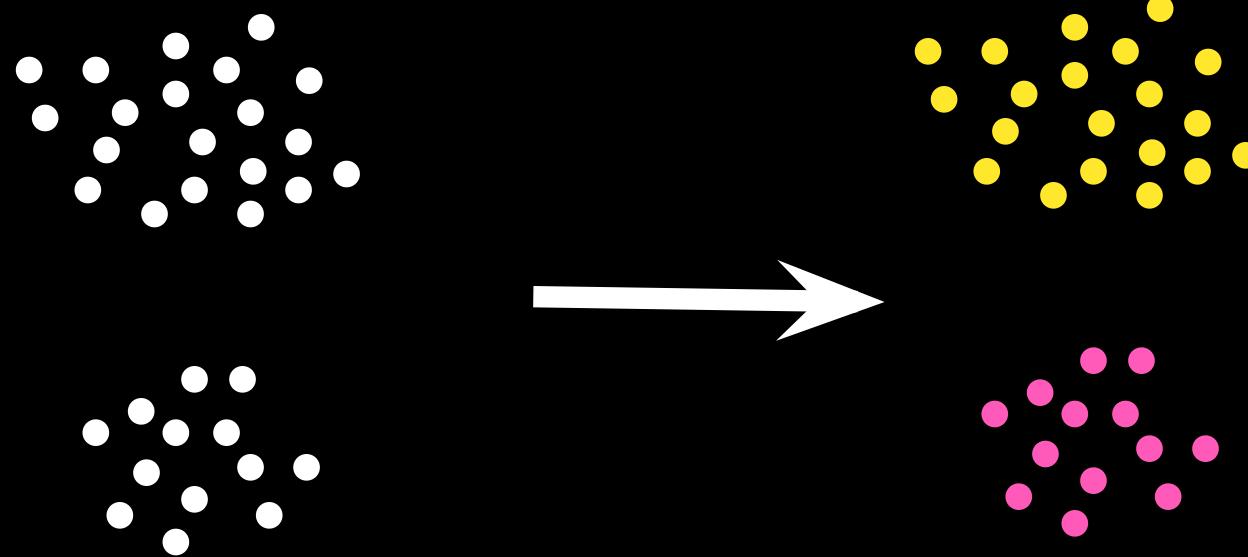


Beyond Pairwise Clustering

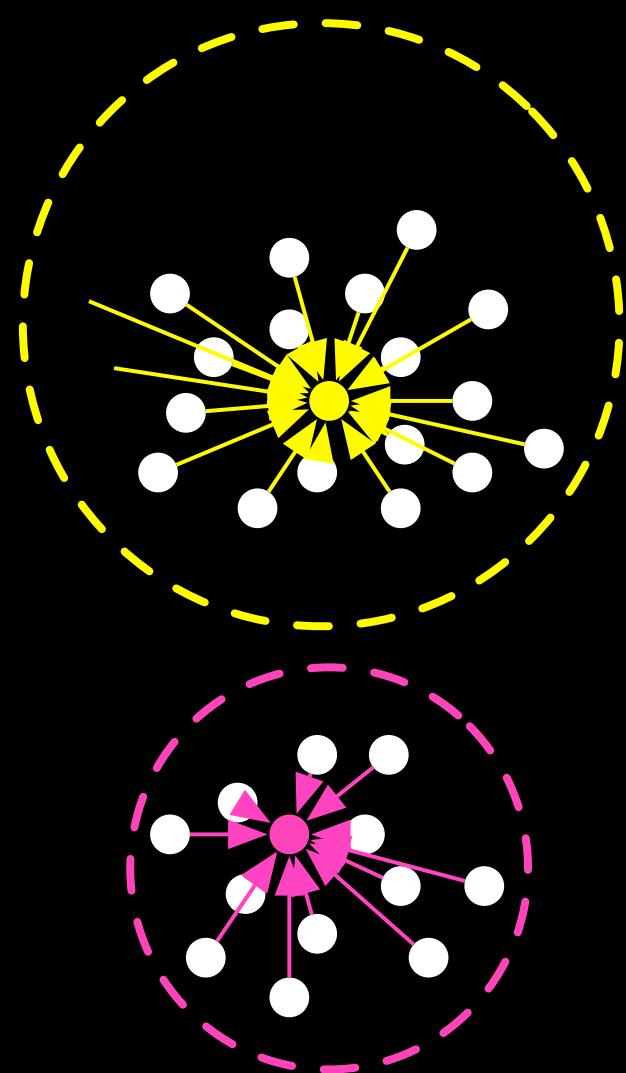
Sameer Agarwal, Jongwoo Lim, Lihi Zelnik-Manor
Pietro Perona, David Kriegman, Serge Belongie

