

# Making Personal Network Analysis More Accessible

Bernie Hogan, Wojciech Gryc, Jeffrey Wong and Barry Wellman

NetLab, University of Toronto

<http://individual.utoronto.ca/berniehogan/>

24th HCIL Symposium

University of Maryland, College Park



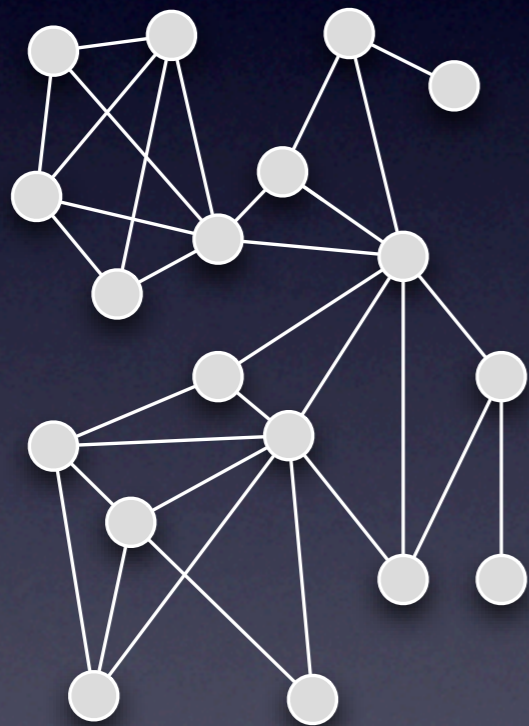
# Overview

- What is personal network analysis?
- How are these network captured / encoded?
- Existing approaches to network analysis
- NetLab tools: *Guess+GraphModifier* & *Egotistics*

