Emergent Groups:
Detecting an Emergent Subgroup

-Clumpiness
-Regions
-Subgroups
Transitivity

- Proportion of triples with 3 ties as a proportion of triples with 2 or more ties
  - Aka the clustering coefficient

\[
\text{cc} = \frac{2}{6} = 33\%
\]

\{C,T,E\} is a transitive triple, but \{B,C,D\} is not. \{A,D,T\} is not counted at all.
Network Regions
Network Regions

• Large “contiguous” areas
• Areas that contain cohesive subgroups
• We will cover:
  – Components
  – K-Cores
Graph Terminology

- A graph \( G(V,E) \) consists of a set of nodes \( V \) and a set of lines \( E \). Each line \( e \in E \) consists of a pair of nodes \((u,v)\)
- A graph \( G' \) is a subgraph of a graph \( G \) if every line in \( E(G') \) is in \( E(G) \), and every node in \( V(G') \) is in \( V(G) \).
- The subgraph \( S \) induced by a set of nodes consists of those nodes together with all ties among them.
Components

• A subgraph $S$ of a graph $G$ is a component if $S$ is maximal and connected
  – Connected means that every node can reach every other by some path (no matter how long)
Components in Digraphs

• If $G$ is a digraph, then
  – $S$ is a weak component if it is a component of the underlying (undirected) graph
    • i.e., we allow semi-paths rather than require true directed paths
  – $S$ is a strong component if for all $u, v$ in $S$, there is a path from $u$ to $v$
Notes on Components

• Isolates are (very small) components
• Finding components is often first step in analysis of large graphs
  – Often analyze each component separately, or discard very small components
  – Many network measures require a connected graph, so they don’t work on graphs with multiple components
Alpha Operator

• Let $\alpha(S_1,S_2)$ be the number of ties from members of set $S_1$ to members of the set $S_2$

• $\alpha(u,S)$ is number of ties node $u$ has with members of set $S$

• $\alpha(S) = \alpha(S,V-S)$ is number of ties from members of set $S$ to members of $V-S$ (i.e., all other nodes)
K-Core

- A subgraph $S$ is a $k$-core if for all $u \in S$, $\alpha(u,S) \geq k$, and $S$ is maximal.

- $S=G$ is 1-core & 2-core; $S = \{1..8\}$ is 3-core
- There is no 4-core or higher
K-Core Notes

• Finds areas within which cohesive subgroups may be found
• Identifies fault lines across which cohesive subgroups do not span
• In large datasets, you can successively examine the 1-cores, the 2-cores, etc.
  – Progressively narrowing to core of network
Cohesive Subgroups
Cohesive Subgroups

• Initially conceived of as formalizations of fundamental sociological concepts
  – Primary groups
  – Emergent groups

• Now typically thought of in terms of a technique for identifying groups within networks
Canonical Hypothesis

• Members of group will have similar outcomes
  – Ideas, attitudes, illnesses, behaviors
• Due to interpersonal transmission
  – transference
  – Influence / persuasion
  – Co-construction of beliefs & practices
    • As in communities of practice
• So group membership is independent var used to predict commonality of attitudes, beliefs, etc.
**Typology of Subgroups**

<table>
<thead>
<tr>
<th>Network / Graph theory</th>
<th>Process</th>
<th>Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Newman-Girvan</td>
<td>Clique, n-clique, n-clan, n-club, k-plex, ls-set, lambda-set, k-core, component</td>
</tr>
<tr>
<td>Proximities / Clustering</td>
<td>Johnson’s Hierarchical clustering; k-means; MDS</td>
<td>Factions, combinatorial optimization</td>
</tr>
</tbody>
</table>
Types of Approaches

Subgroups

- graph
  - Outcome
    - Distance
      - component
      - n-clique
      - n-clan
      - n-club
  - Density
    - k-core
    - k-plex
    - ls-set
    - lambda set

- proximities
  - Process
    - Newman-Girvan
    - Negopy
  - Process
    - Hiclus
    - Kmeans
  - Outcome
    - factions
    - comb opt
Subgroups

graph

Distance
- component
- n-clique
- n-clan
- n-club

Density
- k-core
- k-plex
- ls-set
- lambda set

Outcome
-- clique

Process
-- Newman-Girvan
-- Negopy

proximities

Process
- Hiclus
- Kmeans
- Newman

Outcome
-- factions
-- comb opt
Clique

- **Definition**
  - Maximal, complete subgraph
  - Set $S$ s.t. for all $u, v$ in $S$, $(u, v)$ in $E$

- **Properties**
  - Maximum density (1.0)
  - Minimum distances (all 1)
  - Overlapping
  - Strict

---

{c, d, e} is the only clique
10 cliques found.

