Graph Clustering Algorithms

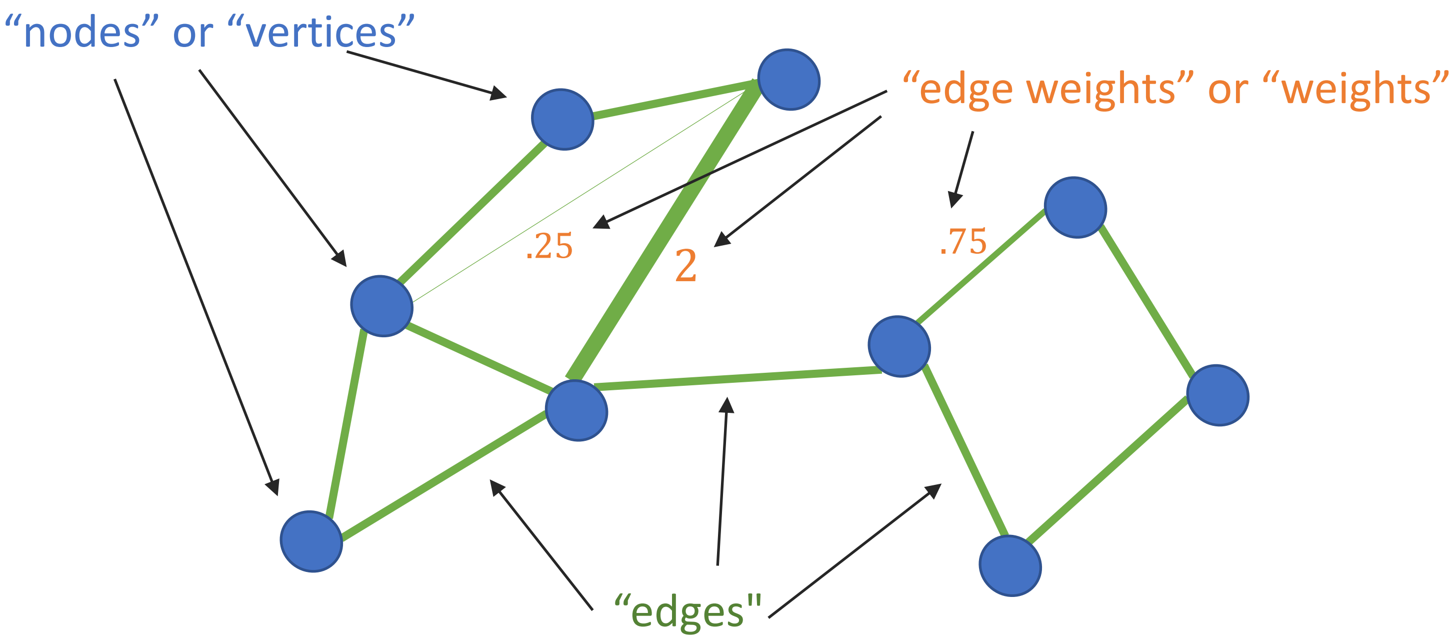
Tselil Schramm

(Simons Institute)

9/28/2017 @ GraphXD

What is a graph?

a.k.a. a network:

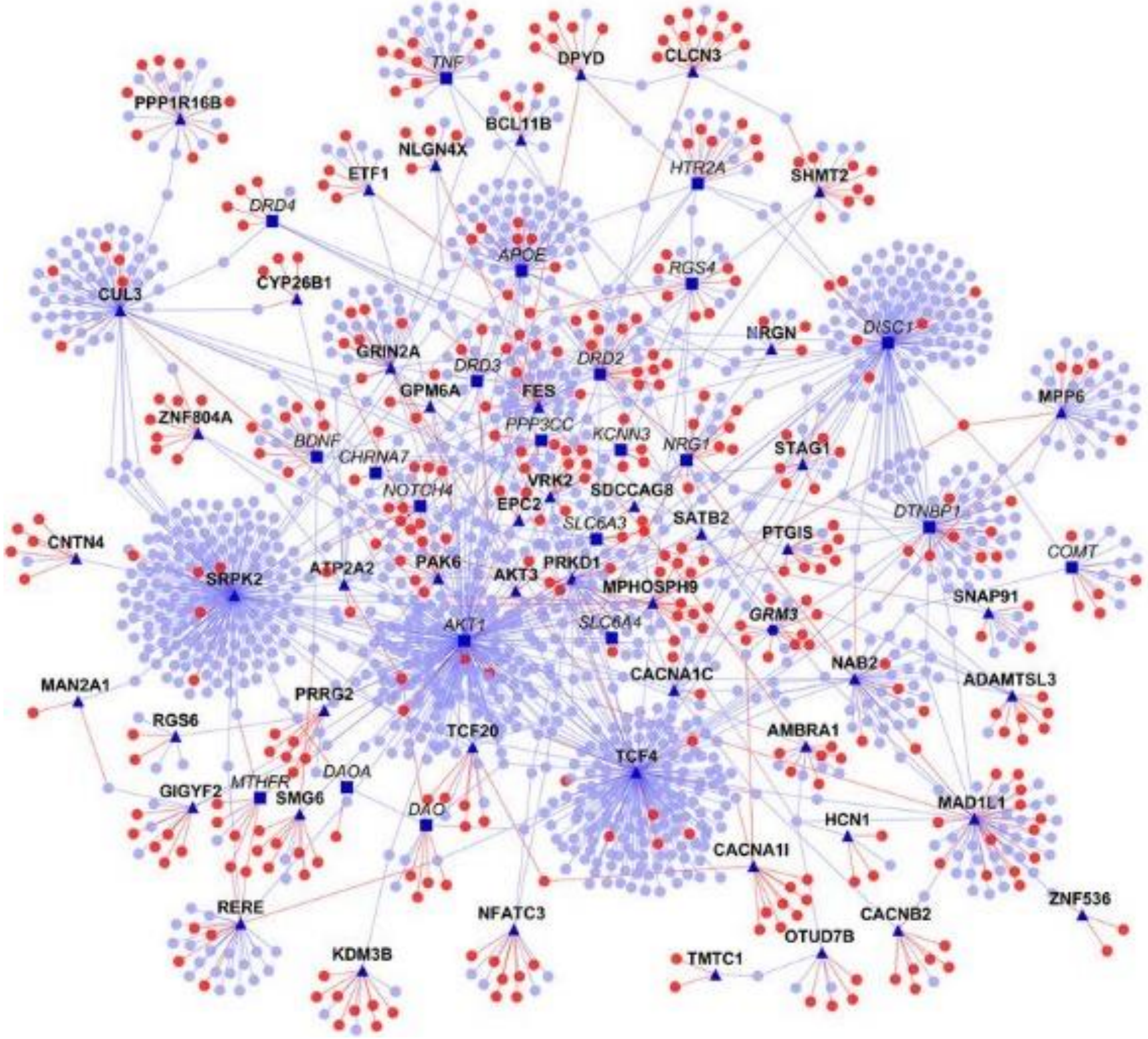


Graphs all over

Social Networks

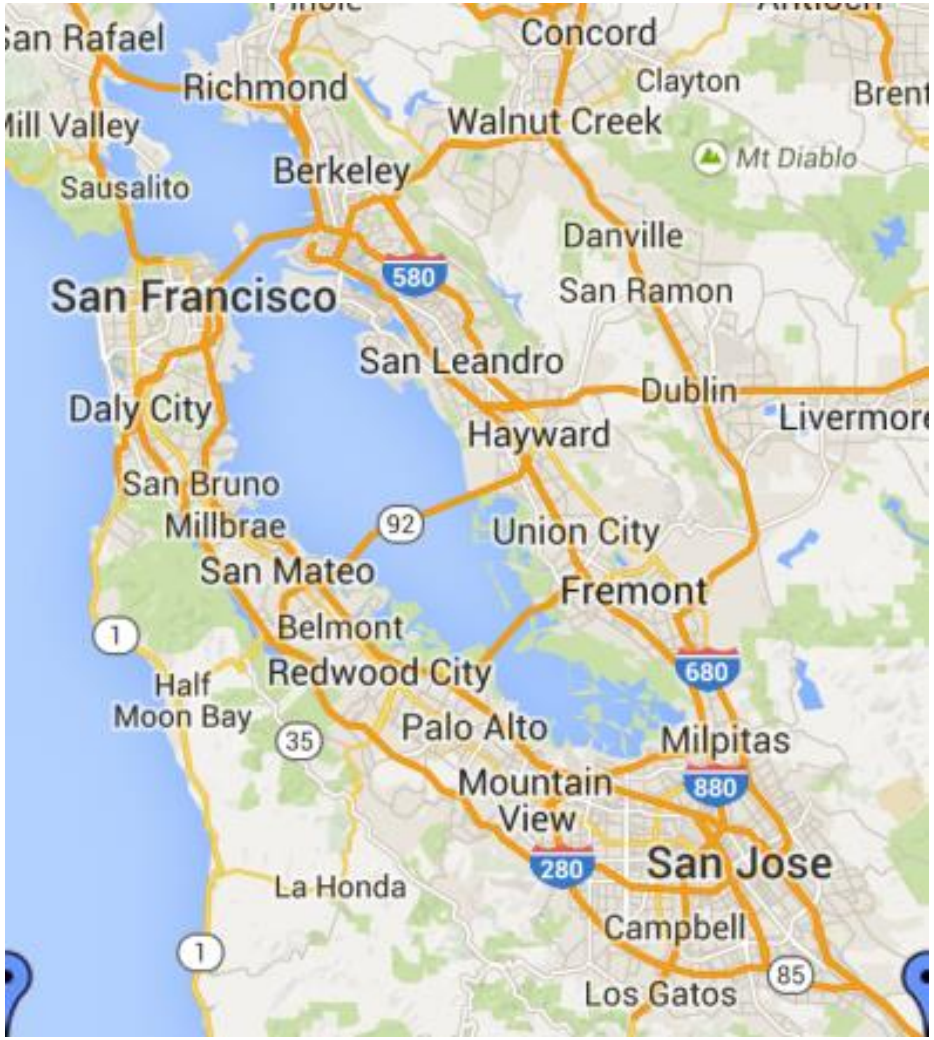
Graphs all over

Schizophrenia protein-protein interaction network (Ganapathiraju et al.)



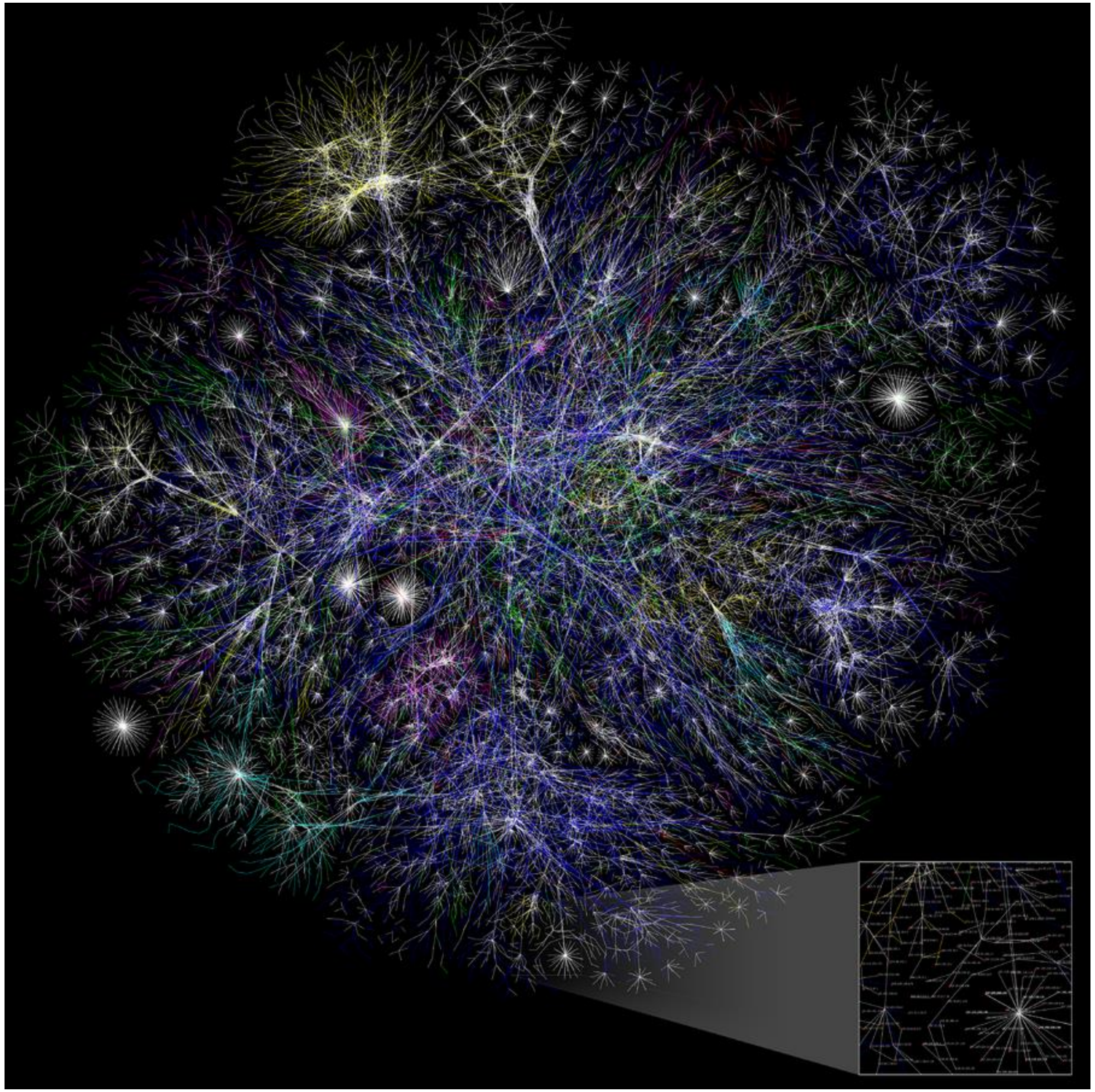
Graphs all over

Road Networks



Graphs all over

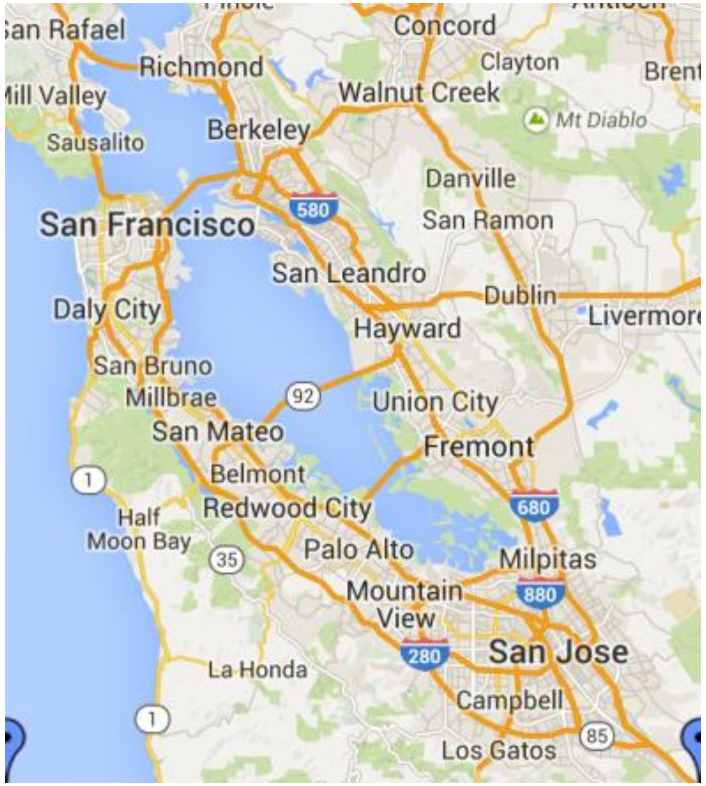
The internet

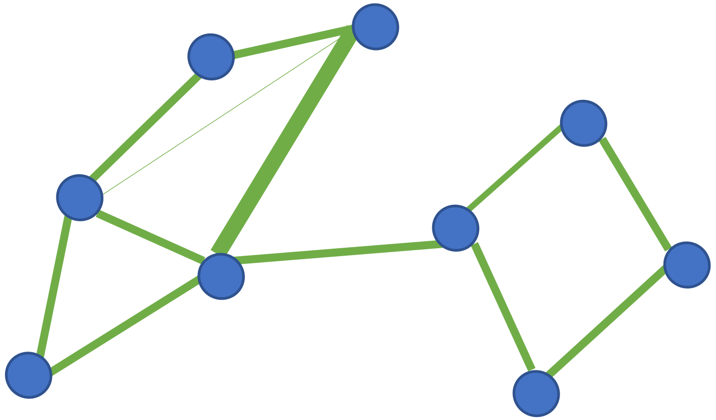


Power of abstraction

Fastest way to drive from
Berkeley to San Mateo?

Shortest path between
two nodes.

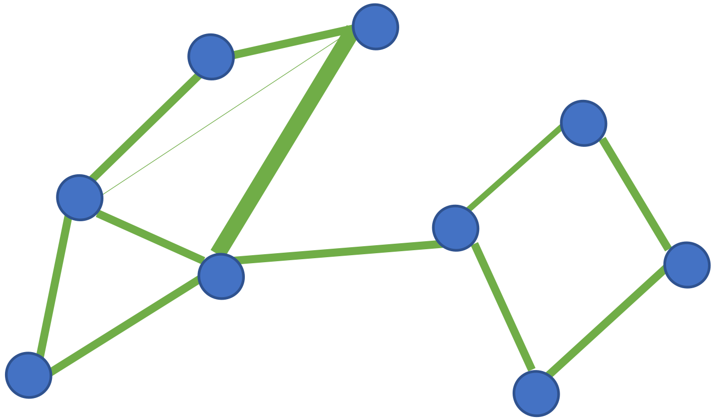




Power of abstraction

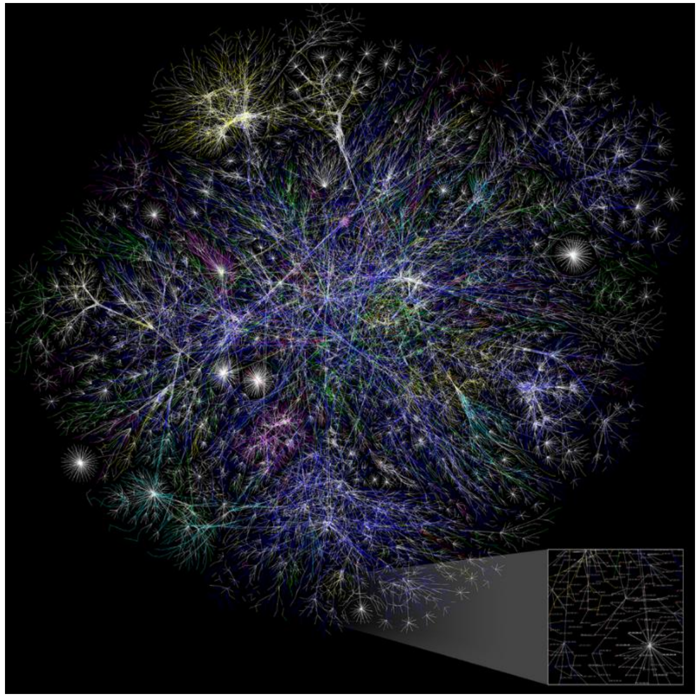
How close am I to Barack Obama?

Shortest path between two nodes.

