

Clustering

Matteo Pardo

National Research Council (CNR), Italy &
Max Planck Institute for Molecular Genetics,
Berlin, Germany

Outline

- Intro: learning problems
- Distance measures
- Basic clustering algorithms:
 - Partition methods
 - Hierarchical methods
- Filtering features before clustering
- Cluster validation
- Cluster clustering methods





Contents lists available at [ScienceDirect](#)

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec



Data clustering: 50 years beyond K-means [☆]

Anil K. Jain ^{*}

Department of Computer Science and Engineering, Michigan State University, East Lansing, Michigan 48824, USA
Department of Brain and Cognitive Engineering, Korea University, Anam-dong, Seoul, 136-713, Korea

According to CiteSeer, his book, *Algorithms for Clustering Data* is ranked # 91 in the [Most Cited Articles in Computer Science](#) (over all times) and his paper "Data Clustering: A Review" (ACM Computing Surveys, 1999) is consistently ranked in the [Top 10 Most Popular Magazine and Computing Survey Articles Downloaded](#)



Learning problems

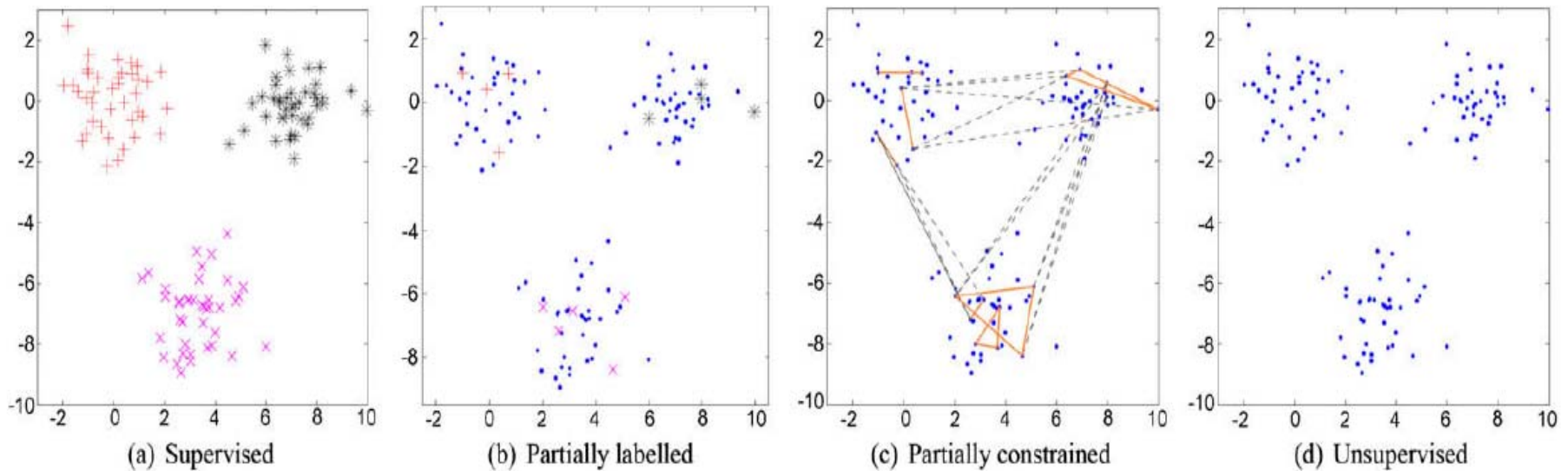


Fig. 1. Learning problems: dots correspond to points without any labels. Points with labels are denoted by plus signs, asterisks, and crosses. In (c), the must-link and cannot-link constraints are denoted by solid and dashed lines, respectively (figure taken from Lange et al. (2005)).

Cancer classification	Class discovery	Class prediction
Machine learning	Unsupervised learning	Supervised learning
Statistics	Cluster analysis	Discriminant analysis

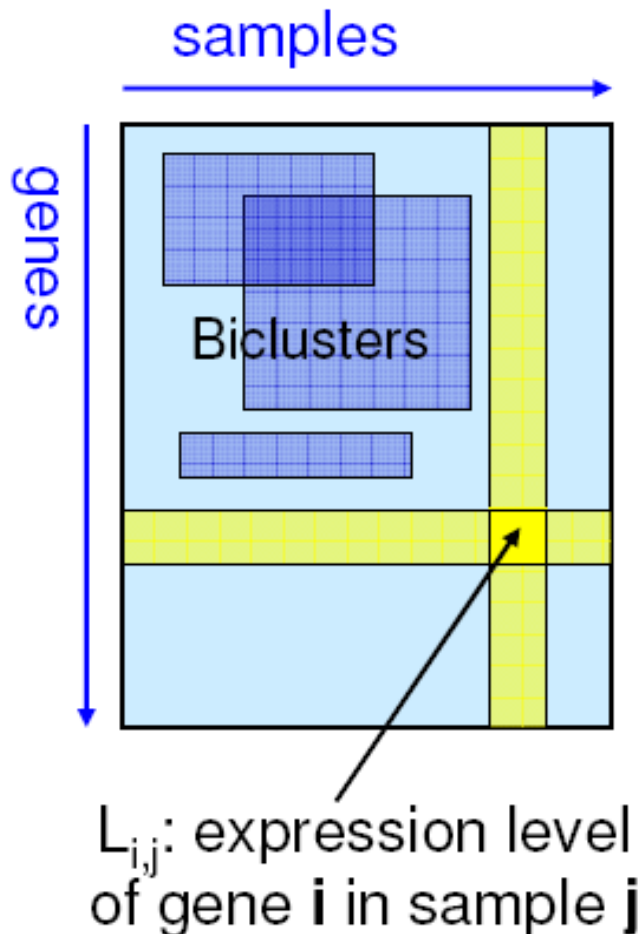
Biological setting: Tumor Characterization Using Gene Expression

Three main types of statistical problems associated with tumor classification:

- Identification of **new/unknown** tumor classes using gene expression profiles (**unsupervised learning – clustering**)
- Classification of malignancies into **known** classes (**supervised learning – discrimination**)
- Identification of “marker” genes that characterize the different tumor classes (**feature or variable selection**).



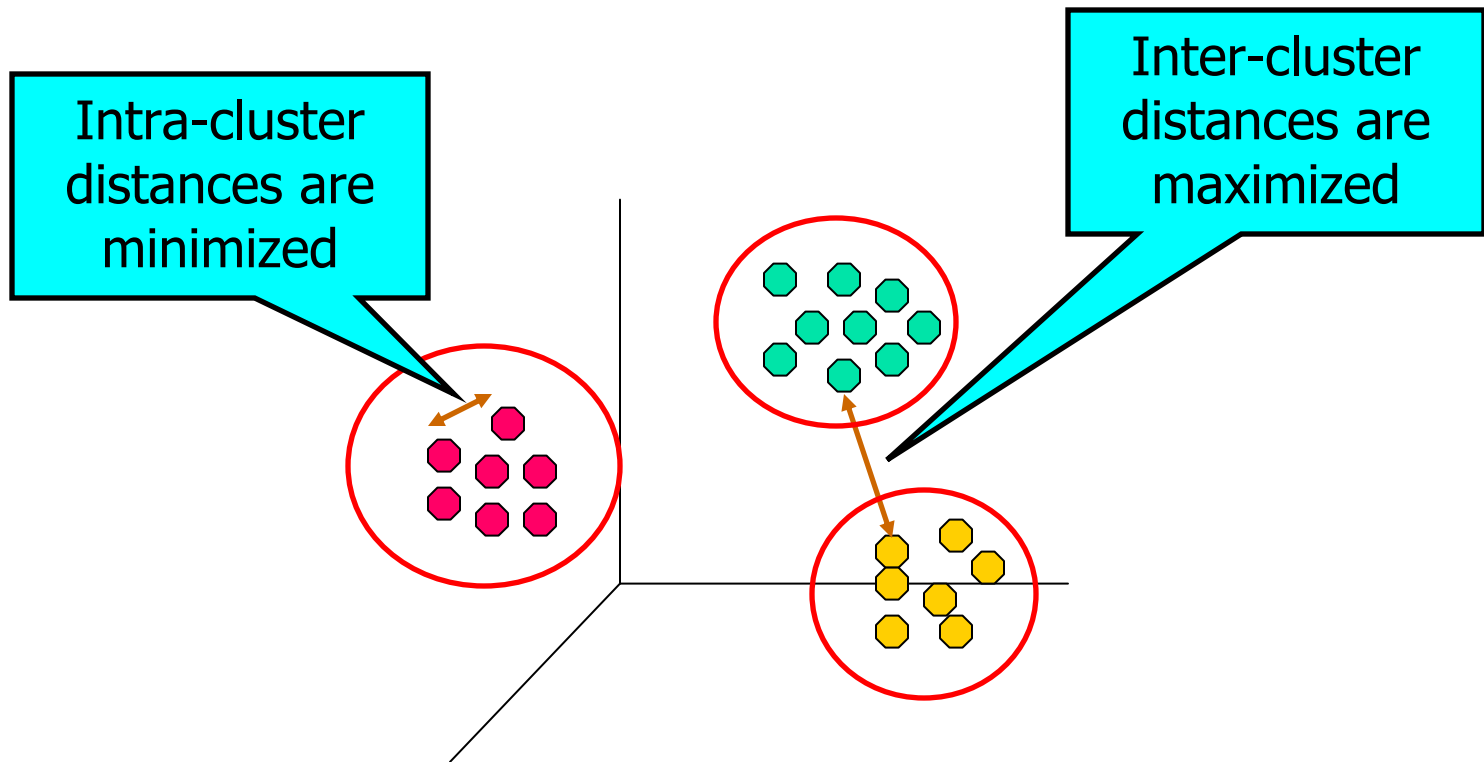
What to cluster



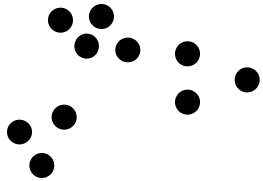
- **Samples:** To discover novel subtypes of the existing groups or entirely new partitions. Their utility needs to be confirmed with other types of data, e.g. clinical information.
- **Genes:** To discover groups of co-regulated genes/ESTs. Infer function where it is unknown using members of the groups with known function.

What is Cluster Analysis?