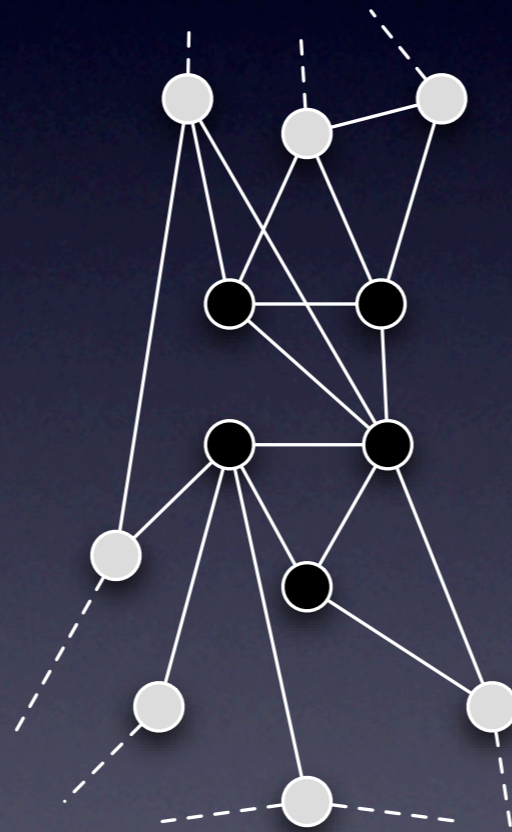


# Three Network Types

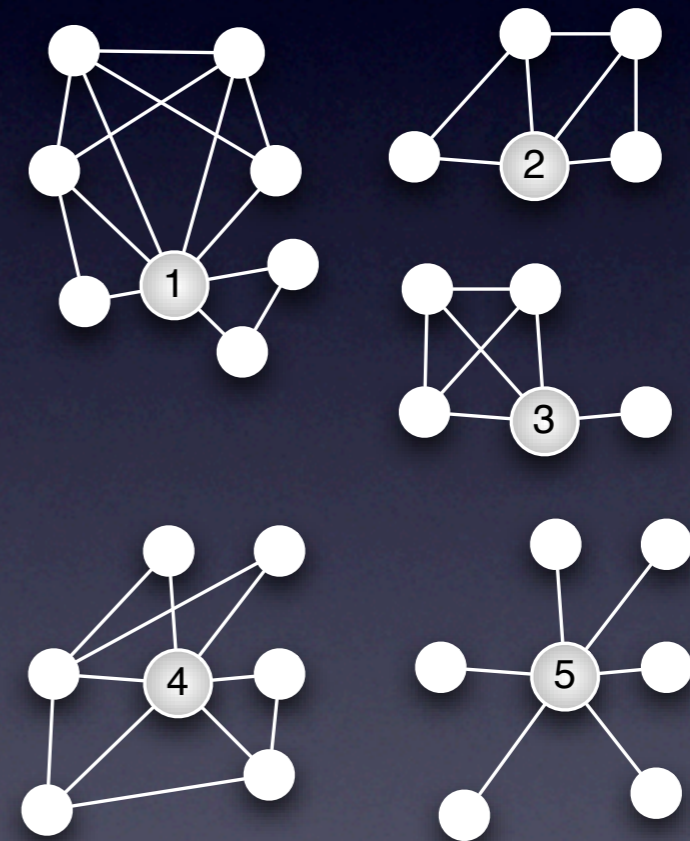
Whole Network



Partial Network



Five Ego Networks





# Personal Network Analysis

- Common method
- Goal: Compare networks of sampled individuals
- Typically only measured from one point of view - the respondent (a cognitive social network)
- Individuals recall trends such as frequency better than specific dates or occasions



# Common Questions

- Does the egonet of a smoker contain disproportionate smokers? (McCarty 2000)
- Do people get important advice from a single core group (of perhaps kin) or is advice scattered across varied groups? (Wellman and Wortley, 1990)
- How important am I in controlling the flow of information between parts of my network? (Burt 1992)

