Meta-Clustering

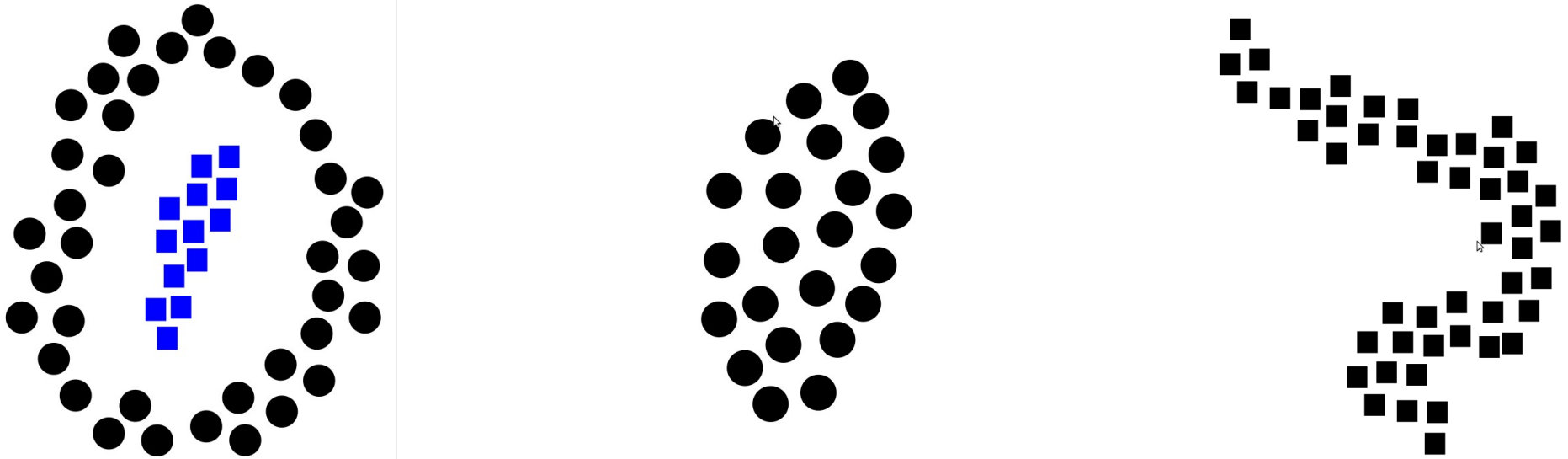
Parasaran Raman
PhD Candidate
School of Computing

What is Clustering?

- Goal: Group similar items together
- Unsupervised – No labeling effort
- Popular choice for large-scale exploratory data analysis
- Many algorithms to find the “right” clustering

What is “right”?

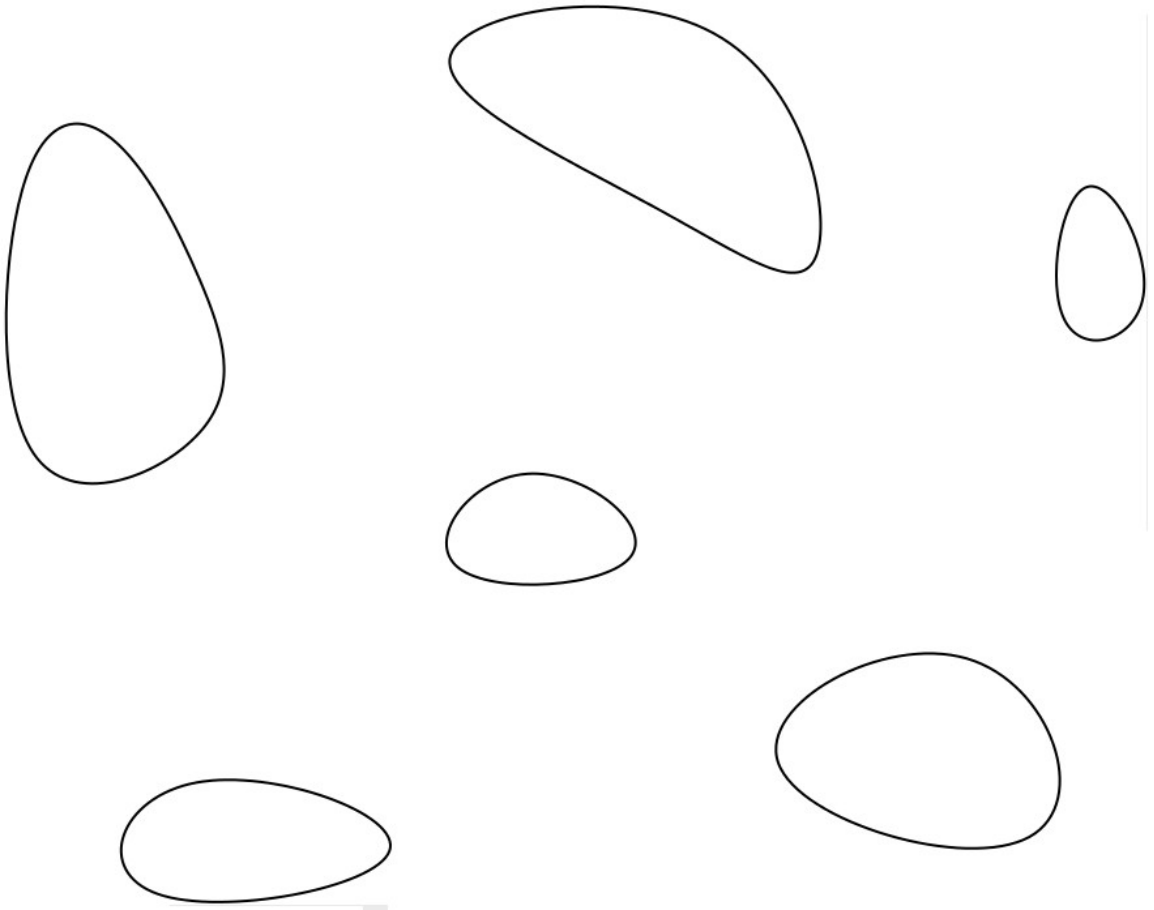
“Right” kind of structures in data



Different shapes present in the data

What is “right”?

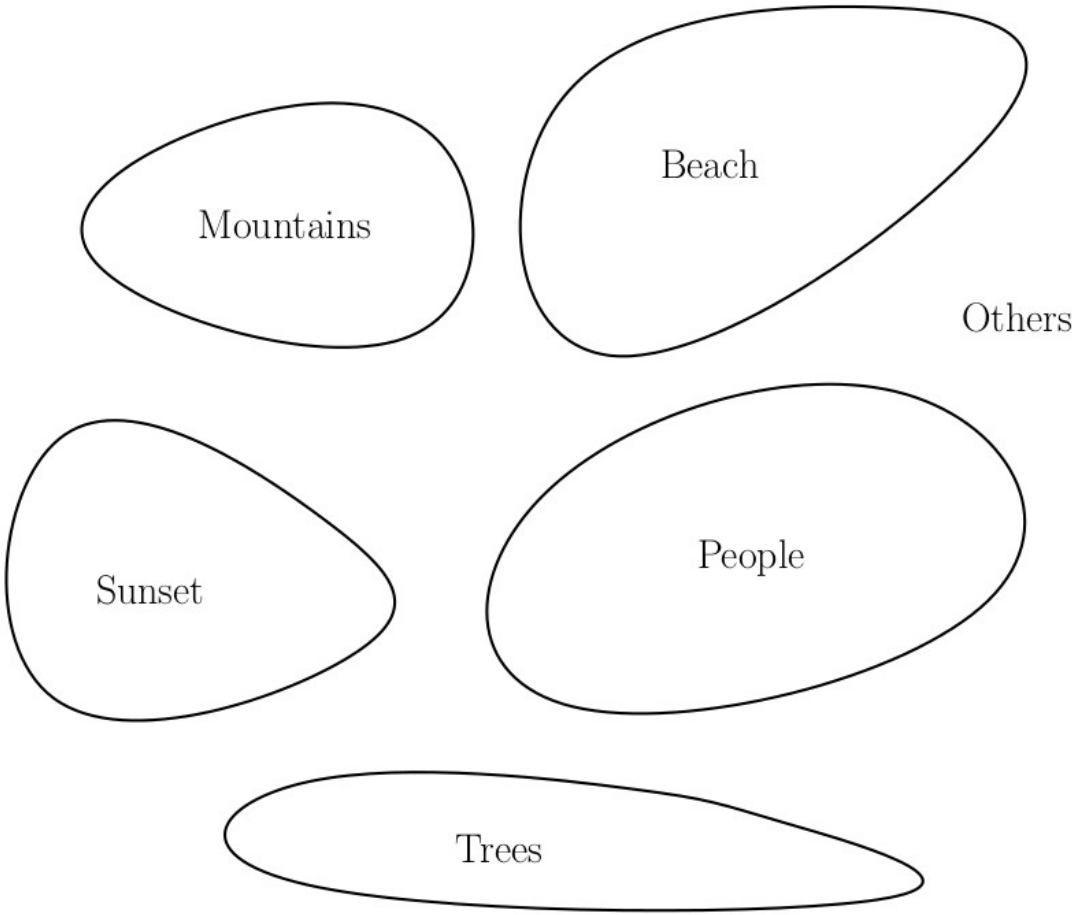
“Right” according to a quality measure



Tight chunks of far-away clusters

What is “right”?

“Right” for the user



Meaningful labels to each cluster

Meta-Questions on Clusterings

- Can we learn better by integrating different clustering techniques?
- Will a particular method will be successful against a specific kind of data?
- Can we evaluate the quality of the clustering?
- Can we compare the results of two clustering methods?

The What?

- Systematic study of various metaclustering problems
  - Organize different approaches to clustering data for robust data analysis
- Raw data → Clustering → Other data analysis activities
  - Important early step in data exploration
- Look at clustering as an exploratory tool
  - organizing clusterings in a way that helps us form robust opinions about the clusterings and the data
  - When faced with new kinds of data, we are unsure about the nature of structures present in it

The Why?

- Hard to find a clustering method that would cluster all kinds of data
  - according to any specific criterion (i.e. shape or color)
  - unless it is explicitly incorporated into the optimization objective
- A typical clustering method would construct a model that provides some signal both about the instances' shapes
  - For example, averaging such models might be useful
- Wide range of applications in the biology realm
  - Clustering Gene expression
  - Protein Sequences

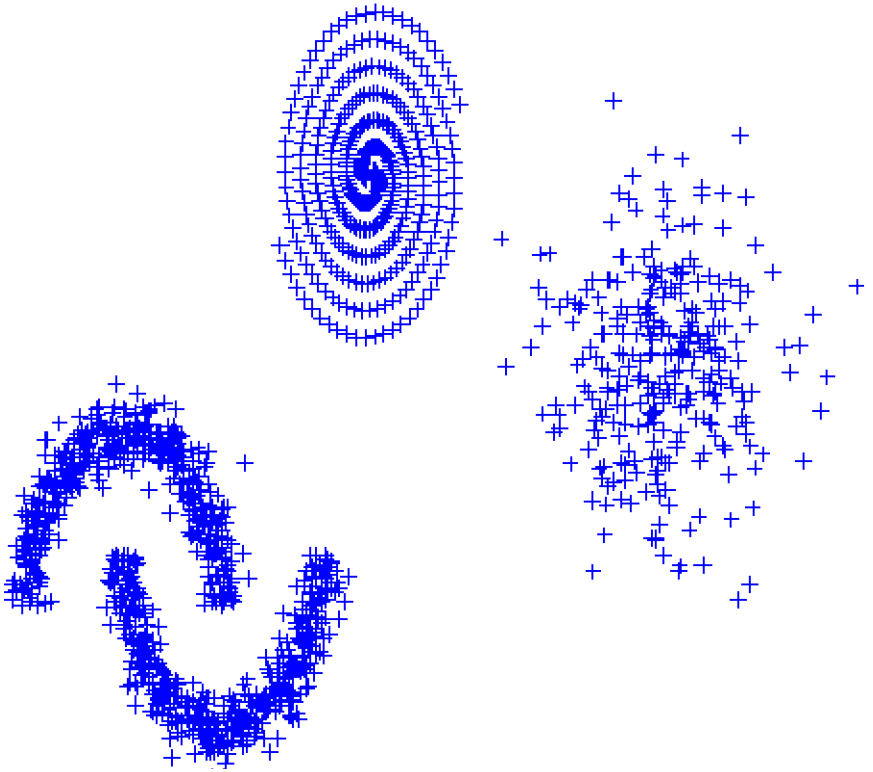
Topic 1 : Consensus Clustering

Spatially-Aware Comparison and Consensus for Clusterings

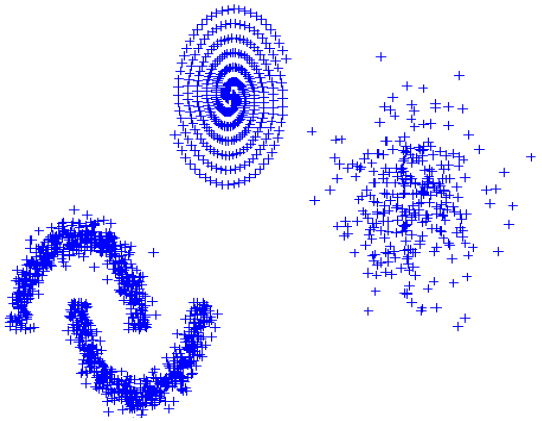
Siam Data Mining (SDM 2011)

Joint work with:
Jeff M. Phillips, *University of Utah*
Suresh Venkatasubramanian, *University of Utah*

