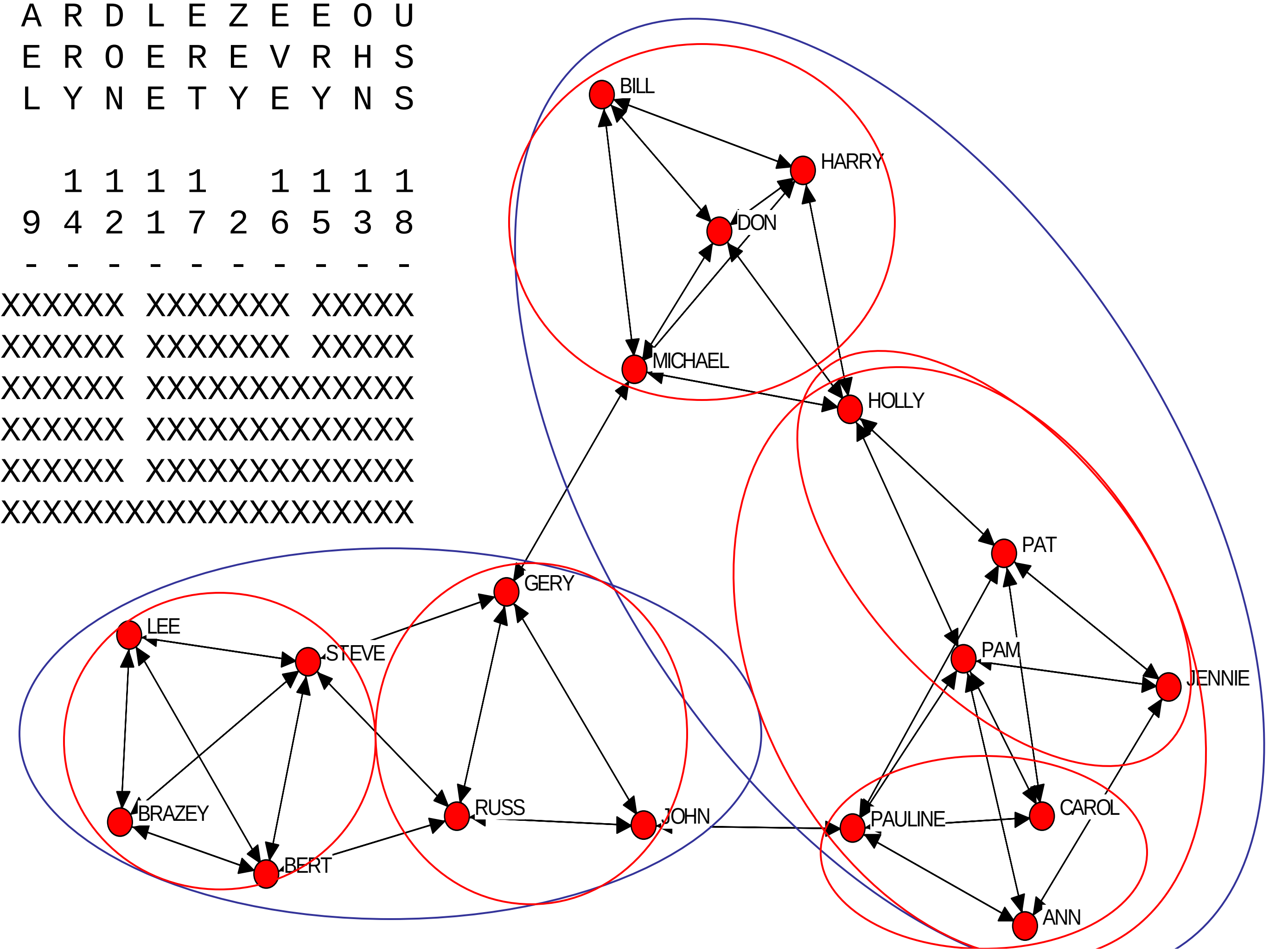
Clustering Network Data

- Clustering requires symmetric data
  - – It will symmetrize if you don’t
- Does better with a range of values

```
                    P   M
          B       J A   I
        H R C     E U   C         H   S
        O A A     N L   H B     J A G T B R
        L Z R P P N I A A I L D O R E E E U
        L E O A A I N N E L E O H R R V R S
        Y Y L M T E E N L L E N N Y Y E T S
                          1 1 1 1 1 1 1 1 1
Level   1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8
-----   - - - - - - - - - - - - - - - - - -
    0   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

Average method helps…

```
             P           M
             A     J     I         B
           C U   H E     C H       R S
           A L   O N   B H A     B A T G J R
         P R I P L N A I A R D L E Z E E O U
         A O N A L I N L E R O E R E V R H S
         T L E M Y E N L L Y N E T Y E Y N S

                       1   1 1 1 1   1 1 1 1
 Level   5 3 7 4 1 6 8 0 9 4 2 1 7 2 6 5 3 8
------   - - - - - - - - - - - - - - - - - -
1.0000   XXXXX XXX XXX XXXXXXX XXXXXXX XXXXX
0.6667   XXXXX XXXXXXX XXXXXXX XXXXXXX XXXXX
0.5657   XXXXX XXXXXXX XXXXXXX XXXXXXXXXXXXX
0.5185   XXXXXXXXXXXXX XXXXXXX XXXXXXXXXXXXX
0.0689   XXXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXX
0.0119   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```



Applying HiClus to Network Data

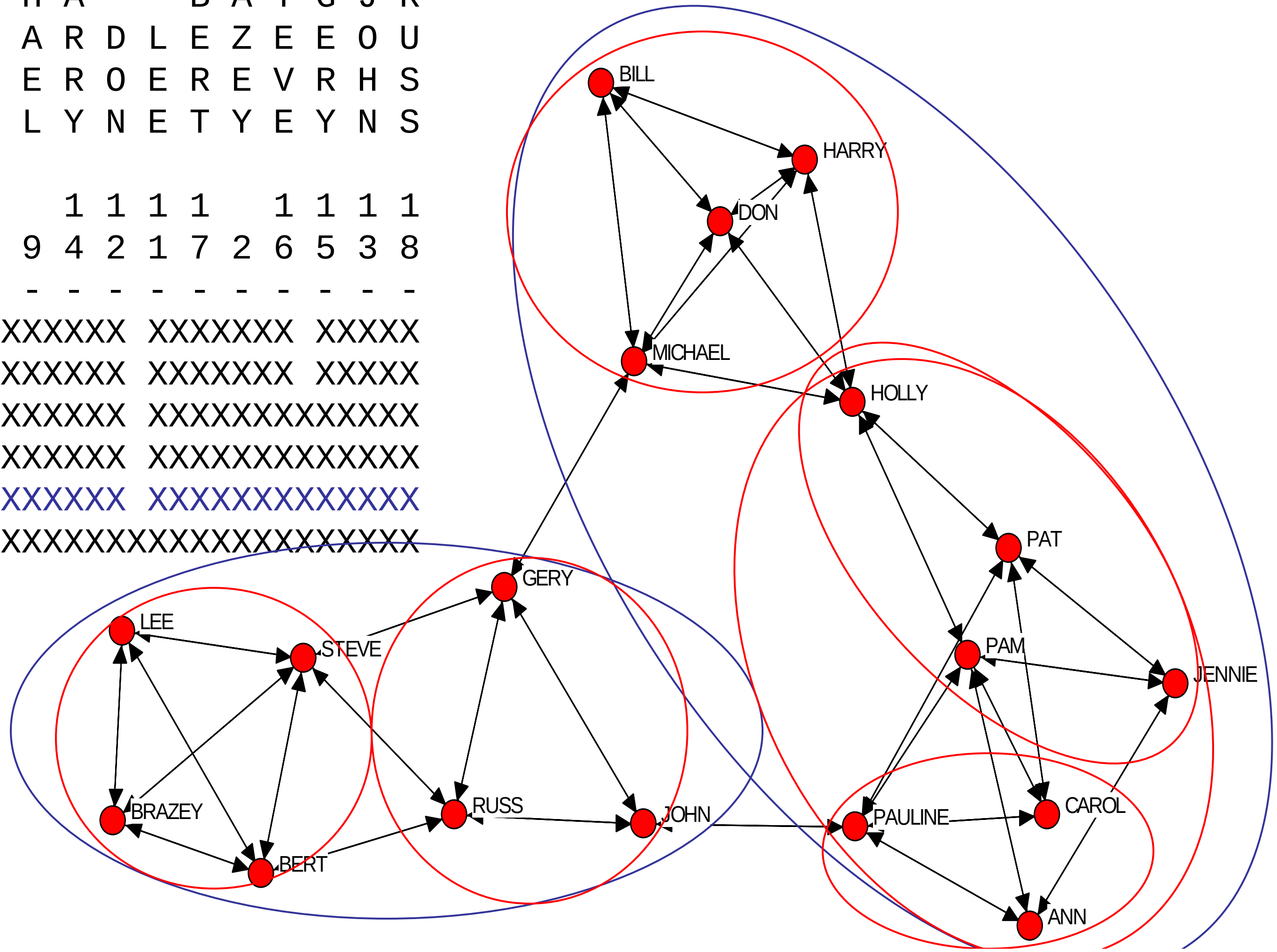
- BETTER: Compute geodesic distances first, then cluster the distance matrix

- Or cluster the structural equivalence matrix

```
                                 Geodesic Distances

                               1 1 1 1 1 1 1 1 1
             1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8
             H B C P P J P A M B L D J H G S B R
             - - - - - - - - - - - - - - - - - -
 1    HOLLY  0 4 2 1 1 2 2 2 1 2 4 1 3 1 2 3 4 3
 2   BRAZEY  4 0 5 5 5 6 4 5 3 4 1 4 3 4 2 1 1 2
 3    CAROL  2 5 0 1 1 2 1 2 3 4 5 3 2 3 3 4 4 3
