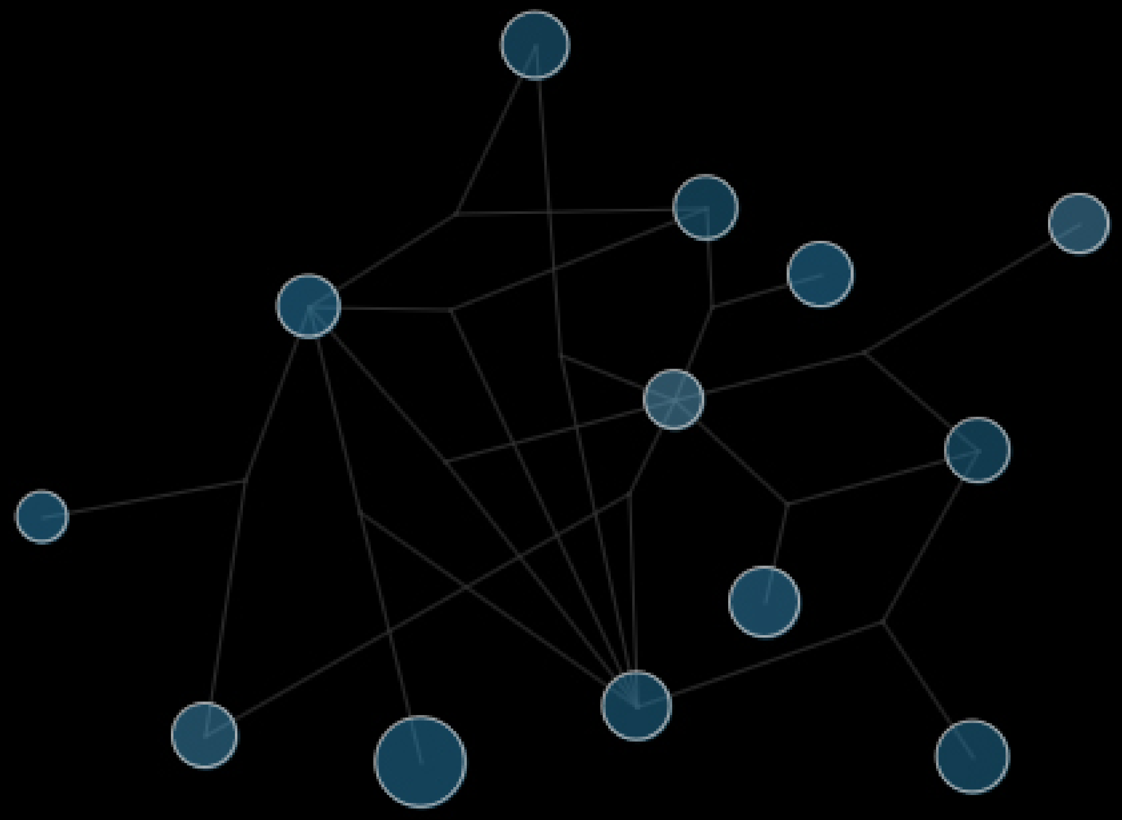
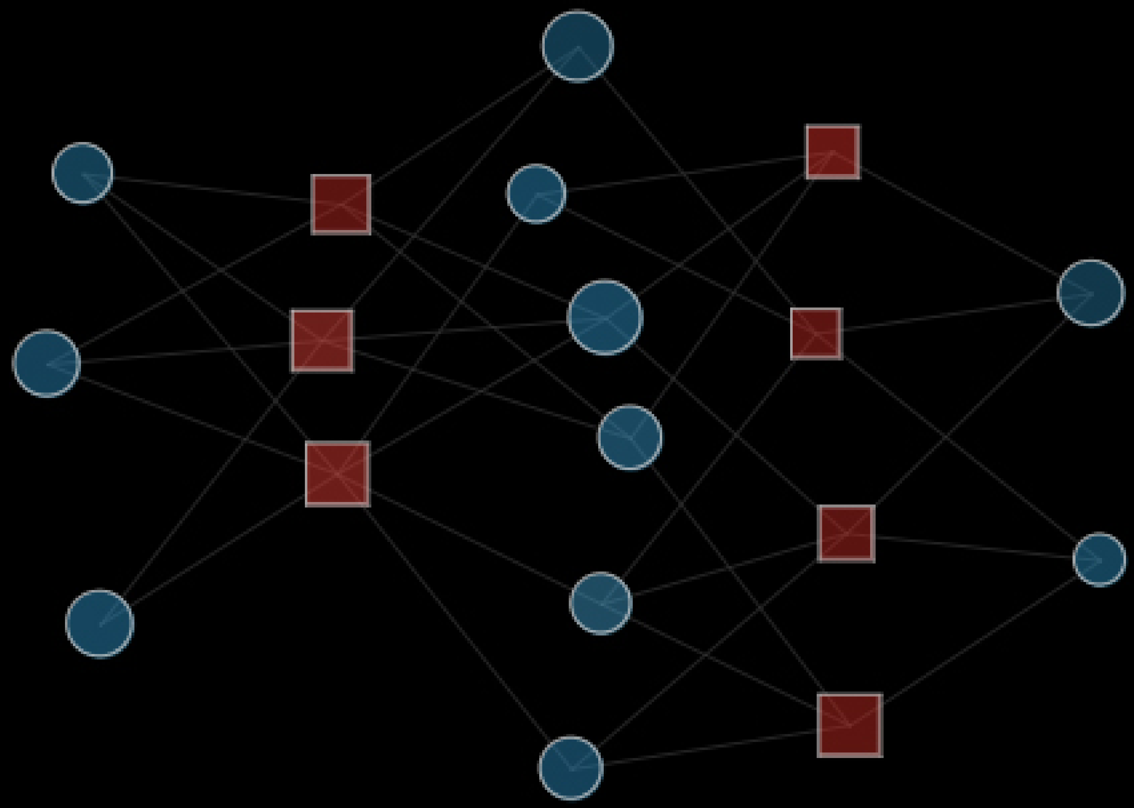
Methods for Hypergraphs and k-Partite Graphs