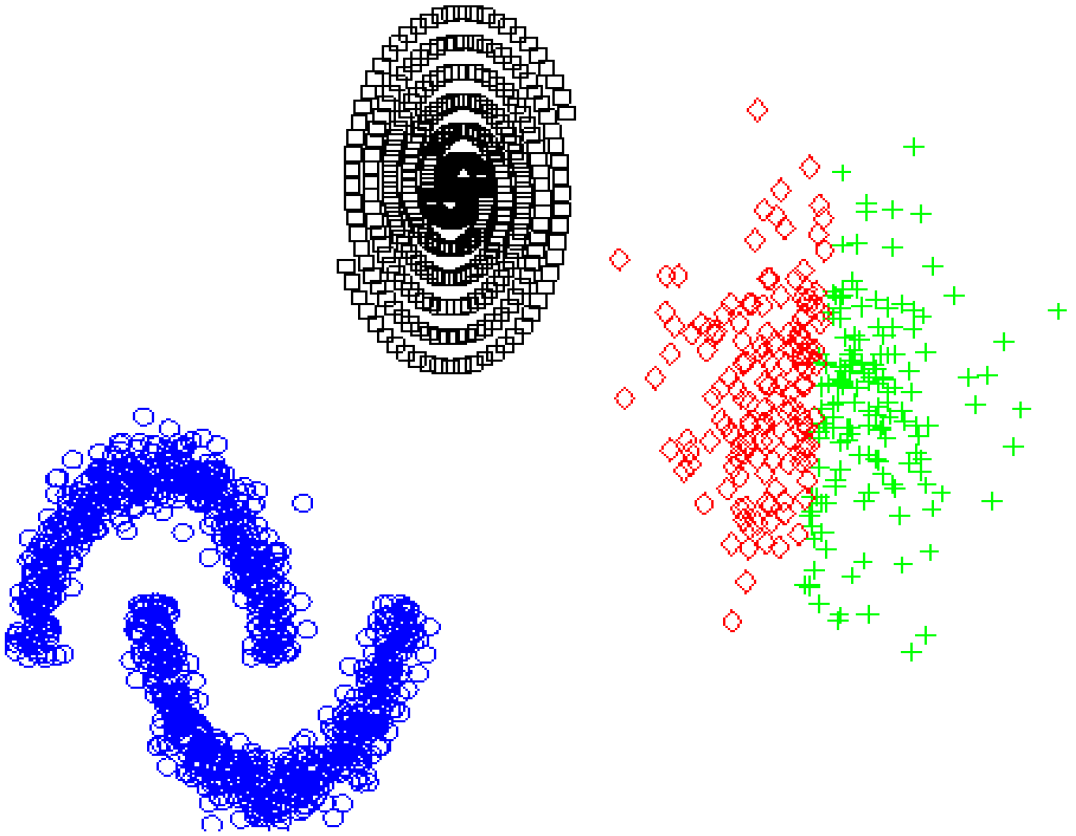
Clustering



Clustering

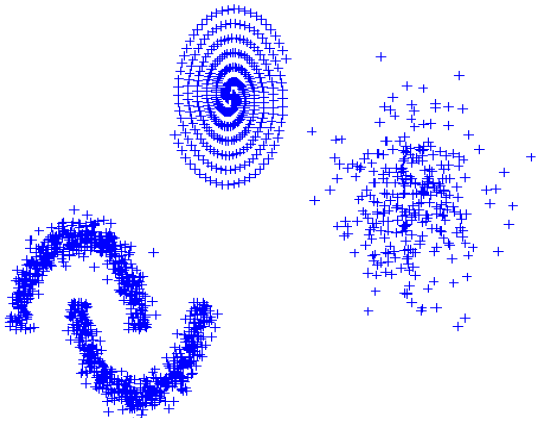


Data

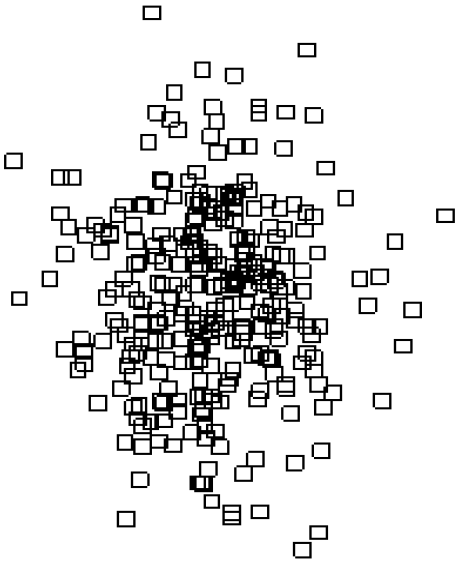


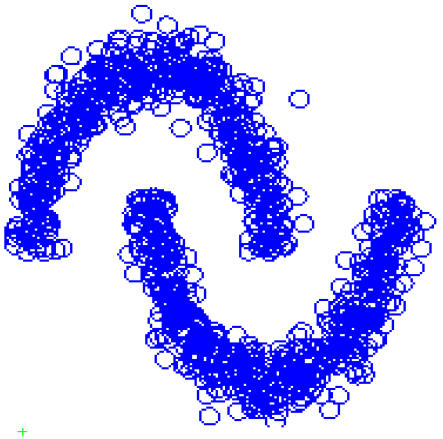
KMeans

Clustering

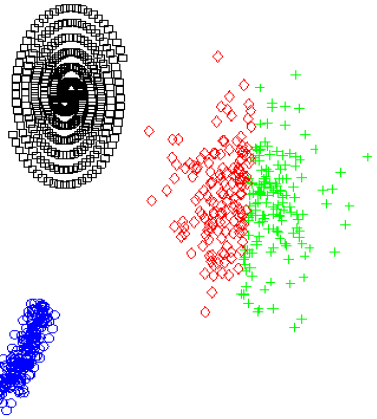


Data



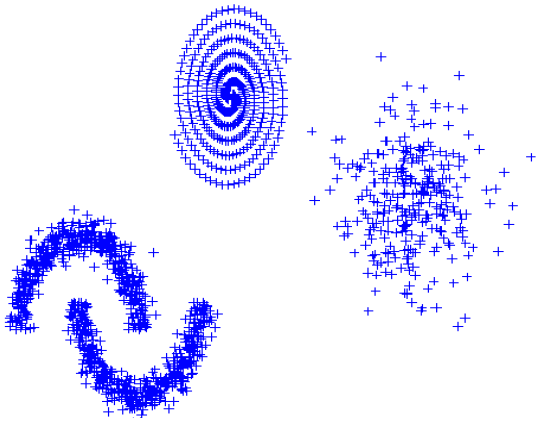


DBSCAN

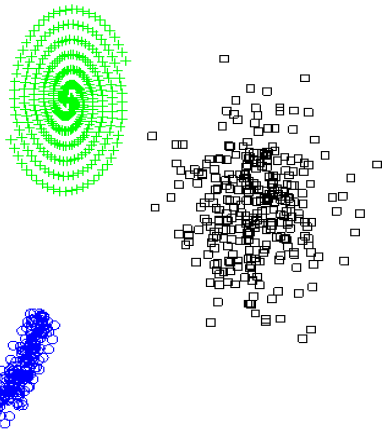


Kmeans

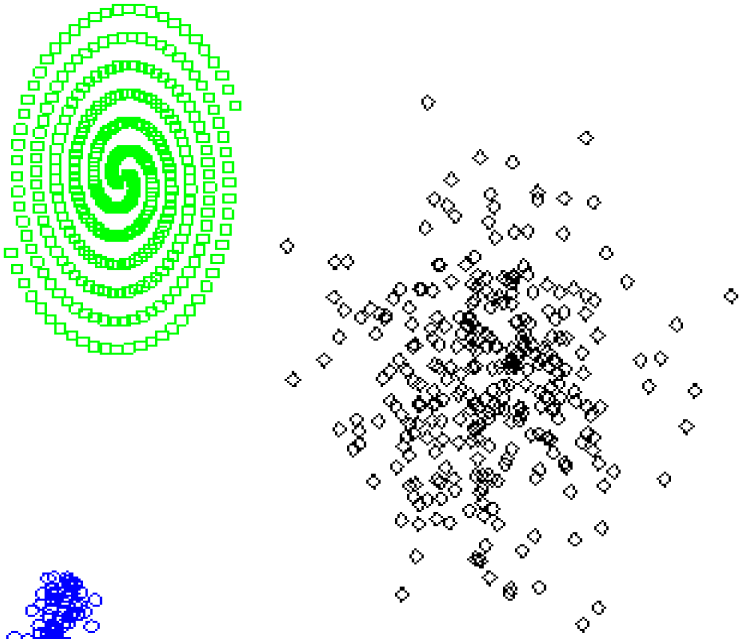
Clustering



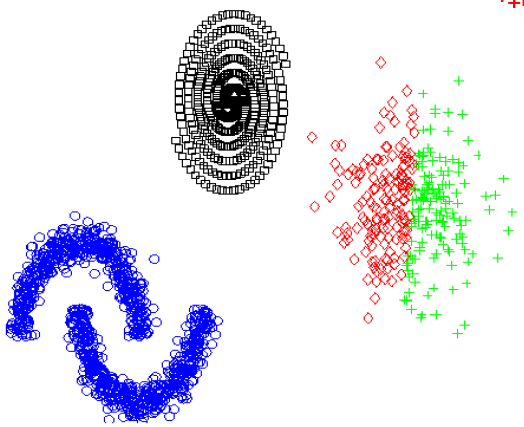
Data



DBSCAN

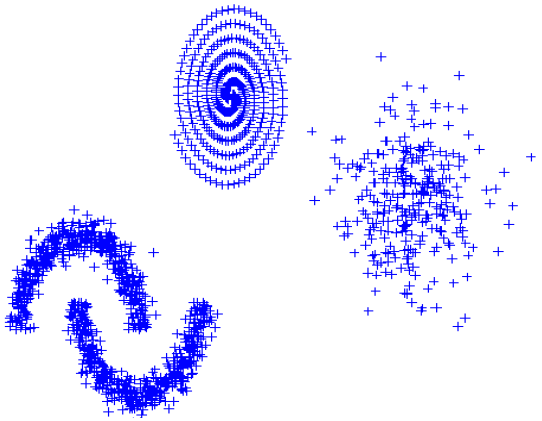


Complete Linkage HAC

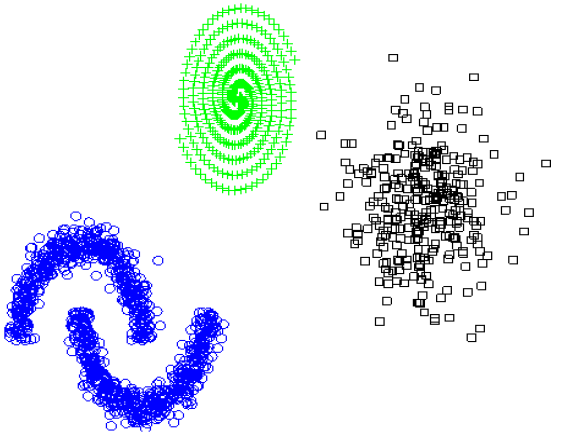


Kmeans

Clustering



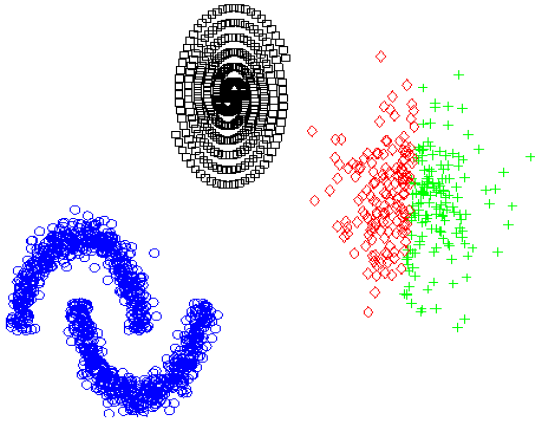
Data



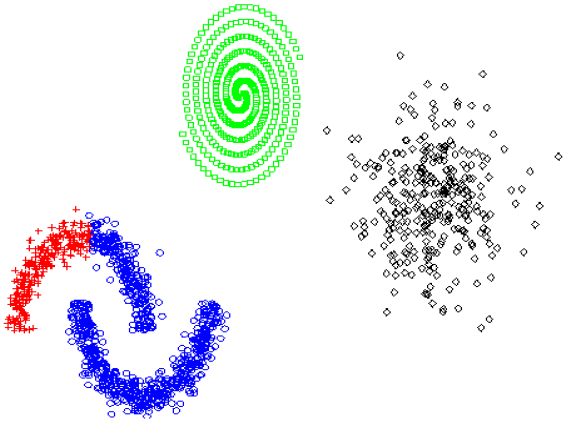
DBSCAN

Different clustering methods output different partitions!

Which method do I pick?

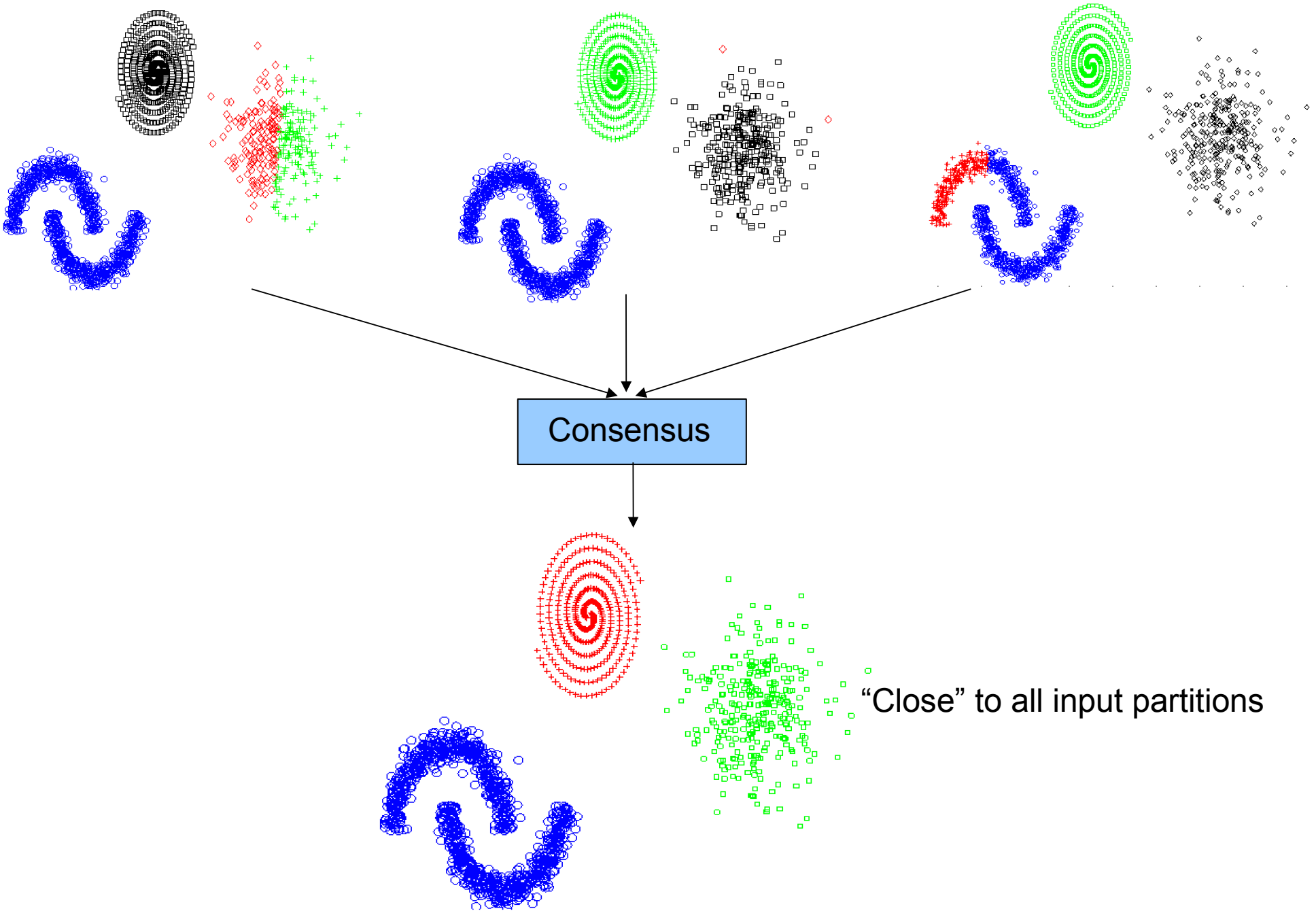


Kmeans



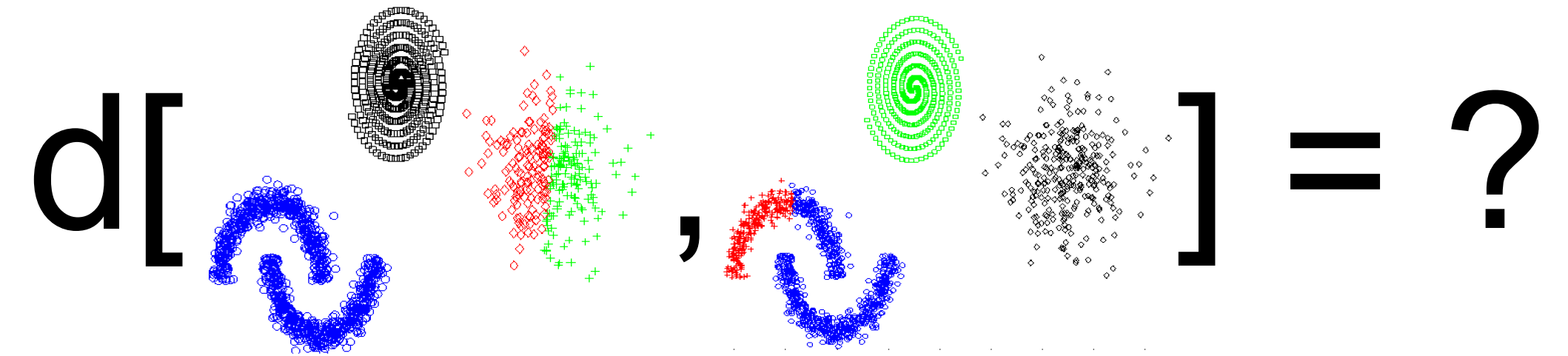
Complete Linkage HAC

Reconcile!



"Close" to all input partitions

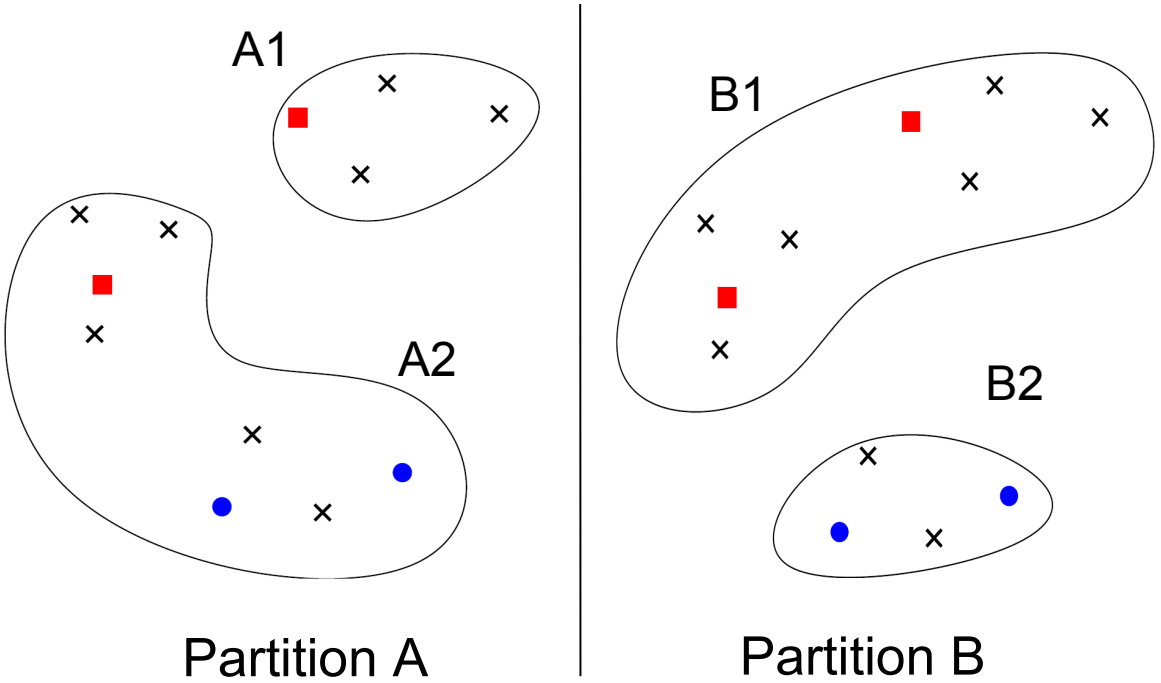
What is “close”?



Comparing Partitions: Combinatorial

- What objects are clustered together?

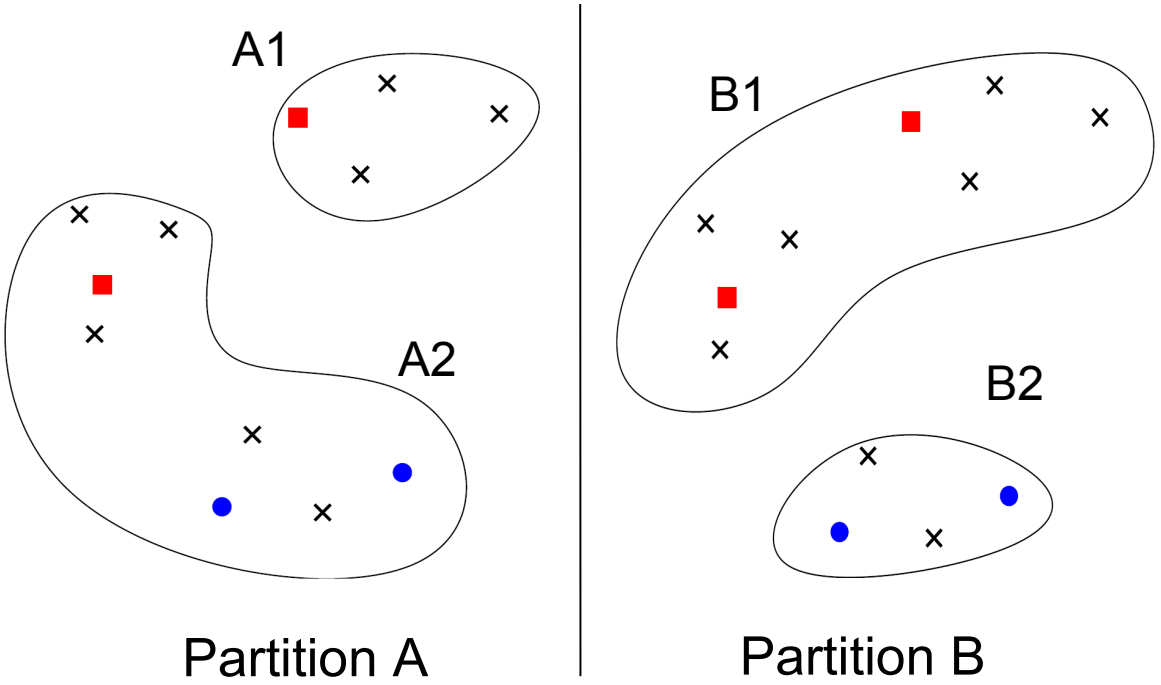
A \ B	Same Cluster	Different Cluster
Same Cluster	$N_{00} = 18$	$N_{01} = 16$
Different Cluster	$N_{10} = 16$	$N_{11} = 16$



Comparing Partitions: Combinatorial

- What objects are clustered together?

A \ B	Same Cluster	Different Cluster
Same Cluster	$N_{00} = 18$	$N_{01} = 16$
Different Cluster	$N_{10} = 16$	$N_{11} = 16$



$$\textit{Rand Index} = \frac{N_{00} + N_{11}}{\text{nC2}}$$

$$= 0.515$$

Variance of Information (VI)

Normalized Mutual Information

Comparing Partitions: Spatial

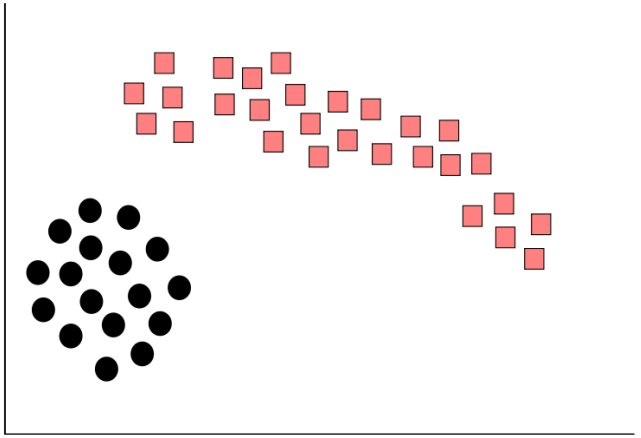
- How compact are the clusters?

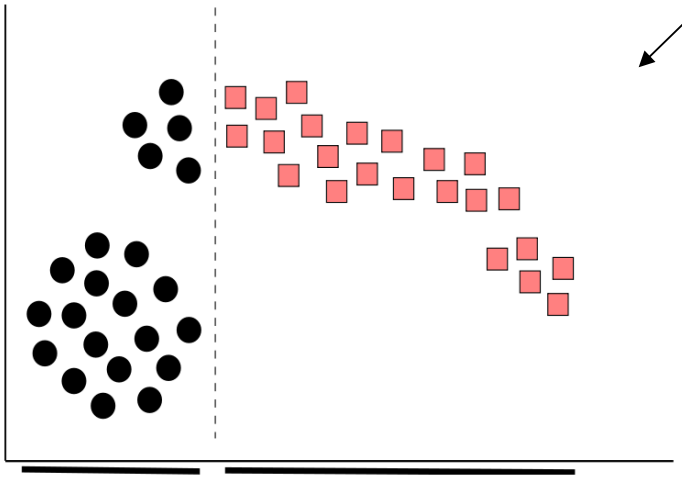
CDistance *[Coen, Ansari, Fillmore]*

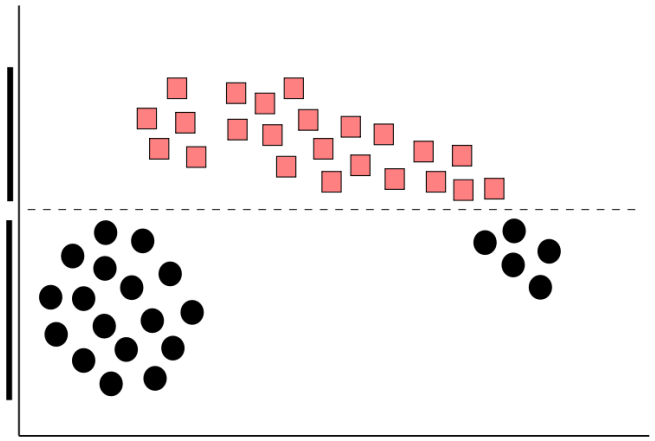
CC *[Zhou, Li, Zha]*

D_{ADCO} *[Bae, Bailey, Dong]*

Reference Partition (RP)







First Partition (FP)

Second Partition (SP)

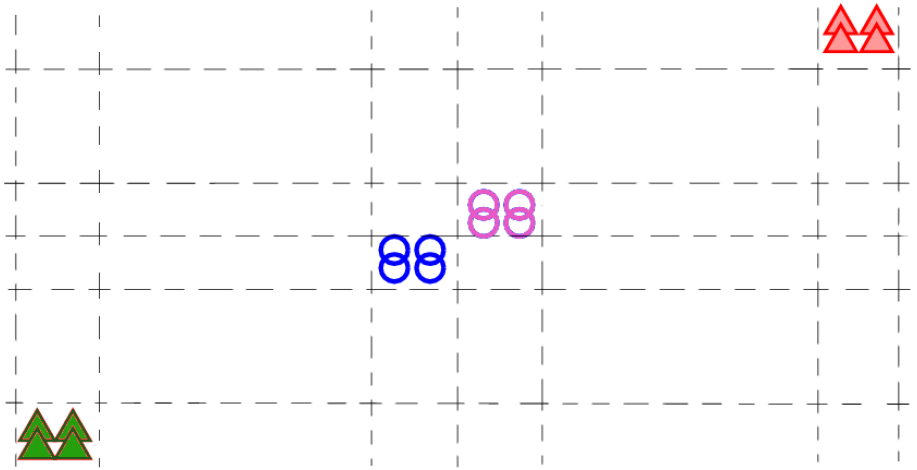
Current Issues in Spatial-Awareness

- CDistance *[Coen et. al.]*
  - Earth Mover's between clusters
  - Expensive: $O(n^3)$

Current Issues in Spatial-Awareness

- CDistance *[Coen et. al.]*
  - Earth Mover's between clusters
  - Expensive: $O(n^3)$

- D_{ADCO} *[Bae et. al.]*
  - Binning & Histogram

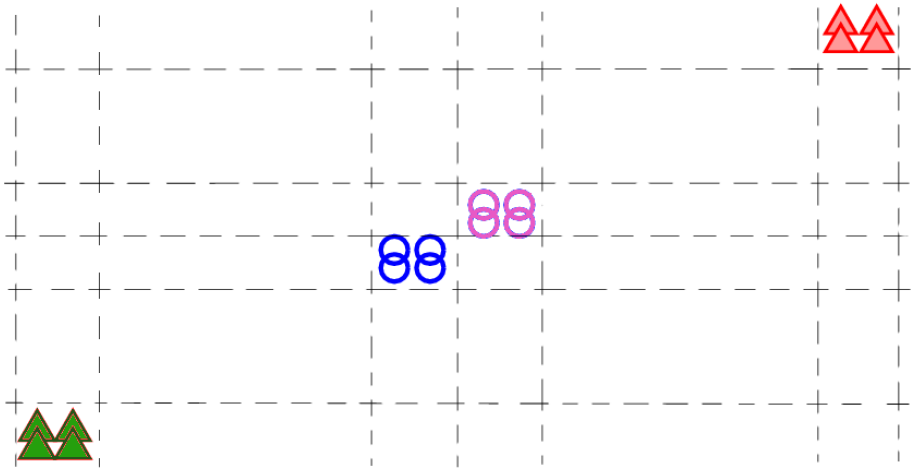


Current Issues in Spatial-Awareness

- CDistance *[Coen et. al.]*
  - Earth Mover's between clusters
  - Expensive: $O(n^3)$

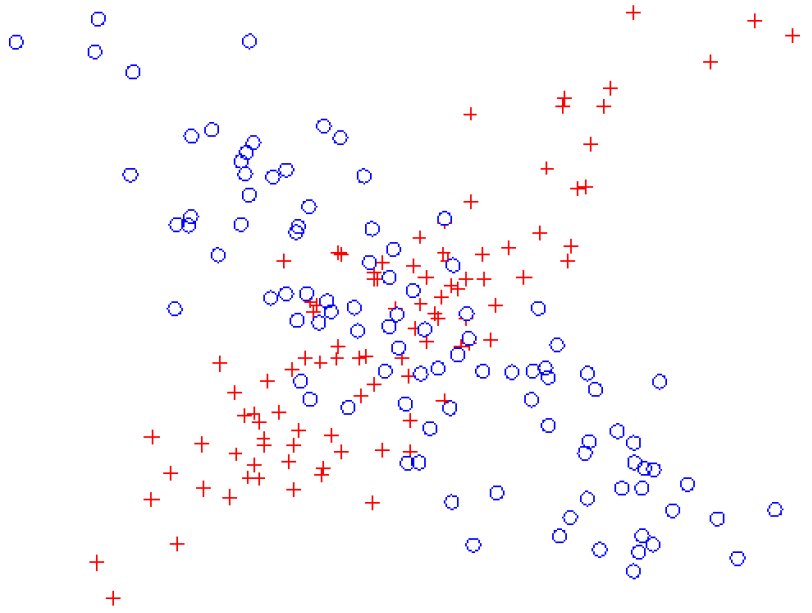
- D_{ADCO} *[Bae et. al.]*
  - Binning & Histogram

d[,] = d[,]