 4      PAM  1 5 1 0 2 1 1 1 2 3 5 2 2 2 3 4 4 3
 5      PAT  1 5 1 2 0 1 1 2 2 3 5 2 2 2 3 4 4 3
 6   JENNIE  2 6 2 1 1 0 2 1 3 4 6 3 3 3 4 5 5 4
 7  PAULINE  2 4 1 1 1 2 0 1 3 4 4 3 1 3 2 3 3 2
 8      ANN  2 5 2 1 2 1 1 0 3 4 5 3 2 3 3 4 4 3
 9  MICHAEL  1 3 3 2 2 3 3 3 0 1 3 1 2 1 1 2 3 2
10     BILL  2 4 4 3 3 4 4 4 1 0 4 1 3 1 2 3 4 3
11      LEE  4 1 5 5 5 6 4 5 3 4 0 4 3 4 2 1 1 2
12      DON  1 4 3 2 2 3 3 3 1 1 4 0 3 1 2 3 4 3
13     JOHN  3 3 2 2 2 3 1 2 2 3 3 3 0 3 1 2 2 1
14    HARRY  1 4 3 2 2 3 3 3 1 1 4 1 3 0 2 3 4 3
15     GERY  2 2 3 3 3 4 2 3 1 2 2 2 1 2 0 1 2 1
16    STEVE  3 1 4 4 4 5 3 4 2 3 1 3 2 3 1 0 1 1
17     BERT  4 1 4 4 4 5 3 4 3 4 1 4 2 4 2 1 0 1
18     RUSS  3 2 3 3 3 4 2 3 2 3 2 3 1 3 1 1 1 0
```

Hierarchical Clustering

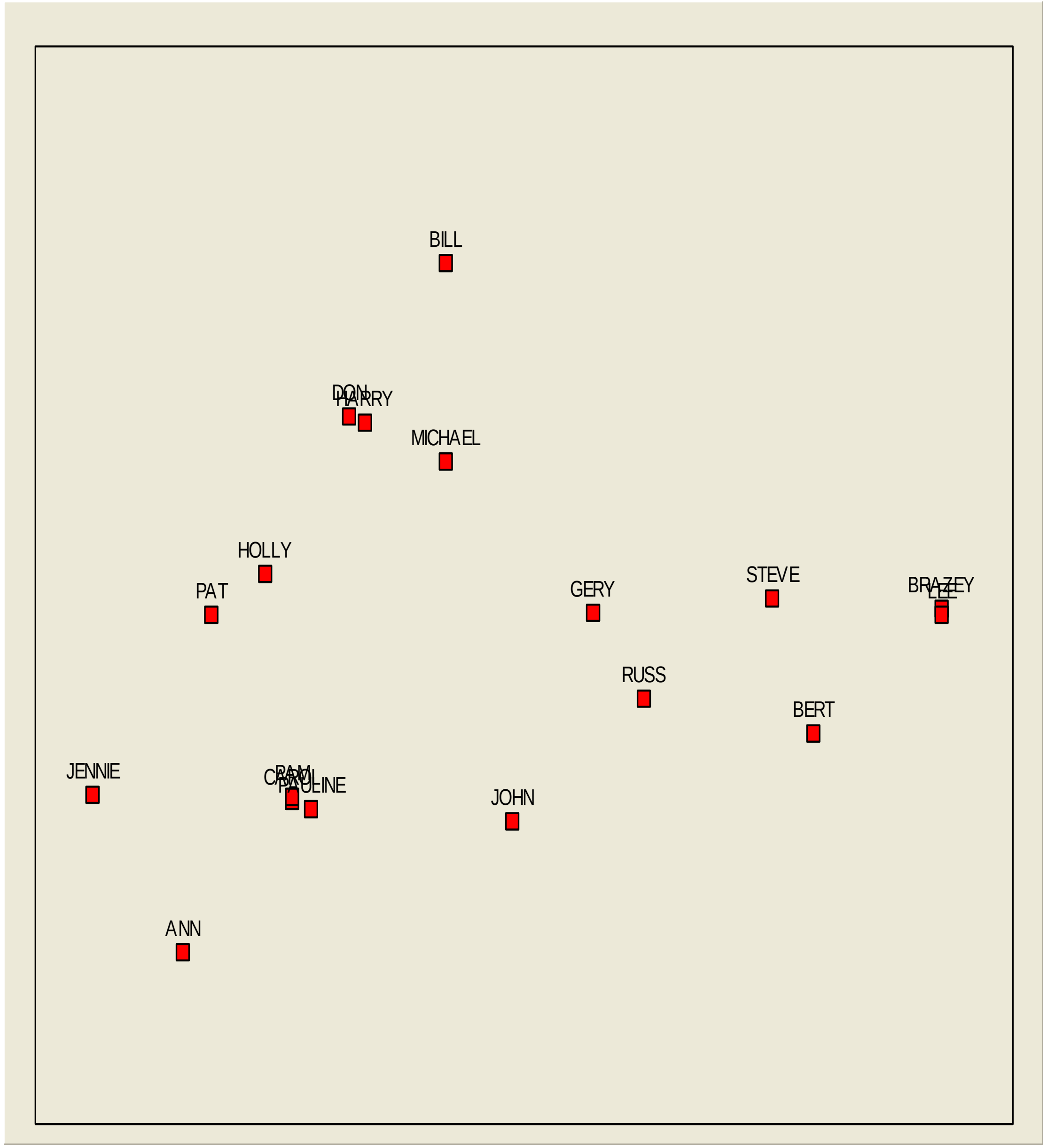
```
            P           M
            A     J     I         B
          C U   H E     C H       R S
          A L   O N   B H A     B A T G J R
        P R I P L N A I A R D L E Z E E O U
        A O N A L I N L E R O E R E V R H S
        T L E M Y E N L L Y N E T Y E Y N S

                      1   1 1 1 1   1 1 1 1
Level   5 3 7 4 1 6 8 0 9 4 2 1 7 2 6 5 3 8
-----   - - - - - - - - - - - - - - - - - -
1.000   XXXXX XXX XXX XXXXXXX XXXXXXX XXXXX
1.333   XXXXX XXXXXXX XXXXXXX XXXXXXX XXXXX
1.457   XXXXX XXXXXXX XXXXXXX XXXXXXXXXXXXX
1.481   XXXXXXXXXXXXX XXXXXXX XXXXXXXXXXXXX
2.723   XXXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXX
3.142   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```



MDS

- Goal: Position nodes in space with “similar” nodes near each other
    - Nodes connected to the same people are “similar”

- Works better with more discriminating data (range of values)



K-Means

- Given a set of points in space
  - Place K new points, called centroids, (as far away from each other as possible) among the data
  - Associated each point with the closest centroid
  - Now, based on which points are associated with each centroid, reposition the centroid to the “middle” of all of them
  - Repeat until recentering does not move the centroids.