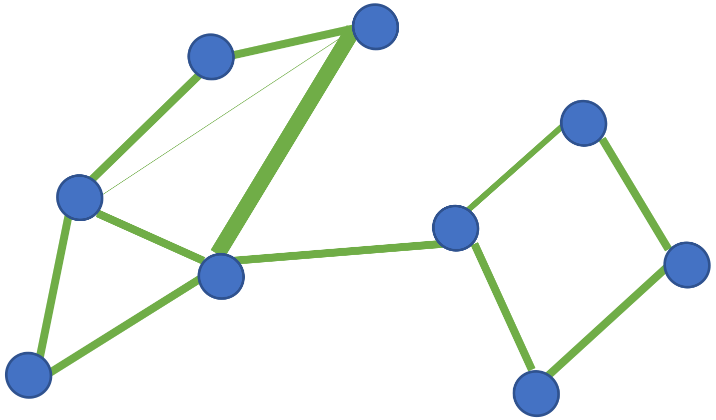


Power of abstraction

How many routers see my packets
when I google myself?



Shortest path between
two nodes. (sort of)



Graphs in theoretical Computer Science

- Algorithmic Perspective

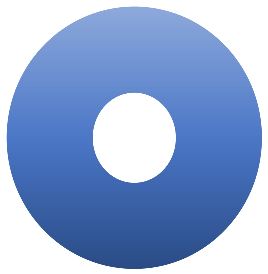
  (1) Can we design fast algorithms for general graph problems?

  Shortest path, minimum cut, largest clique, clustering, …

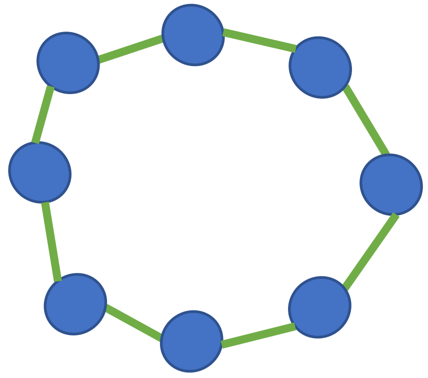
  (2) Why are some graph problems easy and others hard?

- Mathematician’s Perspective

  (3) Graph structure/geometry



vs.



Some examples (from my research)

(1) Can we design algorithms?

- Clustering algorithms
  - Arbitrary graphs
  - Specialized settings (stochastic block models)
- Computing the minimum cut when the graph isn't fully known

(2) Why are some graph problems hard?

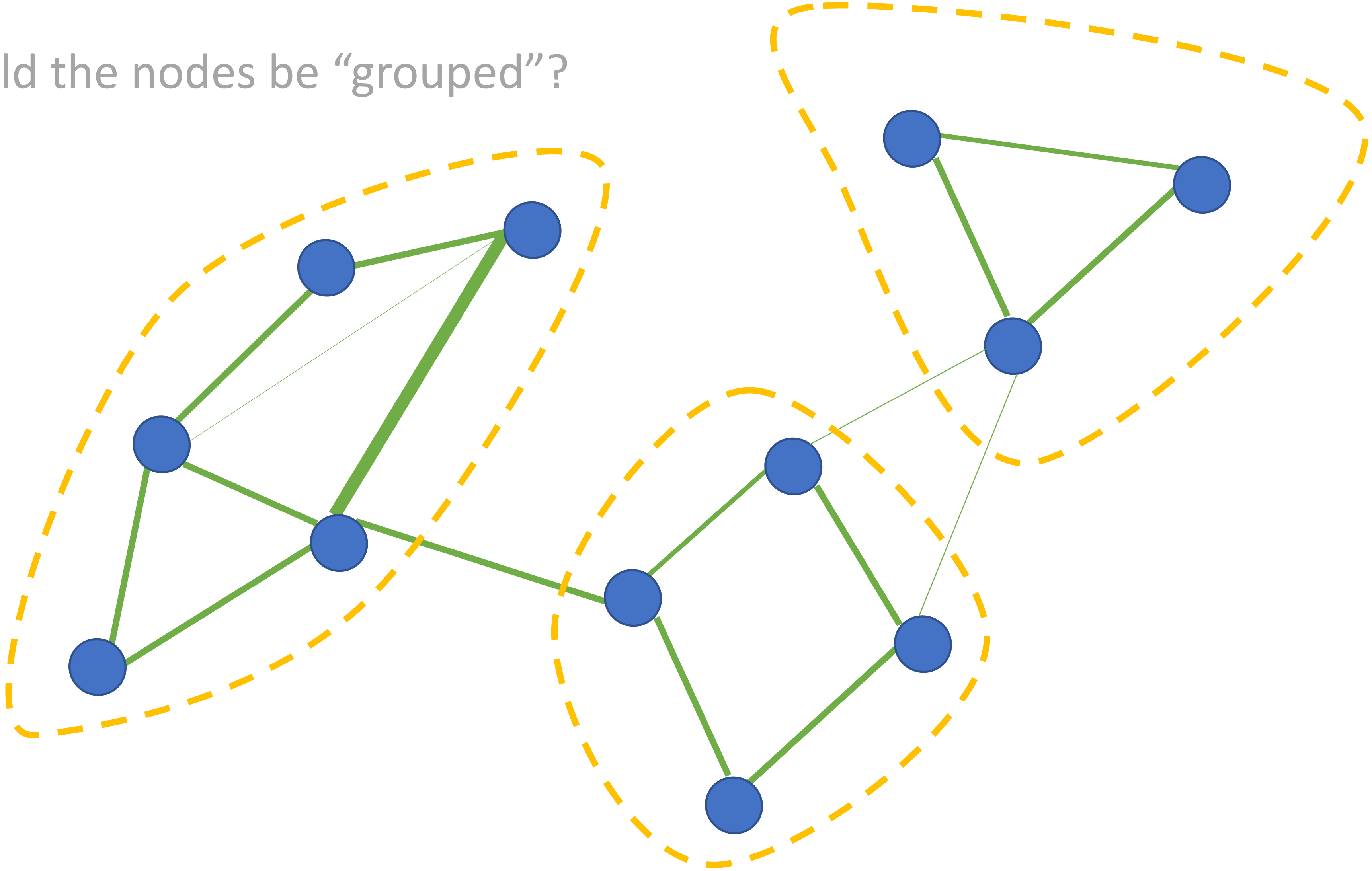
- Showing algorithms fail to find largest clique in random graphs

(3) Understanding graph structure

- Spectral ("geometric") properties of random graphs

Graph clustering

How should the nodes be “grouped”?

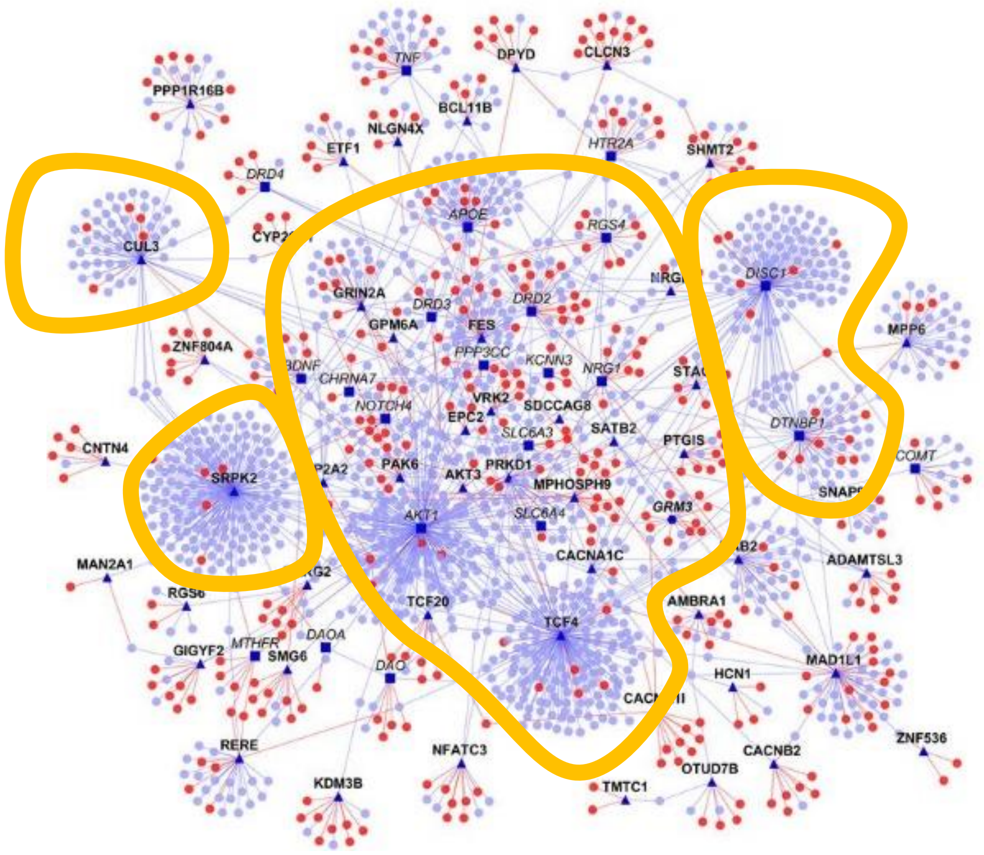


Graph clustering

How should the nodes be "grouped"?

Protein-protein interaction network:

"[Clustering] can suggest possible functions for members of the cluster which were previously uncharacterized."
From *Knowledge Discovery in Bioinformatics: Techniques, Methods and Application*



Graph clustering

NCBI Resources How To

PubMed.gov
US National Library of Medicine
National Institutes of Health

PubMed | PPI clustering | Search

Create RSS Create alert Advanced

how to cluster?

1. [Protein complex prediction via dense subgraphs and false positive **analysis**.]
Hernandez C, Mella C, Navarro G, Olivera-Nappa A, Araya J.
PLoS One. 2017 Sep 22;12(9):e0183460. doi: 10.1371/journal.pone.0183460. eCollection 2017.
PMID: 28937982

2. [Identification of protein complexes by using a spatial and temporal active protein interaction network.]
Li M, Meng X, Zheng R, Wu FX, Li Y, Pan Y, Wang J.
IEEE/ACM Trans Comput Biol Bioinform. 2017 Sep 7. doi: 10.1109/TCBB.2017.2749571. [Epub ahead of print]
PMID: 28885159
Similar articles

3. [Protein Complexes Prediction Method Based on Core-Attachment Structure and Functional Annotations.]
Li B, Liao B.
Int J Mol Sci. 2017 Sep 6;18(9). pii: E1910. doi: 10.3390/ijms18091910.
PMID: 28878201 **Free Article**
Similar articles

4. [Fusion of expression values and protein interaction information using multi-objective optimization for improving gene **clustering**.]
Dutta P, Saha S.
Comput Biol Med. 2017 Aug 1;89:31-43. doi: 10.1016/j.compbiomed.2017.07.015. [Epub ahead of print]
PMID: 28783536
Similar articles

5. [Neighbor Affinity-Based Core-Attachment Method to Detect Protein Complexes in Dynamic **PPI** Networks.]
Lei X, Liang J.
Molecules. 2017 Jul 24;22(7). pii: E1223. doi: 10.3390/molecules22071223.
PMID: 28737728 **Free Article**
Similar articles

applying clustering

6. [Bioinformatic **analysis** of computational identified differentially expressed genes in tumor stoma of pregnancy-associated breast cancer.]
Zhou Q, Sun E, Ling L, Liu X, Zhang M, Yin H, Lu C.
Mol Med Rep. 2017 Sep;16(3):3345-3350. doi: 10.3892/mmr.2017.6947. Epub 2017 Jul 12.
PMID: 28713995
Similar articles

7. [Bioinformatics **analysis** of key genes and signaling pathways associated with myocardial infarction following telomerase activation.]
Yang Y, Yang G, Du H, Dong N, Yu B.
Mol Med Rep. 2017 Sep;16(3):2915-2924. doi: 10.3892/mmr.2017.6938. Epub 2017 Jul 6.
PMID: 28713962
Similar articles

8. [Development of an in silico method for the identification of subcomplexes involved in the biogenesis of multiprotein complexes in Saccharomyces cerevisiae.]
Glatigny A, Gambette P, Bourand-Plantefol A, Dujardin G, Mucchielli-Giorgi MH.
BMC Syst Biol. 2017 Jul 11;11(1):67. doi: 10.1186/s12918-017-0442-0.
PMID: 28693620 **Free PMC Article**
Similar articles

9. [Predicting novel genes and pathways associated with osteosarcoma by using bioinformatics **analysis**.]
Dong B, Wang G, Yao J, Yuan P, Kang W, Zhi L, He X.
Gene. 2017 Sep 10;628:32-37. doi: 10.1016/j.gene.2017.06.058. Epub 2017 Jul 4.
PMID: 28687333
Similar articles

Graph clustering

How should the nodes be "grouped"?

Social Networks:

What are the "social circles" or "social groups"?

Graph clustering

Google Scholar | clustering "social networks"

Articles | About 15,800 results (**0.08** sec)

how to cluster?

[BOOK] Research methods in social network analysis
LC Freeman - 2017 - books.google.com
... How are actors **clustered** in a social network into groups or cliques? ... A symmetric and transitive relation, for example, will **cluster** actors into equivalence sets (cliques). **Clustering** and positional problems thus come late in our ordering of the logical priorities of network analysis. ...
Cited by 236 Related articles All 2 versions

Clustering by Well-Being in Workplace **Social Networks**: Homophily and Social Contagion.
J Chancellor, K Layous, S Margolis, S Lyubomirsky - 2017 - psycnet.apa.org
Abstract 1. Social interaction among employees is crucial at both an organizational and individual level. Demonstrating the value of recent methodological advances, 2 studies conducted in 2 workplaces and 2 countries sought to answer the following questions:(a) Do
Related articles All 4 versions

The contagious spread of violence among US adolescents through **social networks**
RM Bond, BJ Bushman - American journal of public ..., 2017 - ajph.aphapublications.org
... 4 A **cluster** is an "aggregation of cases of a disease that are closely grouped in ... the online version of this article at http://www.ajph.org) to assess behavior **clustering** in **social** ... analysis did not indicate a causal relationship; rather, they indicated a baseline of **clustered** behaviors on ...
Cited by 1 Related articles All 8 versions

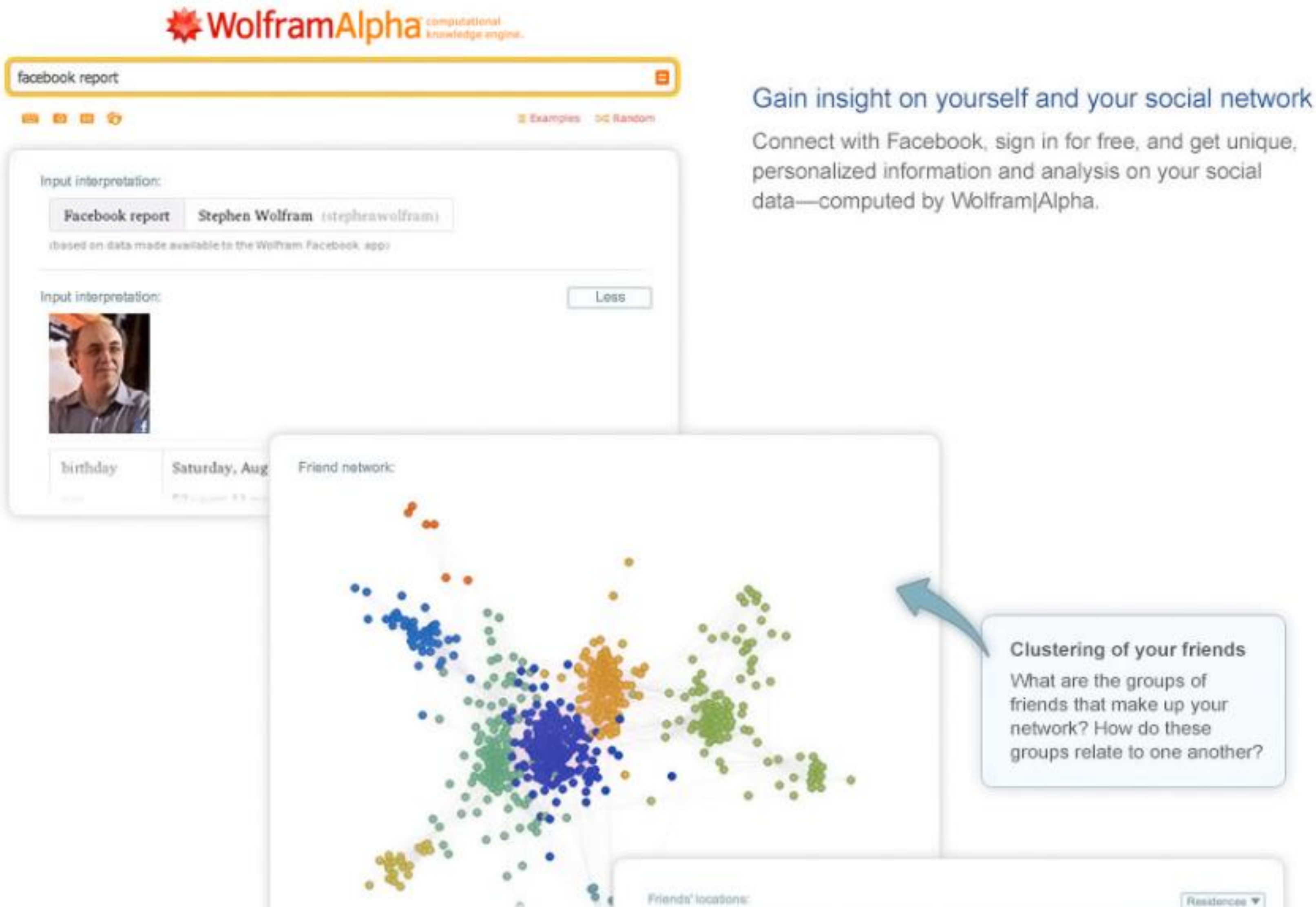
Respondent-driven sampling bias induced by community structure and response rates in **social networks**
LEC Rocha, AE Thorson, R Lambiotte... - Journal of the Royal ..., 2017 - Wiley Online Library
... Such a situation is not unlikely in highly **clustered** subpopulations where coupons may simply move around the same ... Each approach to model **social networks** has its own advantages and limitations. ... Network **clustering** is particularly important in the context of **social networks**. ...
Cited by 5 Related articles All 4 versions

Towards detecting compromised accounts on **social networks**
M Egele, G Stringhini, C Kruegel... - IEEE Transactions on ..., 2017 - ieeexplore.ieee.org
... on **Social Networks** Manuel Egele, Gianluca Stringhini, Christopher Kruegel, and Giovanni Vigna, Member, IEEE ... Ç 1 INTRODUCTION ONLINE **social networks**, such as Facebook and Twitter, have become one of the main media to stay in touch with the rest of the world. ...
Cited by 10 Related articles All 10 versions

Media fragmentation in the context of bounded **social networks**: How far can it go?
JM Riles, A Pilny, D Tewksbury - New Media & Society, 2017 - journals.sagepub.com
Related articles