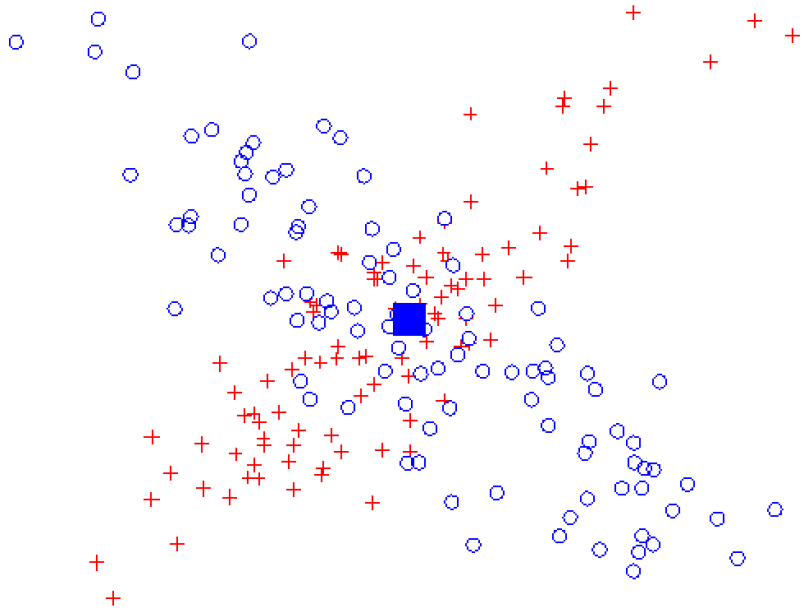
Current Issues in Spatial-Awareness

- CC *[Zhou et. al.]*
  - Measures distance between centroids
  - Lossy



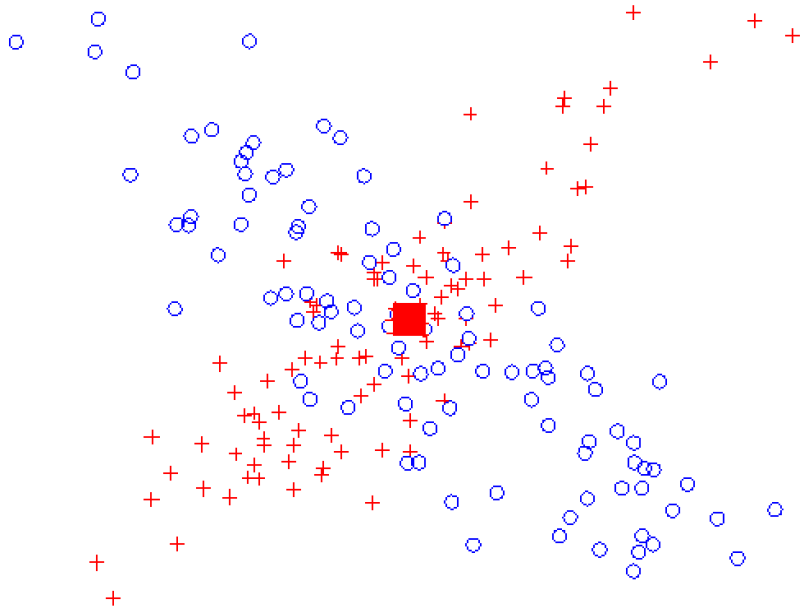
Current Issues in Spatial-Awareness

- CC *[Zhou et. al.]*
  - Measures distance between centroids
  - Lossy



Current Issues in Spatial-Awareness

- CC *[Zhou et. al.]*
  - Measures distance between centroids
  - Lossy



Contributions

- LiftEMD
  - Distance metric between partitions
  - Spatial and combinatorial

Contributions

- LiftEMD
  - Distance metric between partitions
  - Spatial and combinatorial

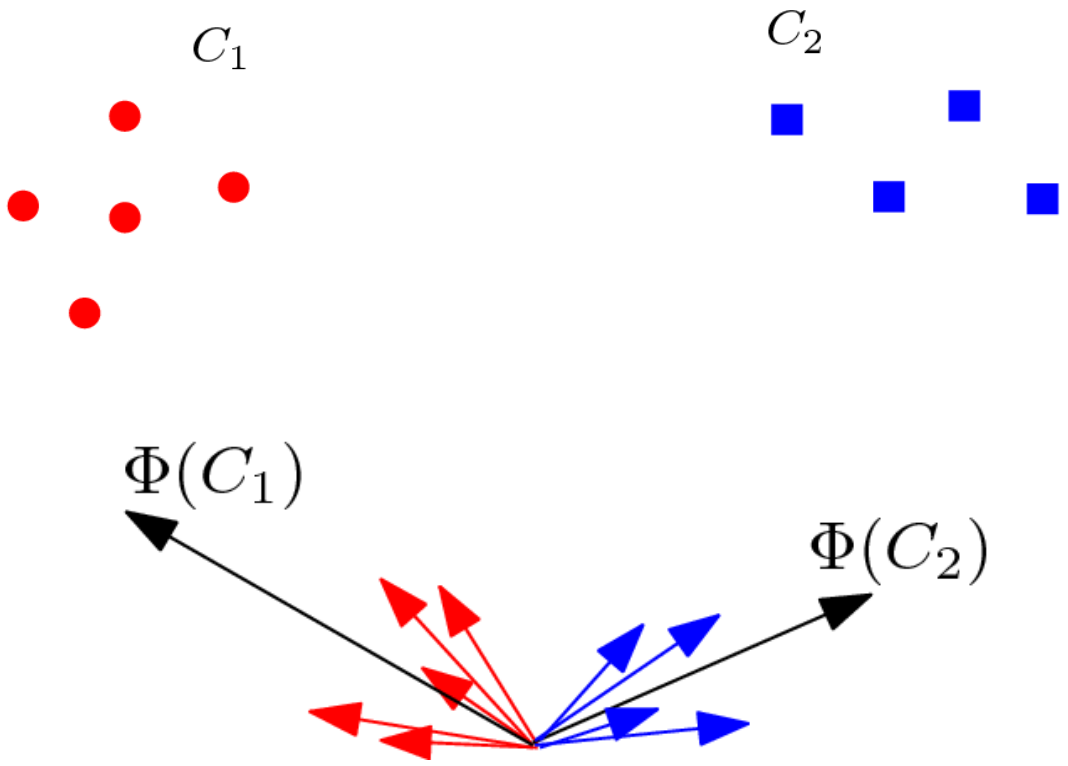
- Consensus
  - Reduce problem to simple Euclidean clustering
  - Fast
  - Can handle large data
  - Spatially-Aware

Key Idea

- Signature for clusters
  - A point set can be losslessly mapped to a target feature space
  - Cluster representative: Sum of feature maps of points

Key Idea

- Signature for clusters
  - A point set can be losslessly mapped to a target feature space
  - Cluster representative: Sum of feature maps of points

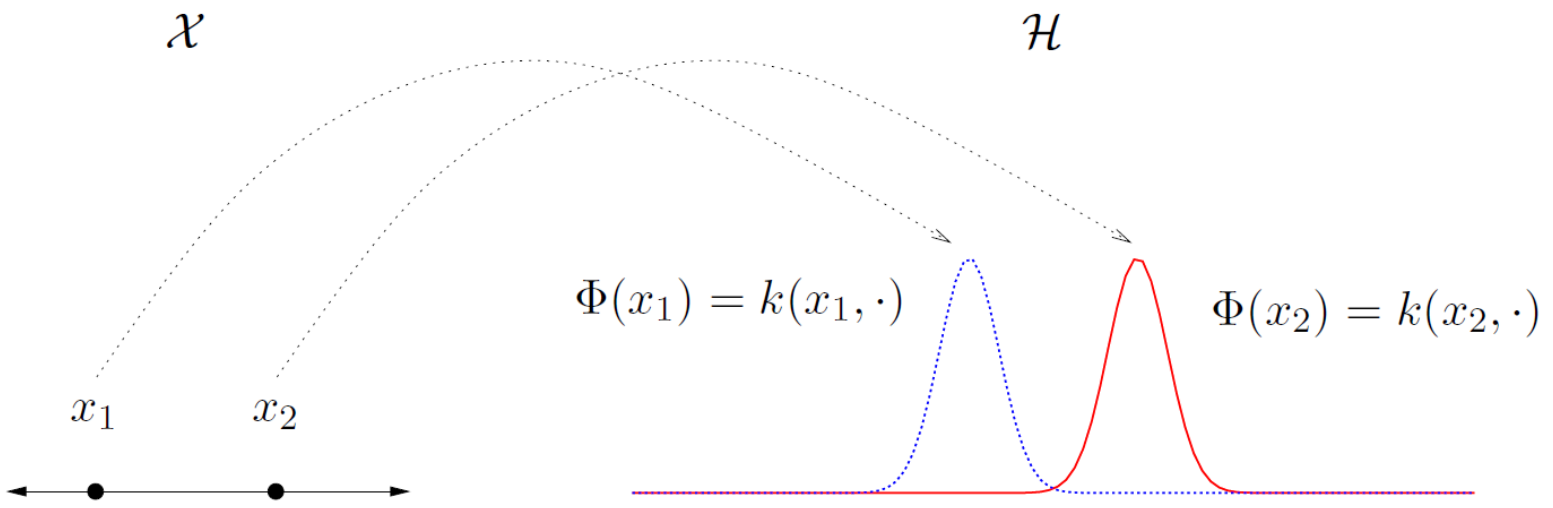


Reproducing Kernel Hilbert Space

- Reproducing kernel $\kappa : X \times X \to \mathbb{R}$
  - Induces a Hilbert space $\mathcal{H}_\kappa$

- Lifting Map $\Phi : X \to \mathcal{H}_\kappa$
  - Property: $\kappa(x, y) = \langle \Phi(x), \Phi(y) \rangle_\kappa$

$\Phi(p) = \int_X \kappa(\cdot, x) dp(x)$



Courtesy: Christian Igel

Reproducing Kernel Hilbert Space

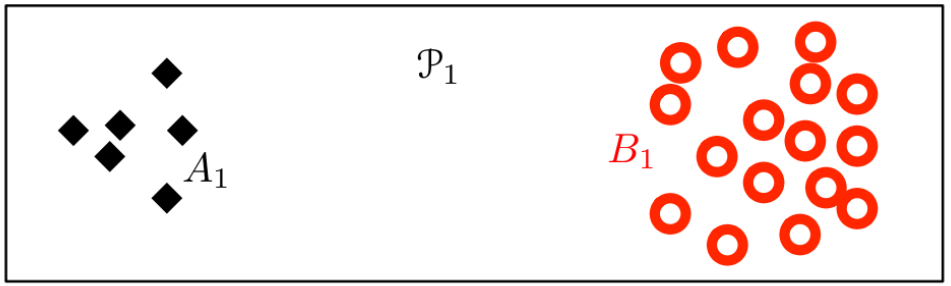
- Reproducing kernel $\kappa : X \times X \to \mathbb{R}$
  - Induces a Hilbert space $\mathcal{H}_\kappa$

- Lifting Map $\Phi : X \to \mathcal{H}_\kappa$
  - Property: $\kappa(x, y) = \langle \Phi(x), \Phi(y) \rangle_\kappa$
    $\Phi(p) = \int_X \kappa(\cdot, x) dp(x)$

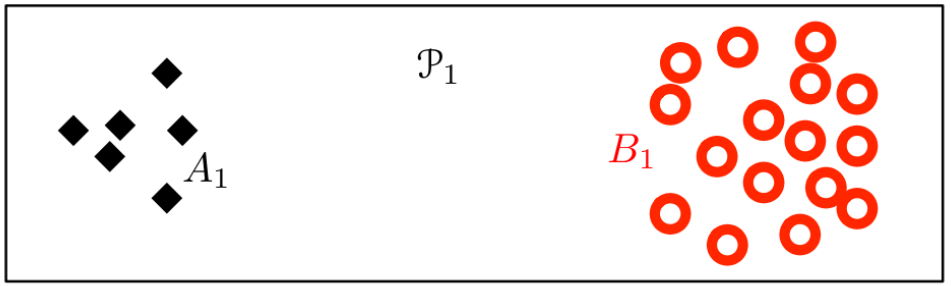
- Approximate Representation
  - Random feature $\tilde{\Phi} : X \times X \to \mathbb{R}^\rho$
  - Error: $\left| \|\tilde{\Phi}(x) - \tilde{\Phi}(y)\|_2 - \|\Phi(x) - \Phi(y)\|_{\mathcal{H}_\kappa} \right| \leq \varepsilon$

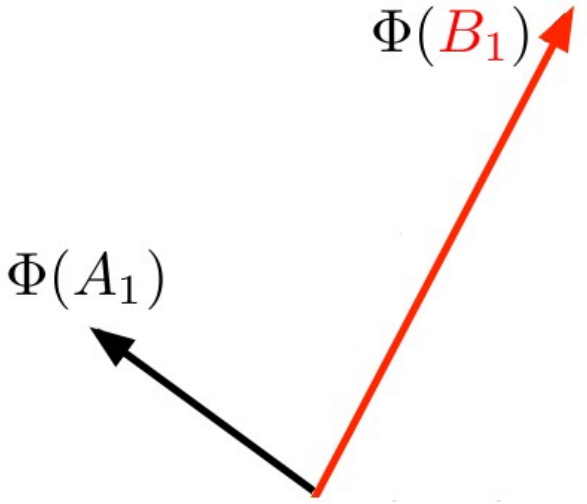
1. A. Rahimi, B. Recht, NIPS 2007
2. S. Joshi, R.V. Kommaraju, J.M. Phillips, S. Venkatasubramanian, SoCG 2011

LiftEMD: Comparing Partitions in RKHS



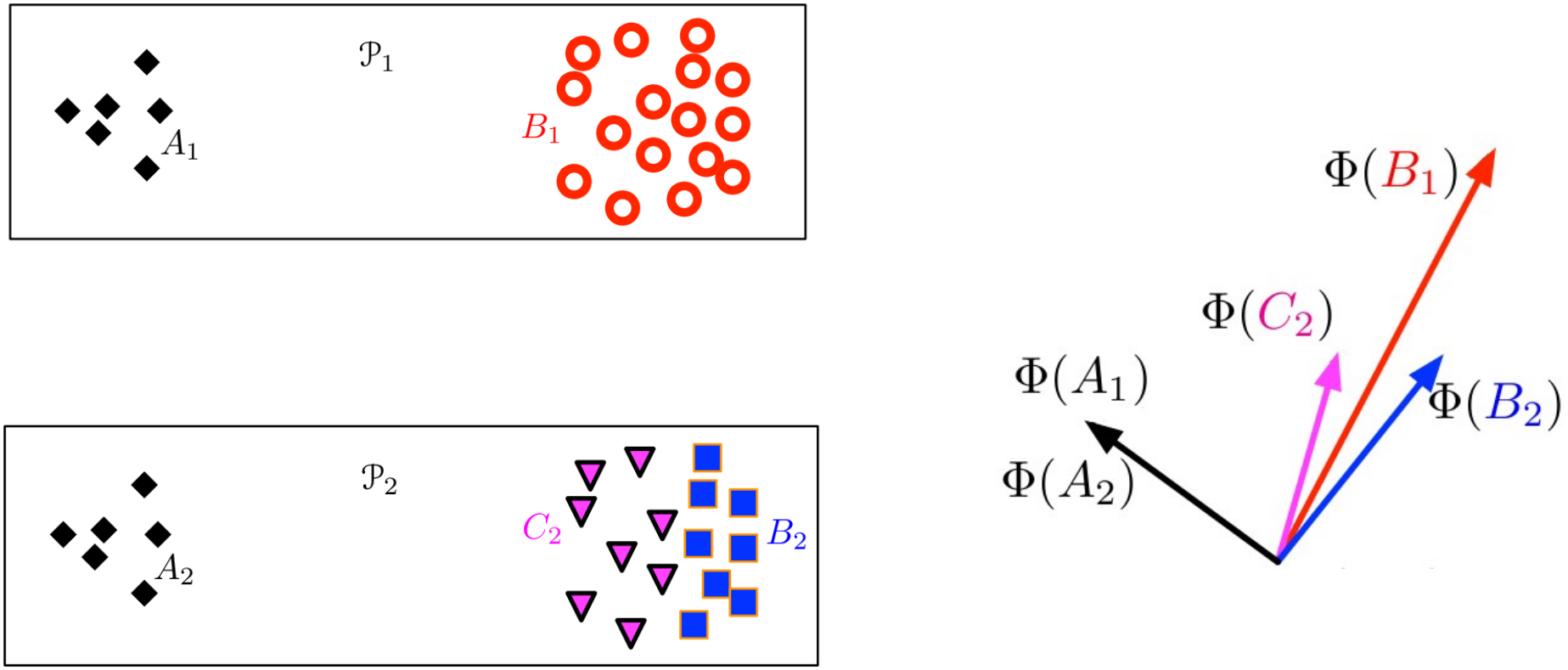
LiftEMD: Comparing Partitions in RKHS





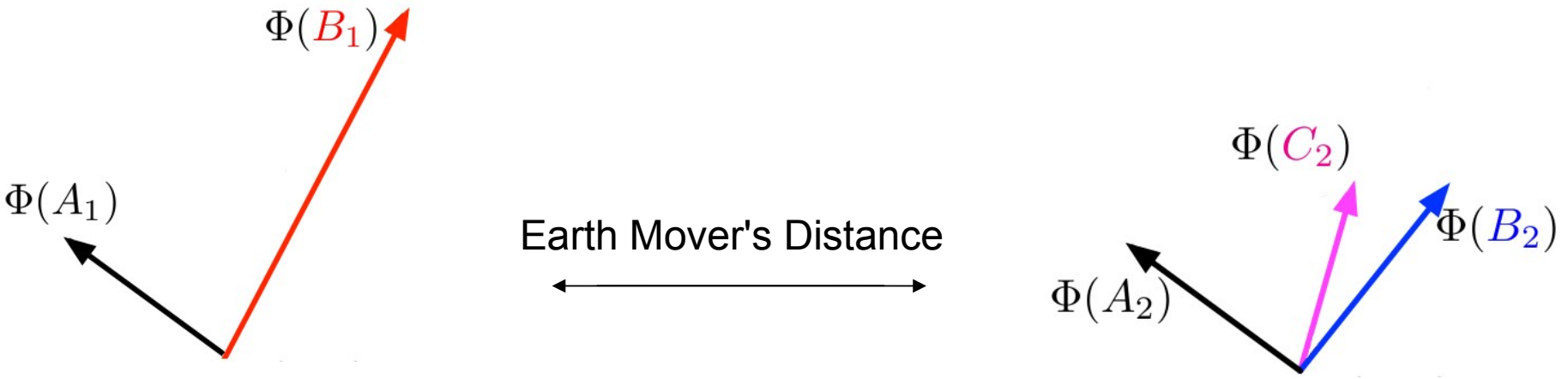
Step 1: Lift clusters to vectors in RKHS

LiftEMD: Comparing Partitions in RKHS



Step 1: Lift clusters to vectors in RKHS

LiftEMD: Comparing Partitions in RKHS



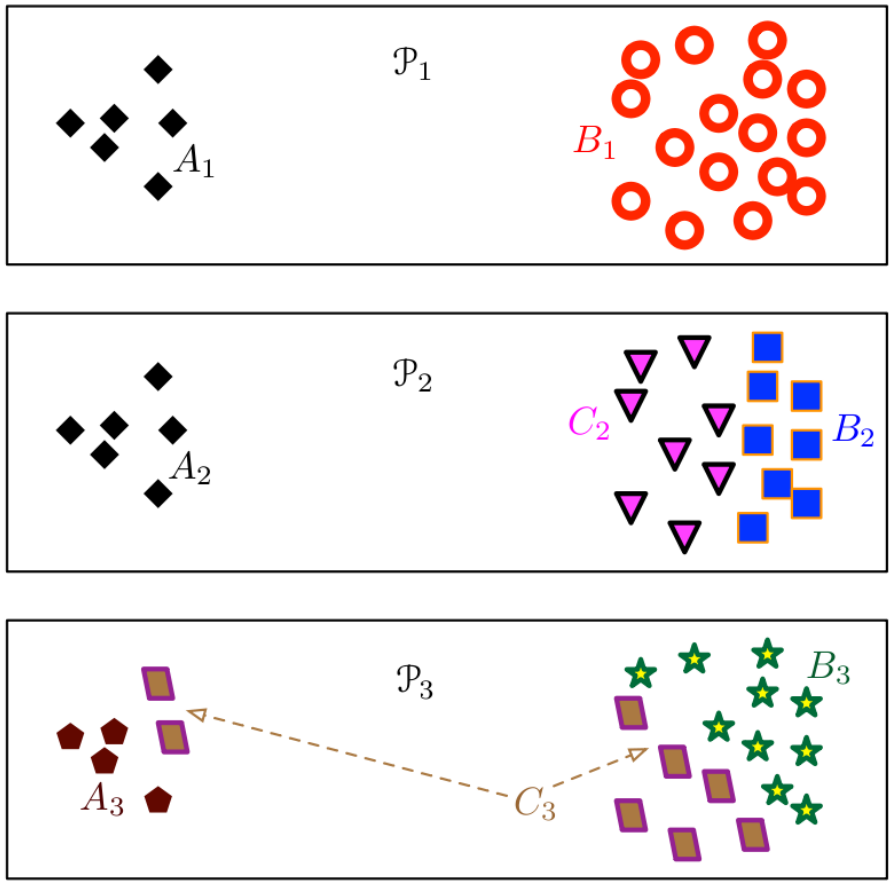
Step 2: EMD between RKHS vectors

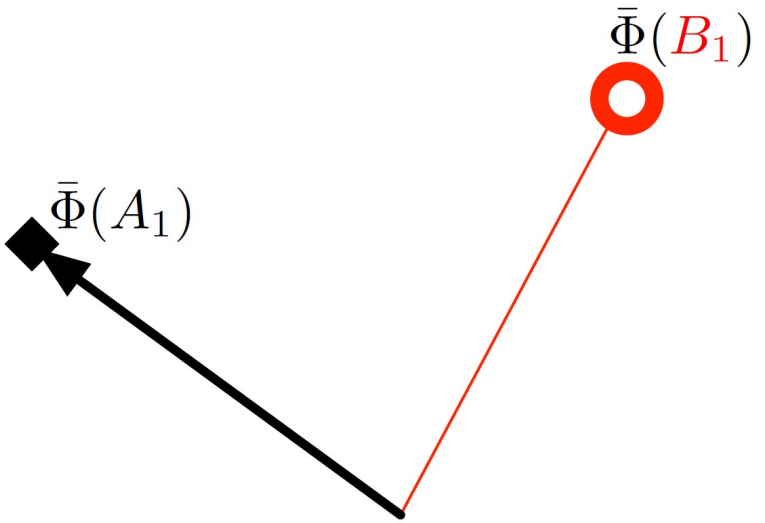
Consensus in RKHS

- Reduction to Euclidean clustering
  - Step 1: Lift clusters to vectors in RKHS
  - Step 2: Run any standard clustering algorithm on the vectors