- Basically, groups are “clustered” based on their spatial relation to each other and these centroids.

Groups defined by an algorithm based on graph theoretic properties

	Group by defined Algorithm	Group by defined Characteristics
Network/Graph Theory	Newman-Girvan	**Distance**: Component, Clique, n-clique, n-clan, n-club **Density**: Clique, k-core, k-plex, ls-set, lamba set (Core/Periphery)
Proximity/Distance	Hierarchical Clustering MDS K-Means	Factions (Core/Periphery) (Combinatorial Optimization)

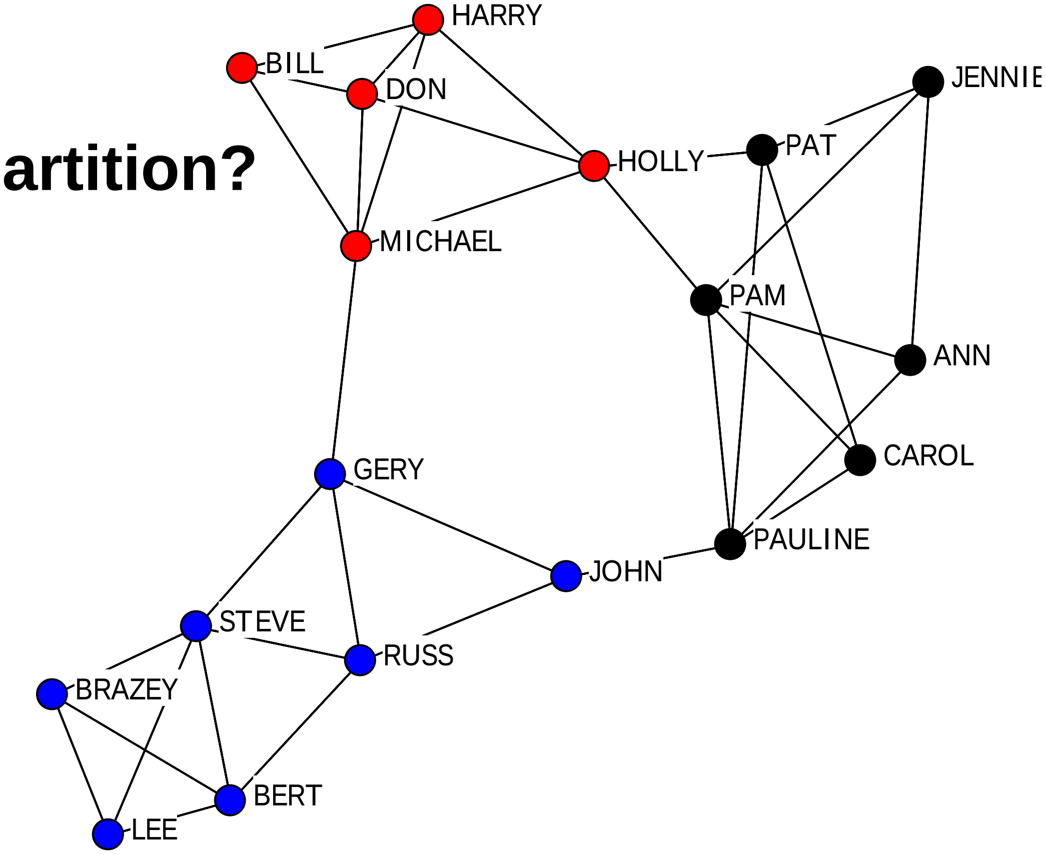
Newman-Girvan

- Calculate edge-betweenness for graph
- Remove edge with highest edge betweenness
- If number of components increases, create partition
- Recalculate edge betweenness & repeat until all nodes are isolates or maximum number of clusters reached/exceeded

Campnet Example
Newman-Girvan 2 Cluster Partition

Where do you think the 2 Cluster Partition was?

What about the 4 Cluster Partition?



Groups w/specified characteristics, based on Graph Theoretic Measures

	Group by defined Algorithm	Group by defined Characteristics
Network/ Graph Theory	Newman-Girvan	**Distance**: Component, Clique, n-clique, n-clan, n-club **Density**: Clique, k-core, k-plex, ls-set, lamba set (Core/Periphery)
Proximity /Distance	Hierarchical Clustering MDS K-Means	Factions (Core/Periphery) (Combinatorial Optimization)

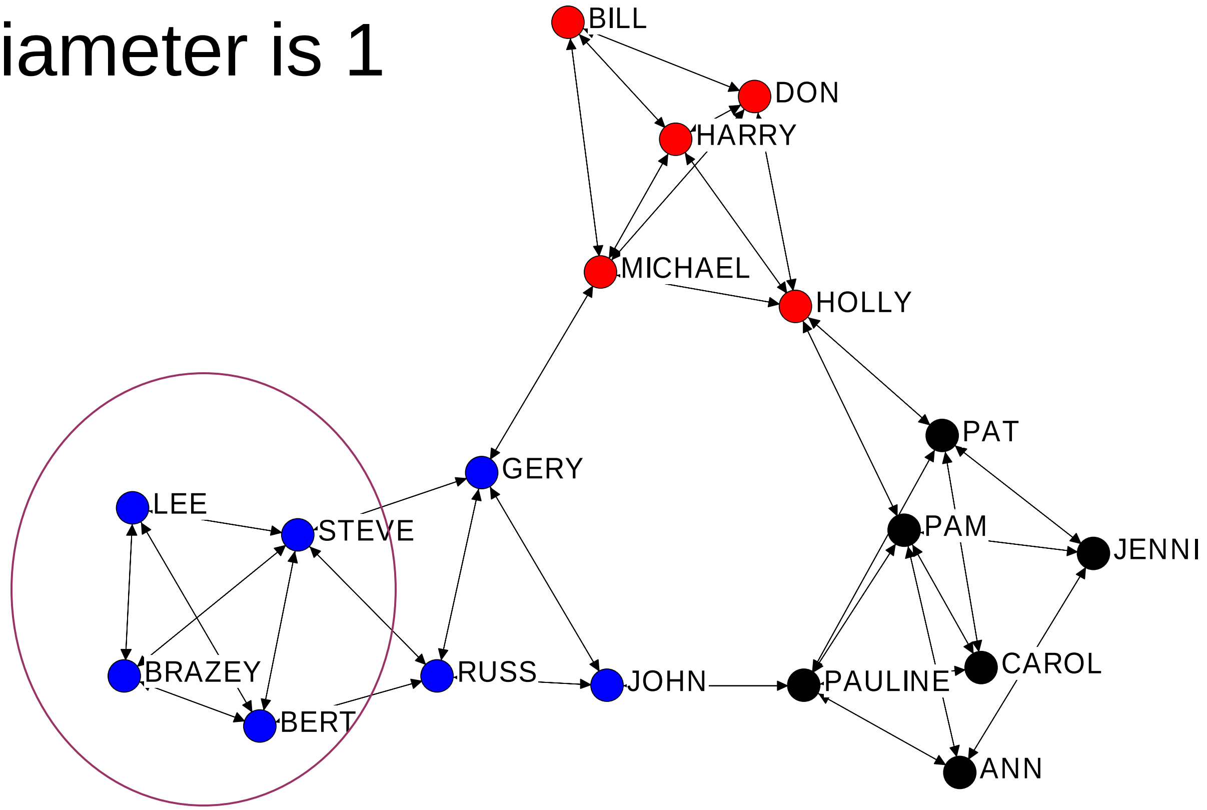