1: HOLLY MICHAEL DON HARRY
2: BRAZEY LEE STEVE BERT
3: CAROL PAT PAULINE
4: CAROL PAM PAULINE
5: PAM JENNIE ANN
6: PAM PAULINE ANN
7: MICHAEL BILL DON HARRY
8: JOHN GERY RUSS
9: GERY STEVE RUSS
10: STEVE BERT RUSS
Types of Relaxations

• Distance (length of paths)
  – N-clique, n-clan, n-club

• Density (number of ties)
  – K-plex, ls-set, lambda set, k-core, component
N-cliques

• Definition
  – Maximal subset s.t. for all $u,v$ in $S$, $d(u,v) \leq n$
  – Distance among members less than specified maximum
  – When $n = 1$, we have a clique

• Properties
  – Relaxes notion of clique
    • Avg distance can be greater than 1

Is \{a,b,c,f,e\} a 2-clique? yes
10 2-cliques found.

1: HOLLY MICHAEL BILL DON HARRY GERY
2: MICHAEL JOHN GERY STEVE RUSS
3: PAULINE JOHN GERY RUSS
4: HOLLY PAULINE GERY
5: BRAZEY LEE GERY STEVE BERT RUSS
6: JOHN GERY STEVE BERT RUSS
7: HOLLY CAROL PAM PAT JENNIE PAULINE ANN
8: CAROL PAM PAT PAULINE ANN JOHN
9: HOLLY PAM PAT MICHAEL DON HARRY
10: PAM PAT MICHAEL JOHN
Issues with N-Cliques

• Overlapping
  – \{a,b,c,f,e\} and \{b,c,d,f,e\} are both 2-cliques

• Membership criterion satisfiable through non-members

• Even 2-cliques can be fairly non-cohesive
  – Red nodes belong to same 2-clique but none are adjacent
Subgraphs

- Set of nodes
  - Is just a set of nodes
- A subgraph
  - Is set of nodes together with ties among them
- An induced subgraph
  - Subgraph defined by a set of nodes
  - Like pulling the nodes and ties out of the original graph

Subgraph induced by \{a,b,c,f,e\}
N-Clan

• Definition
  – An n-clique S whose diameter in the subgraph induced by S is \( \leq n \)
  – Members of set within n links of each other without using outsiders

• Properties
  – More cohesive than n-cliques

Is \{a,b,c,f,e\} a 2-clan?
8 2-clans found.

1: HOLLY MICHAEL BILL DON HARRY GERY
2: MICHAEL JOHN GERY STEVE RUSS
3: PAULINE JOHN GERY RUSS
5: BRAZEY LEE GERY STEVE BERT RUSS
6: JOHN GERY STEVE BERT RUSS
7: HOLLY CAROL PAM PAT JENNIE PAULINE ANN
8: CAROL PAM PAT PAULINE ANN JOHN
9: HOLLY PAM PAT MICHAEL DON HARRY

2-Cliques that are not 2-Clans:

4: HOLLY PAULINE GERY
10: PAM PAT MICHAEL JOHN
N-Clan Issues

• n-clique membership a bother
  – Is \{a,b,c,f\} a 2-clan?
  – List all 2-clans
• few found in data
• overlapping
N-Club

• Definition
  – A maximal subset $S$ whose diameter in the subgraph induced by $S$ is $\leq n$
  – No $n$-clique requirement

• Properties
  – Painful to compute
  – More plentiful than $n$-clans
  – Overlapping

Is $\{a, b, c, f\}$ a 2-club?
K-Plexes

• Definition
  – A k-plex is a [maximal] subset $S$ s.t. for all $u$ in $S$, $\alpha(u,S) \geq |S|-k$, where $|S|$ is size of set $S$

• Properties
  – Subsets of k-plexes are k-plexes
  – Limited diameter (i.e., get distance as freebie)
    • If $k < (|S|+2)/2$ then diameter $\leq 2$
  – Very numerous & overlapping
  – Sometimes better match to intuition than distance relaxations
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?
LS-Sets