Consensus in RKHS

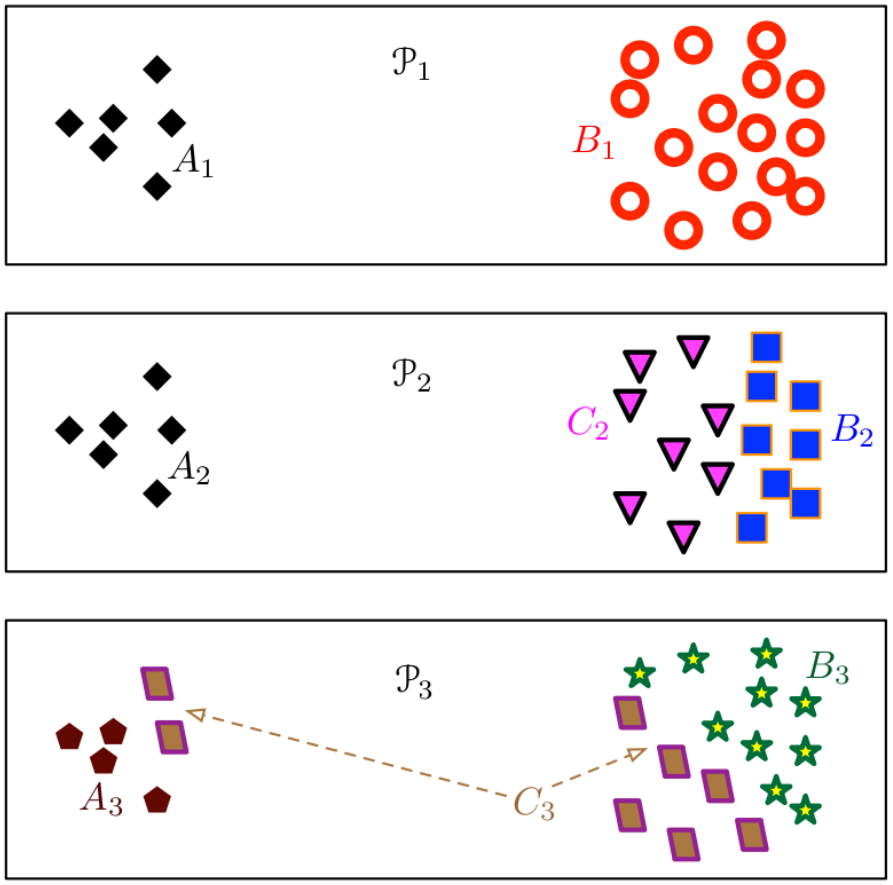
- Reduction to Euclidean clustering
  - Step 1: Lift clusters to vectors in RKHS
  - Step 2: Run any standard clustering algorithm on the vectors

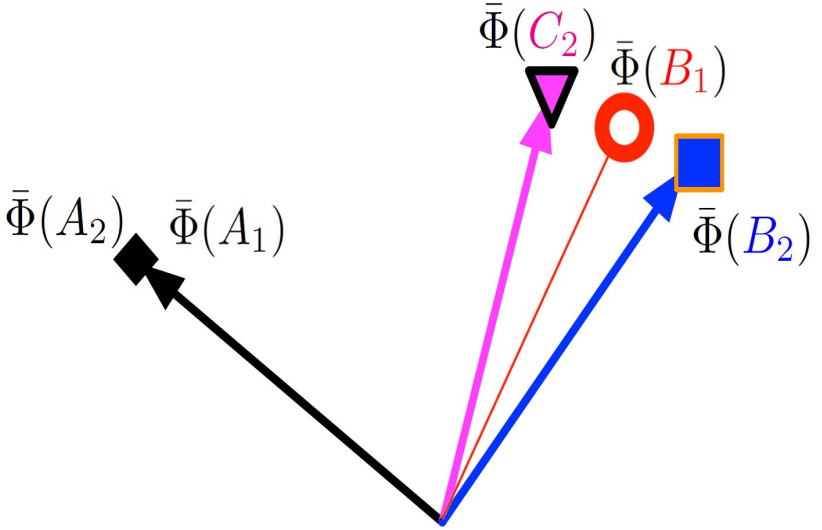




Consensus in RKHS

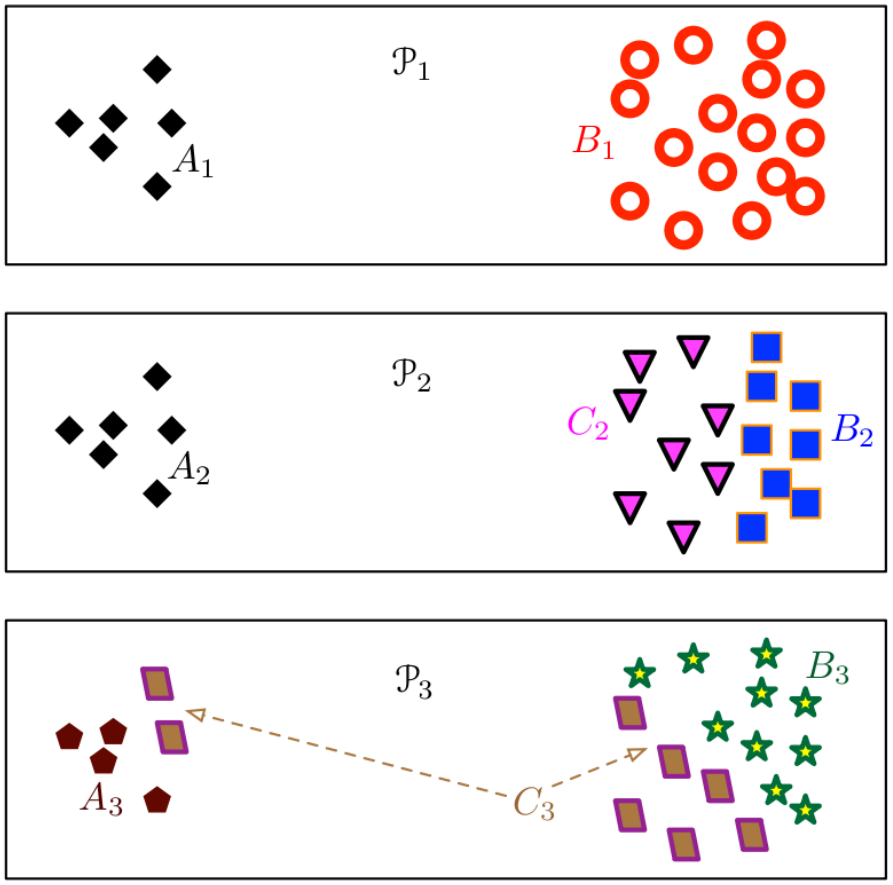
- Reduction to Euclidean clustering
  - Step 1: Lift clusters to vectors in RKHS
  - Step 2: Run any standard clustering algorithm on the vectors

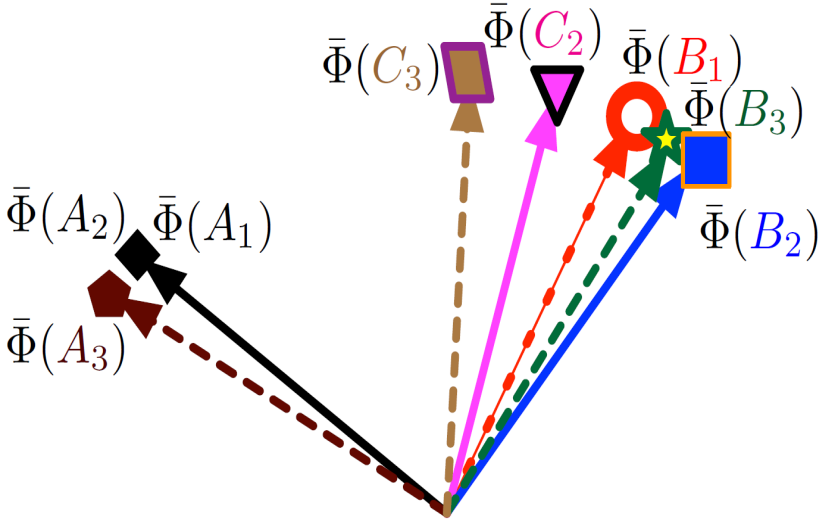




Consensus in RKHS

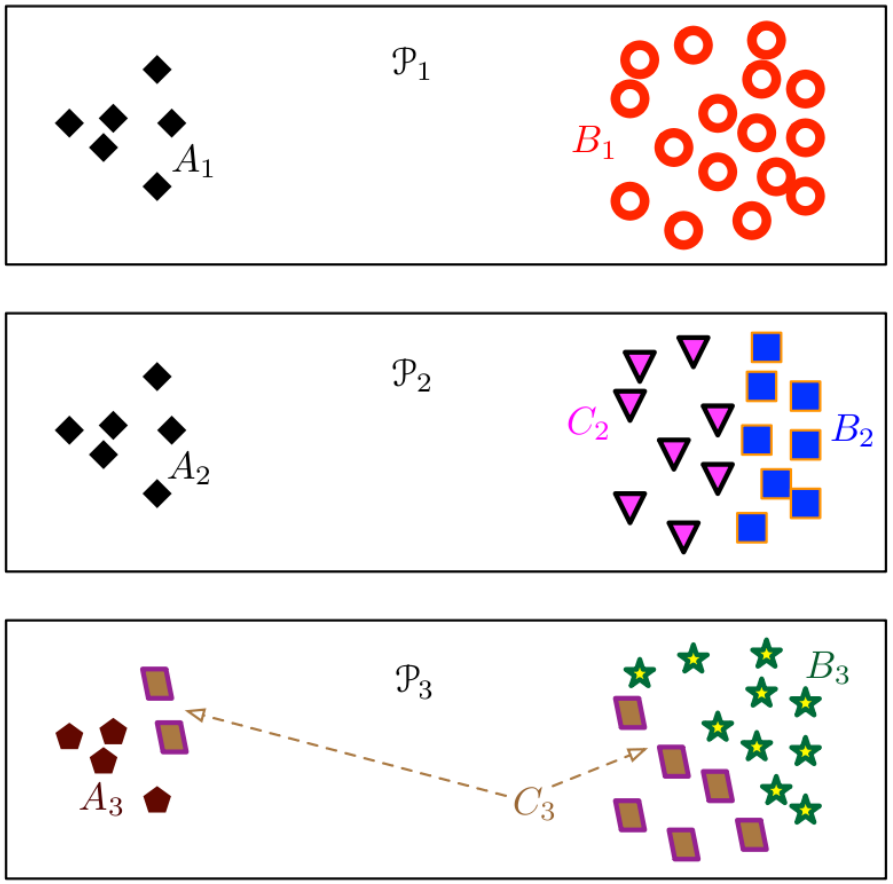
- Reduction to Euclidean clustering
  - Step 1: Lift clusters to vectors in RKHS
  - Step 2: Run any standard clustering algorithm on the vectors

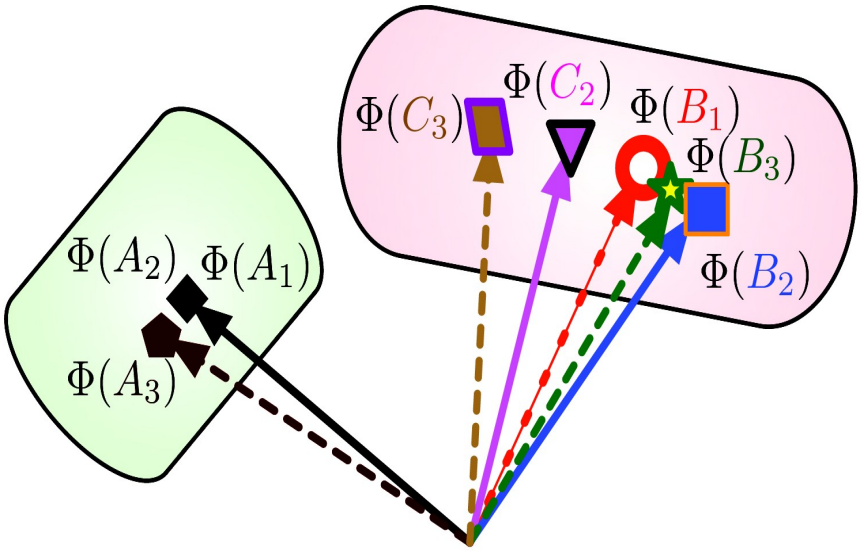




Consensus in RKHS

- Reduction to Euclidean clustering
  - Step 1: Lift clusters to vectors in RKHS
  - Step 2: Run any standard clustering algorithm on the vectors





Runtime Analysis

- Consensus

  Q: # of RKHS vectors

  m: # of input partitions

  n: # of points

  ρ: # of dimensions of the RKHS (~ log n)

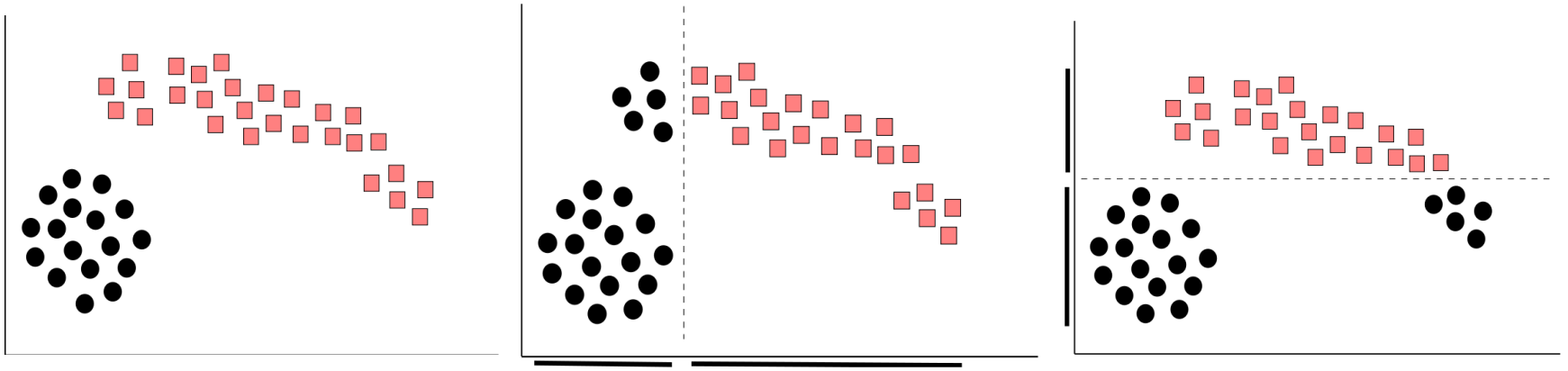
  - Runtimes
    - RKHS Vector: O(mn log n)
    - Consensus Iteration: O(|Q|k log n)
    - Final partition: O(n(log n + k) + |Q|)

- Usually $k, |Q| << n$. Assuming m is constant,
  - Consensus: O(n log n)

Experimental Setup

Dataset	# points	# dim	# clusters
Synthetic Data			
2D2C	45	2	2
2D3C	24	2	3
UCI Datasets			
Wine	178	13	3
Glass	214	10	7
Ionosphere	351	35	2
Soybean	307	35	19
MNIST	60000	728	10

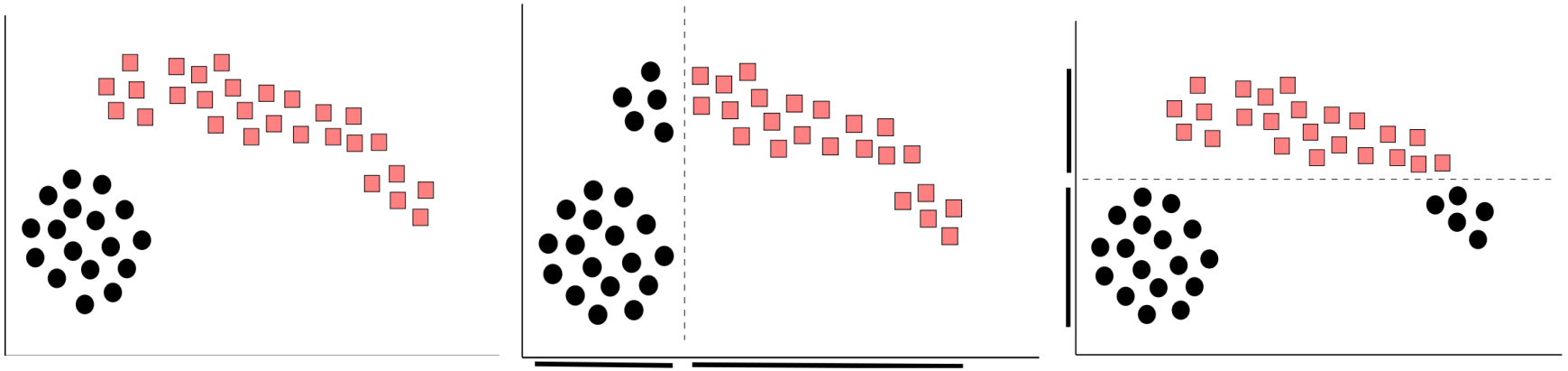
Results: Spatial Awareness



Ref. Partition (RP) First Partition (FP) Second Partition (SP)

Technique	***d(RP,FP) < d(RP,SP)***	
D_{ADCO}	1.71	1.78
Cdistance	0.24	0.35
LiftEMD	0.43	0.512
LiftKD	0.29	0.325
LiftH	0.41	0.49

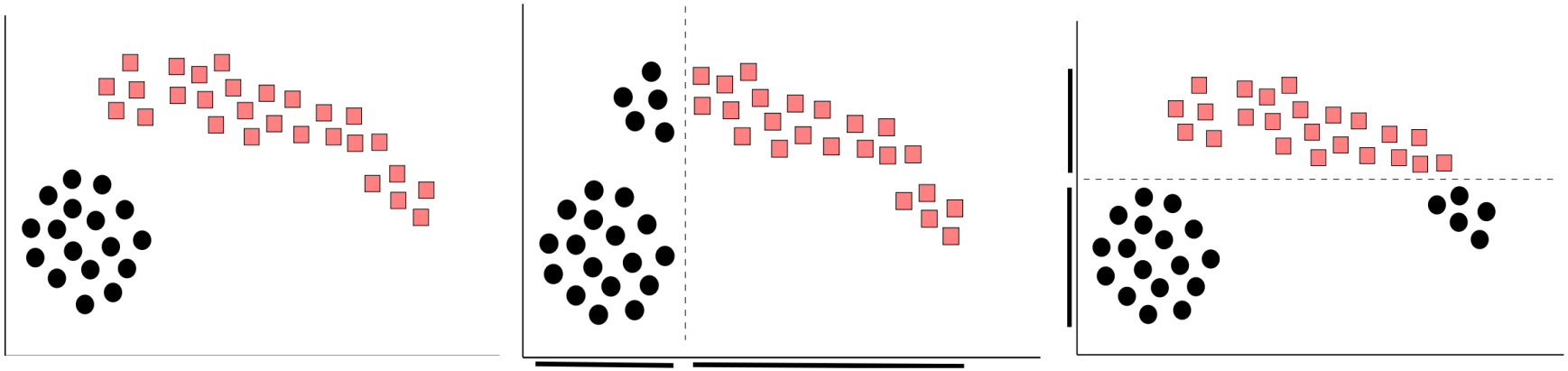
Results: Spatial Awareness



Ref. Partition (RP) First Partition (FP) Second Partition (SP)

Technique	***d(RP,FP) < d(RP,SP)***	
D_{ADCO}	1.71	1.78
Cdistance	0.24	0.35
LiftEMD	0.43	0.512
LiftKD	0.29	0.325
LiftH	0.41	0.49

Results: Spatial Awareness



Ref. Partition (RP) First Partition (FP) Second Partition (SP)

Technique	***d(RP,FP) < d(RP,SP)***	
D_{ADCO}	1.71	1.78
Cdistance	0.24	0.35
LiftEMD	0.43	0.512
LiftKD	0.29	0.325
LiftH	0.41	0.49

Results: Runtimes

- LiftEMD vs CDistance
  - O(n log n) vs $O(n^3)$

Dataset	**# points**	**# dimensions**	**Cdistance**	**LiftEMD**
2D3C	24	2	2.03 ms	1.02 ms
2D2C	45	2	4.10 ms	1.95 ms
Wine	178	13	18.80 ms	6.90 ms
MNIST test data	10000	784	1360.20 s	303.90 s
MNIST training data	60000	784	202681 s	1774.20 s

Results: Consensus

- Consensus performance
  - Distance from true partition using LiftEMD metric
  - Compare against CSPA, HGPA and MCLA *[Strehl et. al.]*

Dataset	***CSPA***	***HGPA***	***MCLA***	***LiftKm***	***LiftHAC***
IRIS	**0.113**	0.295	0.812	**0.106**	0.21
Glass	0.573	**0.519**	0.731	**0.531**	0.54
Ionosphere	**0.729**	0.767	0.993	0.731	**0.72**
Soybean	0.51	0.495	0.951	**0.277**	**0.29**
Wine	0.873	0.875	0.917	**0.831**	**0.842**
MNIST test data	0.182	-	0.344	**0.106**	**0.112**