Components

- Maximally **connected** subgraph
  - In digraph there are strong and weak components:
    - Strong components mean everyone can reach everyone else, even when considering the one-way streets in the network
    - Weak components means, if we ignore the directionality of the ties, everyone is reachable by everyone else
    - A single weak component my comprise multiple strong components (pseudo-hierarchical, 2-levels)

Clique

- A maximal **complete** subgraph
  - – Everyone is adjacent to everyone else
  - – Distance & Diameter is 1
  - – Density is 1

- Limitations
  - – Undirected
  - – 3+ nodes

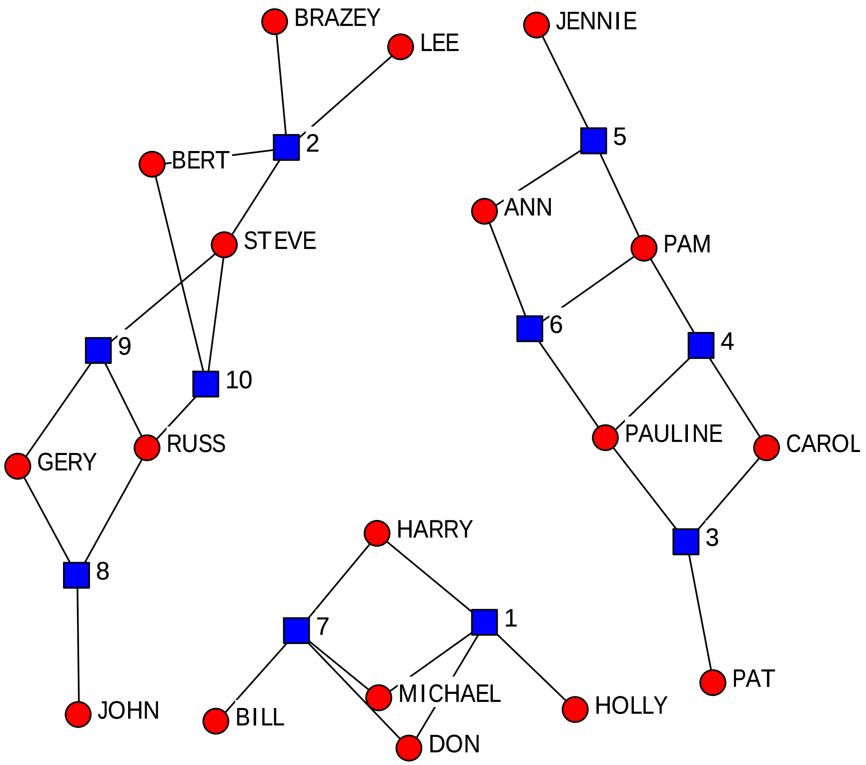


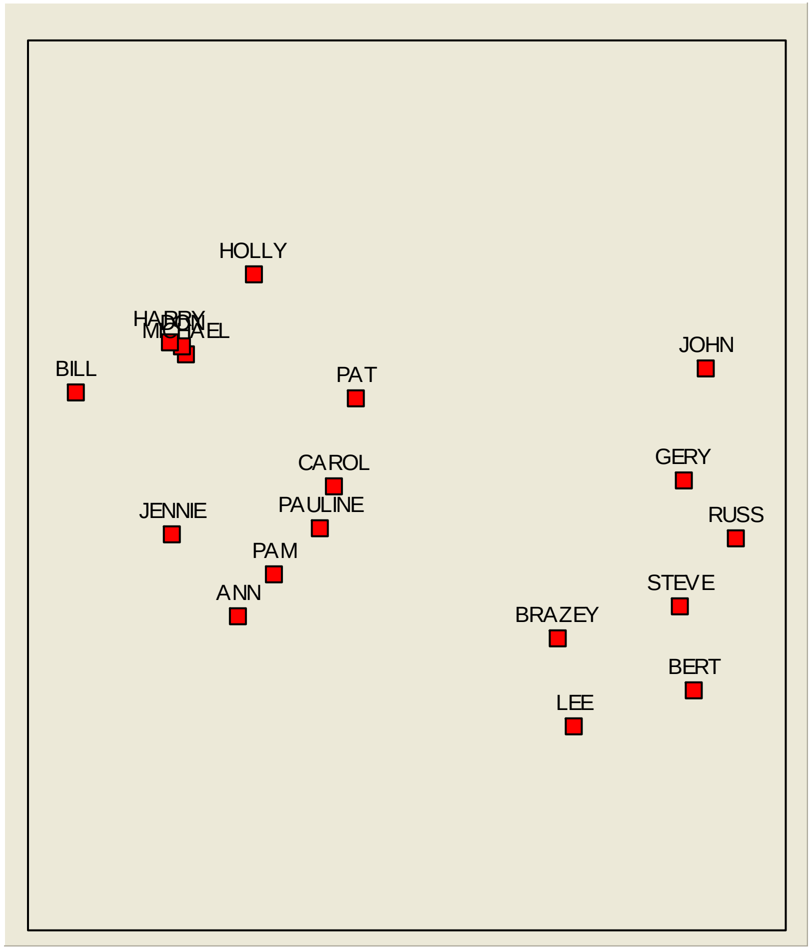
Problems with Cliques

- Can be too many or too few
- If too many:
  - Can put minimum on size
  - Can look at overlap
- If too few, relax requirements in terms of
  - Distance:
    - n-cliques, n-clans, n-clubs
  - Density
    - k-cores, k-plexes, ls-sets, lamba sets

If too many….

- Look at CliqueSets
  - 2-mode matrix in GLA
- Or CliqueOverlap
  - Square matrix in MDS
  - Or Hierarchical Cluster





Too Few, RELAX (Don't Do It) Distance Requirement

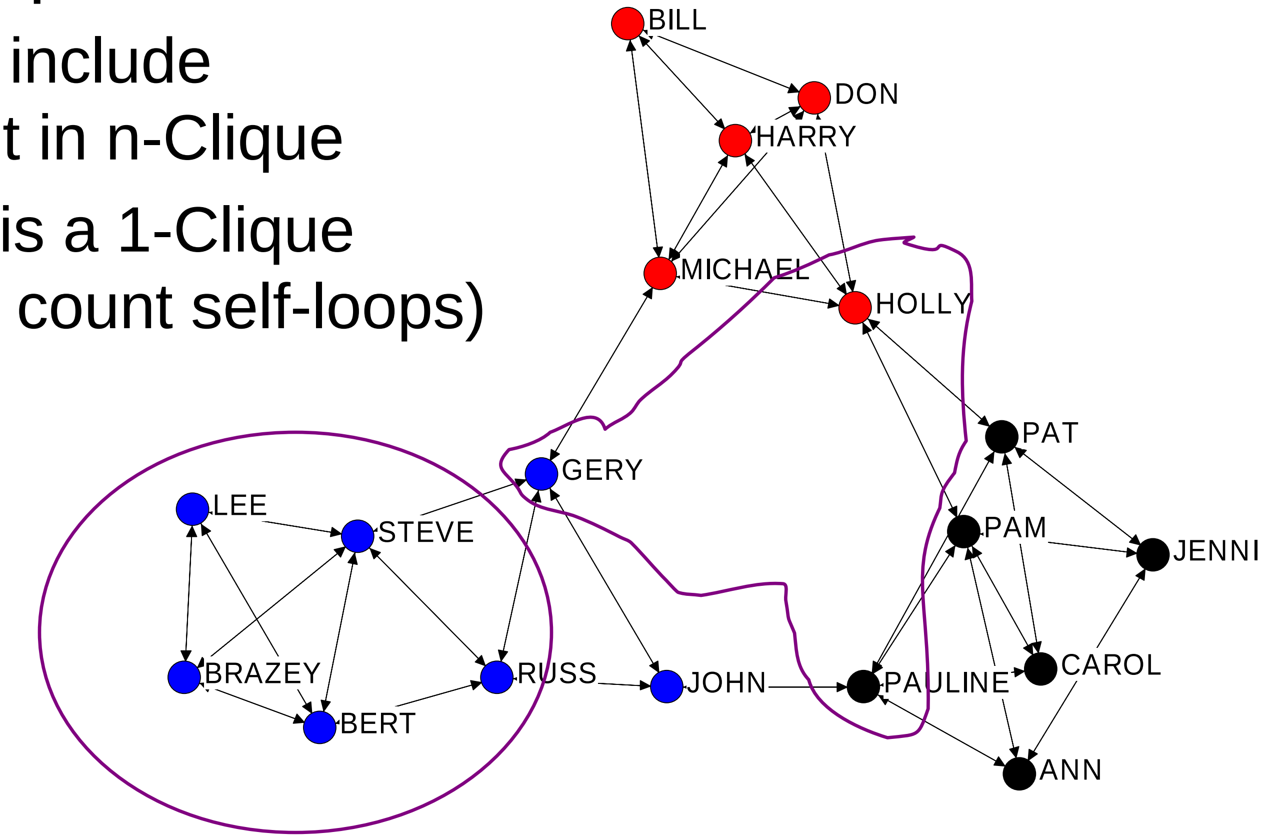
- n-Clique
  - Maximal subset with all nodes within n steps of each other
    - Path can include nodes not in n-Clique
    - A Clique is a 1-Clique (we don't count self-loops)

Is this a 2-Clique?