• Definition
  – Given a graph \( G(V,E) \), let \( H \) be a subset of \( V \), and let \( K \) be any proper subset of \( H \)
  – \( H \) is LS if \( \alpha(K,H-K) > \alpha(K,V-H) \) 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 \( \alpha(K) > \alpha(H) \)
    • 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 (doesn’t just consider density inside density)
  – Contain no minimum weight cutsets (lie on either side of “fault lines”)
  – Multiple edge-independent paths within
    • High edge-connectivity
Lambda Operator

• Let $\lambda(u,v)$ be the number of edge-independent paths from node $u$ to node $v$

• $\lambda(u,v)$ is also the minimum number of ties that must be removed from the network in order to disconnect $u$ and $v$
Lambda Sets

• Definition
  – A set of nodes $S$ is a lambda set if for all $a, b, c$ in $S$ and $d$ not in $S$, $\lambda(a,b) > \lambda(c,d)$
    • More independent paths to other group members than to outsiders

• Properties
  – Robust
    • very difficult to disconnect even with intelligent attack
  – Mutually exclusive or wholly inclusive
    • No partially overlapping groups
  – Pure – like n-clubs, defined on a single attribute
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}
Subgroups

graph clustering

Outcome
-- clique
  Distance
- component
- n-clique
- n-clan
- n-club
  Density
- k-core
- k-plex
- ls-set
- lambda set

Process
-- Newman-Girvan
-- Negopy

clustering

Outcome
-- factions
-- comb opt

Process
- Hiclus
- Kmeans
Newman-Girvan

- Successively deleting the tie with the most edge betweenness, and identifying components, then recalculating betweenness
- Yields a hierarchical clustering
Proximities / Clustering and Scaling Methods

• First compute dyadic cohesion matrix
  – E.g. geodesic distance

• Then cluster or scale
  – Two major kinds of clustering routines
    • Process-defined
    • Outcome-defined

• Typical result is a partition
Partitions

• Partition P is just an assignment of nodes to classes
  – P(i) gives the class of node i
  – Every node assigned to one & only one class

• A partition P is nested in partition M if for all nodes i and j, P(i)=P(j) implies M(i)=M(j)

• Trivial partitions
  – Identity: P(i) = i for all i
  – Complete: P(i) = 1 for all i
Process-Defined Clustering

• Heuristic definitions
  – Multivariate methods
    • Johnson’s hierarchical
    • Wards
    • K-means
  – Graph-theoretic / Network methods
    • Newman-Girvan

• Sometimes specify number of groups a priori, sometimes not
Subgroups

Graph clustering

Outcome
- clique

Distance
- n-clique
- n-clan
- n-club

Density
- k-core
- k-plex
- ls-set
- lambda set
- component

Process
- Newman-Girvan
- Negopy

Process
- Hiclus
- Kmeans

Outcome
- factions
- comb opt
Johnson’s Hierarchical Clustering

• Output is a set of nested partitions, starting with identity partition and ending with the complete partition

• Different flavors based on how distance from a point to a cluster is defined
  – Single linkage; connectedness; minimum
  – Complete linkage; diameter; maximum
  – Average, median, etc.
Closest distance is NY-BOS = 206, so merge these.
<table>
<thead>
<tr>
<th></th>
<th>BOS</th>
<th>DC</th>
<th>MIA</th>
<th>CHI</th>
<th>SEA</th>
<th>SF</th>
<th>LA</th>
<th>DEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOS/ NY</td>
<td>0</td>
<td>223</td>
<td>1308</td>
<td>802</td>
<td>2815</td>
<td>2934</td>
<td>2786</td>
<td>1771</td>
</tr>
<tr>
<td>DC</td>
<td></td>
<td></td>
<td>1075</td>
<td>671</td>
<td>2684</td>
<td>2799</td>
<td>2631</td>
<td>1616</td>
</tr>
<tr>
<td>MIA</td>
<td>1308</td>
<td>1075</td>
<td>0</td>
<td>1329</td>
<td>3273</td>
<td>3053</td>
<td>2687</td>
<td>2037</td>
</tr>
<tr>
<td>CHI</td>
<td>802</td>
<td>671</td>
<td>1329</td>
<td>0</td>
<td>2013</td>
<td>2142</td>
<td>2054</td>
<td>996</td>
</tr>
<tr>
<td>SEA</td>
<td>2815</td>
<td>2684</td>
<td>3273</td>
<td>2013</td>
<td>0</td>
<td>808</td>
<td>1131</td>
<td>1307</td>
</tr>
<tr>
<td>SF</td>
<td>2934</td>
<td>2799</td>
<td>3053</td>
<td>2142</td>
<td>808</td>
<td>0</td>
<td>379</td>
<td>1235</td>
</tr>
<tr>
<td>LA</td>
<td>2786</td>
<td>2631</td>
<td>2687</td>
<td>2054</td>
<td>1131</td>
<td>379</td>
<td>0</td>
<td>1059</td>
</tr>
<tr>
<td>DEN</td>
<td>1771</td>
<td>1616</td>
<td>2037</td>
<td>996</td>
<td>1307</td>
<td>1235</td>
<td>1059</td>
<td>0</td>
</tr>
</tbody>
</table>