Aaron L Bramson





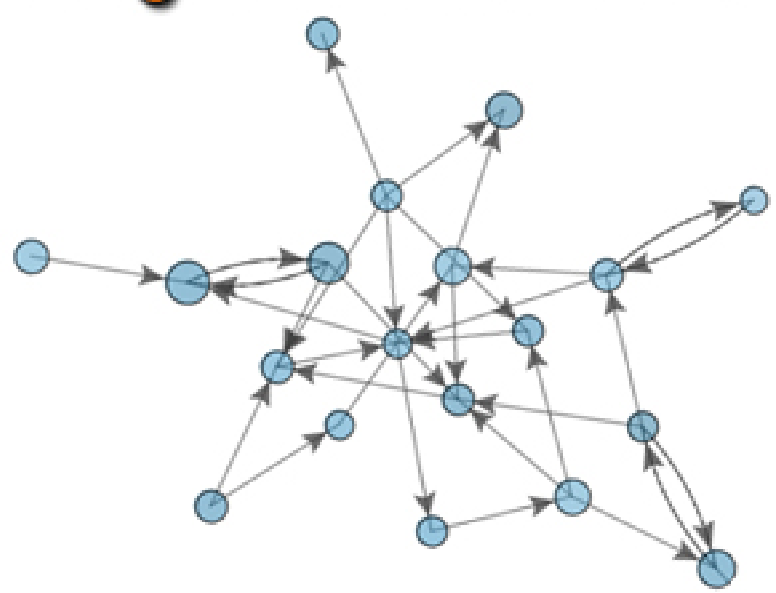
There Is So Much Network Research

- Network theory, analysis, and models are all playing increasingly important roles in research spanning nearly every discipline.
- The breadth of applications results from:
  1) the power of the network representation;
  2) the ability to generate and manipulate networks;
  3) the intuitive visualizations producible from networks; and
  4) the availability of powerful software for networks.
- Network research has nevertheless included mostly only the simplest network constructions: single-mode, flat, dyadic graphs.

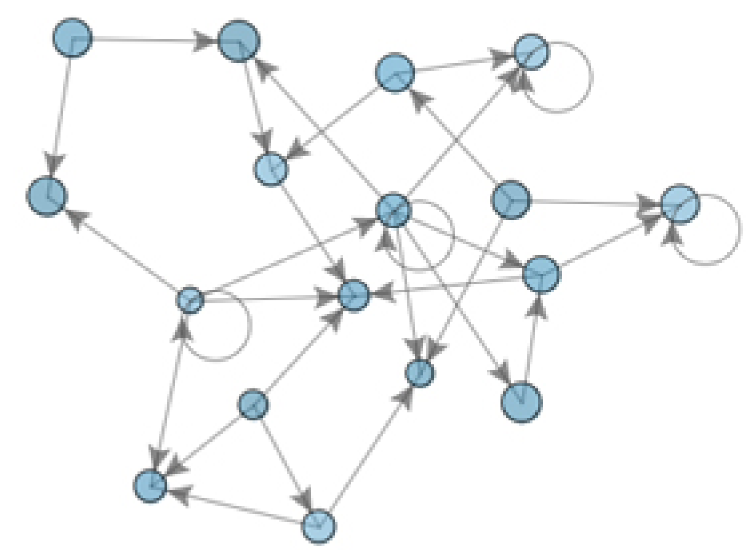
Variations in Standard Network Form

- Directed vs Symmetric: whether edges go *from* one node *to* another node or are *between* two nodes; i.e., whether the order matters.
- Weighted edges and Multigraphs: whether the edges are boolean connections or have values, and whether multiple connections between/to-from the same pair of nodes are possible.
- Reflexive or Irreflexive: whether it is possible for an edge to have the same node at both ends; i.e. whether a node can have connection to itself.