UCSD & Caltech

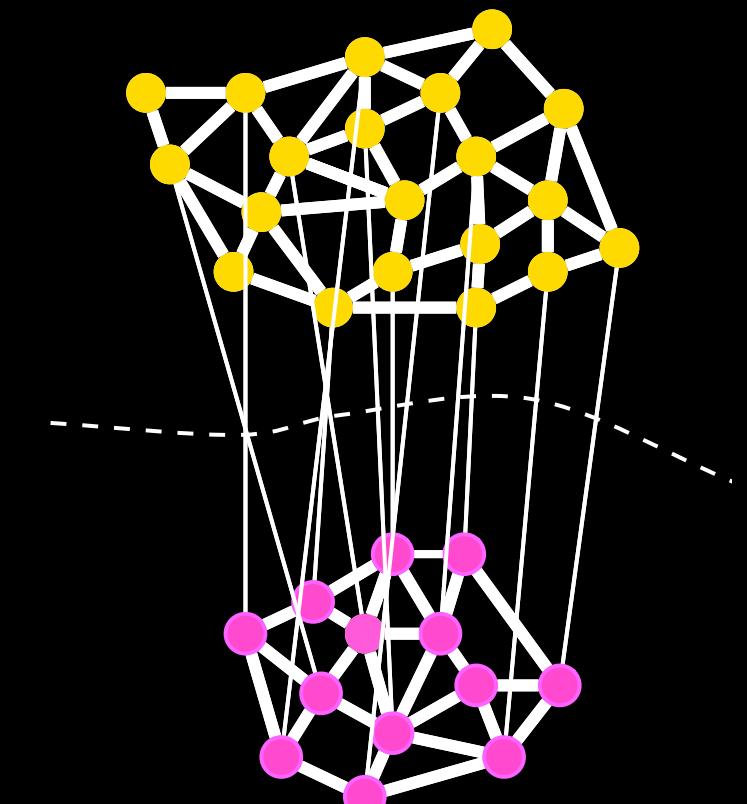
The Clustering Problem



Central Clustering

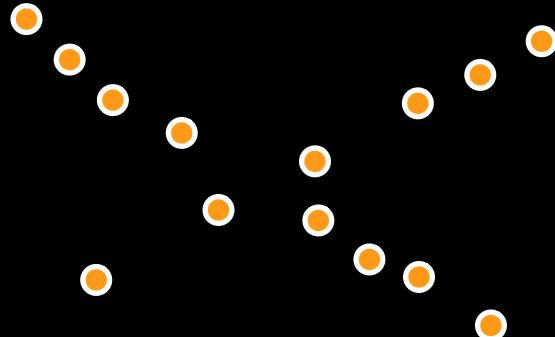


Pairwise Clustering

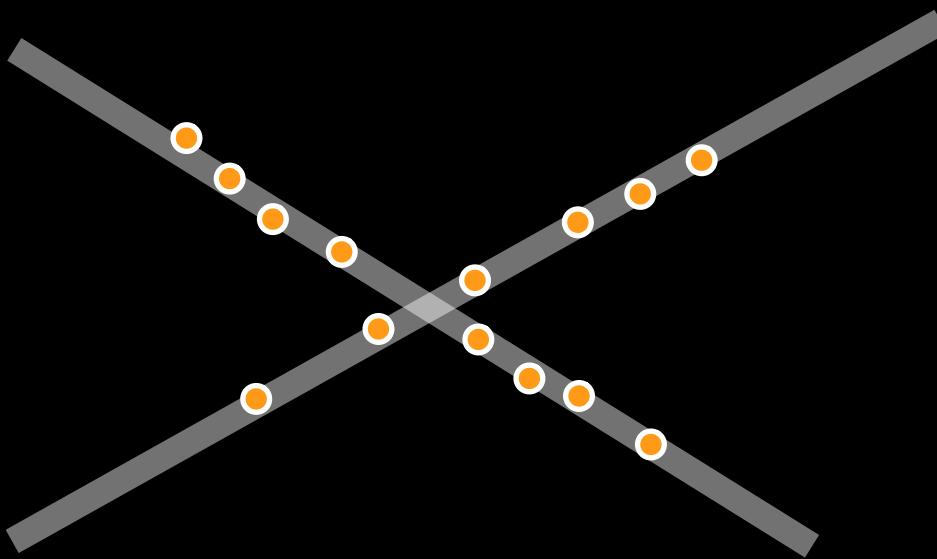


**But what if pairwise
information is not
available?**

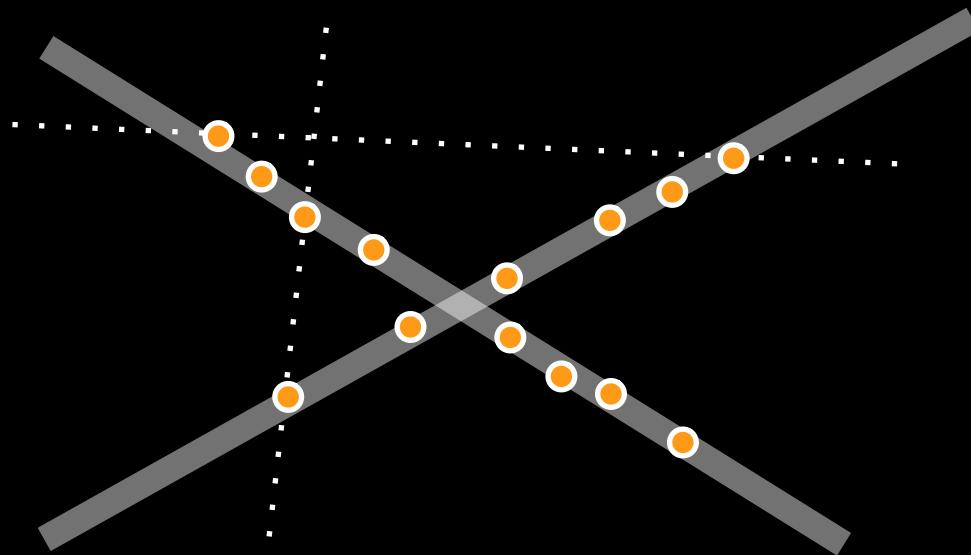
k-Lines Clustering



k-Lines Clustering