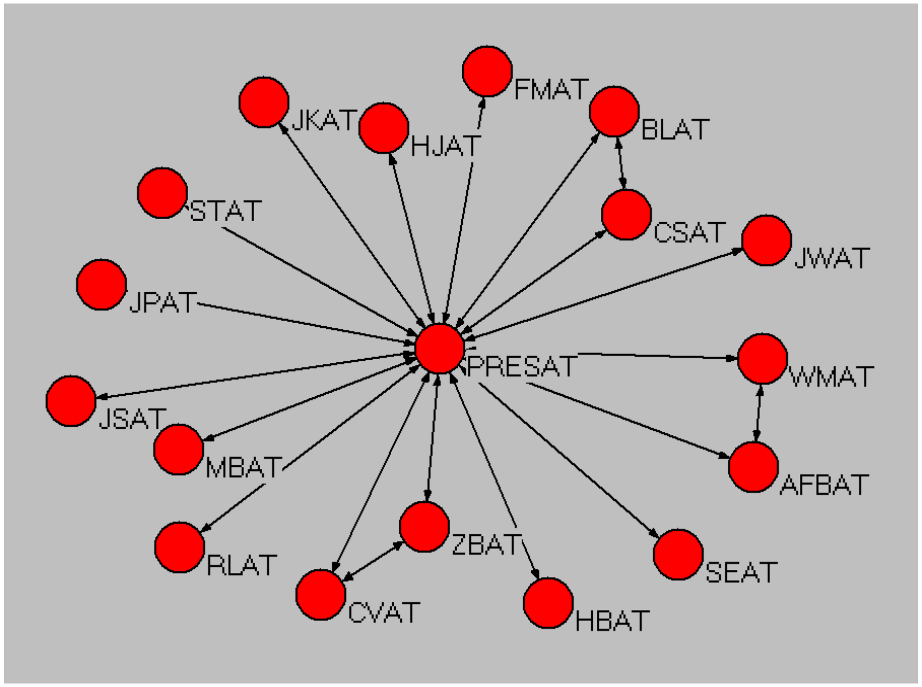
NO!

What about now?

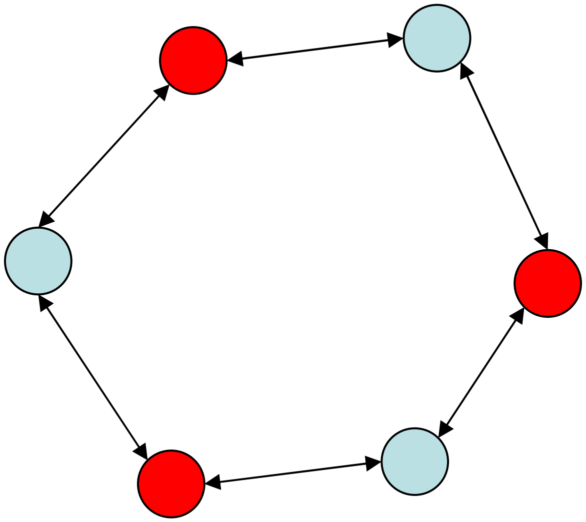
But so is this!!!



Some are counter-intuitive
(And not necessarily cohesive)



This is a 2-Clique

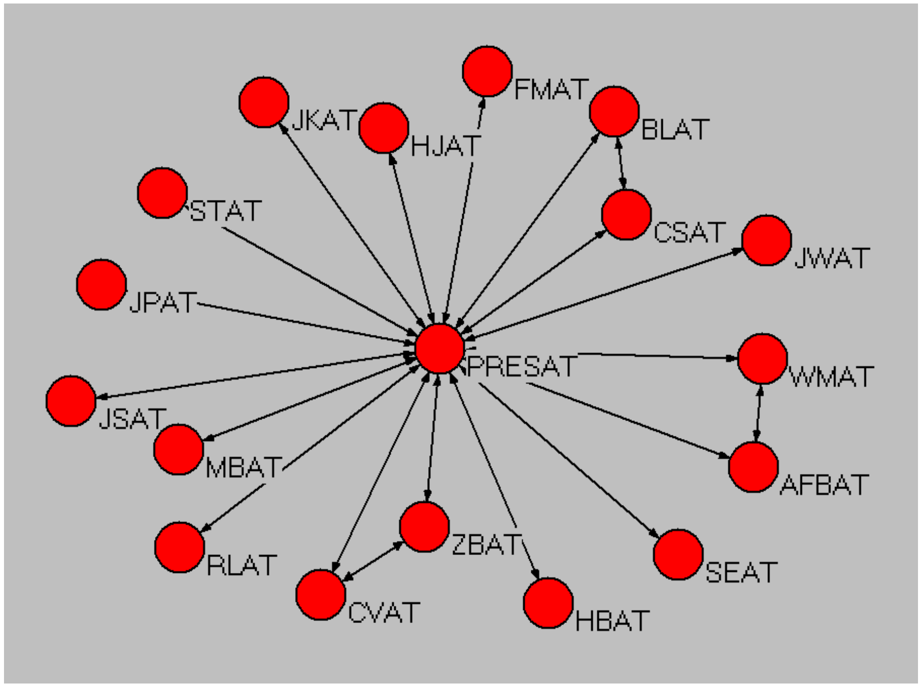


Red Nodes form a
2-Clique, so do Blues

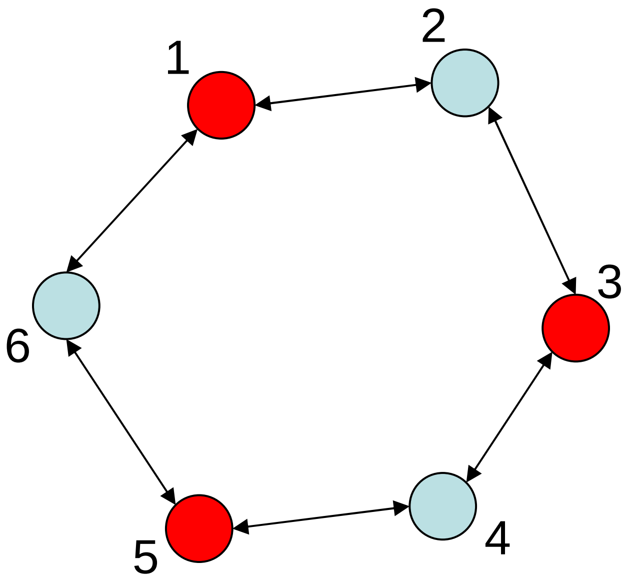
So, we can force more cohesion

- n-Clan is an n-Clique whose diameter in the subgraph induced from the nodes in the n-Clique is <= n
  - Don’t allow paths to go outside subset
- SUBGRAPH
  - A set of ties, together with ties among them
- An INDUCED SUBGRAPH
  - A subgraph defined by a set of nodes (or lines) and ALL the incident ties (or nodes)

2-Cliques vs. 2-Clans



This is a 2-Clique & a 2-Clan



{1,3,5} & {2,4,6} are
2-Cliques but not 2-Clans

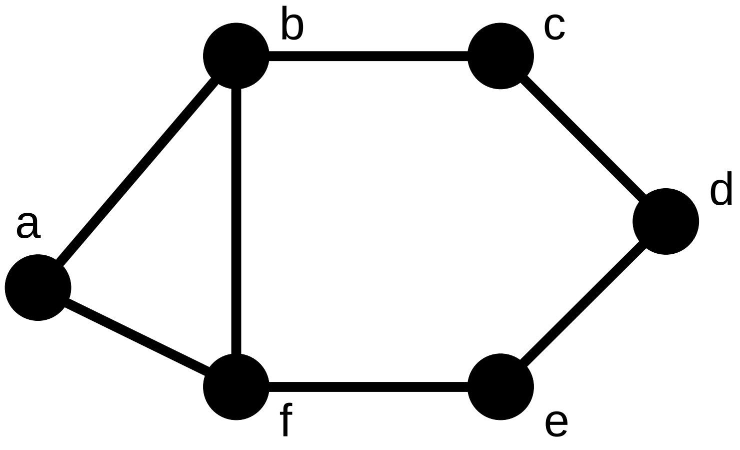
Clans can be overlapping => {1,2,3}, {2,3,4}, {3,4,5}, {4,5,6}, {5,6,1} & {6,1,2} are 2-Clans and 2-Cliques

But, n-Clans have issues, too

- The n-Clique requirement is restrictive, so there are few found in the data

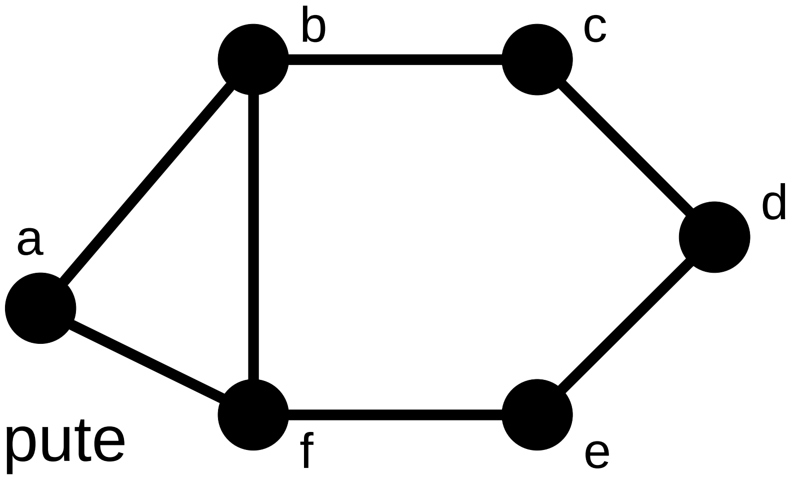
- Is {a,b,c,f} a 2-Clan?

- How many 2-Clans are there in this graph?



Loosening the restriction

- n-Clubs are, effectively, n-Clans that do not have the n-Clique requirement, or…