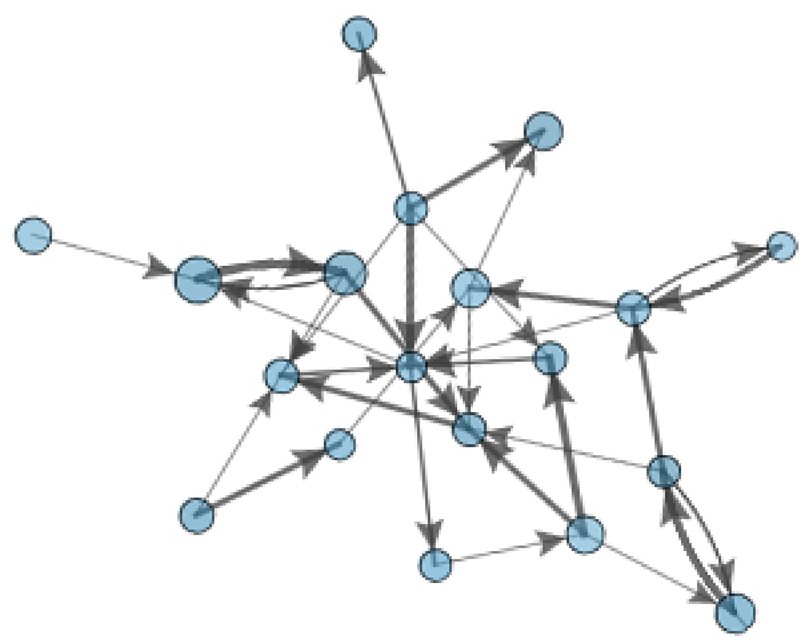
Unweighted Directed Network



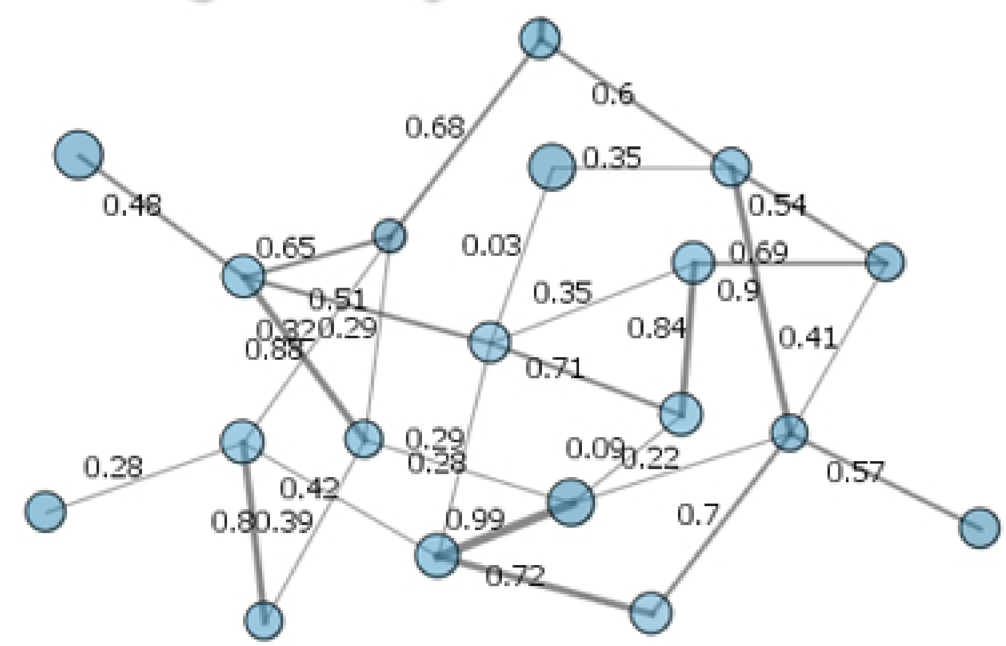
Unweighted Symmetric Network



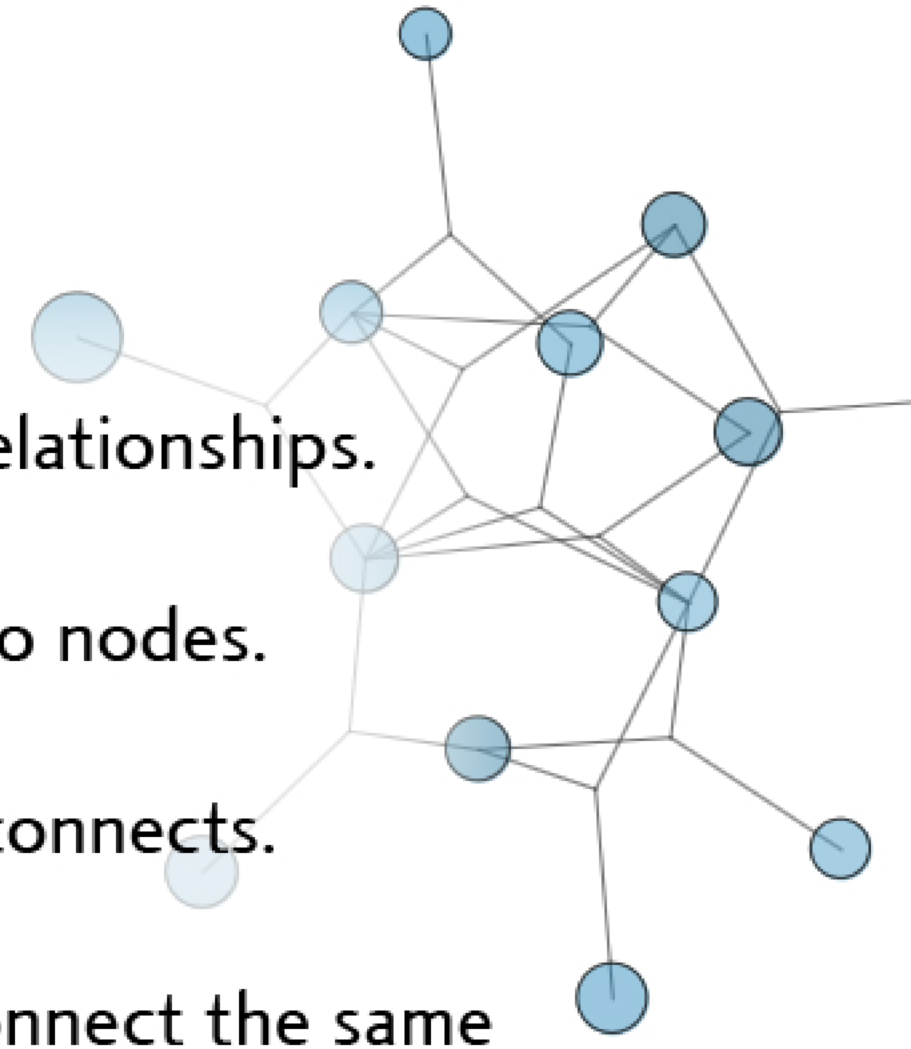
Weighted Directed Network



Weighted Symmetric Network



Introducting Hypergraphs



- A hypergraph is a network with non-dyadic relationships.
- Thus a hyperedge can connect more than two nodes.
- An edge's valence (h_i) is how many nodes it connects.
- If a hypergraph is h-regular, then all edges connect the same number of nodes, though h can be any number > 1.
- Otherwise, specify the maximum number of connections H such that $2 \leqq h_i \leqq H$ for edges.

Directed and Symmetric Hypergraphs

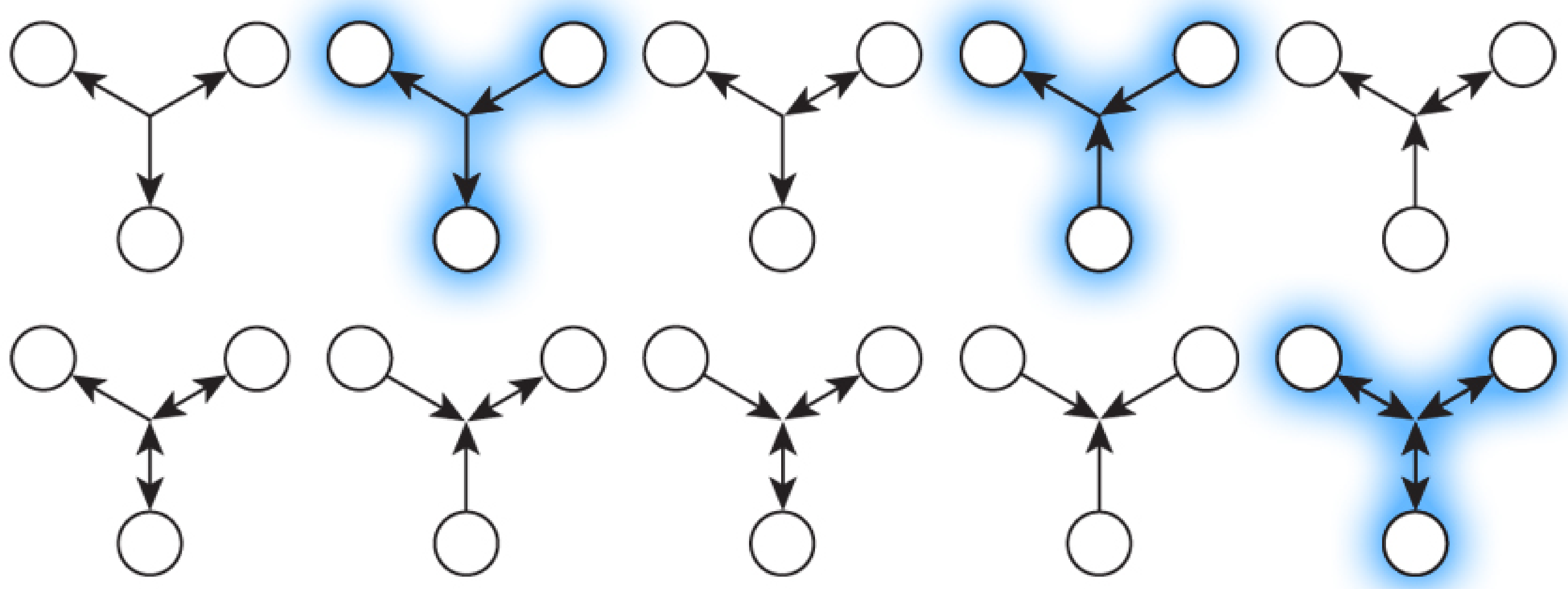
- In a symmetric hypergraph, a hyperedge represents a connection *among* its nodes; the order doesn't matter.