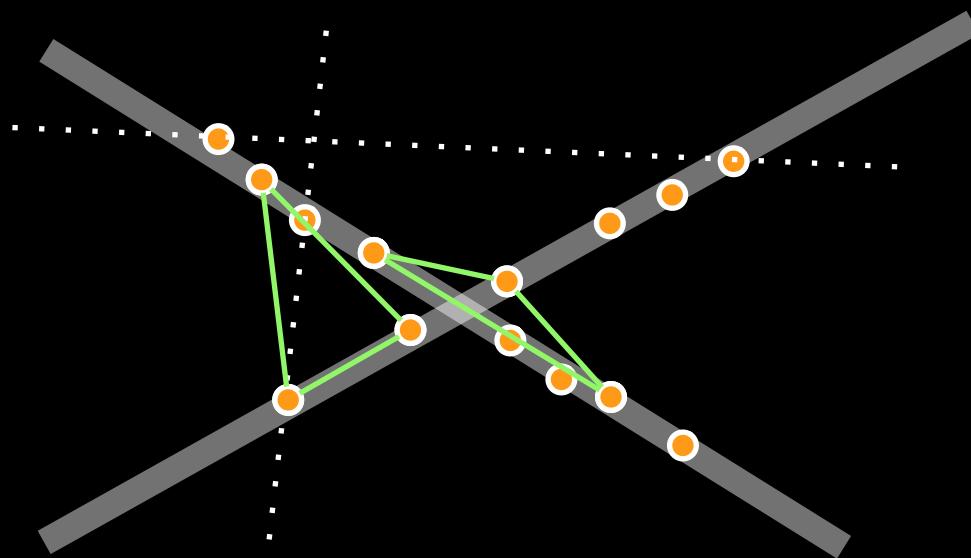


k-Lines Clustering

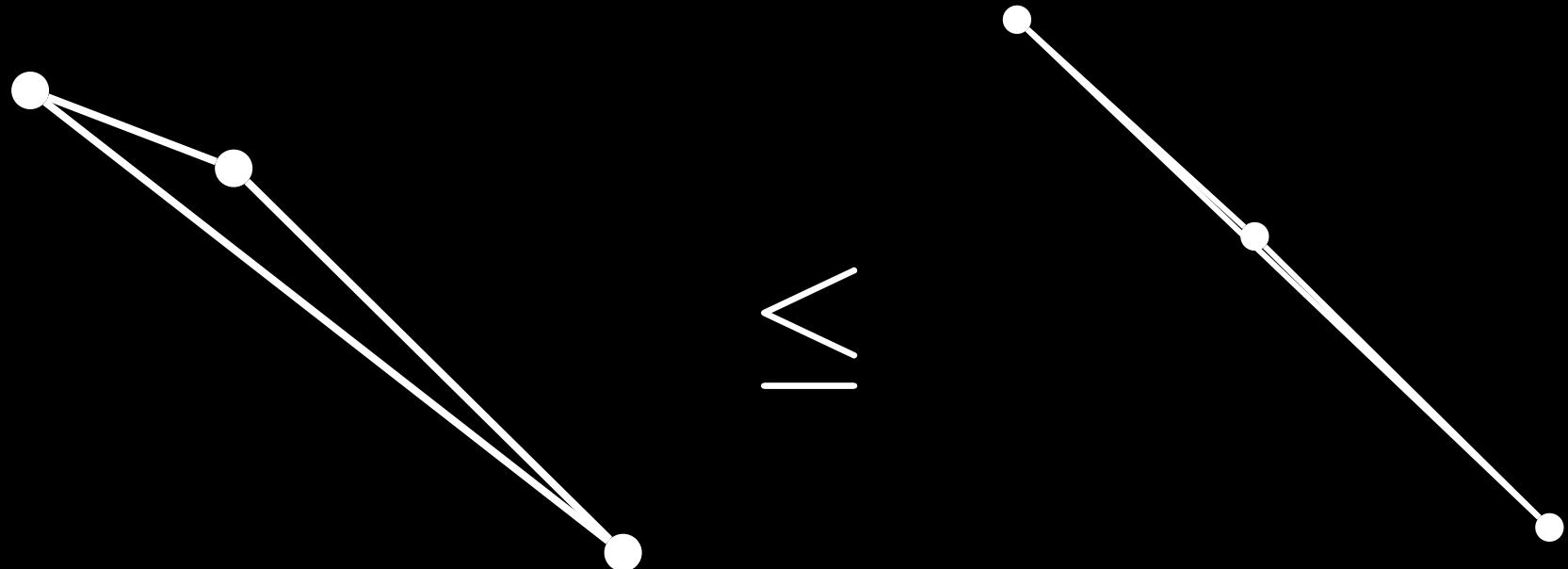


Every pair of points define a line!

k-Lines Clustering



But Three Points...

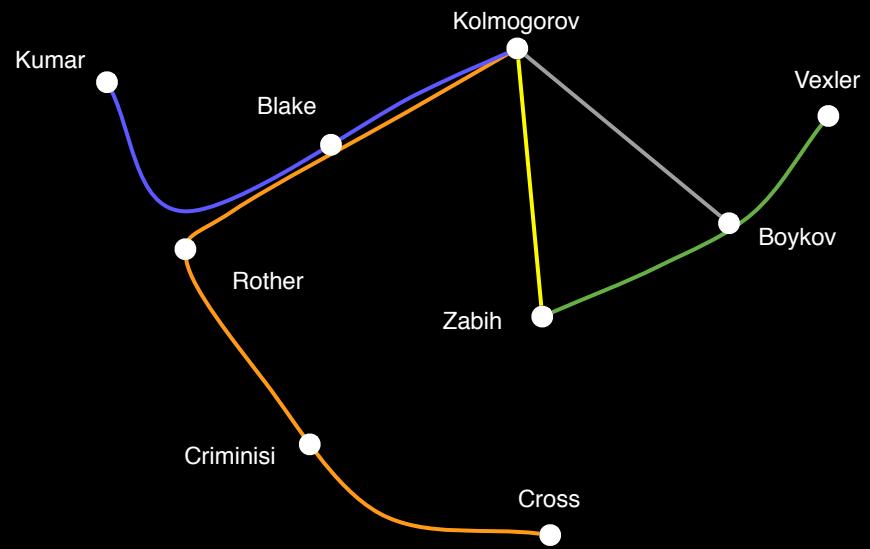


Higher Order Clustering

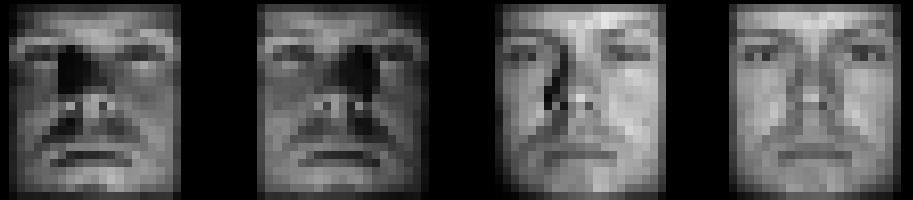
Clustering in domains where affinities/distances
are defined over general subsets of the data.

