Finding the Number of Clusters in a Dataset: A Review and Simulation Study

Hossein Parsaei
Ph.D. Candidate
Systems Design Engineering Department

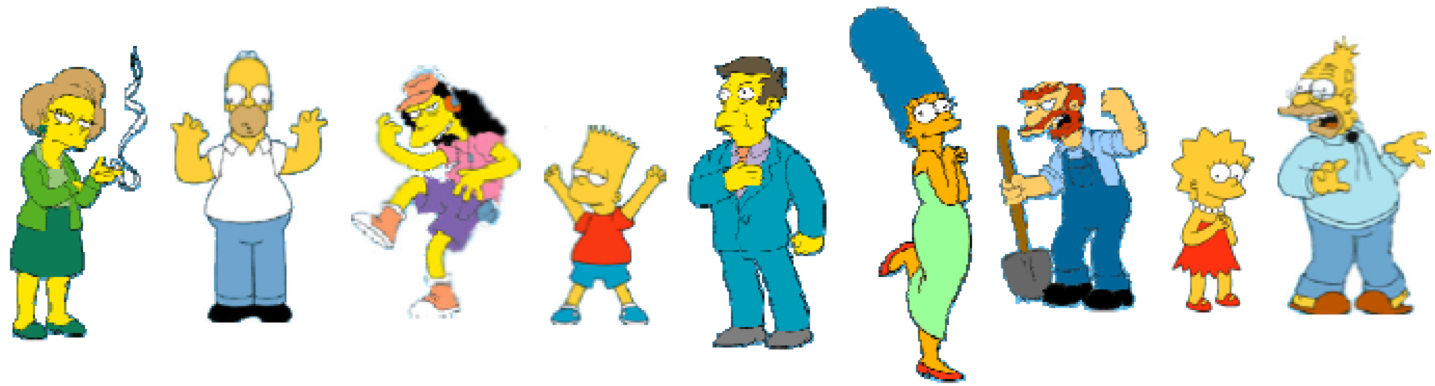
Outline

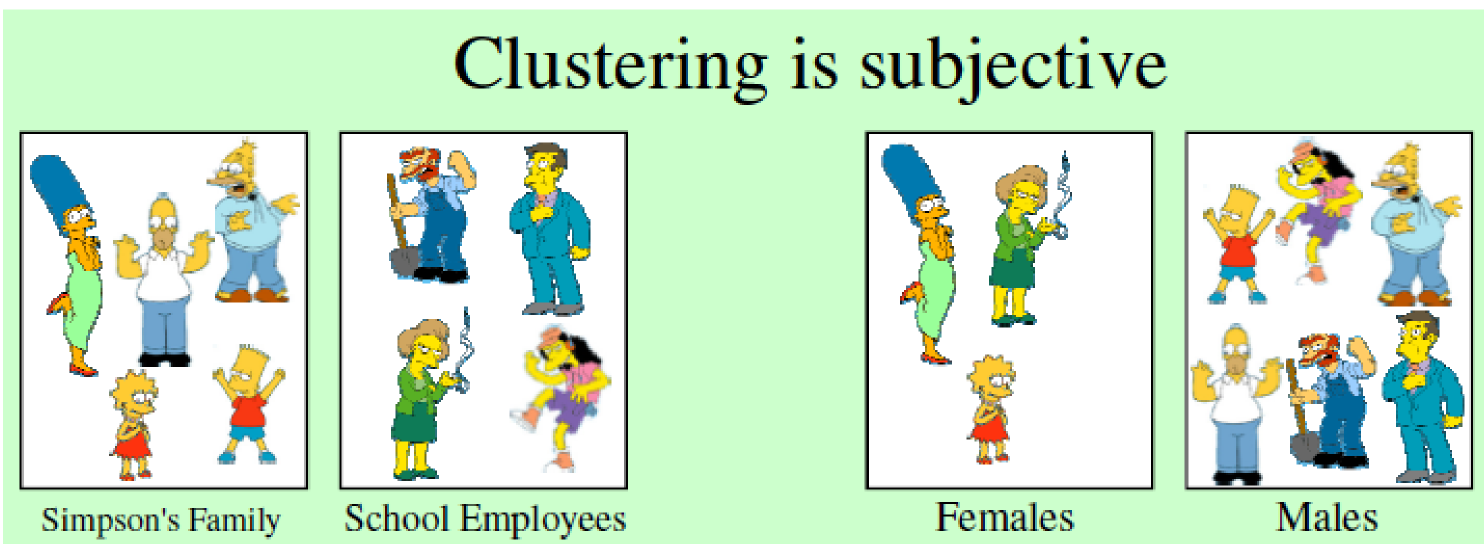
- Clustering
- Finding the number of clusters in a dataset
- Experimental results
- Conclusions

Clustering

- The process of grouping a set of objects into several clusters such that:
    1. Objects within a cluster are as similar as possible.
    2. Objects from different clusters are as dissimilar as possible.

Example




Clustering is subjective
Simpson's Family
School Employees
Females
Males

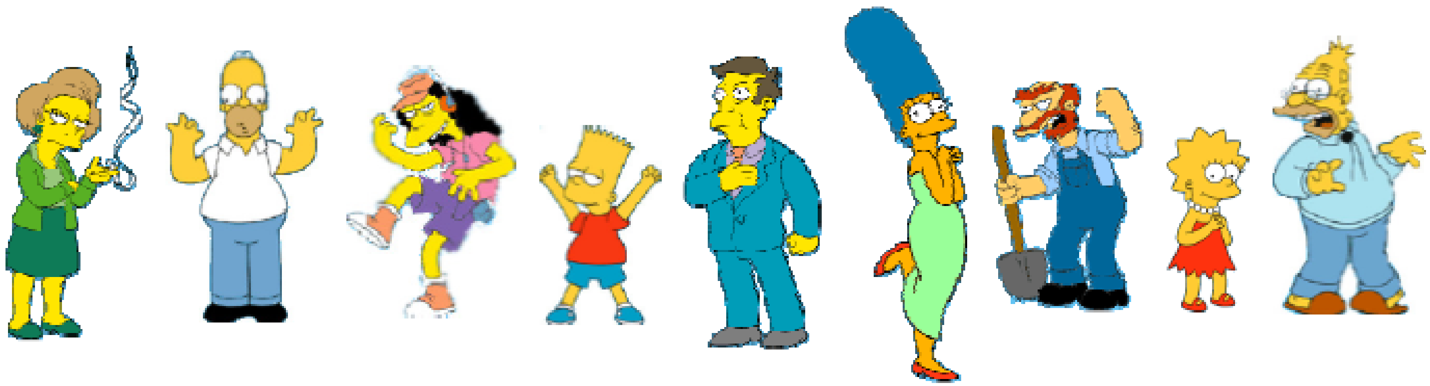
Applications

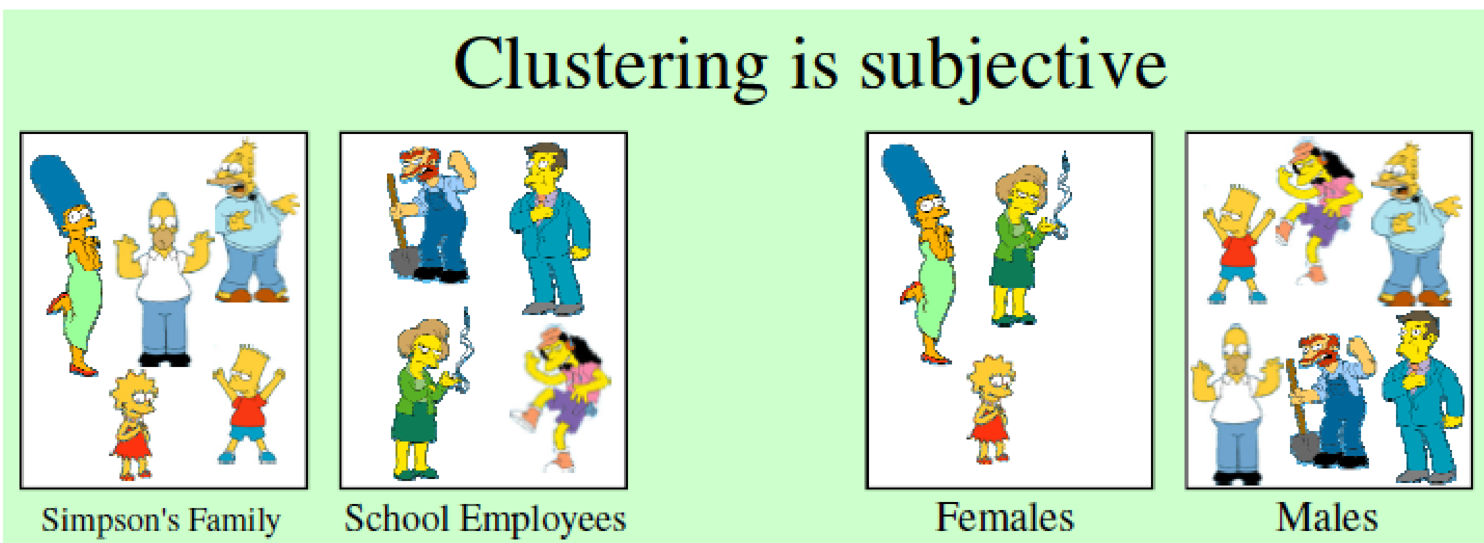
- Pattern recognition
- Bioinformatics
- Petroleum geology
- Image segmentation
- Data analysis
- Data mining

Challenges

- No prior knowledge
- Which similarity measure ?
- Which clustering algorithm?
- How to evaluate the results?
- How many clusters?

Which Result?



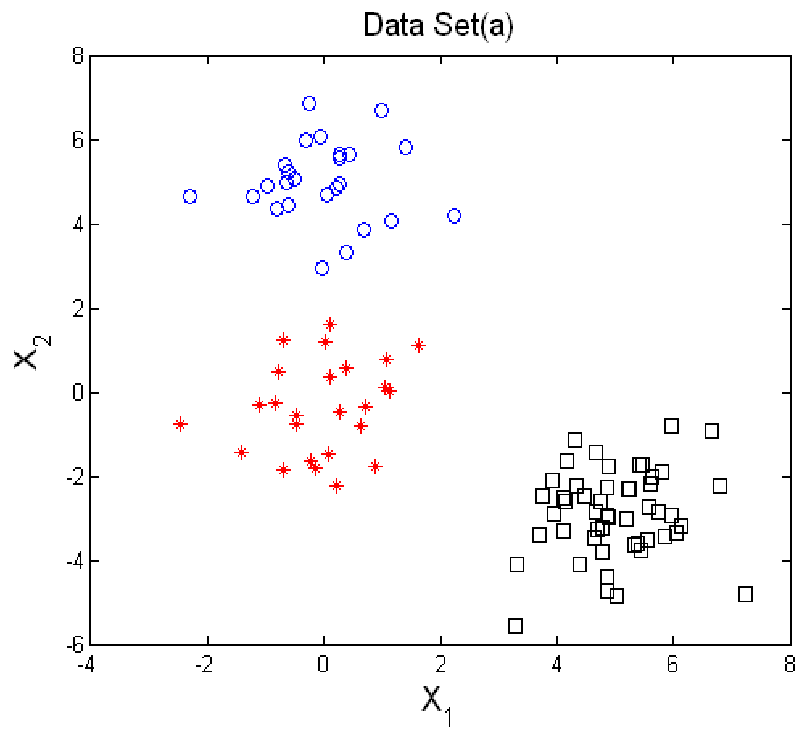


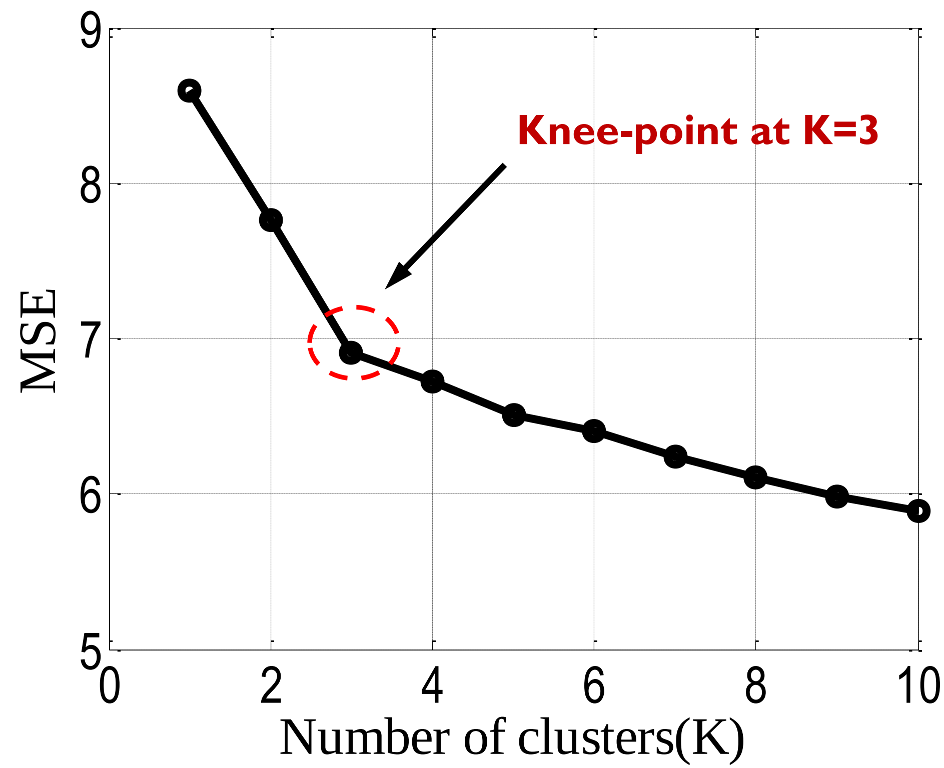
Finding the Right Number of Clusters

- Many methods have been proposed (Milligan & Cooper 1985; Gordon 1999).
- The best performs are presented here:
  1. Gap statistics
  2. Prediction strength method
  3. Jump method
  4. Calinski and Harabasz method
  5. Hartigan method
  6. Krzanowski and Lai method
  7. Silhouette statistic

How do they work ?