- There multiple ways to represent ordering for members of sets:
  1) a directed primal graph representation
  2) a direction for each arm of the hyperedge;
  3) a ranking (first , second, ...) for each arm; or
  4) a ranking of each node connected by the hypergraph.

- Option 2 holds the most promise for network model applications.

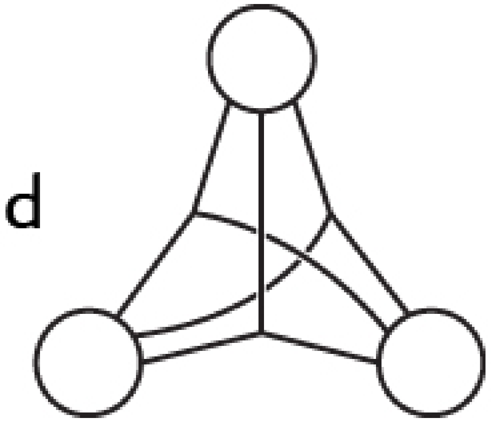
Directed and Symmetric Hypergraphs

- There are $\sum_{i=1}^{h+1} i = \frac{(h+1)(h+2)}{2}$ different shapes of directed hyperedges.
- We may eliminate many of these as meaningless in most cases.



Weighted Hyperedges and MultiHypergraphs

- Nodes in unweighted networks are connected or not connected; it is a Boolean value typically encoded with 0 and 1.
- The connections in weighted networks have values, and so the whole hyperedge possesses the value(s).
- In a multigraph, the same set of nodes can be connected by more than one edge: all the nodes are the same.

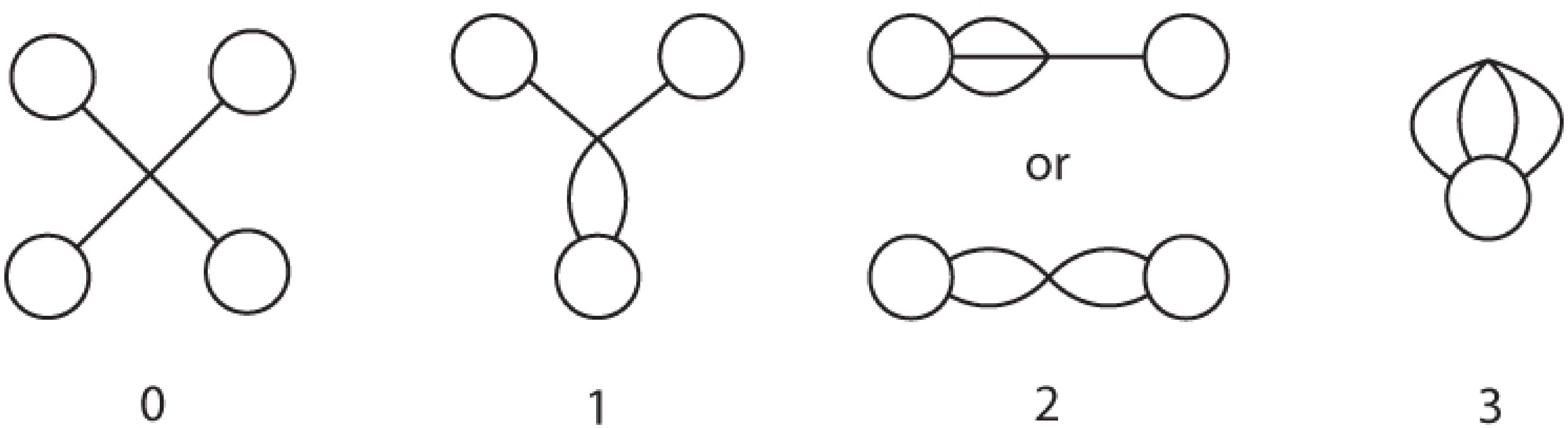


- If the edges are not otherwise weighted, then a multigraph can be represented as a weighted graph; otherwise one needs a weighted multigraph.

