Entering the Information Age

John Hopcroft
Cornell University
Ithaca, New York

Science is changing

A time of change is a time of opportunity.

Those individuals, institutions, and countries who position themselves for the future will benefit immensely.

Science in the 21 Century

Advances in Information and Biology are changing our lives.

I will focus on some of the exciting new research directions in computer science.

Some exciting research directions

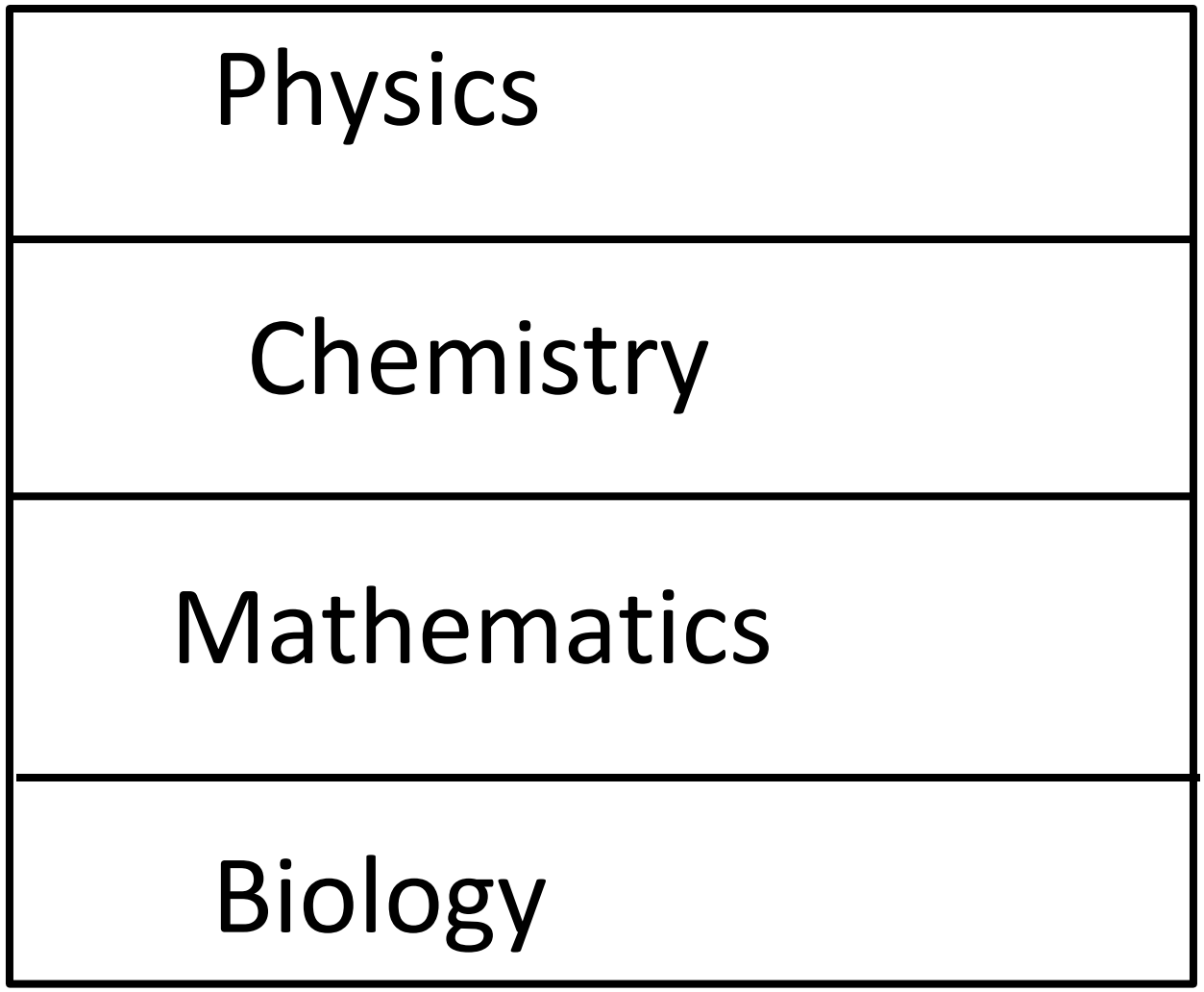
- Social networks and hidden structure
- Learning theory and unsupervised learning
- Deep learning
- Spectral clustering
- MCMC
- High dimensional data

And many more directions

Social networks

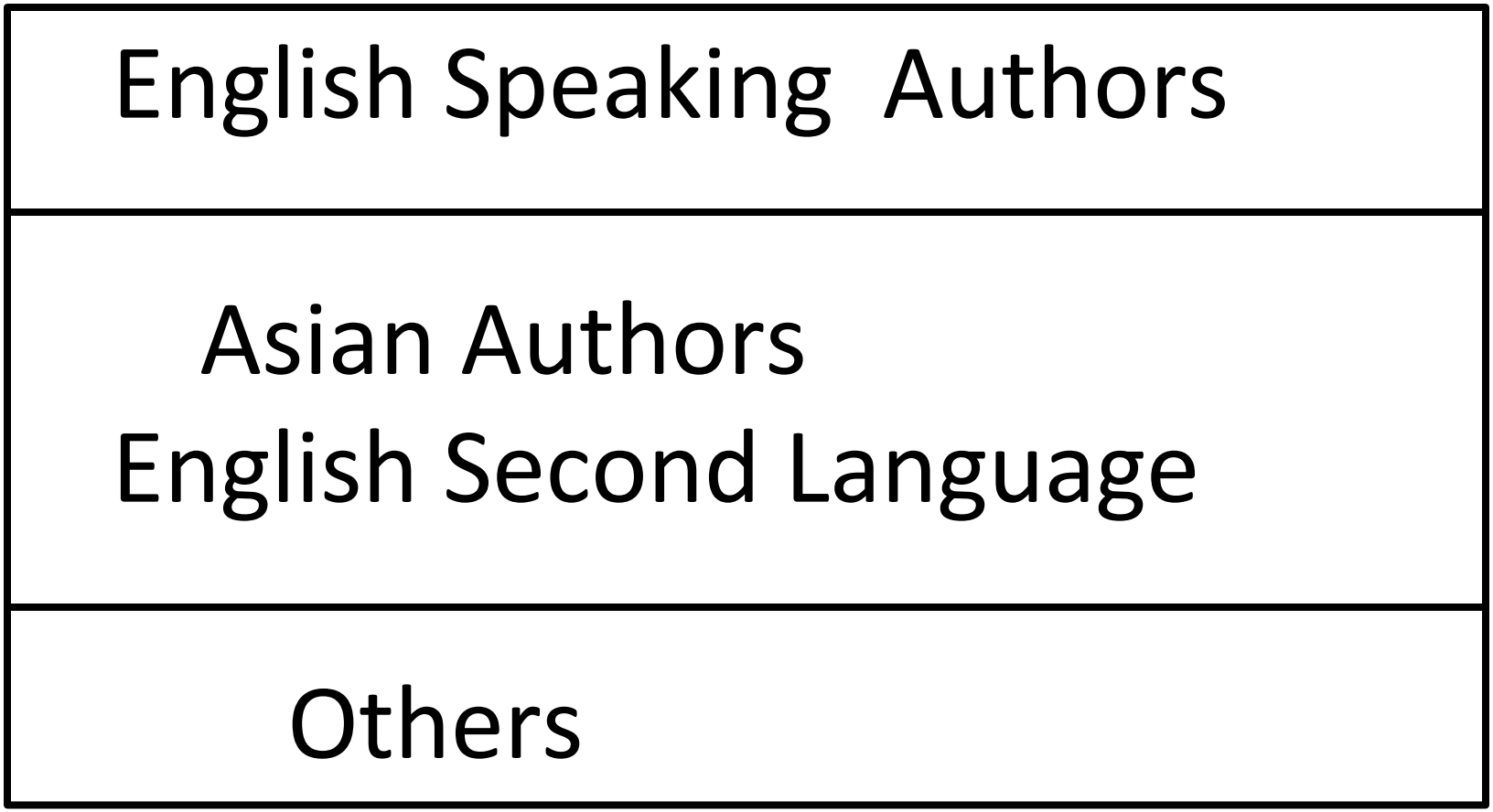
- People Facebook
- Papers Physics archive
- Genes Biological
- Purchases Amazon
- Movies Netflix

Structure in a network

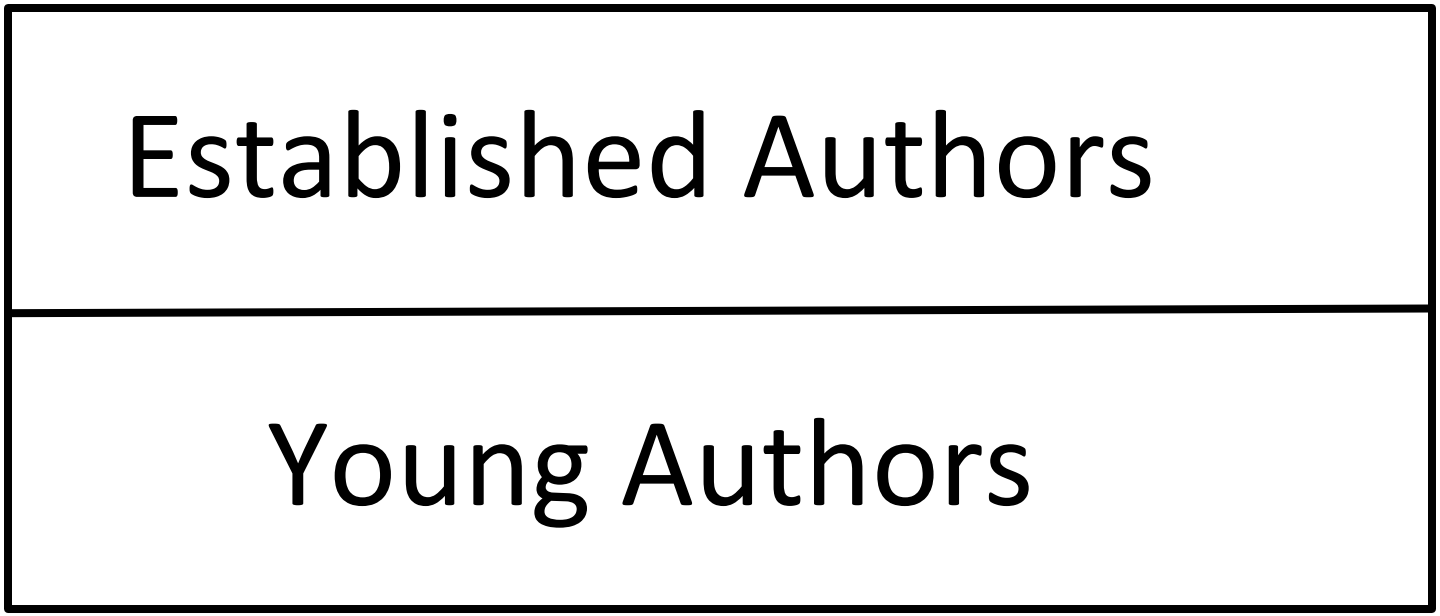


Communities in a network where vertices are papers and edges represent similar word vectors.

A secondary structure in the network



A third structure in the network

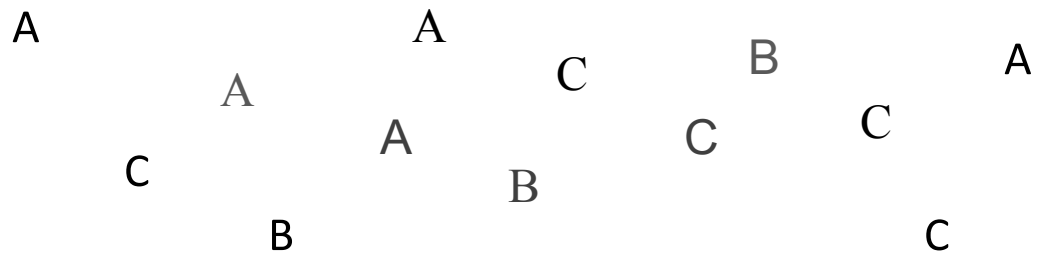


Still another possibly hidden structure

Another example of hidden structure

- Consider handwritten characters.
- Cluster the characters.
- Do you want each letter to be in its own cluster?
- Or do you want the clusters to be characters written by the same person?

How would one cluster the letters?



- By letter

{A A A A A} {B B B} {C C C C C}

- By color

{A A A B C C C C} {A A B B C}

- By type font

{A A B C C} {A A B B C} {A B C}

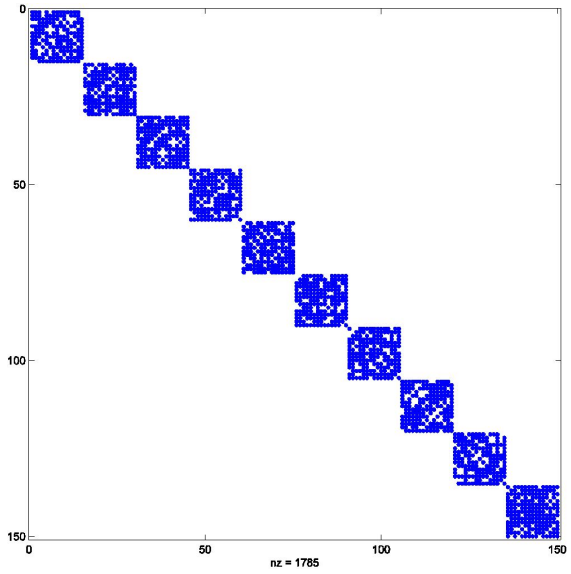
Discovering hidden structures in social networks

K-means works better on synthetic data than real data. Why?

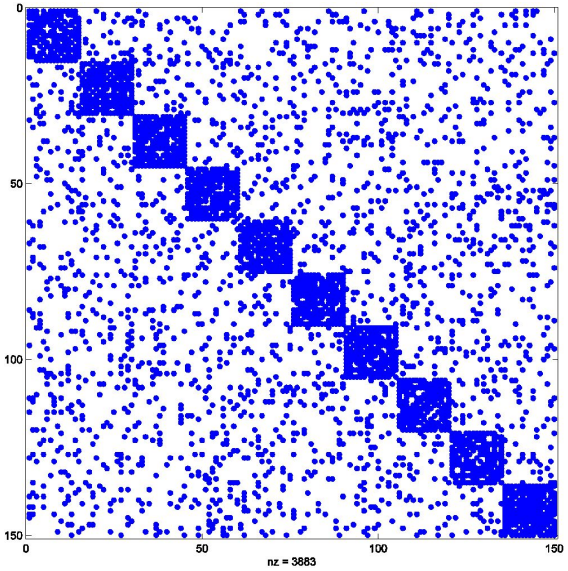
Synthetic networks have uncorrelated random noise.

In real networks what appears to be uncorrelated random noise is really structured noise often due to hidden structure.