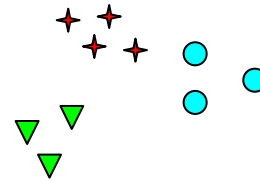
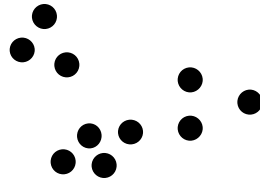
- Finding groups of objects such that the objects in a group will be **similar** (or related) to one another and **different** from (or unrelated to) the objects in other groups



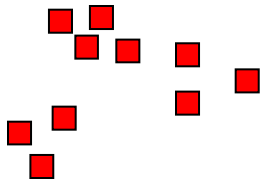
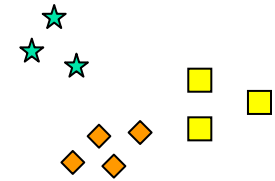
Notion of a Cluster can be Ambiguous



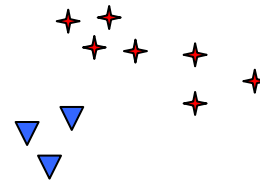
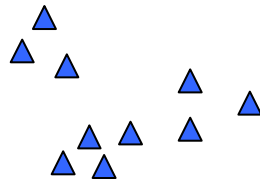
How many clusters?



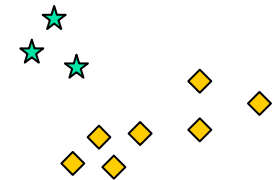
Six Clusters



Two Clusters



Four Clusters



Types/definitions of Clusters

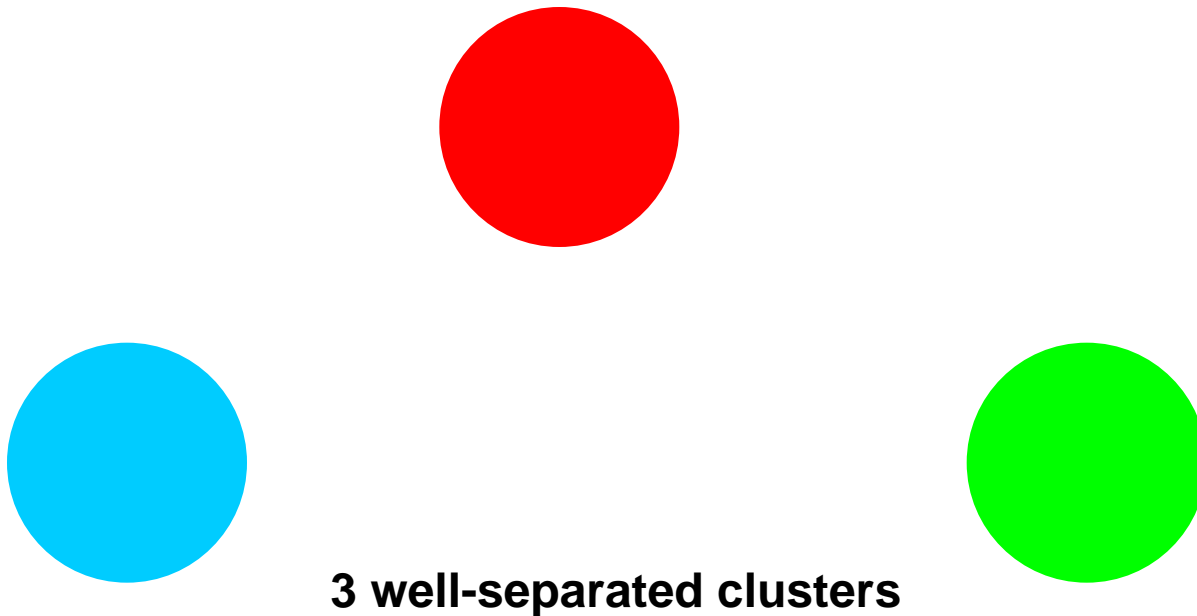
- Well-separated clusters
- Center-based clusters
- Contiguous clusters
- Density-based clusters
- Property or Conceptual



Types of Clusters: Well-Separated

- Well-Separated Clusters:

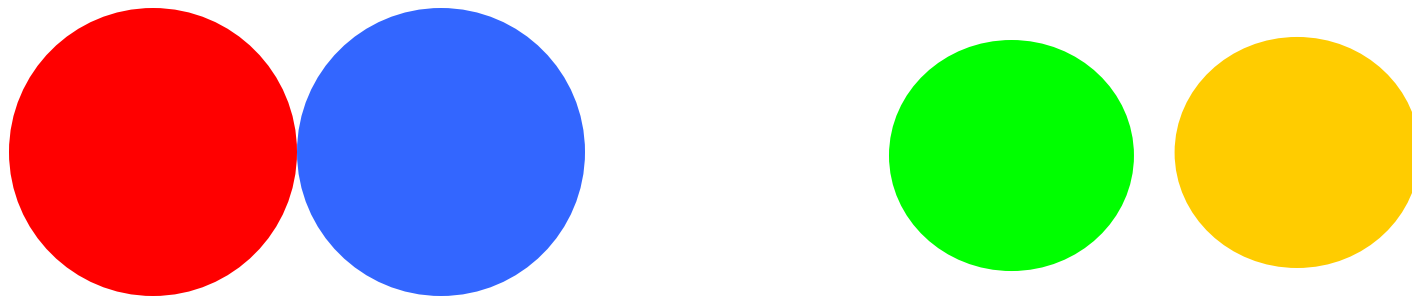
- A cluster is a set of points such that any point in a cluster is **closer to every other point** in the cluster than to **any point not in the cluster**.



Types of Clusters: Center-Based

o Center-based

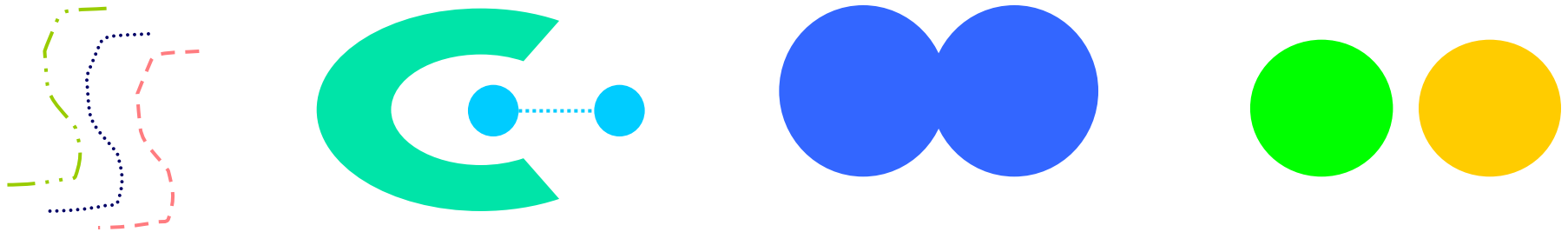
- A cluster is a set of objects such that an object in a cluster is closer (more similar) to the “center” of a cluster, than to the center of any other cluster
- The center of a cluster is often a **centroid**, the average of all the points in the cluster, or a **medoid**, the most “representative” point of a cluster



4 center-based clusters

Types of Clusters: Contiguity-Based

- Contiguous Cluster (Nearest neighbor or Transitive)
 - A cluster is a set of points such that a point in a cluster is closer to **one** or more other points in the cluster than to any point not in the cluster.

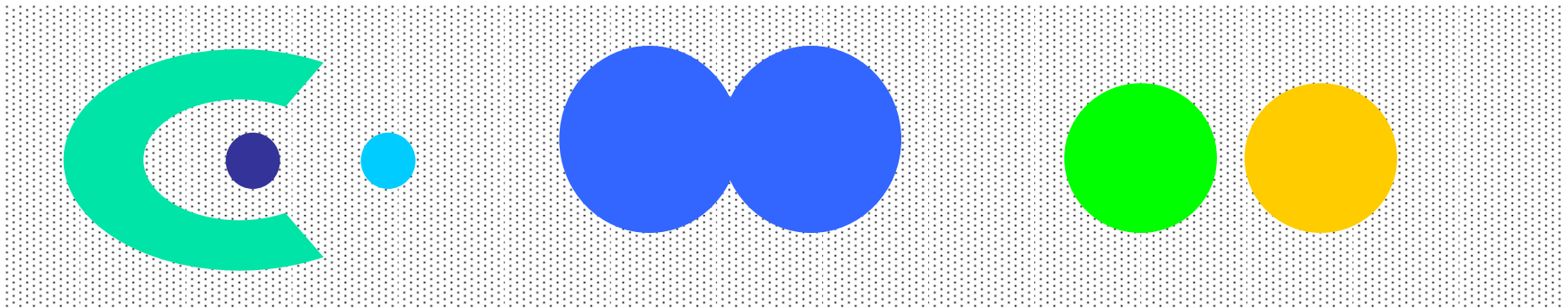


8 contiguous clusters

Types of Clusters: Density-Based

o Density-based

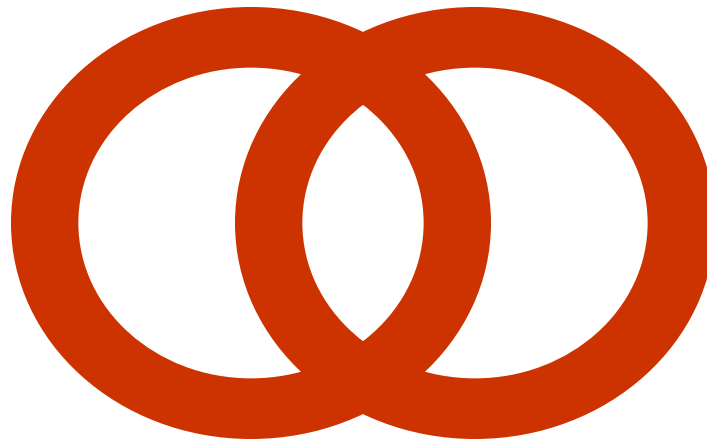
- A cluster is a dense region of points, which is separated by low-density regions, from other regions of high density.
- Used when the clusters are irregular or intertwined, and when noise and outliers are present.



6 density-based clusters

Types of Clusters: Conceptual Clusters

- Shared Property or Conceptual Clusters
 - Finds clusters that share some common property or represent a particular concept.



