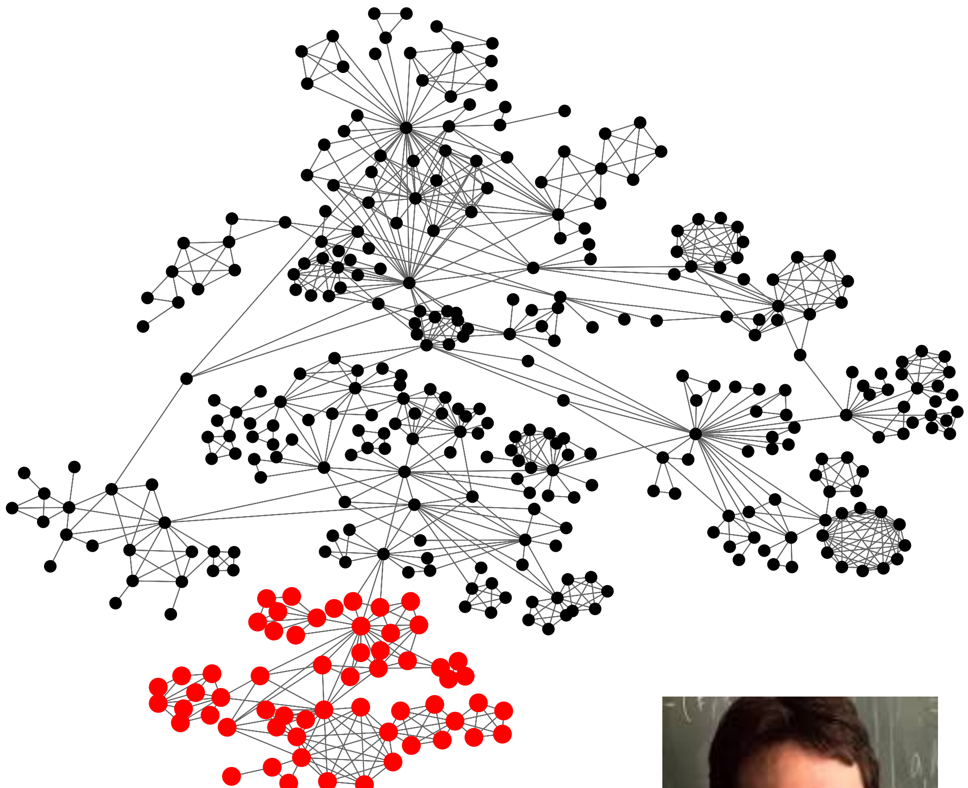
Graph Clustering

September 27, 2022

Advanced Graph Algorithms

Nate Veldt

Graph clustering and partitioning involve separating a graph into pieces with few edges between pieces



Newman's Netscience Graph

Finding clusters or partitions in graphs has many diverse applications.

Computer vision: *image segmentation*
Parallel computing: *load balance*
Social network analysis: *community detection*
Machine Learning: *object classification*
Bioinformatics: *Identifying related genes*

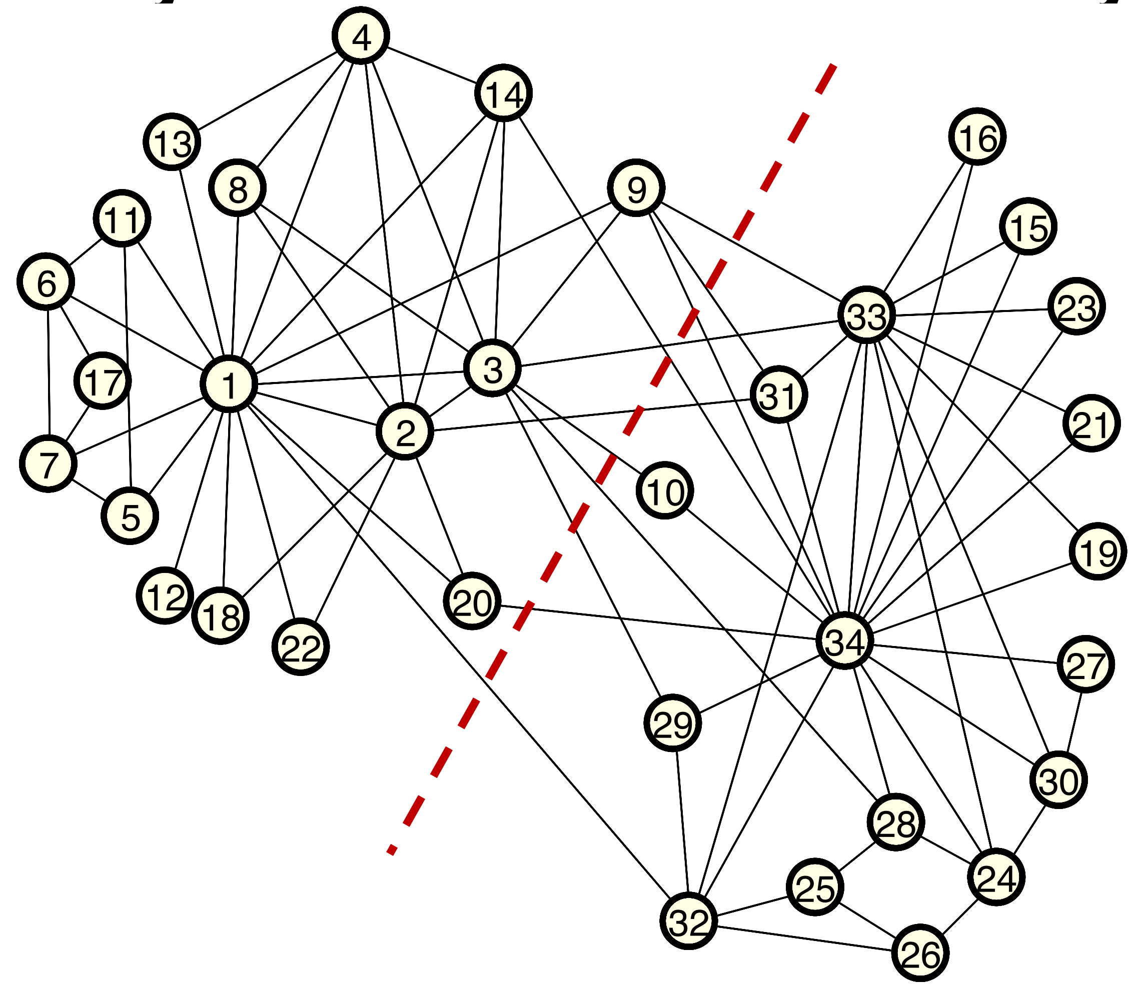
Let's look more closely at a few examples!