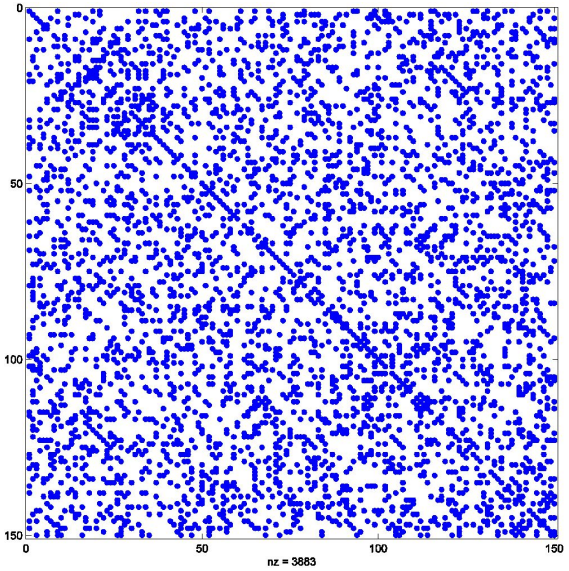
One structure with random noise



One structure

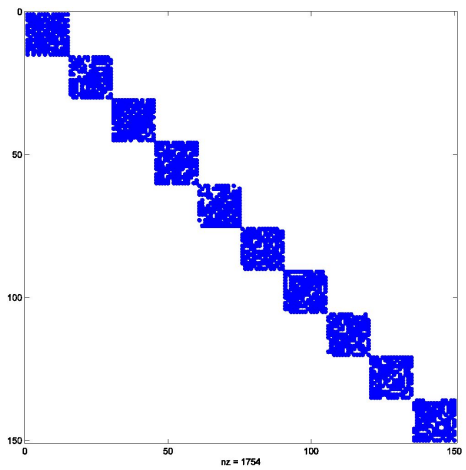


Add random noise

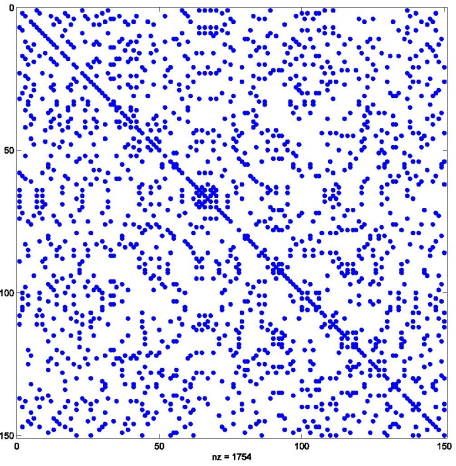


After random permutation

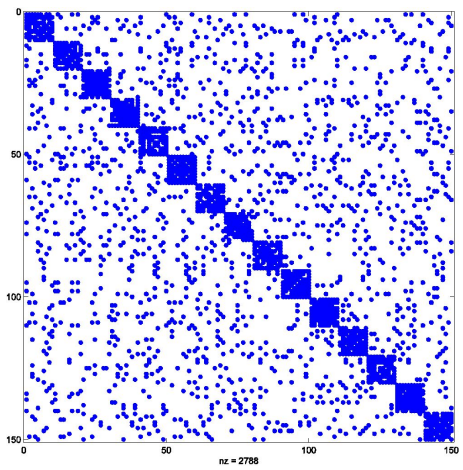
Two structures in a graph



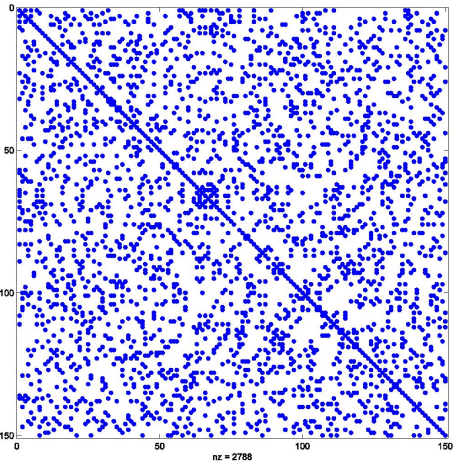
Dominant structure



Randomly permute

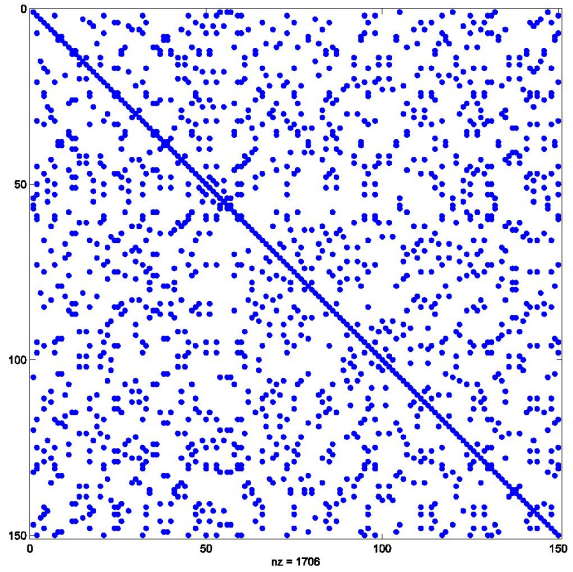


Add hidden structure

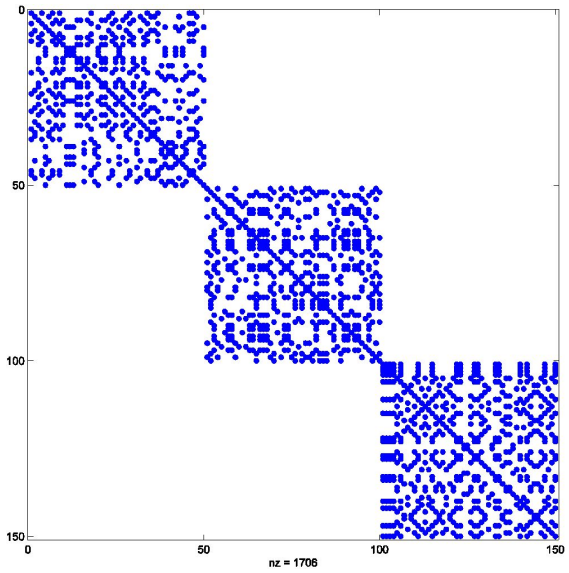


Randomly permute

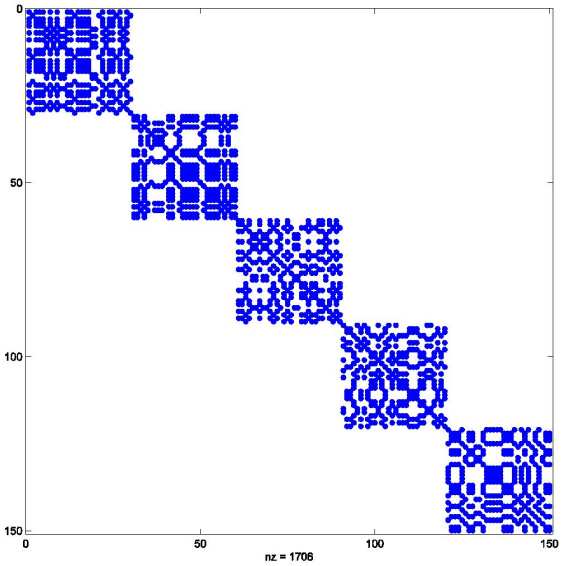
Another type of hidden structure



Randomly permuted



Permuted by
dominant structure



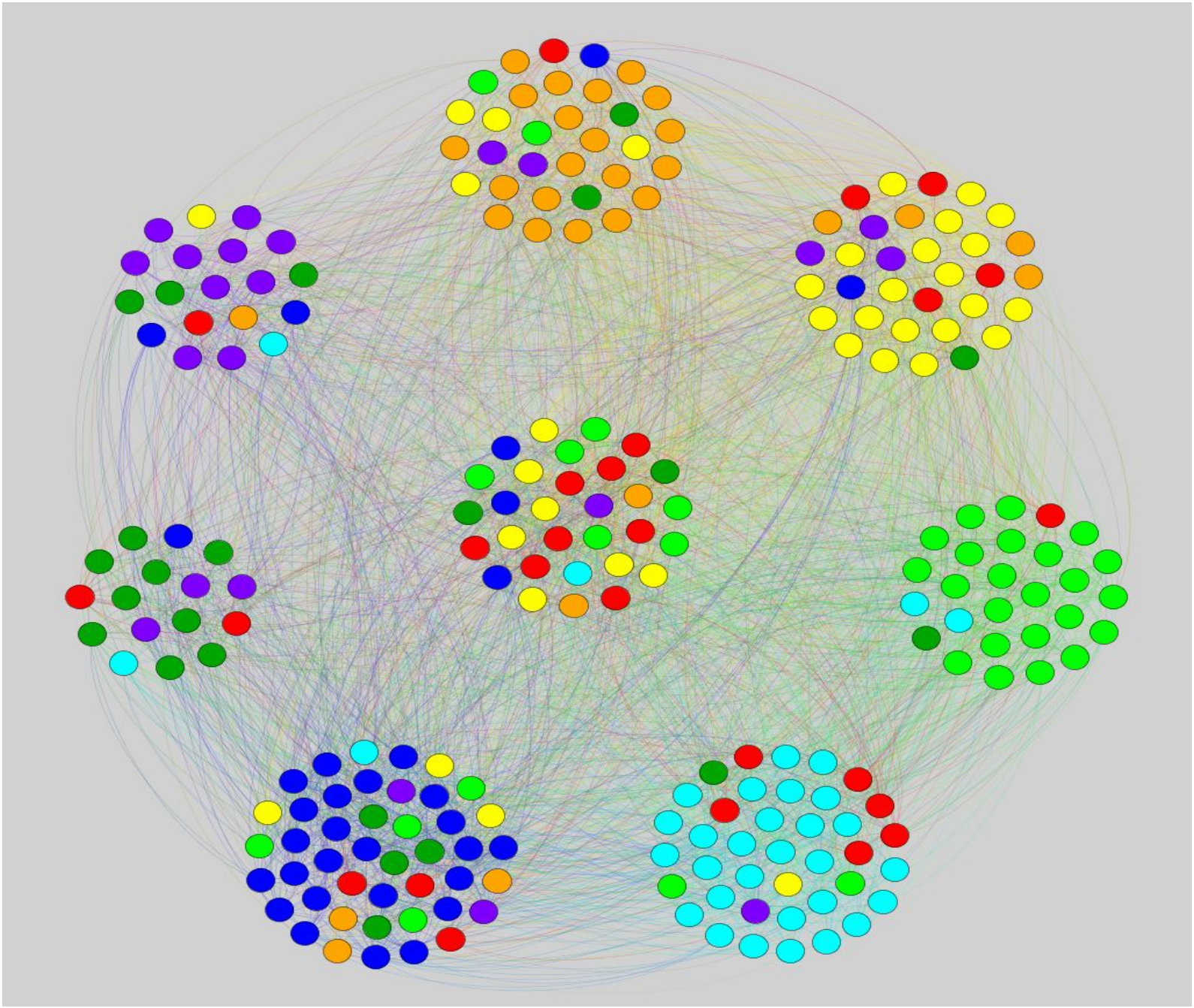
Permuted by
hidden structure

Finding hidden structure

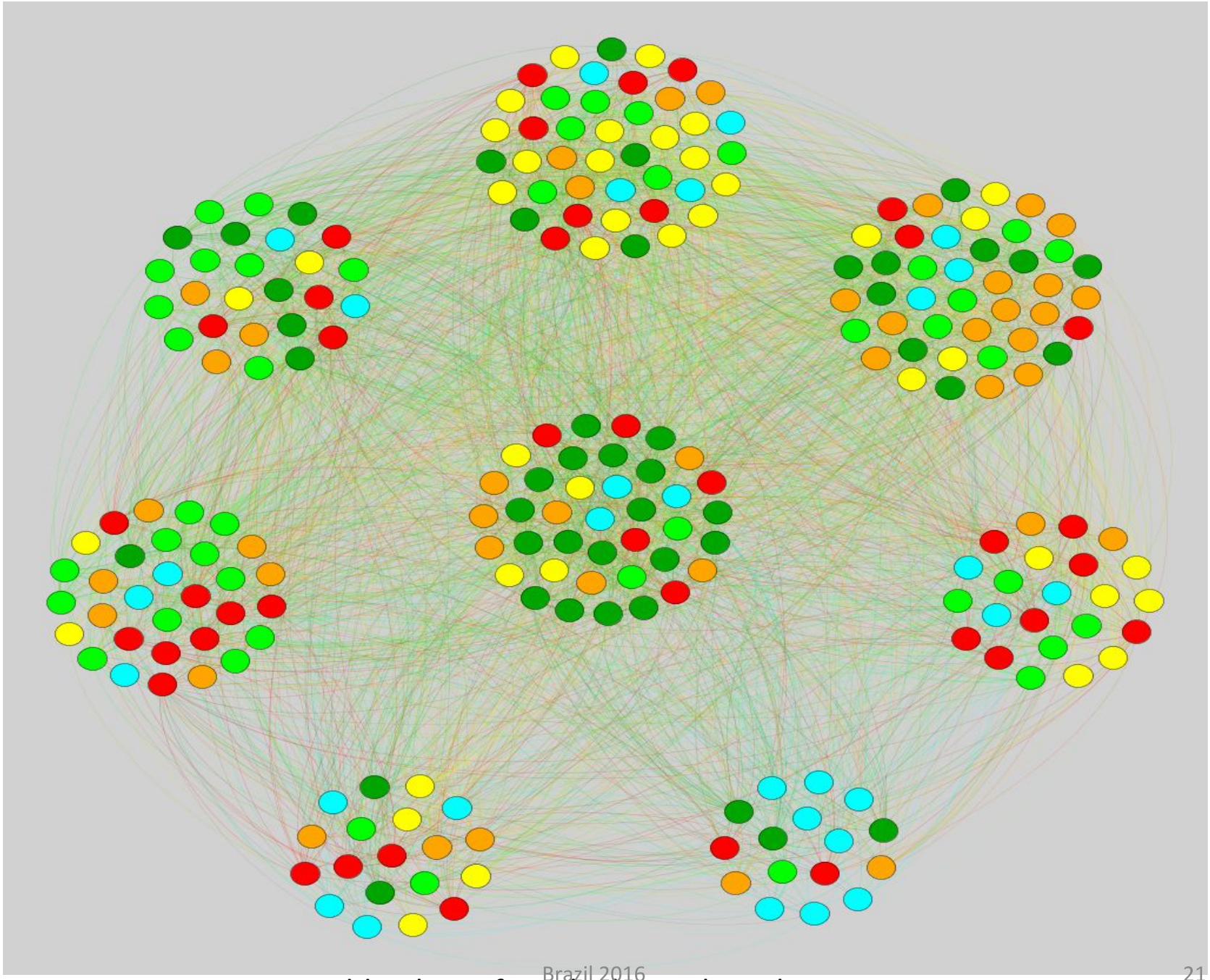
- Created a framework for weakening dominant structure.
- Can be used with your favorite clustering algorithm.
- Allows weaker structures to be found.
- Some real data sets have as many as seven levels of hidden structure.
- Each level has high modularity and is incoherent with earlier levels.

Improving existing clustering algorithms

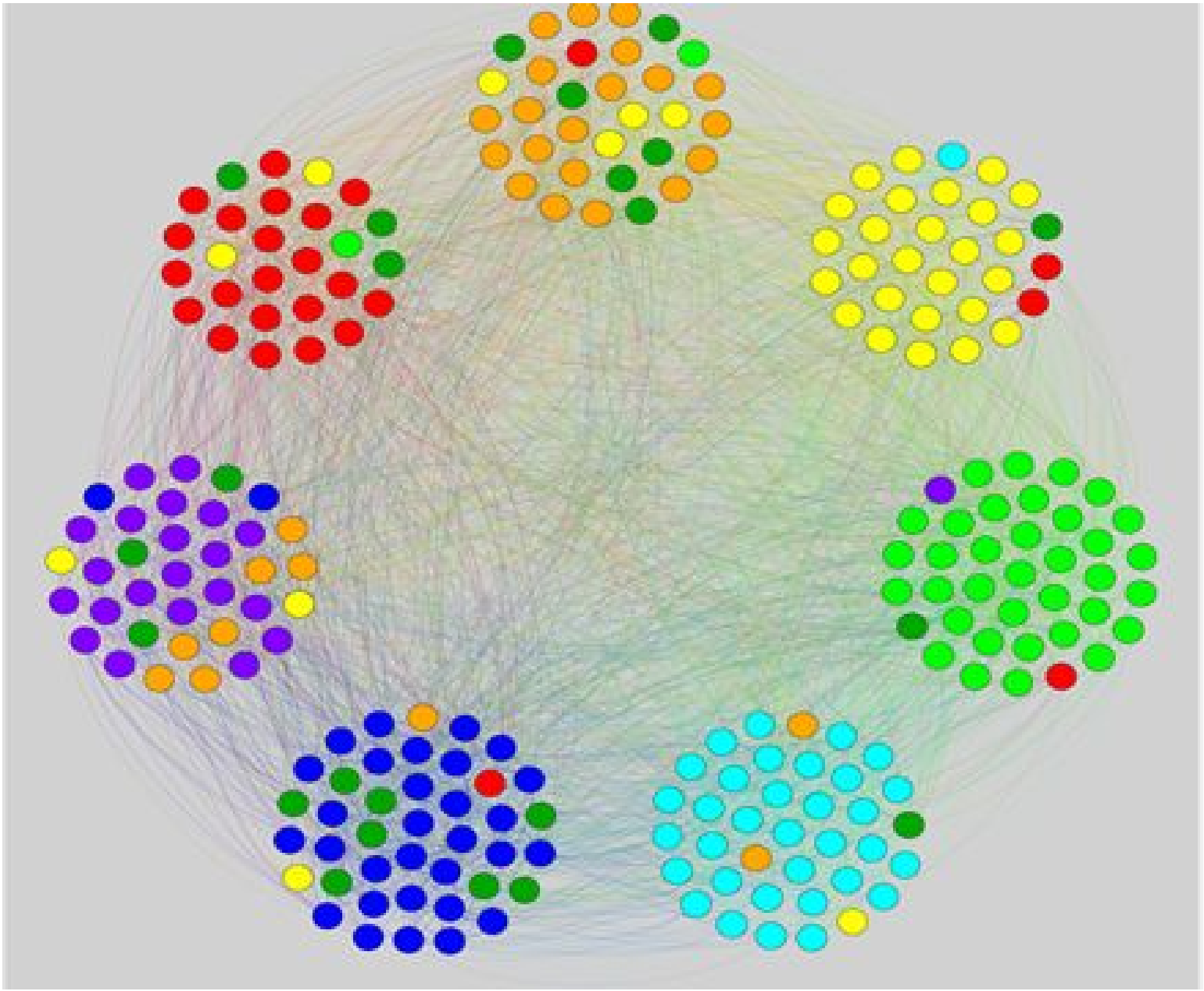
- After weakening the dominant structure to find the hidden structure, weaken the hidden structure.
- This improves most of the common clustering algorithms.
- Can refine the levels of clustering so that they correspond more fully to a single type of relationship.