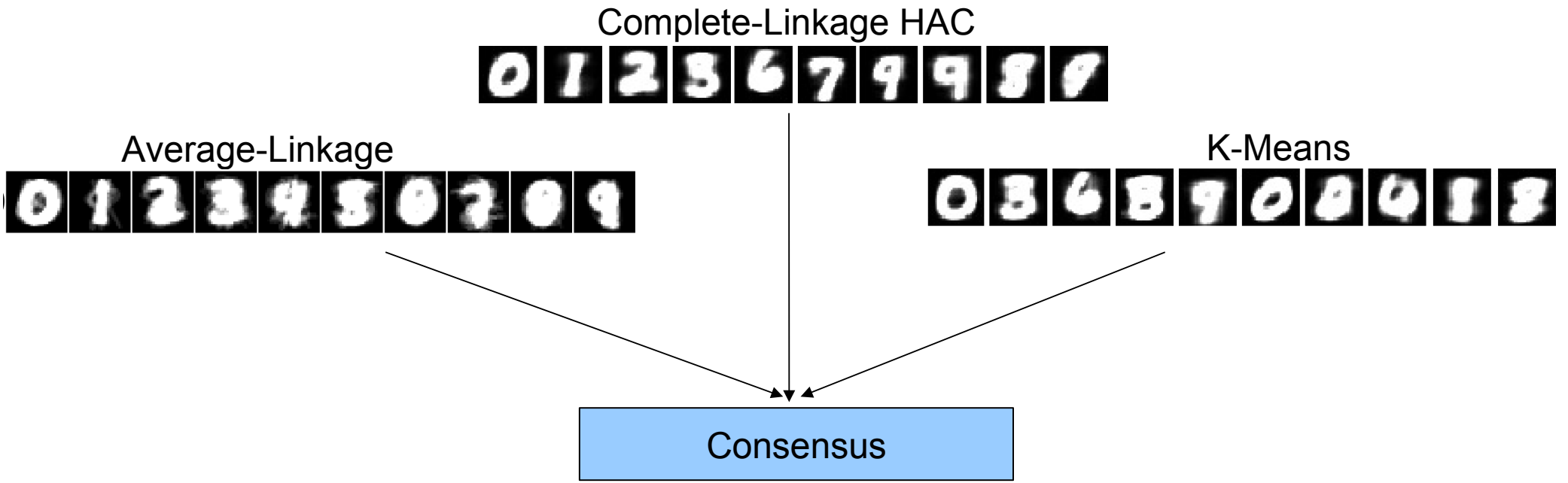
MNIST digits data

Complete-Linkage HAC

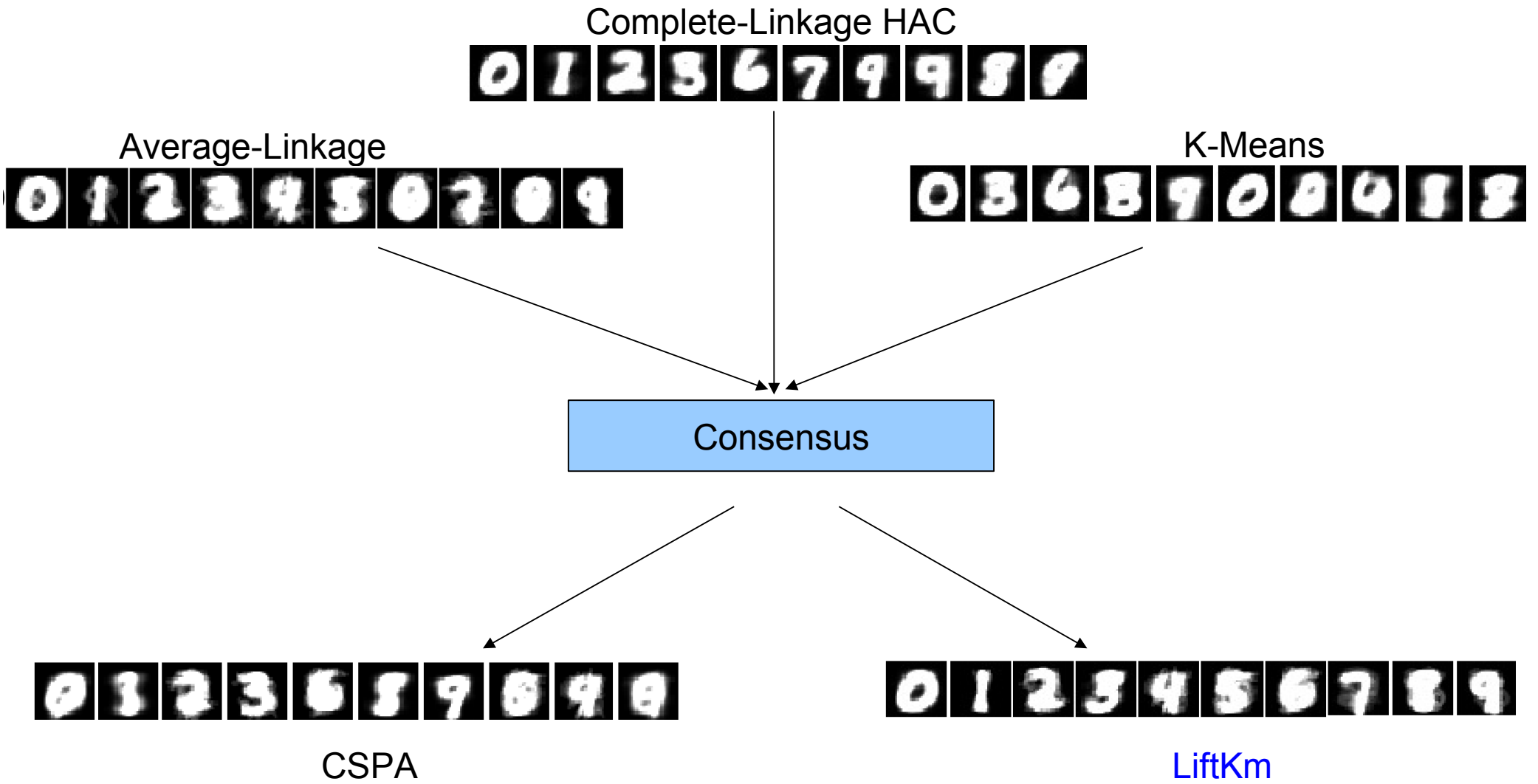
Average-Linkage

K-Means

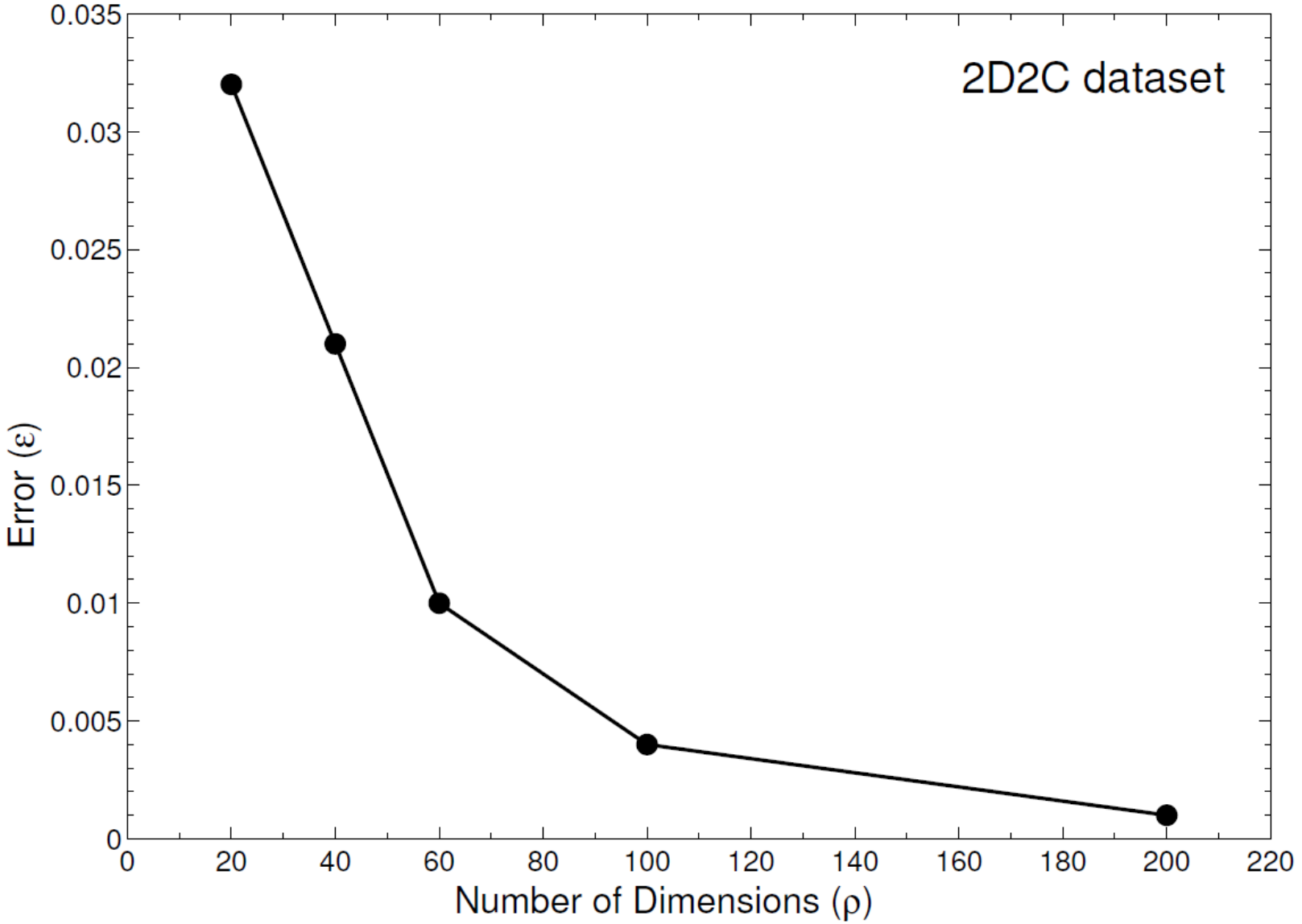
MNIST digits data



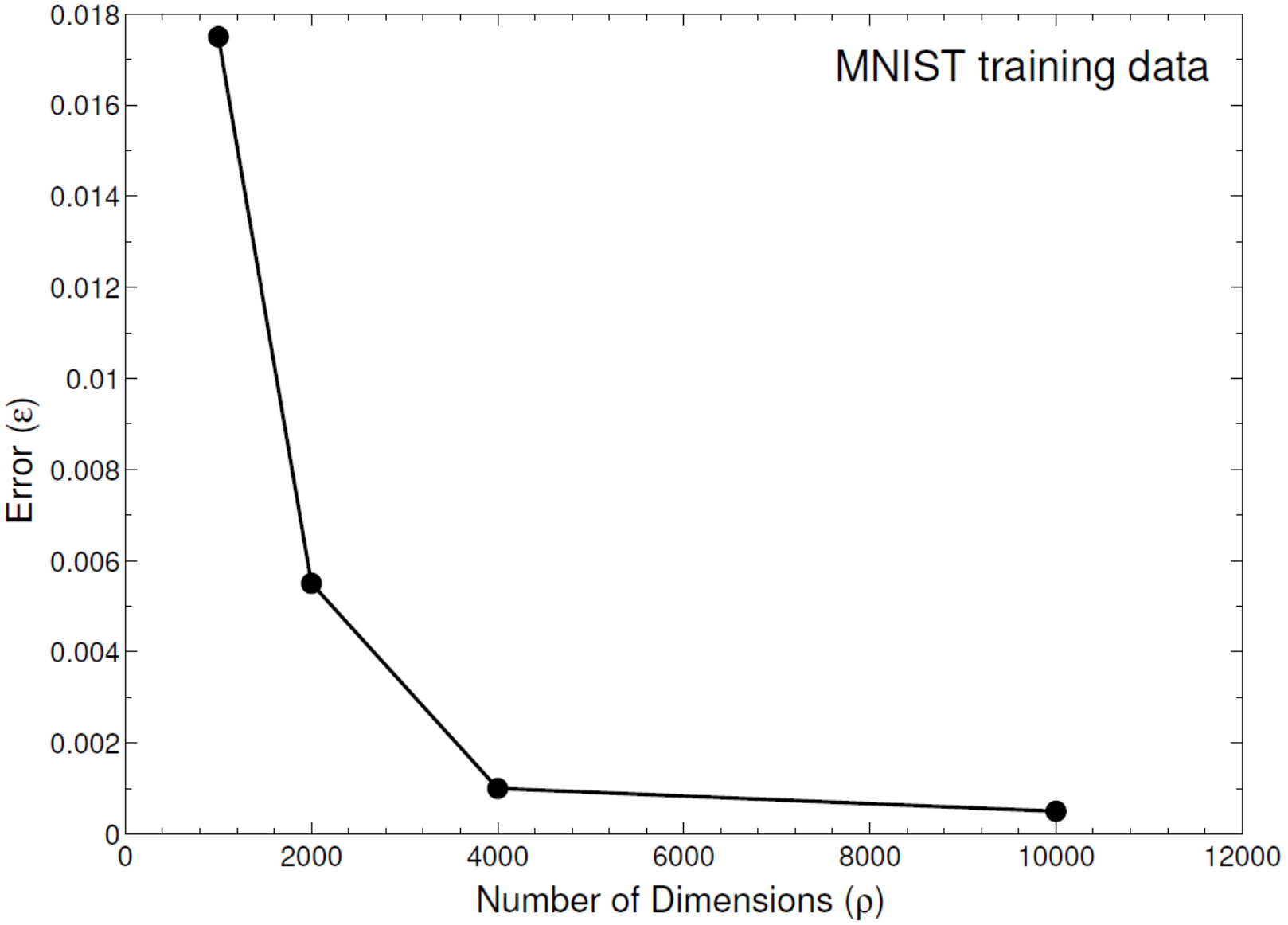
MNIST digits data



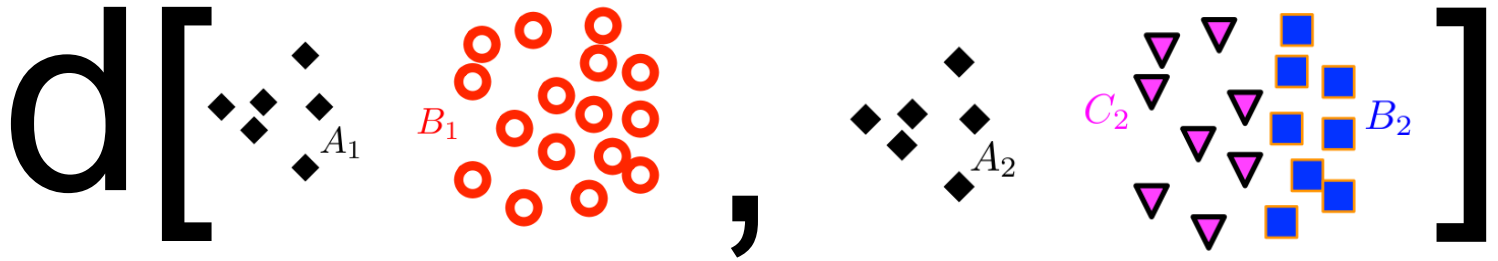
Error in LiftEMD



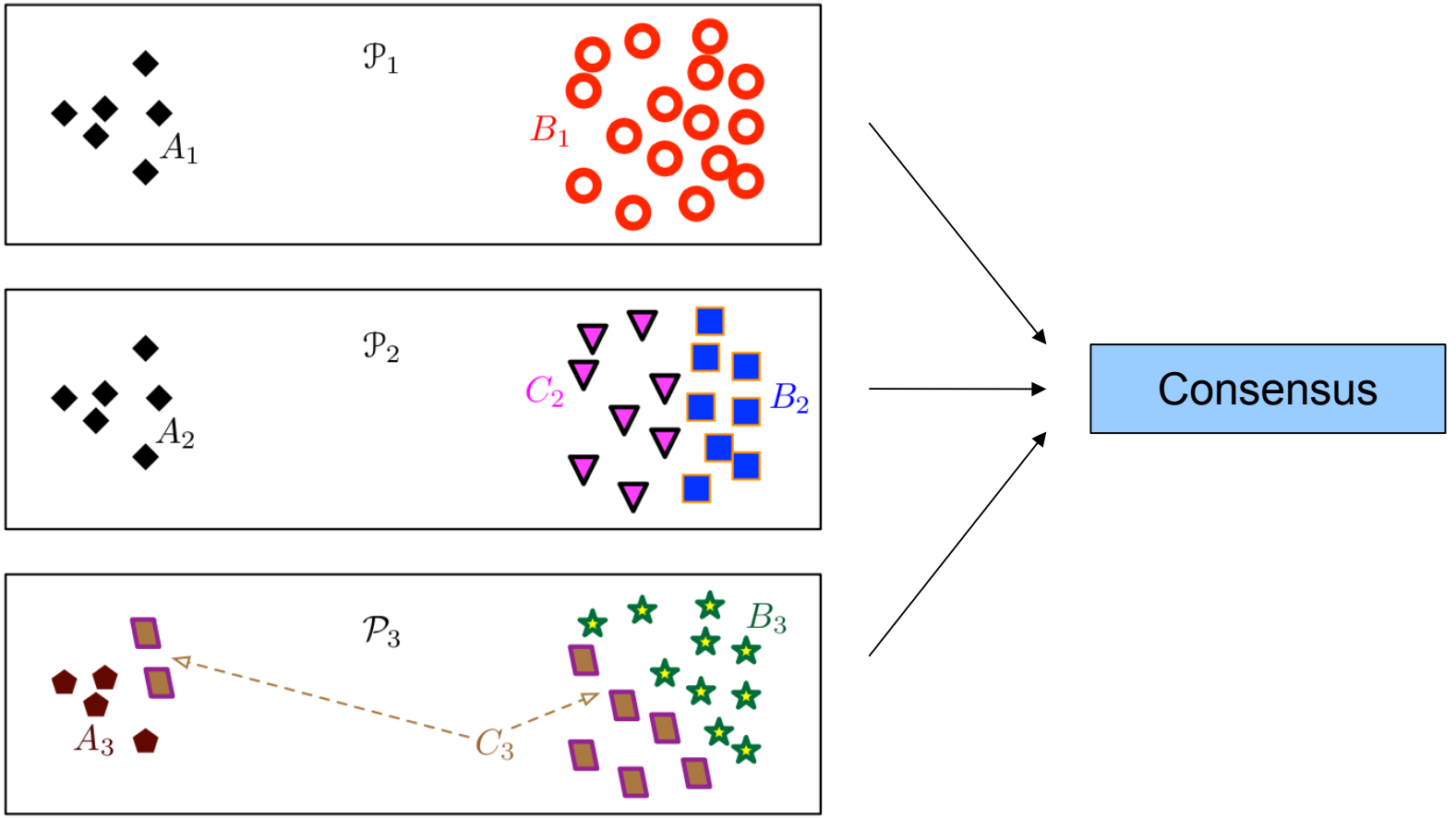
Error in LiftEMD



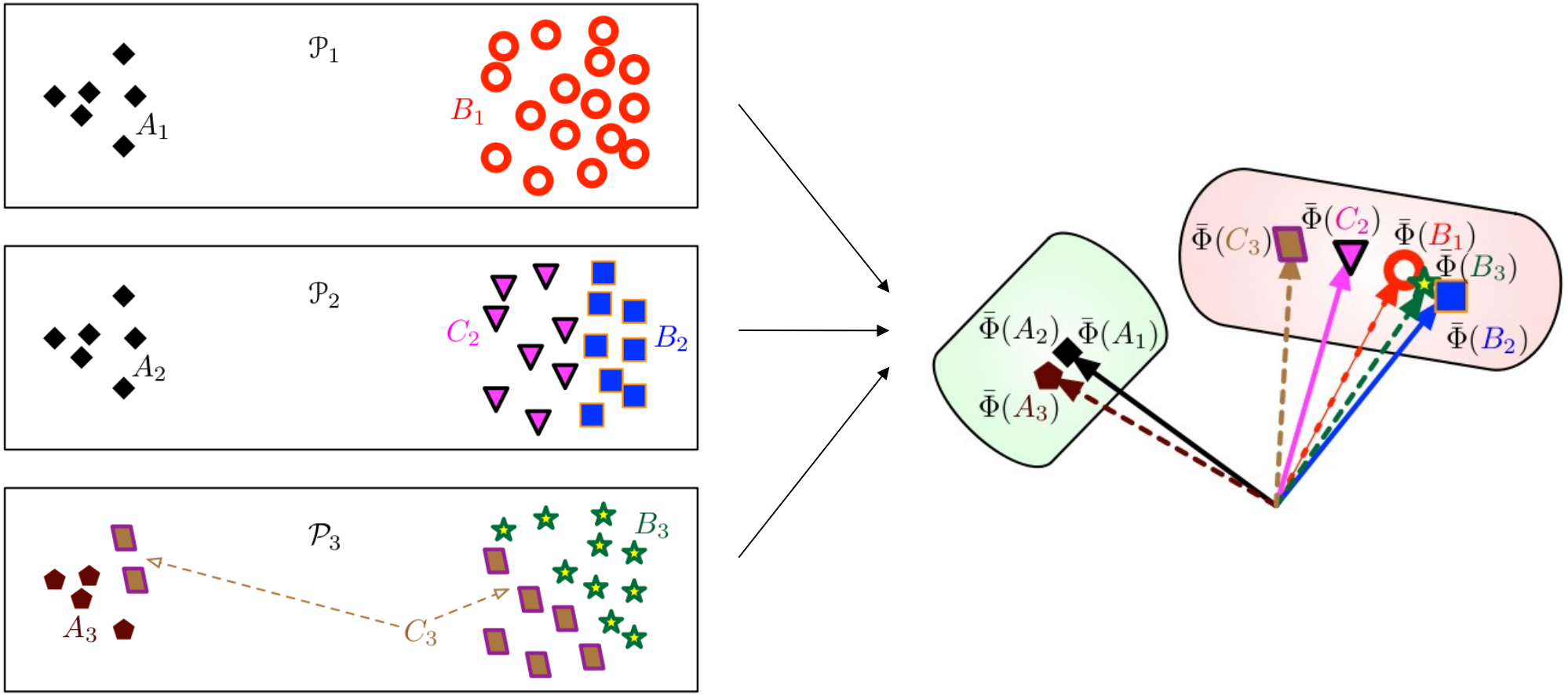
Recap



Recap


$\mathcal{P}_1$
A_1
B_1
$\mathcal{P}_2$
A_2
C_2
B_2
$\mathcal{P}_3$
A_3
C_3
B_3
Consensus