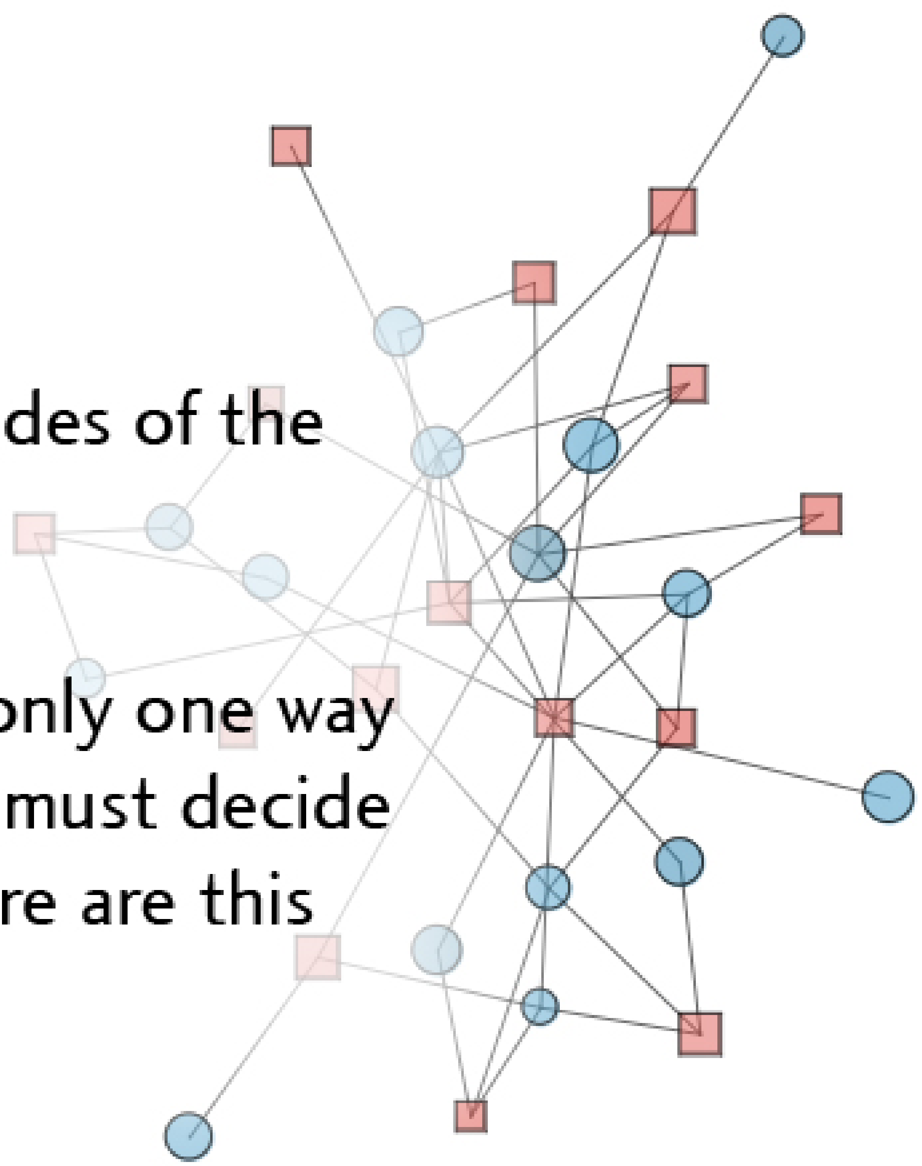
Reflexive Hyperedges

- A reflexive connection is between a node and itself (symmetric), or from a node to itself (directed), and hyperedges increase the number of ways in which that's possible.

- Reflexive hyperedges pose similar representational issues as multihyperedges, especially if they're directed or weighted.



Introducing G-Partite Graphs



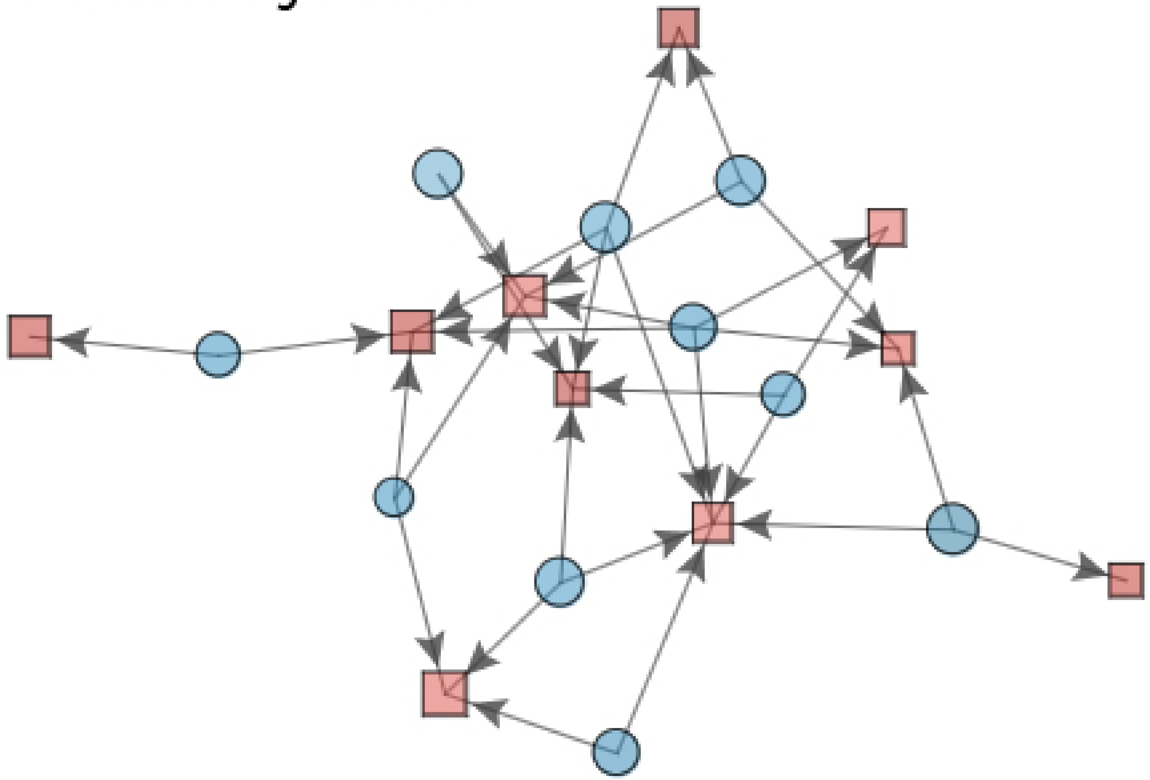
- Nodes are partitioned into *G* types, and nodes of the same type cannot be connected.

- If G = 2, then this is bipartite and there is only one way to separate them; but if G > 2, then one must decide which nodes can be connected...and there are this many ways to do that:

$$\sum_{g=1}^{G-1} 2^g - 1$$

Directed and Symmetric G-Partite Graphs

- G-partite graphs are dyadic, so there are no special considerations for making them directed.
- But not all directions will make sense, they must be appropriate for your data or system.