2 Overlapping Circles

Clustering: two ingredients

- Distance (similarity) measure: when are two objects close to each other?
- Clustering algorithm: procedure to minimize distances of objects within groups and/or maximize distances between groups

Preprocessing (before clustering), common to supervised analysis: :

- Features (variables) standardization/ normalization iff all variables **a priori** have the same importance
- Object standardization/ normalization iff all objects **a priori** have the same importance
- Feature selection



Cluster Analysis – Distance Measures (metrics)

o Two main classes of metric:

- **Correlation coefficients (similarity)**
 - Compares shape of expression curves
 - Types of correlation:
 - Centered.
 - Un-centered.
 - Rank-correlation
- **Distance metrics (dissimilarity)**
 - City Block (Manhattan) distance
 - Euclidean distance



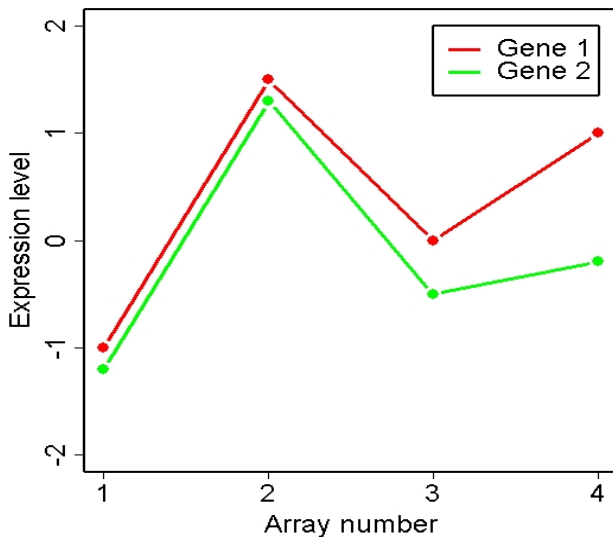
Correlation (a measure between -1 and 1)

o Pearson Correlation Coefficient (centered correlation)

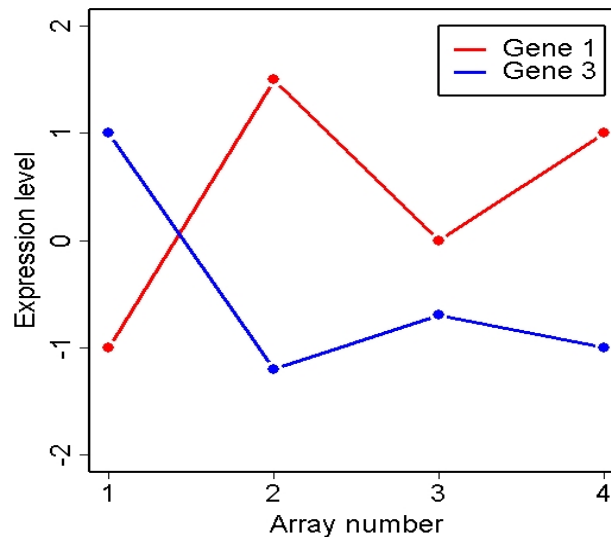
S_x = Standard deviation of x

S_y = Standard deviation of y

$$\frac{1}{n-1} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{S_x} \right) \left(\frac{y_i - \bar{y}}{S_y} \right)$$



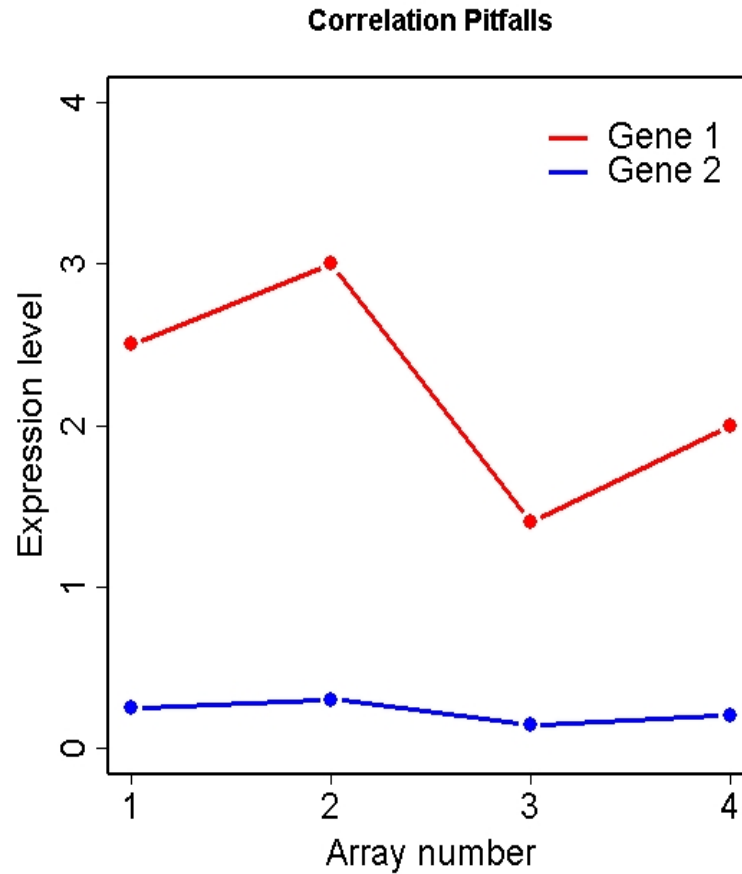
Positive correlation



Negative correlation

You can use **absolute correlation** to capture both positive and negative correlation

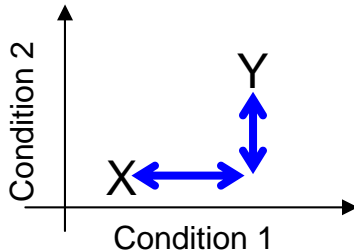
Potential pitfalls



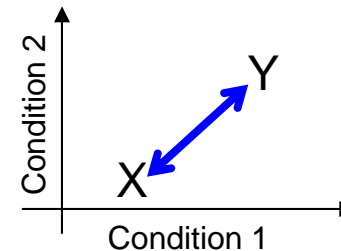
Correlation = 1

Distance metrics

$$d(X, Y) = \sum_i |x_i - y_i|$$



$$d(X, Y) = \sqrt{\sum_i (x_i - y_i)^2}$$



where gene $X = (x_1, \dots, x_n)$ and gene $Y = (y_1, \dots, y_n)$

○ City Block (Manhattan) distance:

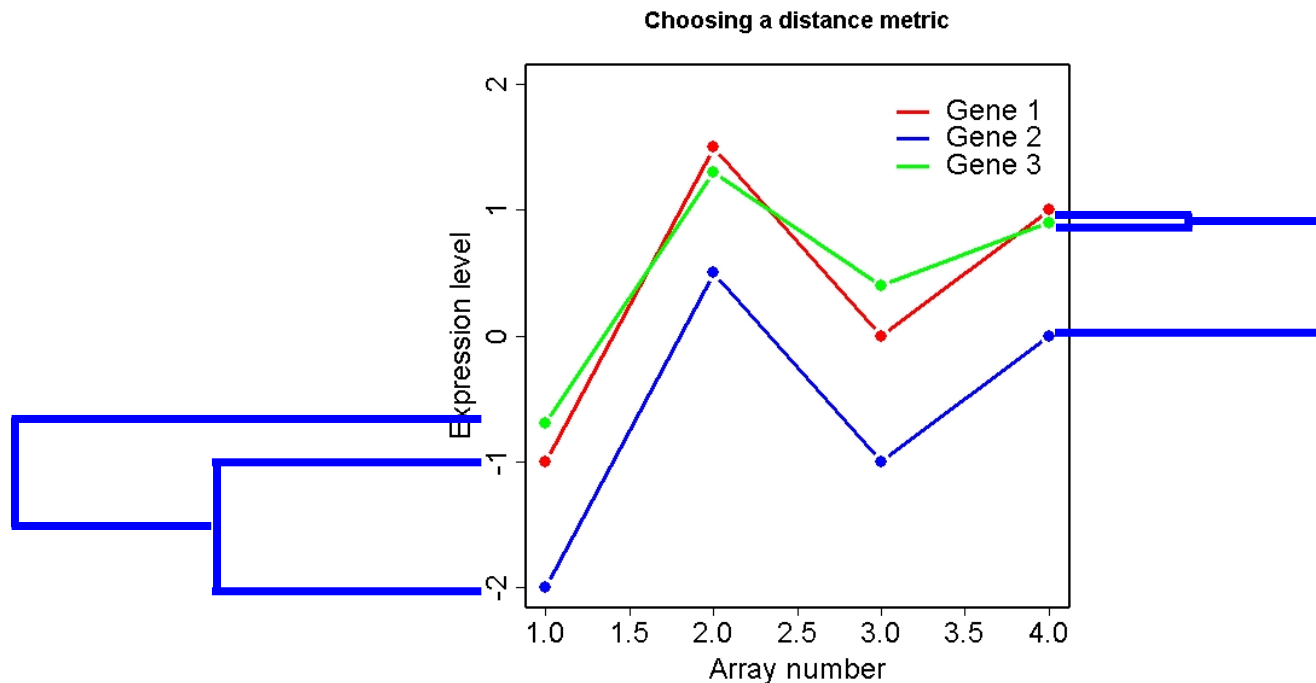
- Less sensitive to outliers
- Diamond shaped clusters

○ Euclidean distance:

- Most commonly used distance
- Sphere shaped cluster

Euclidean vs Correlation

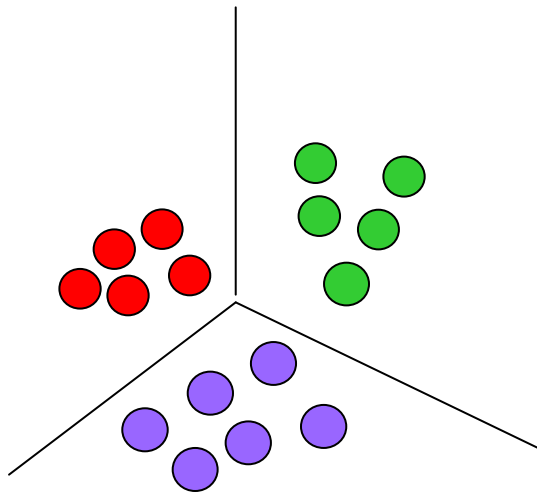
- Euclidean distance
- Correlation



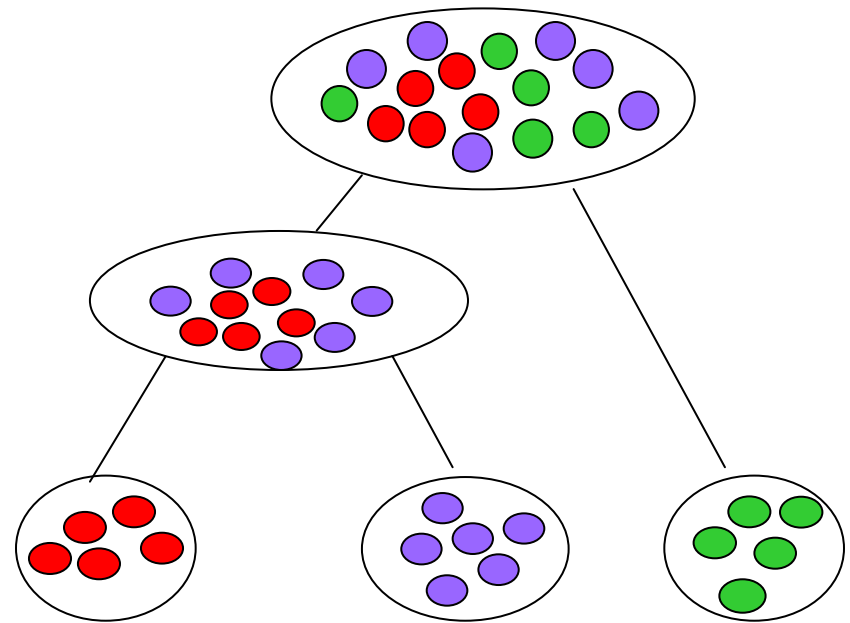
Clustering algorithms

- Clustering algorithm comes in 2 basic flavors:
 - **Partitional Clustering:** a division data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset
 - **Hierarchical clustering:** a set of nested clusters organized as a hierarchical tree