Recap


$\mathcal{P}_1$
A_1
B_1
$\mathcal{P}_2$
A_2
C_2
B_2
$\mathcal{P}_3$
A_3
C_3
B_3
$\bar{\Phi}(A_1)$
$\bar{\Phi}(A_2)$
$\bar{\Phi}(A_3)$
$\bar{\Phi}(C_3)$
$\bar{\Phi}(C_2)$
$\bar{\Phi}(B_1)$
$\bar{\Phi}(B_3)$
$\bar{\Phi}(B_2)$

Topic 2: Alternate or Non-Redundant Clustering

Generating a Diverse Set of High-Quality Clusterings

MultiClust 2011

Joint work with:
Jeff M. Phillips, *University of Utah*
Suresh Venkatasubramanian, *University of Utah*

Best Clustering Vs Choices

- There might not be a *“one-best”* clustering

- User might need a variety of choices
  - Understand structures in data
  - Answer different types of questions on the data

- Best quality partition might not be *interesting* to the user

Issues with finding many partitions

- Data exists in very high dimensions
  - Visualization to understand the structure: infeasible
- Criteria for clustering: Often unknown
  - Clustering precedes many data analysis processes
  - Users have limited idea of what they want
- Running multiple methods can fail
  - Partitions obtained may not be good quality and non-redundant

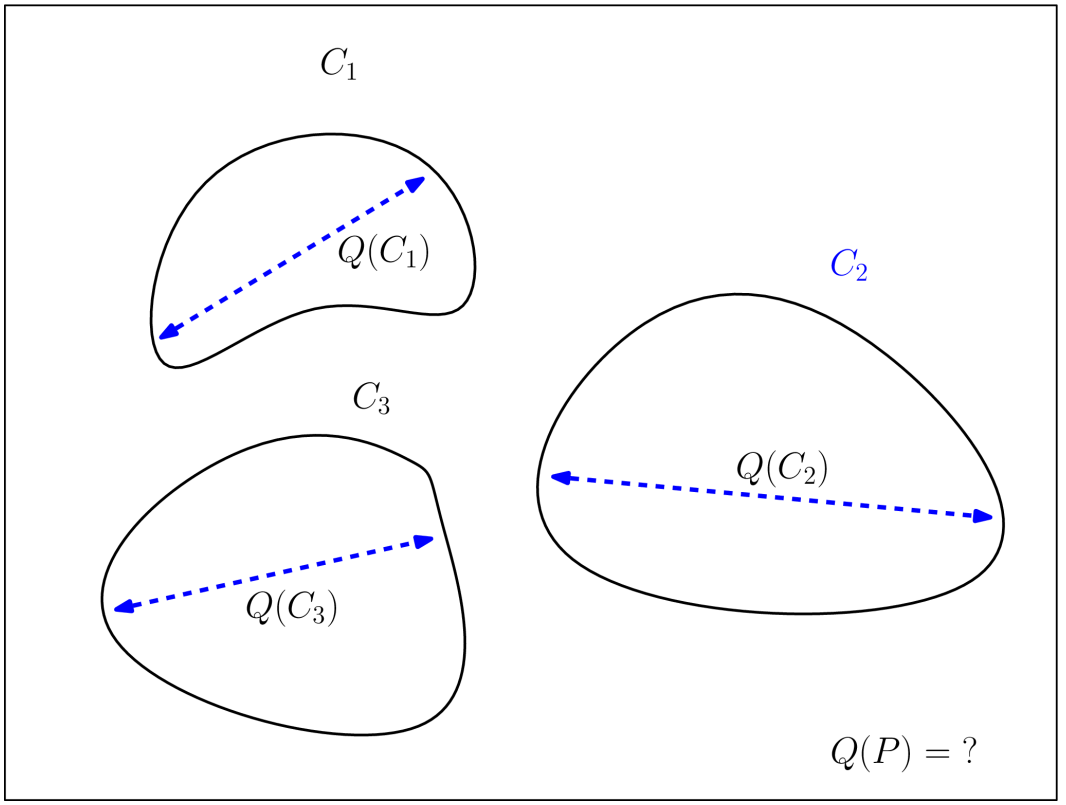
Issues with finding many partitions

- Data exists in very high dimensions
  - Visualization to understand the structure: infeasible
- Criteria for clustering: Often unknown
  - Clustering precedes many data analysis processes
  - Users have limited idea of what they want
- Running multiple methods can fail
  - Partitions obtained may not be good quality and non-redundant

Need a systematic approach to generating many partitions!

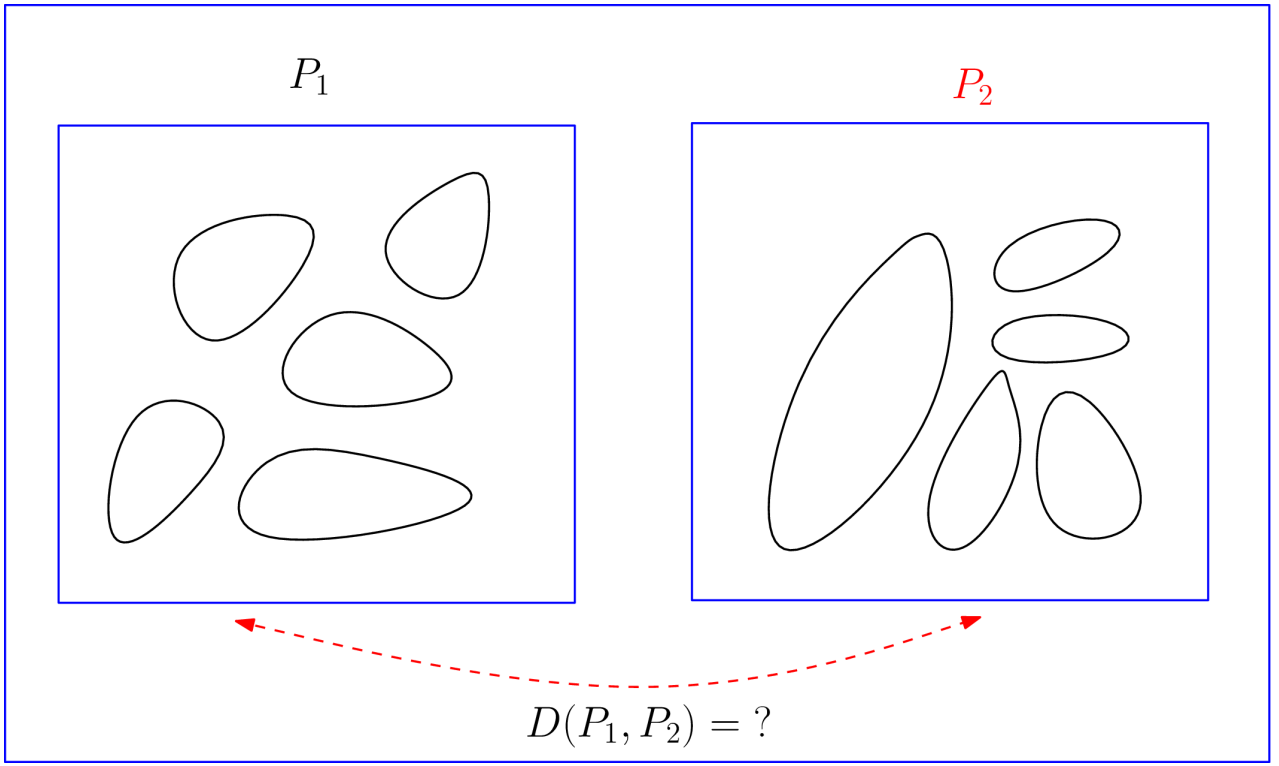
Ingredients

- Is my partition *'good'*?
  - Why? [Need meaningful partitions]
  - Need a measure for quality of a partition
  - Degree to which the structures inside the data is captured



Ingredients

- Are my partitions *'non-redundant'*?
  - Why? [Similar solutions are uninteresting]
  - Need a measure for distance between partitions
  - Dissimilarity between partitions



What we use?

- We need 2 quantities
  - Distance
  - Quality
- We use our previous measures
  - Lift clusters to a high-dimensional Reproducing Kernel Hilbert Space
  - Compute distance and quality in this space
- Pick your choice!
  - Our method is agnostic to the choice of distance and quality

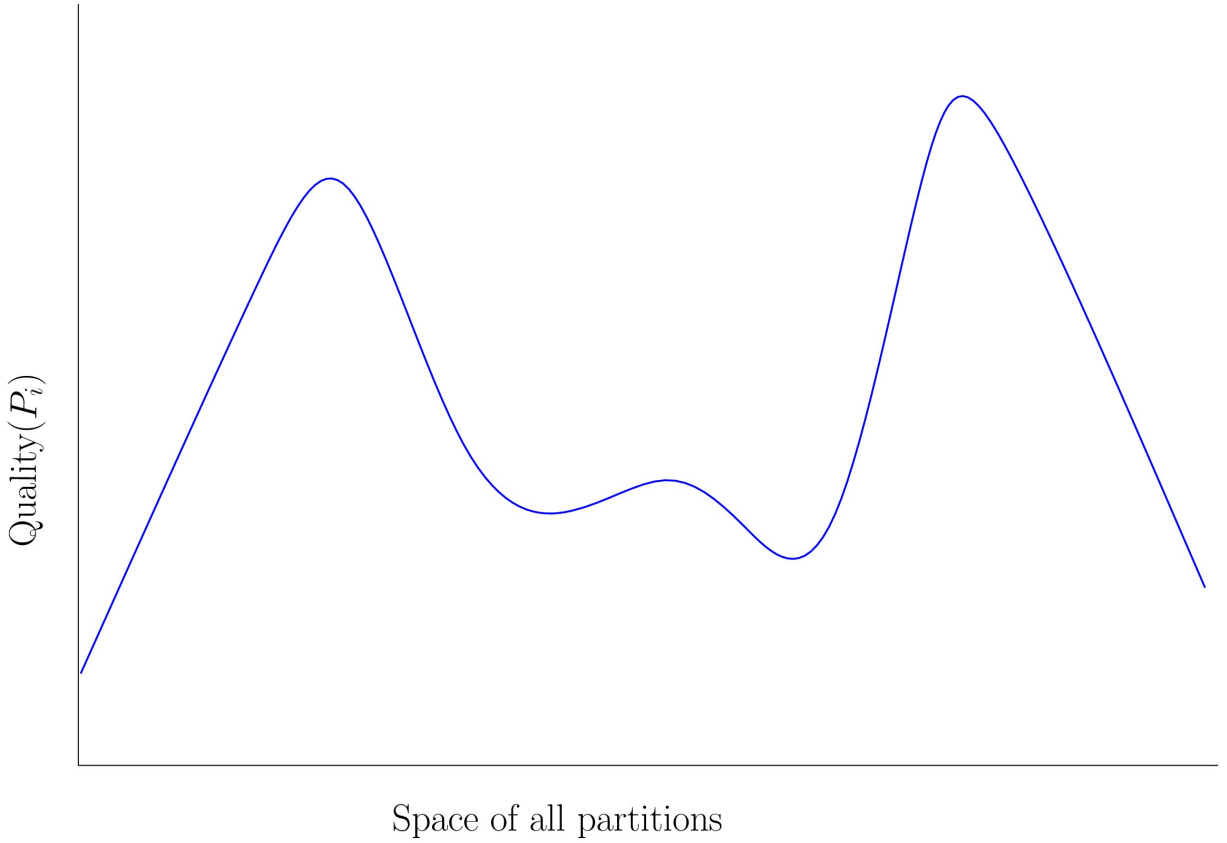
1. A. Rahimi, B. Recht, NIPS 2007
2. S. Joshi, R.V. Kommaraju, J.M. Phillips, S. Venkatasubramanian, SoCG 2011
3. P. Raman, J.M. Phillips, S. Venkatasubramanian, SDM 2011

Goal

- Generate *k* partitions

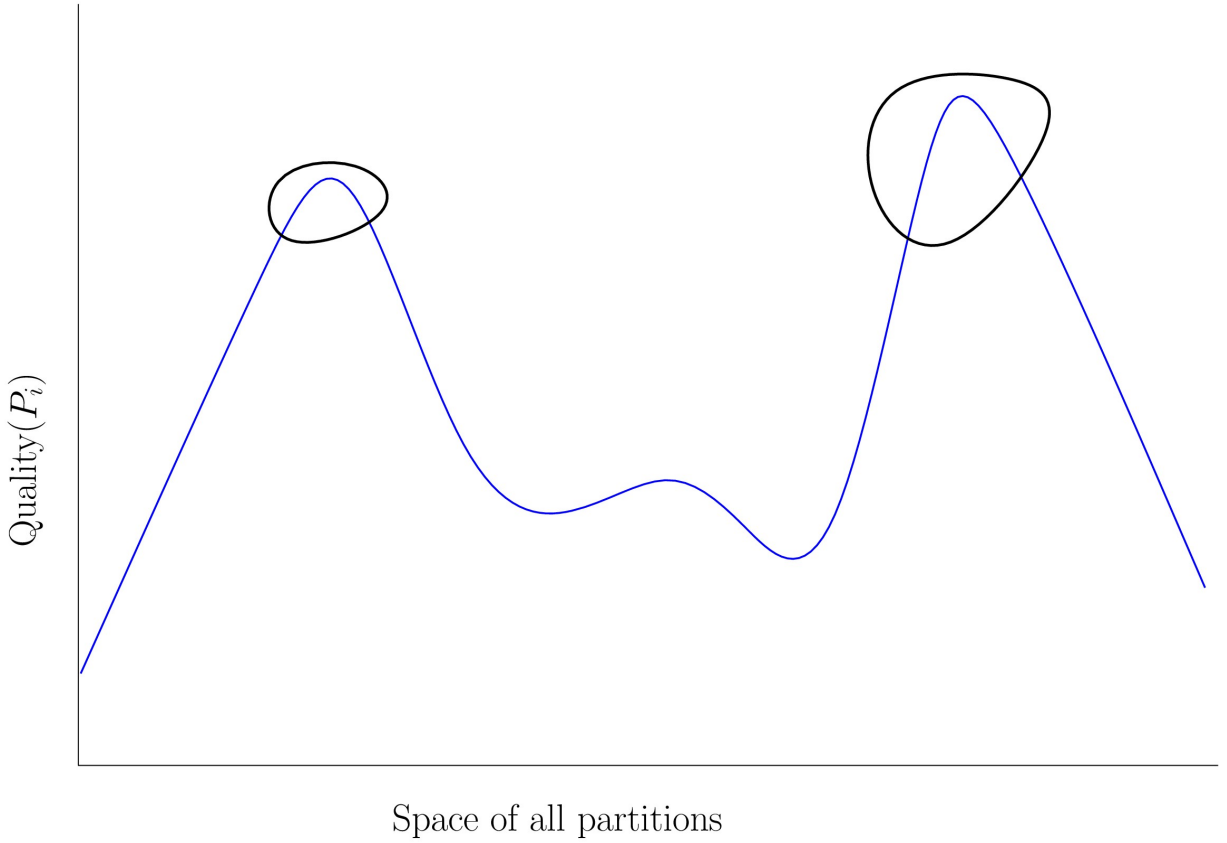
Goal

- Generate *k* partitions
  - that best reflect the high-quality partitions



Goal

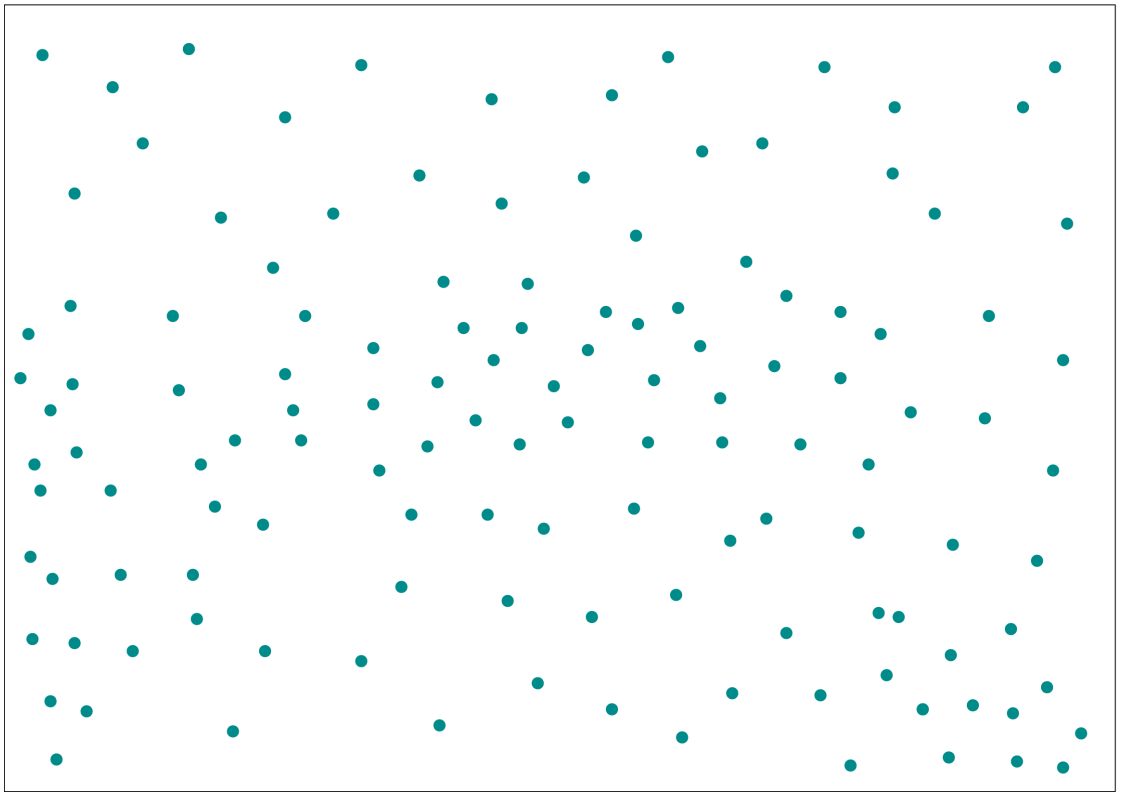
- Generate *k* partitions
  - that best reflect the high-quality partitions



Goal

- Generate *k* partitions
  - that best reflect the high-quality partitions
  - and are non-redundant

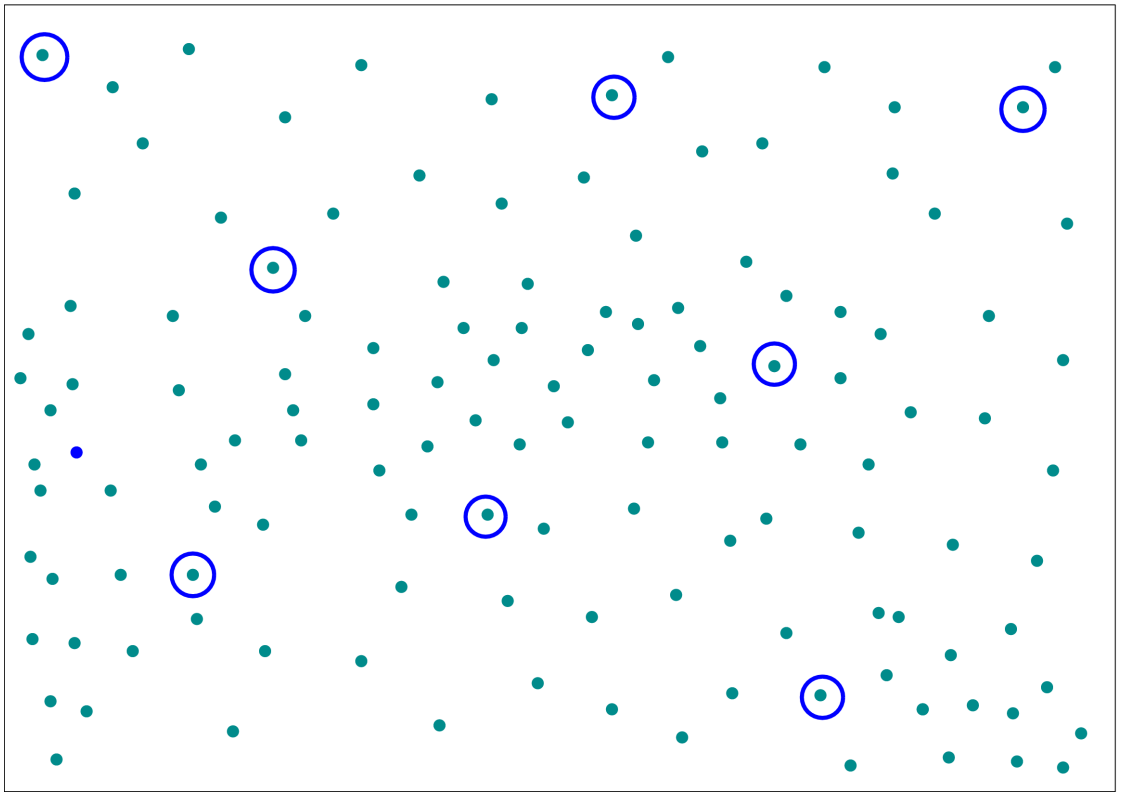
2-d viz of the space of all partitions with k clusters



Goal

- Generate *k* partitions
  - that best reflect the high-quality partitions
  - and are non-redundant