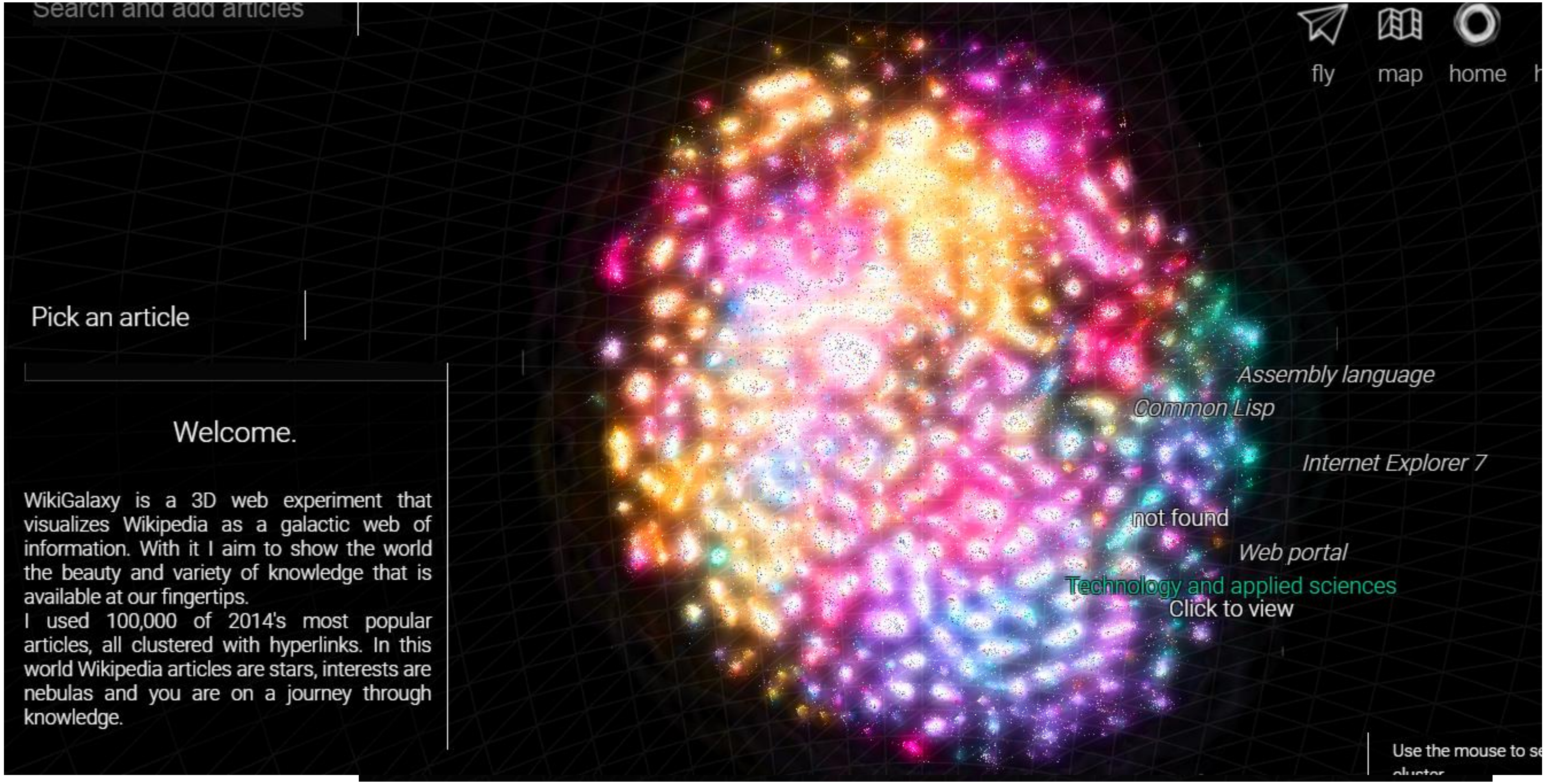
applying clustering

Graph clustering



Graph clustering

Graph clustering



Outline

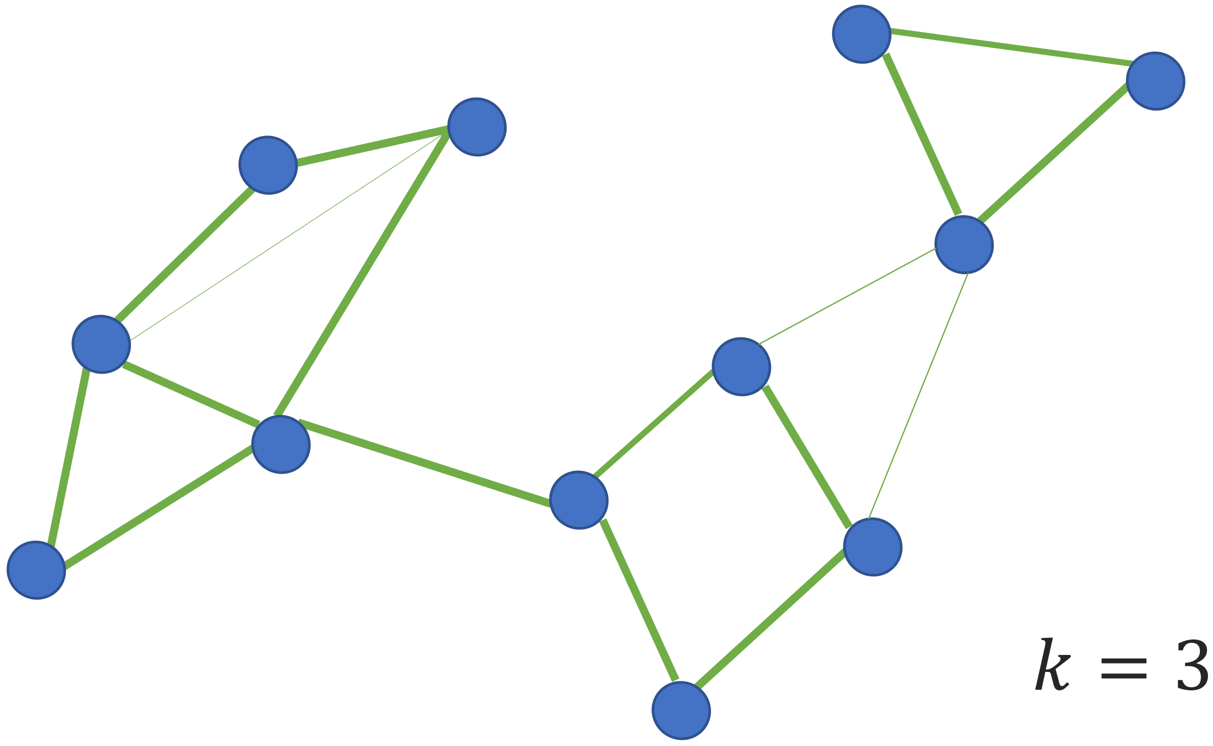
- Formalizing graph clustering?
  pros & cons of some popular formulations
- The "Spectral Embedding"
  from graphs to geometry
- Clustering Graphs with the spectral embedding
  graphs → geometry → k-means++

Formalizing graph clustering

What is it we want, *exactly*?

What is it we want, *exactly*?

Best partition of graph into k pieces.



$k = 3$

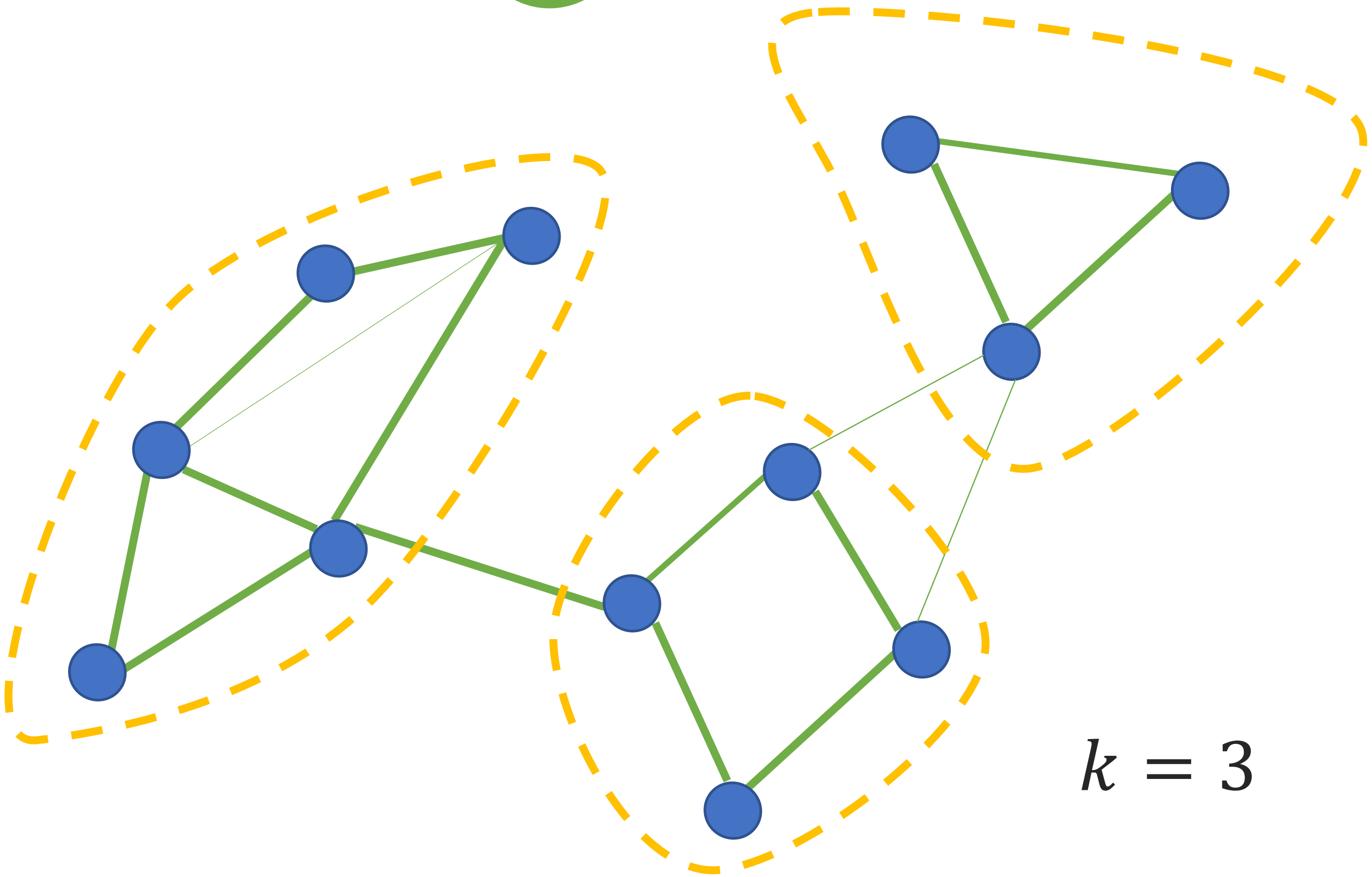
What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 1: cut as few edges as possible?

minimize $(\# \text{ edges cut})$

"objective function"



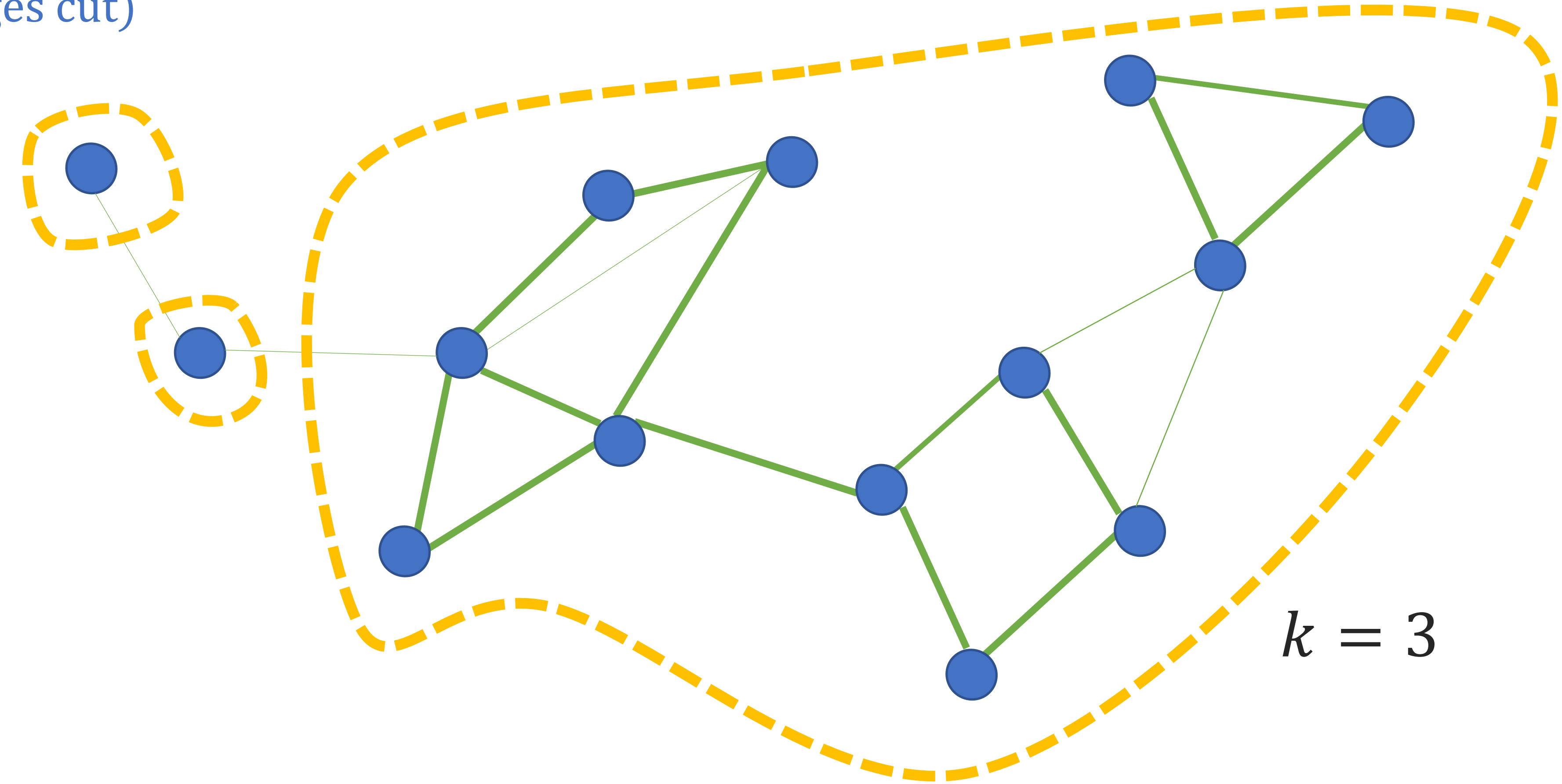
$k = 3$

What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 1: cut as few edges as possible?

minimize $\quad$ (# edges cut)



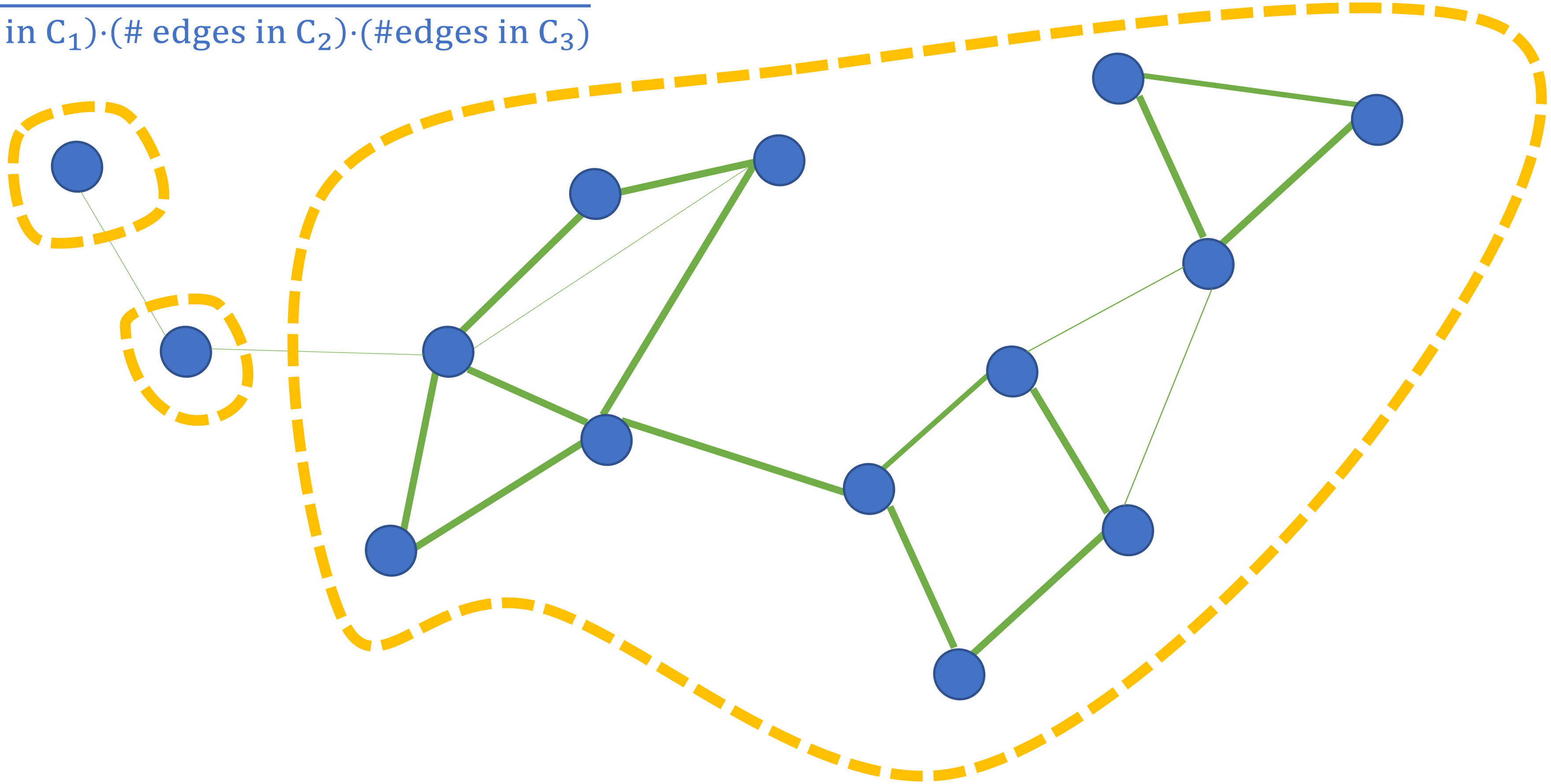
What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 2: the "sparsest" cut?