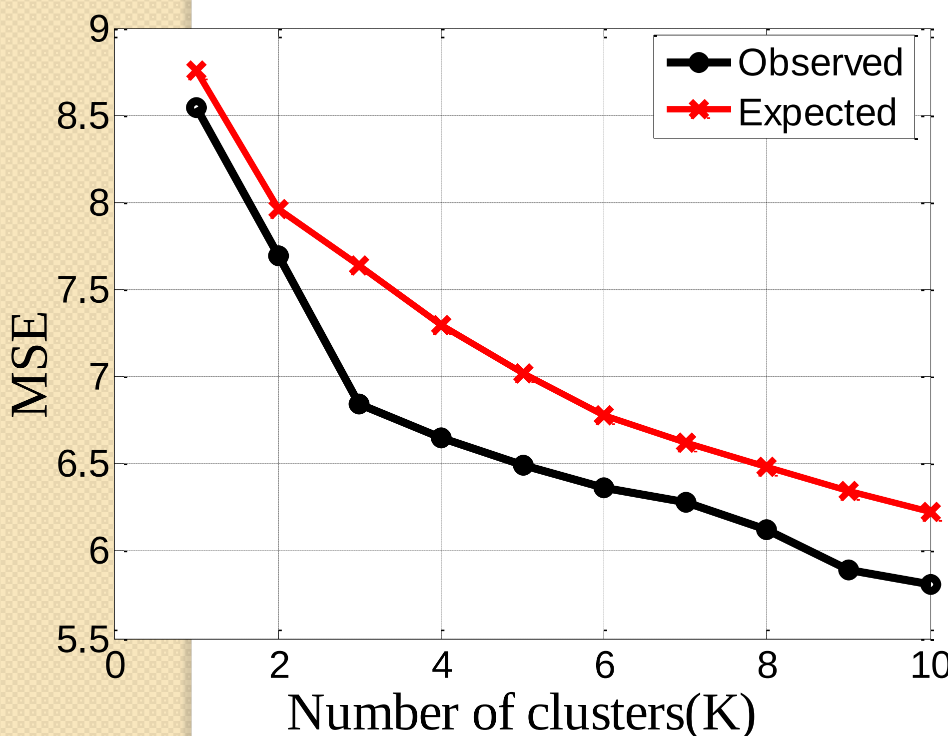
- The more clusters the better quality (e.g., the smaller the MSE/ within cluster variance).
- Small knee-point near the correct value.
- But how to detect?

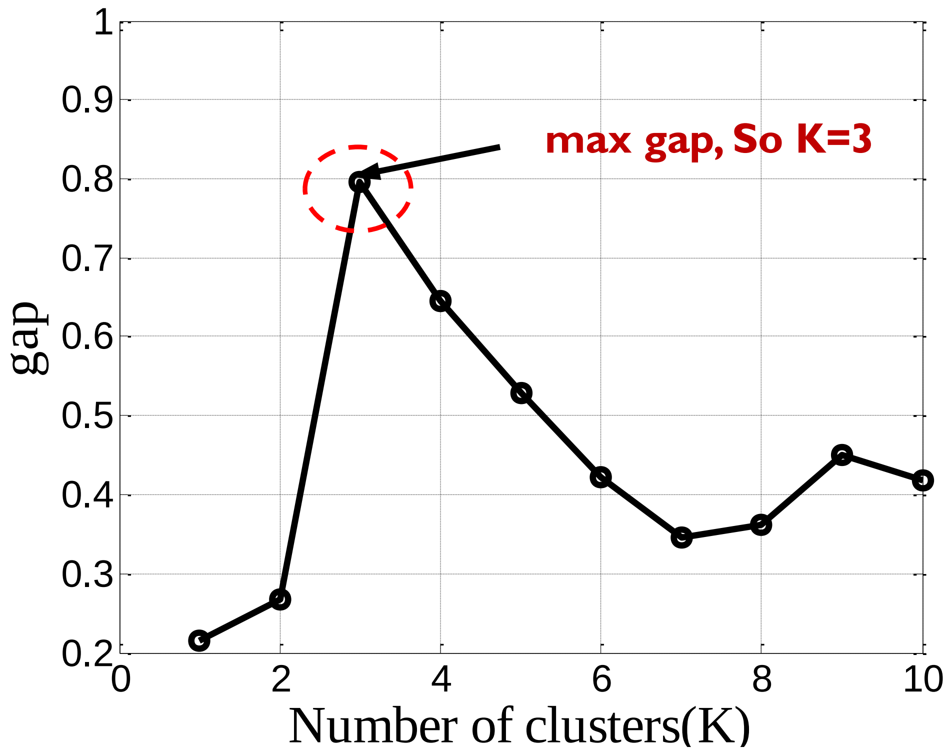




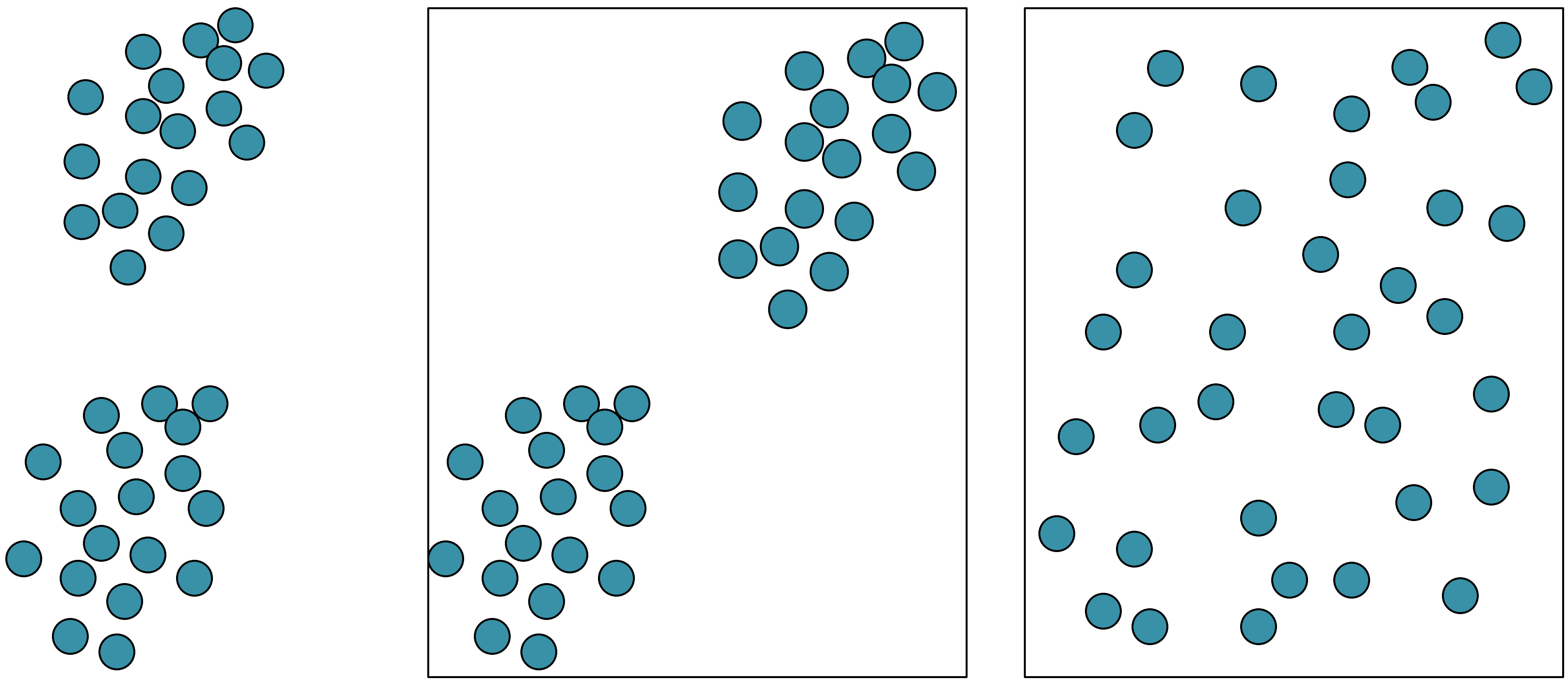
Gap Statistic

- Computes a reference for MSE vs. K
- Looks for max. gap between the two curves.





Gap Statistic: Reference Cluster

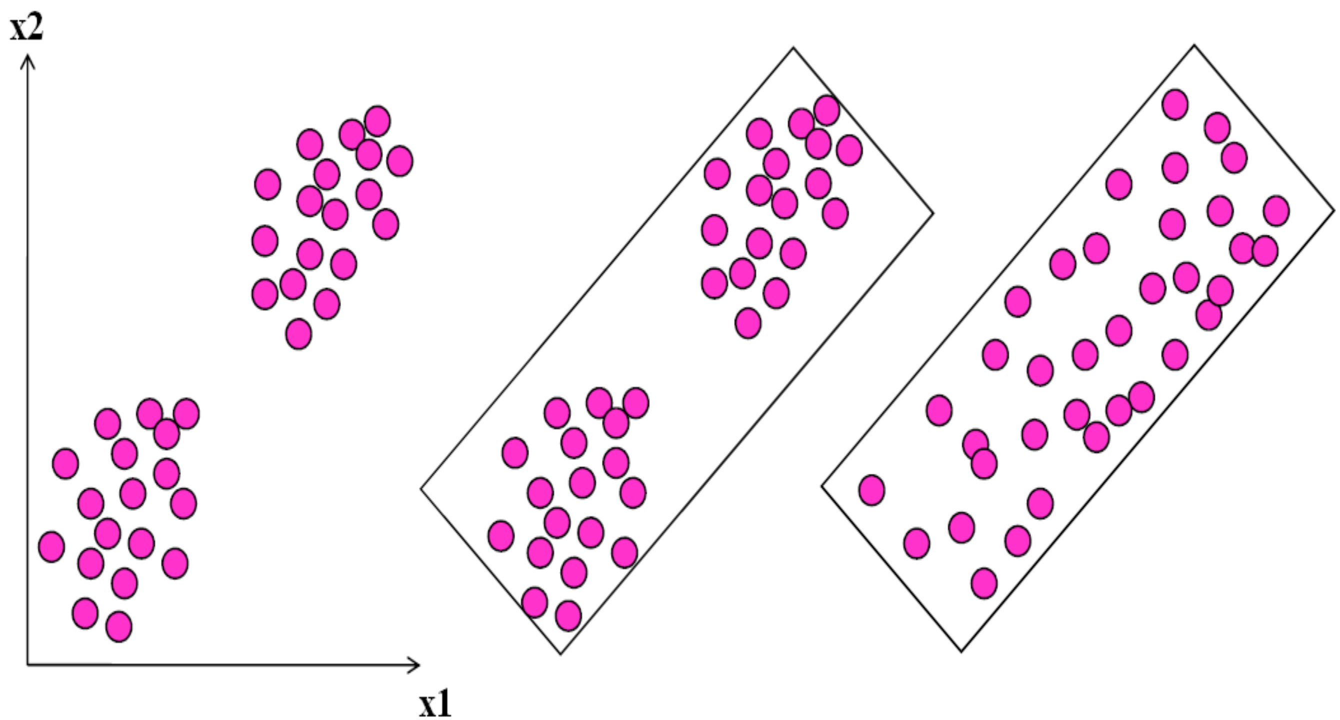


Observations

Bounding Box (aligned with feature axes)

Random Samples (reference)

Gap Statistic: Reference Cluster

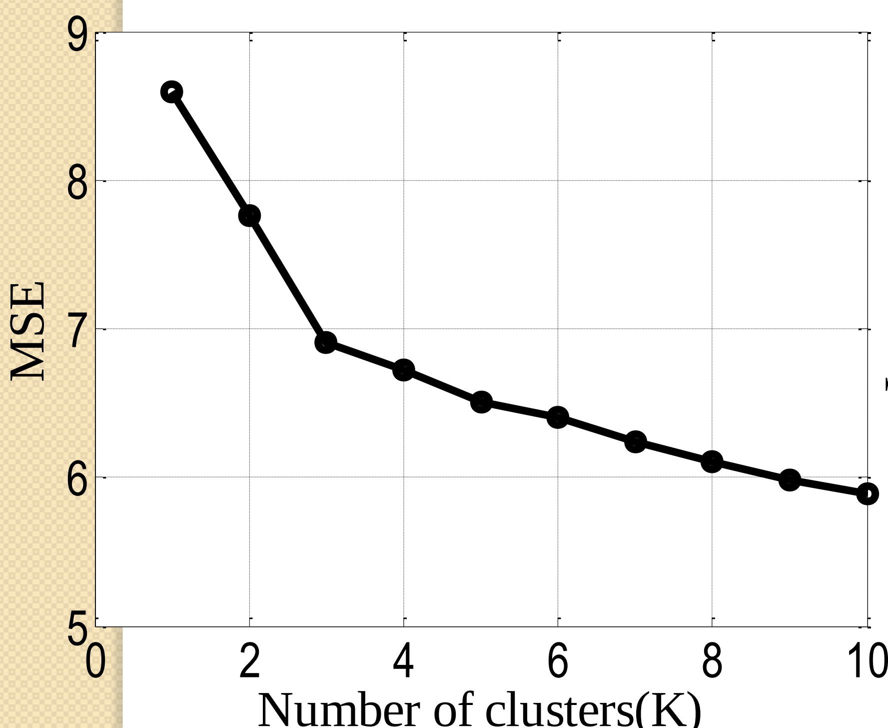


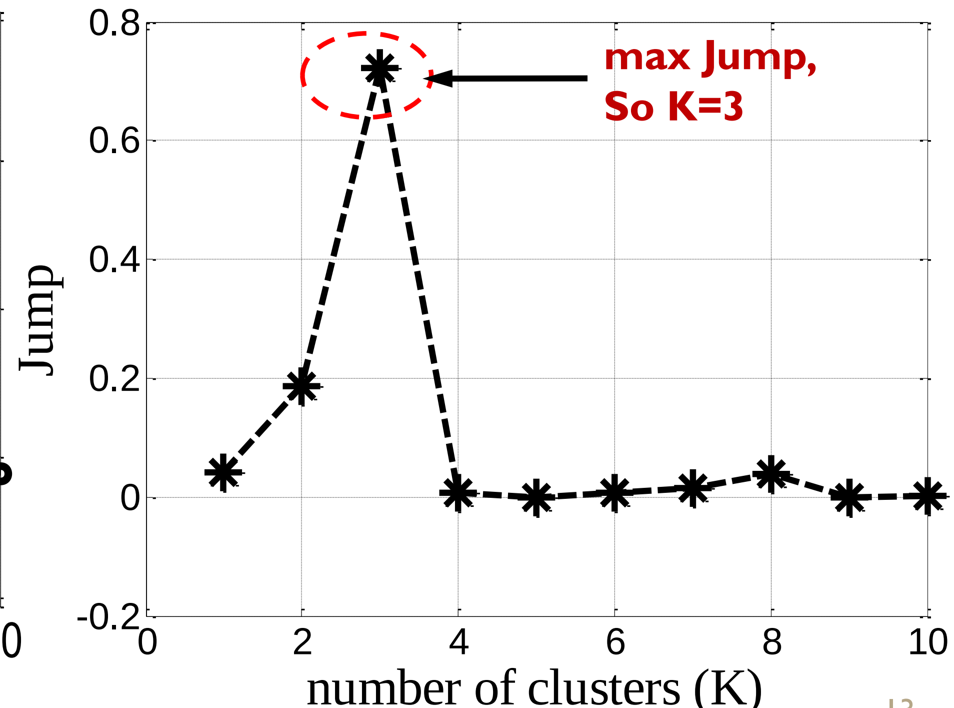
Observations Bounding Box Random Samples (reference)