Closest pair is DC to BOSNY combo @ 223. So merge these.
<table>
<thead>
<tr>
<th></th>
<th>BOS/NY/DC</th>
<th>MIA</th>
<th>CHI</th>
<th>SEA</th>
<th>SF</th>
<th>LA</th>
<th>DEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIA</td>
<td>0</td>
<td>1075</td>
<td>671</td>
<td>2684</td>
<td>2799</td>
<td>2631</td>
<td>1616</td>
</tr>
<tr>
<td>CHI</td>
<td>1075</td>
<td>0</td>
<td>1329</td>
<td>3273</td>
<td>3053</td>
<td>2687</td>
<td>2037</td>
</tr>
<tr>
<td>SEA</td>
<td>2684</td>
<td>3273</td>
<td>2013</td>
<td>0</td>
<td>808</td>
<td>1131</td>
<td>1307</td>
</tr>
<tr>
<td>SF</td>
<td>2799</td>
<td>3053</td>
<td>2142</td>
<td>808</td>
<td>0</td>
<td>379</td>
<td>1235</td>
</tr>
<tr>
<td>LA</td>
<td>2631</td>
<td>2687</td>
<td>2054</td>
<td>1131</td>
<td>379</td>
<td>0</td>
<td>1059</td>
</tr>
<tr>
<td>DEN</td>
<td>1616</td>
<td>2037</td>
<td>996</td>
<td>1307</td>
<td>1235</td>
<td>1059</td>
<td>0</td>
</tr>
</tbody>
</table>
### Geodesic Distances

<table>
<thead>
<tr>
<th></th>
<th>HOLLY</th>
<th>BRAZEY</th>
<th>CAROL</th>
<th>PAM</th>
<th>PAT</th>
<th>JENNIE</th>
<th>PAULINE</th>
<th>ANN</th>
<th>MICHAEL</th>
<th>BILL</th>
<th>LEE</th>
<th>DON</th>
<th>JOHN</th>
<th>HARRY</th>
<th>GERY</th>
<th>STEVE</th>
<th>BERT</th>
<th>RUSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>11</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>6</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>13</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>15</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Hierarchical Clustering

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  XXXXXXX XXXXXXX XXXXXXX XXXXXXXXXXXXX
1.481  XXXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXX
2.723  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
3.142  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
Subgroups

Graph clustering

Outcome
-- clique

Distance
- n-clique
- n-clan
- n-club

Density
- k-core
- k-plex
- ls-set
- lambda set
- component

Process
-- Newman-Girvan
-- Negopy

Clustering

Process
- Hiclus
- Kmeans

Outcome
-- factions
-- comb opt
Factions

• Outcome-Defined Clustering
• Input is proximity matrix X
  – Could be similarities or distances
• Assign items to clusters such that
  – For similarities, maximize similarities within cluster while minimizing similarities between clusters
  – For distances, minimize distance within cluster while maximizing distances between clusters
• Optimize explicit fitness function
  – Correlation with idealized image matrix
• Typically choose # of groups *a priori*
# Factions

<table>
<thead>
<tr>
<th></th>
<th>5</th>
<th>6</th>
<th>3</th>
<th>4</th>
<th>7</th>
<th>8</th>
<th>2</th>
<th>3</th>
<th>6</th>
<th>1</th>
<th>7</th>
<th>5</th>
<th>0</th>
<th>4</th>
<th>2</th>
<th>1</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PAT</td>
<td>JENNIE</td>
<td>CAROL</td>
<td>PAM</td>
<td>PAULINE</td>
<td>ANN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1 1 1</td>
<td>1 1 1</td>
<td>1 1</td>
<td>1 1 1</td>
<td>1 1 1</td>
<td>1 1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BILL</td>
<td>HARRY</td>
<td>DON</td>
<td>BERT</td>
<td>GERY</td>
<td>LEE</td>
<td>STEVE</td>
<td>JOHN</td>
<td>BRAZEY</td>
<td>RUSS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---