Partitioning



Hierarchical



Clustering Algorithms

- o K-means and its variants
- o Hierarchical clustering
- o Density-based clustering



K-means Clustering

- Partitional clustering approach
- Number of clusters, K , must be specified

-
- 1: Select K points as the initial centroids.
 - 2: **repeat**
 - 3: Form K clusters by assigning all points to the closest centroid.
 - 4: Recompute the centroid of each cluster.
 - 5: **until** The centroids don't change
-



K-means Clustering – Details

- Initial centroids are often chosen randomly.
 - Clusters produced vary from one run to another.
- The centroid is (typically) the mean of the points in the cluster.
- 'Closeness' is measured by Euclidean distance, cosine similarity, correlation, etc.
- Most of the convergence happens in the first few iterations.
 - Often the stopping condition is changed to 'Until relatively few points change clusters'
- Complexity is $O(n * K * I * d)$
 - n = number of points, K = number of clusters,
 I = number of iterations, d = number of attributes



Evaluating partitions by Sum of Squared Error (SSE)

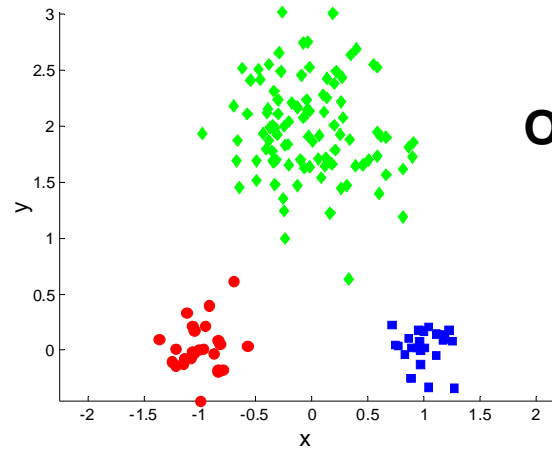
$$SSE = \sum_{i=1}^K \sum_{x \in C_i} dist^2(m_i, x)$$

x is a data point in cluster C_i and m_i is the representative point for cluster C_i

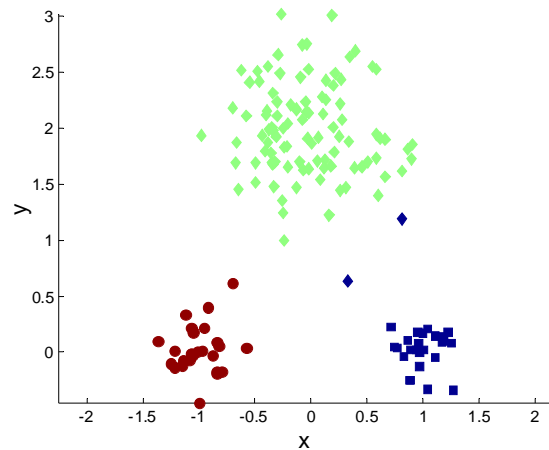
- Given two partitions, we can choose the one with the smallest error
- One easy way to reduce SSE is to increase K , the number of clusters
 - For deciding cluster number: look at the knee in SSE
 - Still: A good clustering with smaller K can have a lower SSE than a poor clustering with higher K