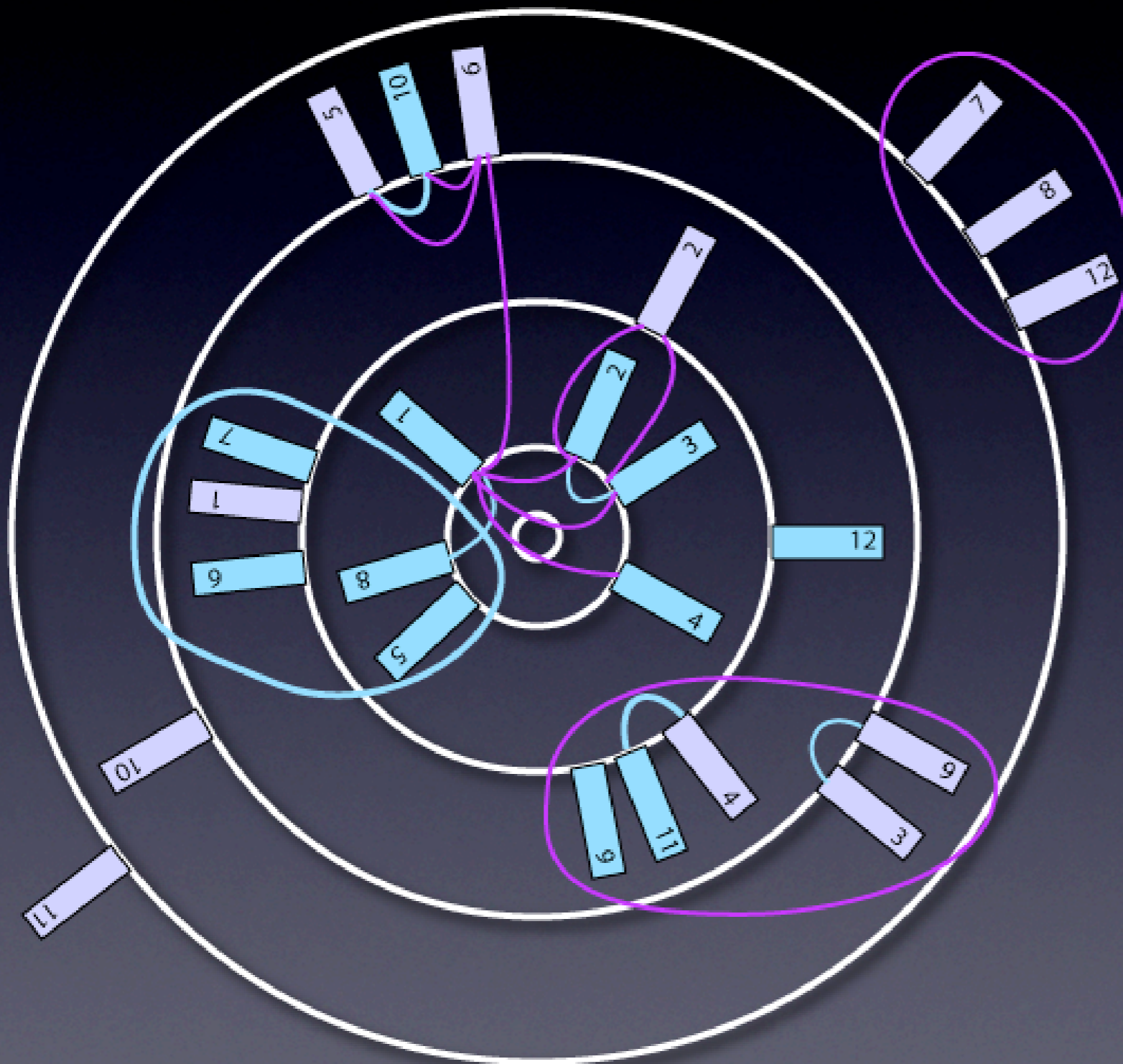


# Collecting Egonet Data

- Name Generators
  - ▶ Recall based on boundary:
    - *Everyone you called last week*
  - ▶ Recall based on prompts:
    - Who could babysit?
    - Who could loan you \$500?
- Online spiders/scrapers



# Participant-Aided Sociogram



## The Background:

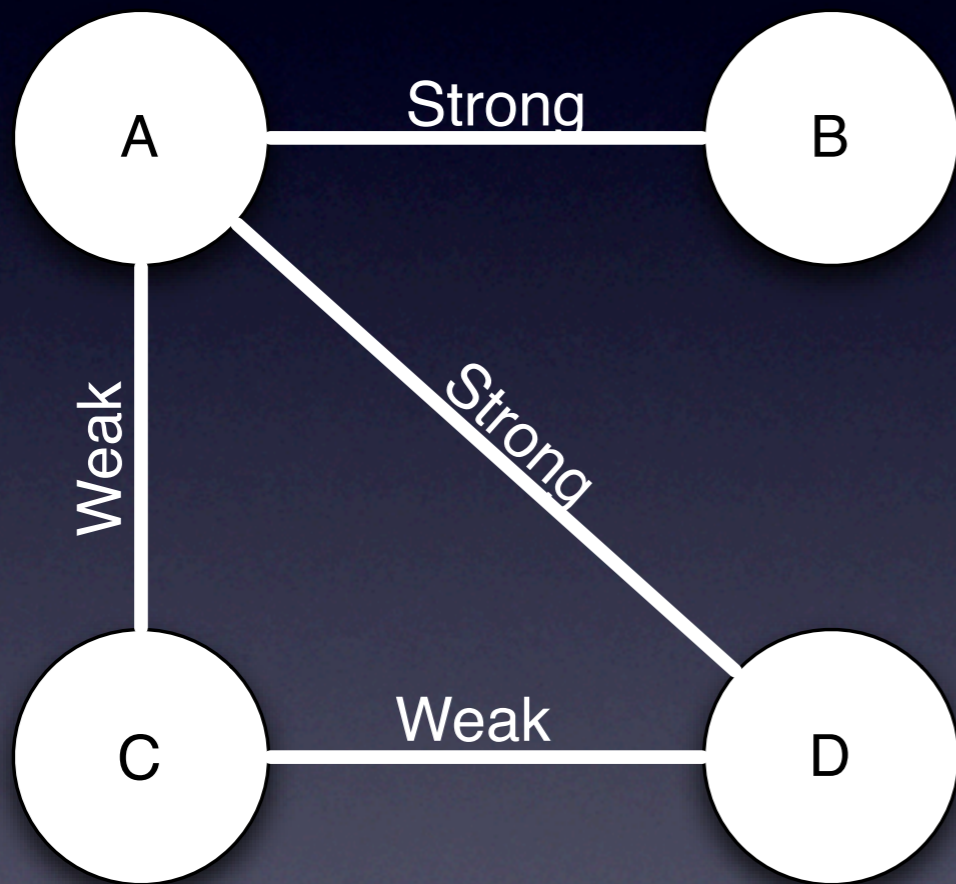
Wellman, Barry and Bernie Hogan with K. Berg and J. Boase and J.A. Carrasco and R. Cote and J. Kayahara and T. L. M. Kennedy and P. Tran. 2004. *Connected Lives: The Project*. In Purcell, P. ed. *The Networked Neighbourhood*. Spring. Pp. 161-216

## The Method:

Hogan, Bernie, Juan Antonio Carrasco and Barry Wellman. "Visualizing Personal Networks: Working With Participant-Aided Sociograms." *Field Methods*. 19(2). Pp 116-144.