Zachary's Karate Club is widely studied in the community detection literature

Nodes = people in a karate club

Edge = social interaction outside club activities

The club split due to disagreement between the president (34) and instructor (1)

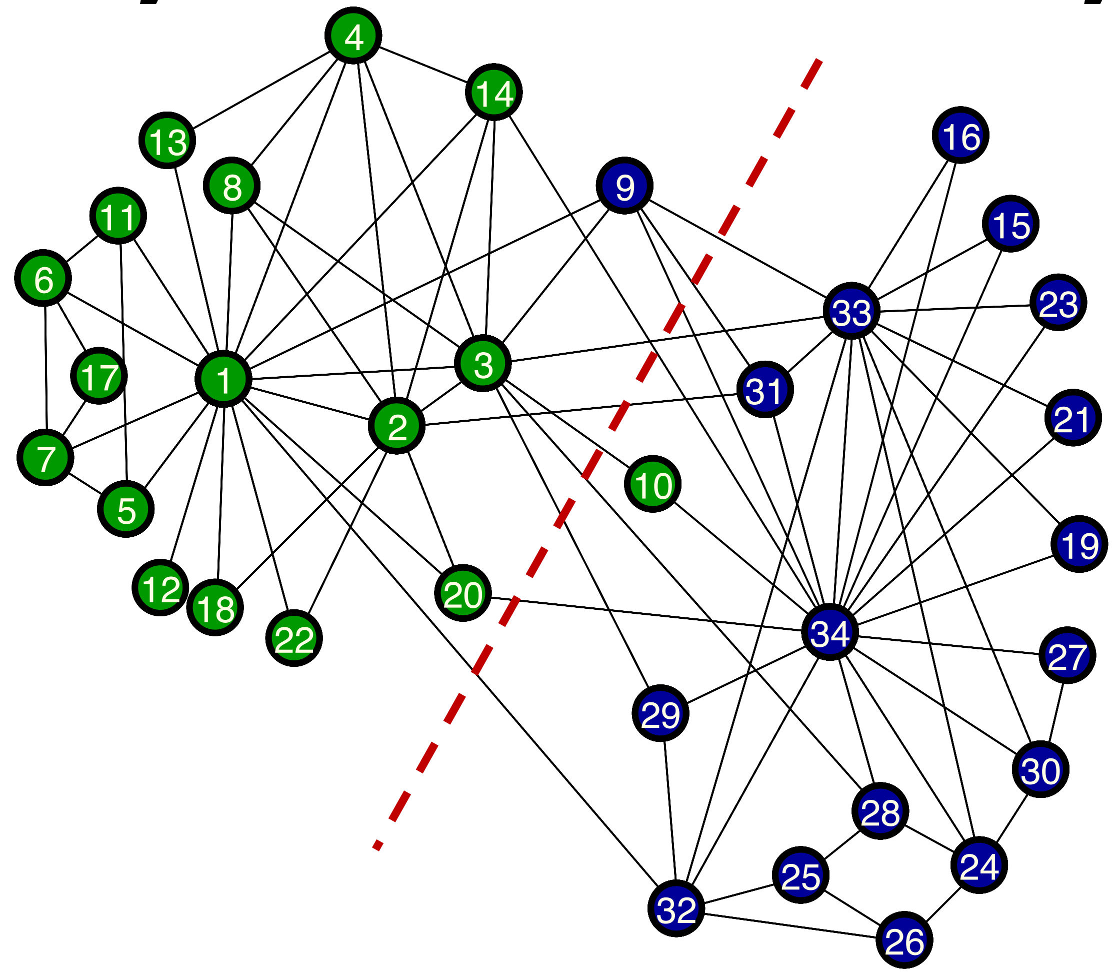


Zachary's Karate Club is widely studied in the community detection literature

Nodes = people in a karate club

Edge = social interaction outside club activities

The club split due to disagreement between the president (34) and instructor (1)



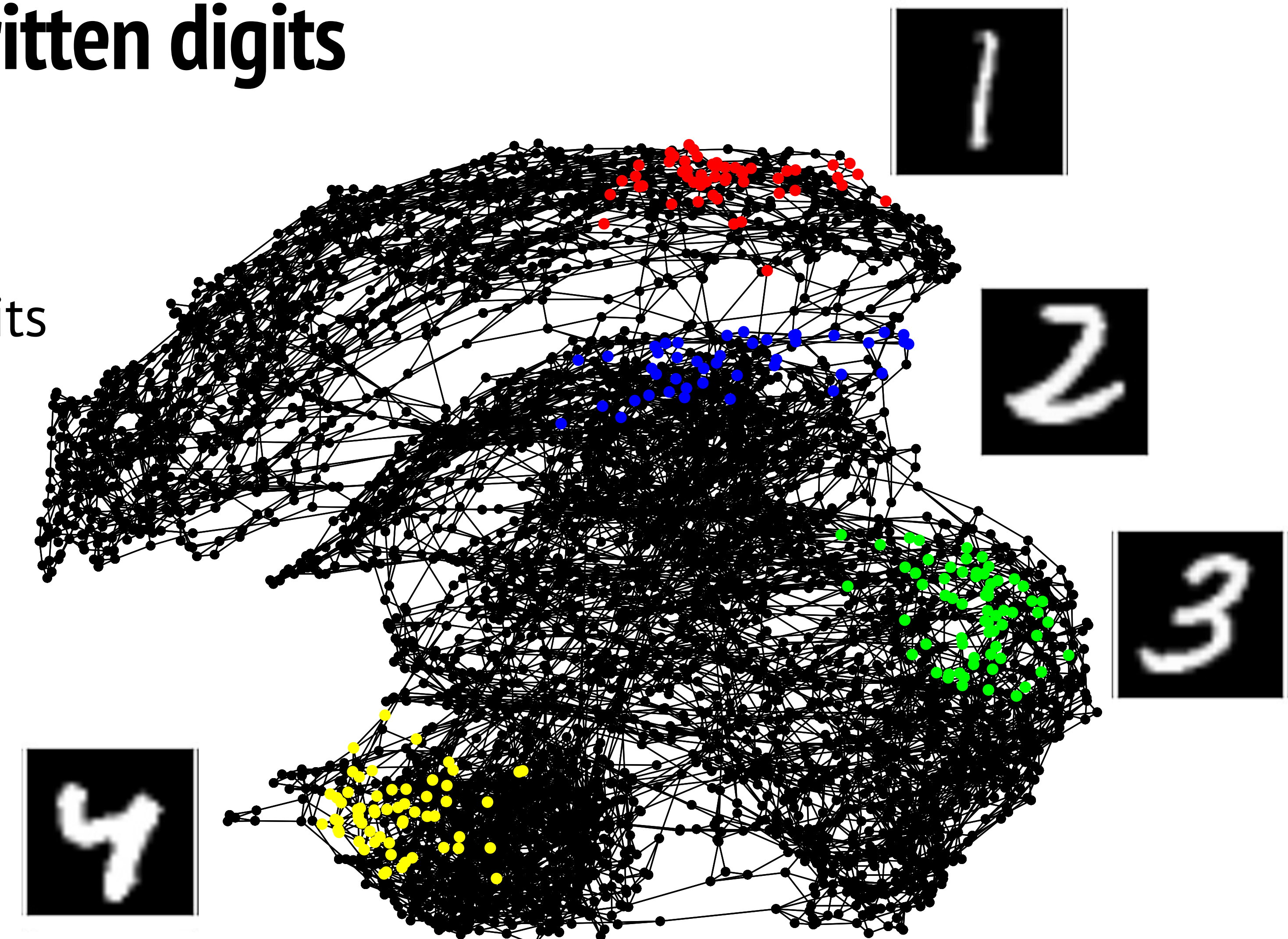
The minimum s-t cut problem can almost exactly predict how the members separated into two new clubs.