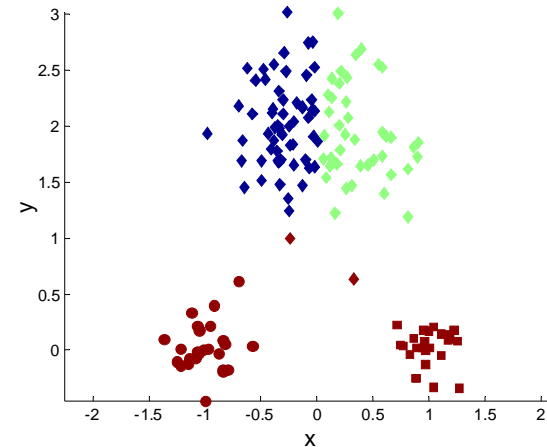
Two different K-means Clusterings



Original Points

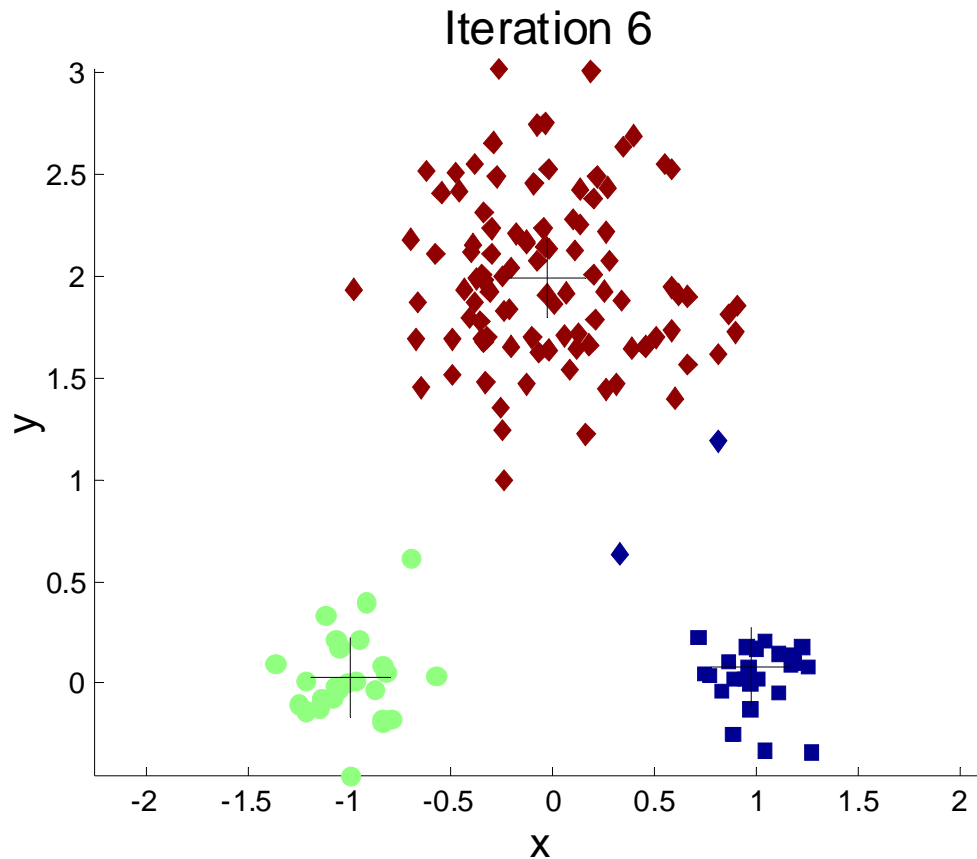


Optimal Clustering

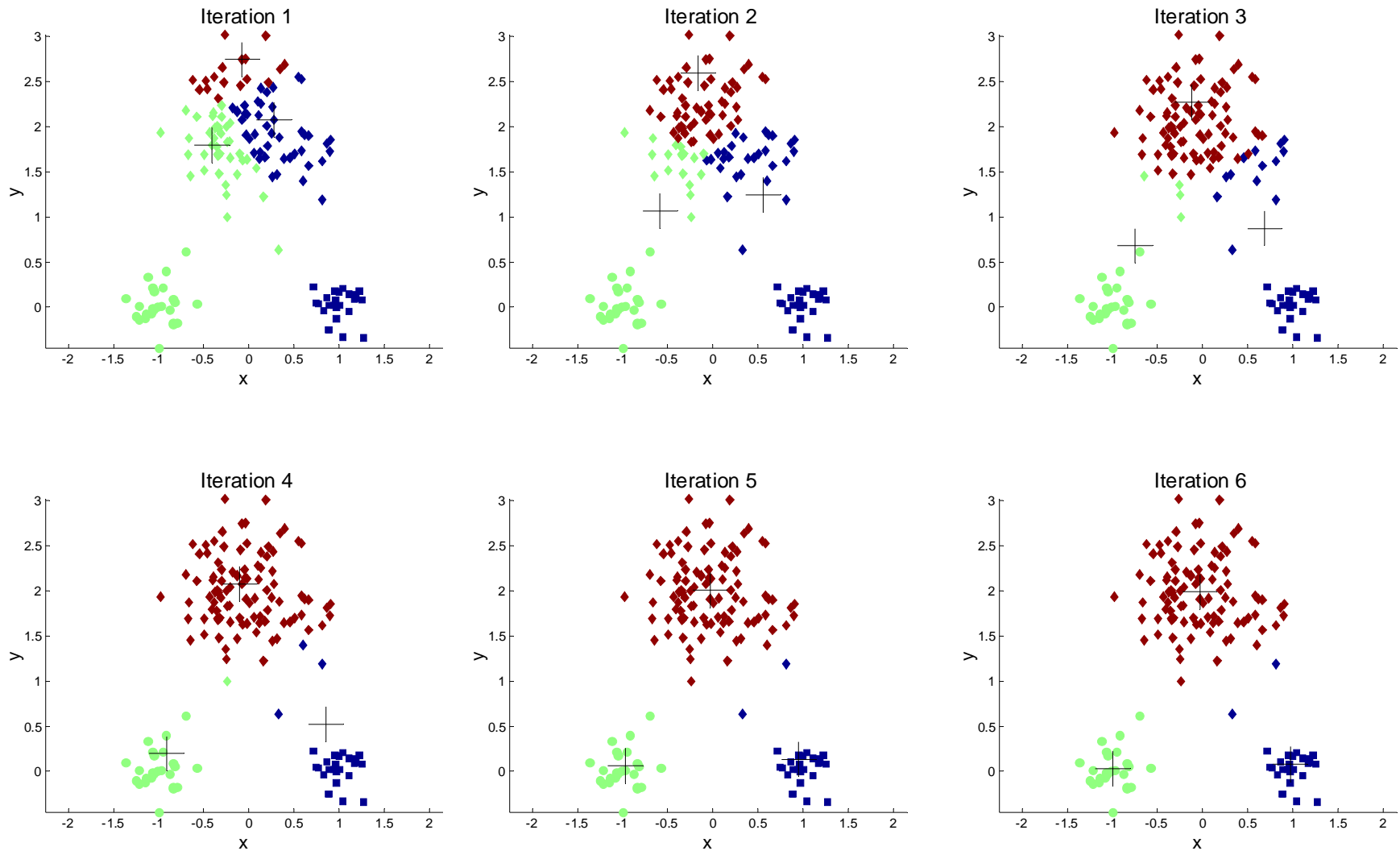


Sub-optimal Clustering

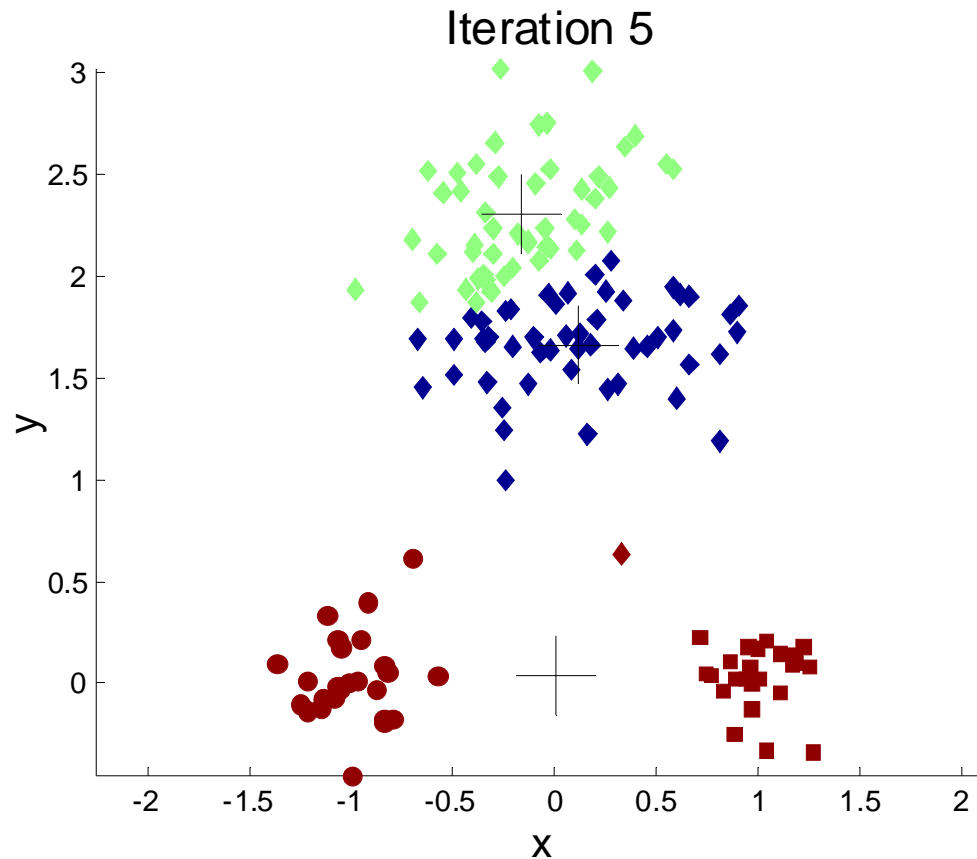
Importance of Choosing Initial Centroids



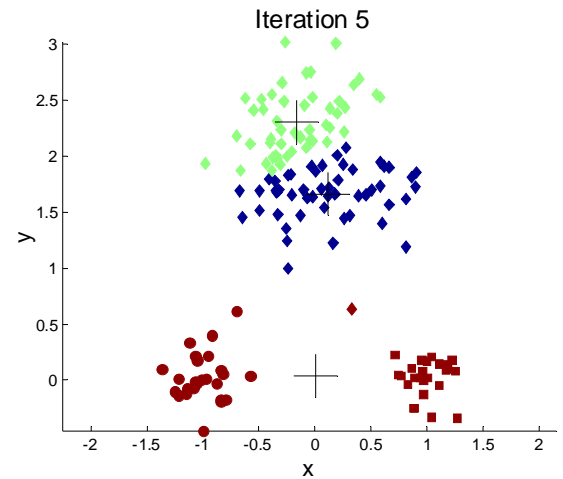
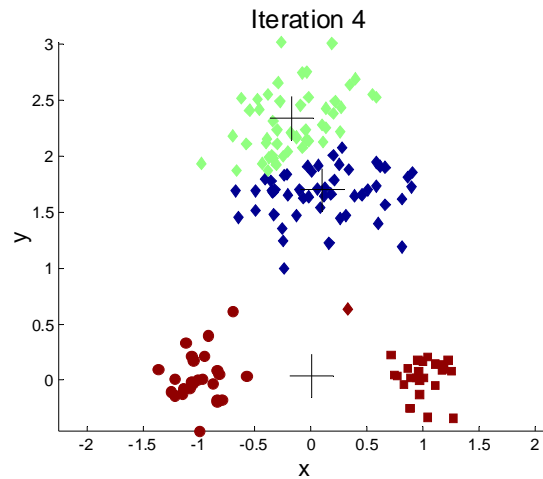
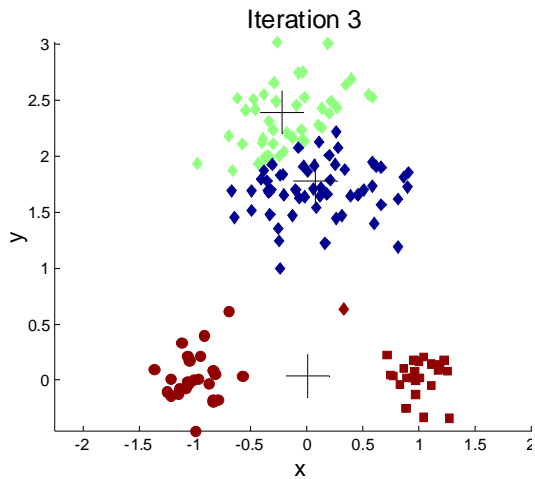
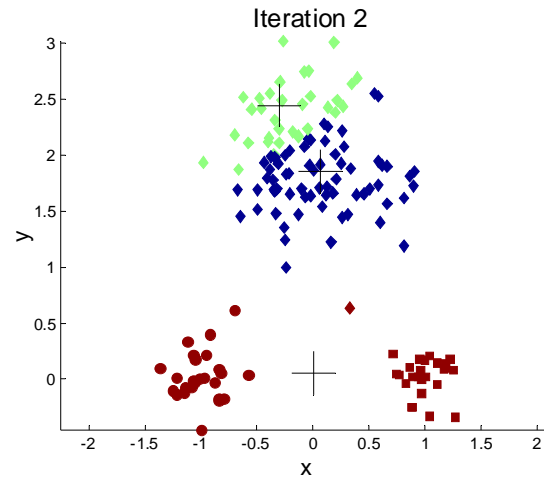
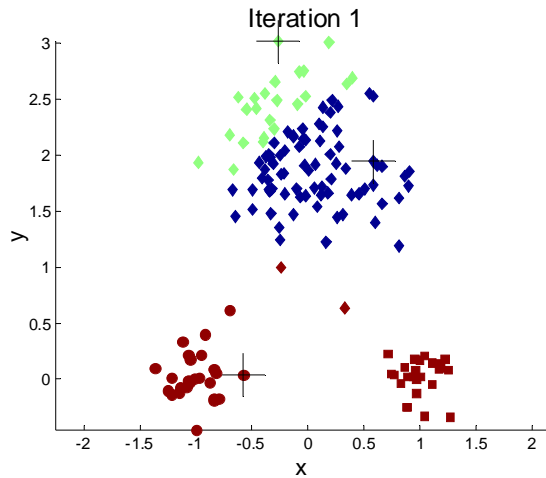
Importance of Choosing Initial Centroids



Importance of Choosing Initial Centroids ...



Importance of Choosing Initial Centroids ...



Solutions to Initial Centroids Problem

- Multiple runs: helps, but long time
- Sample and use hierarchical clustering to determine initial centroids
- Select more than k initial centroids and then select among these initial centroids
 - Select most widely separated
- Postprocessing
- Bisecting K-means
 - Not as susceptible to initialization issues

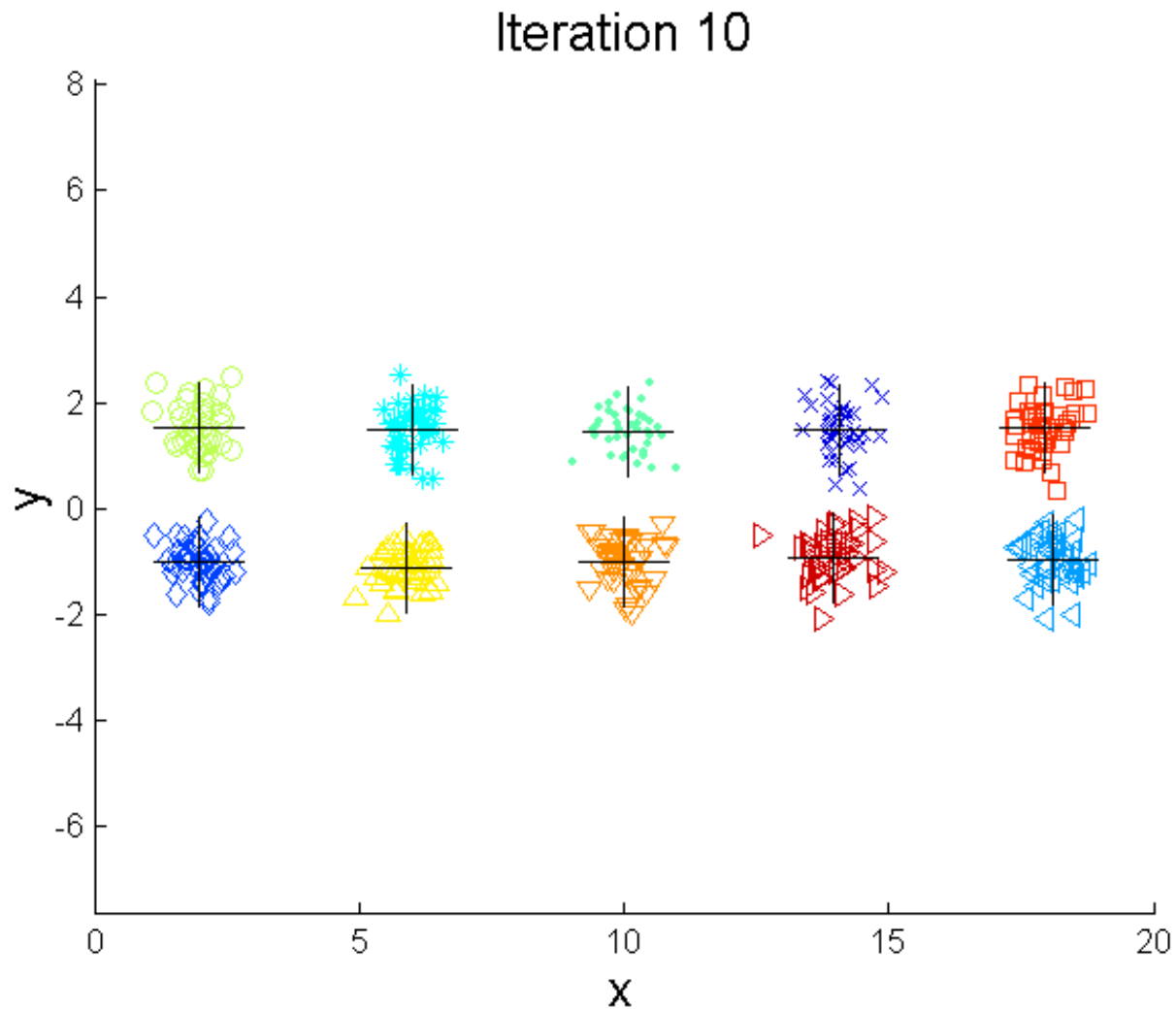


Bisecting K-means

- 1: Initialize the list of clusters to contain the cluster containing all points.
 - 2: **repeat**
 - 3: Select a cluster from the list of clusters
 - 4: **for** $i = 1$ to *number_of_iterations* **do**
 - 5: Bisect the selected cluster using basic K-means
 - 6: **end for**
 - 7: Add the two clusters from the bisection with the lowest SSE to the list of clusters.
 - 8: **until** Until the list of clusters contains K clusters
-