# From Responses to Data



	A	B	C	D
A	-	2	1	2
B	2	-	0	0
C	1	0	-	1
D	2	0	1	-



# But not just once...

- This process is done with every respondent
- This means we start talking about three levels of data:
  - ▶ Alter level - each node in each network
  - ▶ Network level - each set of links in the data set
  - ▶ Ego level - each respondent in the data set



# Herein lies the problem

- Conventional applications are designed for a single network
  - ▶ *UCInet+NetDraw, Pajek*
- Software packages / environments have no GUI and steep learning curve
  - ▶ *R (sna package), JUNG*
- Solutions for multiple networks should make iterated processes efficient:
  - ▶ Templates / Batch modes / Interactivity

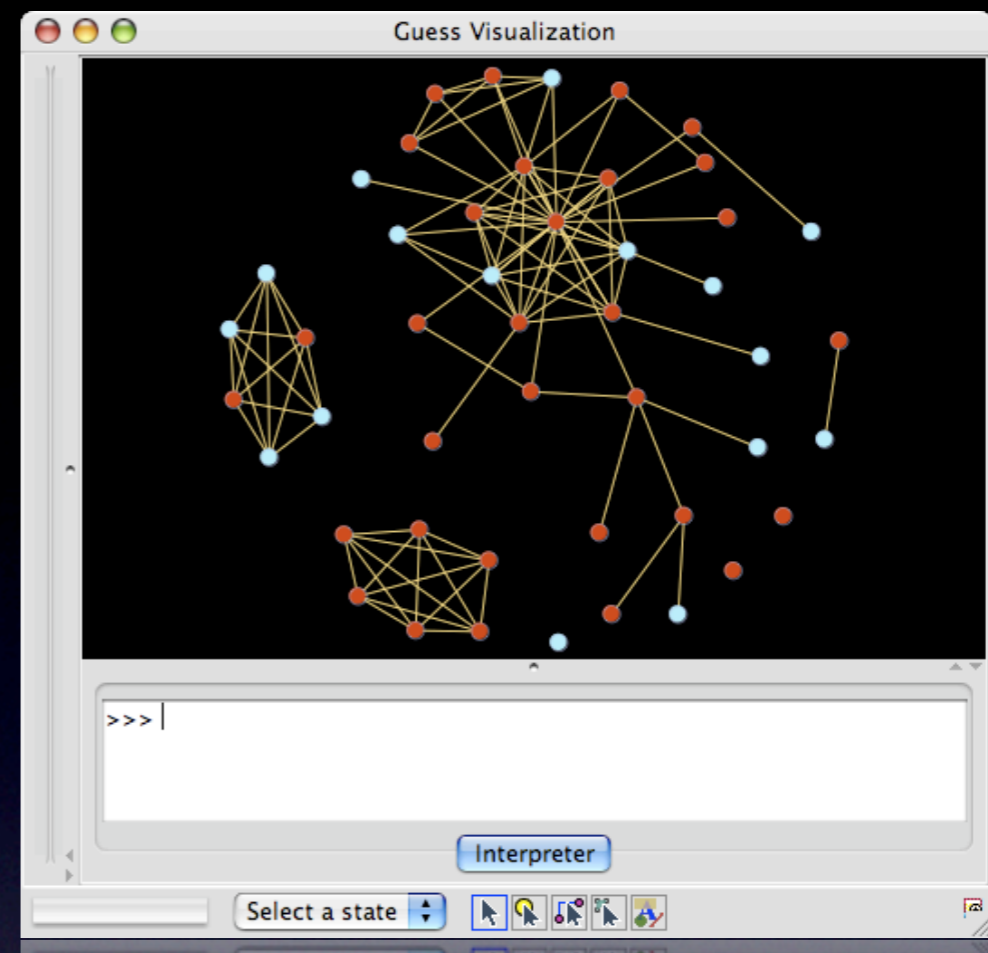


# Visualization Solution: Templates

- Enables the same style to easily be reapplied to networks with the same data structure.
- We use python scripts auto-generated by a point-and-click panel (which are subsequently editable).
- Done in GUESS with GraphModifier
  - ▶ GUESS was coded by Eytan Adar at HP Labs
    - Uses the Piccolo 2D toolkit (from HCIL)
  - ▶ GraphModifier was coded by Jeffrey Wong under my supervision.