2-d viz of the space of all partitions with k clusters

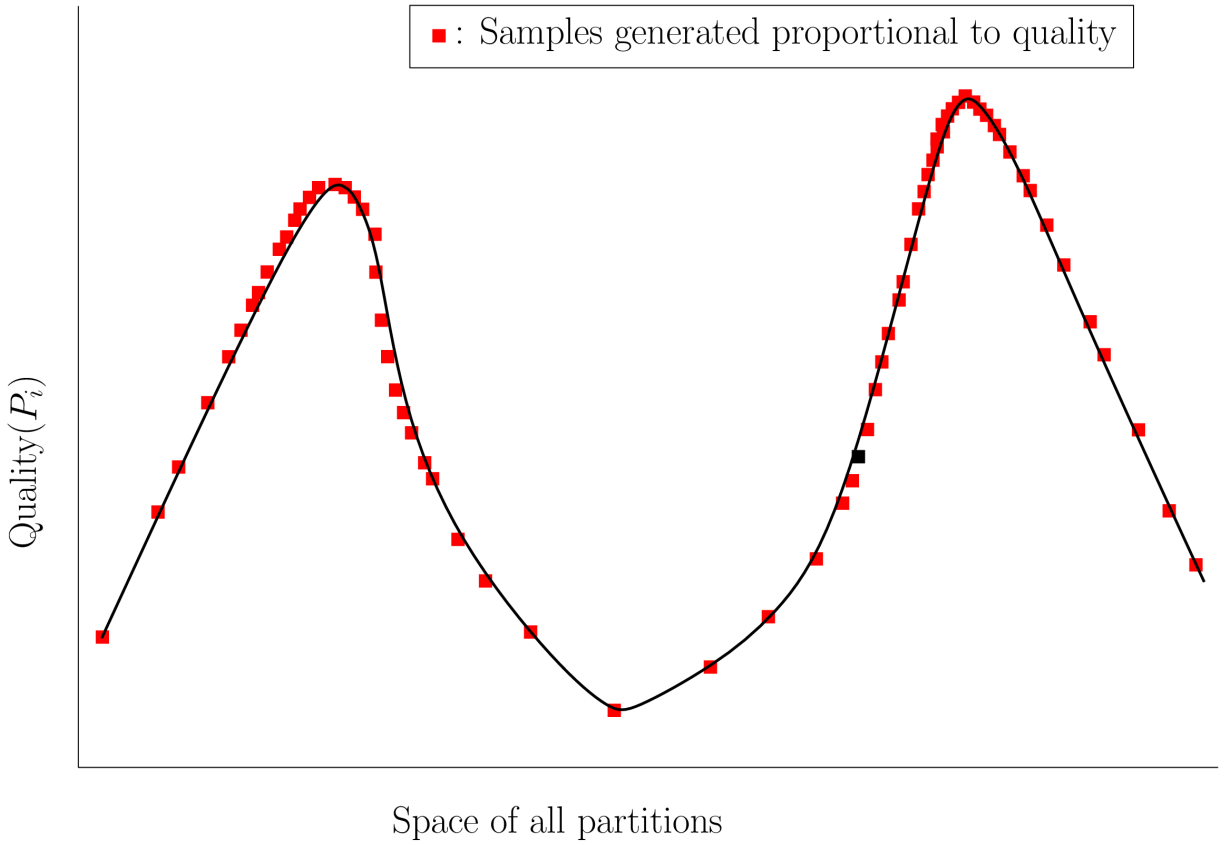


Related Work

- Generate one alternate partition
  - Usually a partition that maximizes function based on quality and distance
  - This might not be the most interesting partition
- Generate *k* partitions
  - Usually iterative
  - Maximize function: f(quality,distance)
  - Quality quickly degrades

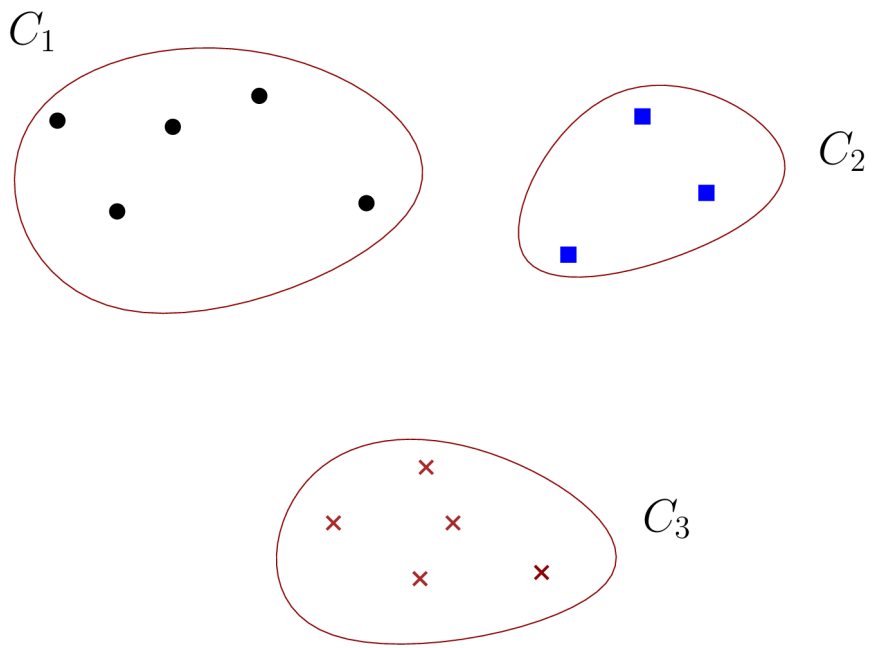
1. Generation [What we do]

- Explore the landscape of partitions
  - Understand the peaks
  - Sample a lot of *'good'* partitions from this landscape



1. Generation [How we do it]

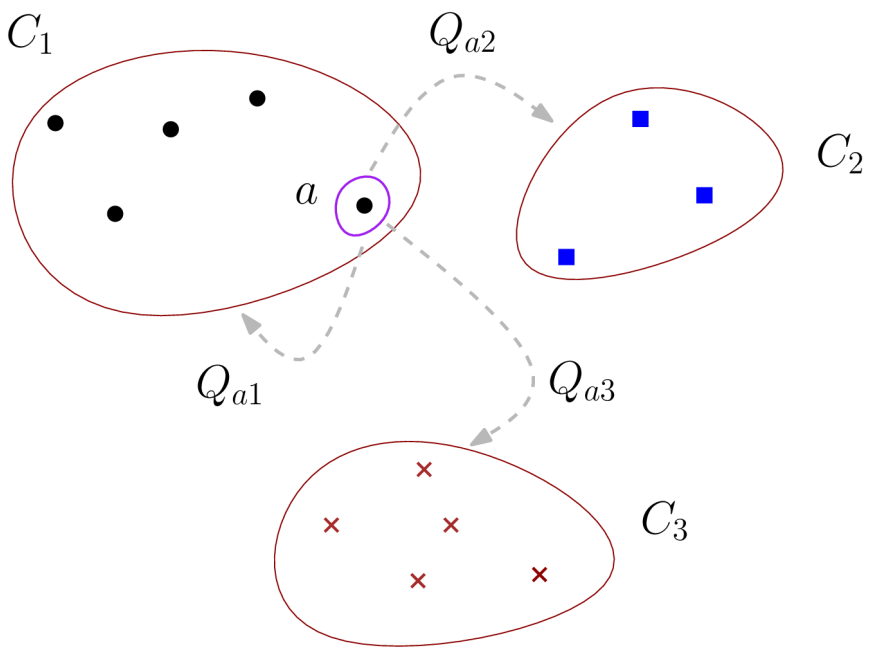
- Sample proportional to quality : Gibbs Sampling
  - Move points across clusters to get a new configuration
  - *'Move'* is proportional to the quality



[Gibbs sampling – generates items proportional to a measure]

1. Generation [How we do it]

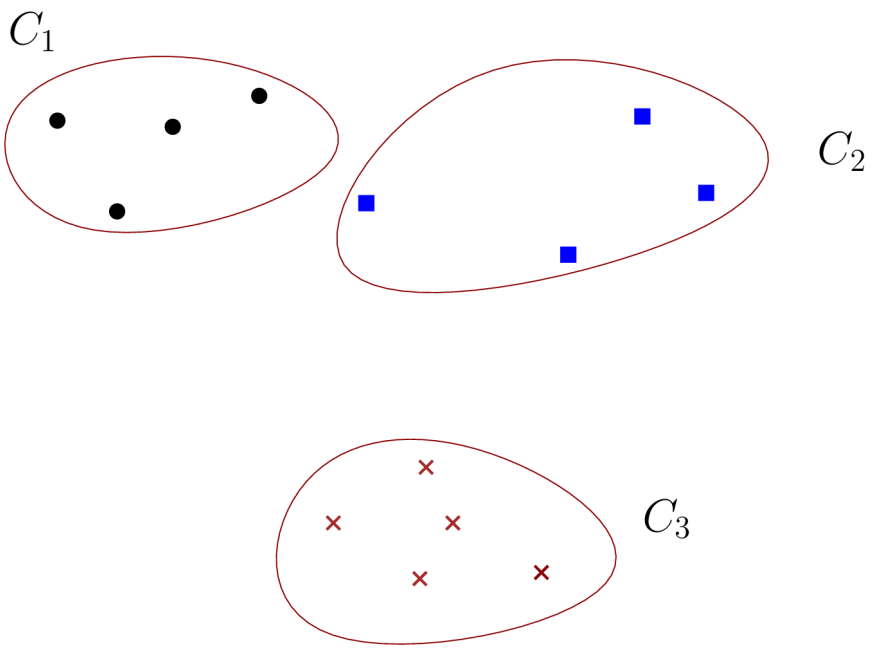
- Sample proportional to quality : Gibbs Sampling
  - Move points across clusters to get a new configuration
  - *'Move'* is proportional to the quality



[Gibbs sampling – generates items proportional to a measure]

1. Generation [How we do it]

- Sample proportional to quality : Gibbs Sampling
  - Move points across clusters to get a new configuration
  - *'Move'* is proportional to the quality

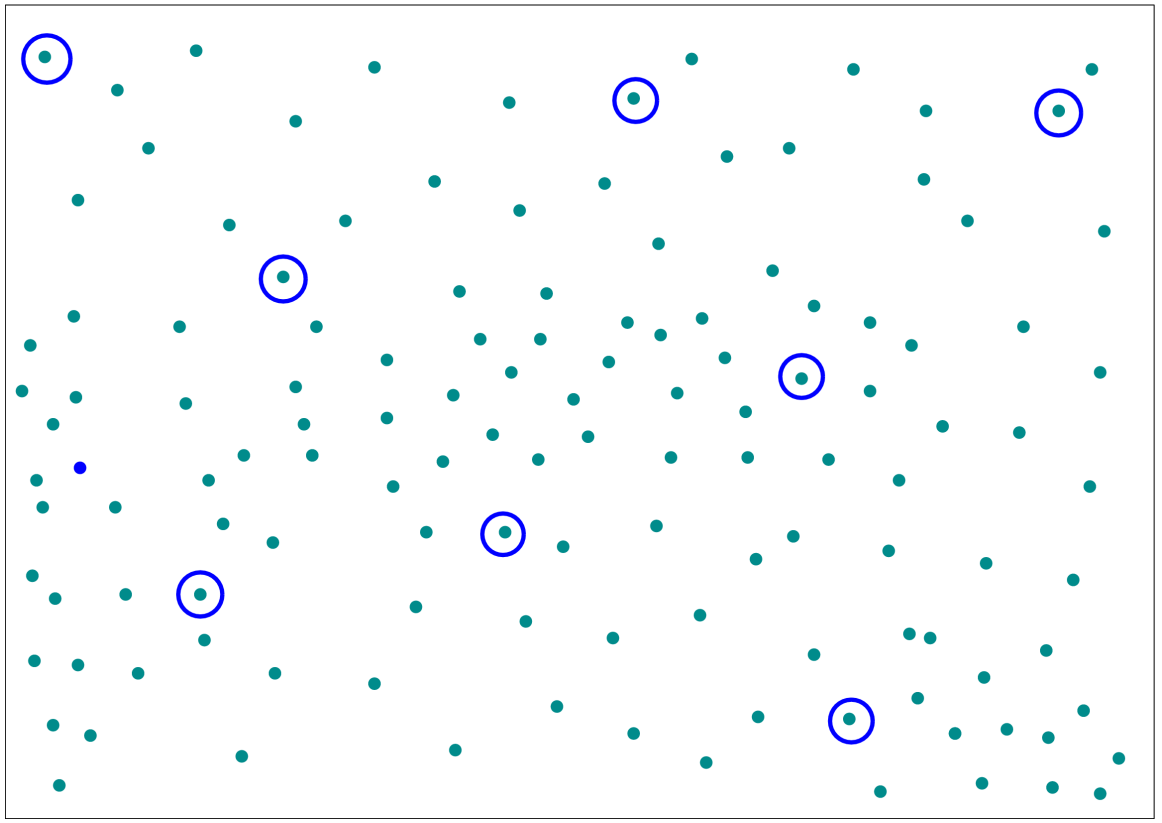


[Gibbs sampling – generates items proportional to a measure]

2. Picking [What we do]

- Pick representative partitions
  - This forms the alternate solutions
  - Pick dissimilar partitions to ensure variety

2-d viz of the space of all partitions with k clusters



2. Picking [How we do it]

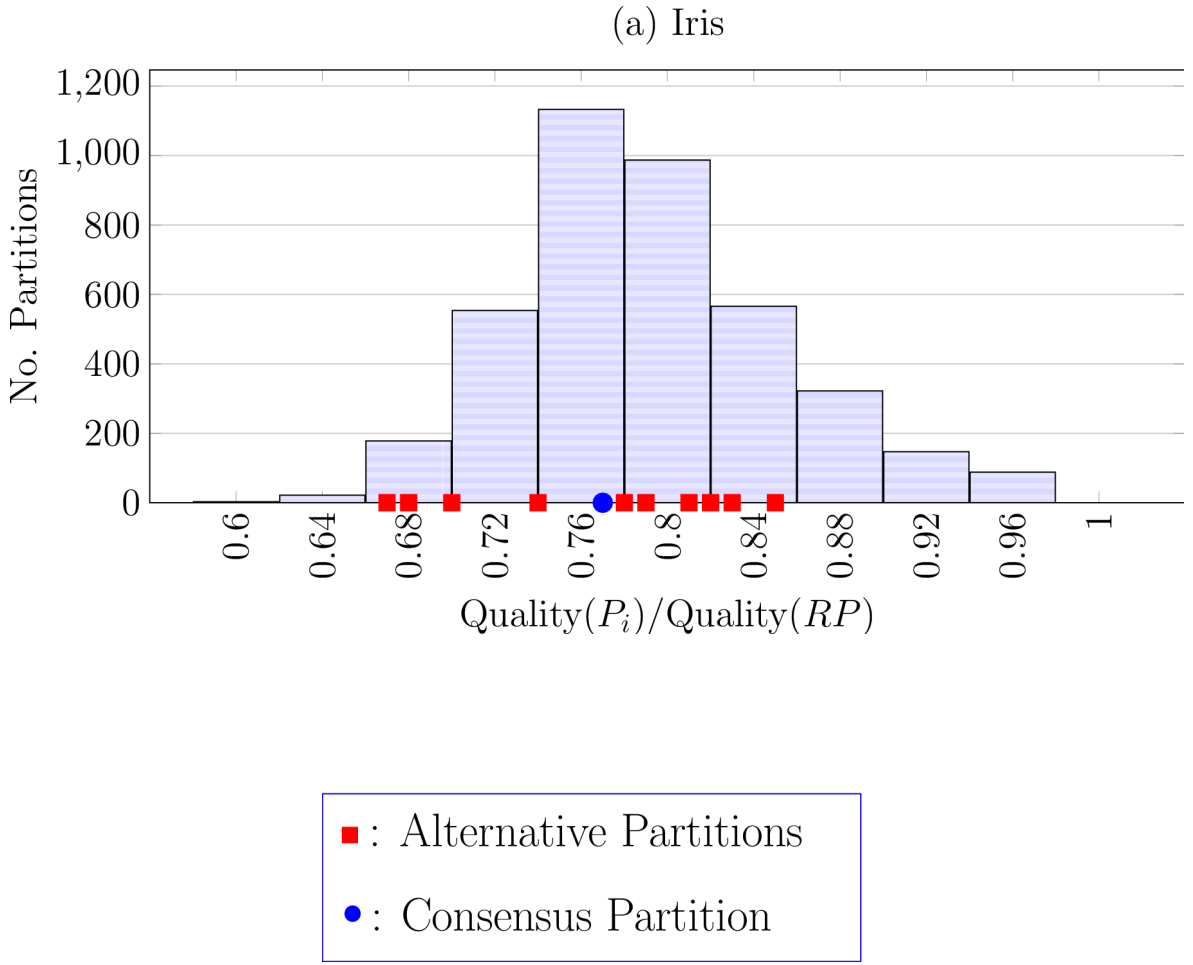
- K-Center
  - Gonzalez method gives a 2-approximation to k-center
  - *Pick* k far-away partitions iteratively
  - Report the *'k-centers'* as the k alternative partitions
  - *Progressive; pick next center when a new request comes in!*
- Pick your choice!
  - Discrete Kmeans?
- Optional Assignment
  - What did other partitions mean?
  - Assign them to the closest alternative partition

Experimental Setup

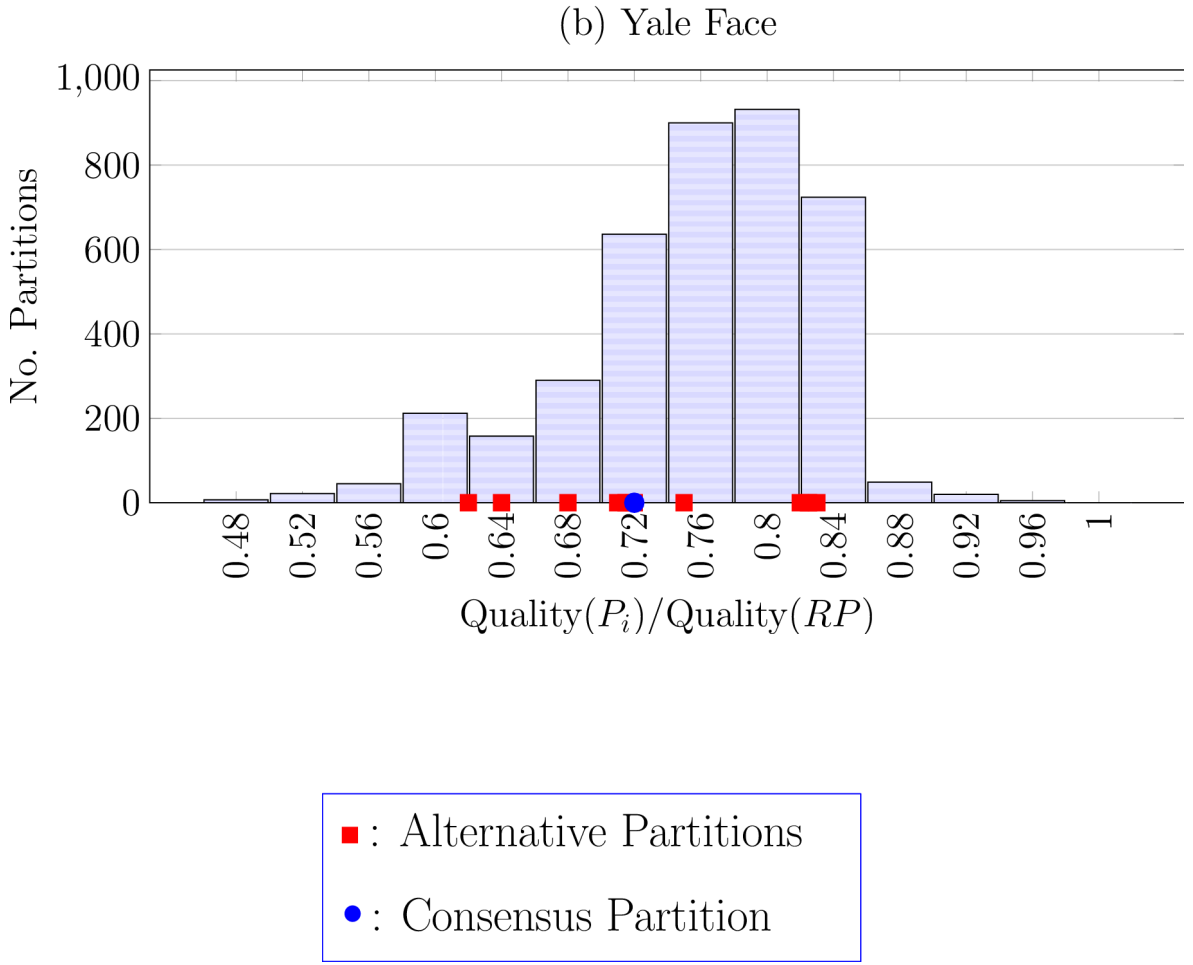
- Data
  - 2D5C [n = 100; d = 2]
  - Iris [n = 150; d = 4]
  - Subset of Yale Face Database B [n = 90; d = 1200]

- What are we looking for?
  - Quality: Did we span the *landscape* of all partitions?
  - Diversity: Did we generate *non-redundant* partitions?
  - Are the partitions visually appealing?