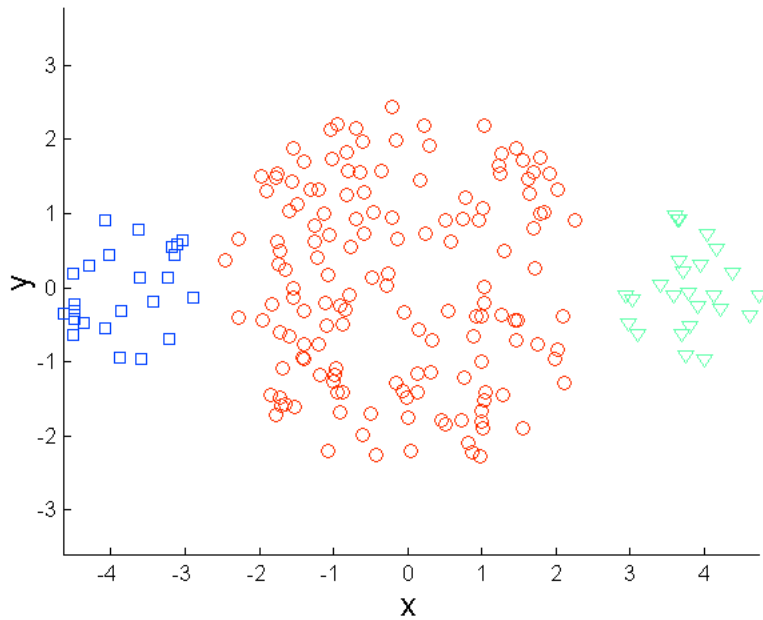
Bisecting K-means Example



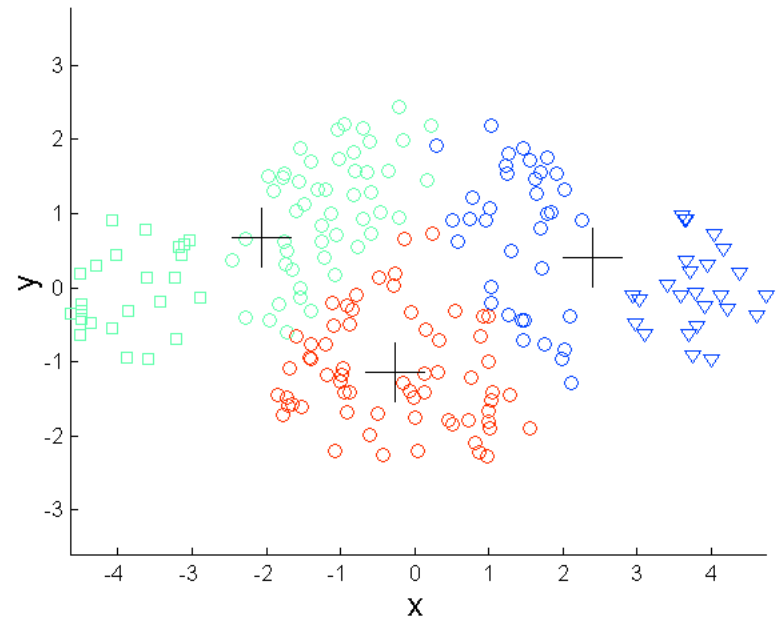
Limitations of K-means

- K-means has problems when clusters are of differing
 - Sizes
 - Densities
 - Non-globular shapes
- K-means has problems when the data contains outliers.

Limitations of K-means: Differing Sizes

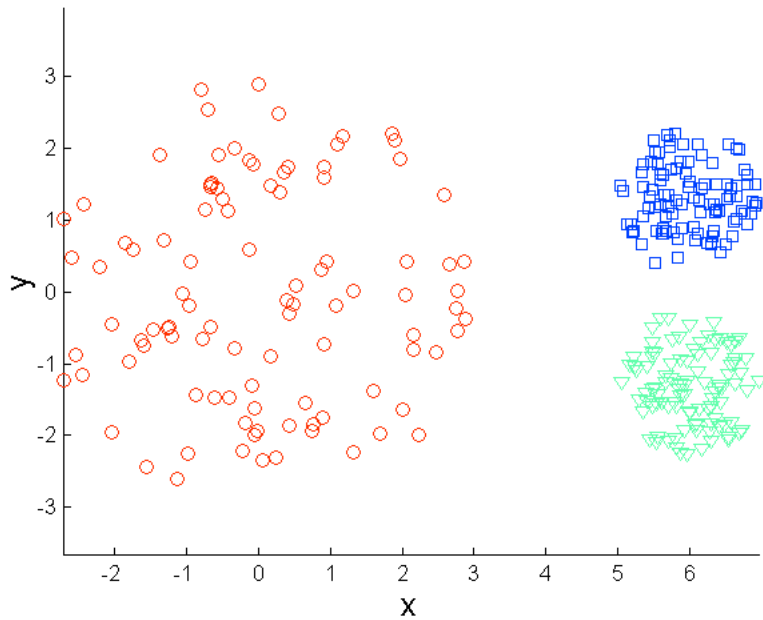


Original Points

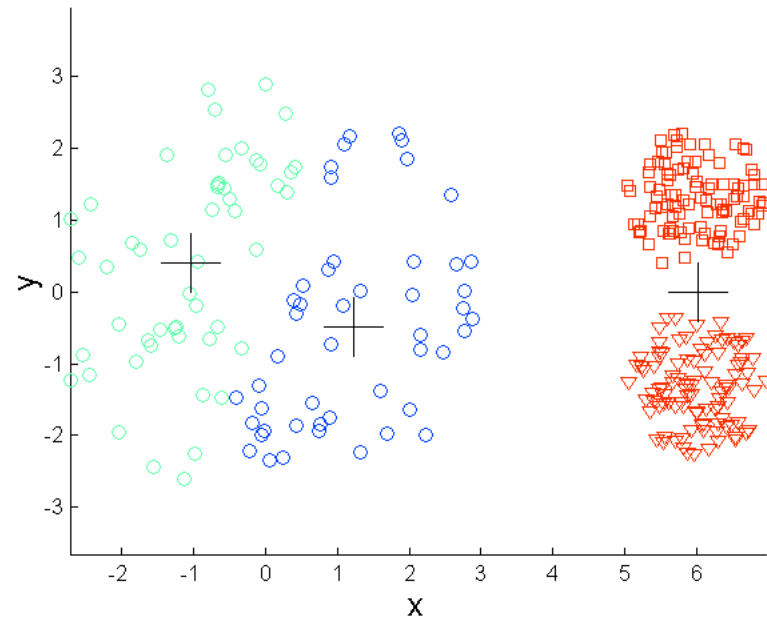


K-means (3 Clusters)

Limitations of K-means: Differing Density

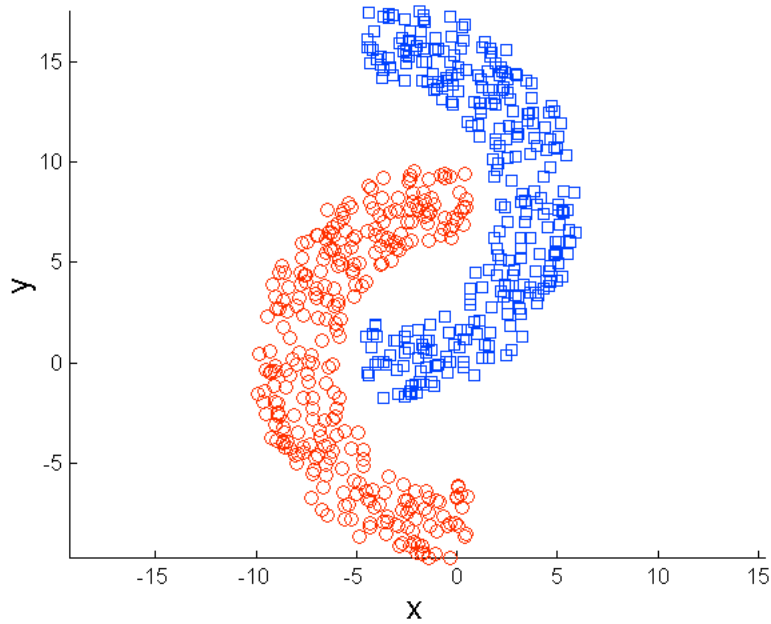


Original Points

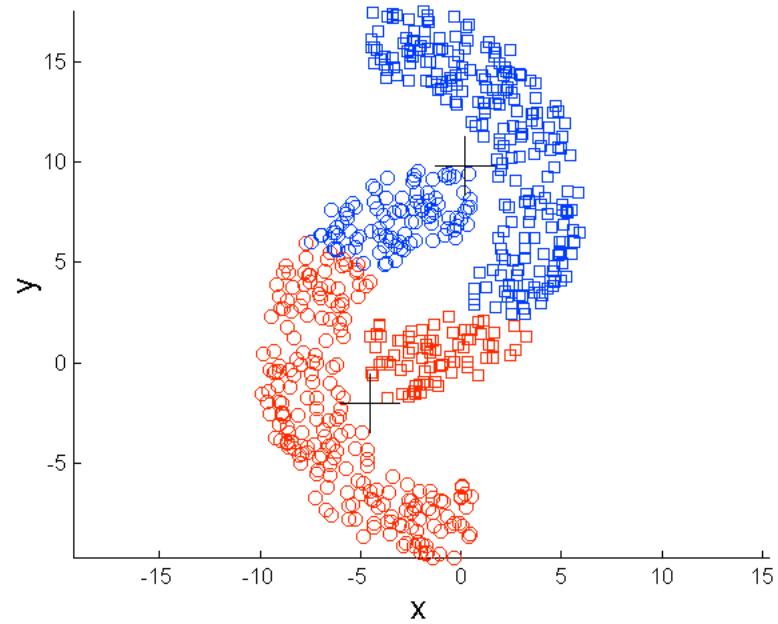


K-means (3 Clusters)