$$\text{minimize} \quad \frac{(\#\text{ edges cut})}{(\#\text{ edges in } C_1)\cdot(\#\text{ edges in } C_2)\cdot(\#\text{edges in } C_3)}$$

cost = ∞

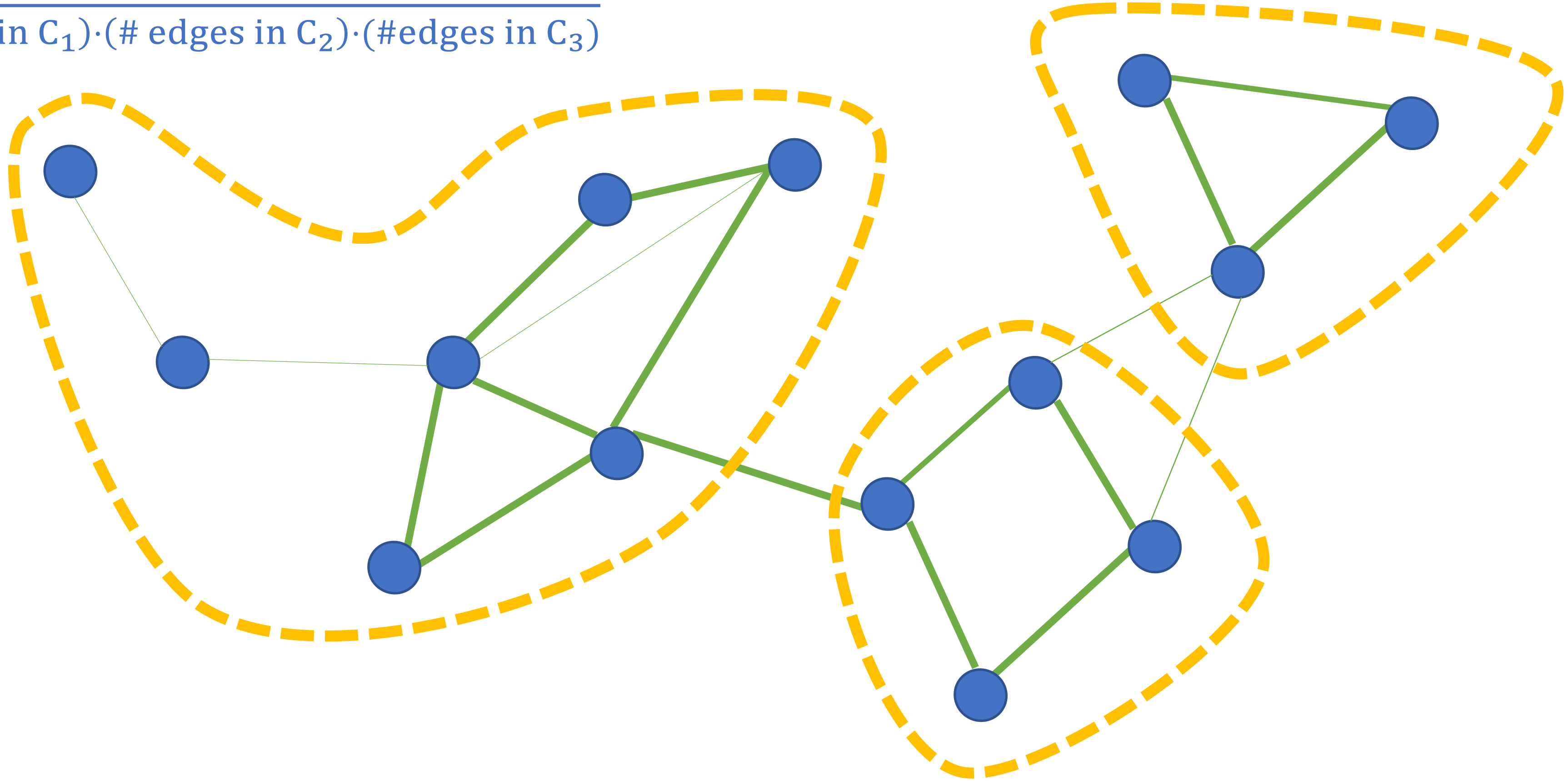


What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 2: the "sparsest" cut?

minimize $\dfrac{\text{(\# edges cut)}}{\text{(\# edges in } C_1) \cdot \text{(\# edges in } C_2) \cdot \text{(\#edges in } C_3)}$



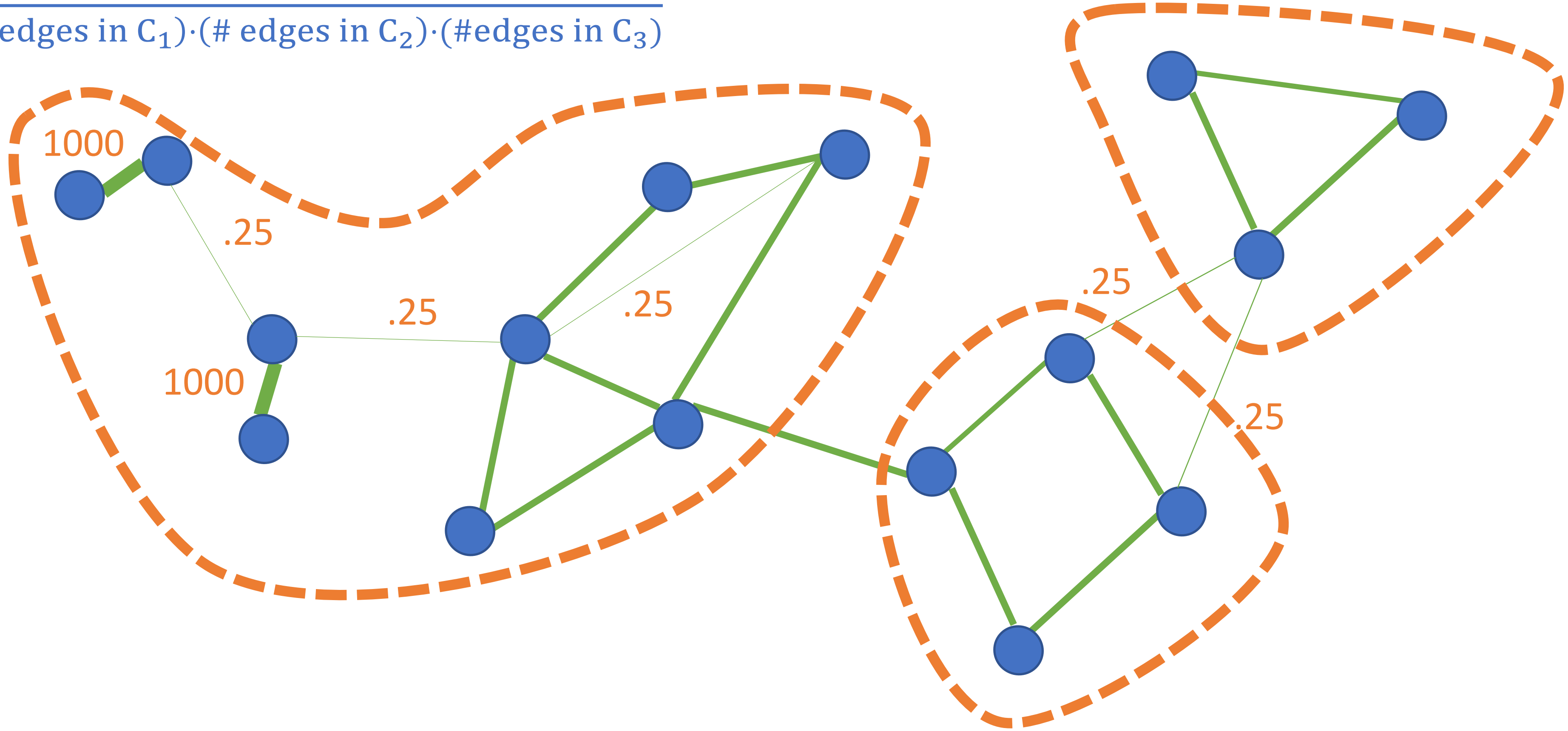
What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 2: the "sparsest" cut?

$$\text{cost} = \frac{.25+.25+1}{(2000+.75+6)\times 4\times 3} \approx 10^{-5}$$

minimize $\dfrac{(\text{\# edges cut})}{(\text{\# edges in } C_1)\cdot(\text{\# edges in } C_2)\cdot(\text{\#edges in } C_3)}$



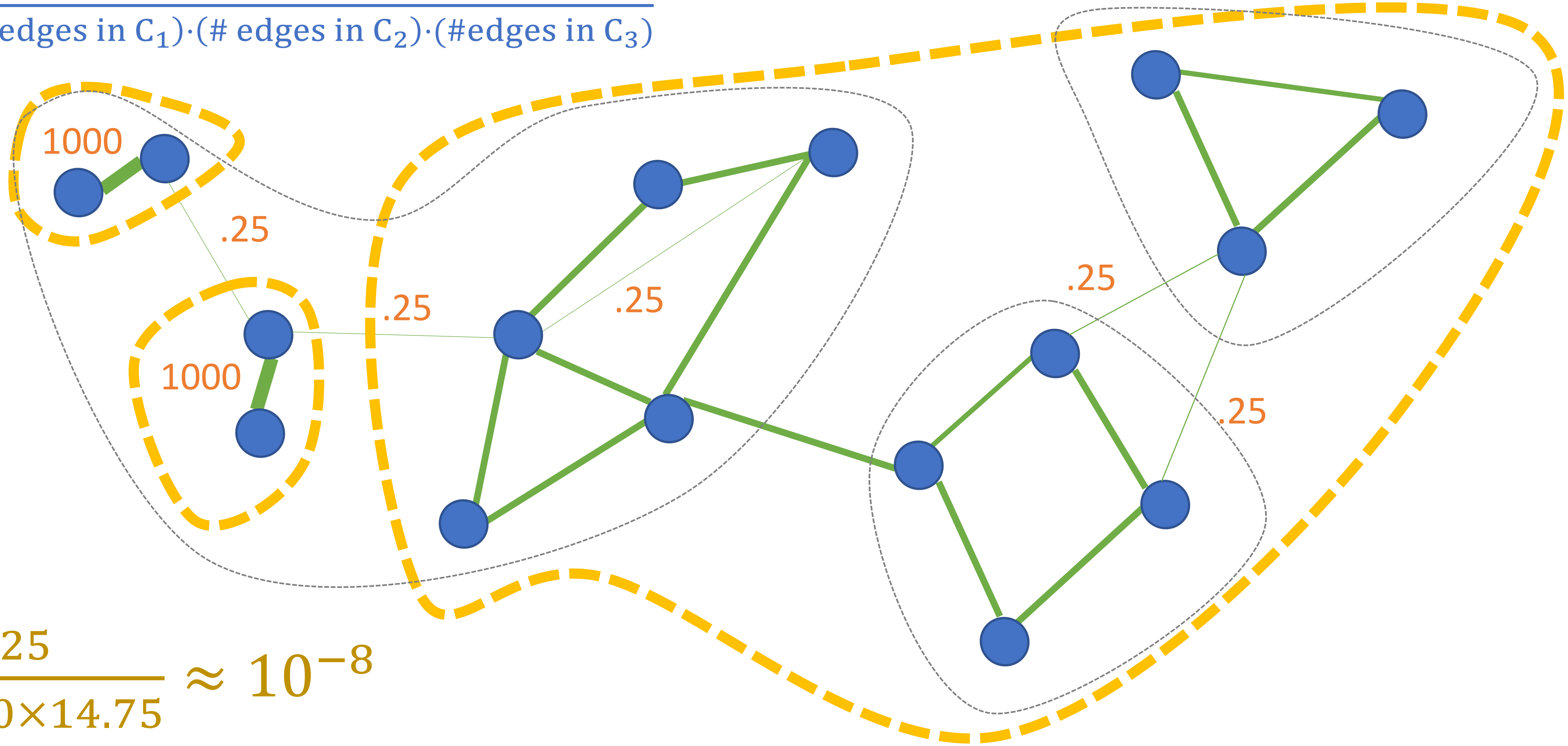
What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 2: the "sparsest" cut?

$$\text{cost} = \frac{.25 + .25 + 1}{(2000 + .75 + 6) \times 4 \times 3} \approx 10^{-5}$$

minimize $\dfrac{(\#\text{ edges cut})}{(\#\text{ edges in } C_1)\cdot(\#\text{ edges in } C_2)\cdot(\#\text{edges in } C_3)}$



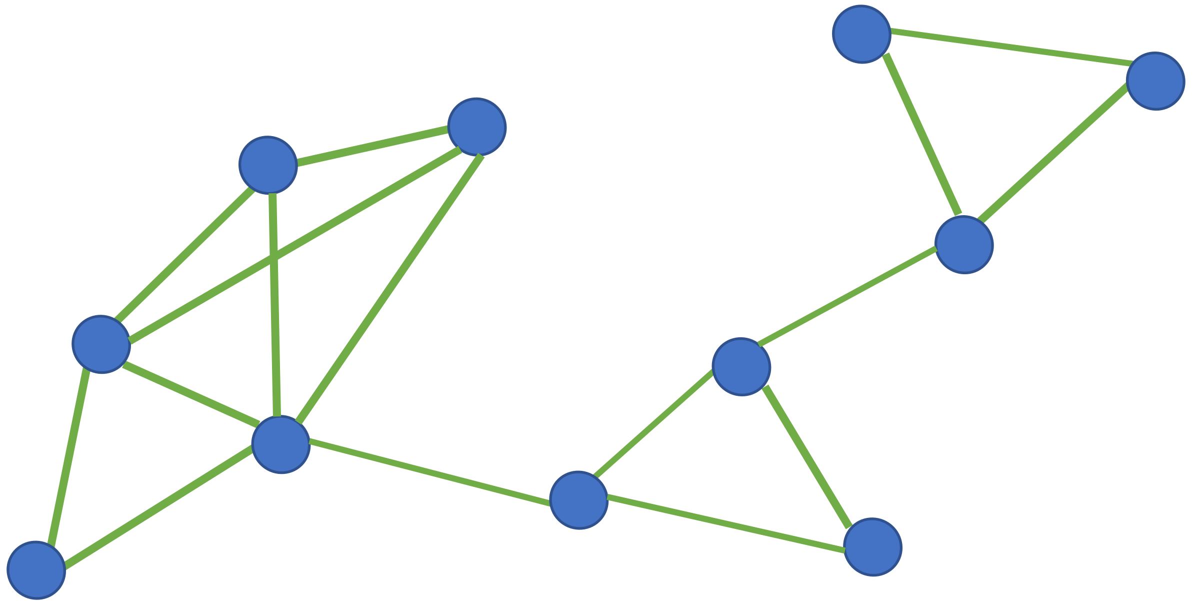
$$\text{cost} = \frac{.25 + .25}{1000 \times 1000 \times 14.75} \approx 10^{-8}$$

What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 3: similar together, different apart?

minimize $\quad (\#\text{ edges cut}) + (\#\text{ non} - \text{edges not cut})$

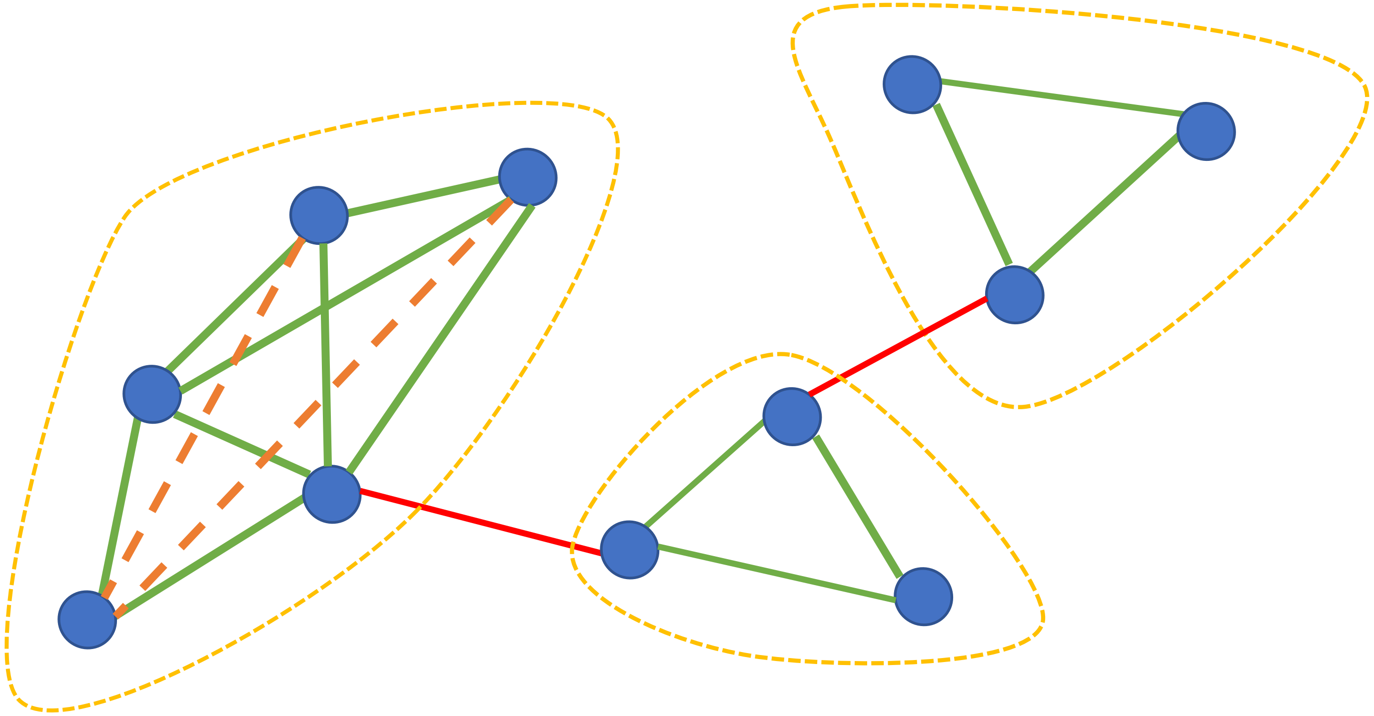


What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 3: similar together, different apart?

minimize $\quad (\#\text{ edges cut}) + (\#\text{ non} - \text{edges not cut})$



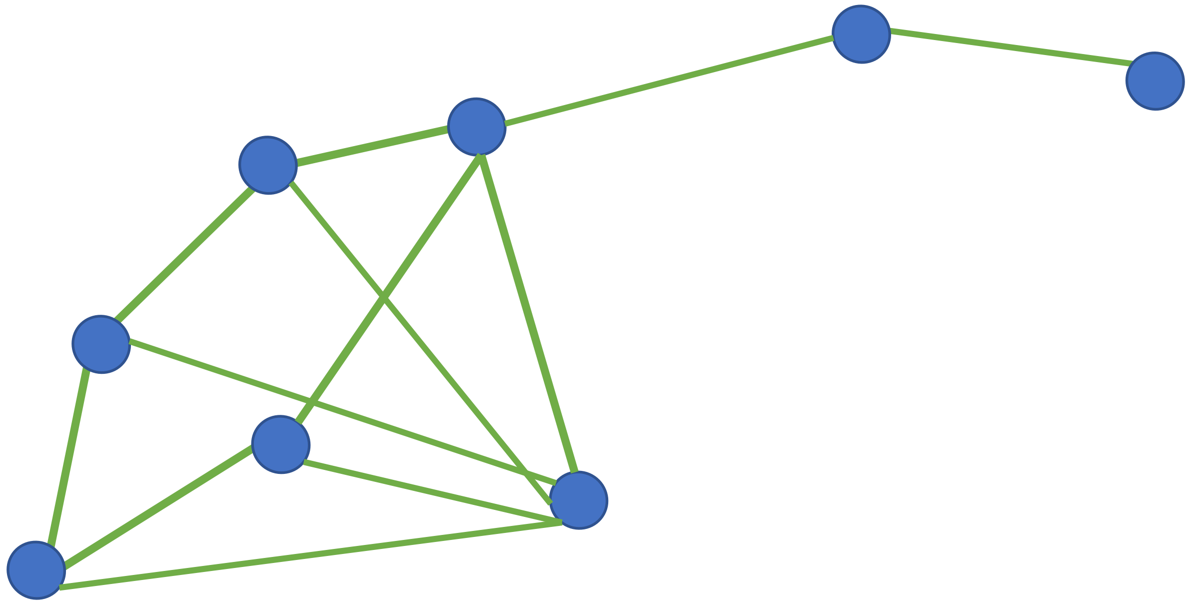
$\text{cost} = 2 + 2 = 4$

What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 3: similar together, different apart?

$$\text{minimize} \quad (\#\text{ edges cut}) + (\#\text{ non} - \text{edges not cut})$$