Jump Method

- Transforms the curve of MSE: $Jump_k = MSE_k^{-\frac{p}{2}}$

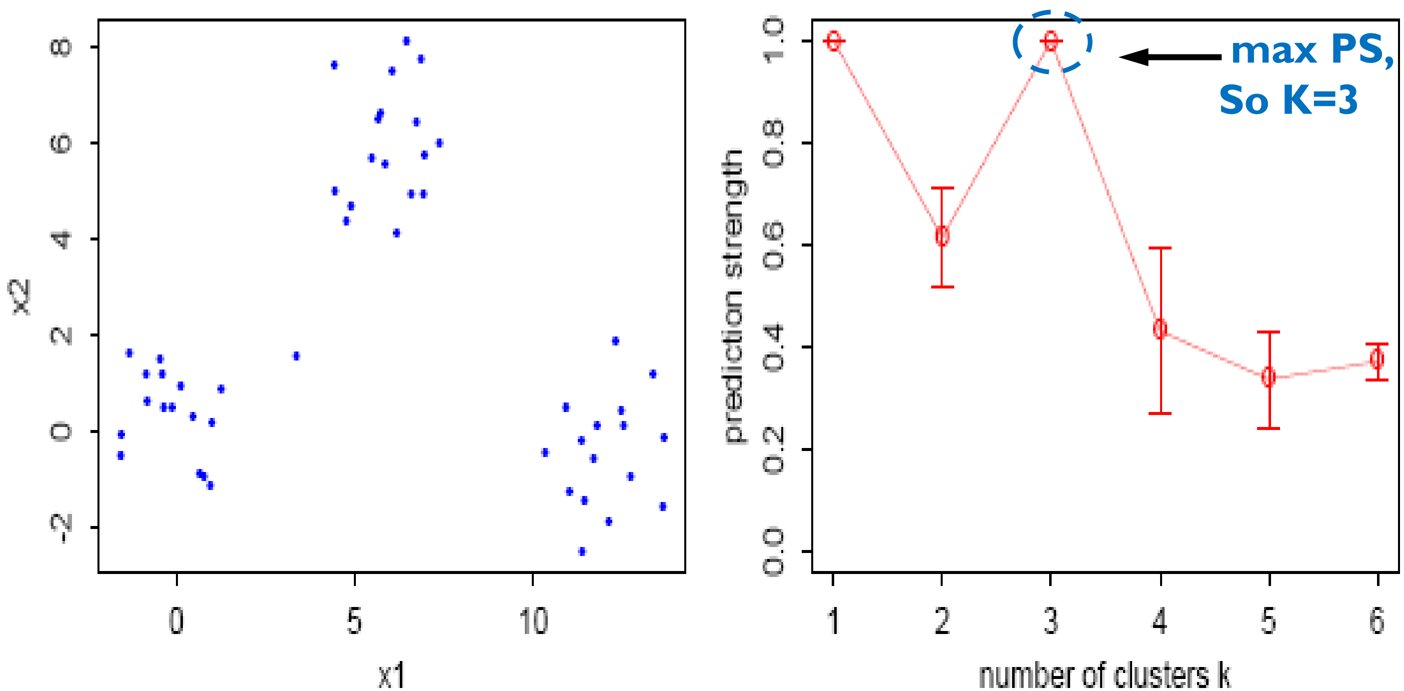
  p : effective # of features.
- Looks for max(jump)





Prediction Strength Method

- Considers clustering as a supervised classification problem
- Looks for max(prediction strength)



Prediction Strength Method

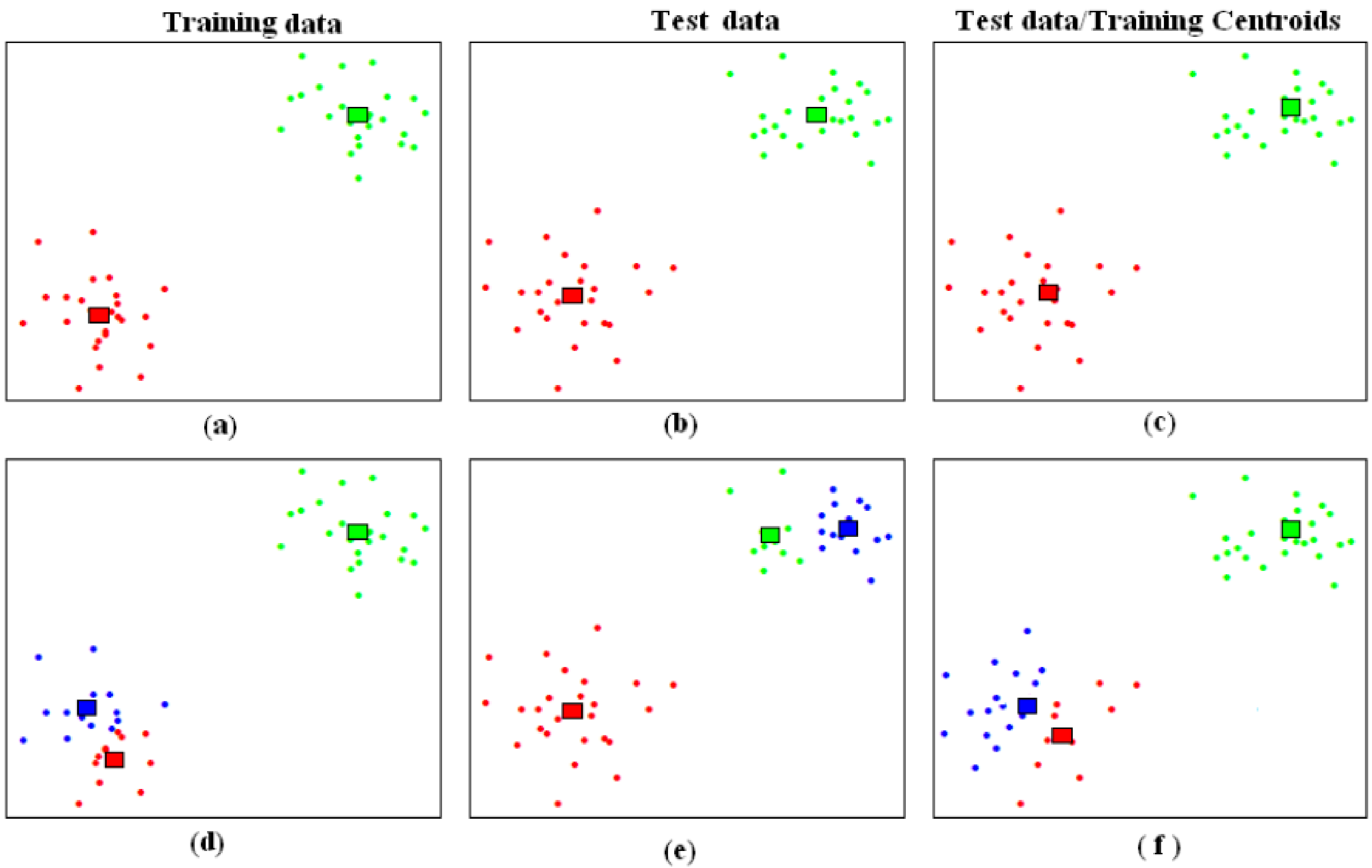


Illustration of intuition behind the prediction strength idea

Calinski and Harabasz Method

Optimum number of clusters in a data is the value of K which maximizes $\mathrm{CH}_K = \dfrac{\mathrm{trace}(\mathrm{SSB}_K)/(K-1)}{\mathrm{trace}(\mathrm{SSW}_K)/(\mathrm{N}-K)}$

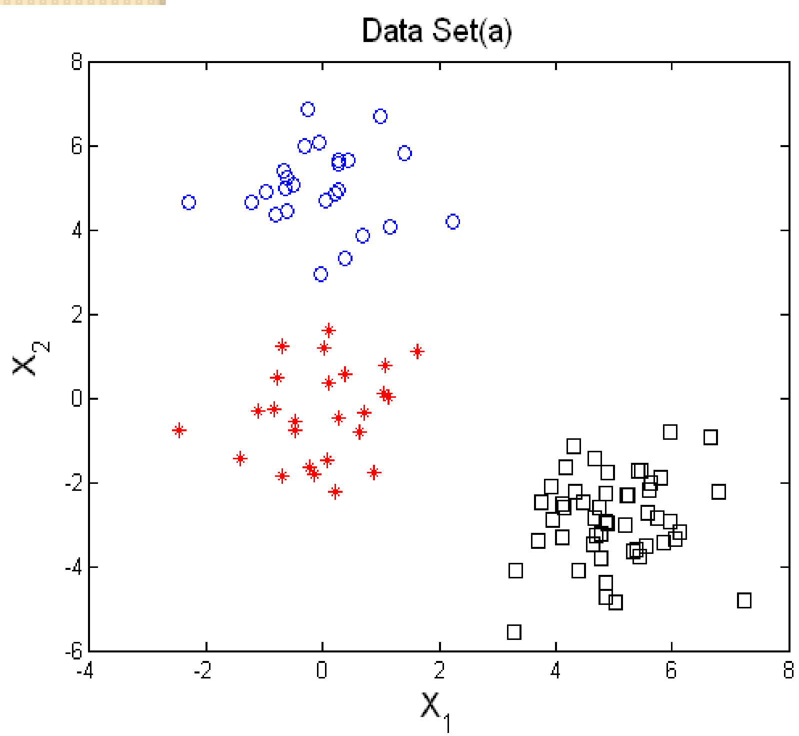
SSW: within-cluster scatter matrix
SSB: between-cluster scatter matrix

$$SSW_K = \sum_{i=1}^{K} \sum_{\underline{x} \in c_i} (\underline{x} - \underline{m}_i)(\underline{x} - \underline{m}_i)^T$$

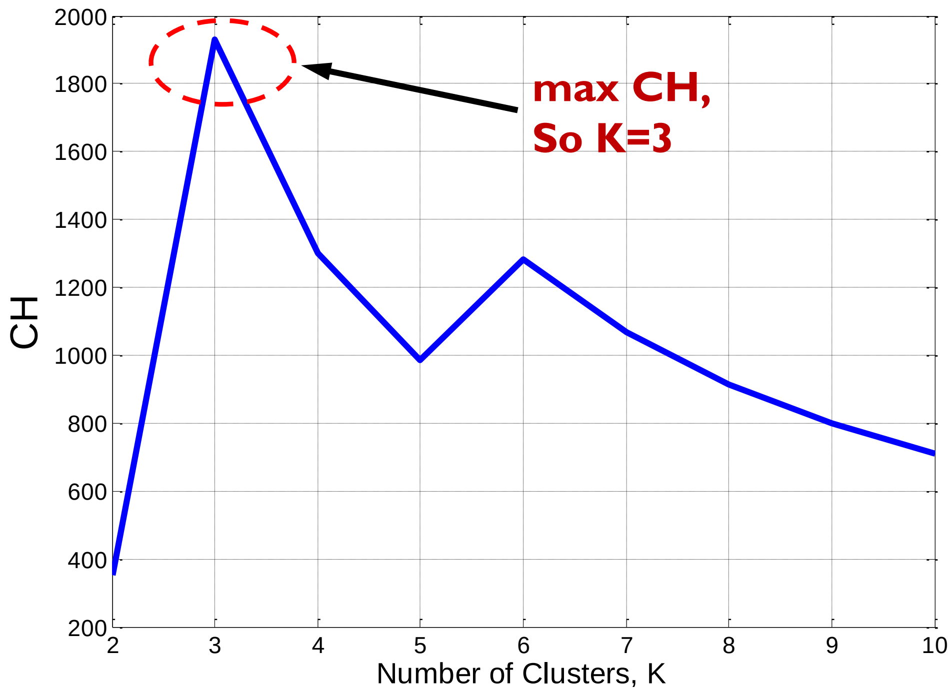
$$SSB_K = \sum_{i=1}^{K} N_i (\underline{m}_i - \underline{m})(\underline{m}_i - \underline{m})^T$$

m_i : the center of cluster C_i
m : the total sample mean of the given data.

Example for CH index



Data

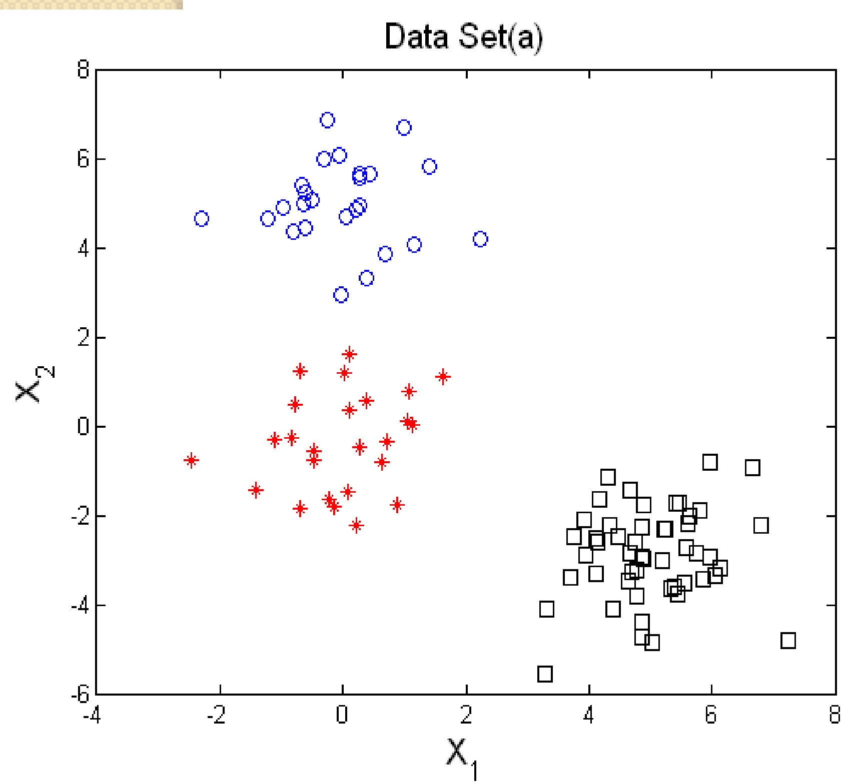


CH index Vs. K

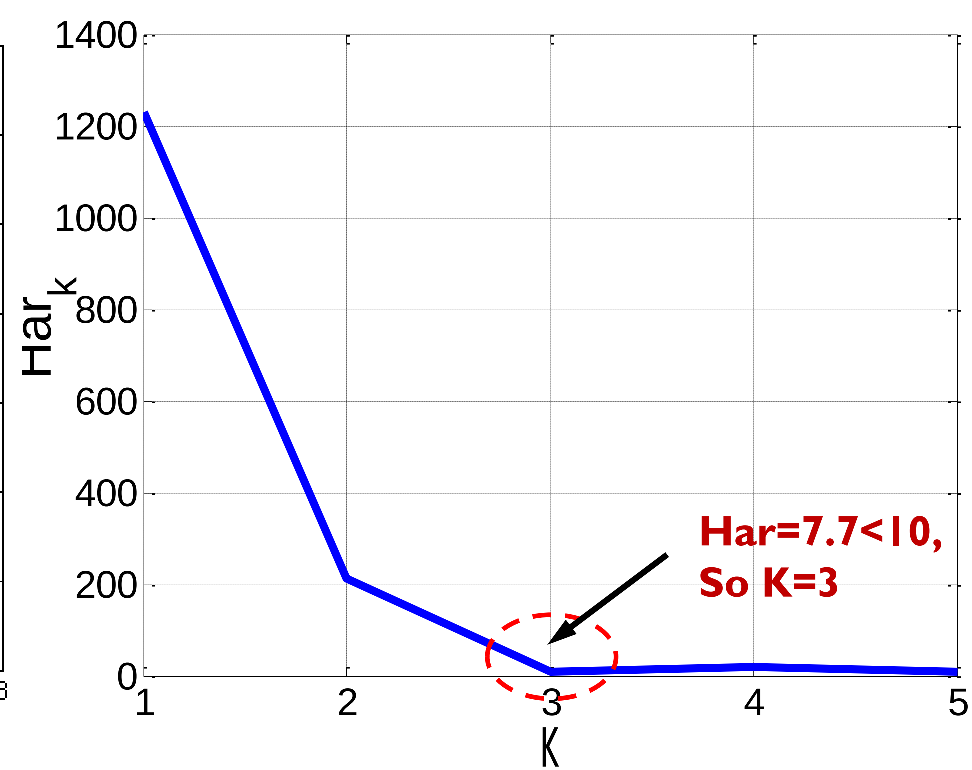
Hartigane Method

The best number of clusters is the smallest K s.t.