Classifying handwritten digits

Nodes = handwritten digits

Edge = high similarity scores between images

Cluster = nodes that correspond to the same digit

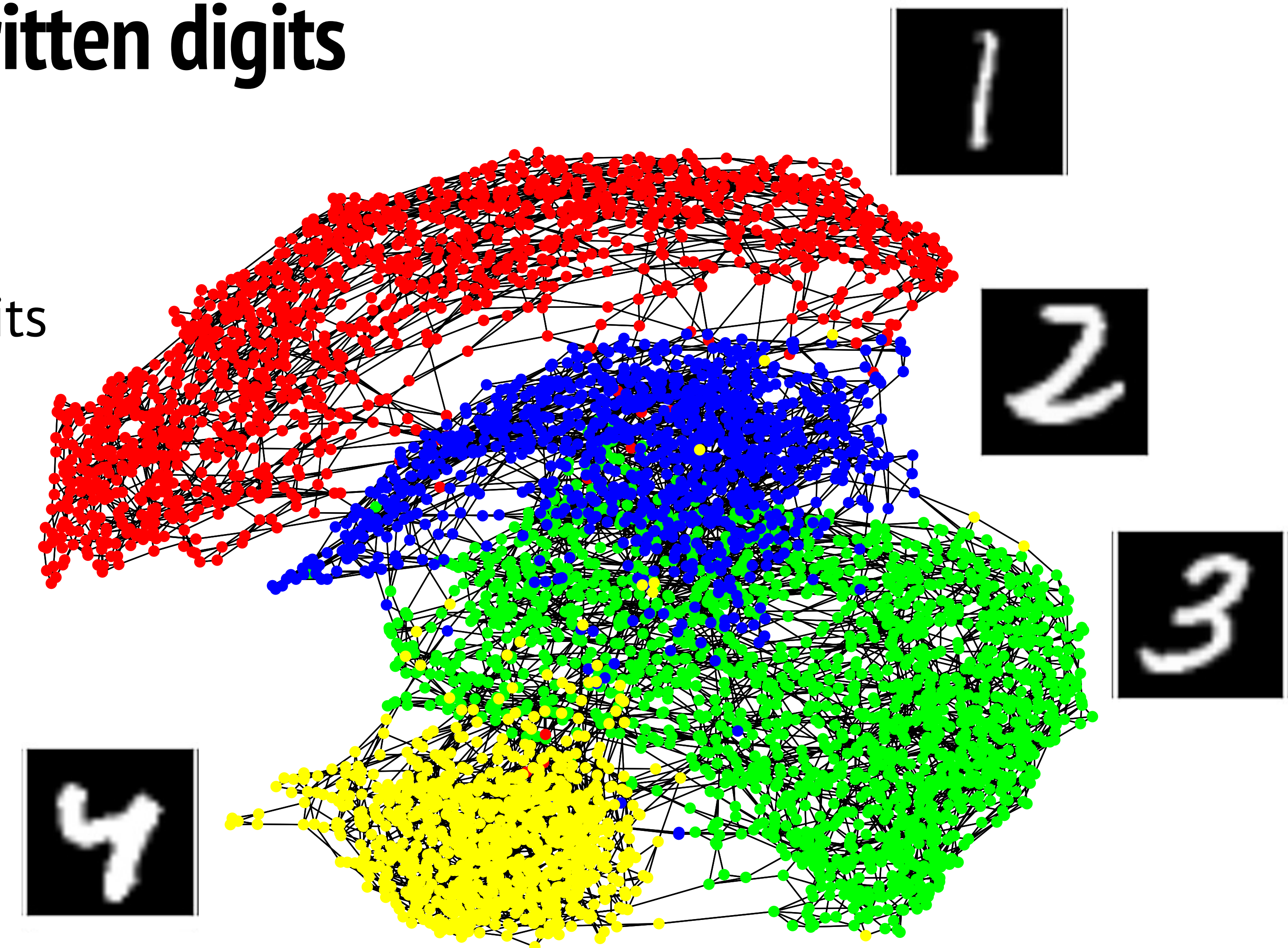


Classifying handwritten digits

Nodes = handwritten digits

Edge = high similarity scores between images

Cluster = nodes that correspond to the same digit



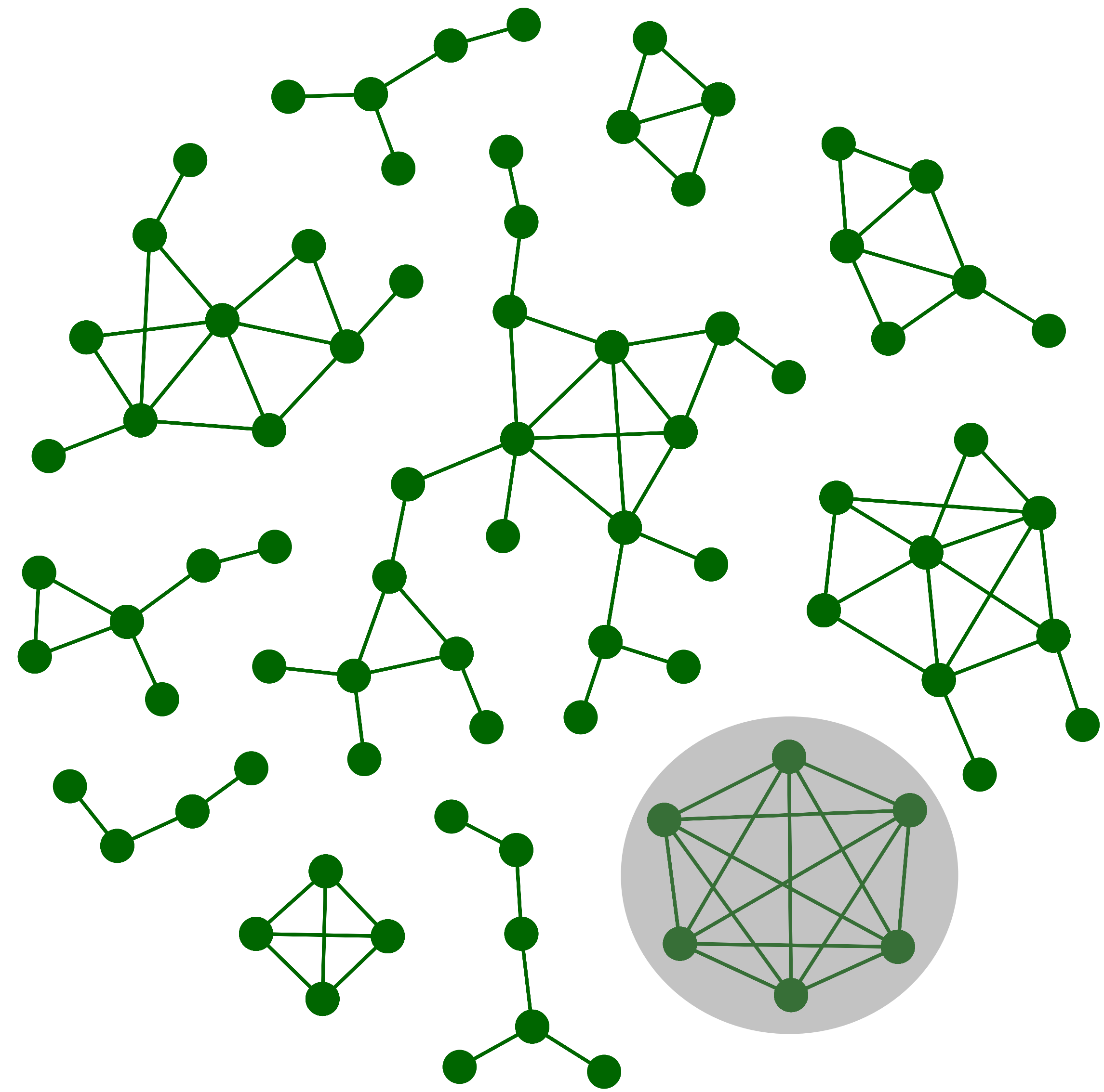
You can (quickly) get a clustering with under a 1% error rate using standard tools.