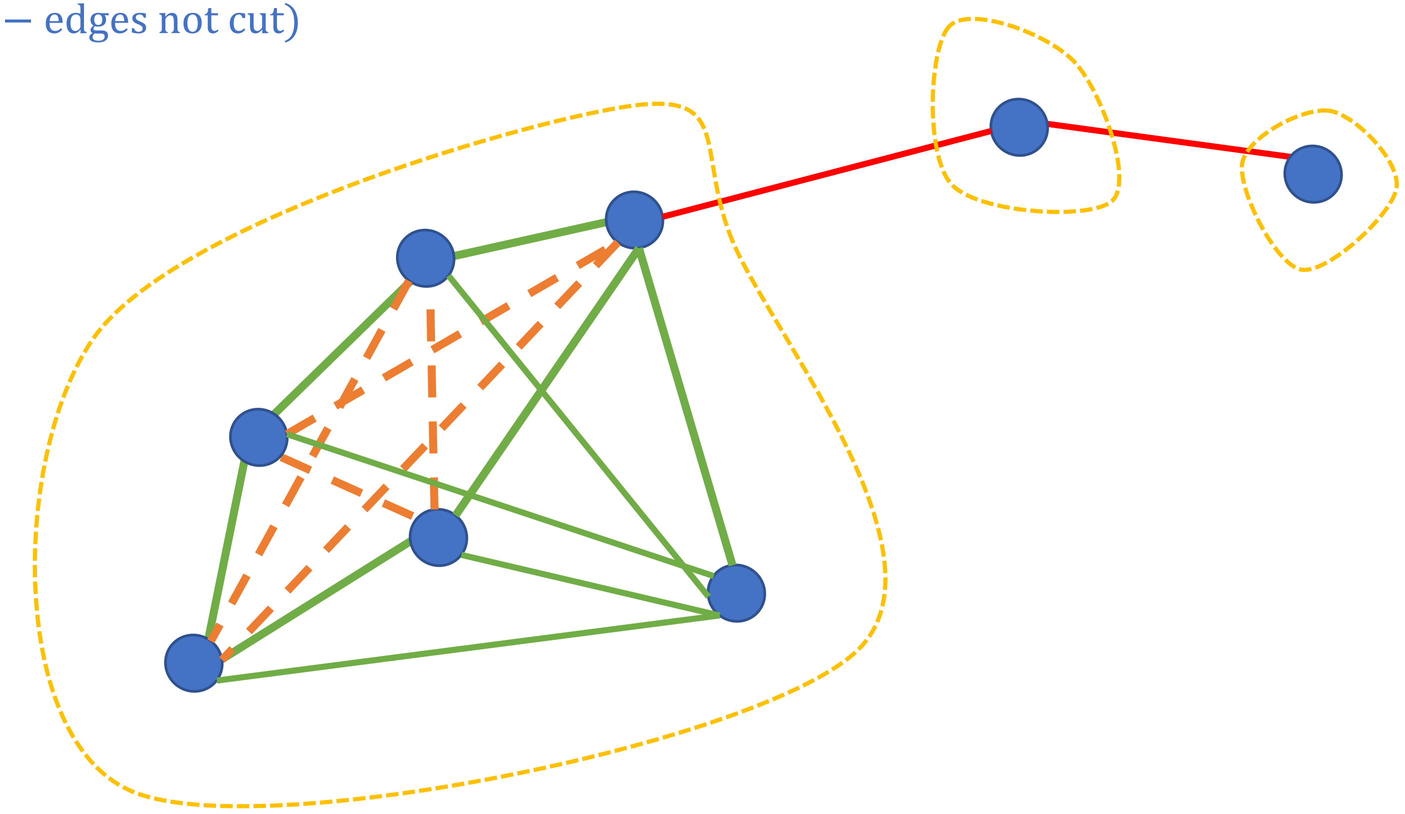
What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 3: similar together, different apart?

$$\text{minimize} \quad (\#\text{ edges cut}) + (\#\text{ non} - \text{edges not cut})$$

$$\text{cost} = 5 + 2 = 7$$



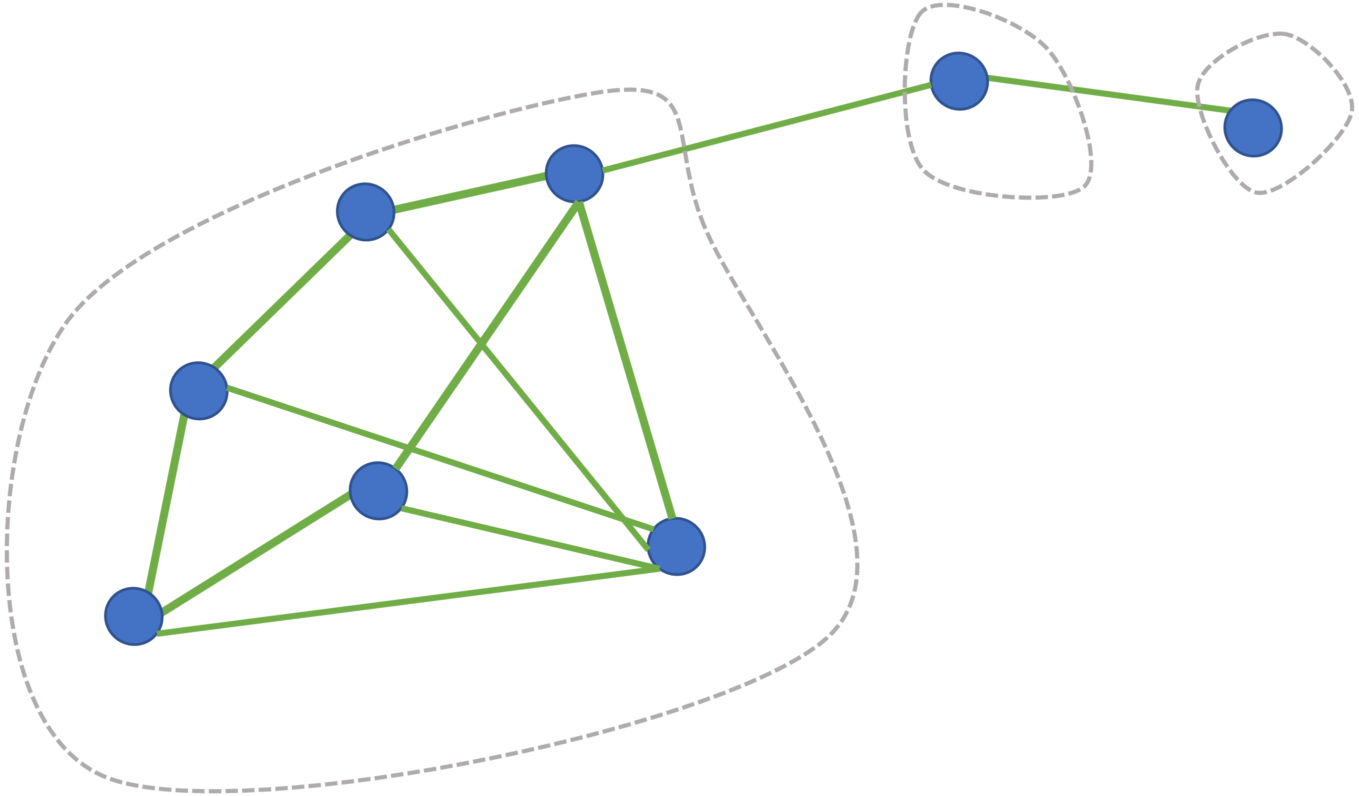
What is it we want, *exactly*?

Best partition of graph into k pieces.

Attempt 3: similar together, different apart?

$$\text{minimize} \quad (\#\text{ edges cut}) + (\#\text{ non} - \text{edges not cut})$$

$\text{cost} = 5 + 2 = 7$



What is it we want, *exactly*?

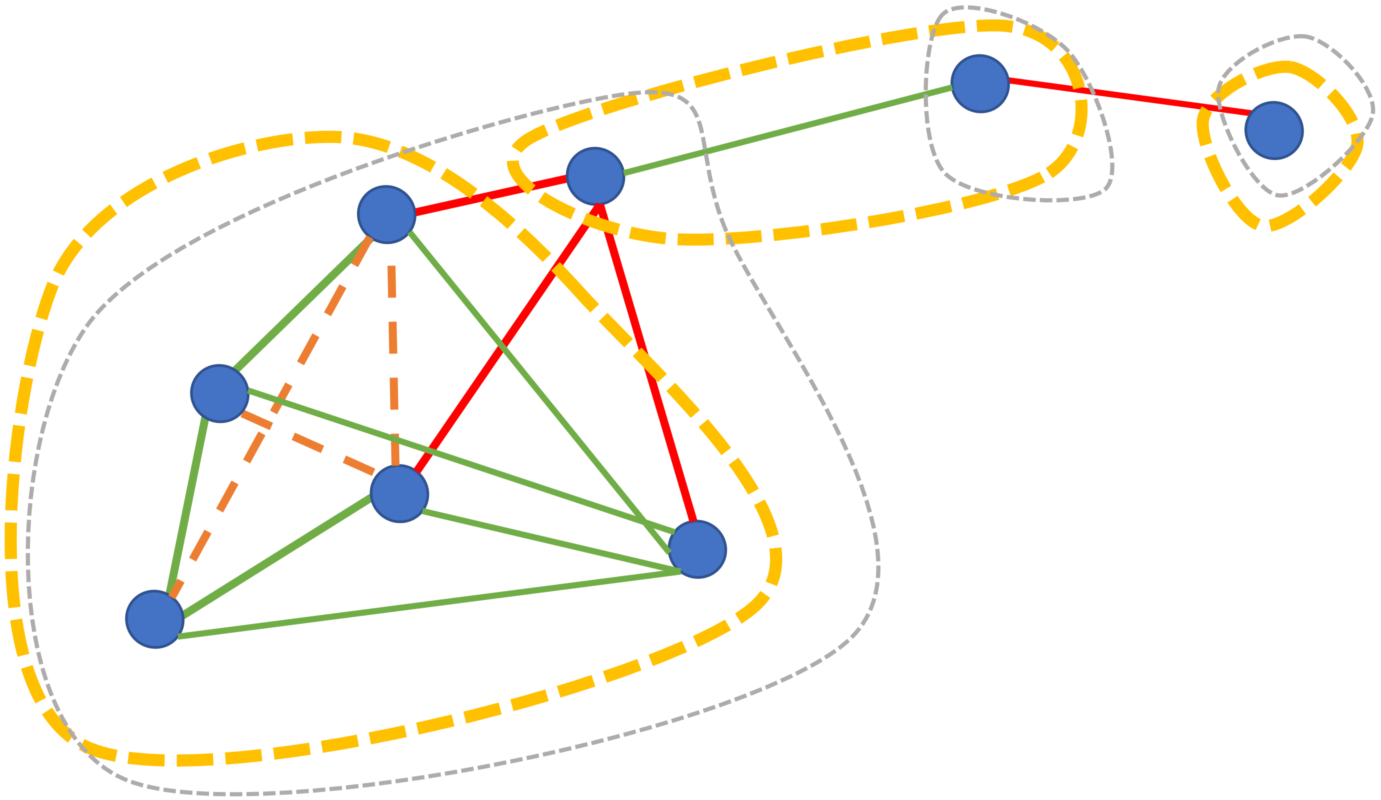
Best partition of graph into k pieces.

Attempt 3: similar together, different apart?

$$\text{minimize} \quad (\#\text{ edges cut}) + (\#\text{ non} - \text{edges not cut})$$

$\text{cost} = 5 + 2 = 7$

$\text{cost} = 3 + 4 = 7$



Theory vs. Practice

Theory:

Want to prove *general* guarantees. Requires optimizing a *fixed* objective.

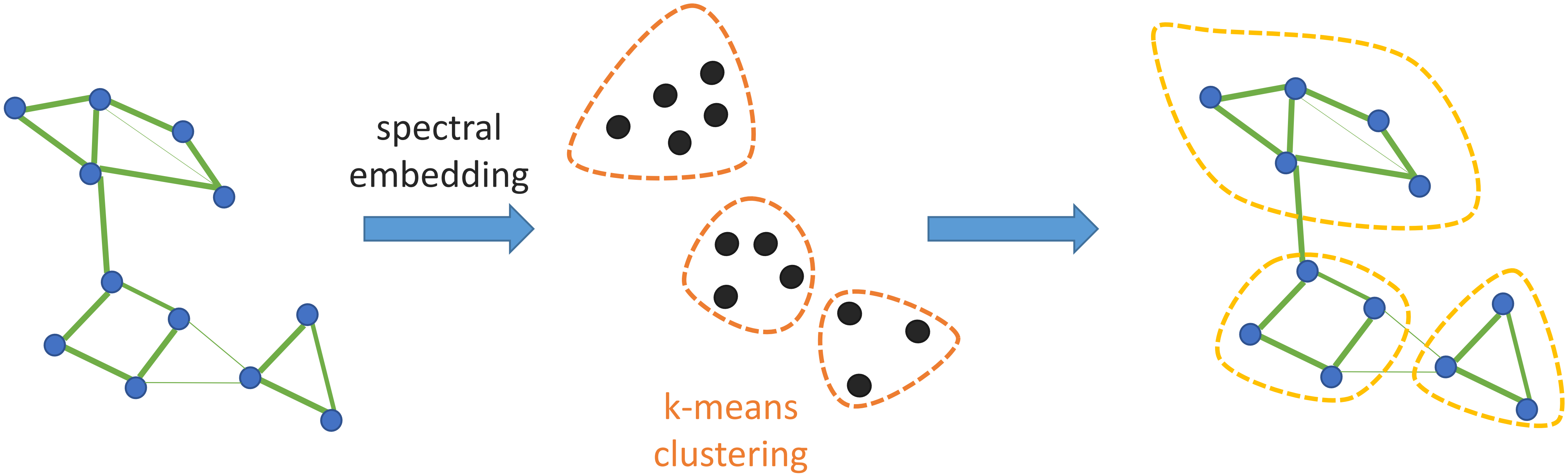
maybe under assumptions about the graph

Practice:

Want a *good clustering*. Mixture of theory & tweaking (pre- and post-processing).

Spectral Embedding + k-means

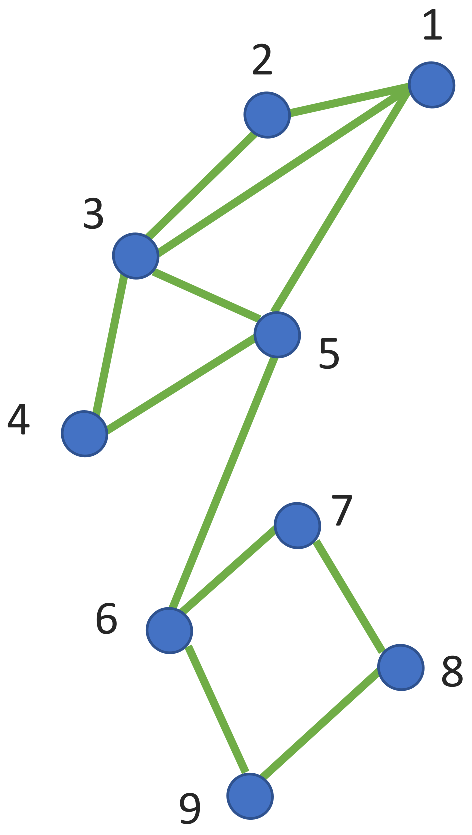
Our strategy:



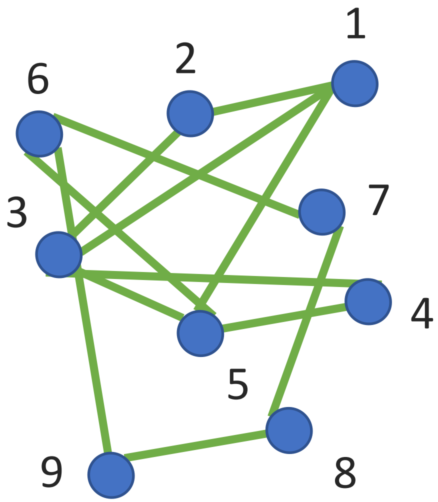
From graphs to geometry

The Spectral Embedding

How to “see” the clusters?

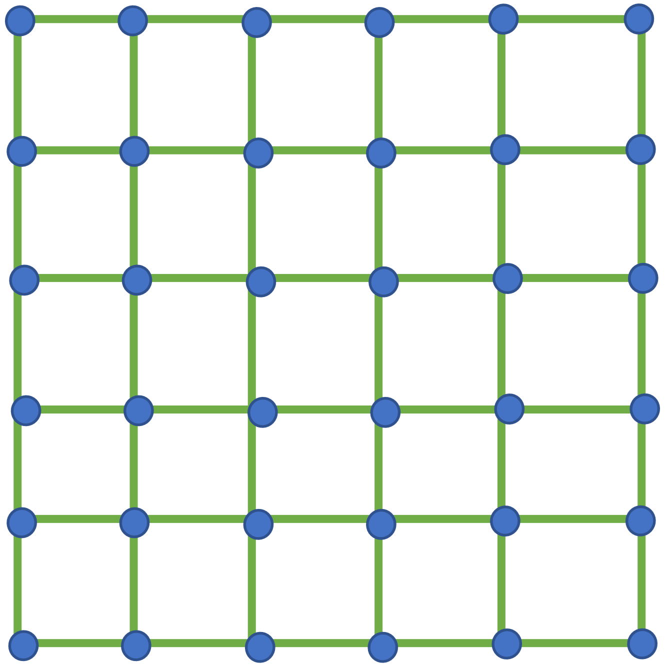


1 -> 2, 3, 5
2 -> 1, 3
3 -> 1, 2, 4, 5
4 -> 3, 5
5 -> 1, 3, 4, 6
6 -> 5, 7, 9
7 -> 6, 8
8 -> 7, 9
9 -> 6, 8