$$Har_K = \left[\frac{\mathrm{trace}(SSW_K)}{\mathrm{trace}(SSW_{K+1})} - 1\right] \times (N - K - 1) \leq 10$$



Data



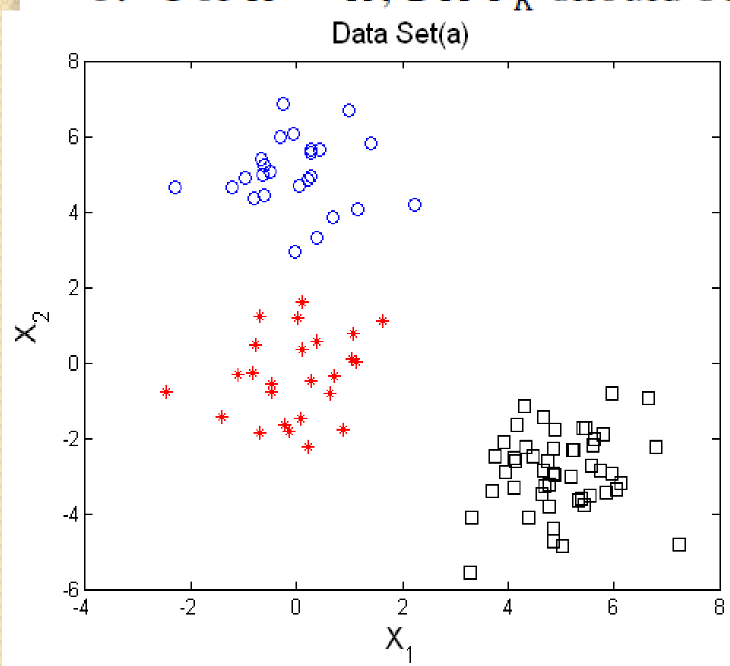
Har index Vs. K

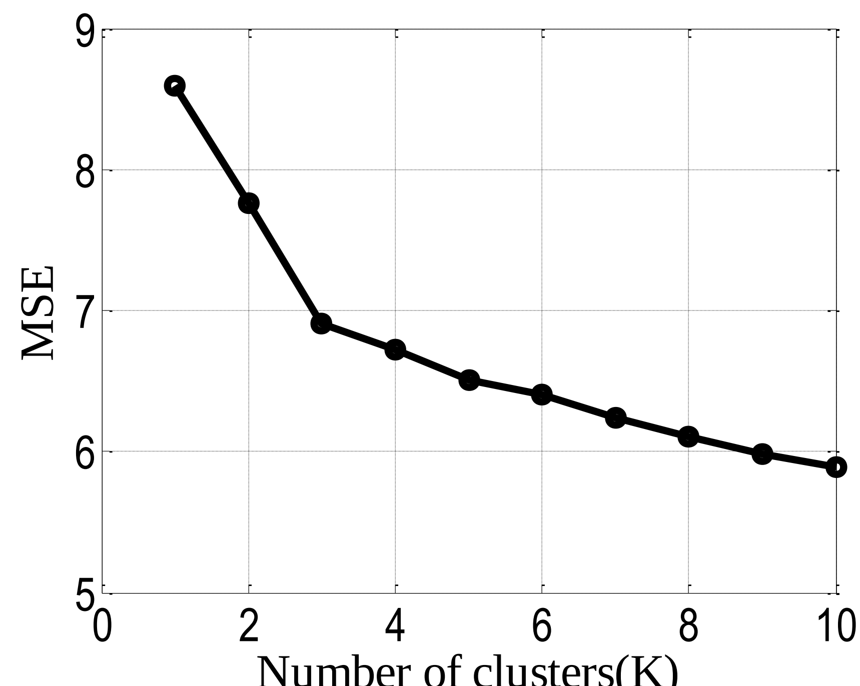
Krzanowski and Lai Method

The right number of clusters is the value of K which maximizes $KL_K = \left|\frac{DIFF_K}{DIFF_{K+1}}\right|$,

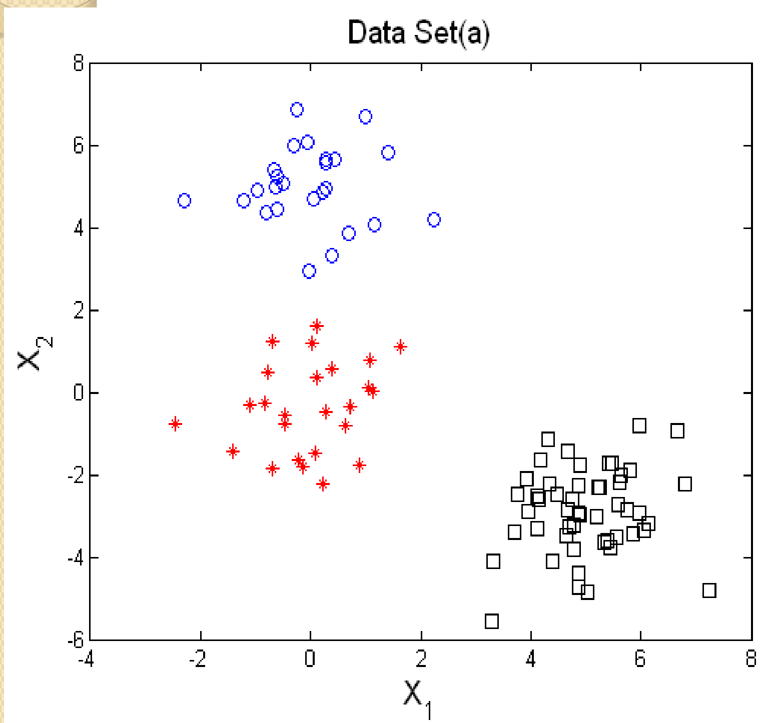
$$DIFF_K = (K-1)^{\frac{2}{p}} trace(SSW_{K-1}) - K^{\frac{2}{p}} trace(SSW_K)$$

1. For $K < \widehat{K}$, both $DIFF_K$ and $DIFF_{K+1}$ should be large.
2. For $K > \widehat{K}$, both $DIFF_K$ and $DIFF_{K+1}$ should be small.
3. For $K = \widehat{K}$, $DIFF_K$ should be large, but $DIFF_{K+1}$ should be small.

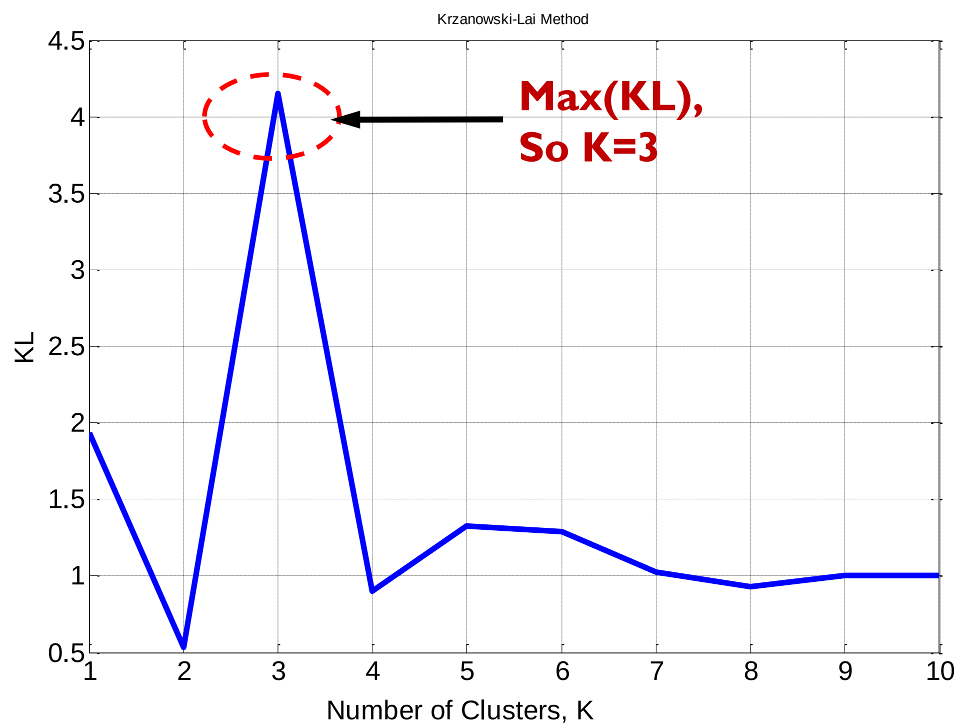




Example for KL index



Data



Kl index Vs. K

Silhouette Statistic

- First presented by Rousseeuw (1987) to show graphically how well each pattern is classified to a cluster.
- For each pattern *i* in class Cr

$$Sil_i = \frac{b(i) - a(i)}{\max\{b(i), a(i)\}}$$

a(i)= average distance to all other patterns in Cr.

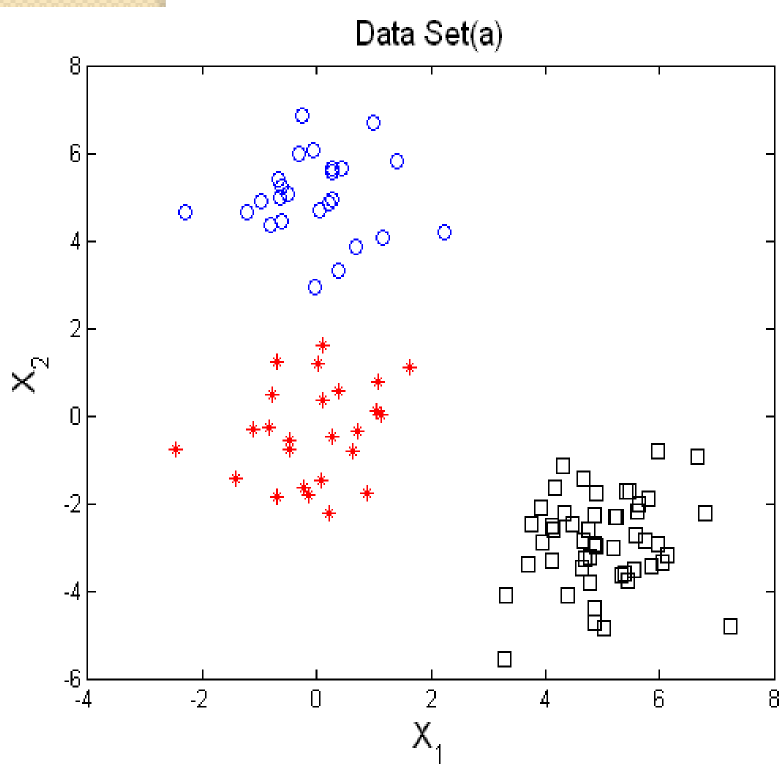
b(i)= average distance to all other patterns in other clusters.

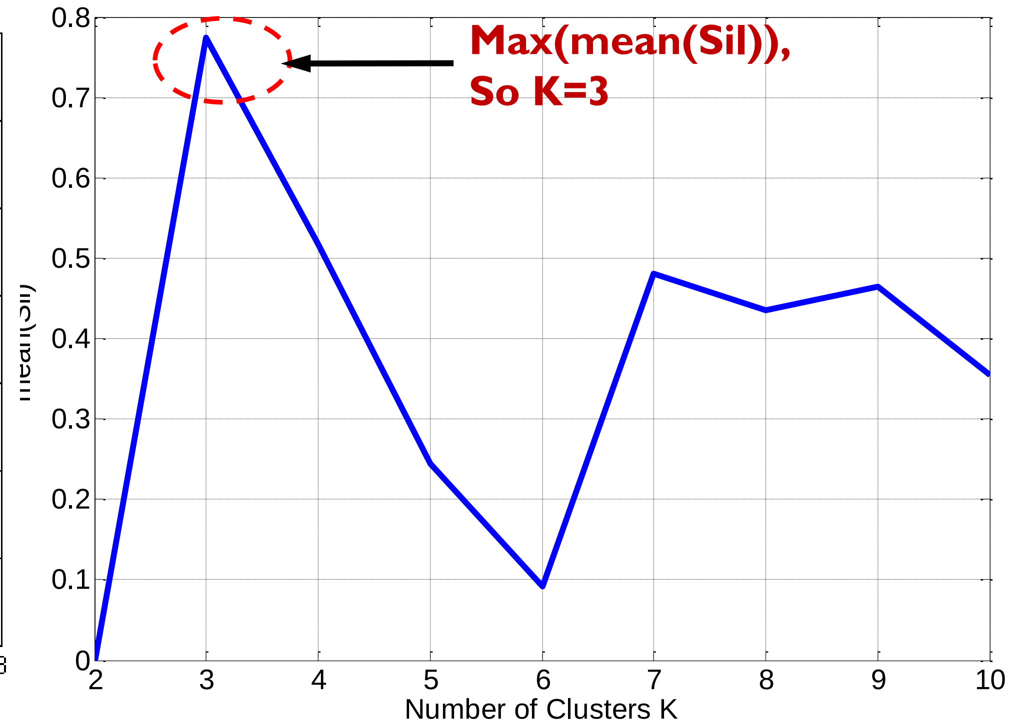
Silhouette Statistic

- . $-1 \leq Sil_i \leq 1$
- Sil=1 : good assignment
- Sil=-1: wrong (bad) assignment
- Sil=0 : don’t know ; pattern could be belong to either its current cluster or its nearest cluster.