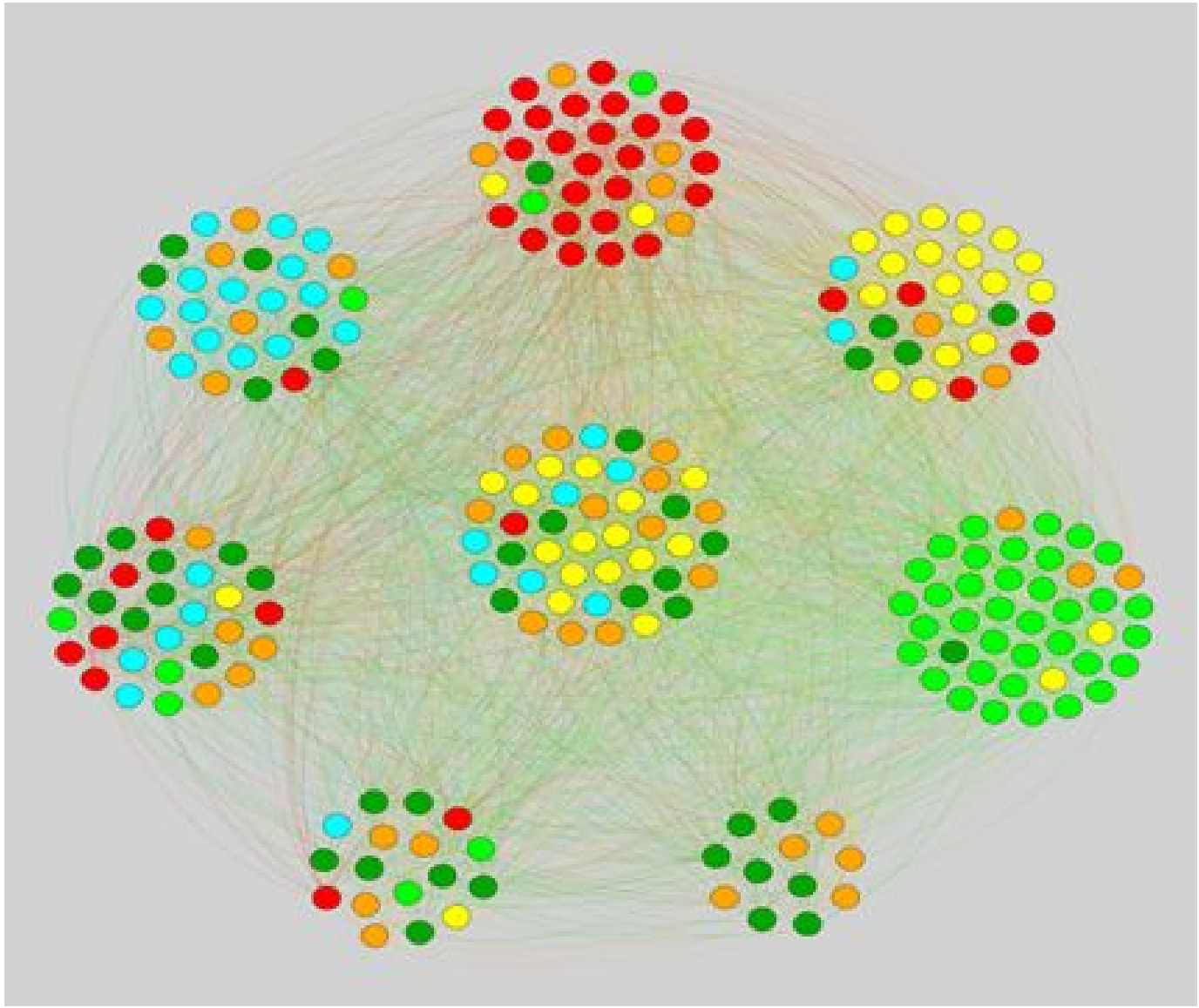
Dominate layer found by base algorithm



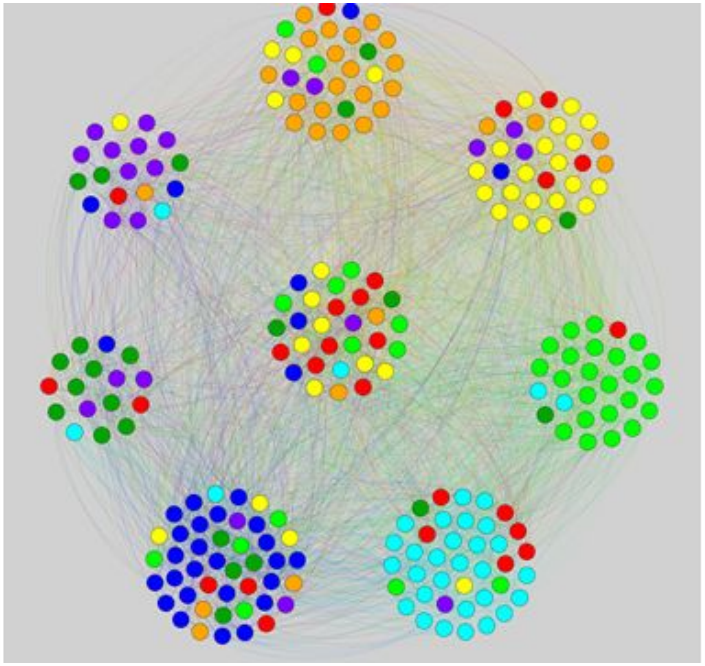
Hidden layer from by base algorithm

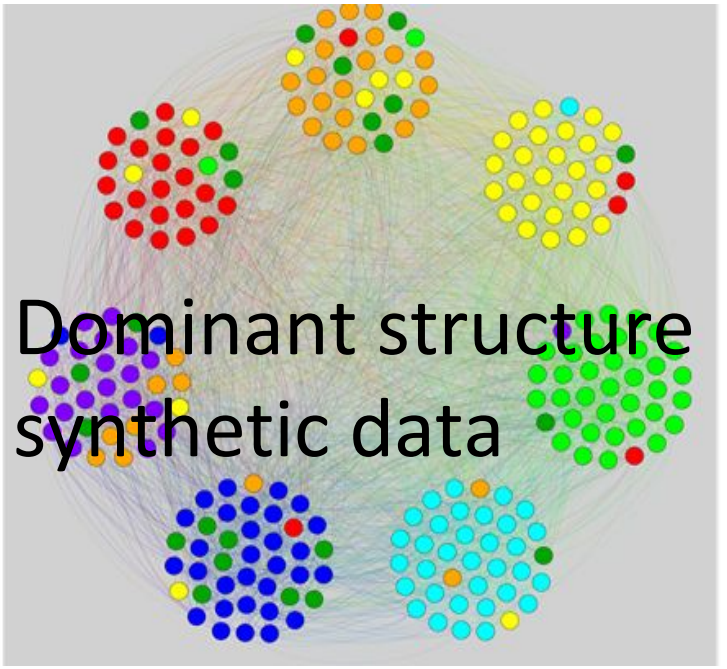


Dominate layer after weakening hidden layer

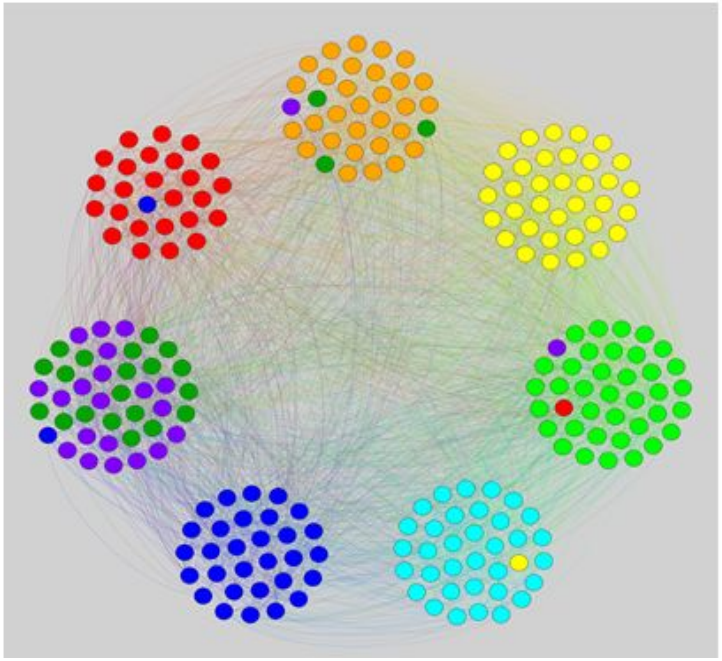


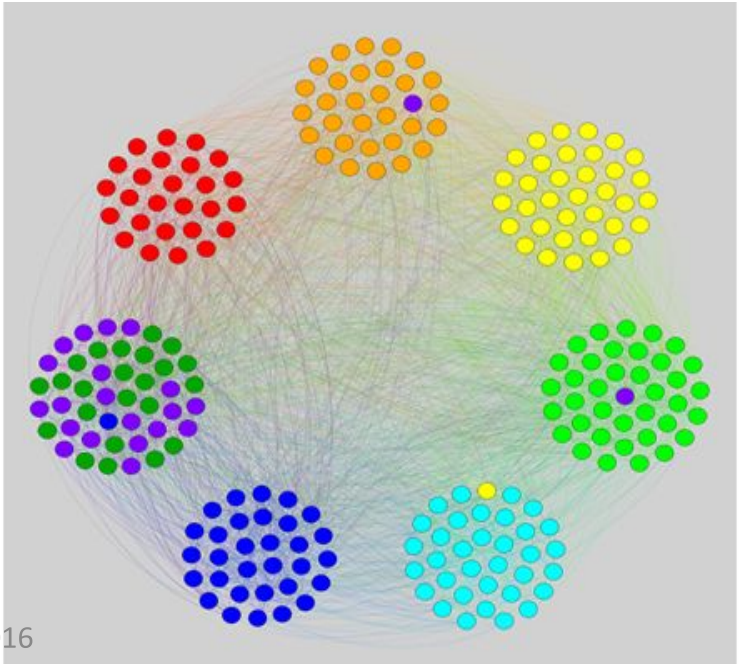
Hidden layer after weakening dominant layer