Limitations of K-means: Non-globular Shapes

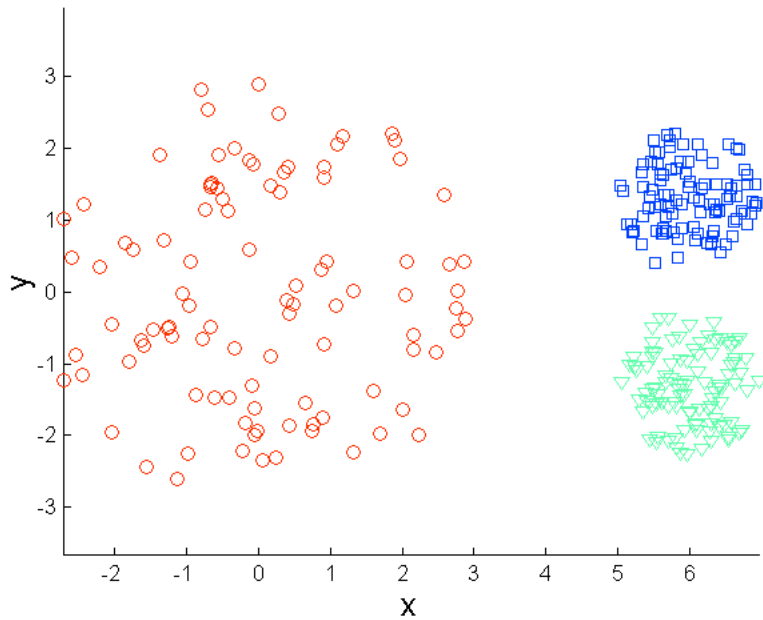


Original Points

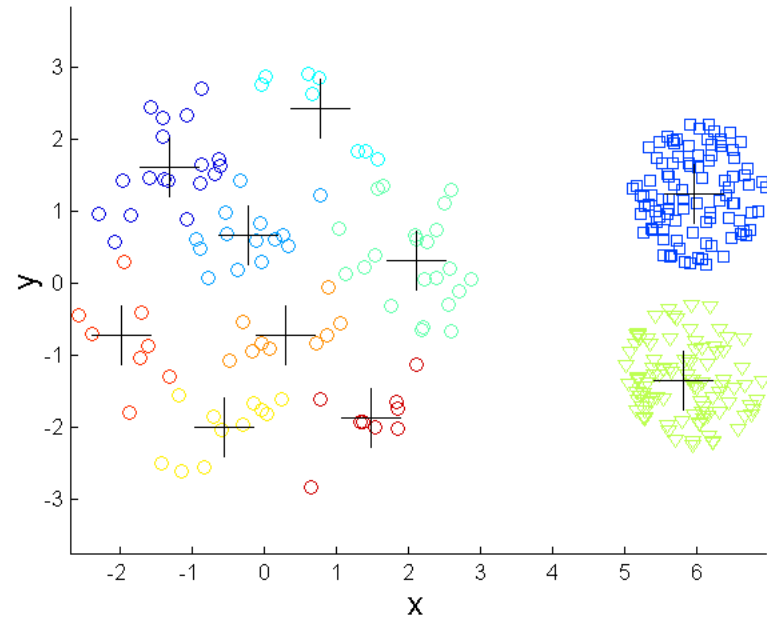


K-means (2 Clusters)

Overcoming K-means Limitations

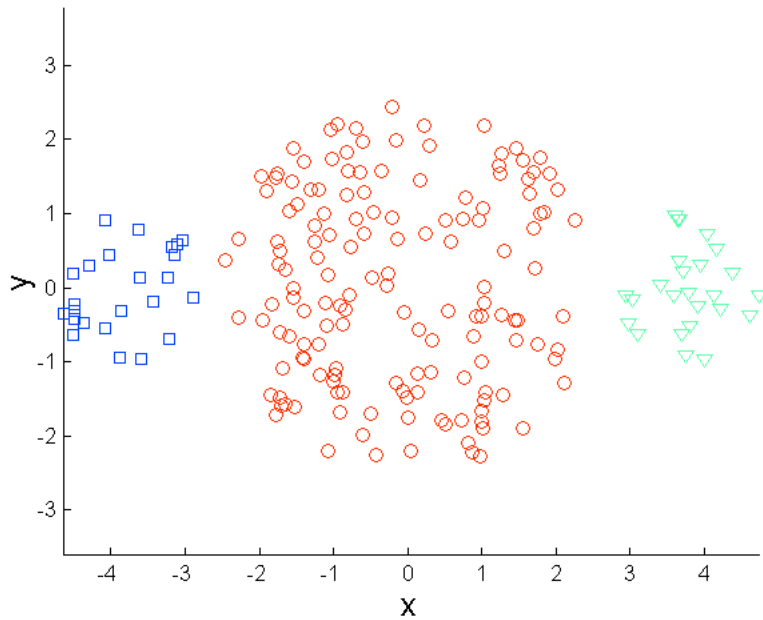


Original Points

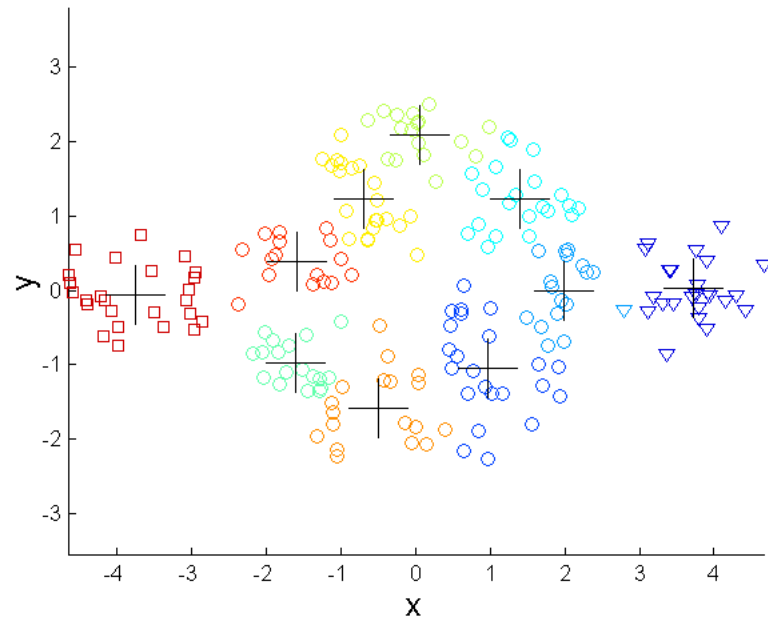


K-means Clusters

Overcoming K-means Limitations?