Examples

- Co-authorship graphs
- Illumination Invariant Clustering
- Motion Clustering
- Mixtures of models
 - k-subspaces

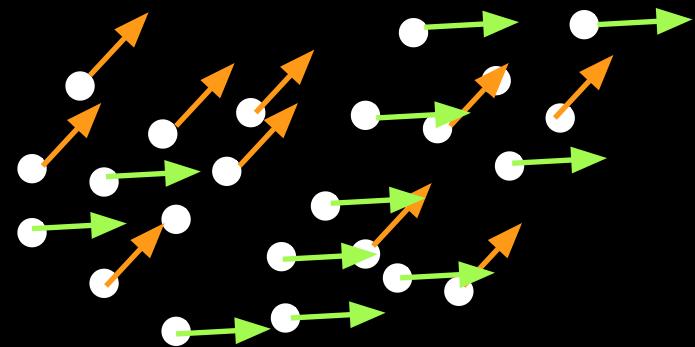


Examples



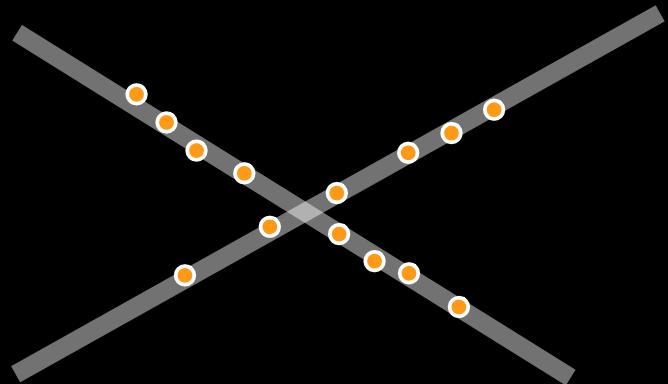
- Co-authorship graphs
- Illumination Invariant Clustering
- Motion Clustering
- Mixtures of models
 - k-subspaces