Gene Clustering

Nodes = yeast genes

Edge = high correlation in microarray expression data

Cluster = nodes associated with same function/attributes



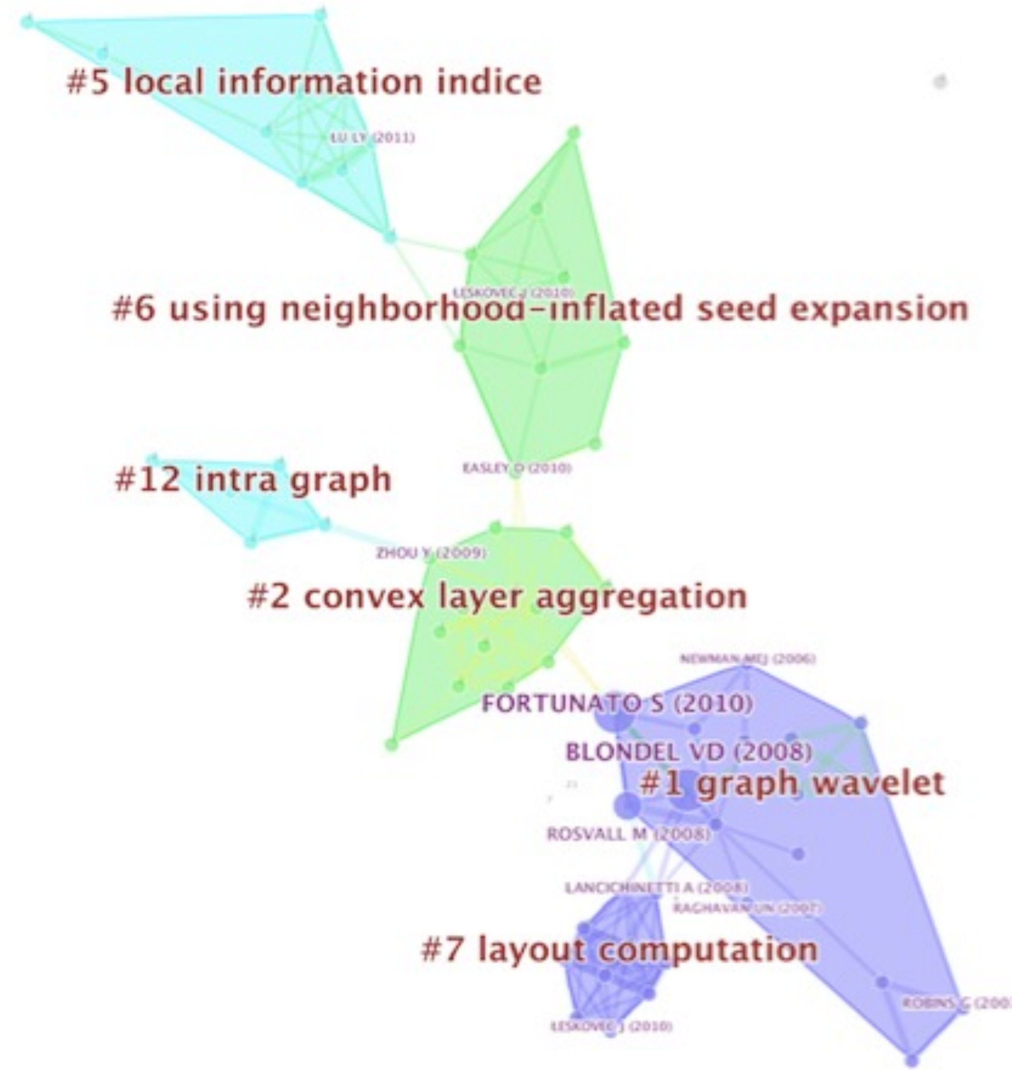
Six nodes associated with functions in the nucleus.
5/6 share in **“transposition”** biological process (which only 1.7% of 6433 genes share in)

Co-citation network

Nodes = papers on graph clustering

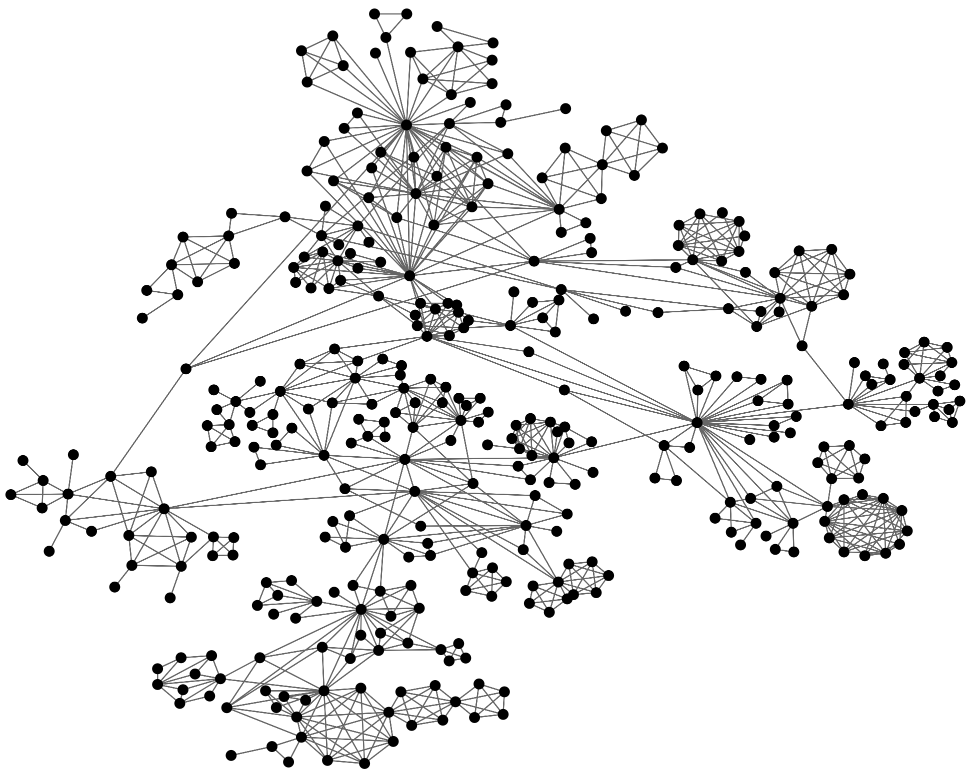
Edge = frequently co-cited together

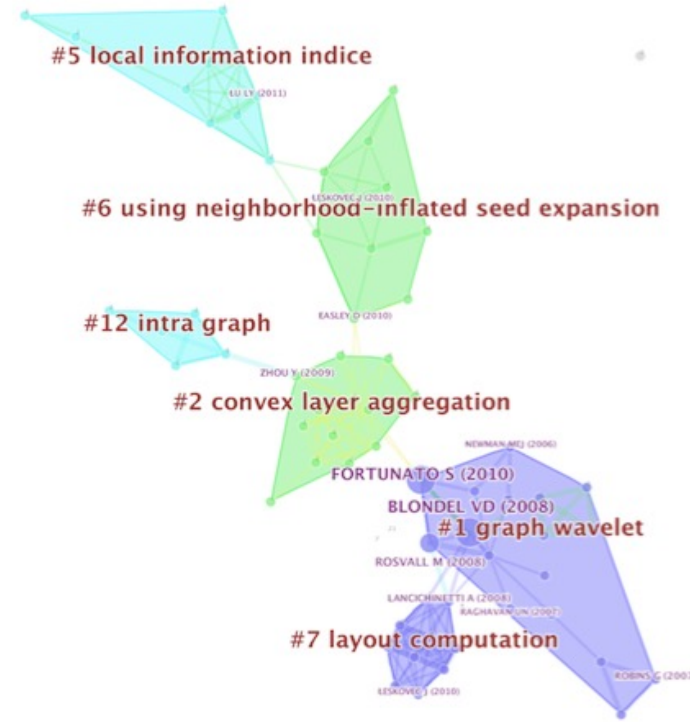
Cluster = "hot topics" in graph clustering literature

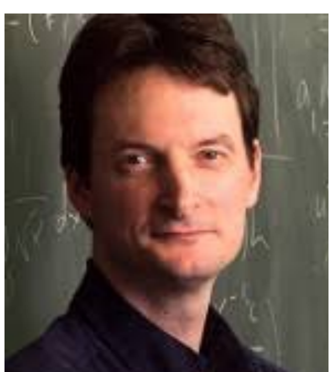


Thanks to Kimon Fountoulakis!