Using Sil Index to Find the best K

- mean(Sil) reflects the within-cluster compactness and between-cluster separation of the resulting clusters
- So the best K maximizes mean(Sil)





Experiment

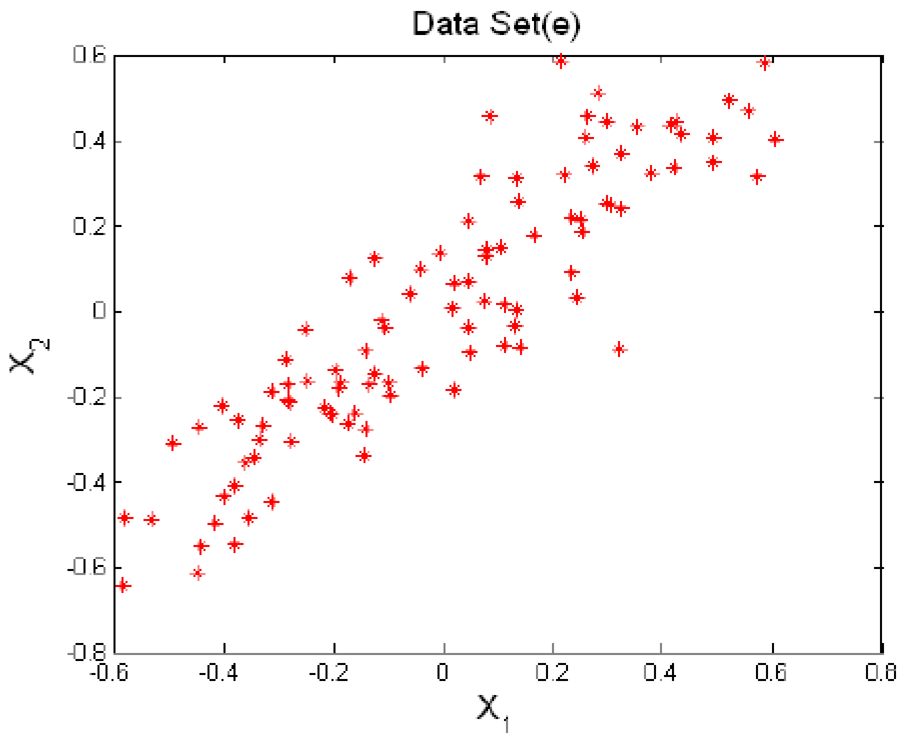
Investigate the accuracy of these 7 methods in estimating the correct number of clusters when:

1. data set contains clusters of different shapes;
2. clusters become less and less separable;
3. number of samples in a data set are decreased;
4. dimension of the data are increased.
5. data are clustered using different clustering algorithms

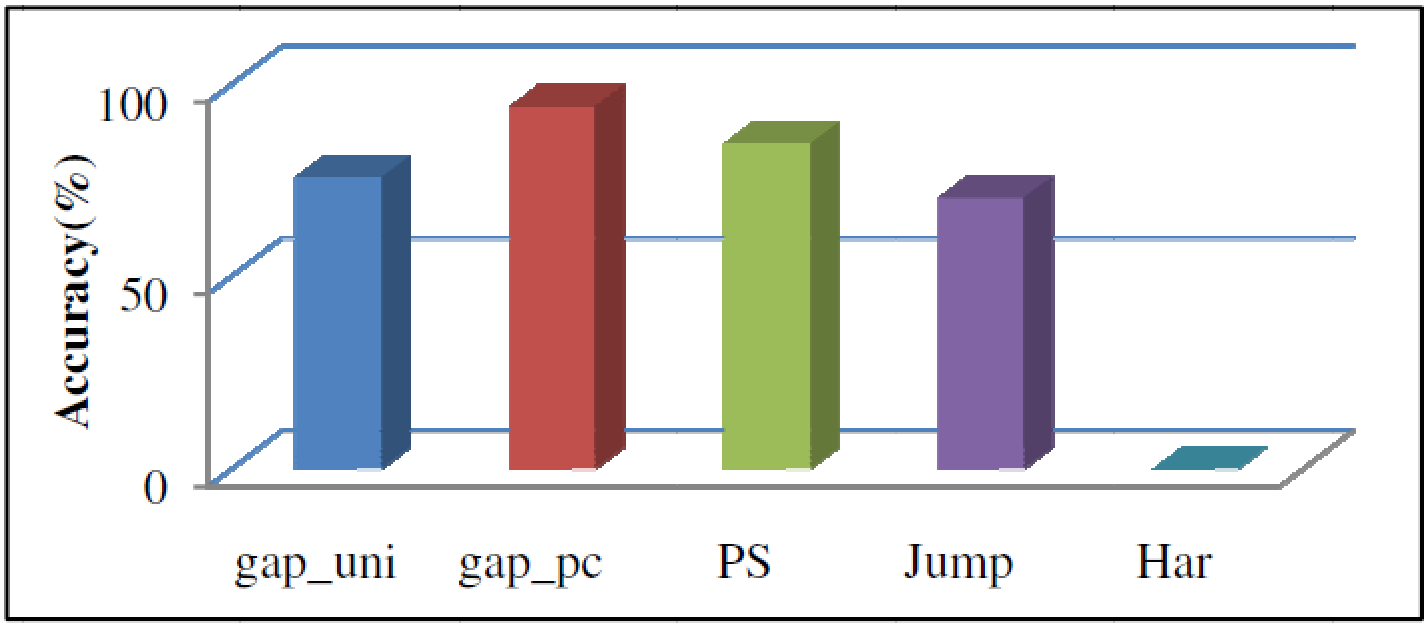
Experimental Results

- Single-cluster data set
  a. Single 2-d Gaussian distribution with zero mean and unit covariance.
  b. As (1), but the variance of two features are not equal.
  c. As (2), but with some correlation.
  d. Uniform distribution U[0,1]
  e. Highly correlated features: x = y = t + z with t increasing by 0.01 from -0.5 to 0.5, z is Gaussian noise.