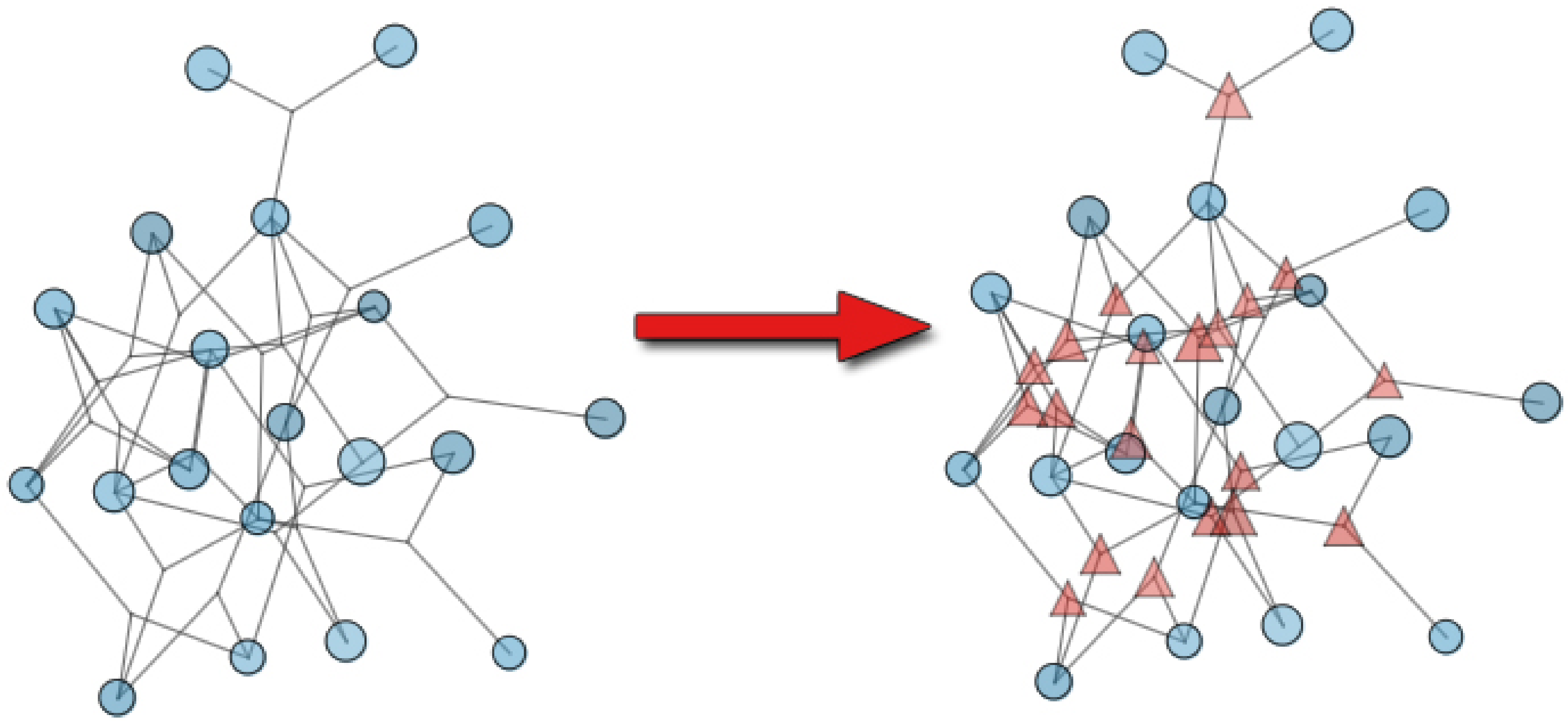


Weighted, Multi, & Reflexive G-Partite Graphs

- Again, because *G*-partite graphs are dyadic, there are no special considerations for making them weighted or multigraphs.
- Because the defining characteristic of *G*-partite graphs is that nodes of the same type cannot be connected, and the types form a partition of the nodes, there cannot be any reflexive connections in *G*-partite graphs.