Examples



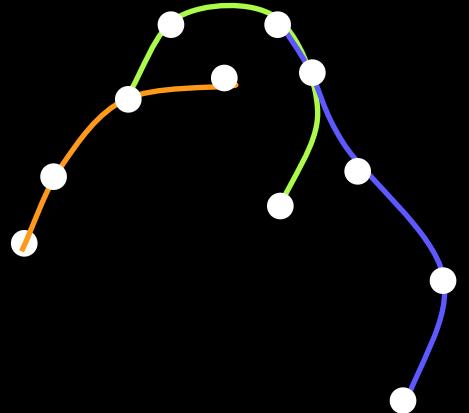
- Co-authorship graphs
- Illumination Invariant Clustering
- Motion Clustering
- Mixtures of models
 - k-subspaces

Examples



- Co-authorship graphs
- Illumination Invariant Clustering
- Motion Clustering
- Mixtures of models
 - k-subspaces

Hypergraphs



- Generalizations of graphs
- Edges can contain any number of vertices
- Represented using multi-dimensional arrays

Related Work

- Multidimensional Scaling
- Data mining
- VLSI CAD
 - Multiscale heuristics
 - Graph approximations

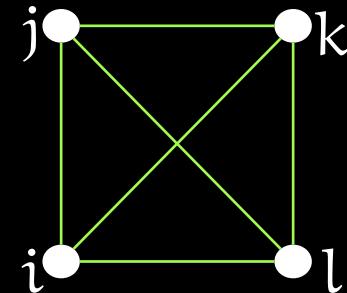
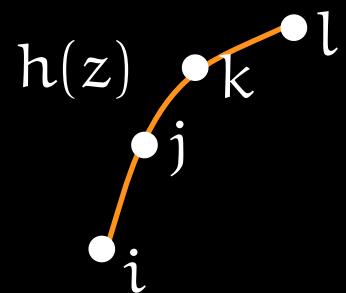
Related Work

- Multidimensional Scaling
- Data mining
- VLSI CAD
 - Multiscale heuristics
 - Graph approximations

Why approximate?