  - A maximal subset S such that the graph induced by the nodes S has a diameter <= n
  - Now {a,b,c,f} is a 2-Club, so is {a,b,e,f}



- Properties:
  - Painful (impossible) to compute
  - More plentiful than n-Clans
  - Overlapping

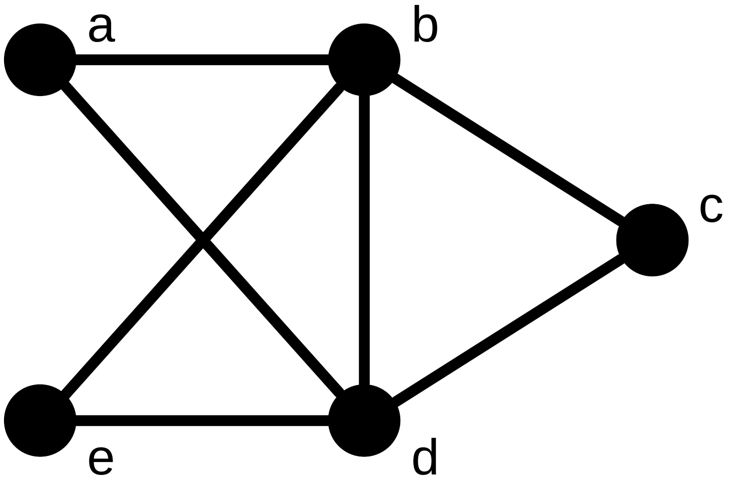
Another approach

- n-Cliques, n-Clans, and n-Clubs all start from the definition of Cliques and relax the distance requirement (all distances = 1) in varying ways
- But, Cliques also have maximum density (d = 1), and we can relax that definition instead…
- But for this, we must define the alpha operator, α, such that $\alpha(u,G)$ is the number of lines from node u to nodes in graph G

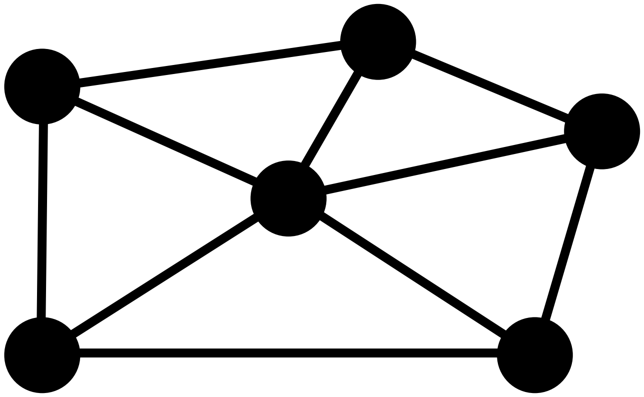
Relaxing the Density Requirement

- k-Plex
  - A clique where members don’t have to be connected to everyone else, just all but *k* members, or…
  - a [maximal] subset S s.t. for all *u* in S, α(*u*,S) >= |S|-k, where |S| is size of set S
    - All subsets of k-plexes are k-plexes (if non-maximal)
    - Get distance for free based on S, k.
      - If k < (|S|+2)/2 then diameter <= 2
    - Numerous & Overlapping
    - May be more intuitive than distance-based measures
    - A Clique is a 1-plex (missing itself)

K-Plex



Is {a,b,d,e} a 2-plex?
Is {a,b,c,d,e} a 2-plex?
Is {a,b,d} a 2-plex?



Is the graph as a whole a 2-plex?
Is it a 3-plex?