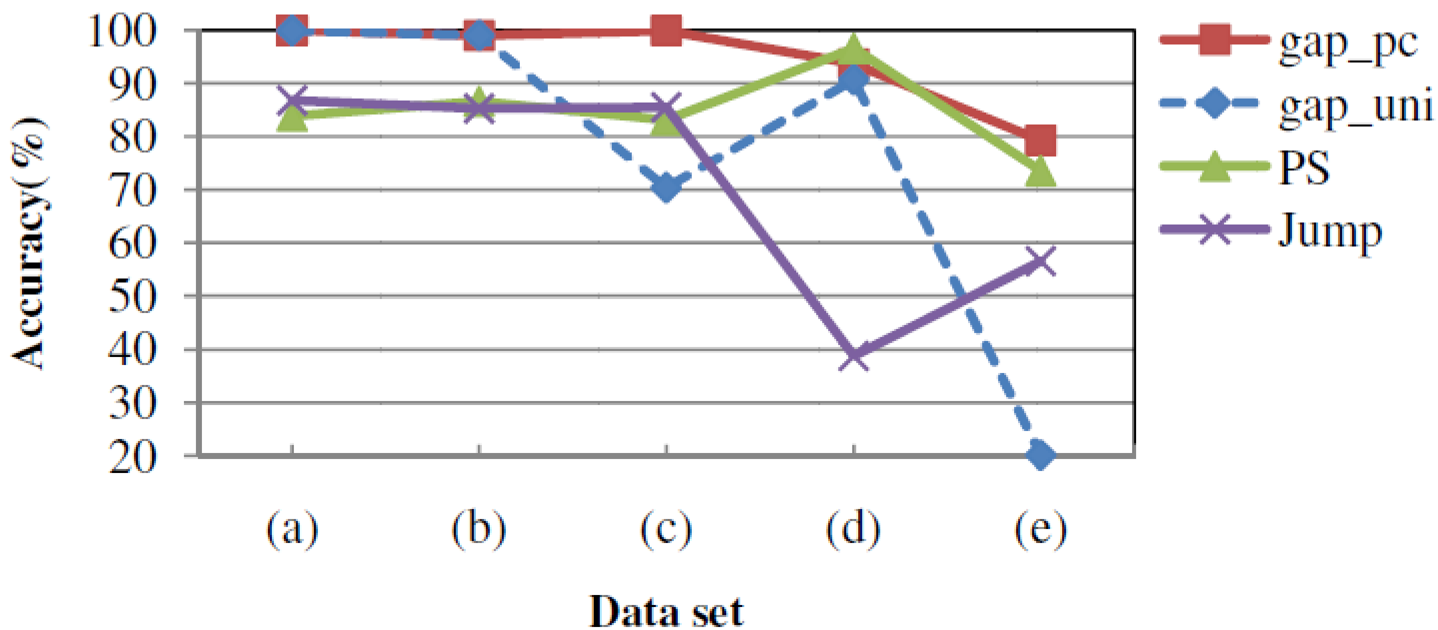
Data set 5



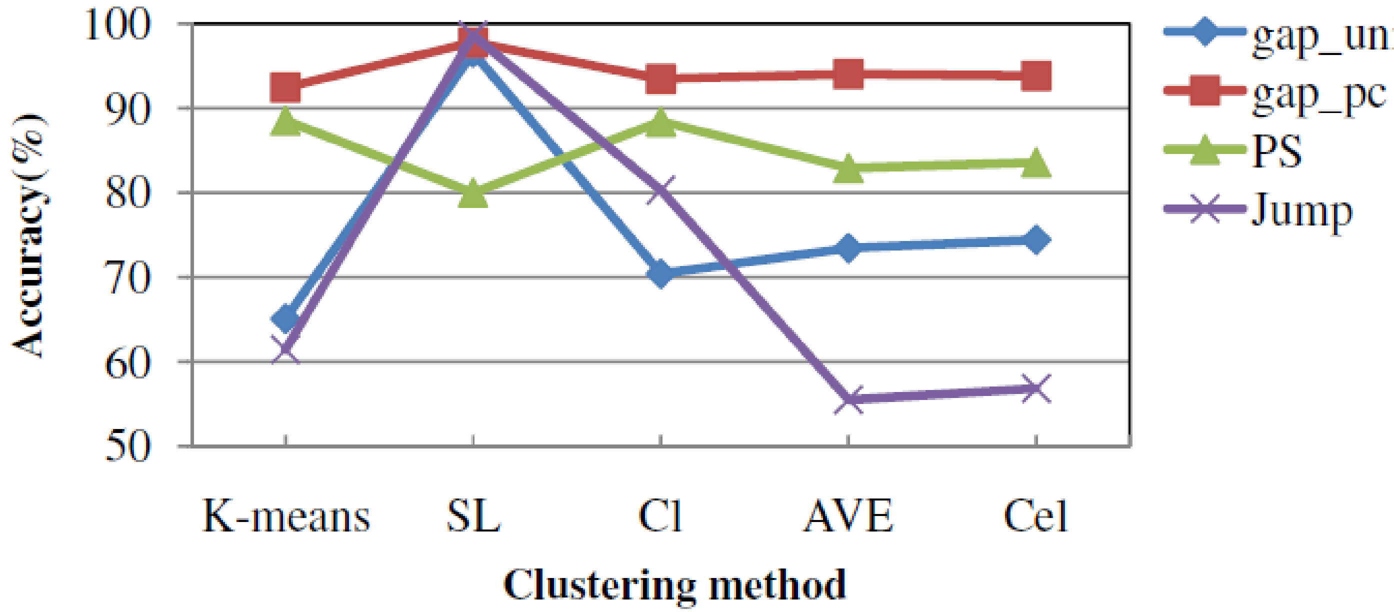
Overall accuracy



Accuracy Vs. Dataset Used

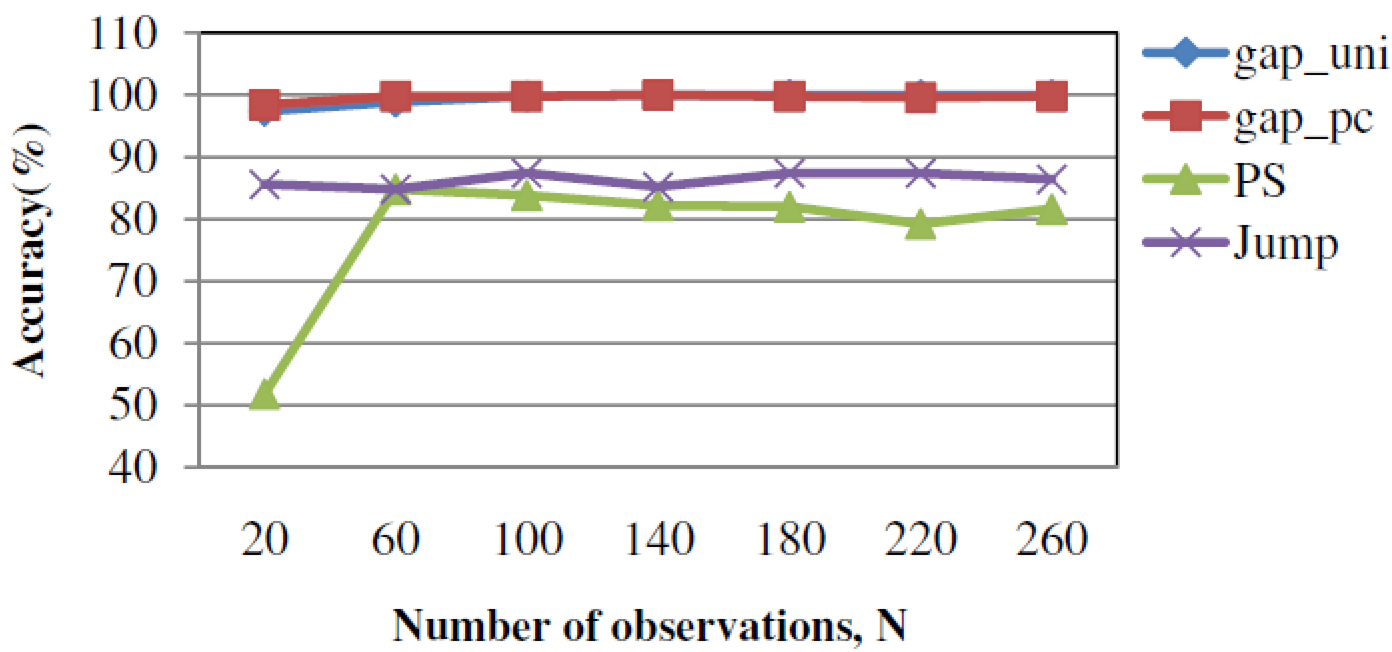


Accuracy Vs. Clustering Method

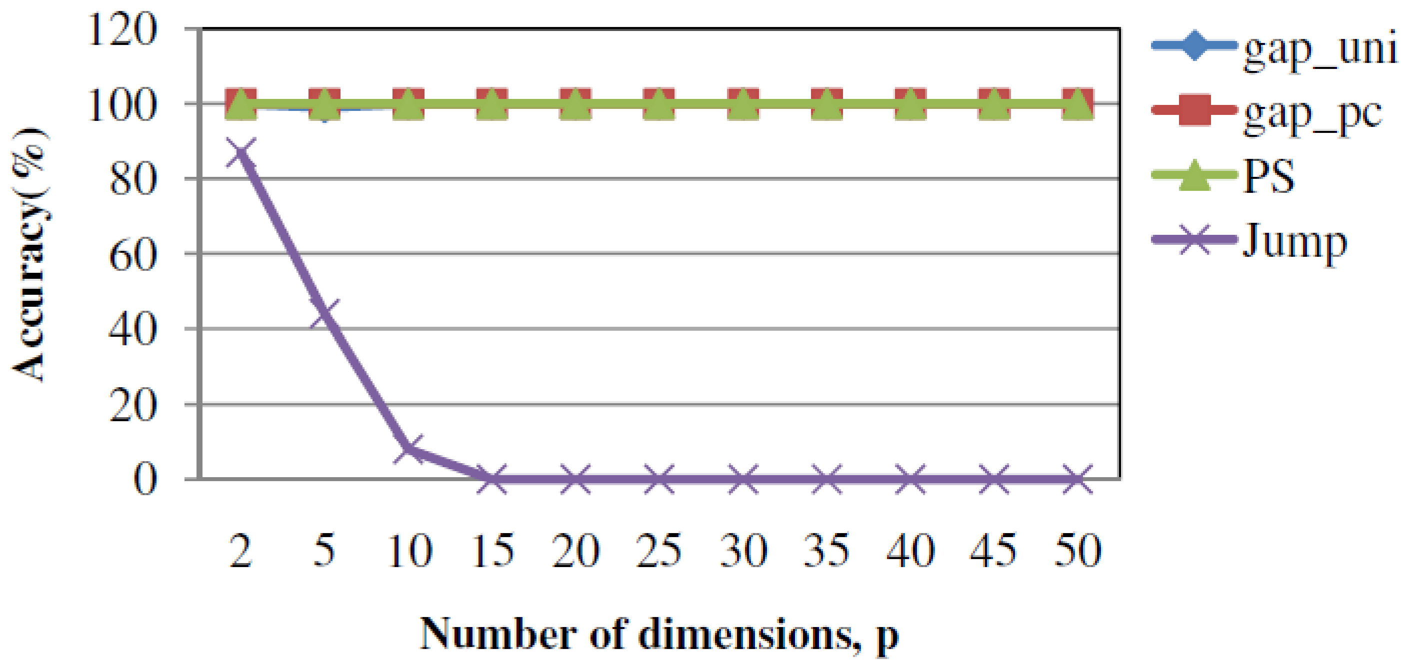


Sl: single-link
Cl: complete-link
AVE: average link
CL: centroid link

Accuracy Vs. Number of Patterns

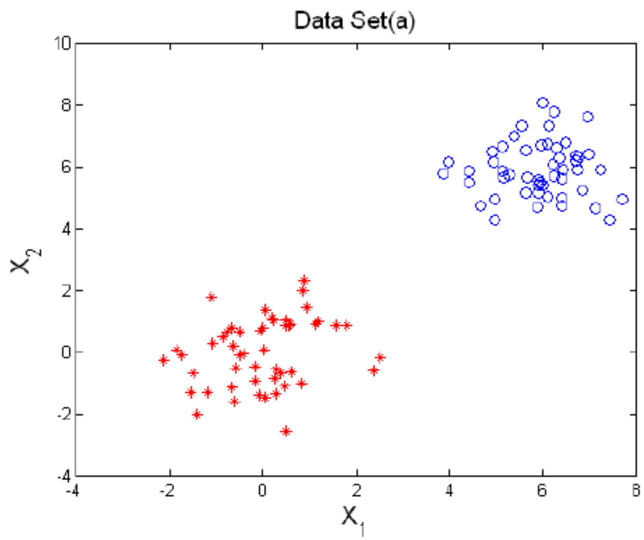


Accuracy Vs. Dimension of Data

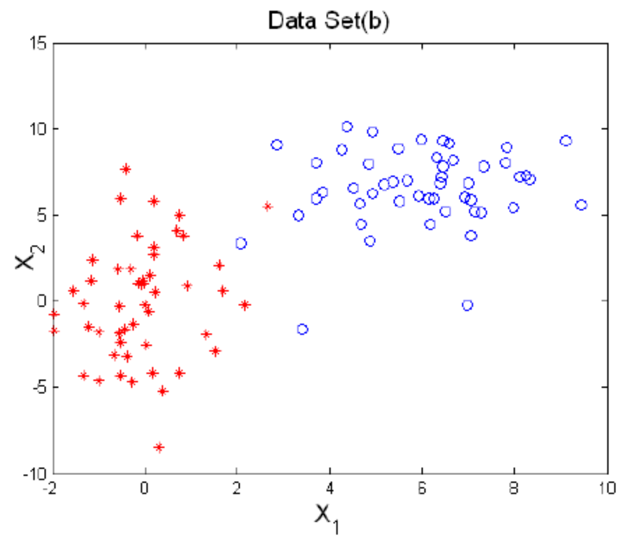


$$Jump_k = MSE_k^{-\frac{p}{2}}$$

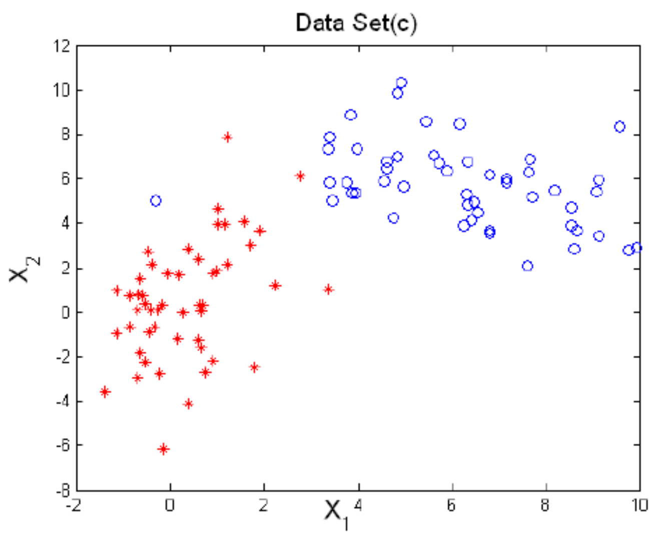
2-cluster dataset



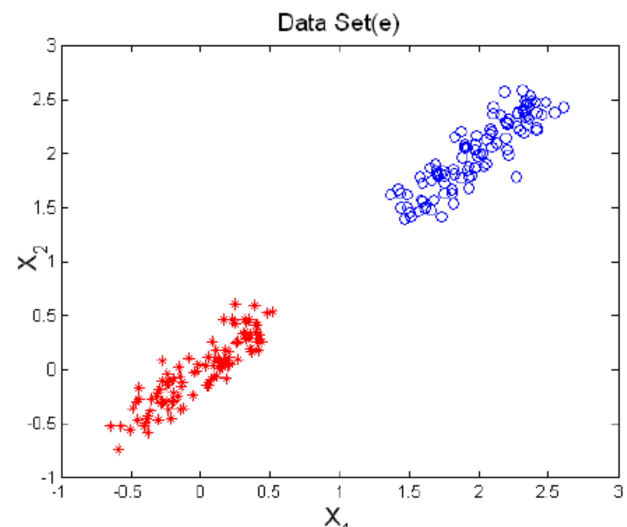
(a)



(b)

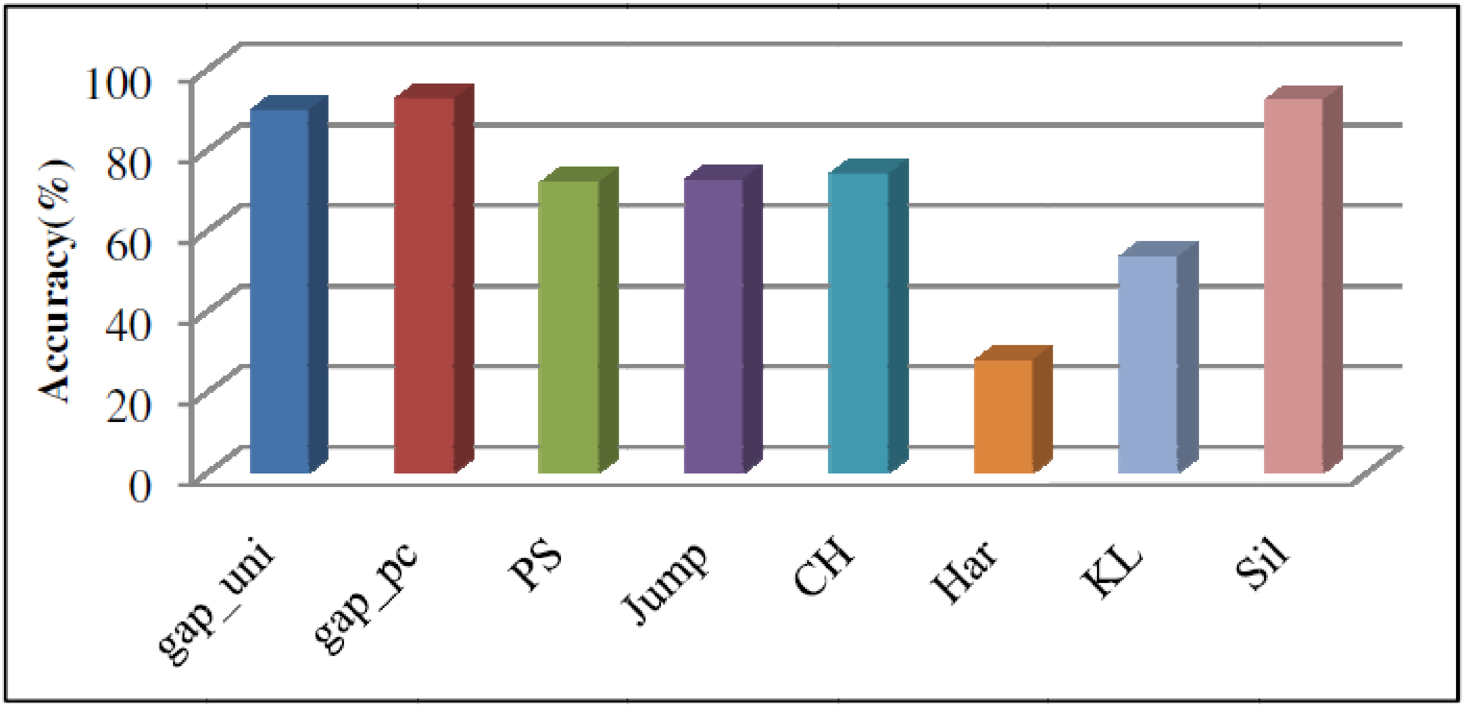


(c)

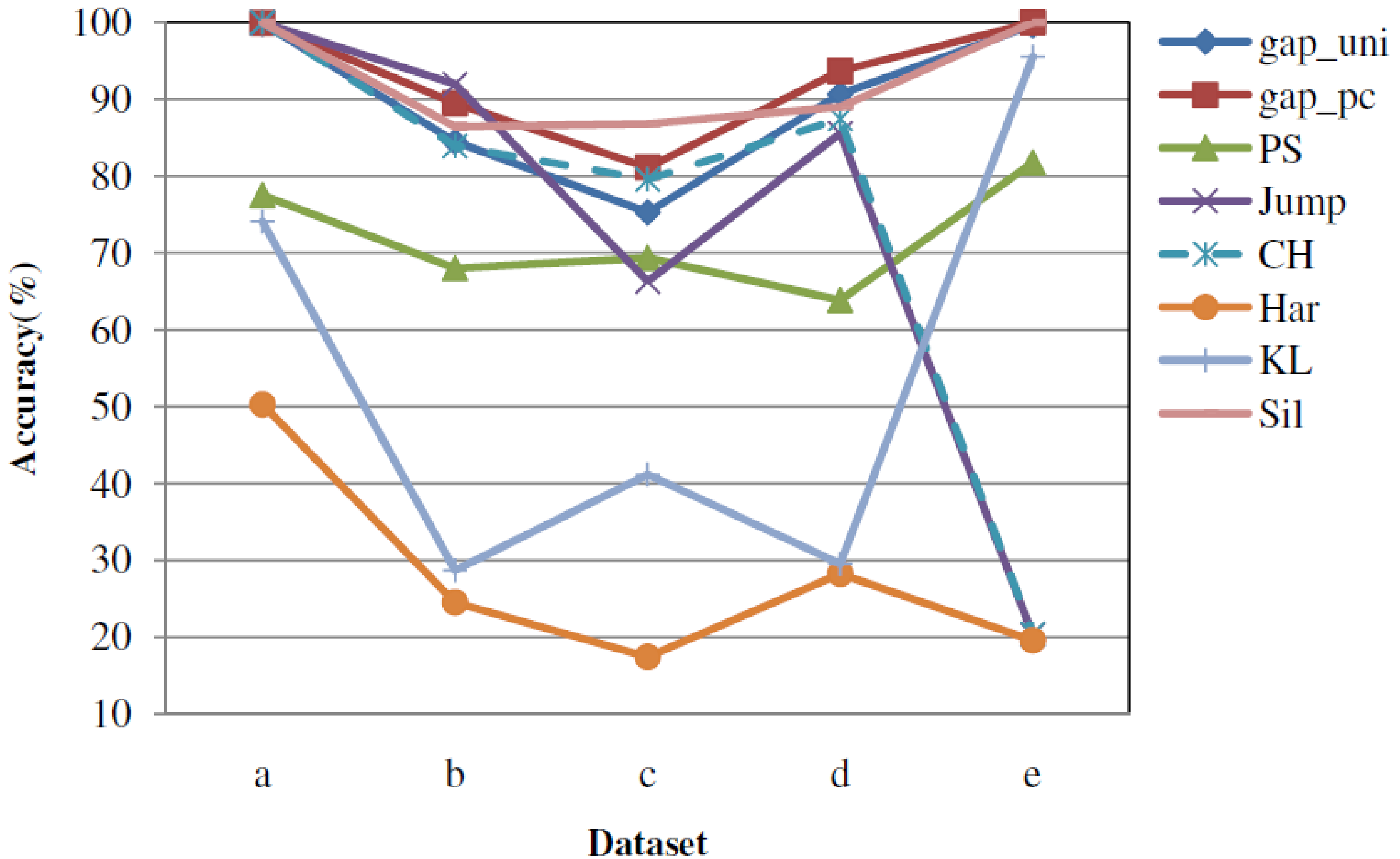


(d)

Overall Accuracy



Accuracy Vs. Dataset Used



Overlapping Clusters

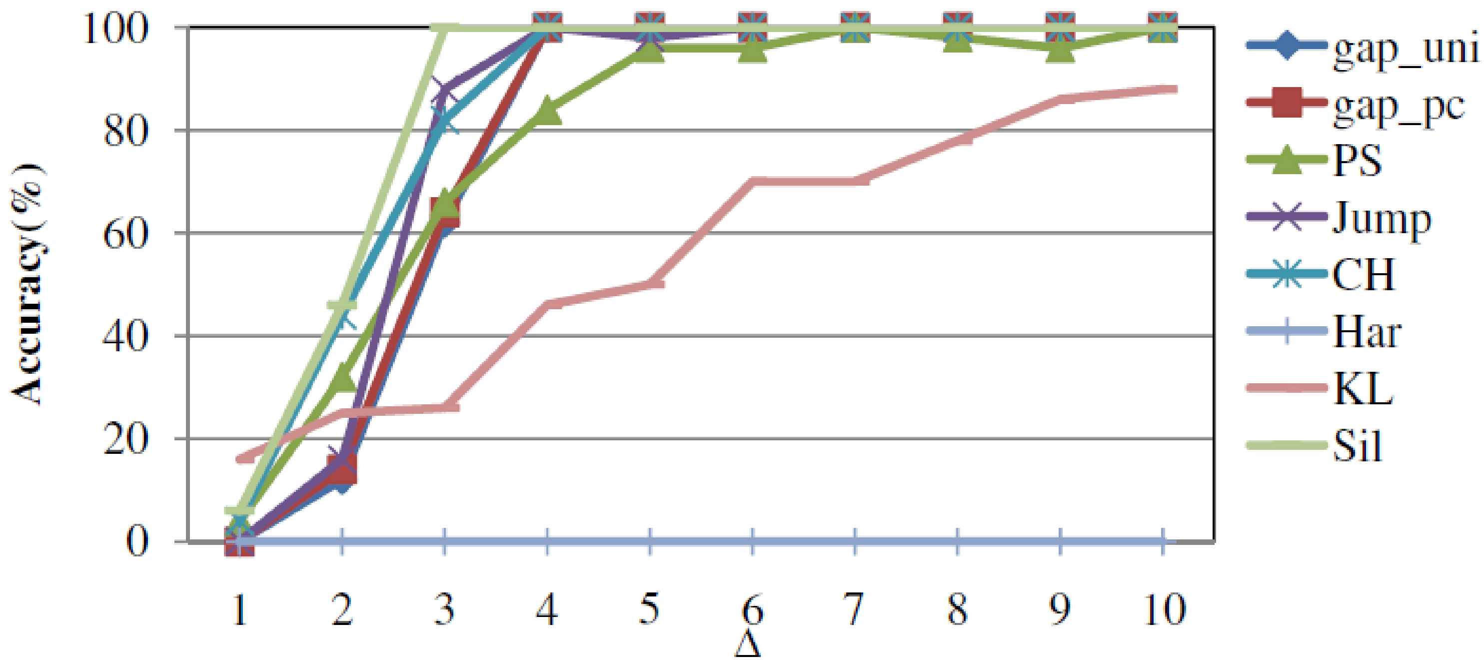


Figure 15. Results for overlapping clusters. High value of Δ show well separation between the two clusters.