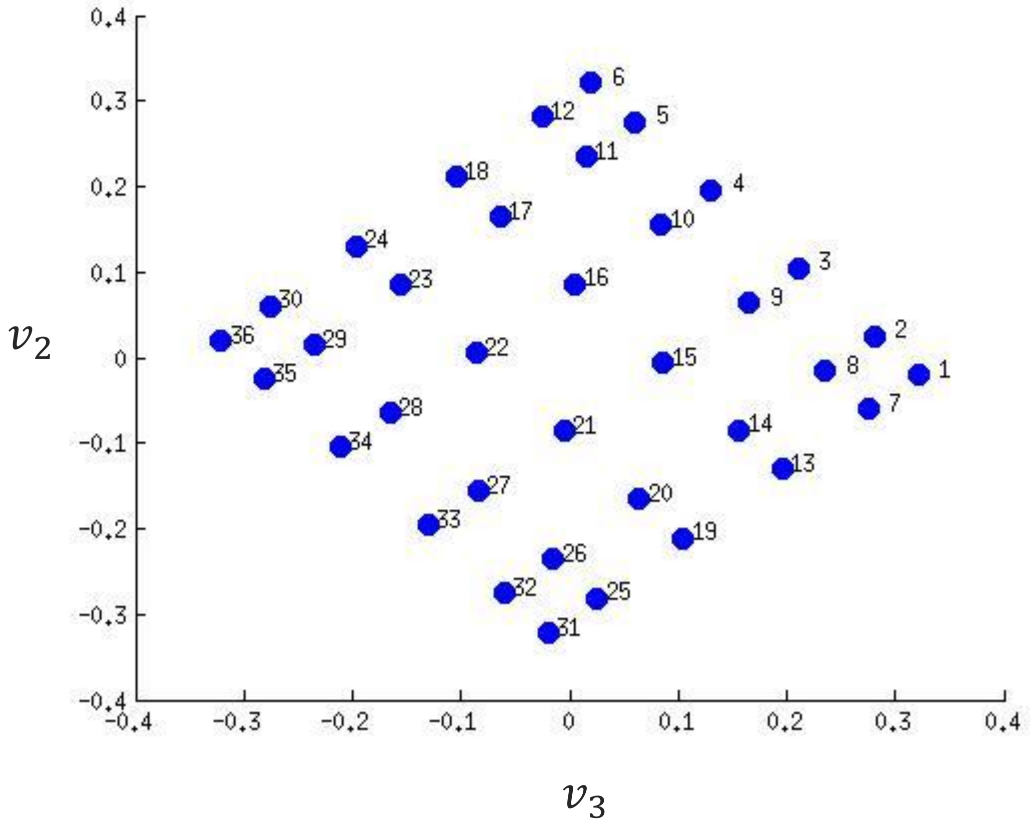


The Spectral Embedding

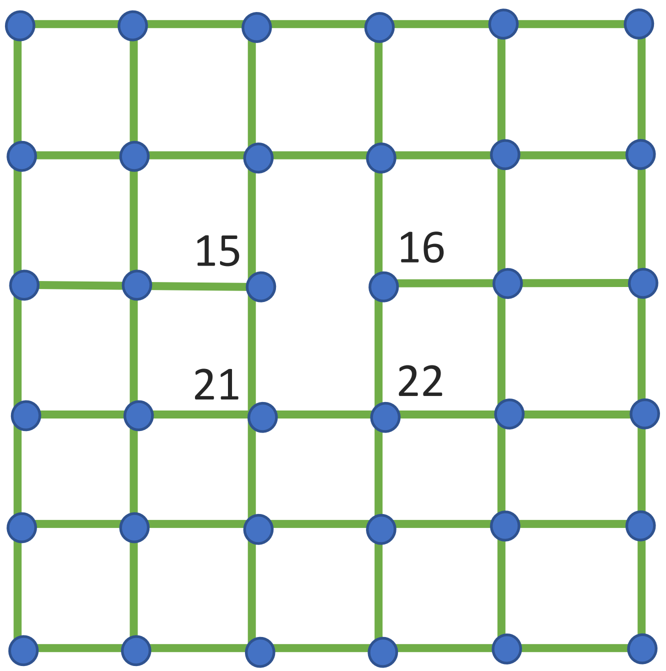
Spectral Embedding in action



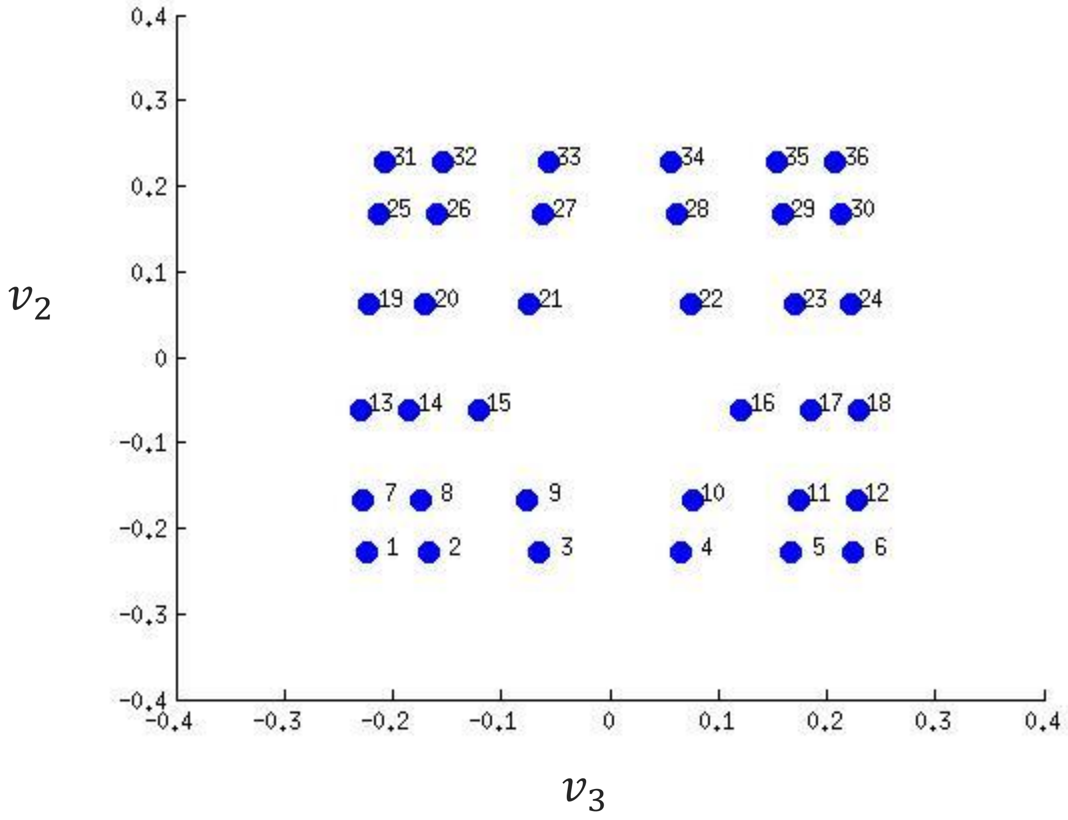
2D spectral embedding of grid



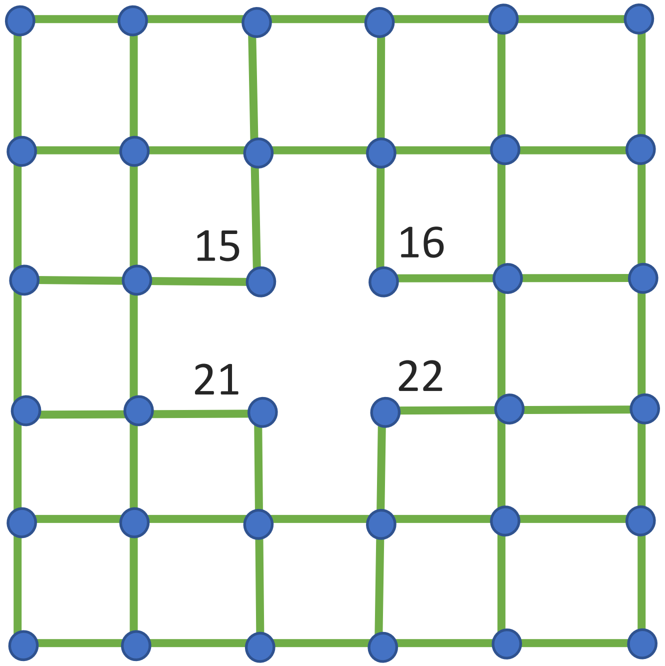
Spectral Embedding in action



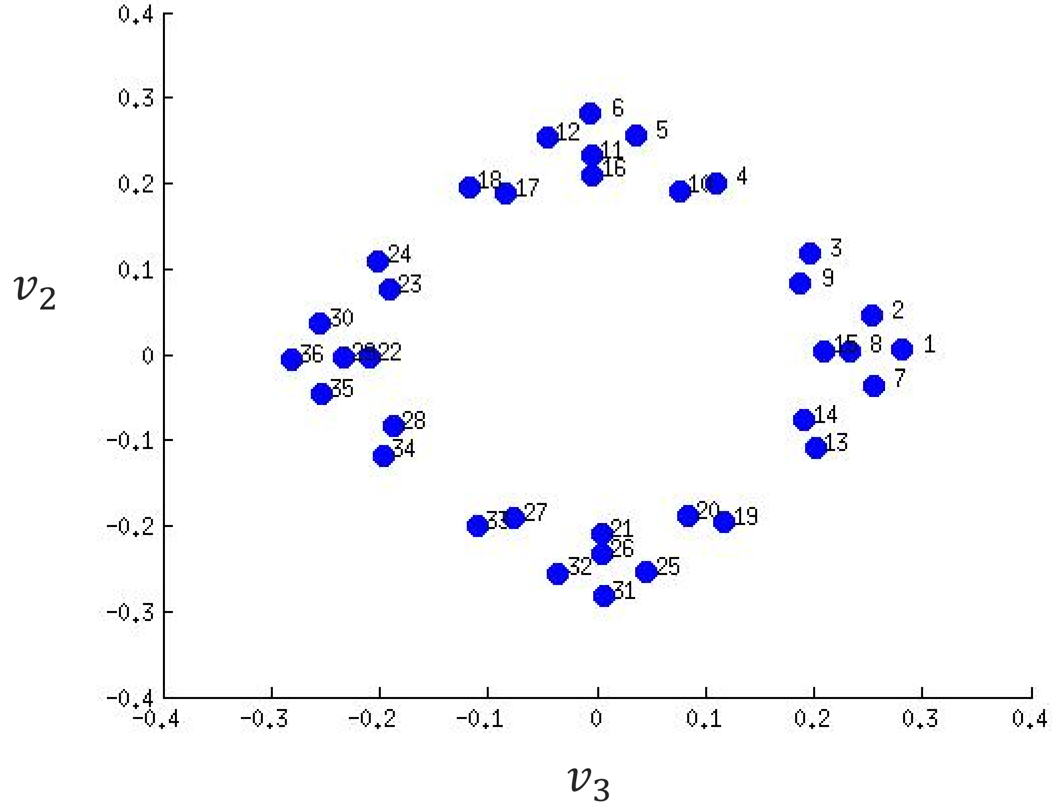
2D spectral embedding of broken grid



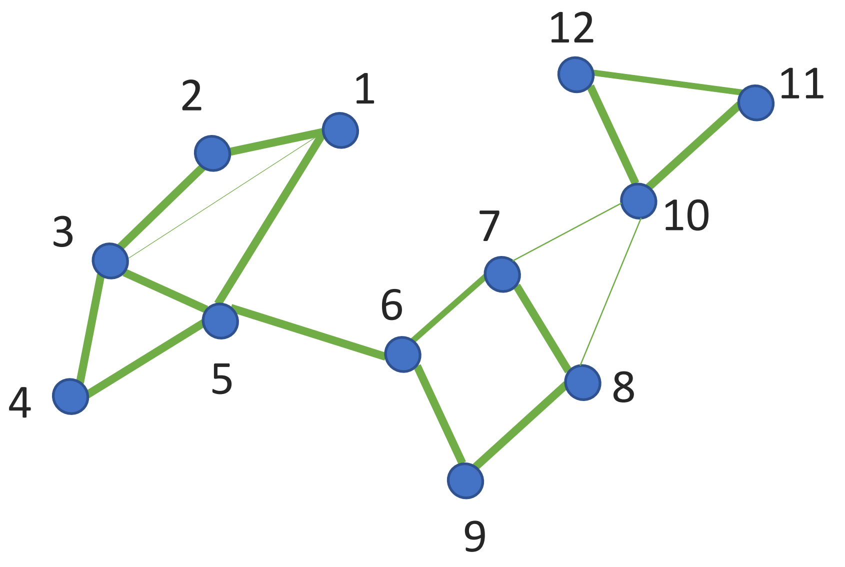
Spectral Embedding in action



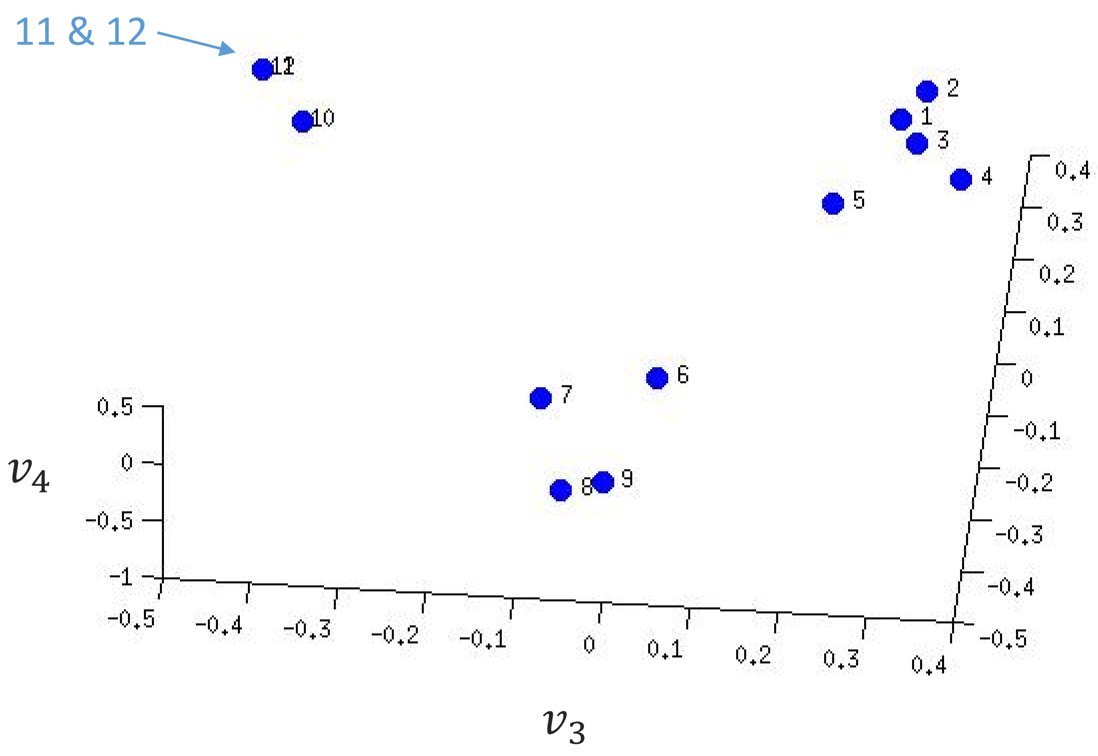
2D spectral embedding of broken grid



Spectral Embedding in action



3D spectral embedding of broken grid



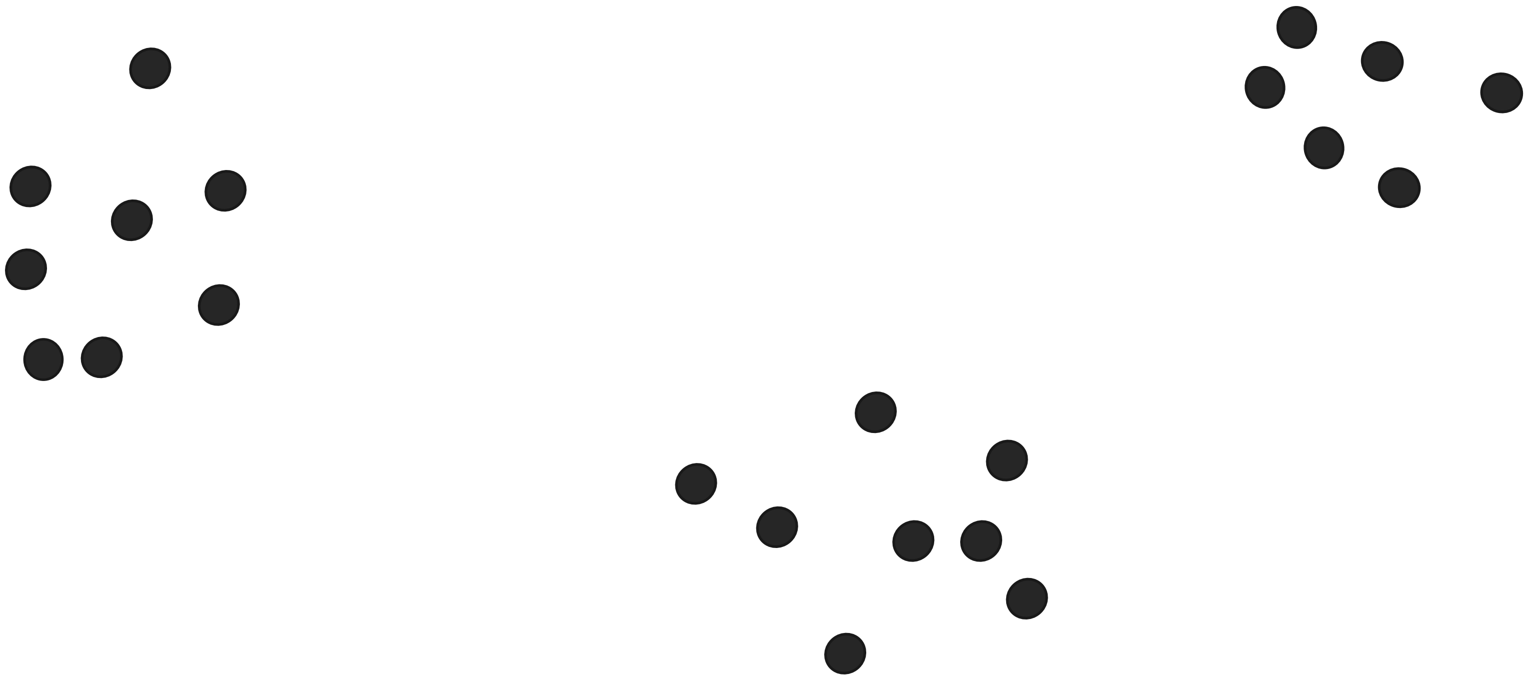
k-means clustering

Geometric Clustering: k-means

$$\underset{\substack{C \\ |C|=k}}{\text{minimize}} \sum_{i \in S} \min_{c \in C} d(i,c)^2$$

Given: set S of n points in space (e.g. $\mathbb{R}^n$)

Choose a set C of k “centers” that minimize sum of squared distances.

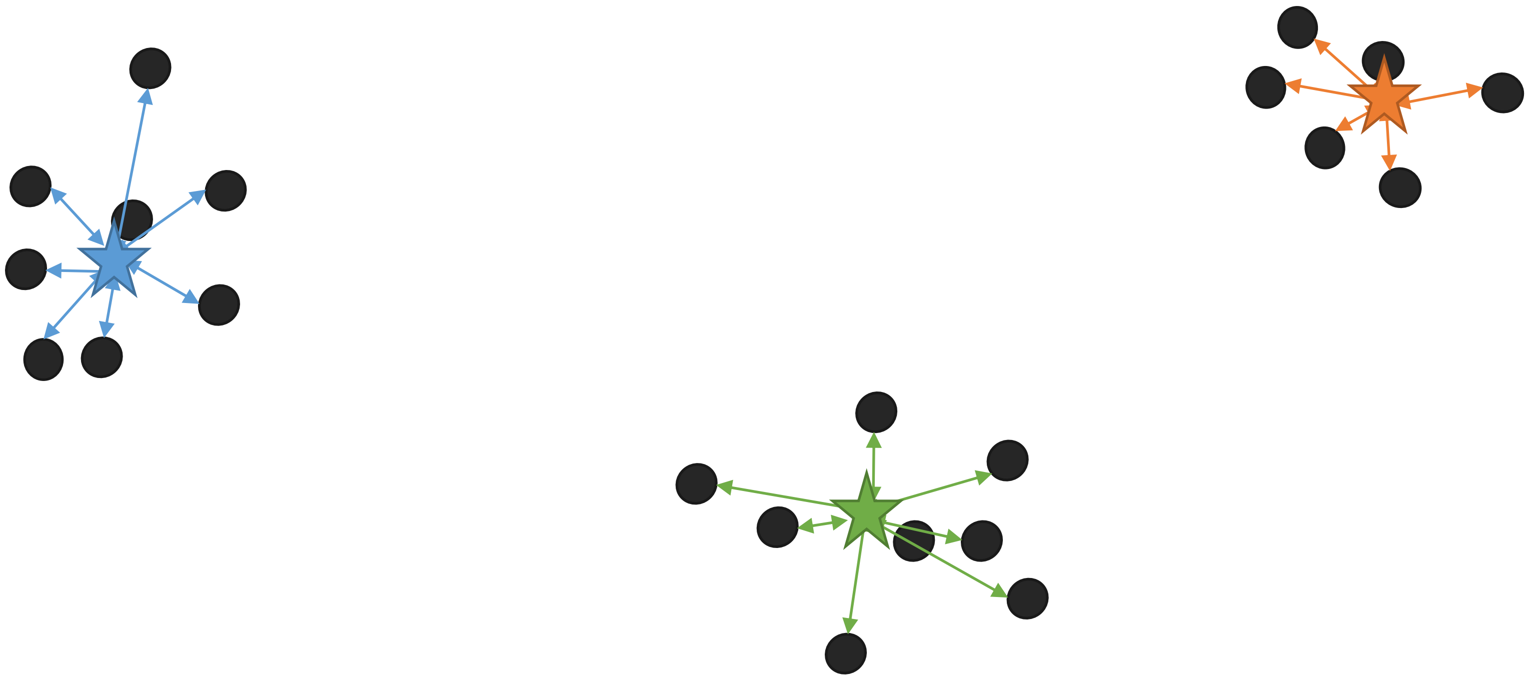


Geometric Clustering: k-means

$$\underset{\substack{C \\ |C|=k}}{\text{minimize}} \sum_{i \in S} \min_{c \in C} d(i,c)^2$$

Given: set S of n points in space (e.g. $\mathbb{R}^n$)

Choose a set C of k “centers” that minimize sum of squared distances.