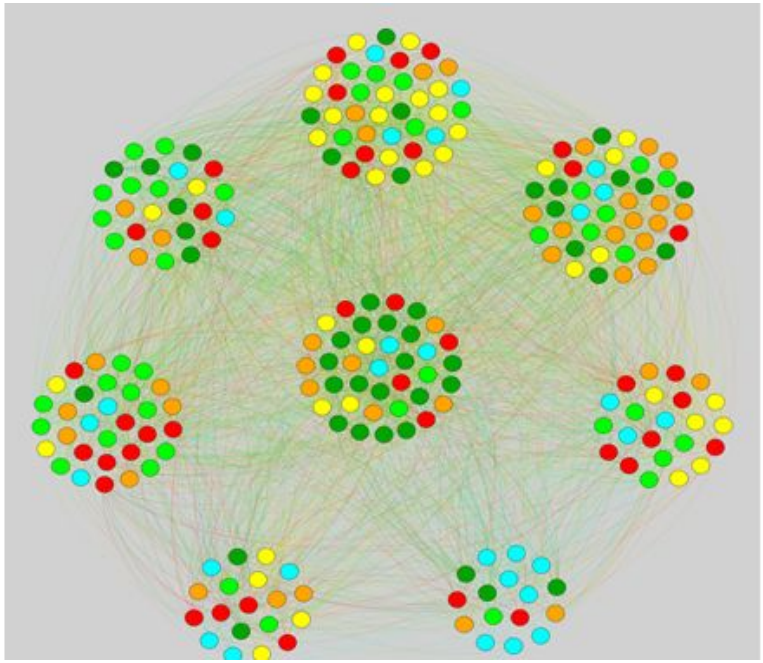


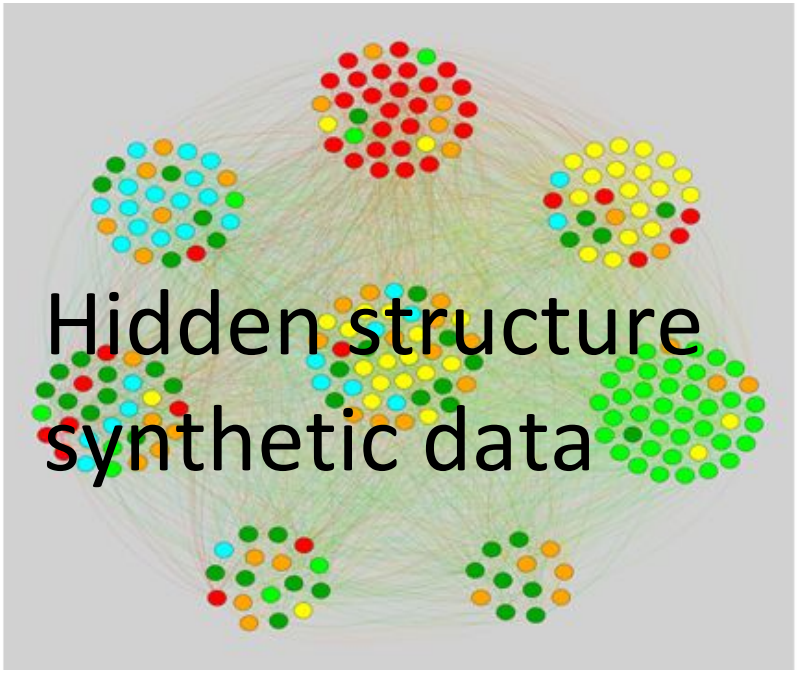


Dominant structure synthetic data

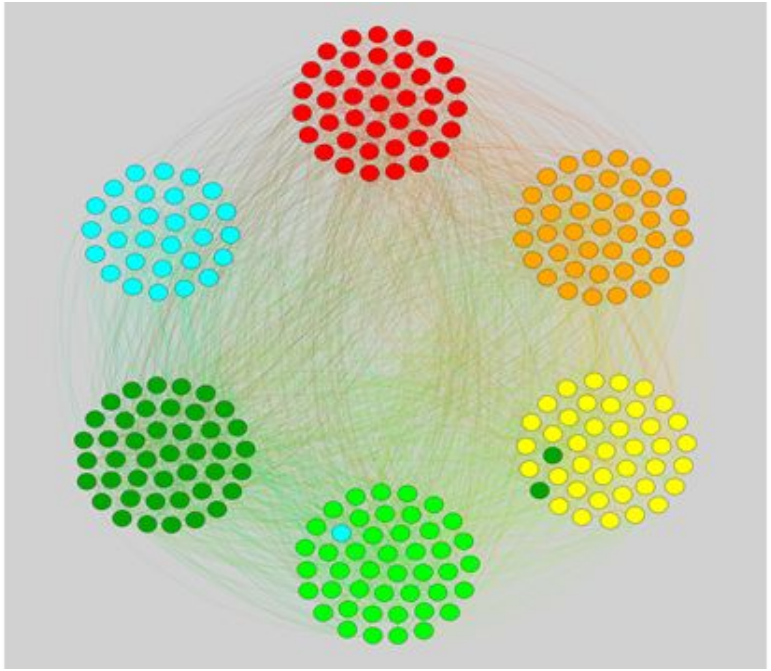


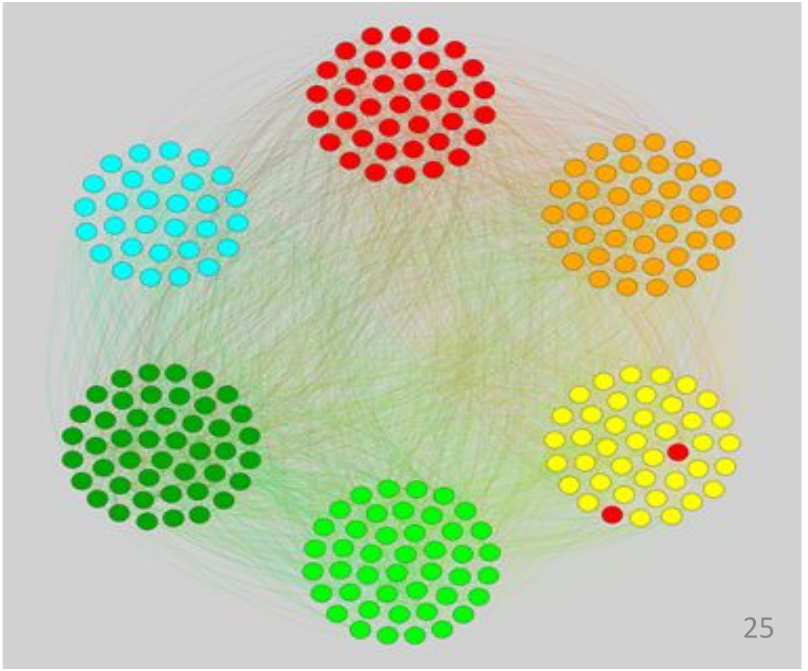






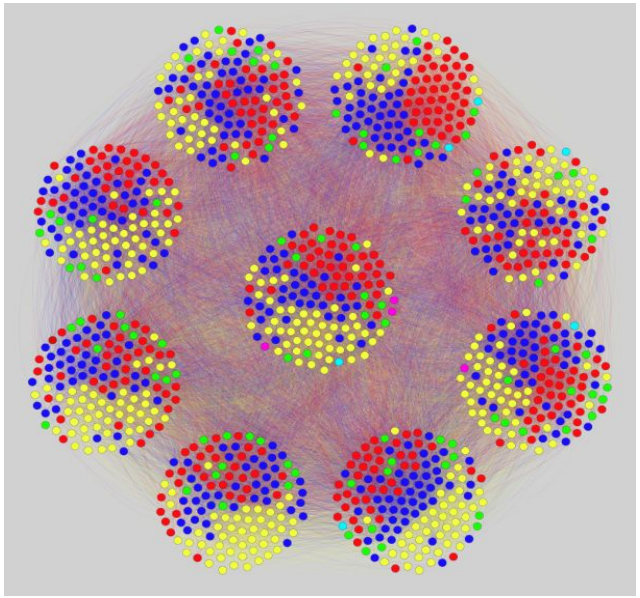
Hidden structure
synthetic data

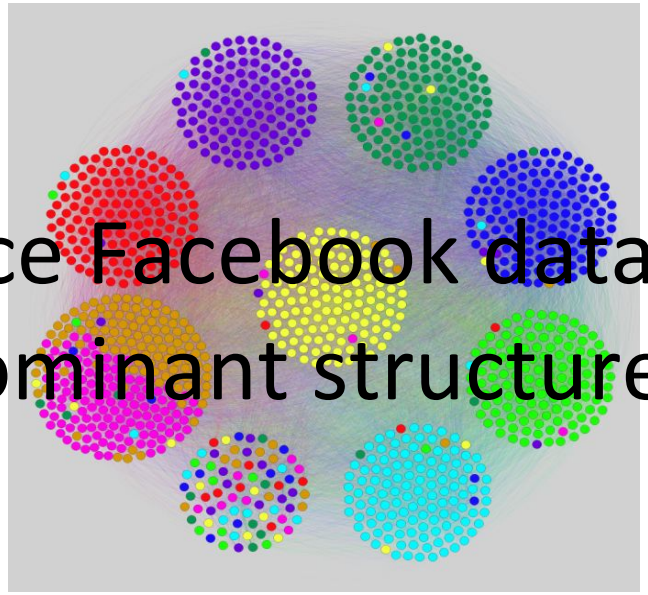




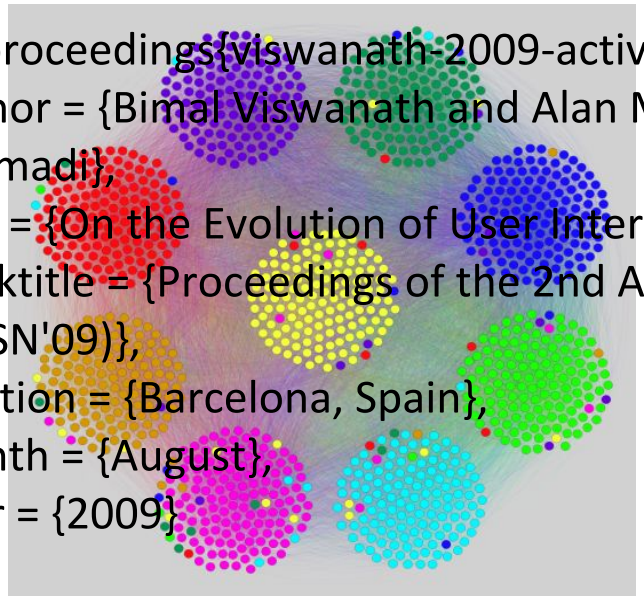
Rice Facebook data

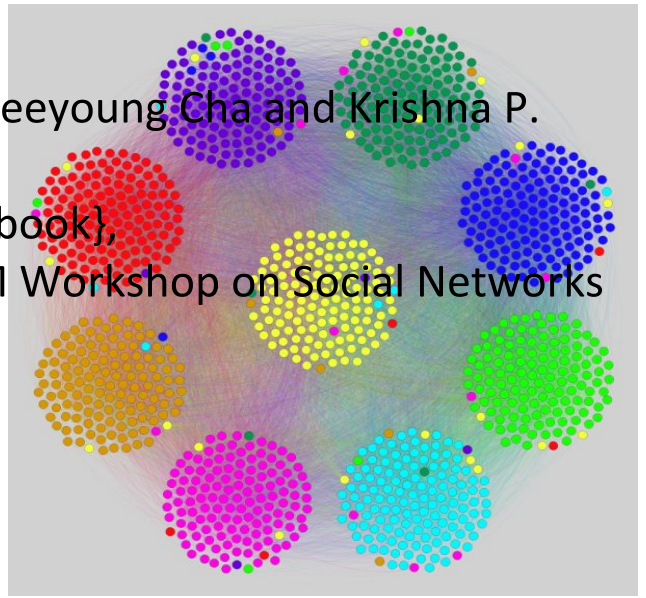
```
@inproceedings{viswanath-2009-activity,
  author = {Bimal Viswanath and Alan Mislove and Meeyoung Cha and Krishna P. Gummadi},
  title = {On the Evolution of User Interaction in Facebook},
  booktitle = {Proceedings of the 2nd ACM SIGCOMM Workshop on Social Networks (WOSN'09)},
  location = {Barcelona, Spain},
  month = {August},
  year = {2009}
}
```



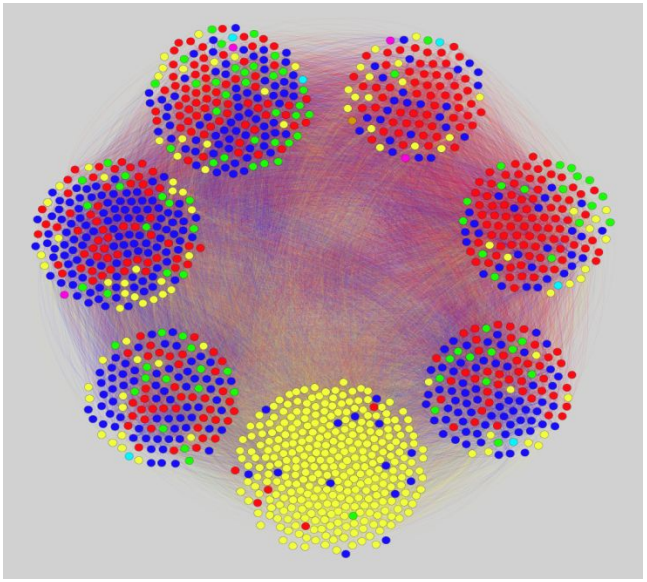


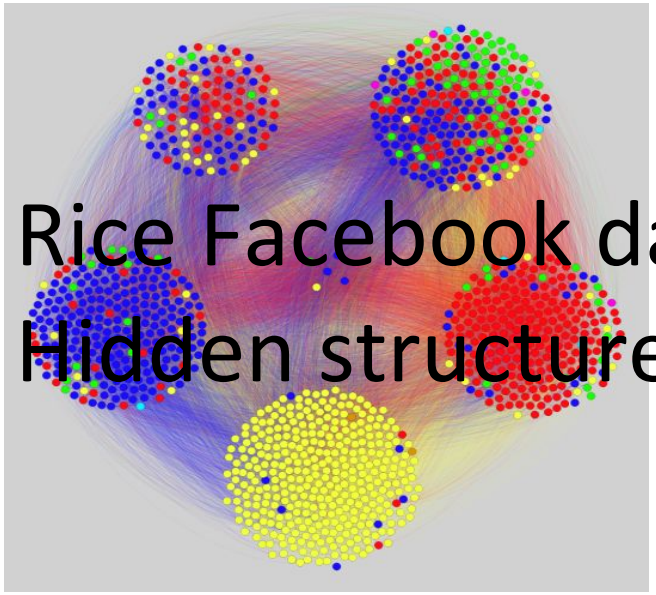
Rice Facebook data
Dominant structure



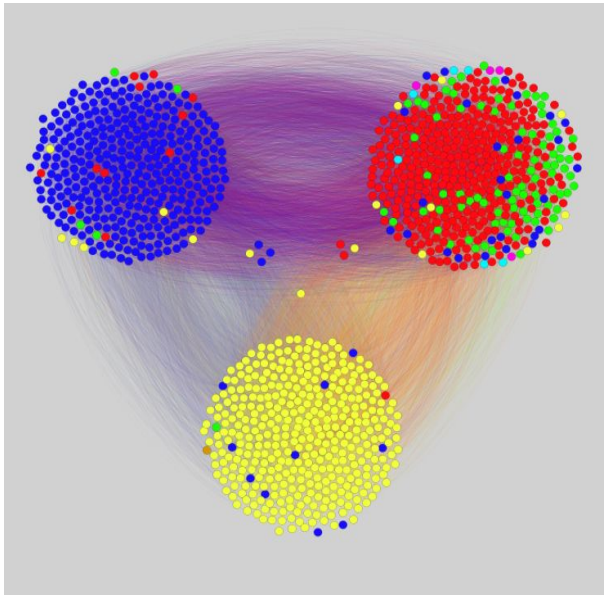


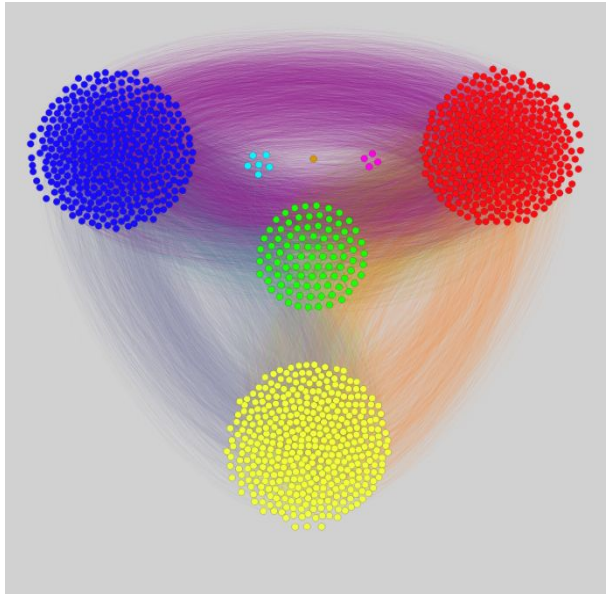
@inproceedings{viswanath-2009-activity,
  author = {Bimal Viswanath and Alan Mislove and Meeyoung Cha and Krishna P. Gummadi},
  title = {On the Evolution of User Interaction in Facebook},
  booktitle = {Proceedings of the 2nd ACM SIGCOMM Workshop on Social Networks (WOSN'09)},
  location = {Barcelona, Spain},
  month = {August},
  year = {2009}
}





Rice Facebook data
Hidden structure

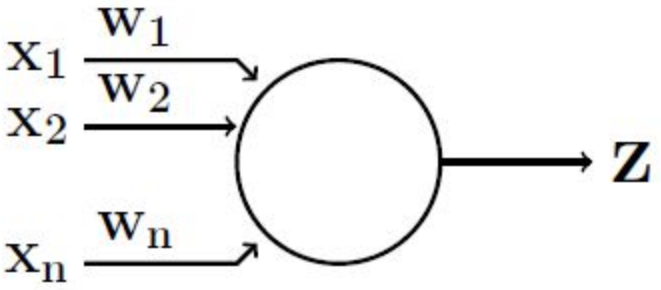




Actually we can find four layers in the Facebook data.

In some networks we were able to extract as many as seven layers.

Machine learning



$$z = \begin{cases} 0 & \sum x_i w_i \geq T \\ 1 & \sum x_i w_i \leq T \end{cases}$$

data $a_1, a_2, \ldots, a_n$ $\qquad\qquad$ $l_i = \pm 1$

Algorithm

Set $w = a_1$

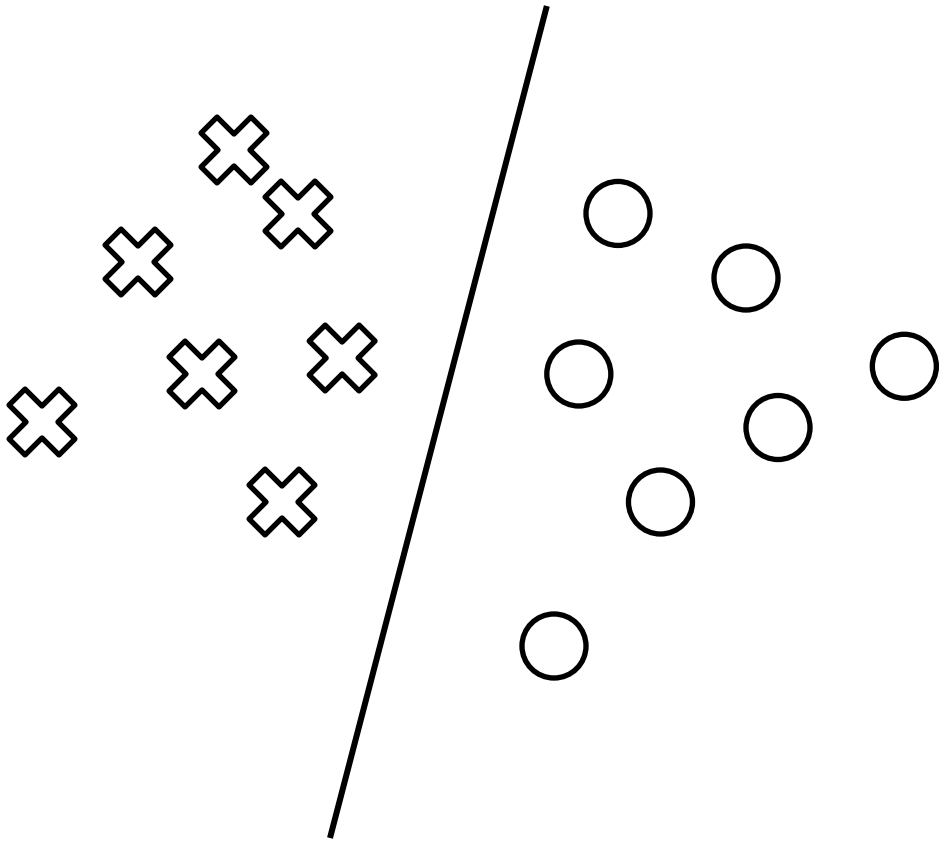
Until all patterns classified correctly

If a_i **not classified correctly** $w = w + a_i l_i$

End

Note: Weight vector is a linear combination of the patterns.

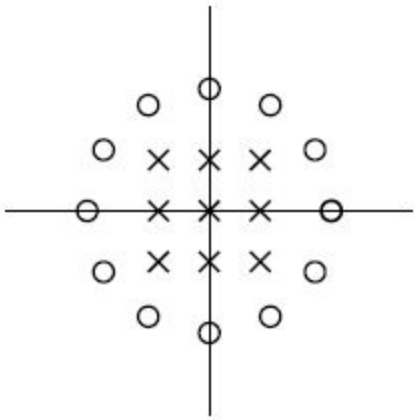
Linearly separated data



Map problem to higher dimensional space where the data is linearly separable.

What if data is not linearly separable?

Solve problem in higher dimensional space



$$\longrightarrow \quad z = x^2 + y^2$$

$$\mathbf{a_i} \xrightarrow{\mathrm{f}} \mathbf{f(a_i)}$$

Mapping may be to an infinite dimensional space.

Do not need to compute images of patterns in the higher dimensional space.

Only need the products of the images of the patterns.

$$\mathbf{a_i} \xrightarrow{\mathbf{f}} \mathbf{f(a_i)}$$

$$\mathbf{w} = \sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{c_i f(a_i)}$$

$$\mathbf{w}\ \mathbf{f(a_j)} = \sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{c_i}\ \underbrace{f(a_i) f(a_j)}$$

$$\mathbf{w} \longleftarrow \mathbf{w} + \mathbf{f(a_j)}$$

If I know the products of mappings of images, I do not need to know the mappings of images.

Just increase the coefficient $\mathbf{c_j}$ of $\mathbf{f(a_j)}$.

Kernels

$$K = \begin{pmatrix} & \\ & \end{pmatrix} \qquad \text{where } k_{ij} = f(a_i)f(a_j).$$

Given a matrix K does there exist a function f such that $K_{ij} = f(a_i)f(a_j)$?

There exists a function f if and only if K is a positive semi definite matrix.