Unbalanced Clusters

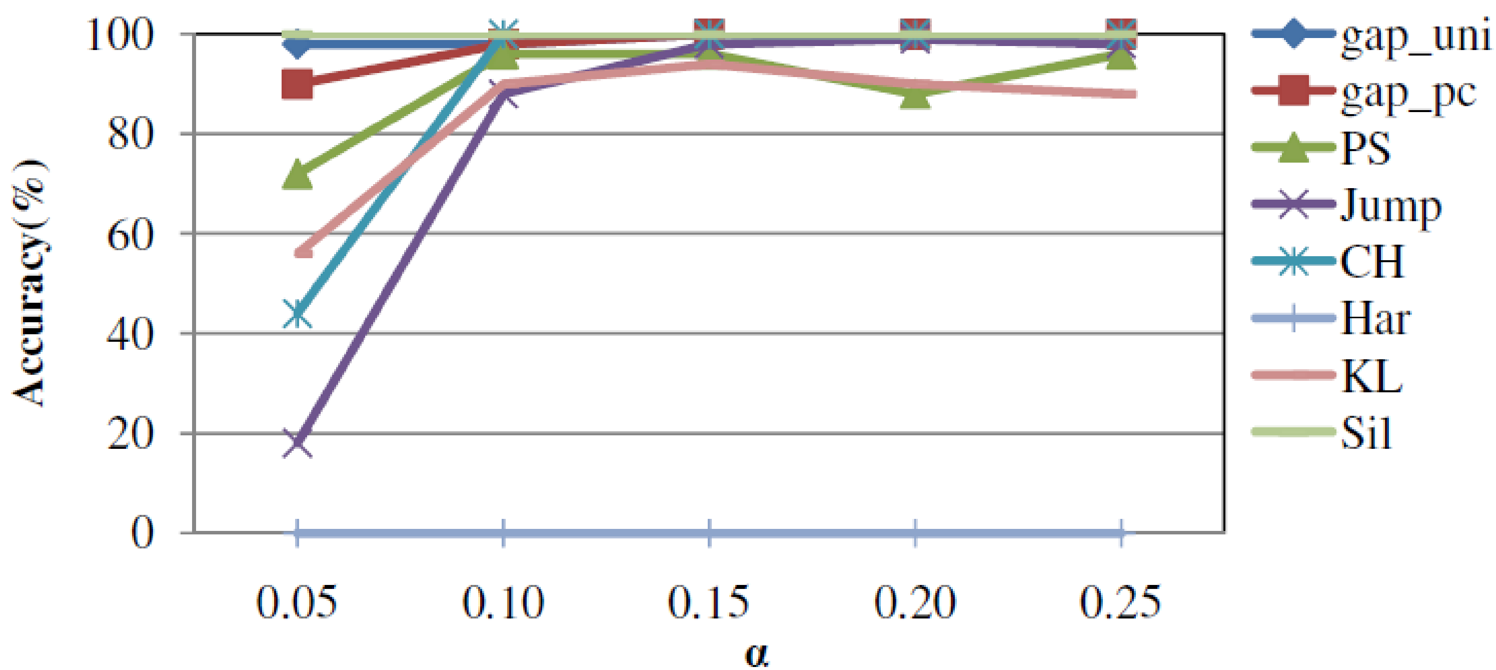
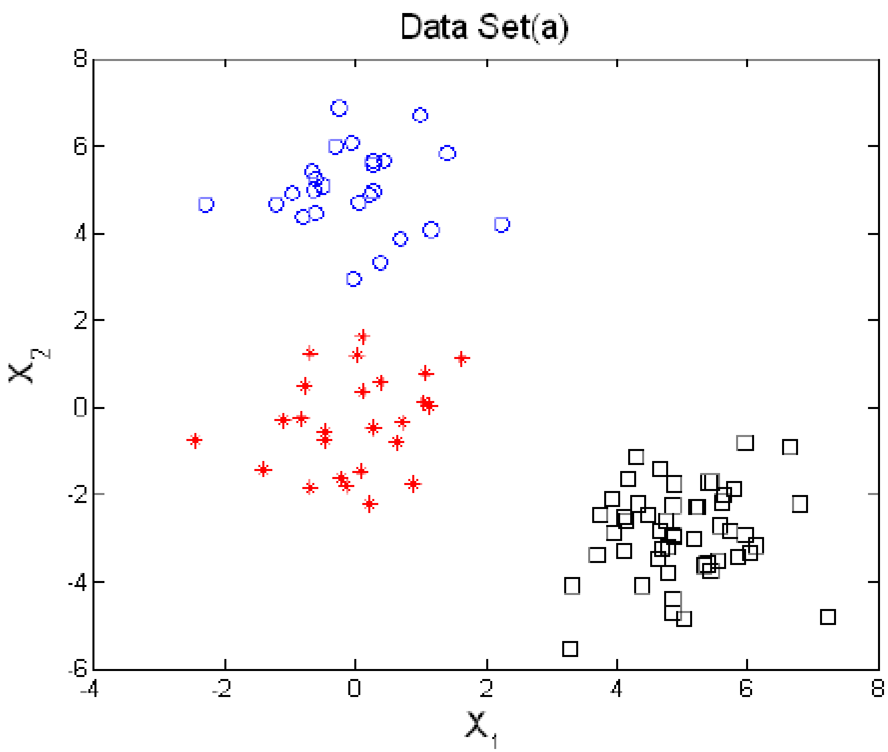


Figure 16. Results for two clusters with different sample sizes. The first cluster contains 100 observations and the second one contains 100α observations.

Multiple-cluster Dataset

a. Three clusters in two dimensions

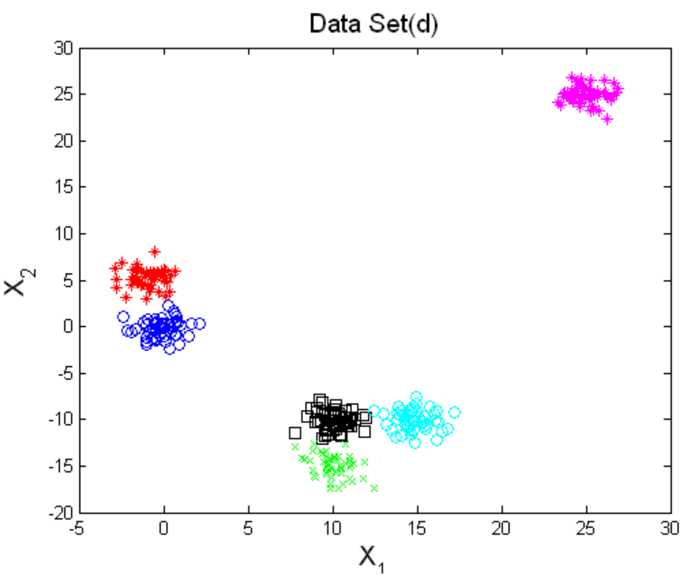


Multiple-cluster Dataset

b. Three clusters in ten dimensions: generated as in (a), but centered at zeros(1,10), 1.6*ones(1,10), (1.6,-1.6, …,-1.6). Each cluster contains 50 observations.

c. Four clusters in ten dimensions with randomly chosen centers.

d. Six clusters in two dimension



Overall Accuracy

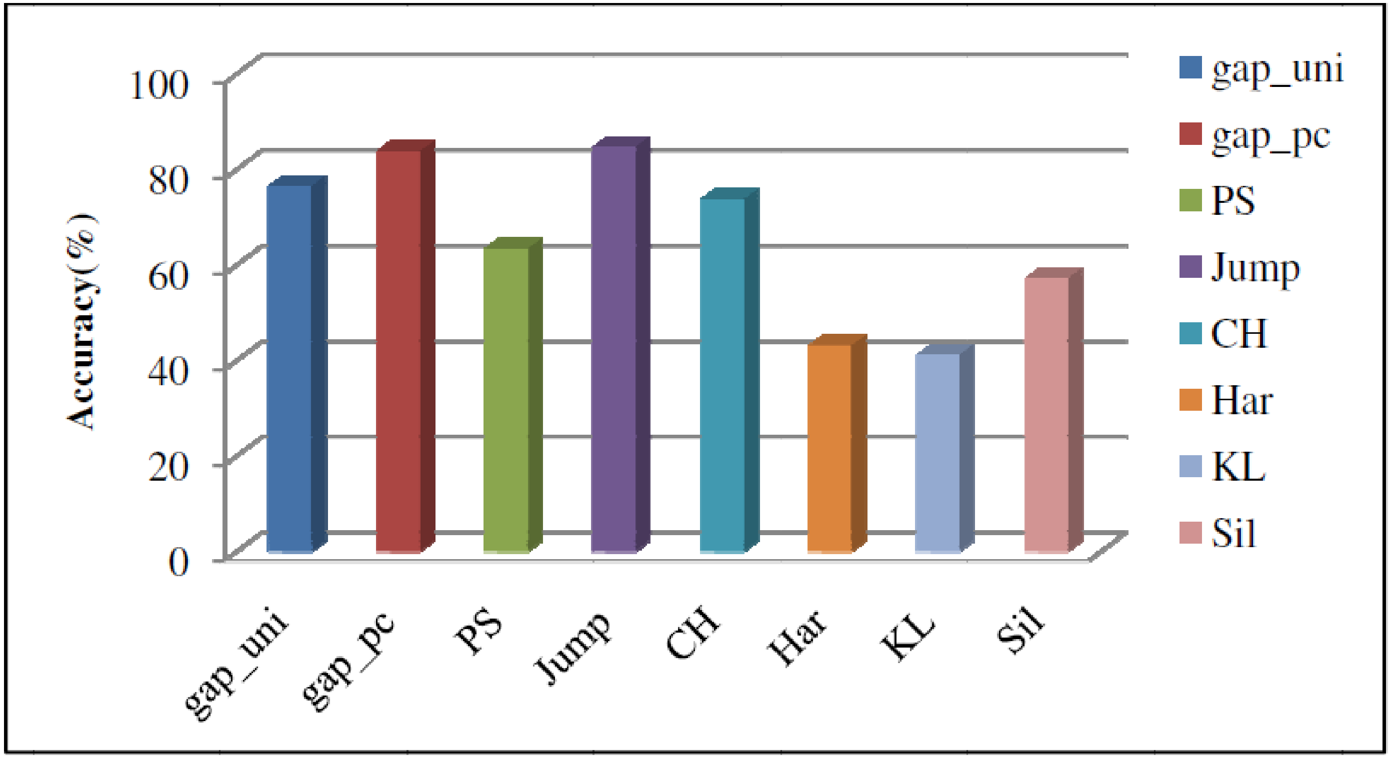
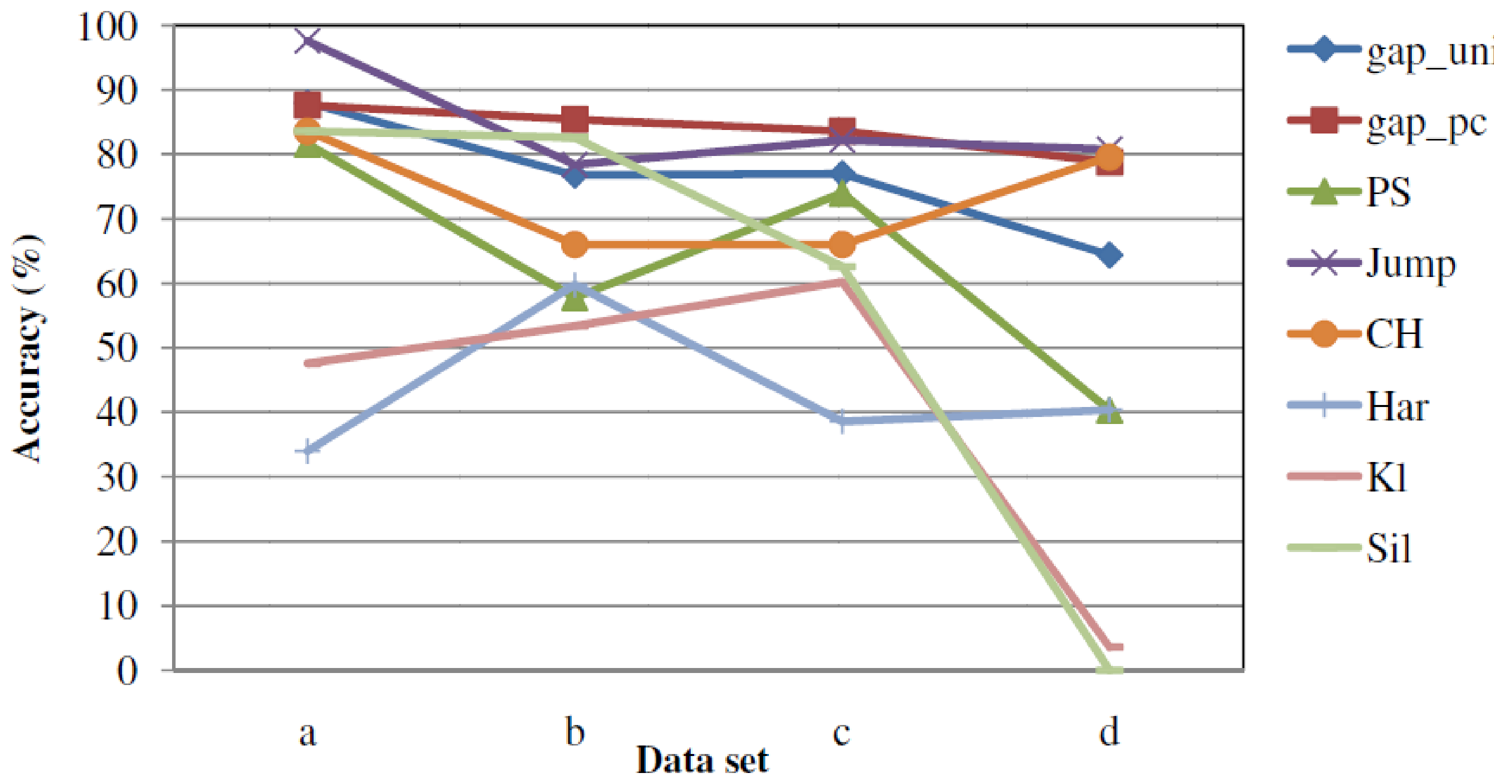
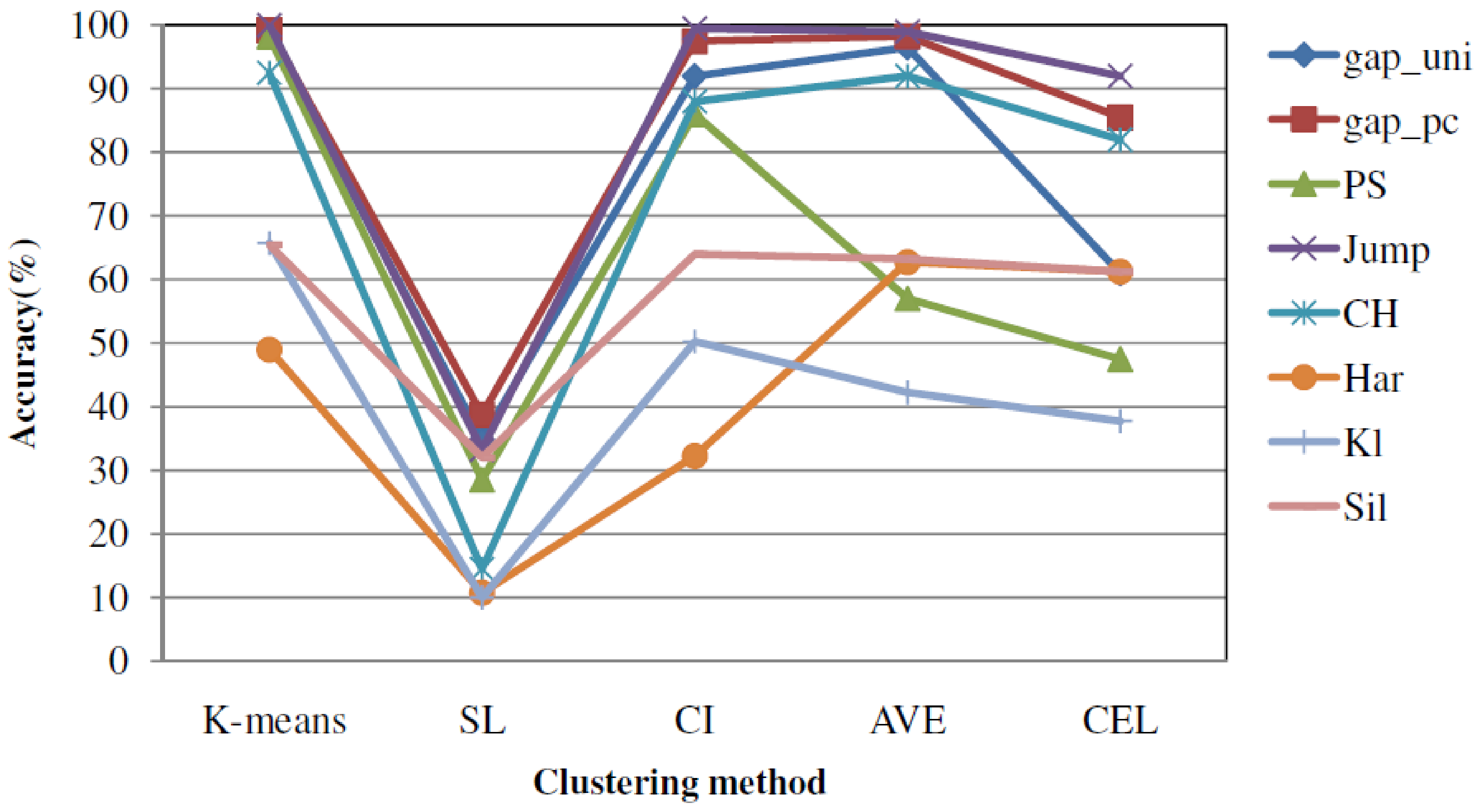