So whose contribution is the most meta?







Mark Newman

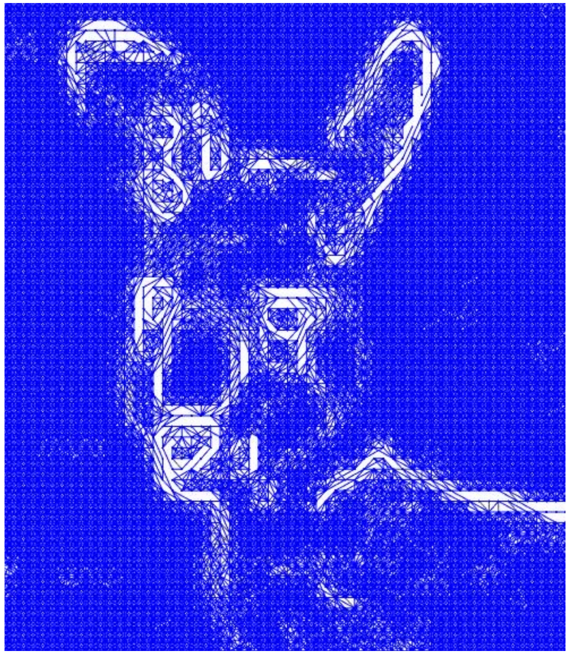
Defining a network of network scientists

Kimon Fountolakis

Graph clustering to find "hot topics" in graph clustering

Image Segmentation

Nodes = pixels
Edge = high similarity scores based on pixel brightness



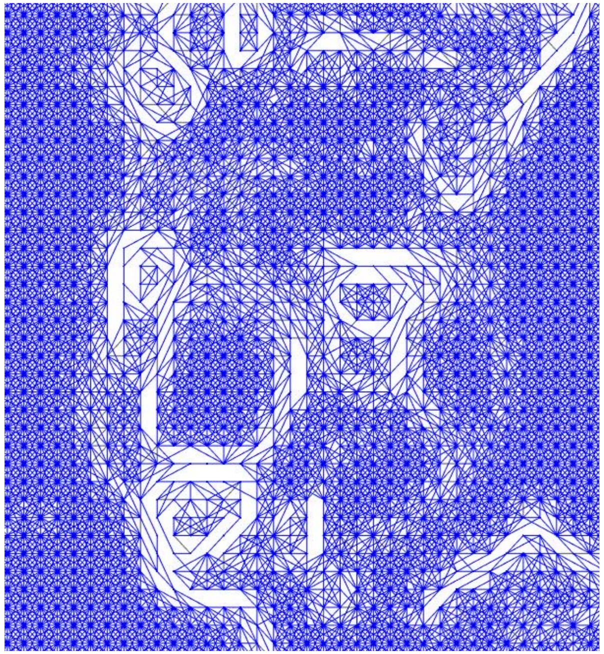
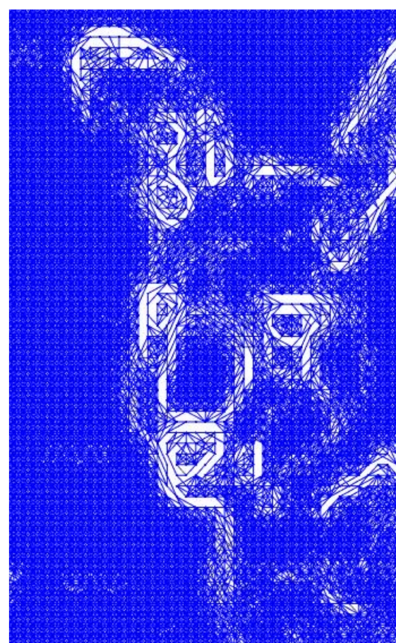


Image Segmentation

Nodes = pixels
Edge = high similarity scores based on pixel brightness
Cluster = object in picture