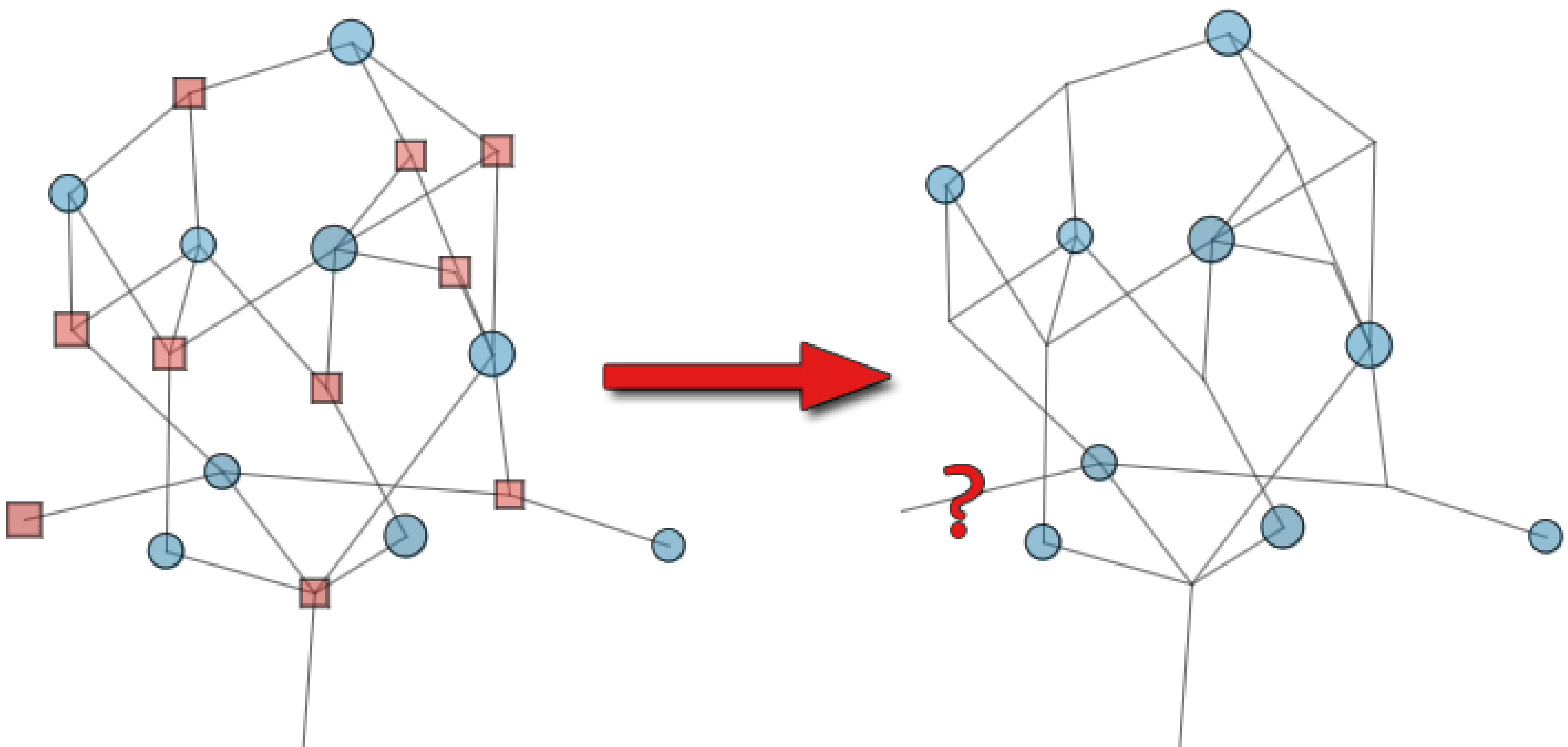
Conversion from Hypergraph to G-partite

- Any hypergraph can be converted into a bipartite graph called its incidence graph: make a new type of node, a nexus, to represent each hyperedge, and the nodes are attached to the nexus if they are connected by that hyperedge.



Conversion from G-partite to Hypergraph

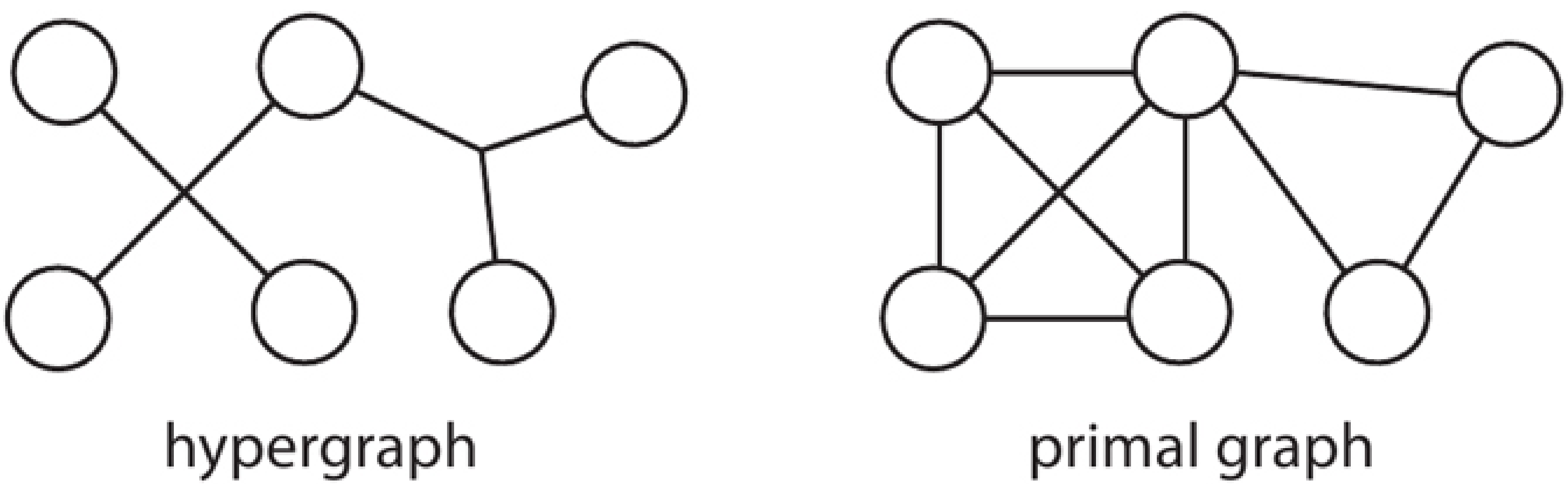
- A *G*-partite graph can be converted into a hypergraph by considering one type to be nexus nodes.
- Not all *G*-partite graphs can be converted this way, they must be appropriately connected.



Projections of G-partite and Hypergraphs

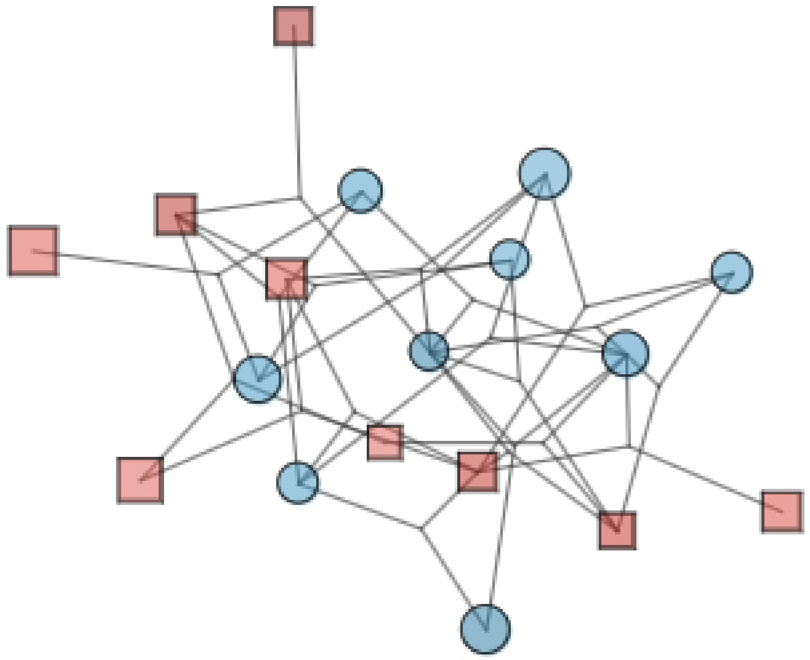
- Because these more advanced representations are not as well known or fully developed, many people use a primal graph projection instead.

- To make the primal graph projection, just make a clique out of the nodes connected by a hyperedge or specific node type.



G-partite Hypergraphs

- A network can be both *G*-partite and a hypergraph.
- But to be truly both, G must be at least H, because there must be enough different types of nodes so that the same types are connected by the same hyperedge.