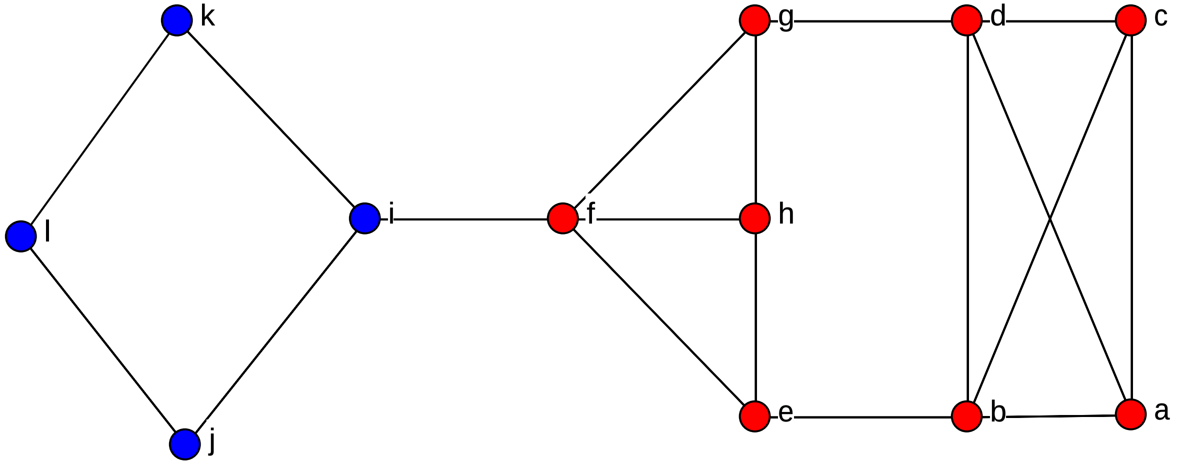
k-Core

- Sort of opposite approach from k-plex
  - Because the size of the group is not taken into account, k-cores are more directly about specifying how many ties MUST be present independent of how many nodes are in the core, whereas the k-plex is about both.

- A k-Core is maximal subgraph within which all nodes have ties to at least k other nodes
  - All nodes in a components are at least 1-Cores
  - Each nodes is assigned a “core” which is the largest k-core to which it belongs (and it therefore also belongs to all lower cores that exist)
  - K-cores are hierarchical and form a partition

Another definition

- A k-core is a maximal subgraph such that for all u in S, $\alpha(u,S) >= k$



  - All nodes are 2-core (and 1-core)
    Red nodes are 3-core.
- Great for analyzing large networks

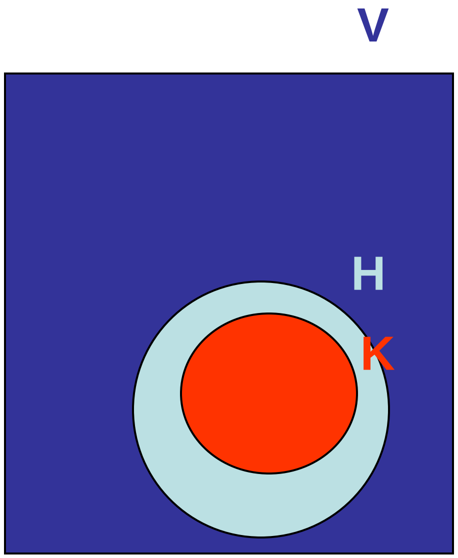
LS-Sets

- Definition
  - Given a graph G(V,E), let H be a subset of V, and let K be <u>any</u> proper subset of H
  - H is **LS** if α(K,H-K) > α(K,V-H) for all K
    - All subsets of the LS set are more connected to other LS members than outsiders of LS set

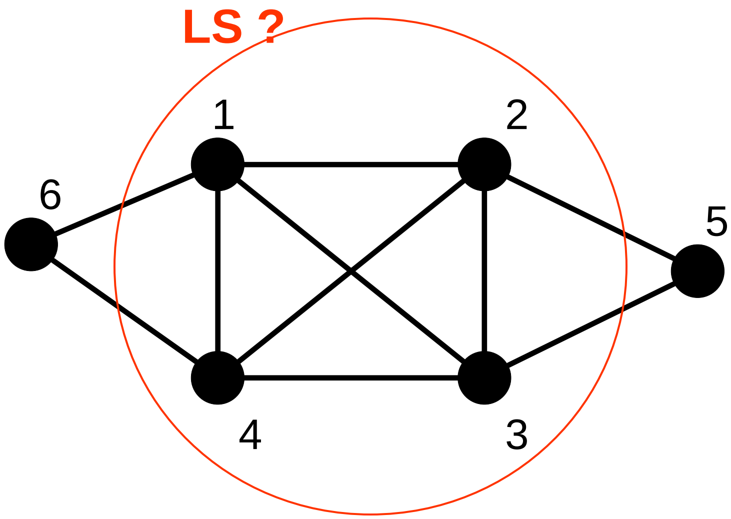
or…

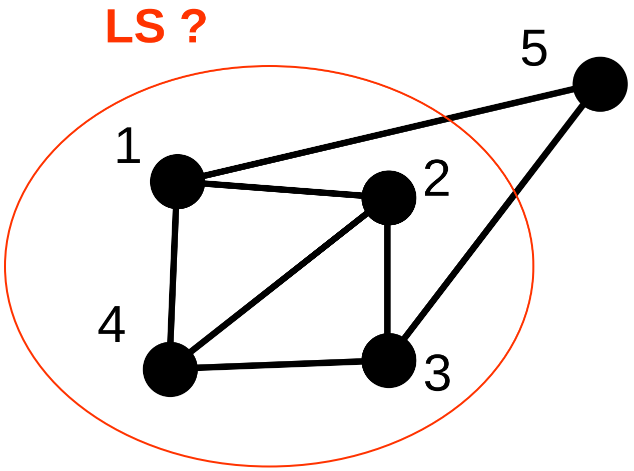
  - H is **LS** if α(K) > α(H),

    where α(K) => α(K,G-K)
    - Subsets better off joining LS set
    - This one's usually easier to compute



LS-Sets

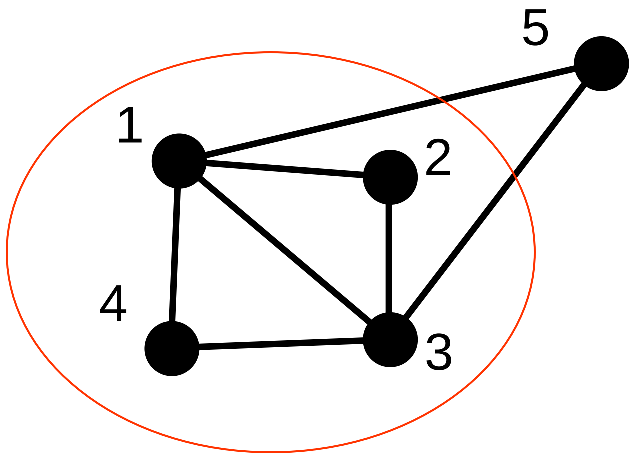




- H is LS if $\alpha(K,H-K) > \alpha(K,V-H)$
  - Use when K is large

or …

- H is LS if $\alpha(K) > \alpha(H)$
  - Use when K is small



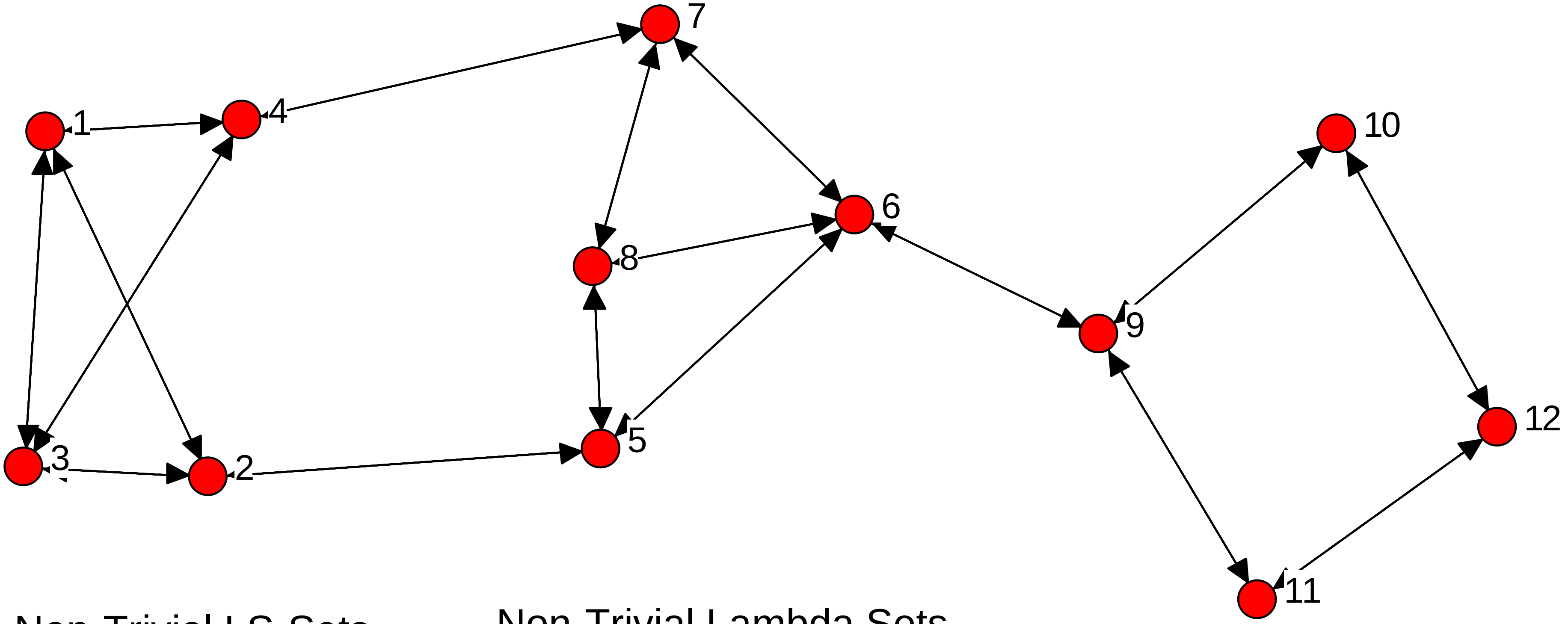
LS-Sets

- Properties – very cohesive
  - Wholly nested or disjoint
    - No partial overlaps
  - More ties within than between
    - Everyone more connected inside than outside
  - Contain no minimum weight cutsets
    - lie on either side of “fault lines”
  - Multiple edge-independent paths within
    - High edge-connectivity

Lambda Sets

- Definition
  - A set of nodes S is a lambda set if, for all, *a, b, c* in S and *d* not in S, $\lambda(a,b) > \lambda(c,d)$
    - where $\lambda(u,v)$ is the number of edge-independent paths from node u to node v , which is also the minimum number of ties that must be removed in order to disconnect u and v
  - Members of Lamba Sets have more independent paths to ALL other group members than ANY outsider
- Properties
  - Robust
    - very difficult to disconnect even with intelligent attack
  - Mutually exclusive or wholly inclusive
    - no partially overlapping groups
  - Pure
    - defined on a single attribute (edge connectivity)

Lambda Sets



Non-Trivial LS-Sets
{1,2,3,4}
{1,2,3,4,5,6,7,8}
{9,10,11,12}

Non-Trivial Lambda Sets
{1,2,3,4}
{1,2,3,4,5,6,7,8}
{9,10,11,12}
{5,6,7,8}

Groups w/specified characteristics, based on Proximities