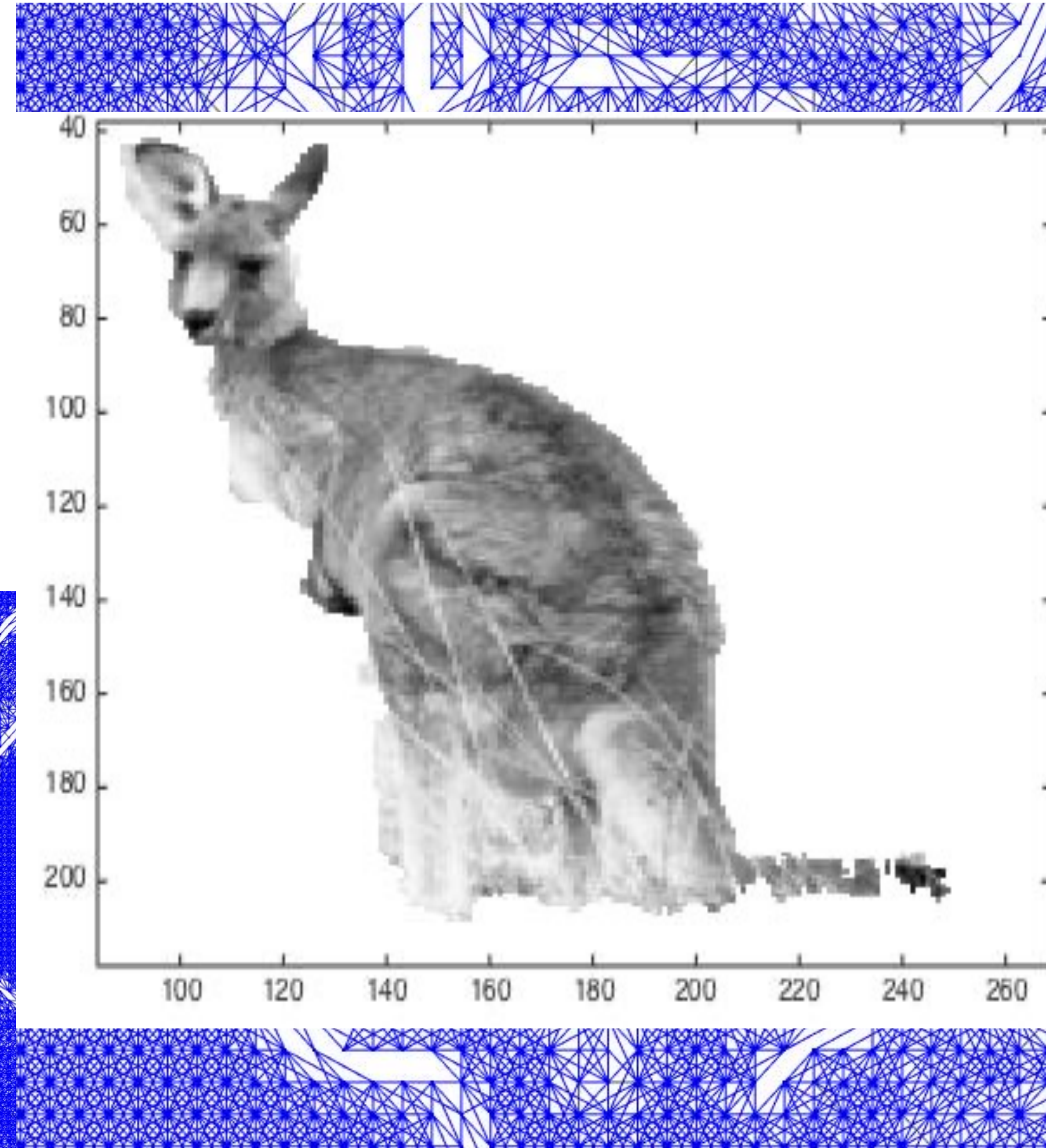
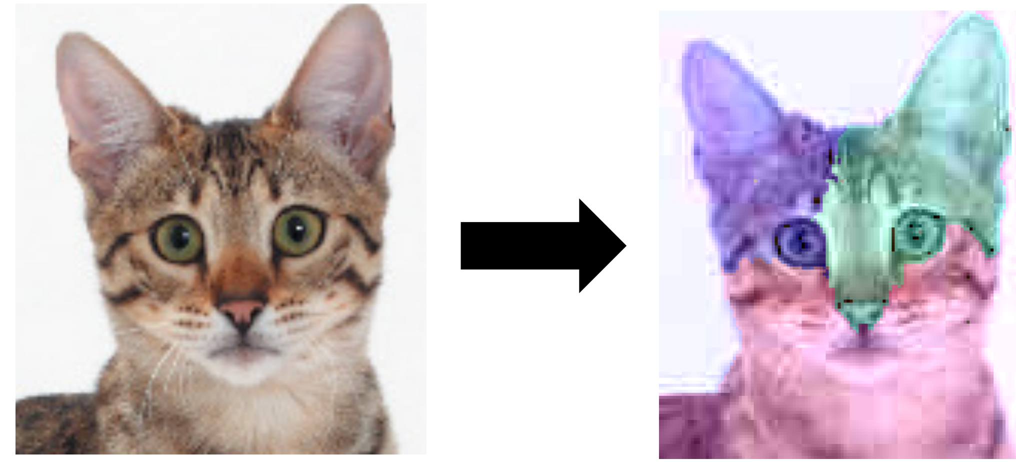
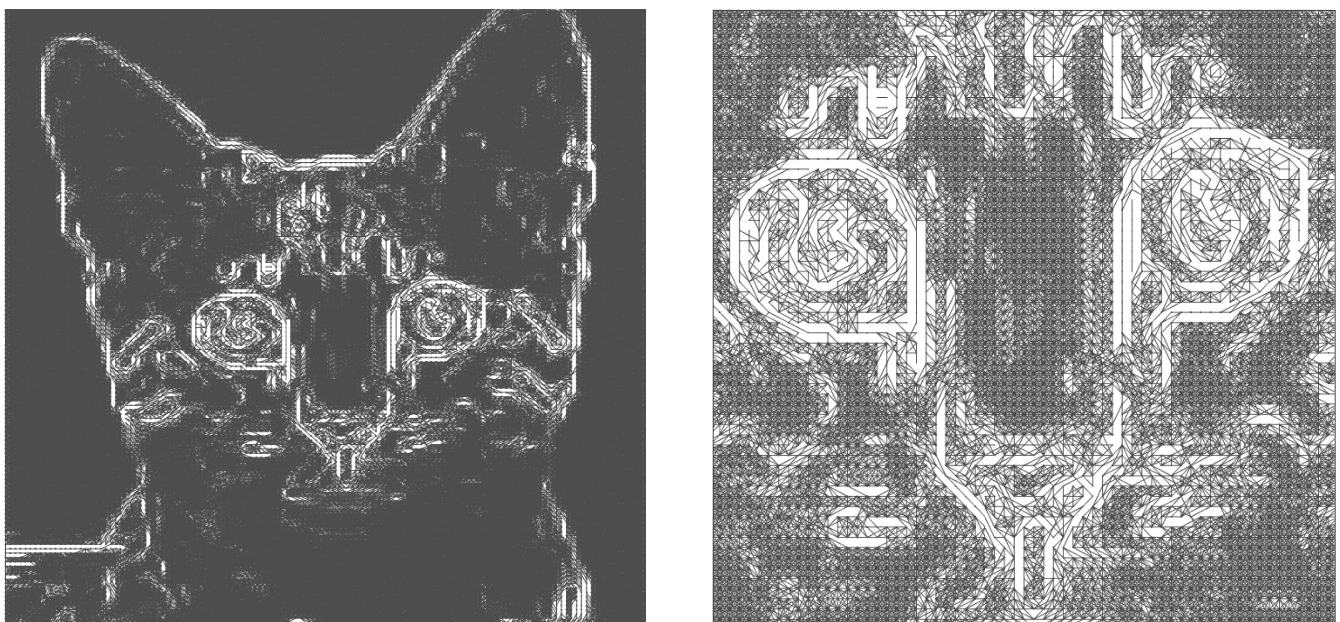


Image segmentation

Nodes = pixels
Edge = high similarity scores based on pixel brightness
Cluster = object in picture





There are many (almost) synonymous terms for this

Often called
"community detection"

Graph Clustering: clusters are sets of nodes with
(a) many internal edges and (b) few edges between clusters.

Typically, there is no restriction on the number of clusters, and they don't have to be balanced in size.

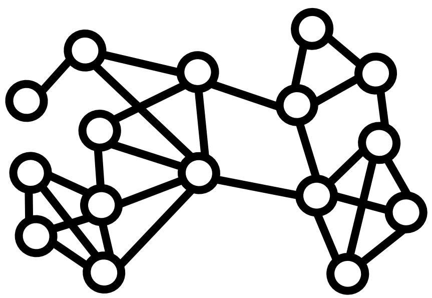
E.g., you use clustering to "detect communities" of people in a social network.

Graph partitioning: separate the nodes into k sets so that the sets are balanced in size and you minimize the number of edges between nodes.

Typically, you fix the number of clusters and balance in cluster size is key. Used often in scientific computing and parallel programming applications.

E.g., you have k processors to solve a task, and you want to partition the workload to minimize communication (cut), and balance the workload (equal sizes clusters).

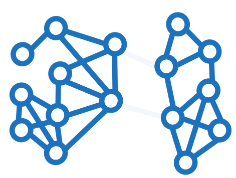
Mathematically, graph clustering can be framed as an optimization problem.