1. Graph partitioning methods are more advanced
2. Combinatorially a simpler problem.

Clique Expansion

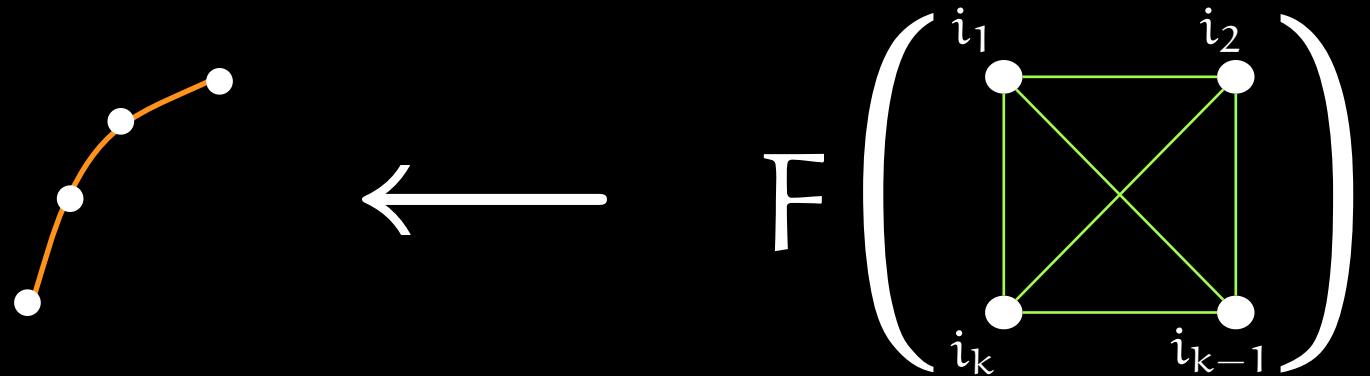


$$z = \{i, j, k, l\}$$

$$g(i, j) = h(z)$$

$$g(i, j) = \mu \sum_{i, j \in z} h(z)$$

A Generative Model



$$h(z) = F(g(i_1, i_2), \dots, g(i_k, i_{k-1}))$$

Properties of F

1. Positive
2. Symmetric
3. Monotonic

A Generative Model

$$F_p(\mathbf{x}) = \lambda \|\mathbf{x}\|_p$$

$$h(z) = \lambda \left(\sum_{i,j \in z} g^p(i,j) \right)^{1/p}$$

$$h^p(z) = \lambda^p \sum_{i,j \in z} g^p(i,j)$$

L_p Model

Examples

$$p = 1 : \quad h_{ijk} = \frac{1}{3} (g_{ij} + g_{jk} + g_{ki})$$

$$p = 2 : \quad h_{ijk}^2 = \frac{1}{9} (g_{ij}^2 + g_{jk}^2 + g_{ki}^2)$$

$$p = \infty : \quad h_{ijk} = \frac{1}{3} \max(g_{ij}, g_{jk}, g_{ki})$$

Clique Averaging

$$h(z) = \binom{k}{2}^{-1} \sum_{i,j \in z} g(i,j)$$

$$h = \lambda \Delta g$$

Clique Averaging

$$h(z) = \binom{k}{2}^{-1} \sum_{i,j \in z} g(i,j)$$

$$h = \lambda \Delta g$$

$$\Delta = \left[\quad \right]^{n \choose k}$$

Edge-hyperedge incidence

Sparse 0/1 matrix
Constant row sum
Constant column sum

Duality

Clique Averaging

$$\lambda \Delta g = h$$

Clique Expansion

$$g^e(i, j) = \mu \sum_{i, j \in z} h(z)$$

$$g^e = \mu \Delta^\top h$$

Duality

Clique Averaging

$$\Delta g = (1/\lambda) h$$

Clique Expansion

$$\Delta g^e = \mu \Delta \Delta^\top h$$

$$\Delta \Delta^\top = \begin{bmatrix} & \\ & \end{bmatrix} \begin{bmatrix} & \\ & \end{bmatrix} = \begin{bmatrix} & \\ & \end{bmatrix}$$

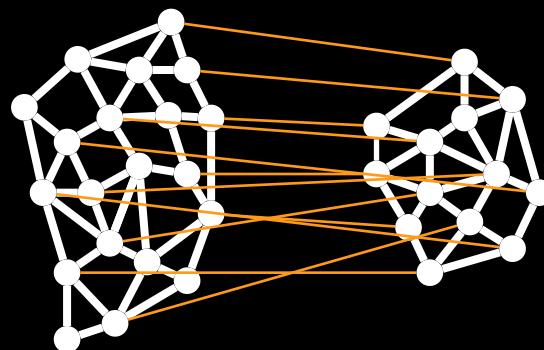
$\Delta \Delta^\top$ is a low pass filter = Loss of information !

So we have a graph...

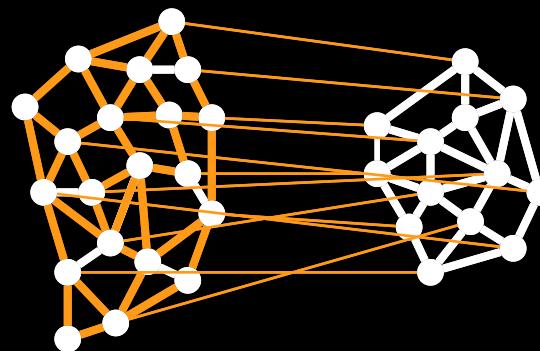
Normalized Cuts

$$\text{Ncut}(A, B) = \frac{\text{cut}(A, B)}{\text{Assoc}(A)} + \frac{\text{cut}(A, B)}{\text{Assoc}(B)}$$

$$\text{cut}(A, B) = \sum_{u \in A, v \in B} w(u, v)$$



$$\text{assoc}(A) = \sum_{u \in A, v} w(u, v)$$

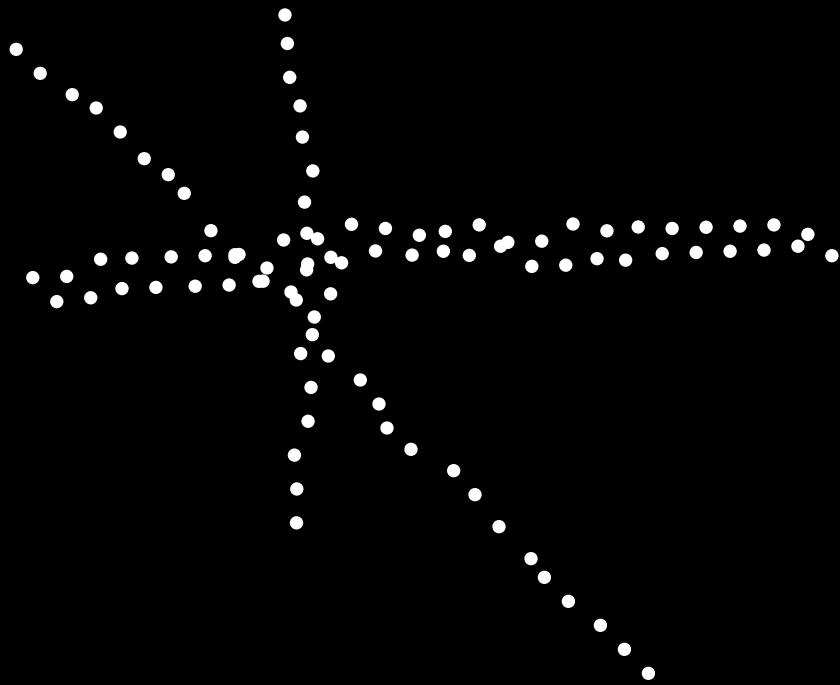


Putting it all together...