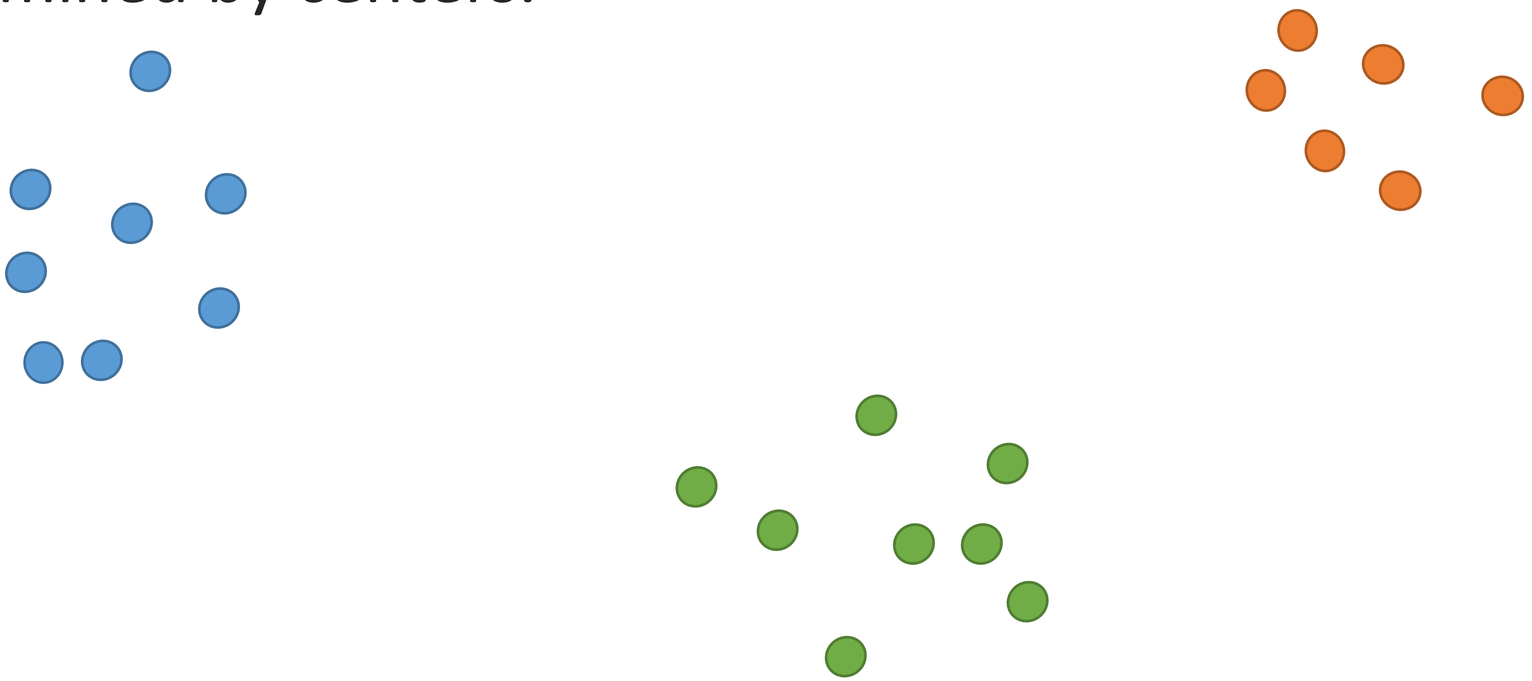
Geometric Clustering: k-means

$$\underset{\substack{C \\ |C|=k}}{\text{minimize}} \sum_{i \in S} \min_{c \in C} d(i,c)^2$$

Given: set S of n points in space (e.g. $\mathbb{R}^n$)

Choose a set C of k “centers” that minimize sum of squared distances.

Clusters determined by centers.



k-means

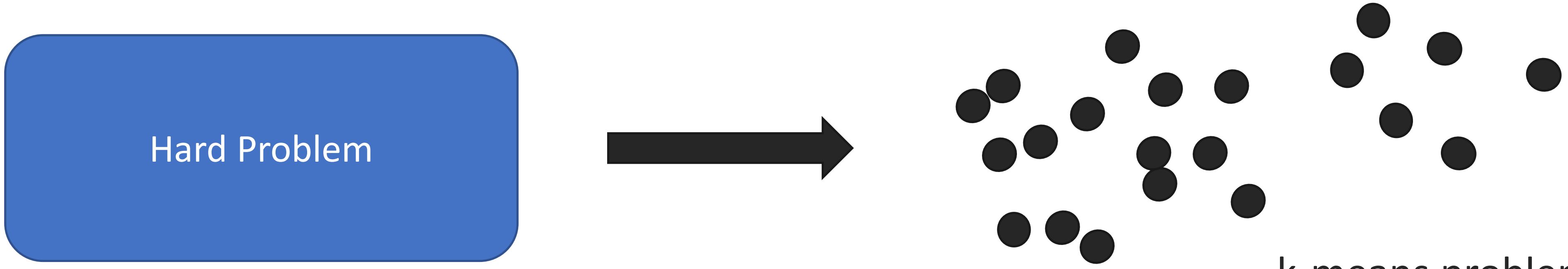
History: [edit]

The term "*k*-means" was first used by James MacQueen in 1967,[2] though the idea goes back to Hugo Steinhaus in 1957.[3] The standard algorithm was first proposed by Stuart Lloyd in 1957 as a technique for pulse-code modulation, though it wasn't published outside of Bell Labs until 1982.[4] In 1965, E. W. Forgy published essentially the same method, which is why it is sometimes referred to as Lloyd-Forgy.[5]

Can’t always get what you want…

Finding the optimal set of centers is **NP-hard**.

If there is an algorithm that exactly solves k-means efficiently, then there is an algorithm that solves many “hard” problems efficiently (*too* efficiently).



k-means problem
(even for $k = 2$)

Practice, and theory

There are k-means algorithms that work well in practice.

"the k-means algorithm" [Lloyd '57]
k-means++ [Ostrovsky-Rabani-Schulman-Swamy '06, Arthur-Vassilvitskii '07]

There are k-means algorithms that work *ok* in theory.

Can get a 2.611-approximation [Byrka-Pensyl-Rybicki-Srinivasan-Trinh '15]

There are k-means algorithms that work *well* in theory **if** we compromise.

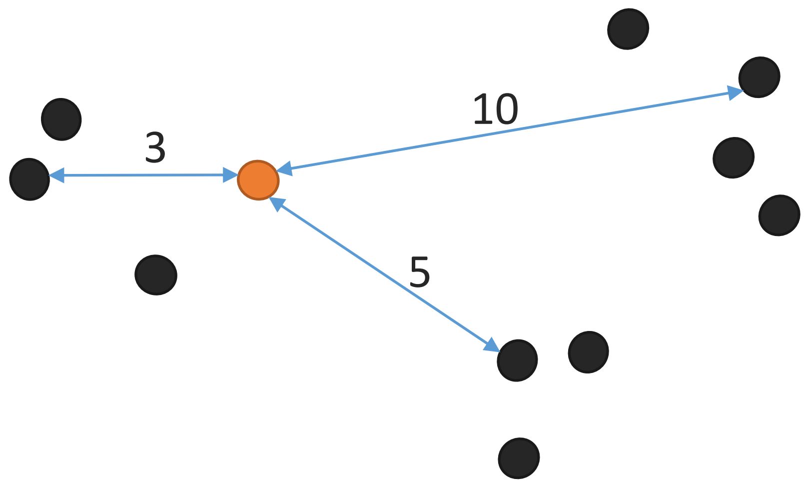
If k is small [Kumar-Subharwal-Sen '04]

If we use $10k$ centers instead of k centers [Makarychev-Makarychev-Sviridenko-Ward '15]

If we know how some points should be clustered [Ailon-Bhattacharya-Jaiswal-Kumar '17]

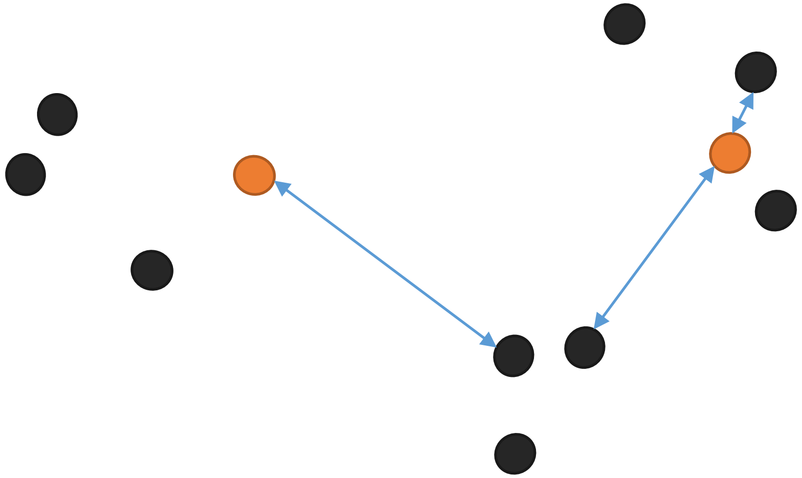
Algorithm: k-means++

1) Initialize C with a random point
2) While there are $< k$ centers
    1) Add x to C with probability proportional to $\min_{c \in C} d(x,c)^2$



Algorithm: k-means++

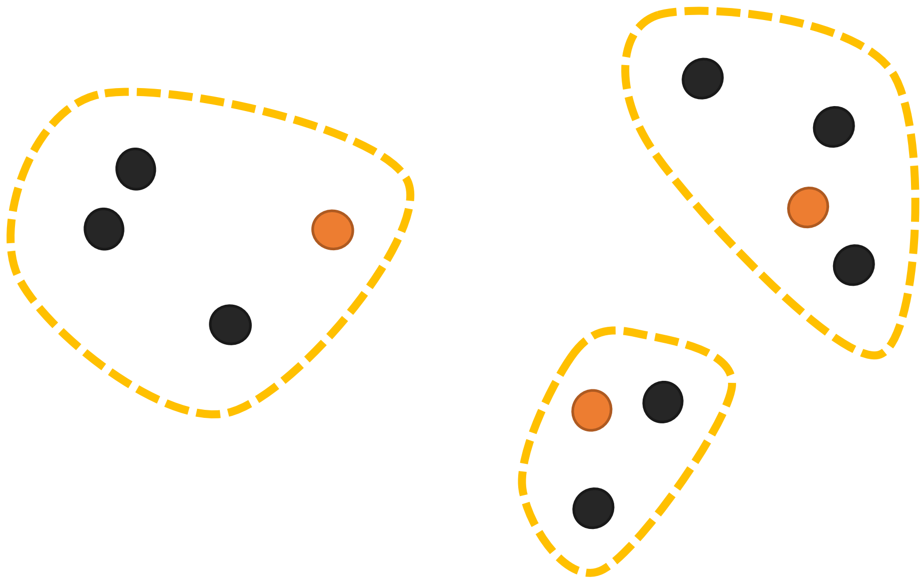
1) Initialize C with a random point
2) While there are $< k$ centers
    1) Add x to C with probability proportional to $\min_{c \in C} d(x, c)^2$



Algorithm: k-means++

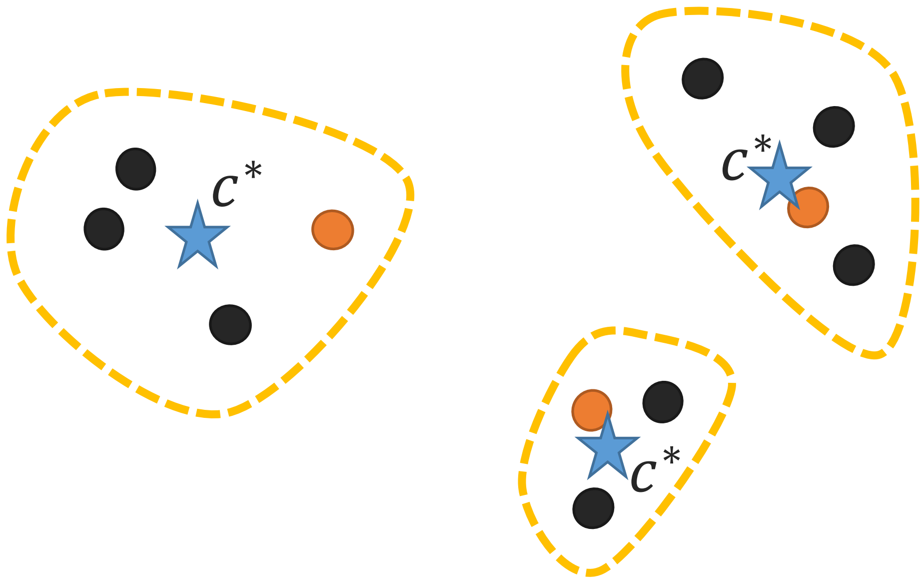
Proven to perform well under clusterability assumptions [Agarwal-Jaiswal-Pal '15]

1) Initialize C with a random point
2) While there are $< k$ centers
    1) Add x to C with probability proportional to $\min_{c \in C} d(x, c)^2$



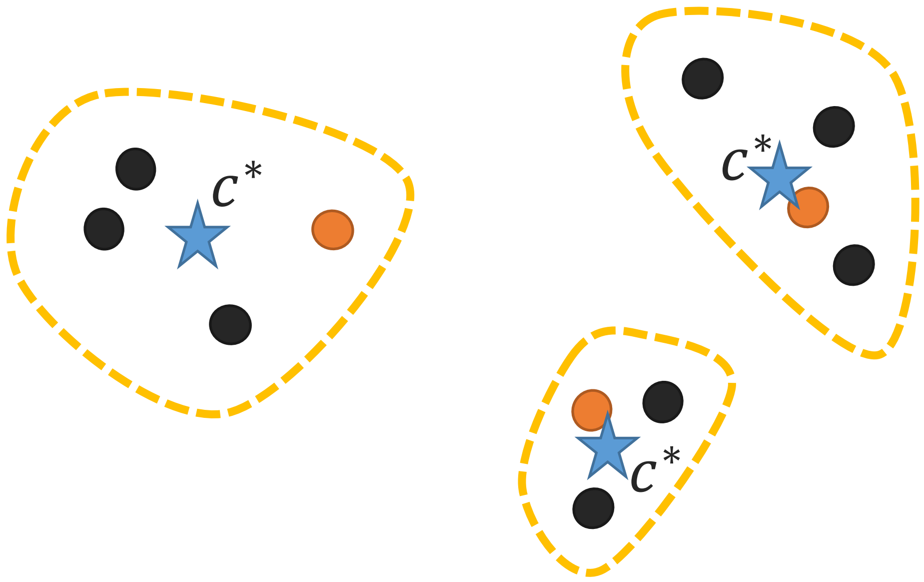
Lloyd's Algorithm (a.k.a. k-means)

1) Initialize C (with k-means++ centroids)
2) Add x to cluster of closest point in C
3) Find the "centroid" c^* of each cluster, update C'
4) Repeat until C stops changing



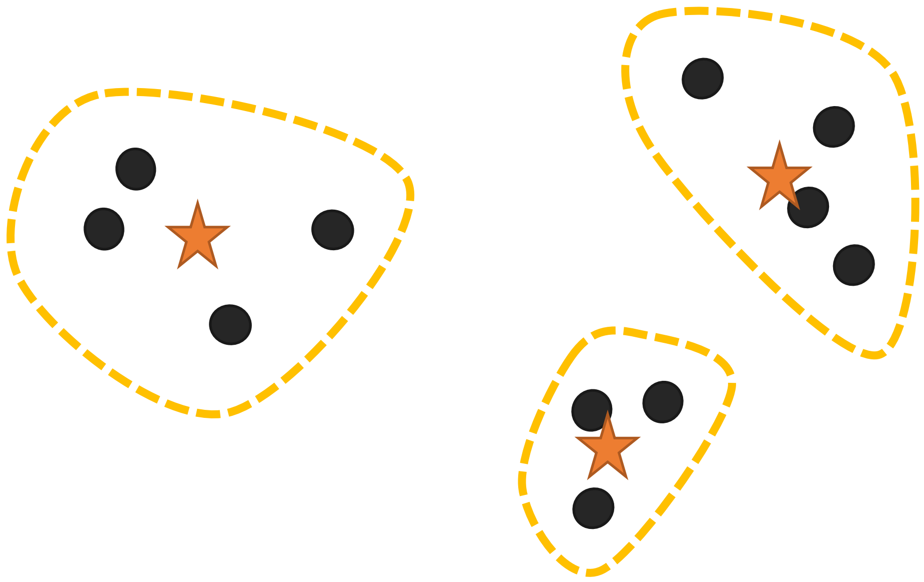
Lloyd's Algorithm (a.k.a. k-means)

1) Initialize C (with k-means++ centroids)
2) Add x to cluster of closest point in C
3) Find the "centroid" c^* of each cluster, update C'
4) Repeat until C stops changing



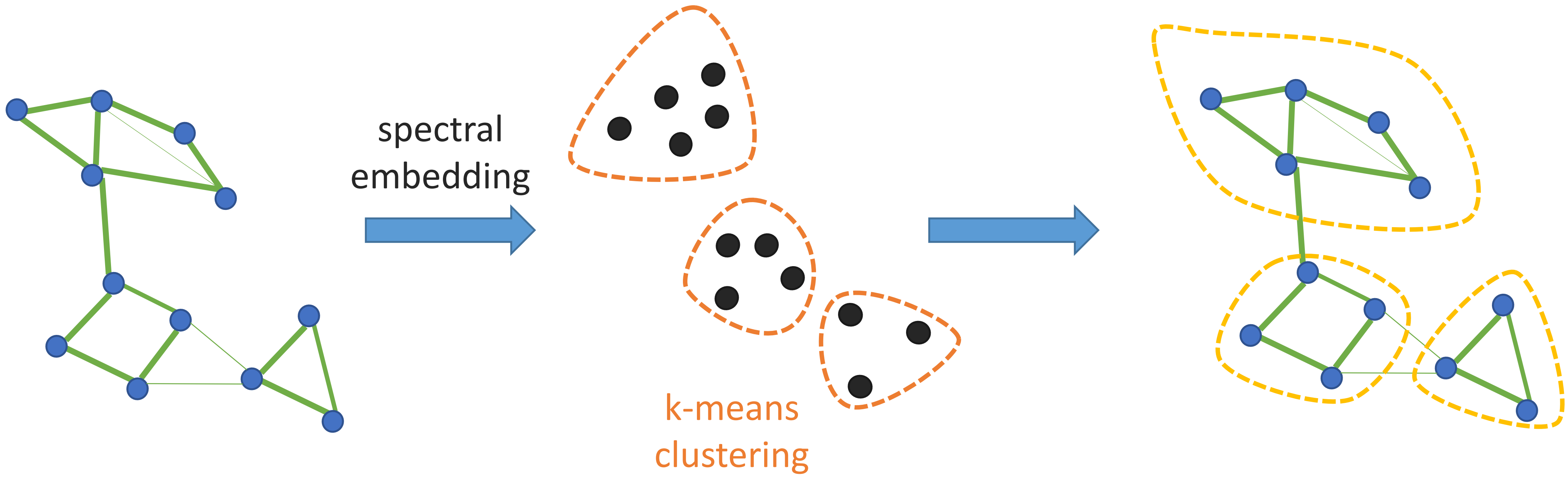
Lloyd's Algorithm (a.k.a. k-means)

1) Initialize C (with k-means++ centroids)
2) Add x to cluster of closest point in C
3) Find the "centroid" c^* of each cluster, update C'
4) Repeat until C stops changing

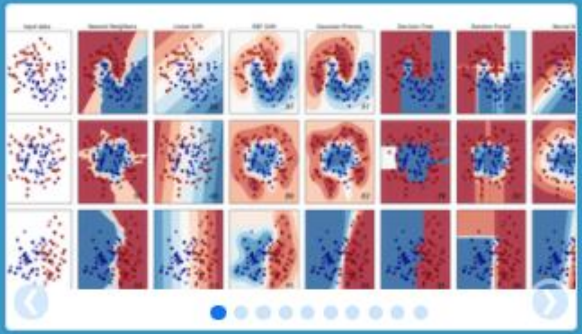


Putting things together

“Spectral Clustering”



Laplacian matrix eigenvectors



scikit-learn

Machine Learning in Python

- Simple and efficient tools for data mining and data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

Please **cite us** if you use the software.