# Guess + GraphModifier



The 'Graph Modifier' window has a configuration section at the top with the following settings:

- Object: odes based on ->
- Property: gender
- Operator: ==
- Value: f

Below the configuration are several buttons: Colour, Show, Hide, Size, Show Label, Hide Label, Change Label, Center (highlighted), and Change History.

A list of actions is shown in a scrollable area:

- center graph
- show all nodes whose property: close == s
- hide all nodes whose property: close == s
- show all nodes whose property: close == s
- change label all nodes whose property: gender == f to Female
- show label all nodes whose property: gender == f
- hide label all nodes whose property: gender == f (highlighted)
- center graph

At the bottom right, there are two buttons: Export Change History and Export GDF.

center graph  
show all nodes whose property: close == s  
hide all nodes whose property: close == s  
show all nodes whose property: close == s  
change label all nodes whose property: gender == f to Female  
show label all nodes whose property: gender == f  
hide label all nodes whose property: gender == f  
center graph

Export Change History

Export GDF



# Analysis Solution: Interactivity

- Batch mode is not enough. It assumes one particular workflow, and can lead to a bloated data set (batch every measure, every time)
- Results should be fed back into data set, rather than shown as isolated output.
- Done in Egotistics
  - ▶ Coded by Wojciech Gryc with help from myself and Elana Koren.



# Egotistics

Fresh From NetLab  
to your Lab.

The screenshot shows the Egotistics software interface. The title bar reads "Project: Egotistics\_demoOut - Egotistics - Version 0.5.2". The menu bar includes "File", "Data", "Ego Measures", "Alter Measures", "Transformations", and "Help". The toolbar contains icons for file operations and help. The main window has tabs for "Console", "Ego Attributes", "Alter Attributes", and "Network". The "Alter Attributes" tab is active, displaying a table with columns: "39:SS: Give", "40:SS: Give", "41:SS: Give", "42:SS: Give", "Degree", and "Betweenness". The table contains 20 rows of data. To the right, the "Project Properties" section has two unchecked checkboxes: "Append connected ego." and "Remove isolates.". Below that, the "Ego Net Selection" section shows a list of ego IDs: 1: 11, 2: 13, 3: 16, 4: 17 (highlighted), 5: 18, and 6: 21. Below the list, it says "Viewing: 4". At the bottom, the console shows the message "Finished getting betweenness." repeated twice.

	39:SS: Give	40:SS: Give	41:SS: Give	42:SS: Give	Degree	Betweenness
0	0	0	0	0	3	0.0
0	0	0	0	0	5	1.52380
0	0	0	0	1	5	1.40000
1	0	0	0	1	5	1.40000
0	0	0	0	1	5	1.40000
0	0	0	0	0	8	13.6857
0	0	0	0	0	7	5.28571
0	0	0	0	1	8	60.4286
0	0	0	0	1	3	16.5
0	0	0	0	0	0	0.0
0	0	0	0	1	1	0.0
0	0	0	0	1	1	0.0
0	0	0	0	0	0	0.0
0	0	0	0	0	6	17.3529
0	0	0	0	1	5	1.52380
1	0	0	0	0	1	0.0
0	0	0	0	0	1	0.0
0	0	0	0	1	1	0.0
0	0	0	0	1	0	0.0
0	0	0	0	0	3	16.5



# Summary

- Personal networks have specific needs for efficient repetition of tasks
- We use templates and interactivity to speed up the workflow
- The goal is not necessarily to raise the peak of the learning curve, but make it easier to climb
- Our software is open-source and generally GPL.
- GraphModifier 1.0 is available now. Egotistics is still in beta ([www.egotistics.net](http://www.egotistics.net)).



# Thank You

Bernie Hogan  
Research Coordinator, NetLab  
PhD Candidate, University of Toronto

[bernie.hogan@utoronto.ca](mailto:bernie.hogan@utoronto.ca)  
<http://individual.utoronto.ca/berniehogan/>

Thanks to our sponsors:  
KMDI Project Open Source / Open Access,  
Intel People and Practices,  
Bell University Labs,  
SSHRC