	Group by defined Algorithm	Group by defined Characteristics
Network/ Graph Theory	Newman-Girvan	**Distance**: Component, Clique, n-clique, n-clan, n-club **Density**: Clique, k-core, k-plex, ls-set, lamba set (Core/Periphery)
Proximity /Distance	Hierarchical Clustering MDS K-Means	Factions (Core/Periphery) (Combinatorial Optimization)

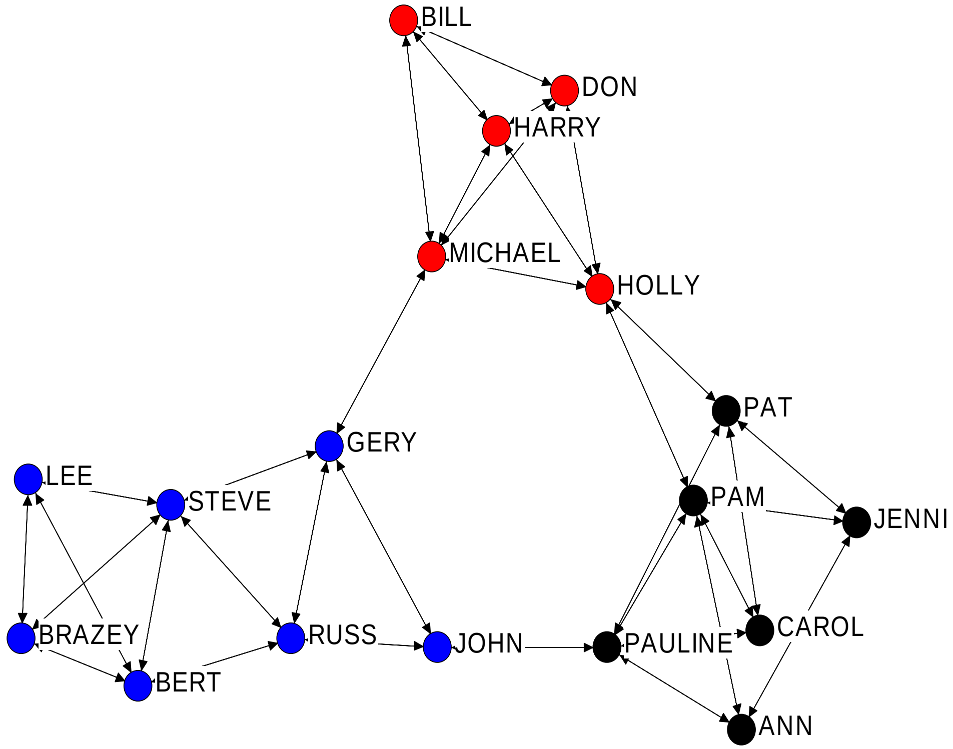
Factions

- Computationally arrange nodes into mutually exclusive groups such that some predefined criteria is optimized
  - For example, make groups that maximize density of internal ties and minimize density of external ties

Campnet Example

```
Group Assignments:

     1:   HOLLY MICHAEL BILL DON HARRY
     2:   CAROL PAM PAT JENNIE PAULINE ANN
     3:   BRAZEY LEE JOHN GERY STEVE BERT RUSS
```

```
                 1 1 1                         1 1   1 1 1 1
               1 0 2 4 9   4 6 8 7 5 3   1 1 3 2 5 6 7 8
               H B D H M   P J A P P C   L J B G S B R
              -----------------------------------------
 1    HOLLY | 1   1     | 1       1   |               |
10     BILL |   1 1 1 1 |             |               |
12      DON | 1   1 1 1 |             |               |
14    HARRY | 1   1 1 1 |             |               |
 9  MICHAEL | 1   1 1 1 |             |               |
            -------------------------------------------
 4      PAM |           | 1 1 1 1     |               |
 6   JENNIE |           | 1 1 1   1   |               |
 8      ANN |           | 1 1 1 1     |               |
 7  PAULINE |           | 1     1 1 1 |               |
 5      PAT | 1         |   1     1 1 |               |
 3    CAROL |           | 1     1 1 1 |               |
            -------------------------------------------
11      LEE |           |             | 1   1   1 1   |
13     JOHN |           |       1     |   1   1     1 |
 2   BRAZEY |           |             | 1   1   1 1   |
15     GERY |         1 |             |       1 1   1 |
16    STEVE |           |             | 1       1 1 1 |
17     BERT |           |             | 1       1 1 1 |
18     RUSS |           |             |       1 1 1 1 |
              -----------------------------------------
```



Core-Periphery Models

- A core periphery structure has a single cohesive subgroup with a set of other nodes, loosely connected to the core
- Core members interact with (lots of) other core members
- Peripheral members interact with (a few) core members

Finding Core/Periphery Structures

- Two ways to deal with it…
  - One is a special case of factions, which maximizes density of core-to-core relations and minimizes all others (categorical)
  - Another is a continuous model that calculates a “coreness” which is how much this node looks like a core node (continuous)