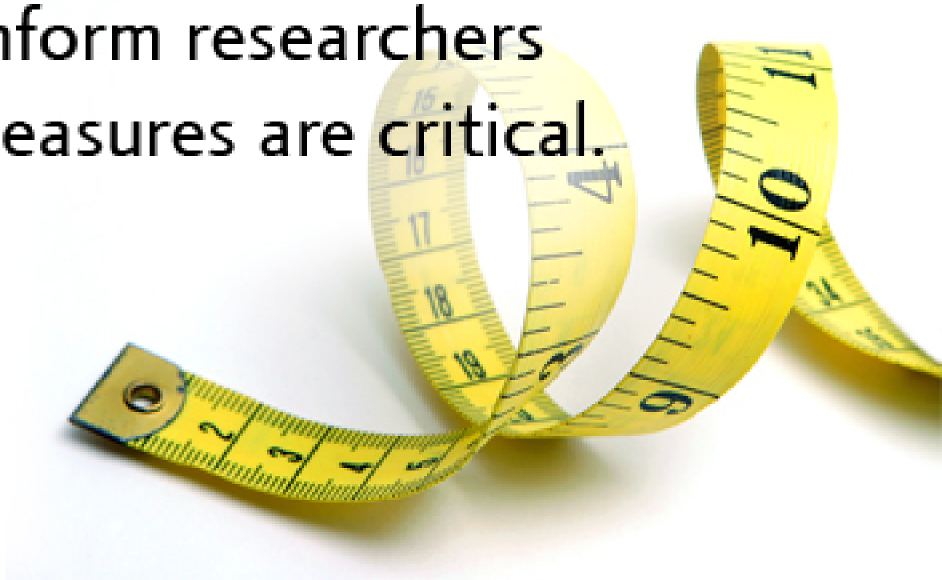


Network Measures

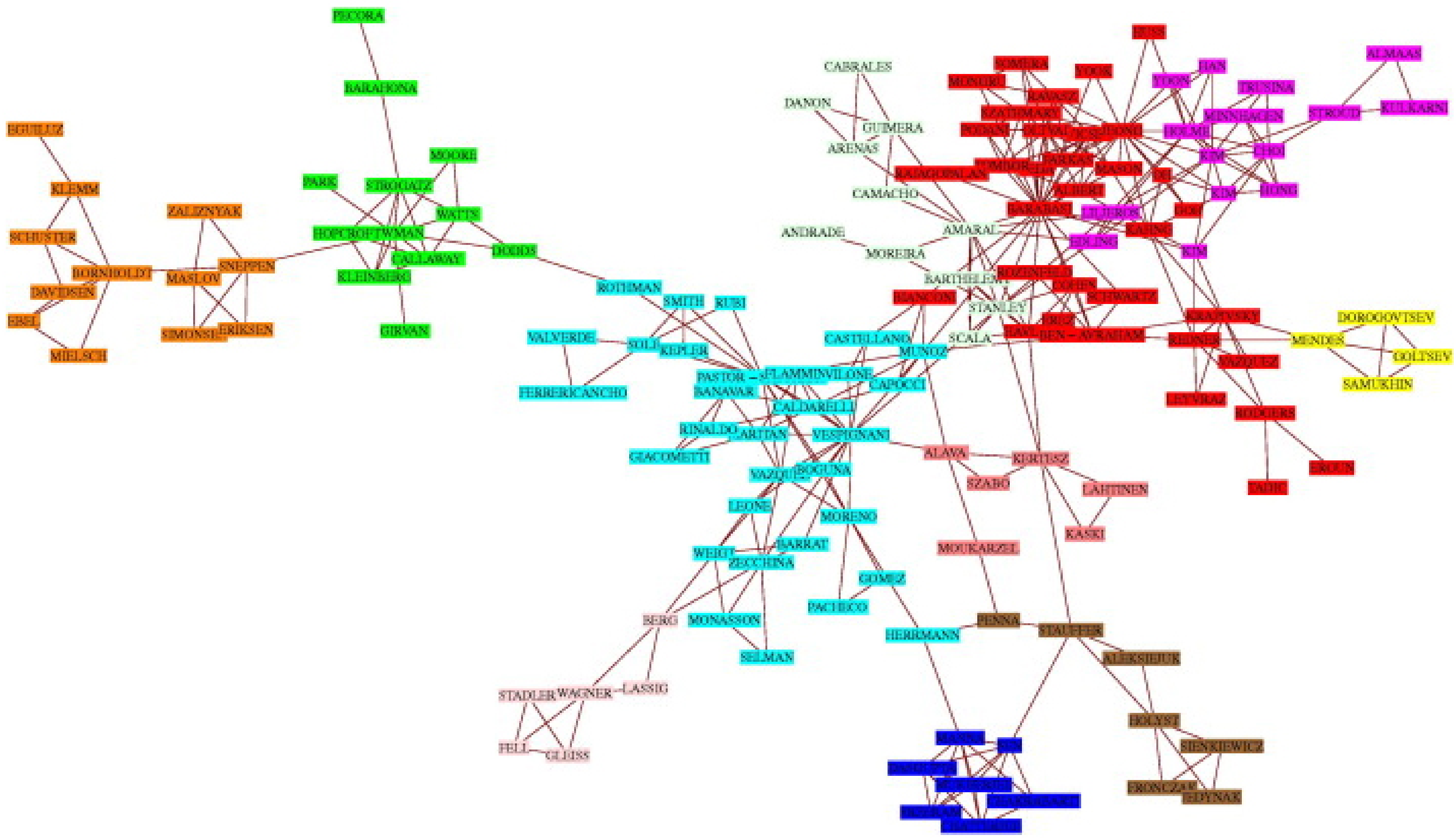
- The reason we care about the more advanced graph structures is because they are better representations of many situations AND because many measures are undefined or yield different values for dyadic, hyper, and G-partite graphs.

- Specifically, people are using primal graph projections because they don't know any better, and because a better alternative is not yet available.

- So we need to invent these measures, and inform researchers of when these more advanced network measures are critical.



Network Measures



Network Measures

- For example, when using the primal graph, distance is preserved, but clustering coefficients, degree, and betweenness are not.
- In fact, these all MEAN something different for hypergraphs; and despite the bipartite equivalence we may need distinct measures.
- Similarly for community structure, components, paths, cycles, and other structural measures.
- Work on generative functions, percolation, and all the other mainstays of network theory has yet to be done too.


That’s all Folks!