Kernel matrix

Gaussian kernel $k_{ij} = e^{\frac{1}{2}\frac{(a_i - a_j)^2}{\sigma^2}}$

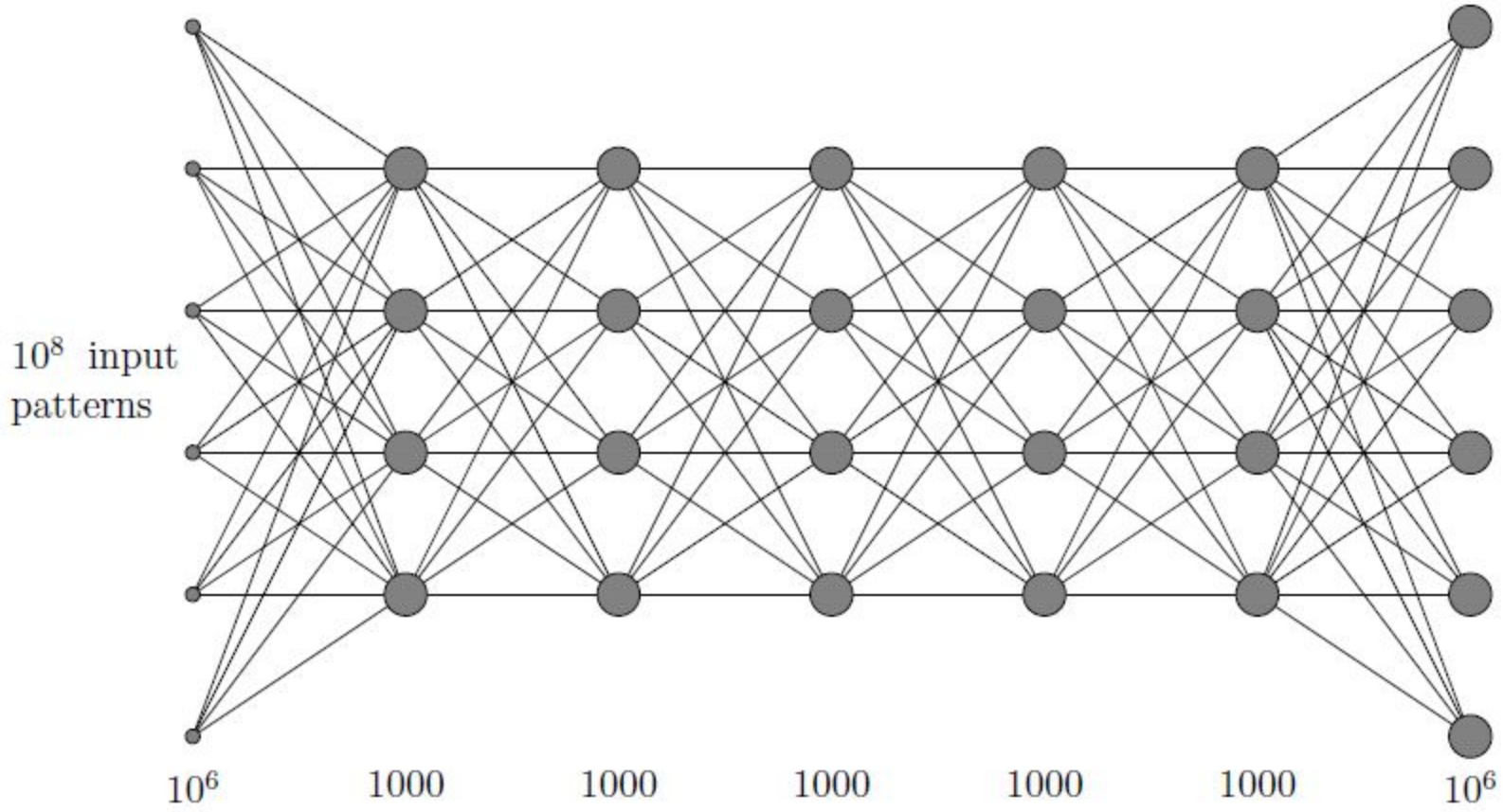
Support vector machine

Kernels and mapping to higher dimensional space is the essence of support vector machines.

There exist many kernels such as the Gaussian kernel.

The next advance is in deep learning.

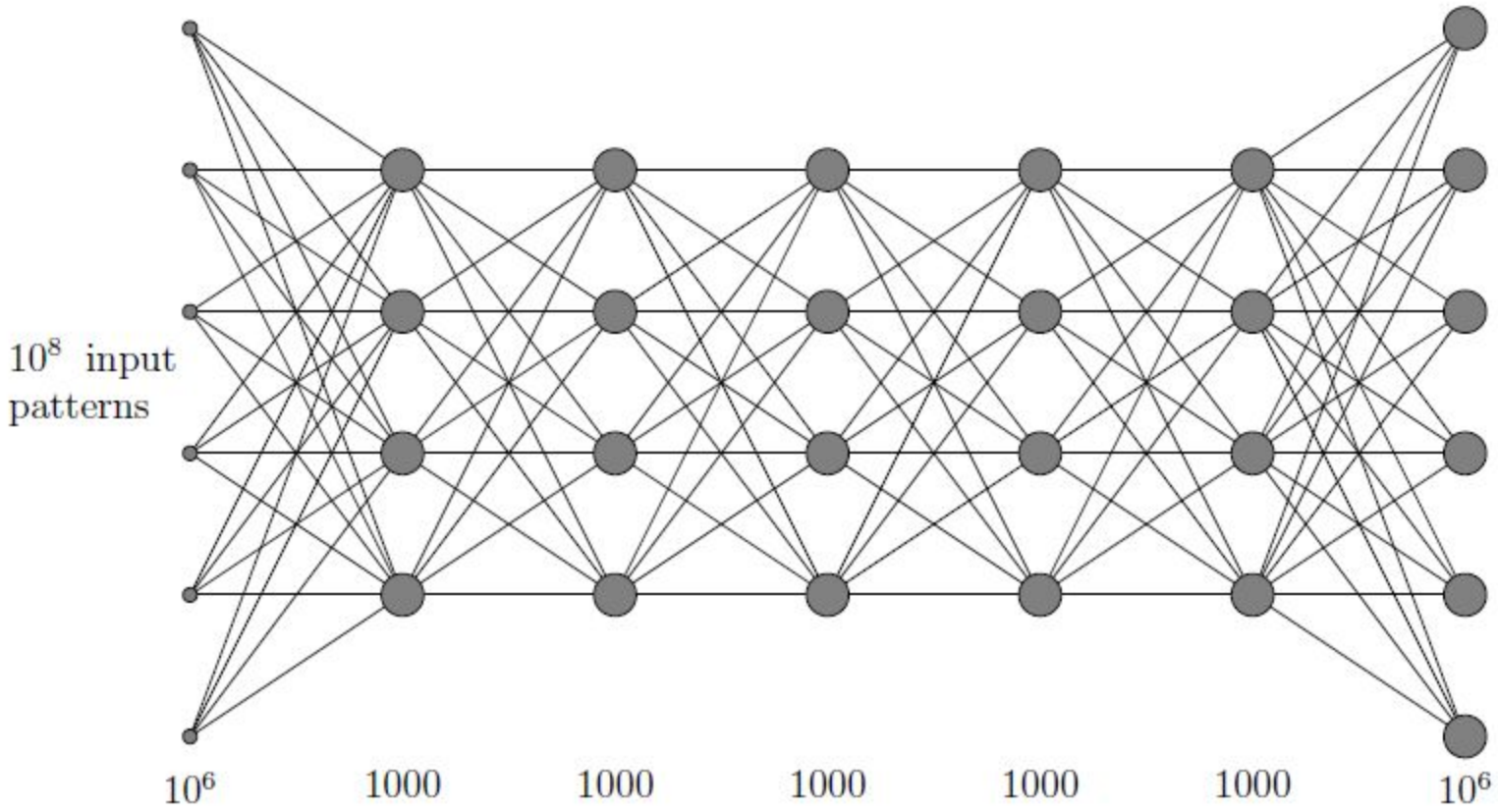
Deep Learning



Issues

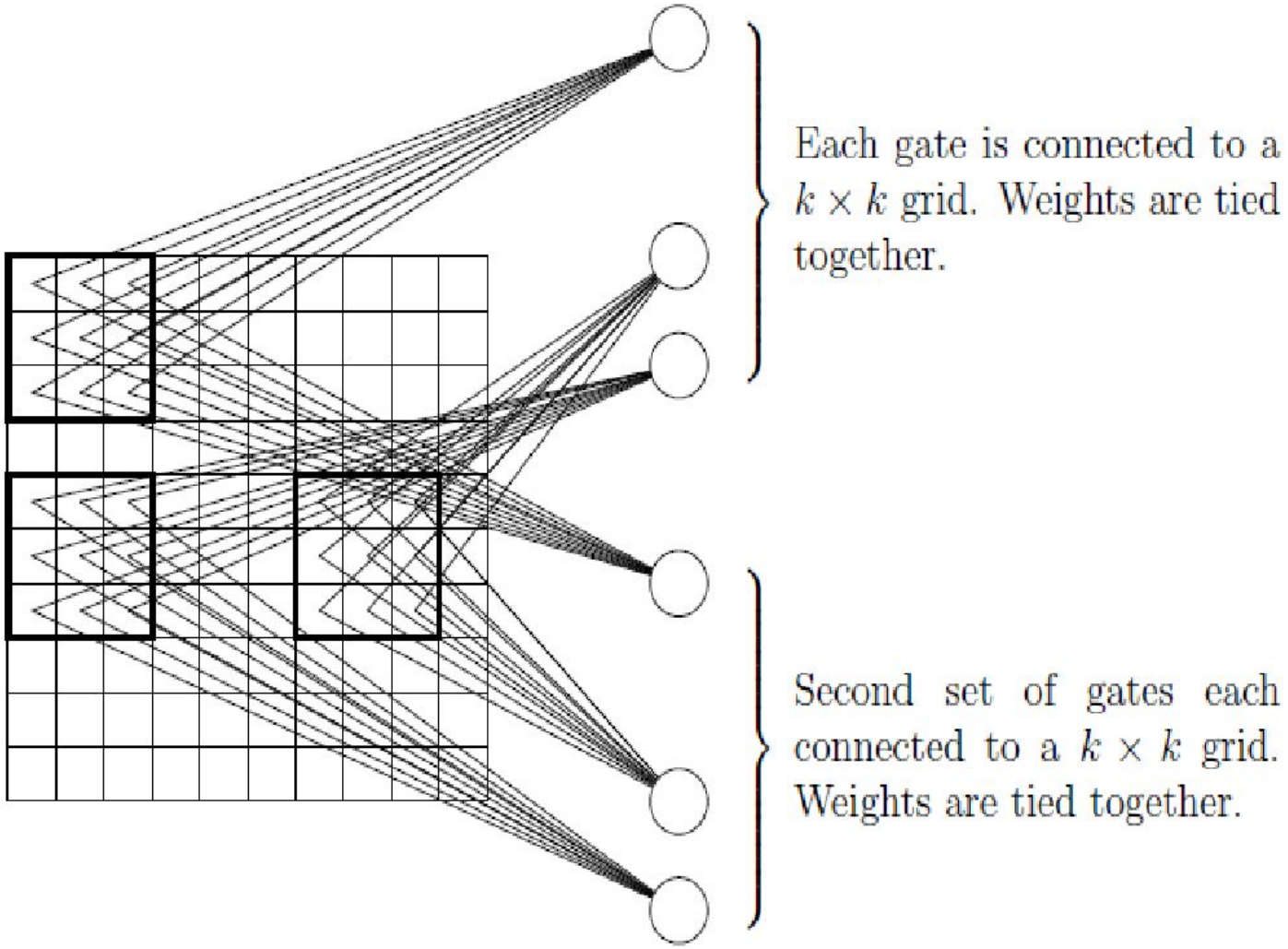
- Unsupervised learning
- Create error function
- Error function needs to be differentiable
- Convergence time

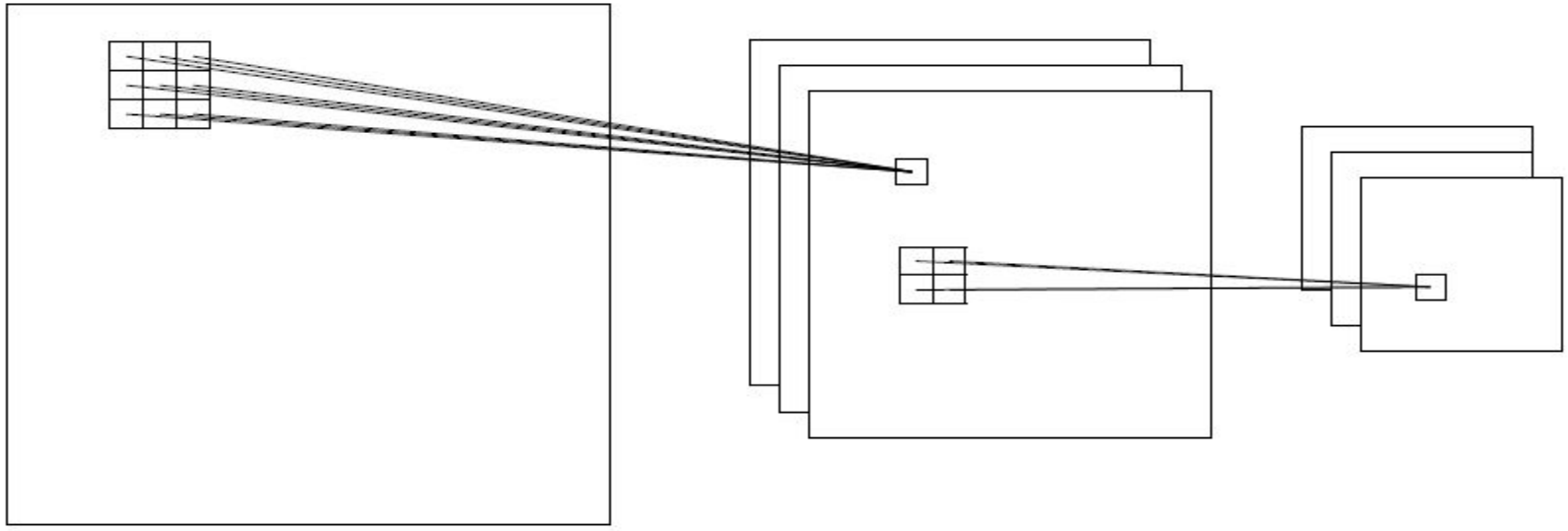
Autoencoder



Some research questions

- What do individual gates learn?
- How does what the second level gates learn differ from what the first level gates learn?
- How does what a gate learns evolve over time?
- Train with two different sets of starting weights. Do gates learn the same things?
- Train two networks with different sets of photographs. Do early gates learn the same things?


Each gate is connected to a $k \times k$ grid. Weights are tied together.
Second set of gates each connected to a $k \times k$ grid. Weights are tied together.


convolution
pooling

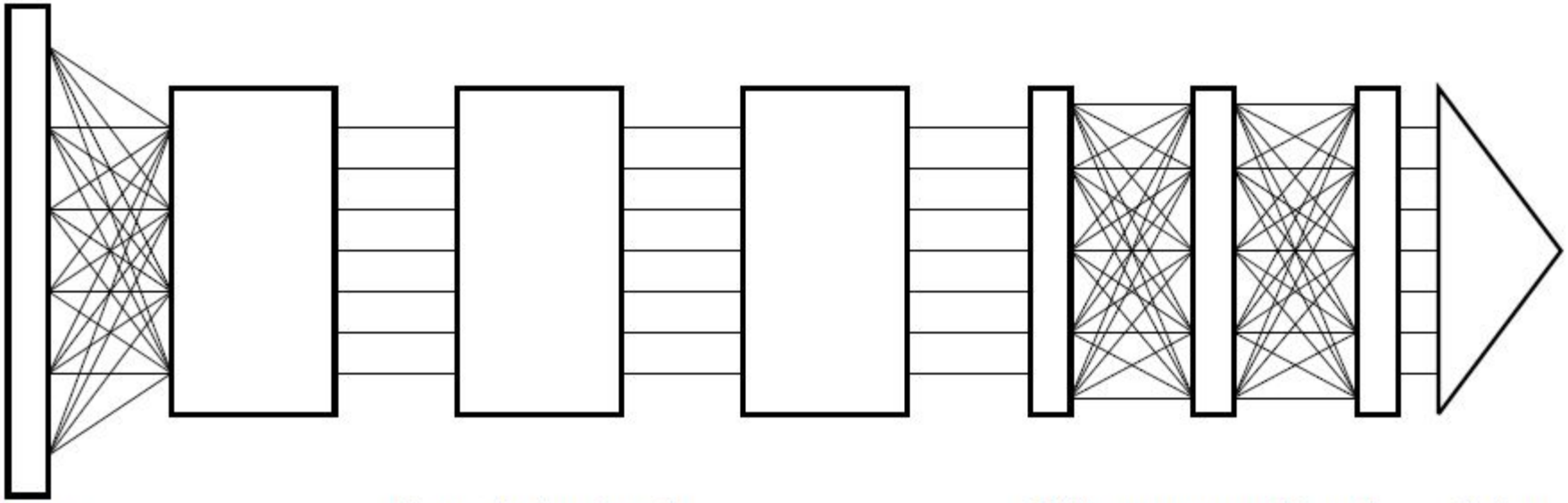


Image
Convolution levels
Fully connected levels
Softmax

Changing young to old

Jacob R. Gardner*, Paul Upchurch*, Matt J. Kusner, Yixuan Li, Kilian Q. Weinberger, Kavita Bala, John E. Hopcroft