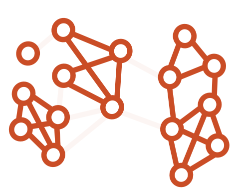
$f(C) = ?$

f assigns scores based on:

1. *Internal density*
2. *External sparsity*

$f($  $) = 0.251$

$f($  $) = 0.89$

$f($  $) = 0.497$

Goal

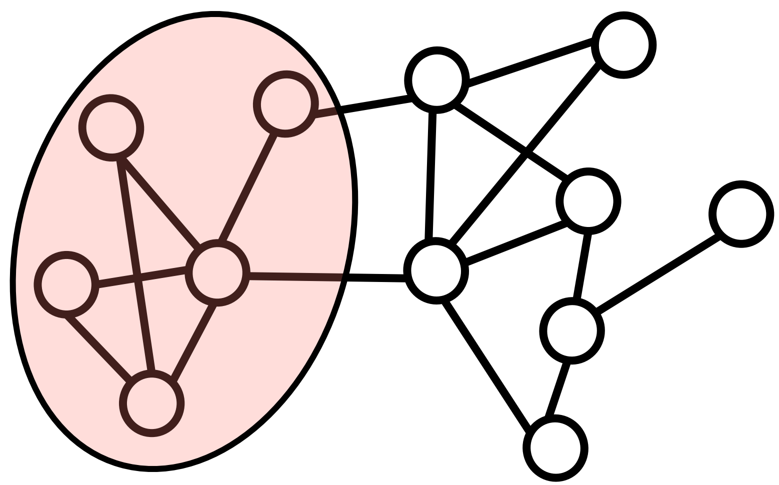
Find the clustering with the "best" (i.e. smallest or largest) score

There are many existing objective functions

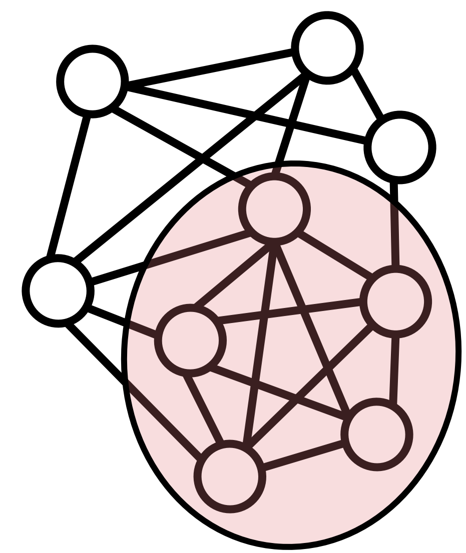
Objective functions

Maximum modularity, sparsest cut, maximum clique, minimum conductance, cluster deletion, etc.