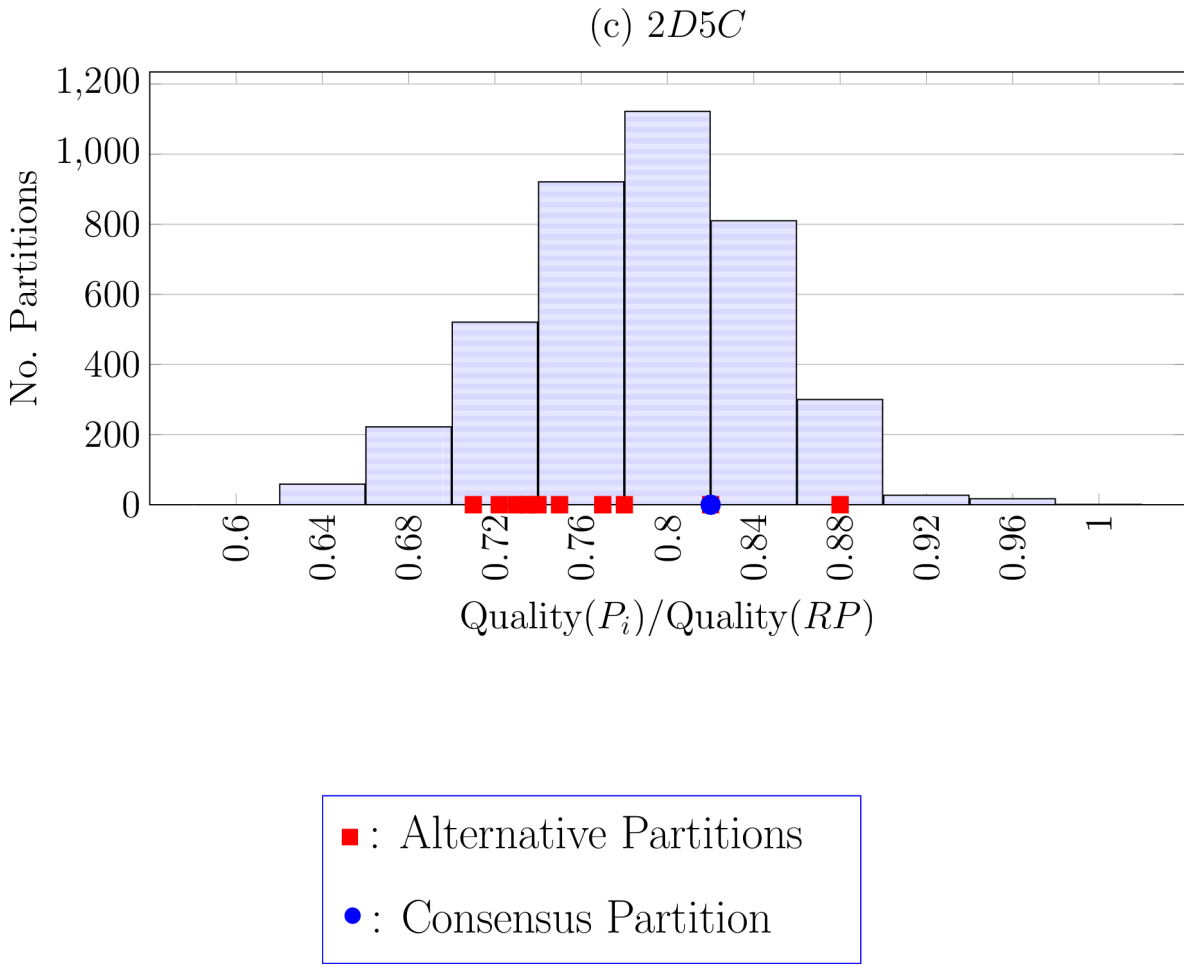
Good Quality Partitions



Good Quality Partitions

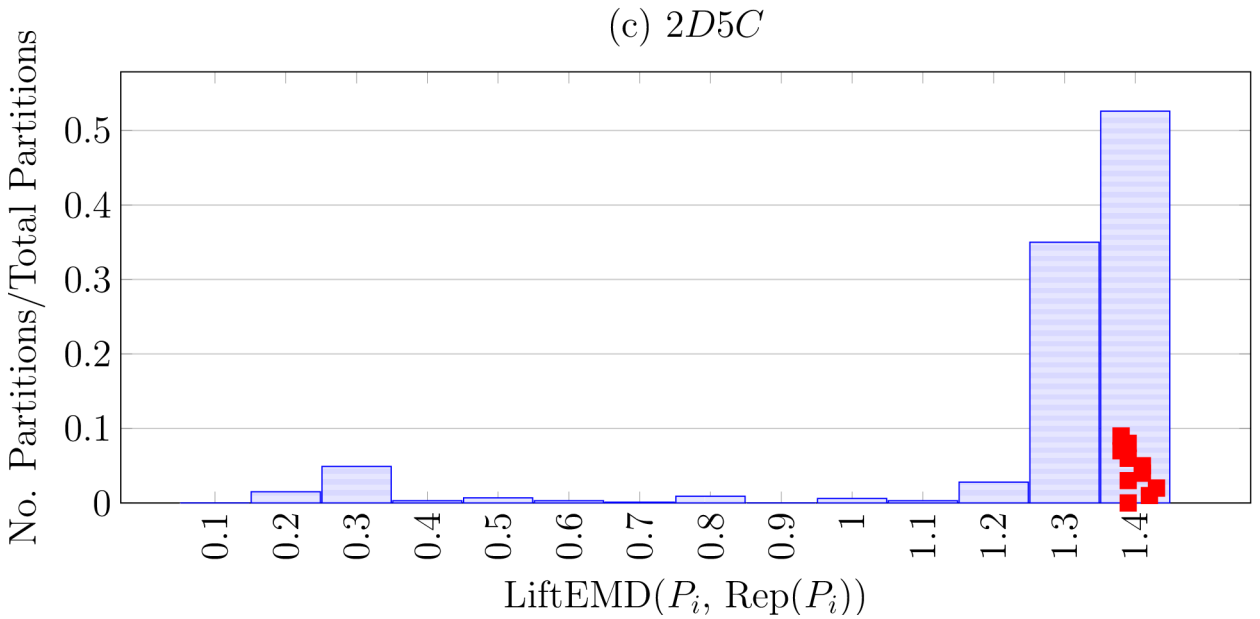


Good Quality Partitions



Non-Redundant Partitions

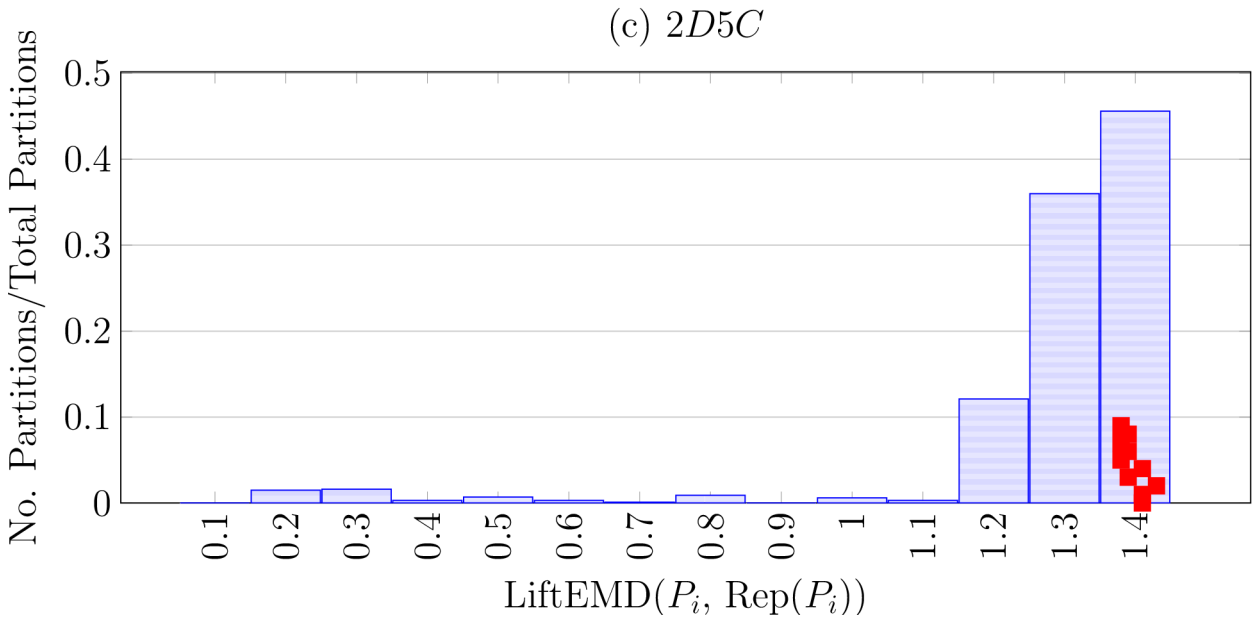
Quality = *Width* of a partition



■ : Distance between an alternative and its closest other alternative

Non-Redundant Partitions

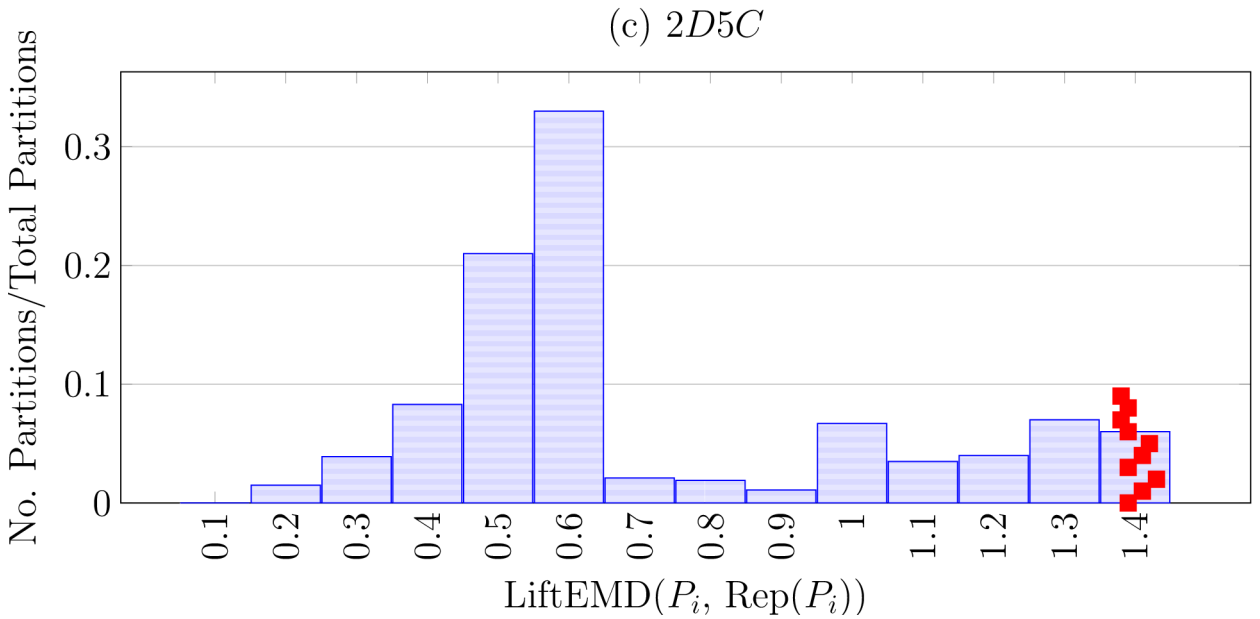
All partitions generated at random



■ : Distance between an alternative and its closest other alternative

Non-Redundant Partitions

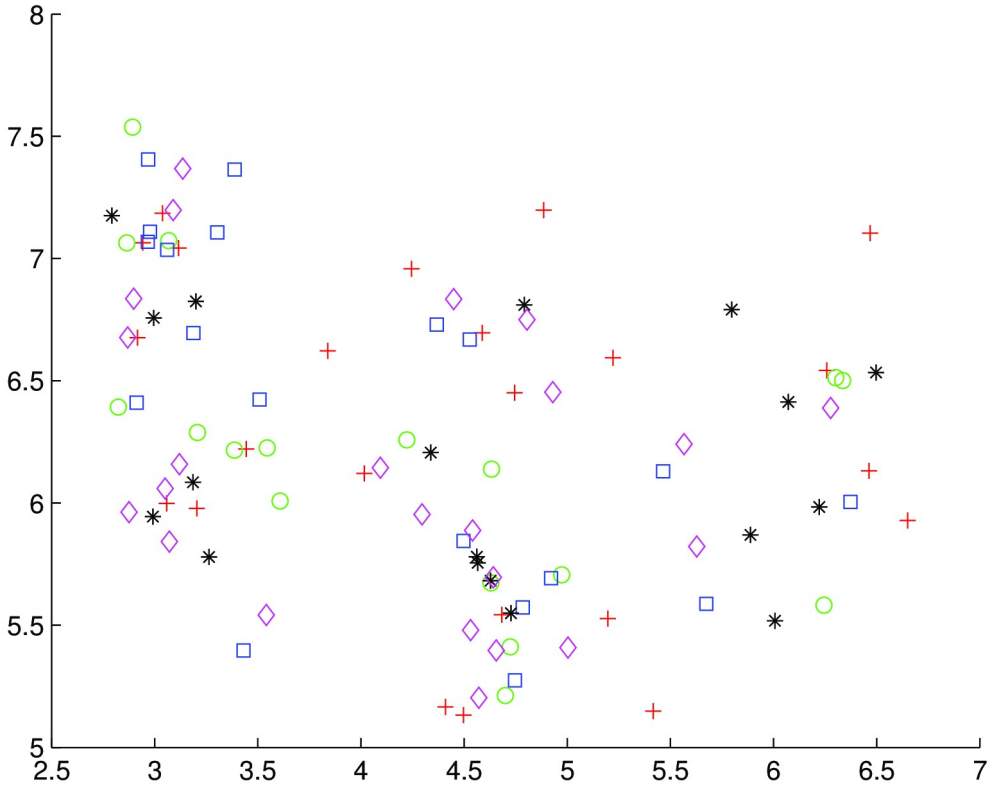
Quality = [*Split + Width] of a partition*



■ : Distance between an alternative and its closest other alternative

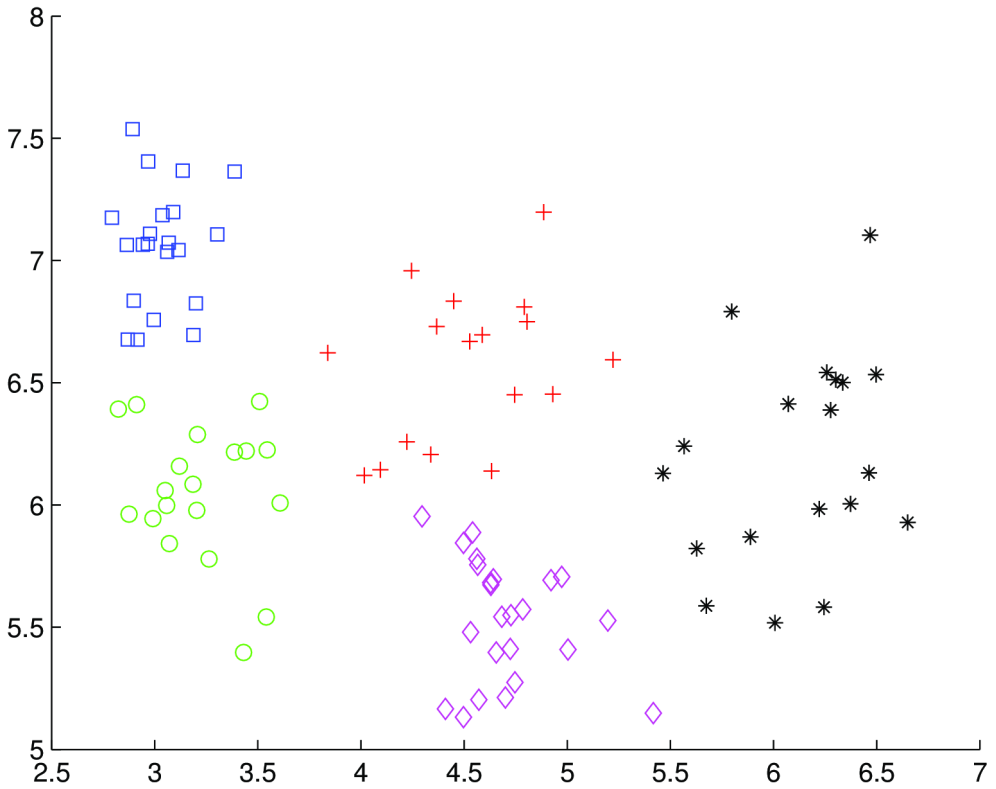
Non-Redundant Partitions

Sample Representative Partition generated using
"Width" as quality

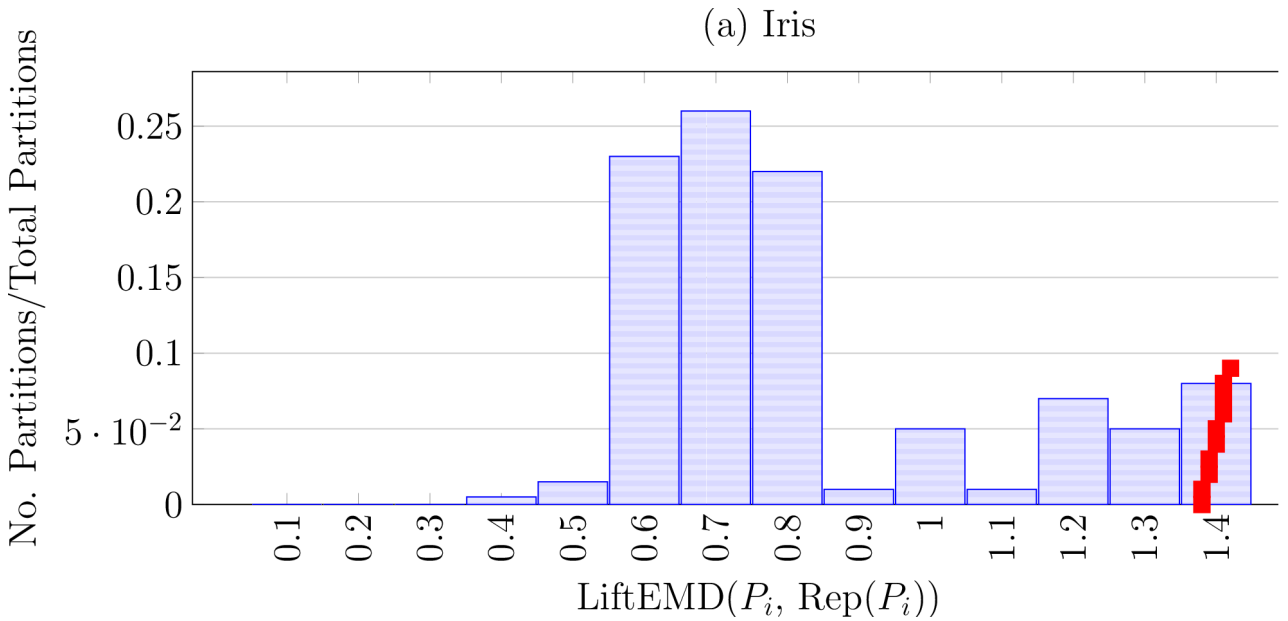


Non-Redundant Partitions

Sample Representative Partition generated using
"Width + Split" as quality

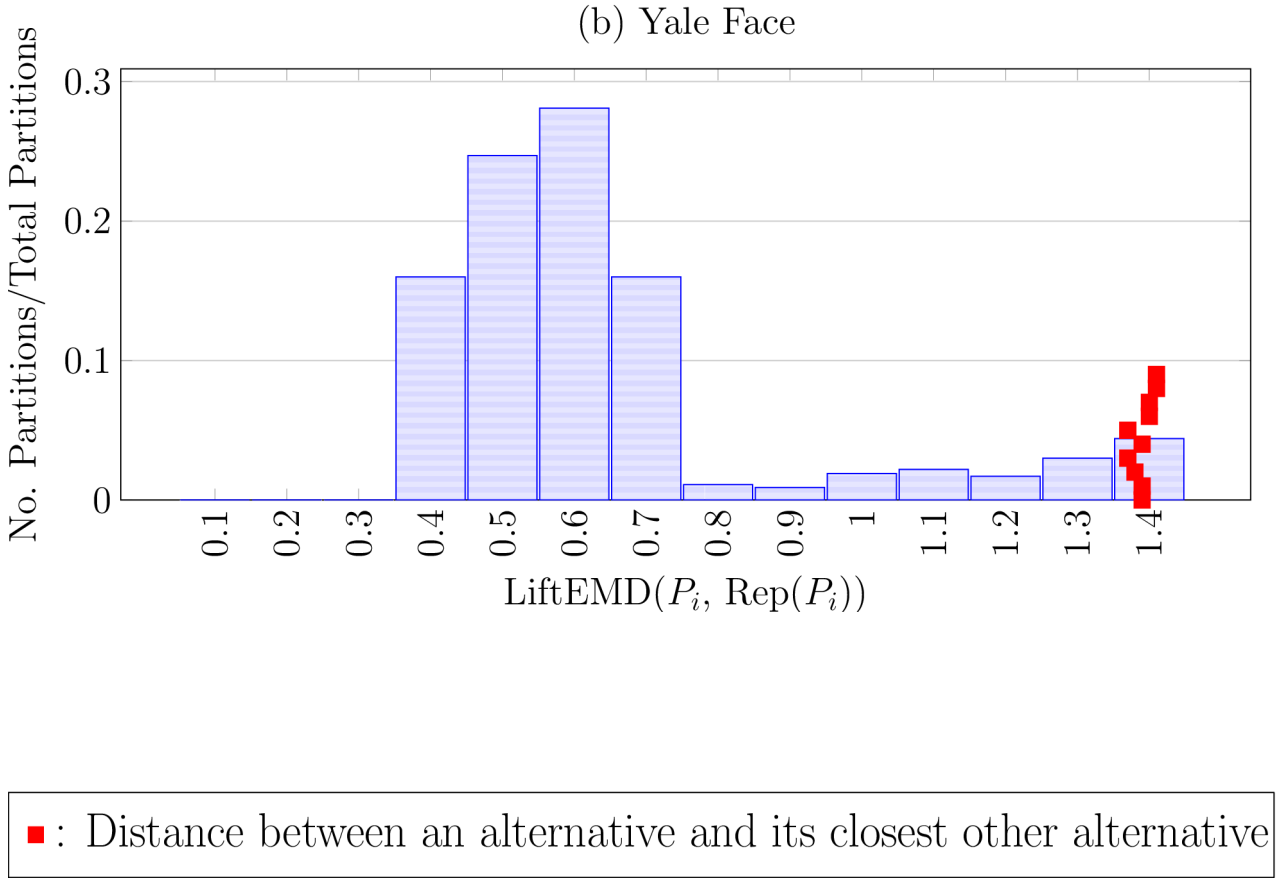


Non-Redundant Partitions

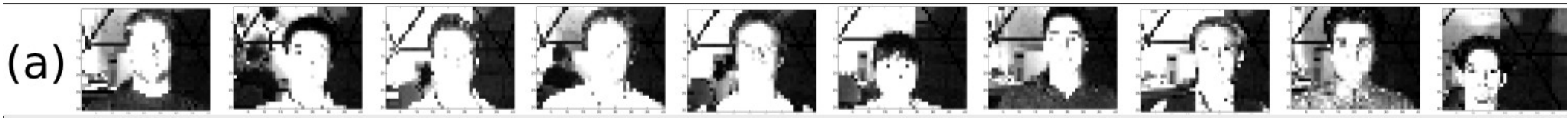


■ : Distance between an alternative and its closest other alternative

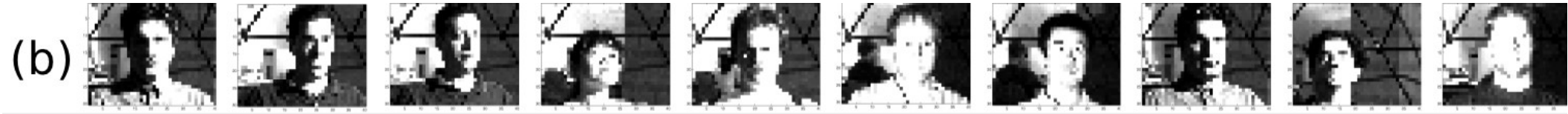
Non-Redundant Partitions



Yale Faces



Clustering A [Similar to the ground truth *'by person'*]



Clustering B [Similar to the ground truth *'by pose'*]

Two different partitions generated on the Yale Faces data

Questions: Looking at the Big Picture

- We have the landscape now
  - Tell apart different quality functions
  - Can we give supply more knowledge than what we were able to do so far?
- Beyond Consensus / Alternate?
  - Should I generate alternate partitions?
  - Do I need a consensus solution?
- What other *Multi-Clust* questions can we answer?

Questions: Looking at the Big Picture

- We have the landscape now
  - Tell apart different quality functions
  - Can we give supply more knowledge than what we were able to do so far?
- Beyond Consensus / Alternate?
  - Should I generate alternate partitions?
  - Do I need a consensus solution?
- What other *Multi-Clust* questions can we answer?

Thanks!