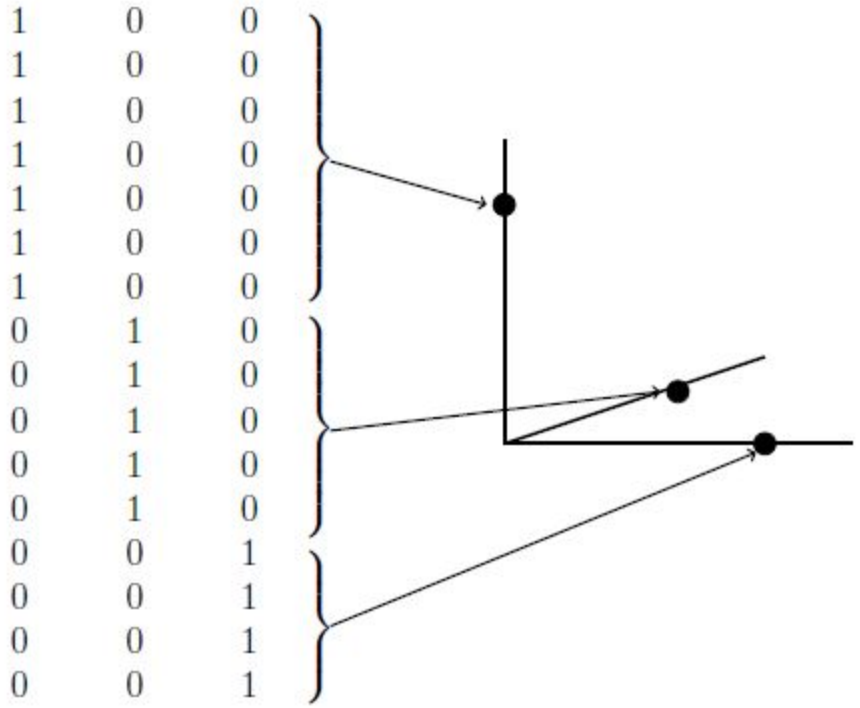
Changing artistic style

A Neural Algorithm of Artistic Style
Leon A. Gatys, Alexander S. Ecker, Matthias Bethge

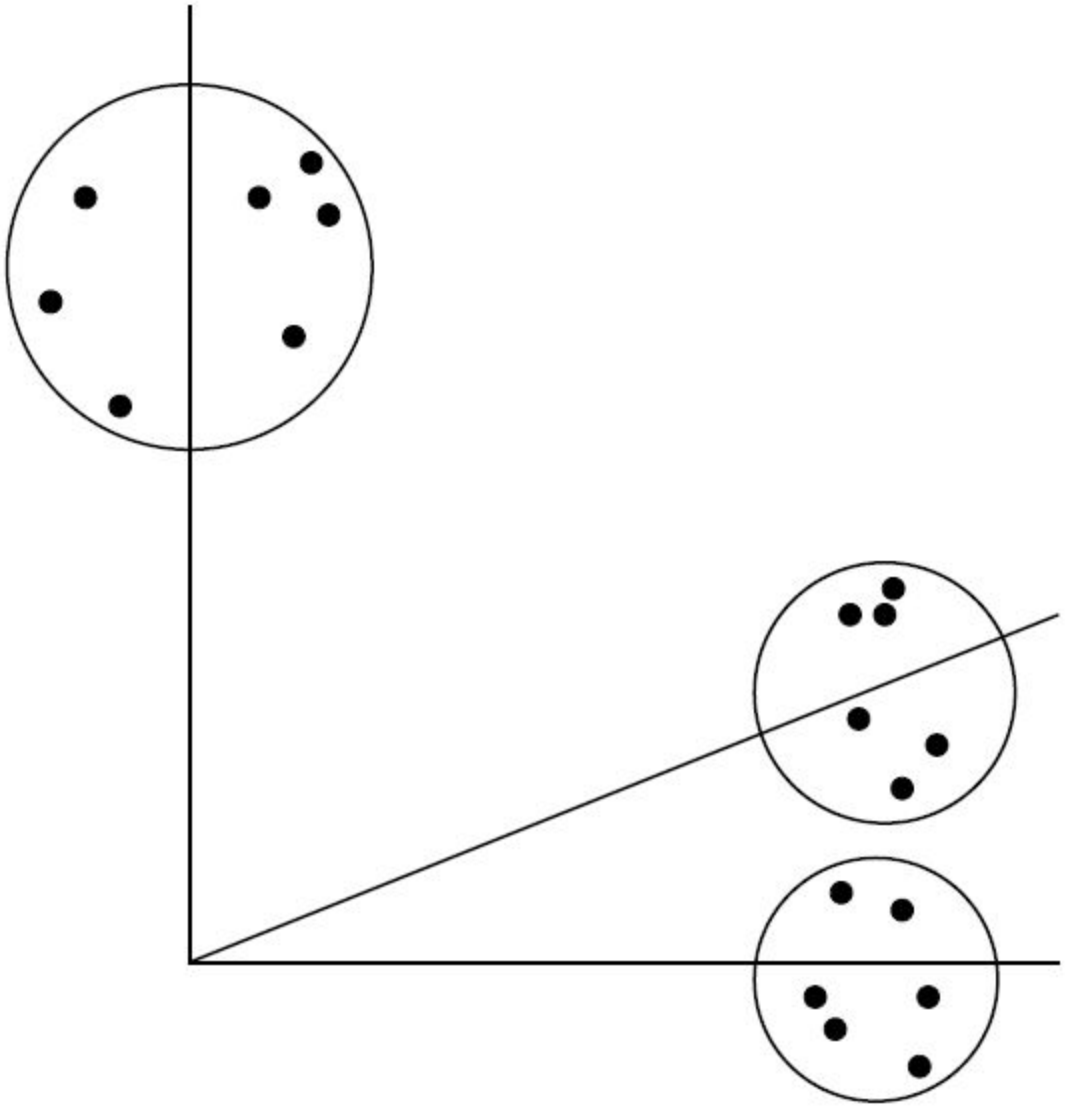
Spectral clustering

$$A = \begin{pmatrix} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ & & \vdots & \\ 1 & 1 & \cdots & 1 \end{pmatrix} & 0 & 0 \\ 0 & \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ & & \vdots & \\ 1 & 1 & \cdots & 1 \end{pmatrix} & 0 \\ 0 & 0 & \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ & & \vdots & \\ 1 & 1 & \cdots & 1 \end{pmatrix} \end{pmatrix}$$

$$\begin{matrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{matrix}$$



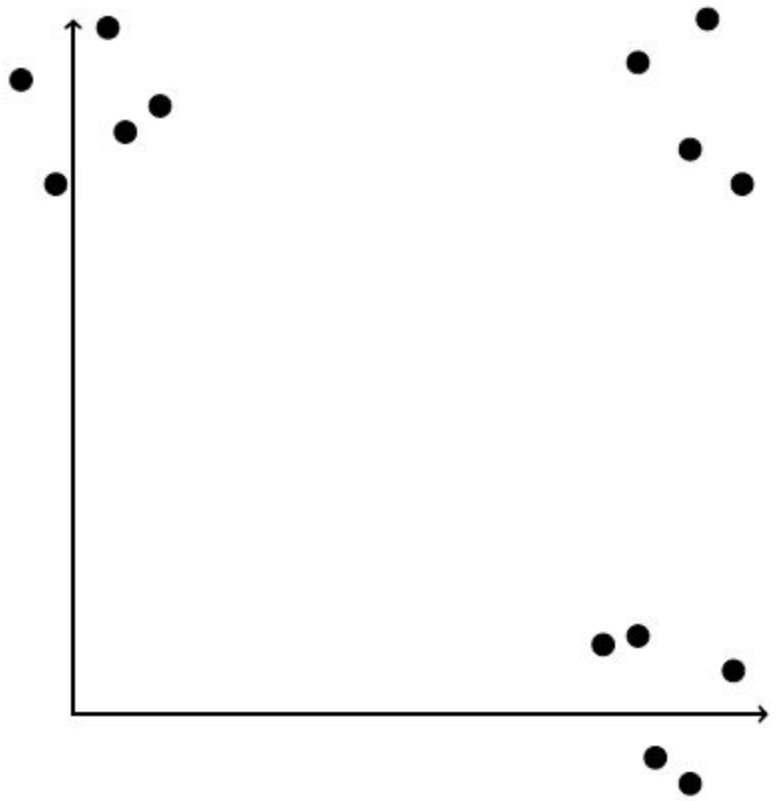
Spectral clustering with K-means



Spectral clustering with K-means

What if communities overlap?

1	0
1	0
1	0
1	1
1	1
0	1
0	1
0	1



Instead of two overlapping clusters, we would find three clusters.

- Instead of clustering the rows of the singular vectors, find the minimum 0-norm vector in the space spanned by the singular vectors.

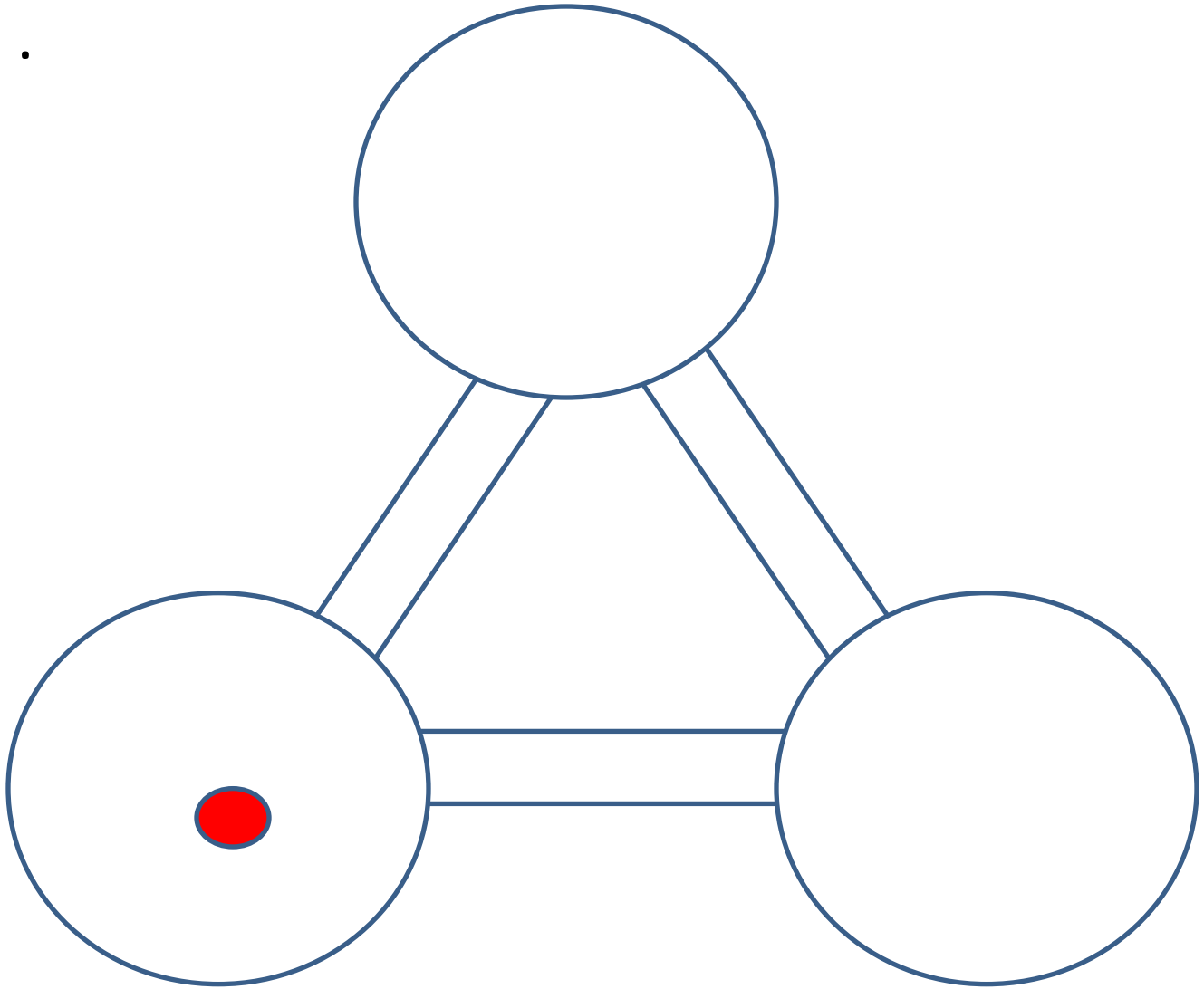
- Finding the minimum 0-norm vector is NP-hard.

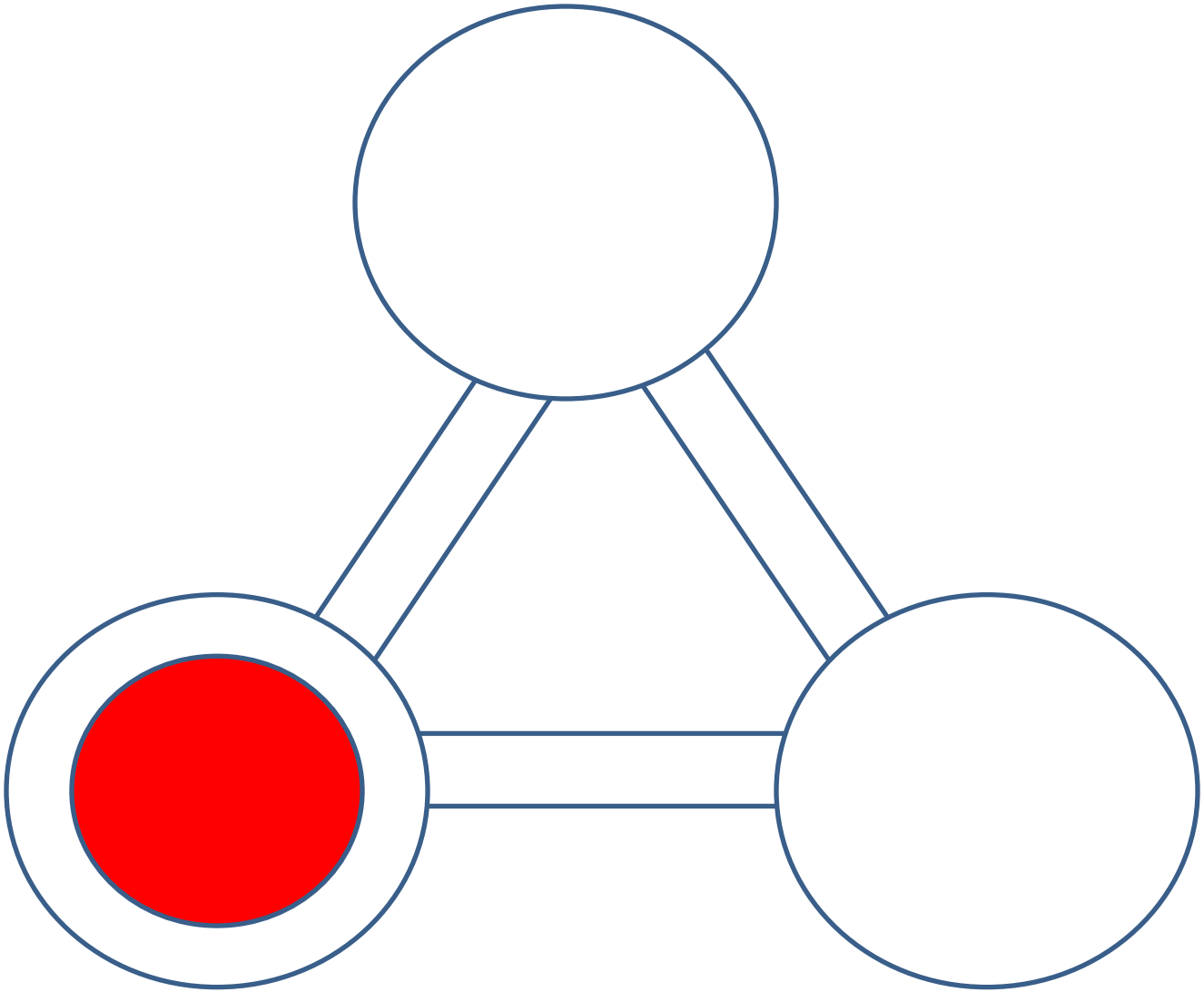
- Use the minimum 1-norm vector as a proxy. This is a linear programming problem.