Figure18. Overall accuracy of the eight studied methods applied to multi-cluster data sets.

Accuracy Vs. Dataset Used



Accuracy Vs. Clustering Method



Summary

- Review seven important methods proposed for finding the right number of clusters in a data set.
- The performance of a method may depend on both the using clustering algorithm and the given dataset.
- The performance of the jump method drops drastically as the dimension of data increases.
- The gap statistic method is the best method for both one-cluster.
- The gap/pc method performed better than the gap/uni.
- Non of the studied methods worked well for all data and all clustering algorithms used.
- We should apply different methods and synthesize the results.

Thanks

Questions?

A Study of Clustering Applied to Multiple Target Tracking Algorithm

Pavlina Konstantinova, Milen Nikolov, and Tzvetan Semerdjiev

Data Association in Target Tracking

- Associate the received observations to existing tracks.
- The most important step in target tracking

Challenges

- The targets may be :
  1. closely spaced
  2. not detected in successive scans
  3. move in large groups.
- Measurement noise

Problem Formulation

- Assign M observations to N targets ,

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} \zeta_{ij}$$

S.t.

$$\begin{cases} \sum_{j=1}^{m} \zeta_{ij} = 1 & i = 1, \ldots, n \\ \sum_{i=1}^{n} \zeta_{ij} = 1 & j = 1, \ldots, m \end{cases}$$

$$\zeta_{ij} = \begin{cases} 1 & \textit{if the observation } j \textit{ is assigned to track } i \\ 0 & \textit{otherwhise} \end{cases}$$

$$c_{ij} = \begin{cases} d_{ij}^2 & \textit{if the observation is in the track's gate } (d_{ij}^2 < G) \\ \infty & \textit{otherwhise} \end{cases}$$

Solution

- Try all possible assignments, but it is very computationally expensive.
- Use clustering technique, reduce search space for each target.

1.Clustering
2 For each cluster:
    2.1 Initialization of the assignment matrix
    2.2.Filling up the assignment matrix and solving the assignment problem
    2.3.Checking the validity of the solution and making associations
3.Track filtering

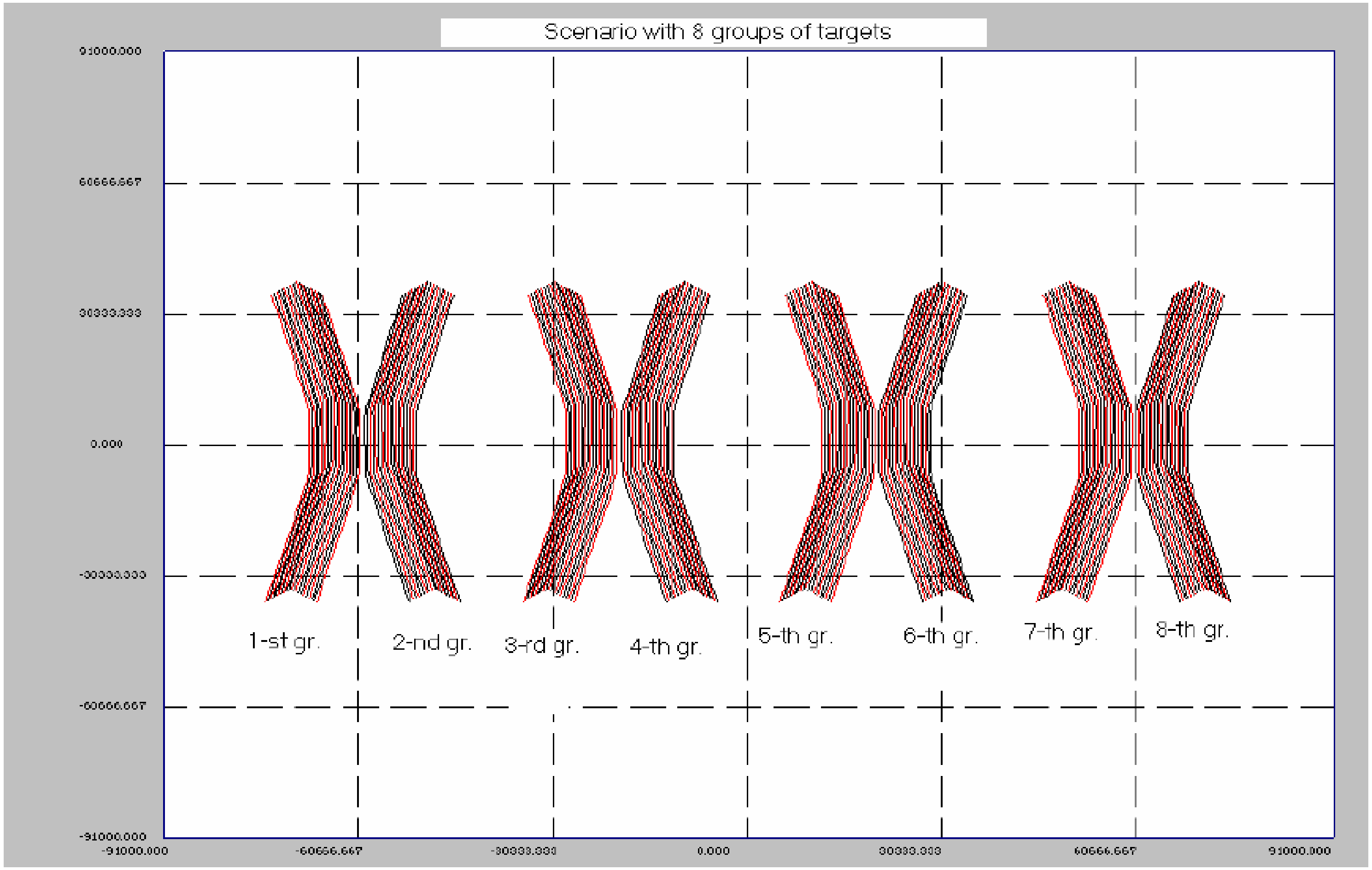
Clustering Procedure

```
For each observation received in the current scan
  NumOfGates=0; //the number of gates in which the observation is fallen
  For each track
     If  the observation is in the track gate
        NumOfGates= NumOfGates+1;
        If the track is not included in cluster form new cluster for the track
           NumOfClusters= NumOfClusters + 1;
           Write the observation in track list.
           If NumOfGates > 1 i.e. the observation falls in the gate of other track
              If OldCluster ≠ <track's cluster>  then MERGE clusters
              If track's cluster is not the last - compress cluster's array
           else
              OldCluster=Track's cluster
  End for each track
End for each observation
Form clusters for tracks without observations (to be filtered "by memory")
```

- Merge two clusters, If an observation is in the cluster of two tracks
- The maximal number of clusters < the number of tracked tracks.

Experimental Setup

8 groups of targets, each group consists of 21 moving targets

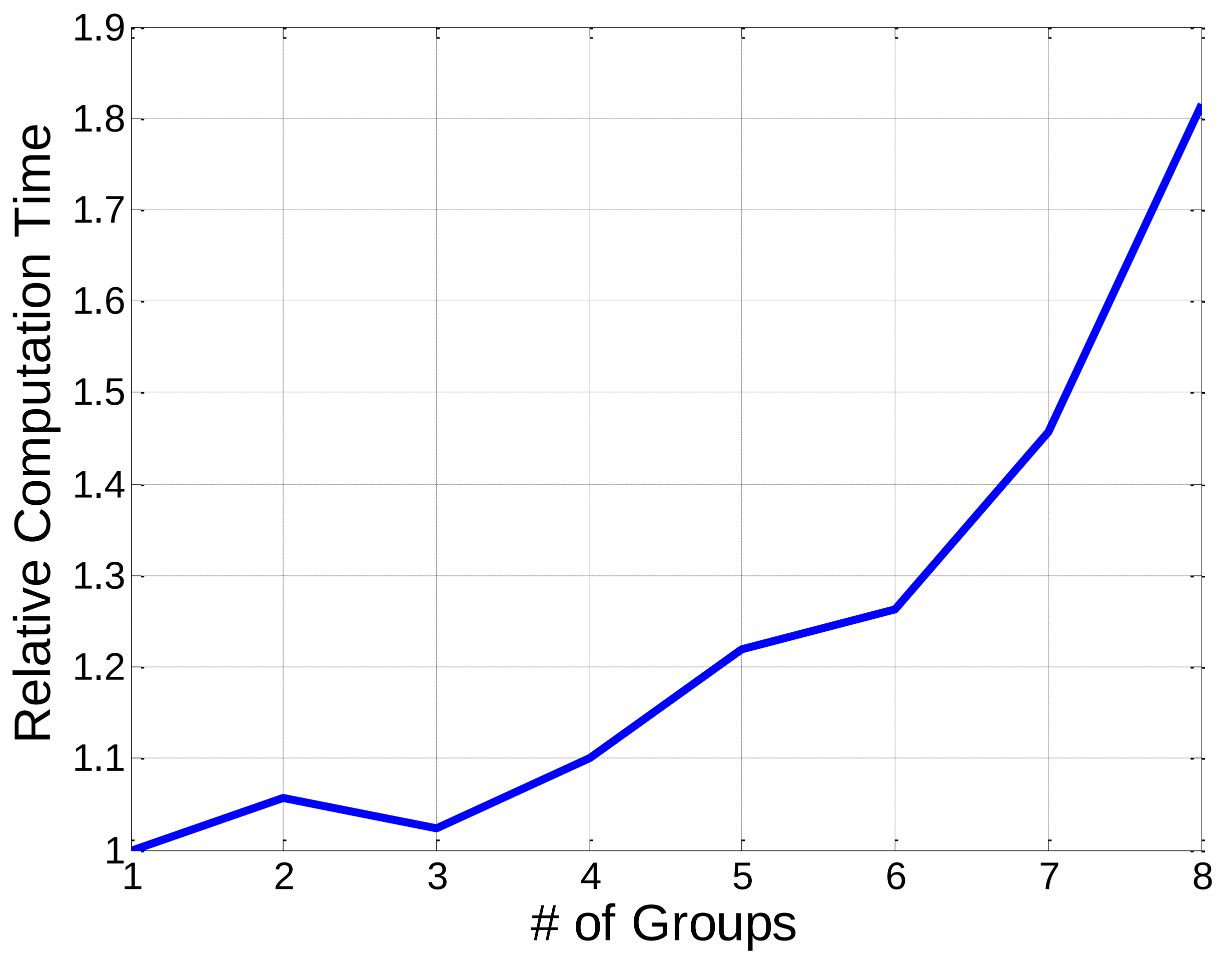


Results

No of experi-ments	Number of groups	Number of targets	Execution time [sec]							
			Pentium S, 133 MHz		Pentium 2, 266 MHz		Celeron, 300 MHz		AMD Athlon, 1050 MHz	
			without clusters	with clusters	without clusters	with clusters	without clusters	with clusters	without clusters	with clusters
1	8	168	122.02	67.22	47.89	22.68	42.53	27.31	10.11	5.60
2	7	147	79.15	54.32	28.89	18.80	28.71	21.68	6.43	4.45
3	6	126	53.16	42.10	17.87	13.95	20.62	16.89	4.34	3.46
4	5	105	38.94	31.92	12.42	10.43	14.79	12.68	3.14	2.58
5	4	84	26.36	23.95	8.24	7.36	10.12	8.98	2.14	1.81
6	3	63	16.76	16.37	5.90	4.78	6.29	5.80	1.32	1.15
7	2	42	9.23	8.73	2.75	2.70	3.51	3.42	0.66	0.66
8	1	21	3.80	3.80	1.10	1.10	1.50	1.50	0.22	0.22

Results

$$\text{Relative Computation time} = \frac{\text{Computation time without clustering}}{\text{Computation time with clustering}}$$



Summary

- Data association step is a crucial step in multiple target tracking.
- Can be computationally expensive especially for large scenario with many targets.
- Using clustering in data association step saves significant computational time.

Thanks

Questions?