All strike a different balance between



External sparsity

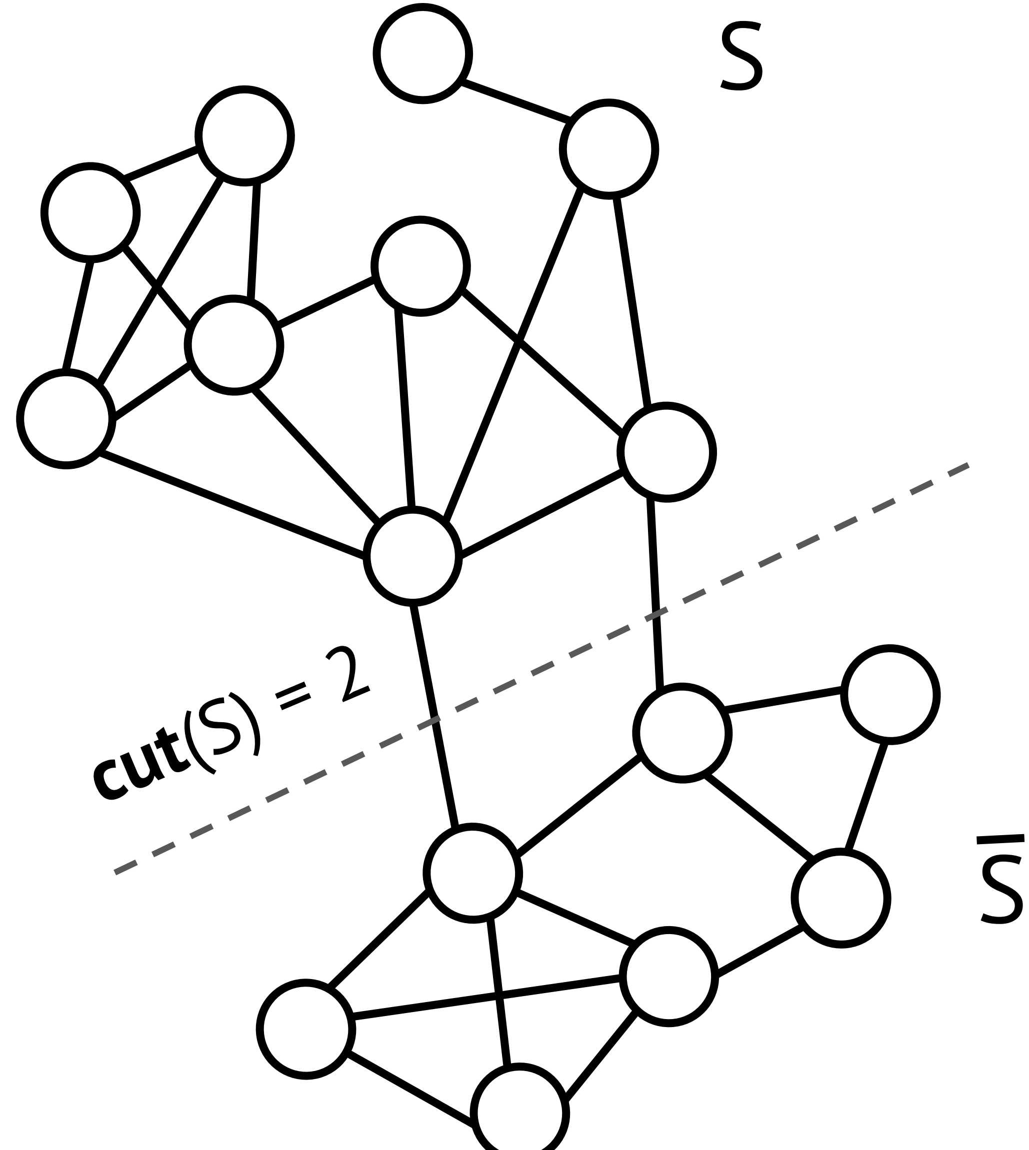


Internal density

One objective we will care about

Sparsest cut

$$\min \quad \frac{\mathbf{cut}(S)}{|S|} + \frac{\mathbf{cut}(S)}{|\bar{S}|}$$



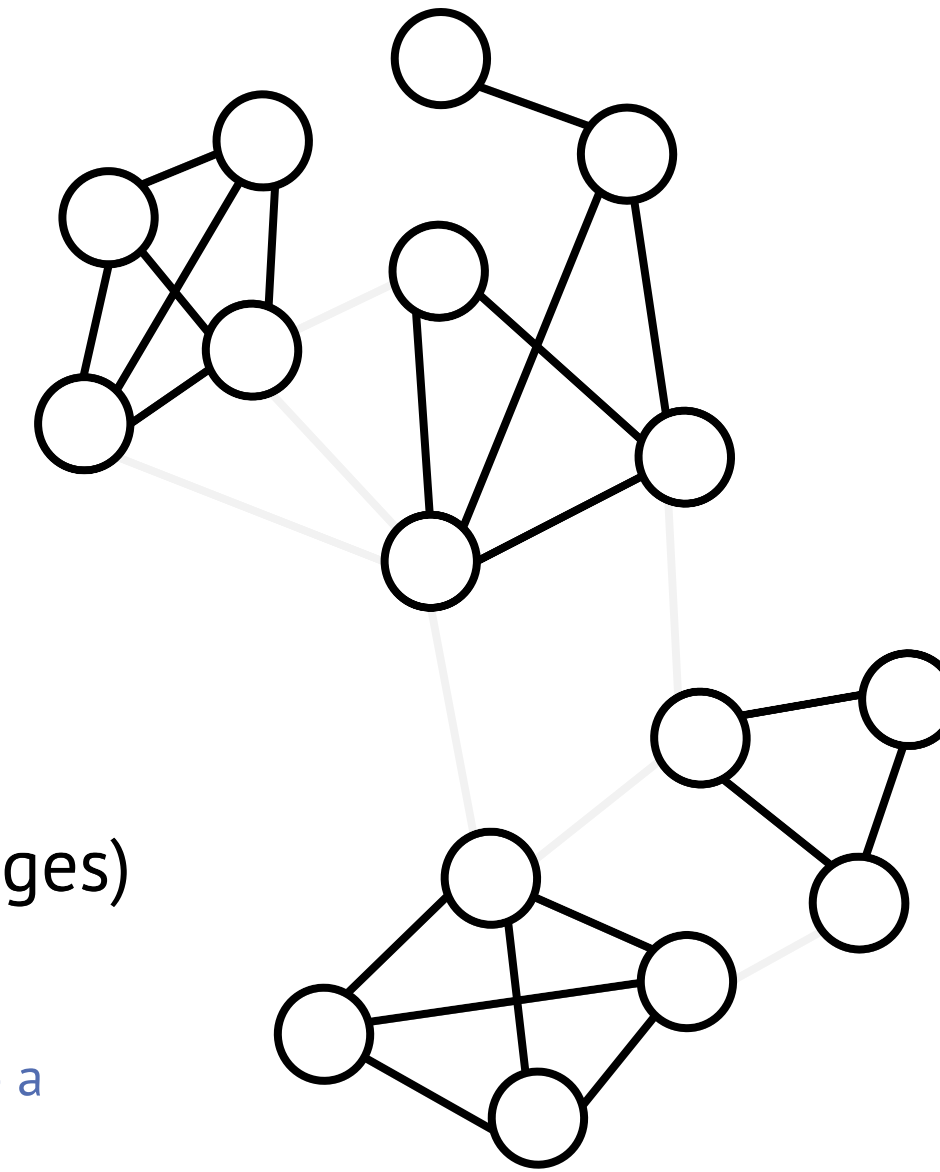
Another common objective

$$\textbf{Mod}(C) = \frac{1}{2|E|}\sum_{i,j}(A_{ij} - P_{ij})\delta(i,j)$$

Informally

Mod(C) = (# interior edges)
− (# *expected* interior edges)

Expectation is measured with respect to a specified mathematical null model



Can we use the techniques we've already learned about?

Minimum s-t cut problem

- It is only in special cases (karate club example) that you get meaningful partitions with s-t cuts
- Typically an s-t cut just returns either the source node by itself, or a very small number of nodes on one side (highly imbalanced)

Dense subgraph discovery

- This misses out on the objective to find sets of nodes that have *few* edges to the rest of the graph.

We'll focus for a while just on two-way cut problems

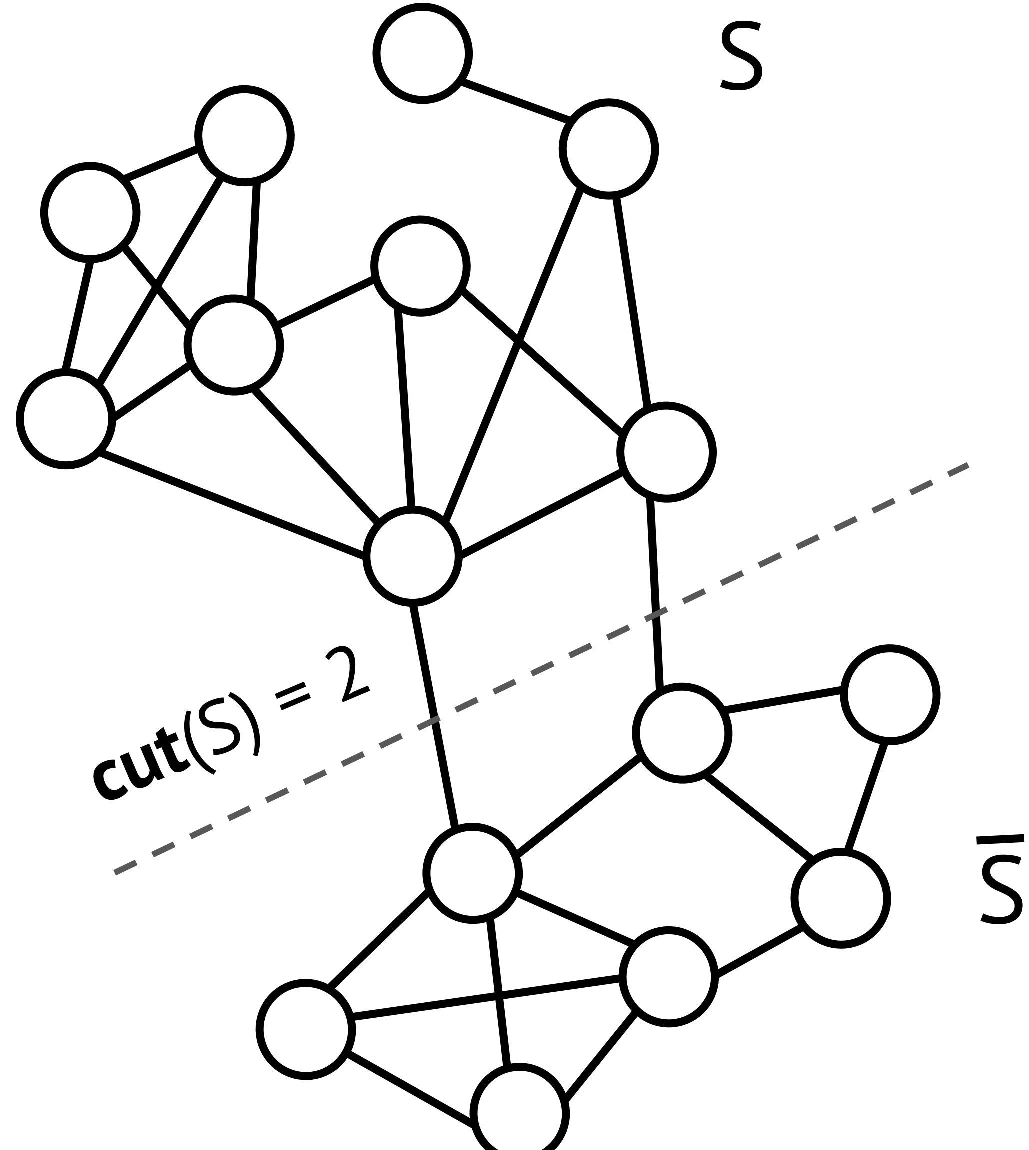
Sparsest cut

$$\min \quad \frac{\textbf{cut}(S)}{|S|} + \frac{\textbf{cut}(S)}{|\bar{S}|}$$

Expansion

Normalized Cut

Conductance



Flow-based methods for graph partitioning/clustering

Flow-based

- In some cases we can still use the minimum s-t cut problem in special ways to get non-trivial clusterings.

We'll see this in our lectures on flow-based local clustering

- More general "multicommodity" flow problems can be used to solve graph partitioning/clustering objectives with more balance

We will not get to these in this course. I can point to you references if you are interested.

Other techniques

Multilevel algorithms with local-improvements

- These begin by merging nodes together to form a smaller graph
- Then the small graph is partitioned
- Then you "undo" the node merging steps, and do "local moves" between two clusters to improve the balance objective
- Standard tool is METIS: http://glaros.dtc.umn.edu/gkhome/metis/metis/overview
- Typically more for graph partitioning than graph clustering.
- We won't cover these in the course.

Spectral Methods

Algorithms for partitioning or clustering a graph based on eigenvectors of a matrix associated with the graph → The focus on the next several lectures.