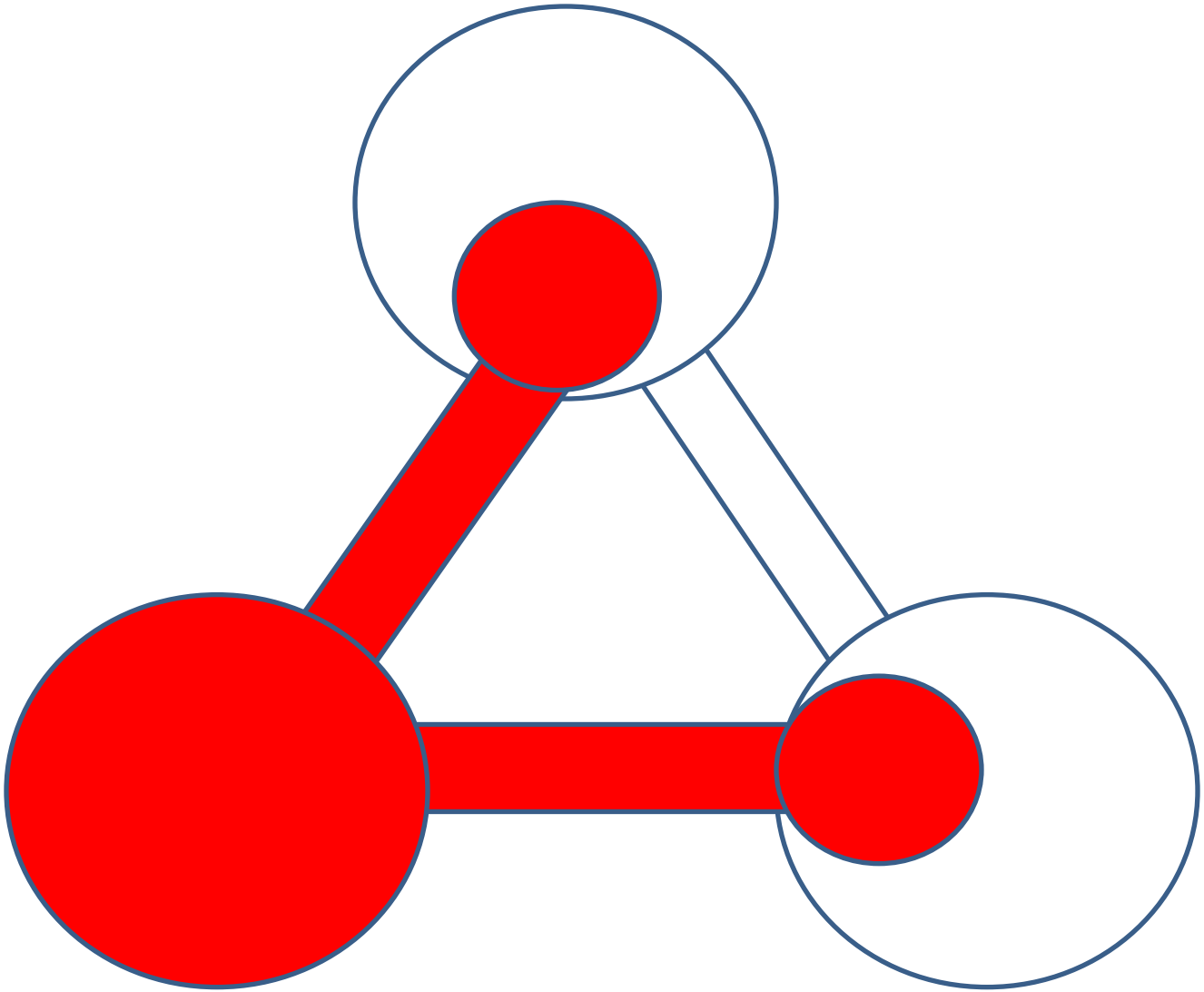
- What we have described is how to find global structure.

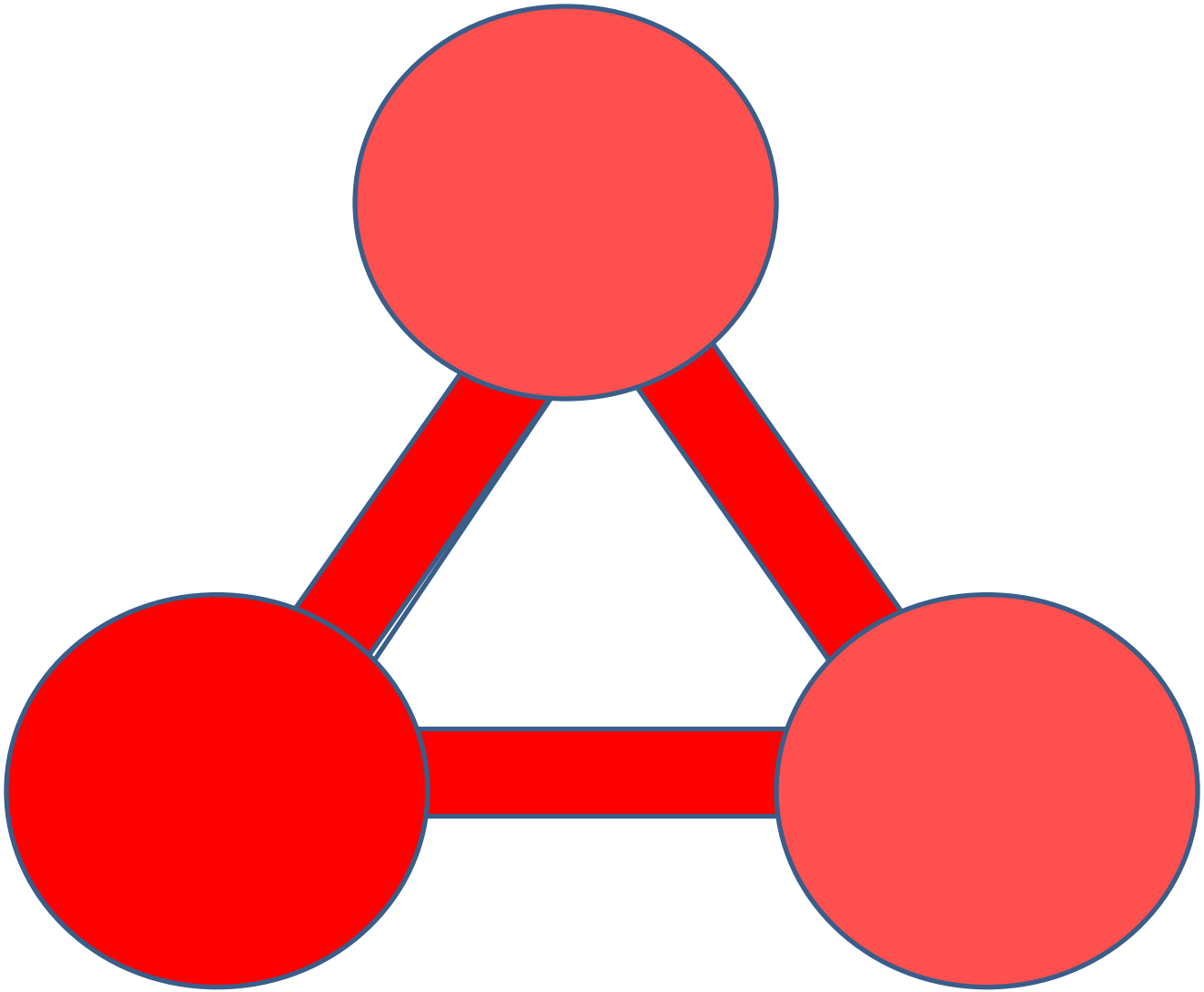
- We would like to apply these ideas to find local structure.

We want to find community of size 50 in a network of size 10^9 .









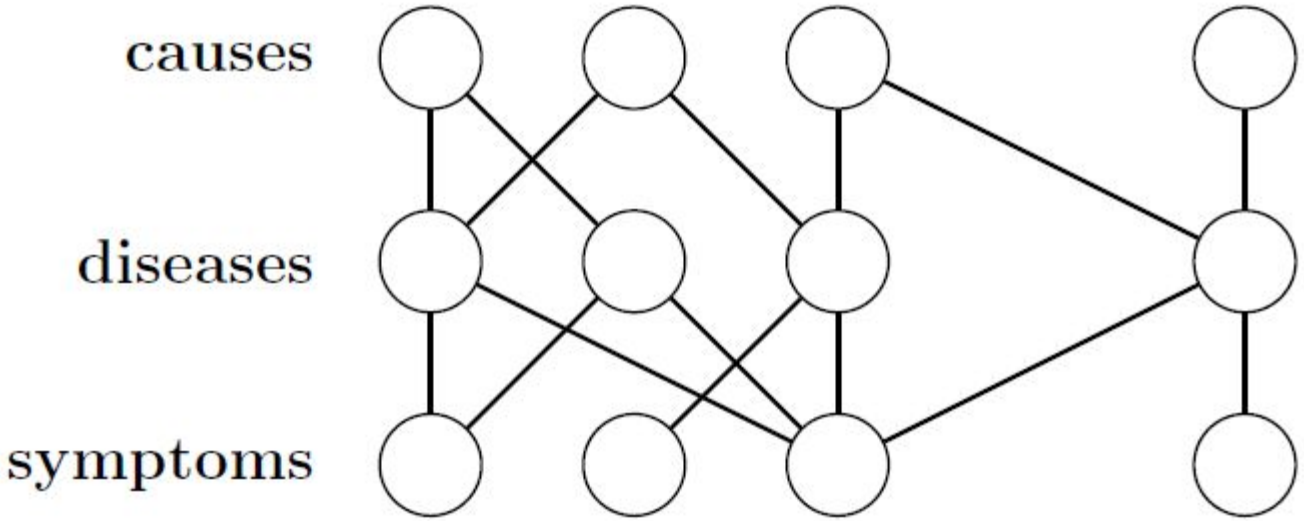
Instead of finding singular vectors, take a small number of steps in a random walk.

Look at early convergence of the random walk.

Find the minimum one norm vector in $A^5 \underbrace{[x, Ax, A^2x]}_{\text{dim of space}}$

(A^5: # of steps)

Graphs with 2^{100} vertices



MCMC

$$\mathbf{x} = (x_1, x_2, \ldots, x_{100})$$

If each $x_i \in \{0, 1\}$, then x has 2^{100} possible values.

How would you compute the expected value of a a function $f(x)$?

$$E(f(x)) = \sum_x f(x)p(x)$$

One cannot sum 2^{100} values so one samples, but they must sample according to $p(x)$.

Markov Chain Monte Carlo

This raises the question, how do you sample according to a given probability distribution?

Construct a graph whose vertices correspond to the values of x and assign probabilities to the edges so that the stationary probability of a random walk is p(x).

Markov Chain Monte Carlo

How do you store a graph with 2^{100} vertices in a computer?

How long does it take for a random walk to converge to its stationary probability?

Expander: Converges in time logarithmic in the number of vertices.

$$\ln 2^{100} = 100$$

Storing a random bit vector of length n in log n bits.

$a_1, a_2, \ldots, a_n \qquad a_i \in \{1, 2, \ldots, m\}$

f_s number of occurrences of symbol s

$\sum_{s=1}^{m} f_s^2$ variance of stream

Question: Do some symbols occur much more frequently than others?

Summation $\sum_{s=1}^{m} f_s^2$ appears to require m counters to compute.

10^{16} counters?

Can estimate $\sum_{s=1}^{m} f_s^2$ with one counter.

Generate $x = (x_1, x_2, \ldots, x_m)$ each $x_i = \pm 1$ with probability $\frac{1}{2}$

Compute $a = \sum_{s=1}^{m} x_s f_s$

a^2 is a good estimator of $\sum_{s=1}^{m} f_s^2$

Problem: How do we store x?

If you can use pseudo random x instead of fully random x, you can store x in log m bits.

$$\log 10^{16}$$

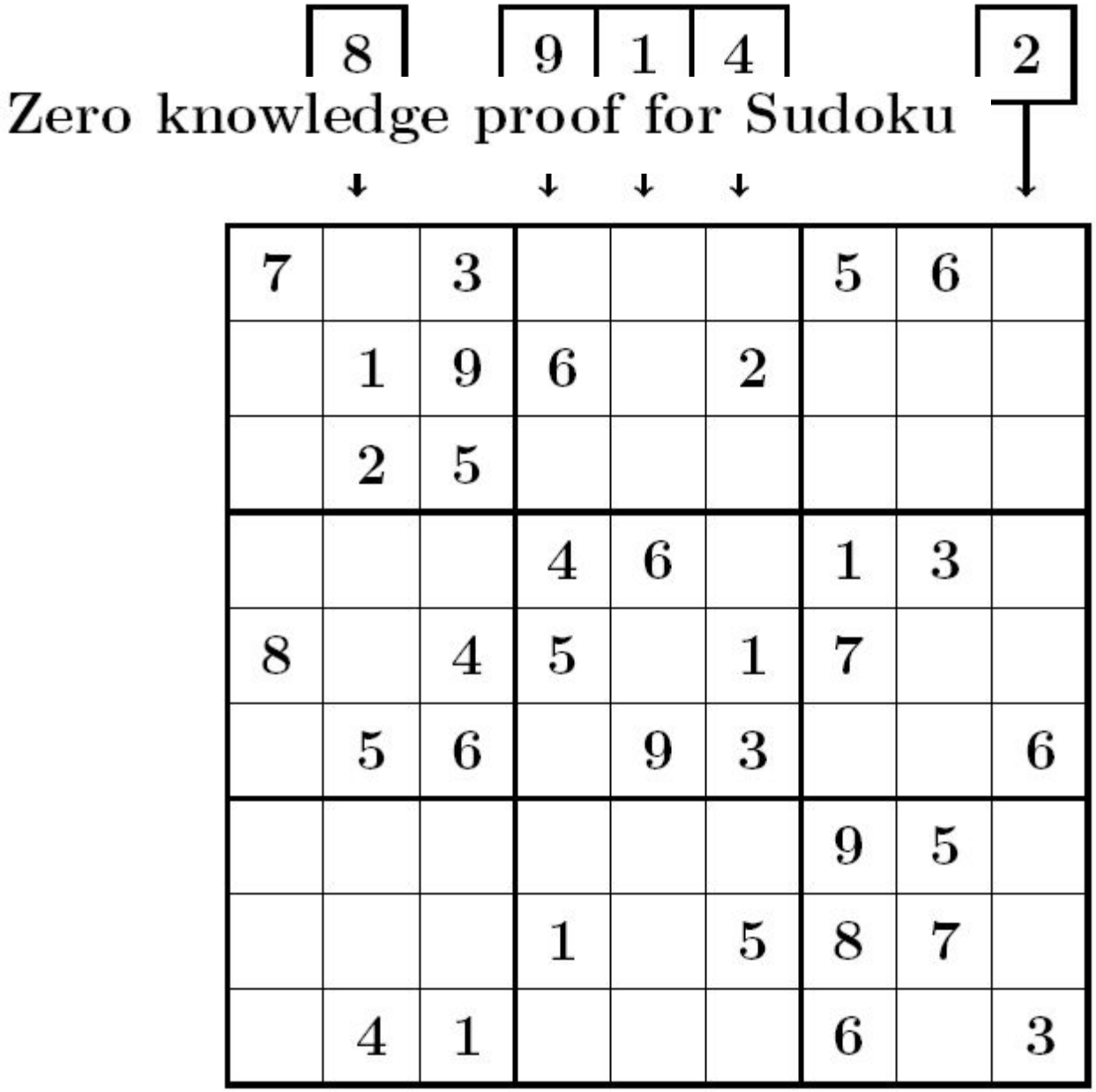
Need only 4-way independence.

Digitization of medical records

- Doctor – needs my entire medical record
- Insurance company – needs my last doctor visit, not my entire medical record
- Researcher – needs statistical information but no identifiable individual information

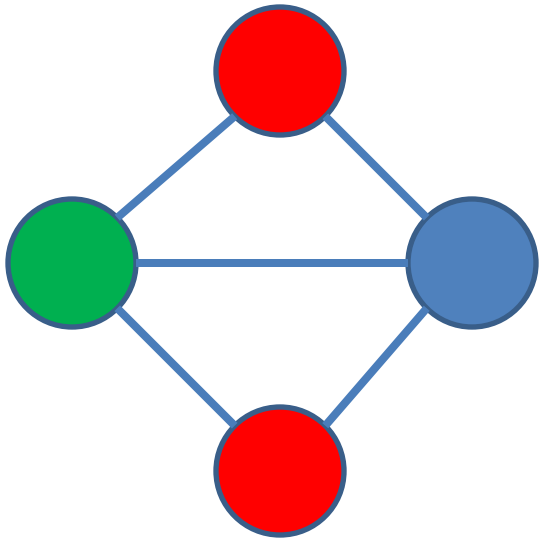
Relevant research – zero knowledge proofs, differential privacy

A zero knowledge proof of a statement is a proof that the statement is true without providing you any other information.