1. Sample k-tuples in the data to construct a hypergraph H.
2. Construct approximate graph G from H.
3. Partition G using Normalized Cuts.

Experiments

k-Lines



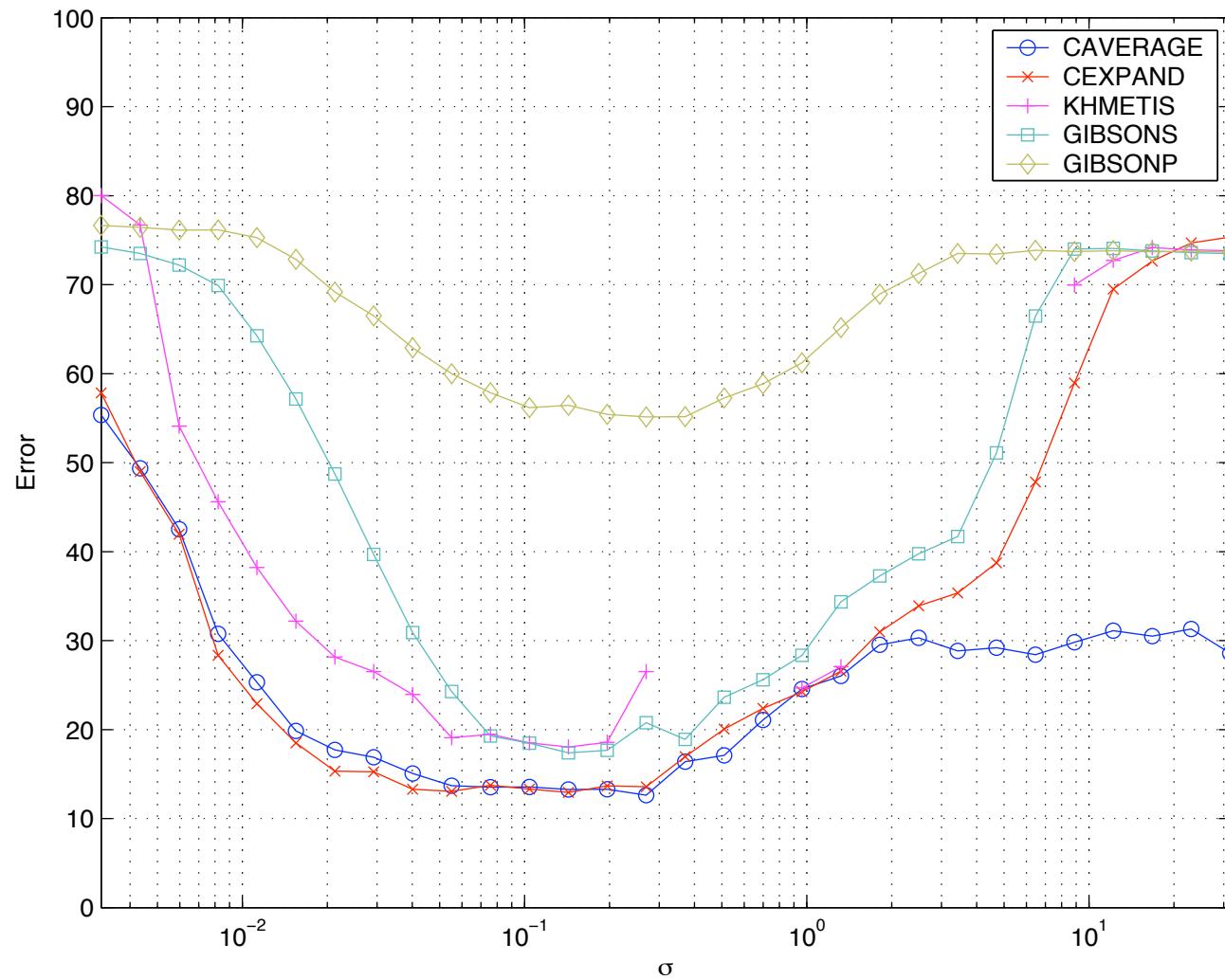
$$h(z) = e^{-\lambda_2/\sigma}$$

k-Lines

AVERAGE	12.6
EXPAND	12.9
GIBSONS	17.3
GIBSONP	55.1
KHMETIS	18.0
CRANSAC	23.4

percent error

k-Lines

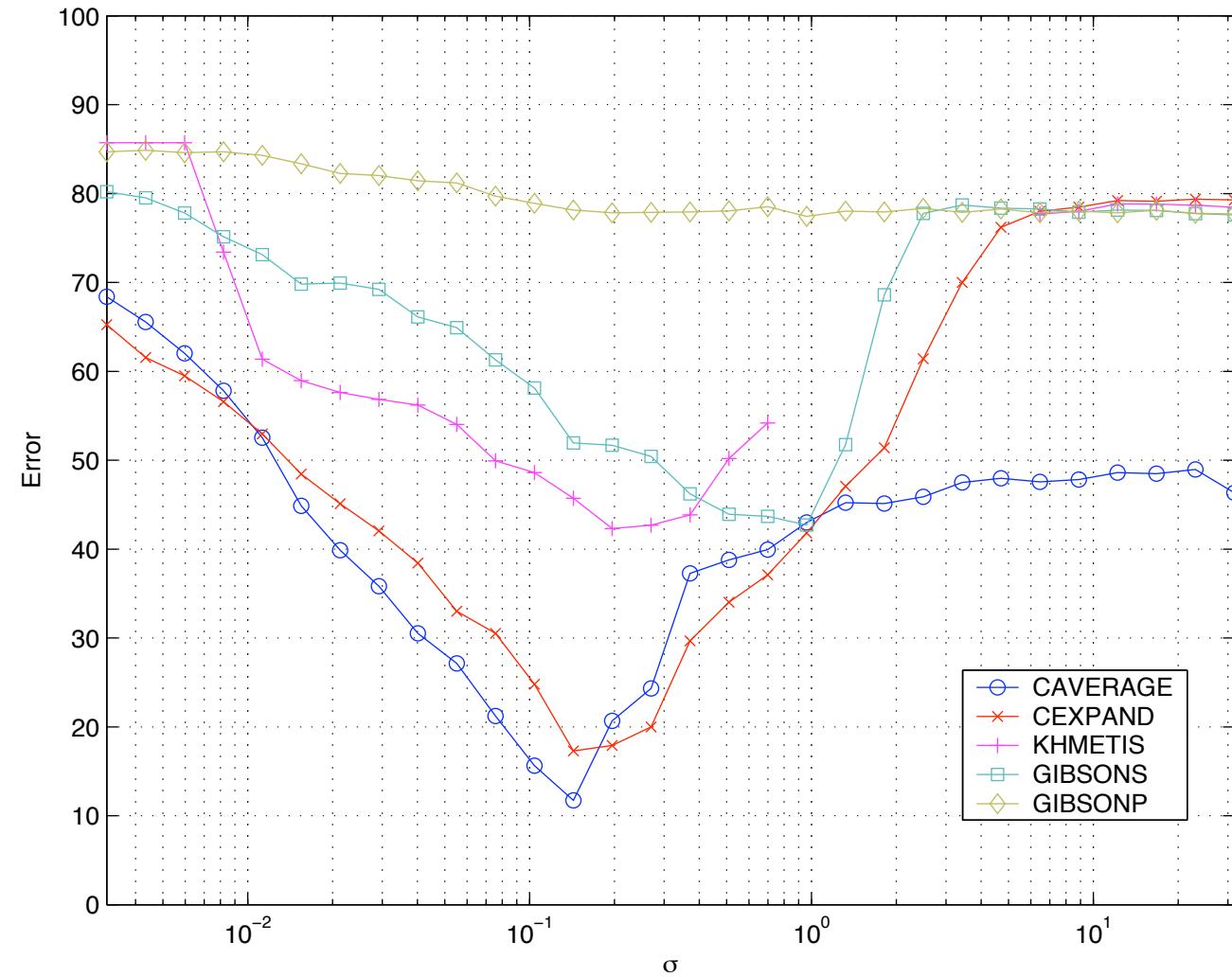


Yale Face Dataset



- 45 Images per person
- Varying illumination $h(z) = e^{-\lambda_4/\sigma}$
- 10 Individuals

Faces



Face Clustering

	4	6	8	10
AVERAGE	4.2 (6.3)	12.7(8.4)	17.4(4.0)	16.0(3.0)
EXPAND	11.8(3.4)	17.6(5.4)	21.8(5.4)	24.9(4.3)
GIBSONS	25.9(7.3)	42.2(3.8)	47.7(3.0)	51.5(2.1)
GIBSONP	67.4(2.3)	75.2(1.2)	79.7(0.8)	82.8(0.7)
KHMETIS	21.5(4.3)	41.9(6.8)	38.4(4.7)	58.3(3.3)
CRANSAC	16.2(9.5)	23.6(9.2)	35.1(7.9)	37.1(6.6)

percent error (std)

Future Work

- Low rank approximations to G
- Sampling bounds
- Better generative models
- Scaling it to VLSI CAD scale problems.

Summary

- A generative model based graph approximation.
- Provably better than Clique Expansion.
- Better empirical performance than existing algorithms.

Acknowledgments

- F. Chung, S. Dasgupta, C. Donner, H. W. Jensen, A. Kahng
- LLNL UCRP (DOE) No. W-7405-ENG-48
- NSF CAREER #0448615
- Alfred P. Sloan Research Fellowship
- Calit2



UCSD
Jacobs

Computer Science
and Engineering

Questions?