2.3. Clustering

2.3. Clustering

Clustering of unlabeled data can be performed with the module `sklearn.cluster` .

Each clustering algorithm comes in two variants: a class, that implements the `fit` method to learn the clusters on train data, and a function, that, given train data, returns an array of integer labels corresponding to the different clusters. For the class, the labels over the training data can be found in the `labels_` attribute.

Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large `n_samples` , medium `n_clusters` with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters	Distances between points
Affinity propagation	damping, sample preference	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with `n_samples`	Many clusters, uneven cluster size, non-flat geometry	Distances between points
Spectral clustering	number of clusters	Medium `n_samples` , small `n_clusters`	Few clusters, even cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints	Distances between points
Agglomerative clustering	number of clusters, linkage type, distance	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints, non Euclidean distances	Any pairwise distance
DBSCAN	neighborhood	Very large `n_samples` ,	Non-flat geometry, uneven	Distances between

Fast and efficient spectral clustering

version 1.10 (11.3 MB) by Ingo

Perform fast and efficient spectral clustering algorithms

10 Ratings
98 Downloads
Updated 13 Sep 2012
View License

Add to Watchlist Download

Overview Functions

datasets/rainbowdash/

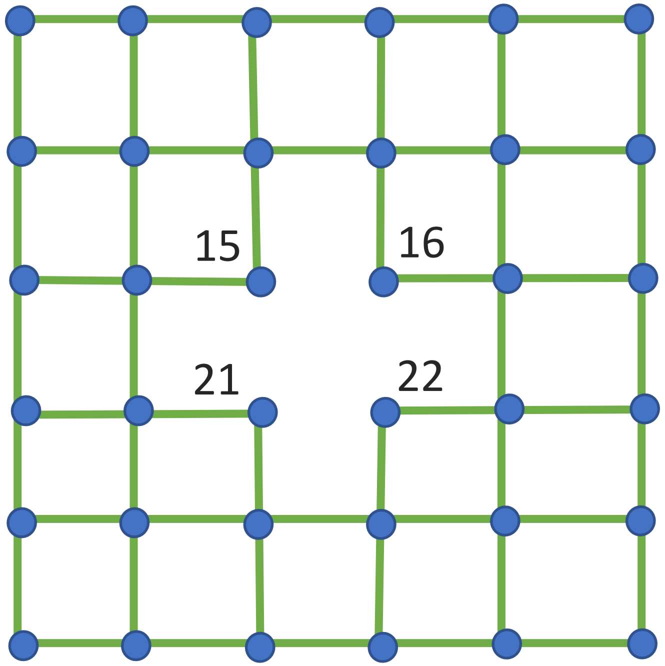
CreateDataset.m
CreateDataset2.m

files/GUI/funcs/

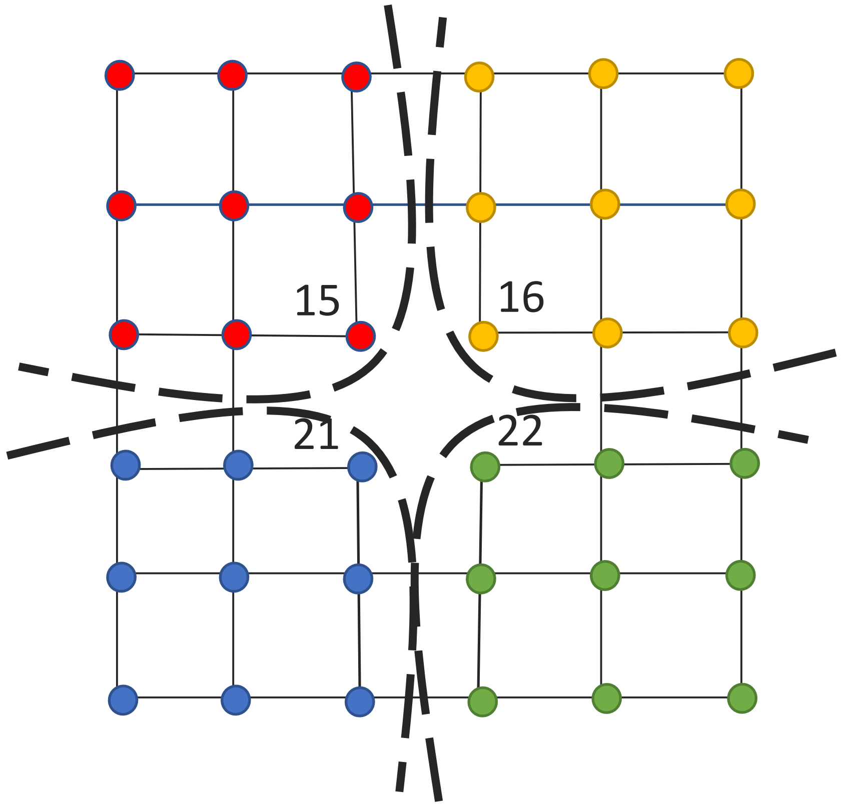
convertClusterVector(M)
normalizeData(Data)
openPlotFigure(hObject, handles,...
saveCurrentFigure(hObject, handl...
updateDataInfo(hObject, handles)

```
function [C, L, U] = SpectralClustering(W, k, Type)
%SPECTRALCLUSTERING Executes spectral clustering algorithm
%   Executes the spectral clustering algorithm defined by
%   Type on the adjacency matrix W and returns the k cluster
%   indicator vectors as columns in C.
%   If L and U are also called, the (normalized) Laplacian and
%   eigenvectors will also be returned.
%
%   'W' - Adjacency matrix, needs to be square
%   'k' - Number of clusters to look for
%   'Type' - Defines the type of spectral clustering algorithm
%            that should be used. Choices are:
%      1 - Unnormalized
%      2 - Normalized according to Shi and Malik (2000)
%      3 - Normalized according to Jordan and Weiss (2002)
%
%   References:
```

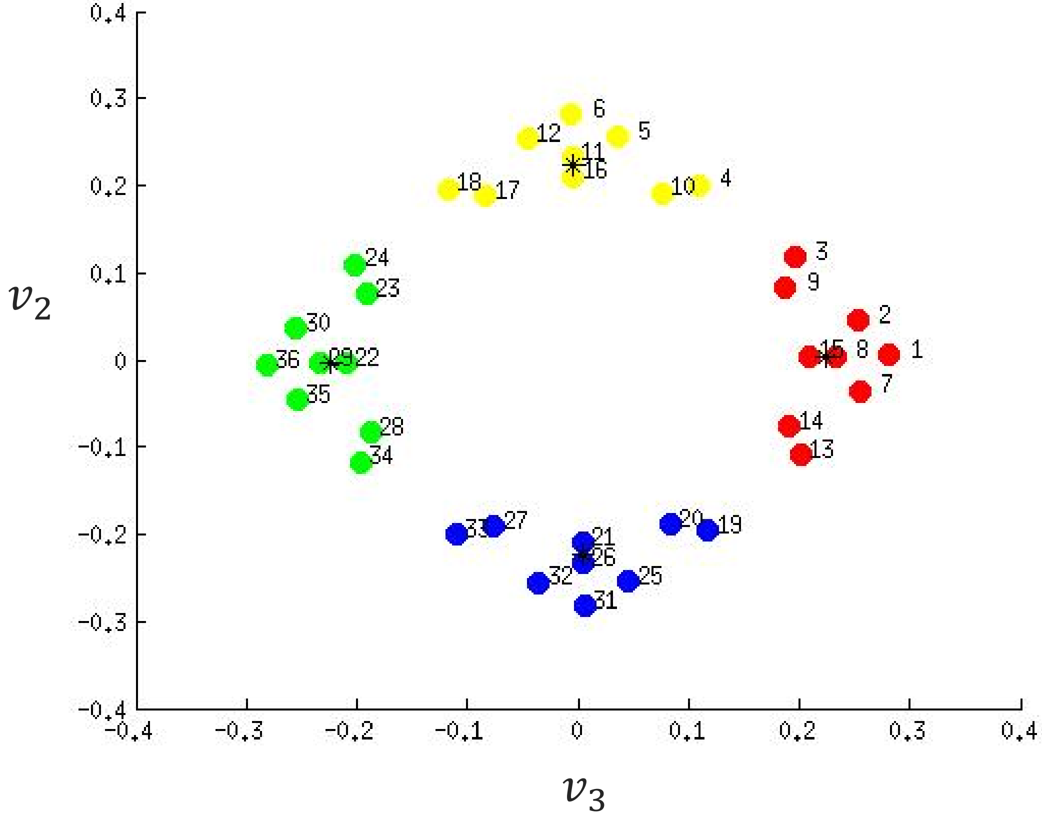
Spectral & k-means++ in action



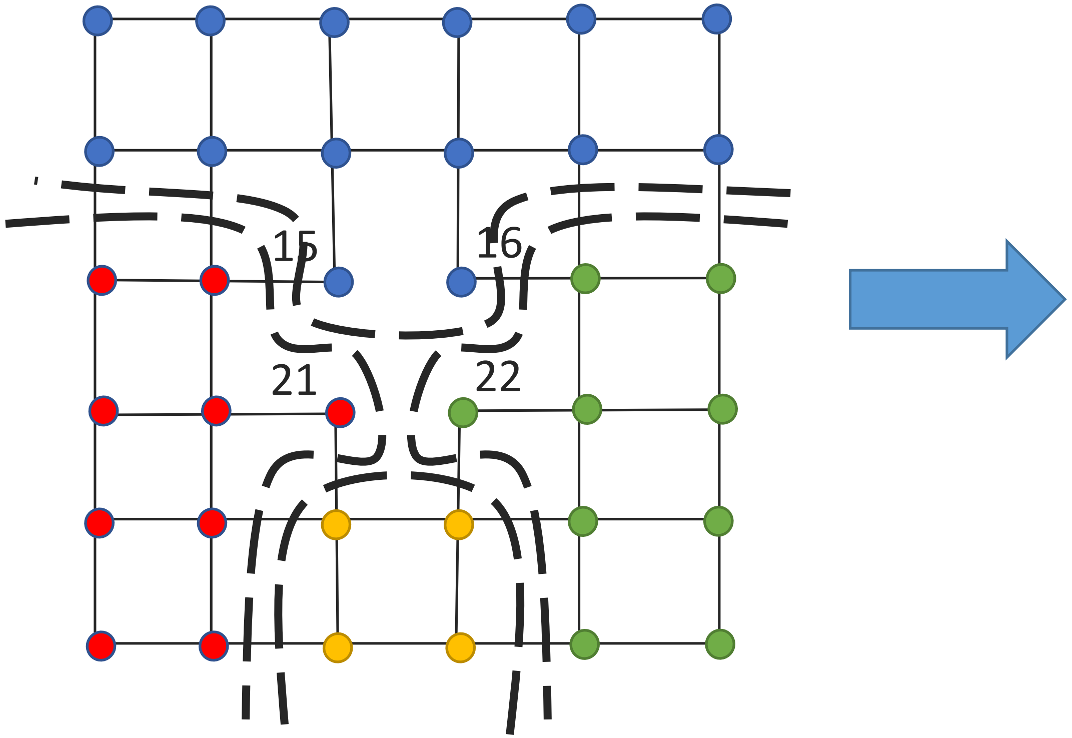
Spectral & k-means++ in action

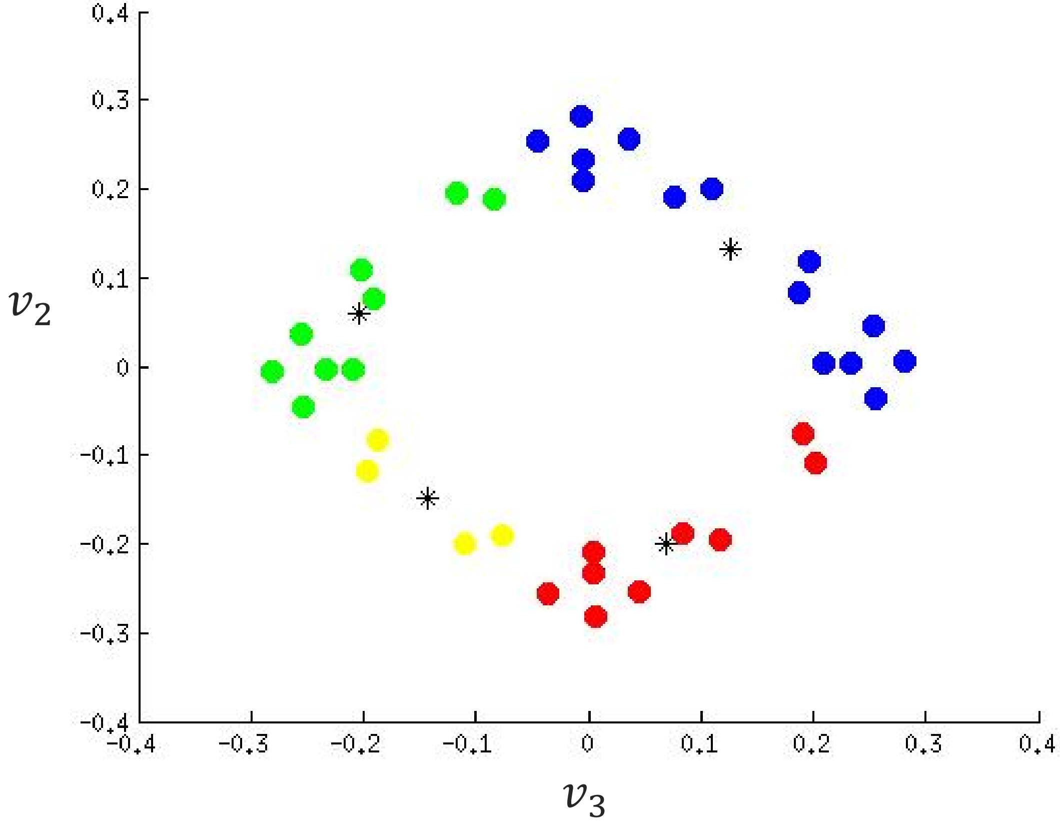


2D spectral embedding + 4-means

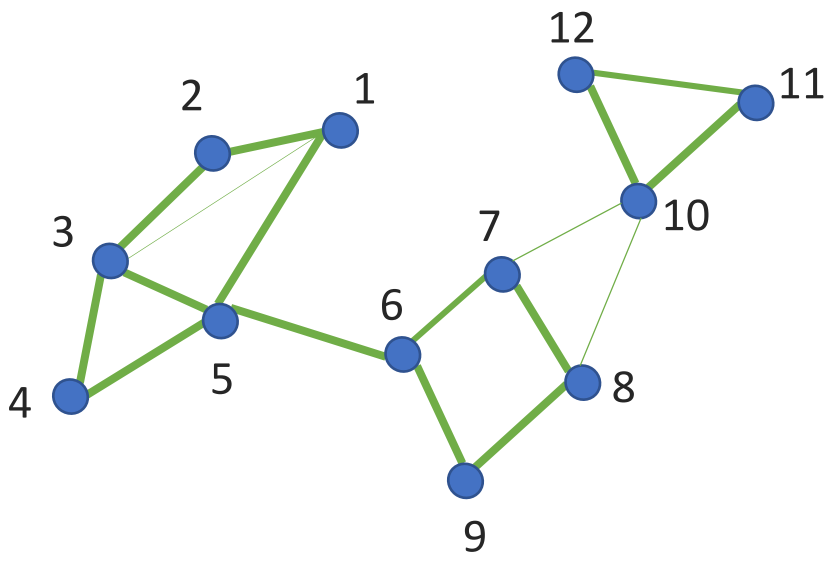


Spectral & k-means++ in action

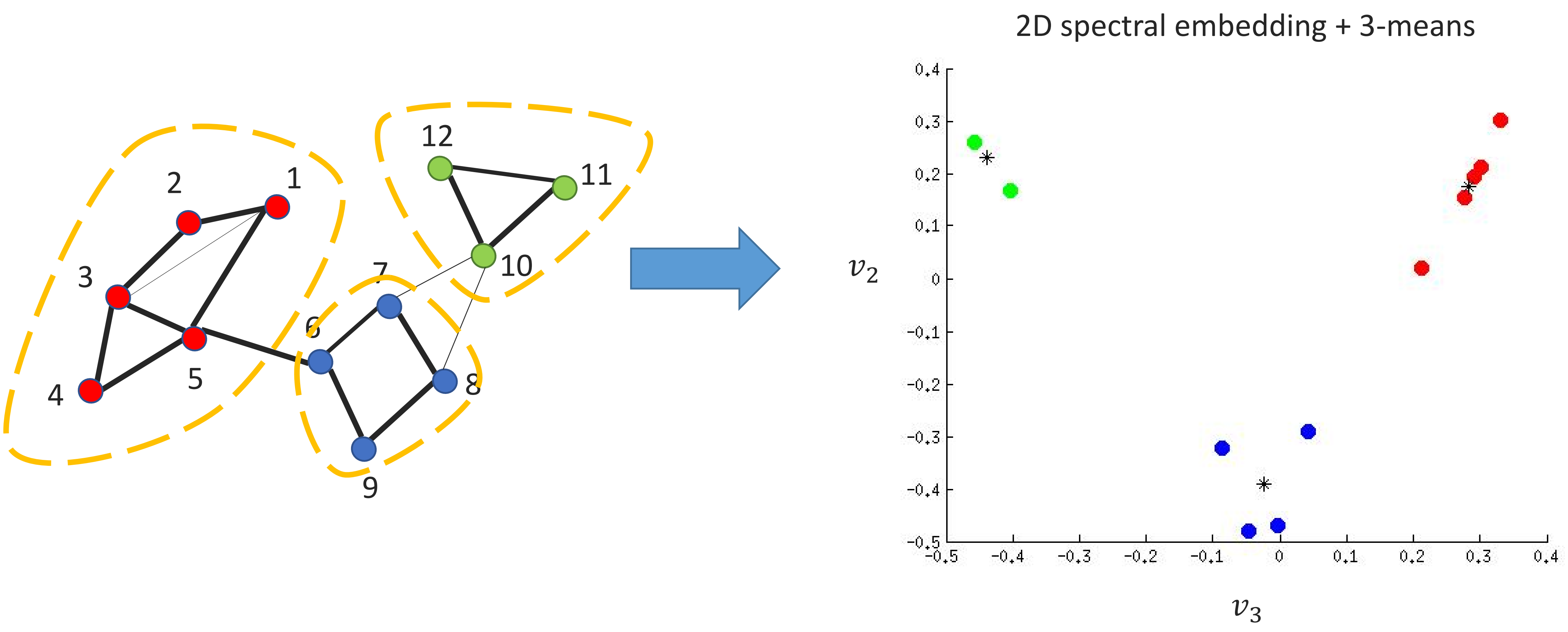




Spectral & k-means++ in action



Spectral & k-means++ in action



Concluding

- Graph clustering is ubiquitous
- Important to formulate correct objective
  Theory vs. Practice
- The Spectral Embedding
  Is awesome! Stay tuned…
- Practice $\Leftrightarrow$ Theory
  Case study: k-means

Thanks!