8
9 1 4
2
Zero knowledge proof for Sudoku
7 3 5 6
1 9 6 2
2 5
4 6 1 3
8 4 5 1 7
5 6 9 3 6
9 5
1 5 8 7
4 1 6 3

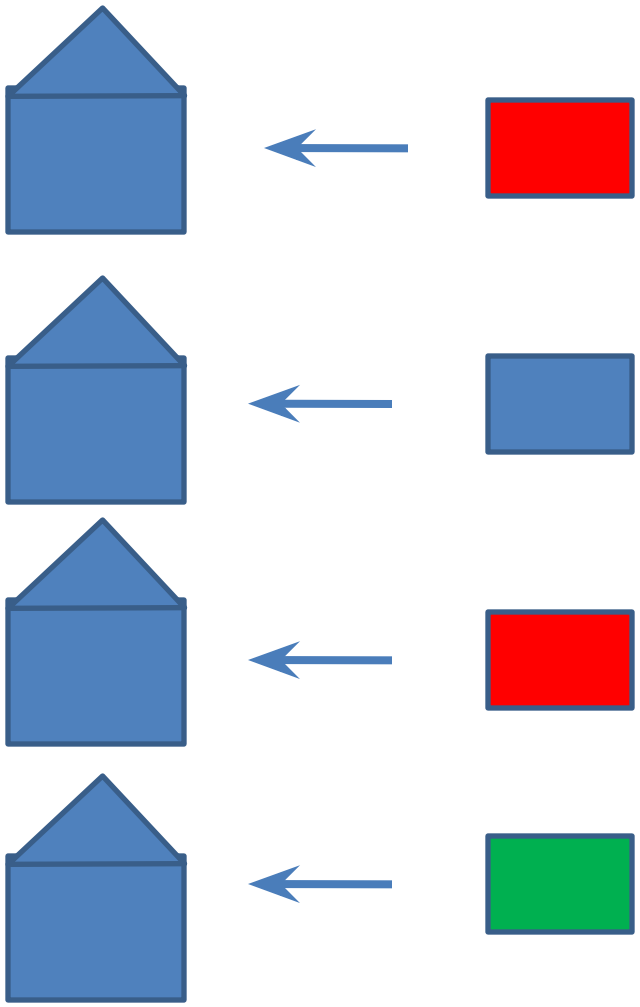
Zero knowledge proof

- Graph 3-colorability



- Problem is NP-hard - No polynomial time algorithm unless P=NP

Zero knowledge proof



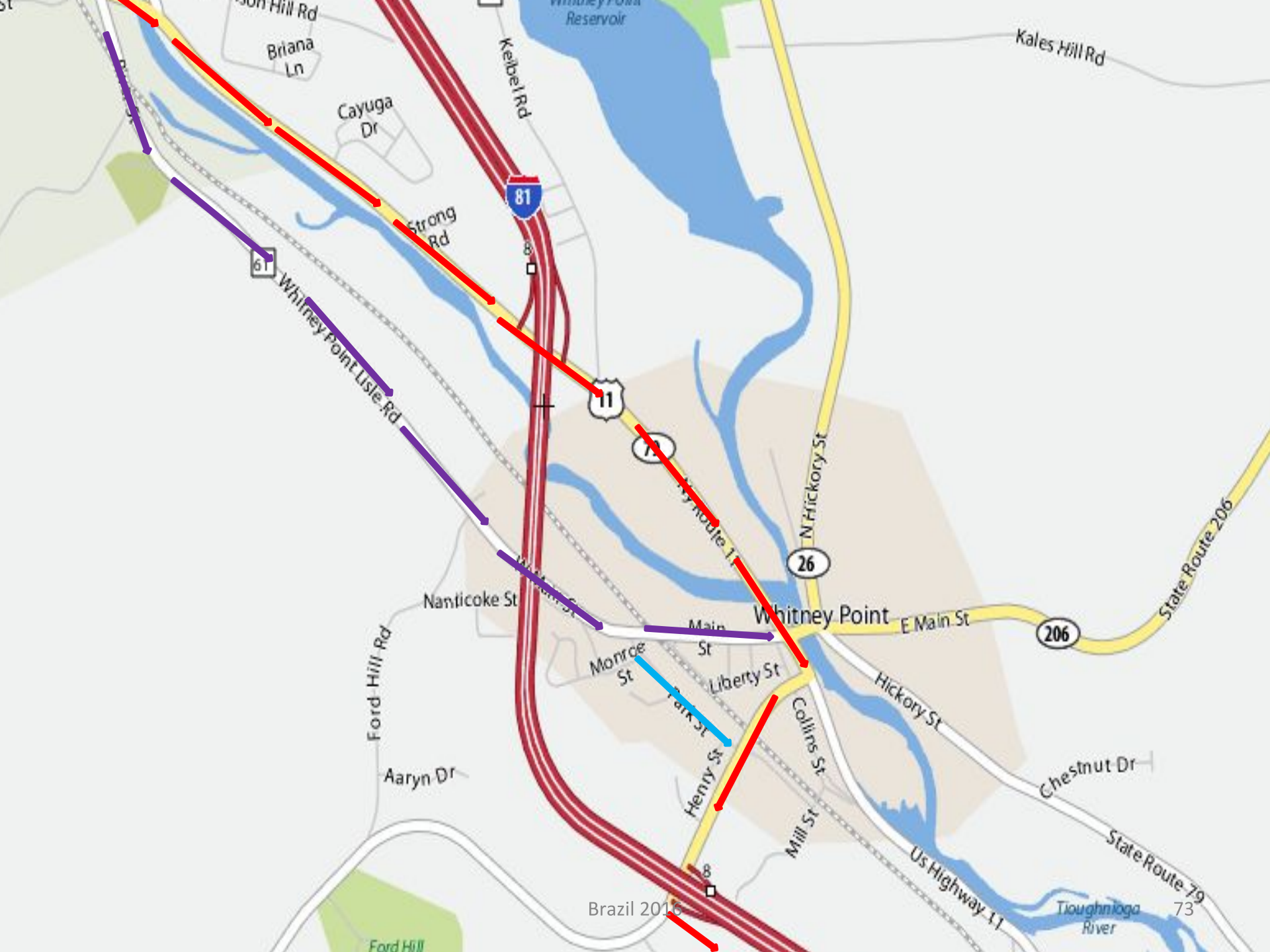
I send the sealed envelopes.

You select an edge and open the two envelopes corresponding to the end points.

Then we destroy all envelopes and start over, but I permute the colors and then resend the envelopes.

Digitization of medical records is not the only system

- Car and road – gps – privacy
- Supply chains
- Transportation systems


Whitney Point Reservoir
Kales Hill Rd
Briana Ln
Cayuga Dr
Kelbel Rd
81
Strong Rd
61
Whitney Point Lisle Rd
11
N Hickory St
26
Nanticoke St
Whitney Point
E Main St
206
State Route 206
Main St
Monroe St
Liberty St
Park St
Hickory St
Collins St
Ford Hill Rd
Aaryn Dr
Henry St
Mill St
Chestnut Dr
State Route 79
Us Highway 11
Tioughnioga River
Ford Hill

High dimensional data

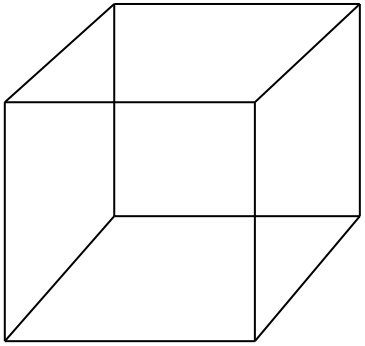
- High dimensional data is inherently unstable.

- Given n random points in d-dimensional space, essentially all n^2 distances are equal.

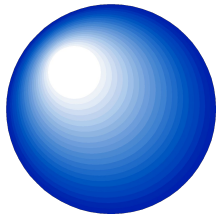
- $$|x - y|^2 = \sum_{i=1}^{d} (x_i - y_i)^2$$

High Dimensions

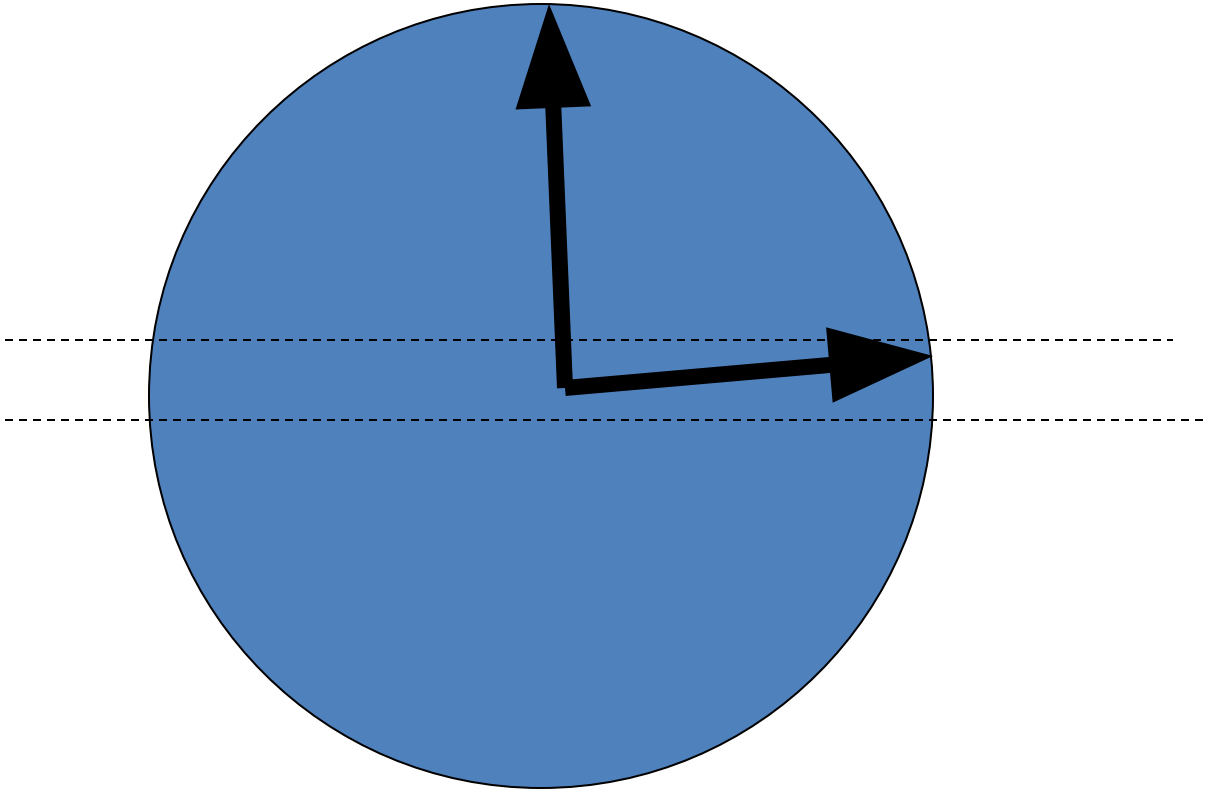
Intuition from two and three dimensions is not valid for high dimensions.

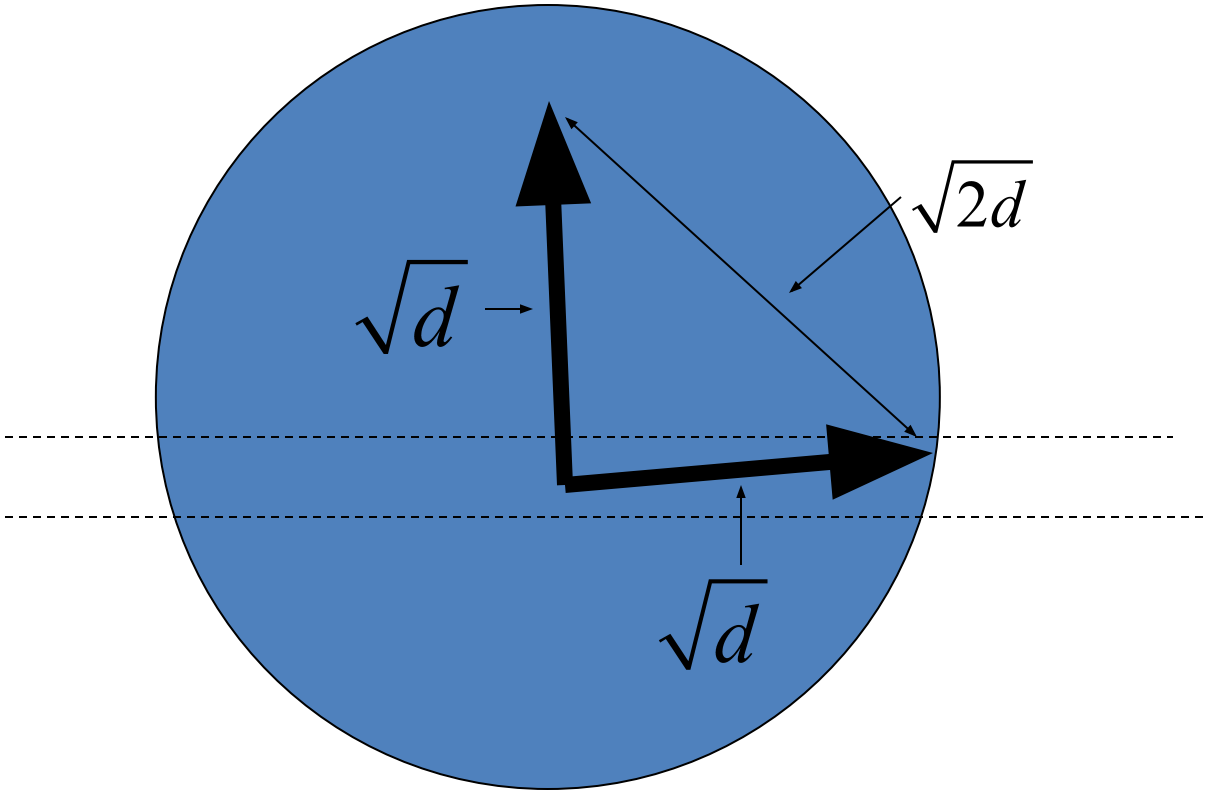


Volume of cube is one in all dimensions.



Volume of unit radius sphere goes to zero.





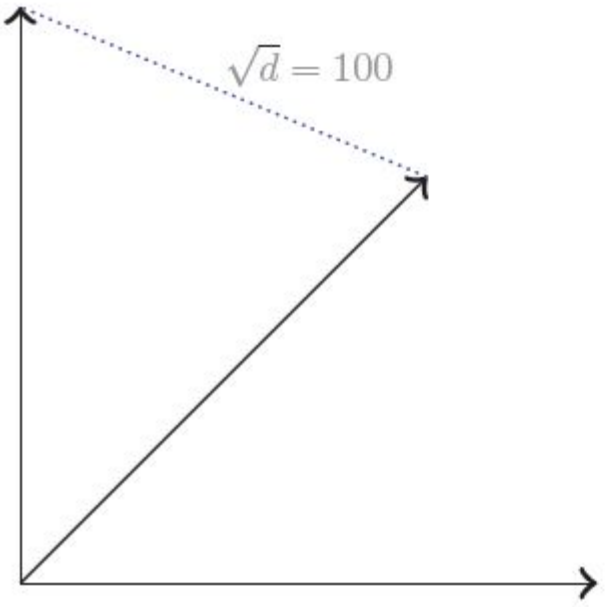
All data points are on the equator, no matter where you put the equator.

Dimension reduction

- **Johnson Lindenstrauss Theorem:** Project n points in d-dimensional space to a k-dimensional space and all distances are reduced by approximately the same amount.

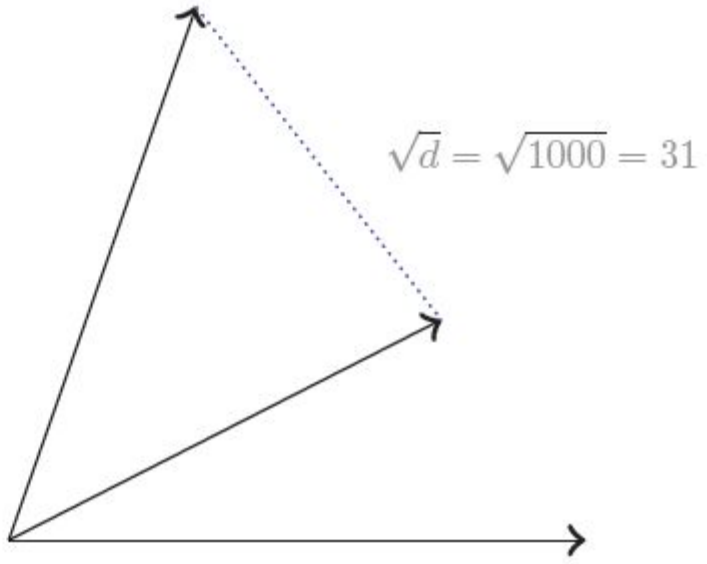
- This allows one to cluster and solve other problems in a lower dimensional space.

- How many orthogonal vectors can I find in a d-dimensional space?
- Select d basis vectors to get d orthogonal vectors.
- What if I only require almost perpendicular?
- Select 10,000 perpendicular vectors in 10,000 dimensions and project down to 1,000 dimensions.
- This creates 10,000 almost perpendicular vectors in 1,000 dimensions.



10,000 coordinates
10,000 dimensions
90 degrees

(a)



10,000 vectors
1000 dimensions
90 ± 5 degrees

(b)

- Computer science has undergone a fundamental change.
- There is very exciting research being undertaken.
- Those individuals, institutions, and nations who position themselves for the future will benefit immensely.

Thank You!