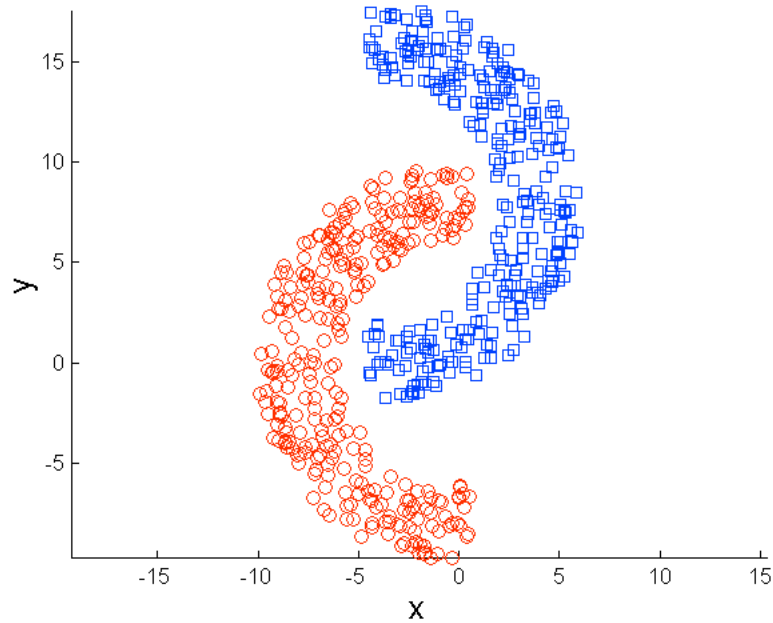


Original Points

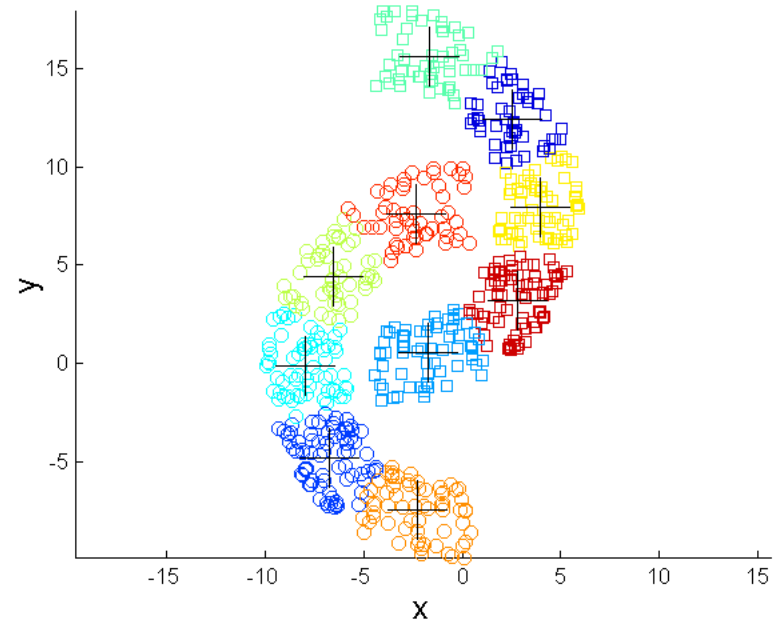


K-means Clusters

Overcoming K-means Limitations



Original Points



K-means Clusters

Pre-processing and Post-processing

o Pre-processing

- Normalize the data
- Eliminate outliers

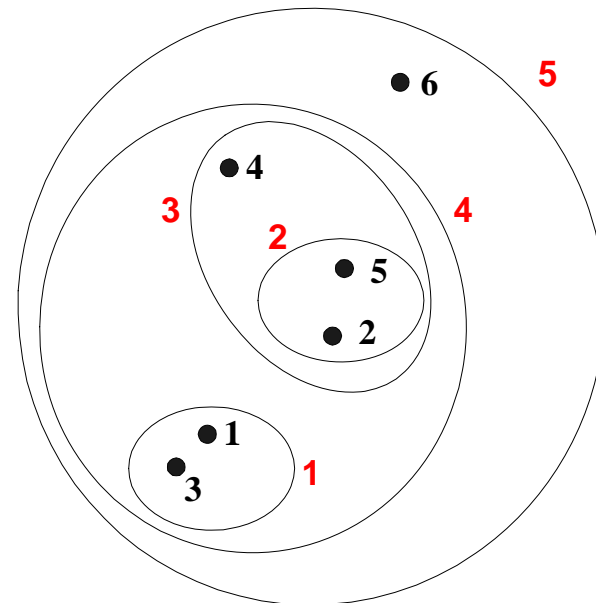
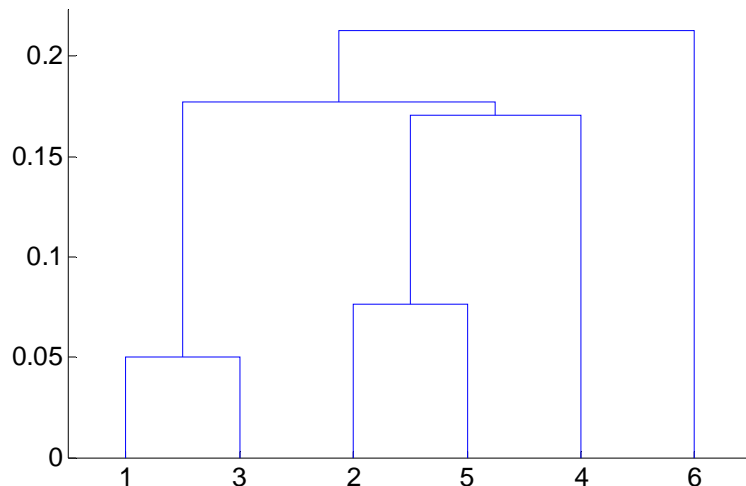
o Post-processing

- Eliminate small clusters that may represent outliers
- Split 'loose' clusters, i.e., clusters with relatively high SSE
- Merge clusters that are 'close' and that have relatively low SSE
- Can use these steps during the clustering process:
ISODATA



Hierarchical Clustering

- Produces a set of nested clusters organized as a hierarchical tree
- Can be visualized as a dendrogram
 - A tree like diagram that records the sequences of merges or splits



Strengths of Hierarchical Clustering

- Do not have to assume any particular number of clusters
 - Any desired number of clusters can be obtained by 'cutting' the dendrogram at the proper level
- They may correspond to meaningful taxonomies
 - Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction, ...)



Hierarchical Clustering

- Two main types of hierarchical clustering
 - Agglomerative:
 - Start with the points as individual clusters
 - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
 - Divisive:
 - Start with one, all-inclusive cluster
 - At each step, split a cluster until each cluster contains a point (or there are k clusters)
- Traditional hierarchical algorithms use a similarity or distance matrix
 - Merge or split one cluster at a time

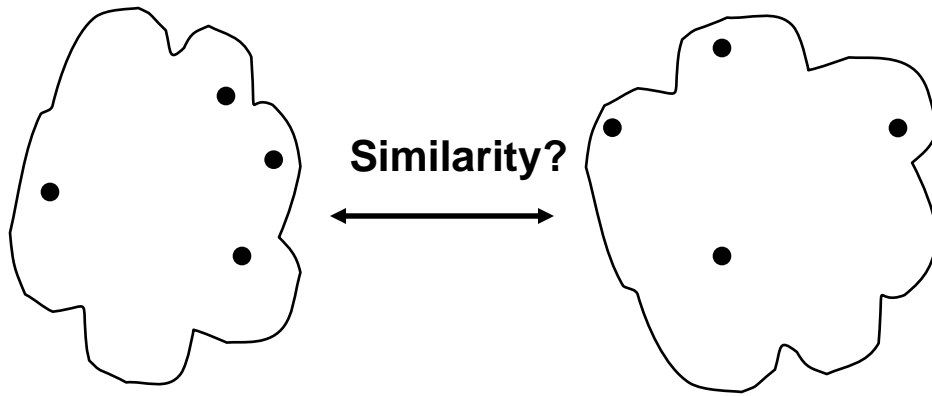


Agglomerative Clustering Algorithm

- o More popular hierarchical clustering technique
- o Basic algorithm is straightforward
 1. Compute the proximity matrix
 2. Let each data point be a cluster
 3. **Repeat**
 4. Merge the two closest clusters
 5. Update the proximity matrix
 6. **Until** only a single cluster remains
- o Key operation is the computation of the proximity of two clusters

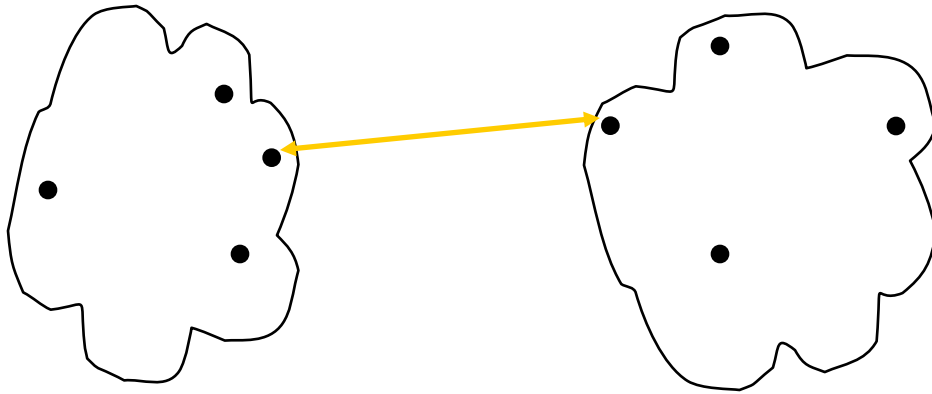


How to Define Inter-Cluster Similarity



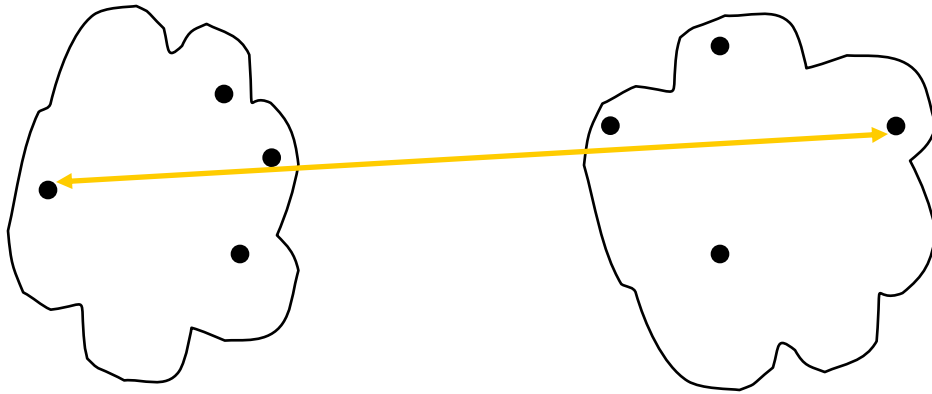
- MIN
- MAX
- Group Average
- Distance Between Centroids

How to Define Inter-Cluster Similarity



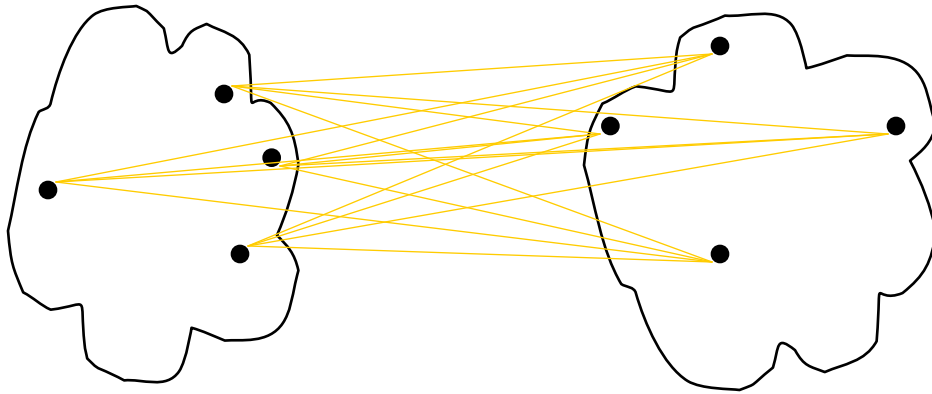
- **MIN**
- MAX
- Group Average
- Distance Between Centroids

How to Define Inter-Cluster Similarity



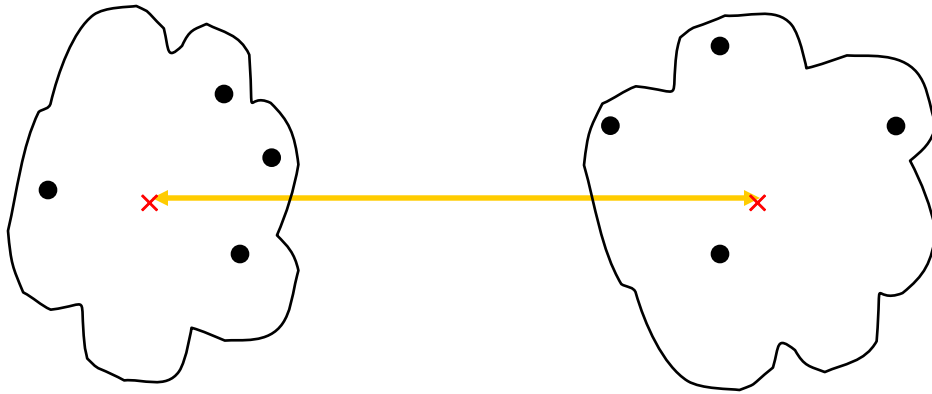
- MIN
- MAX
- Group Average
- Distance Between Centroids

How to Define Inter-Cluster Similarity



- MIN
- MAX
- **Group Average**
- Distance Between Centroids

How to Define Inter-Cluster Similarity



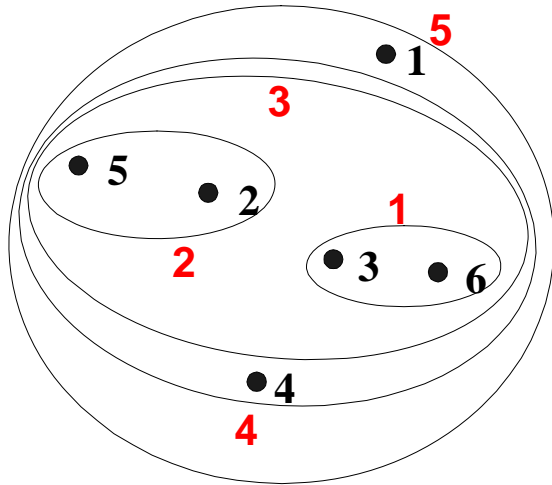
- MIN
- MAX
- Group Average
- **Distance Between Centroids**

Other methods driven by an objective function:

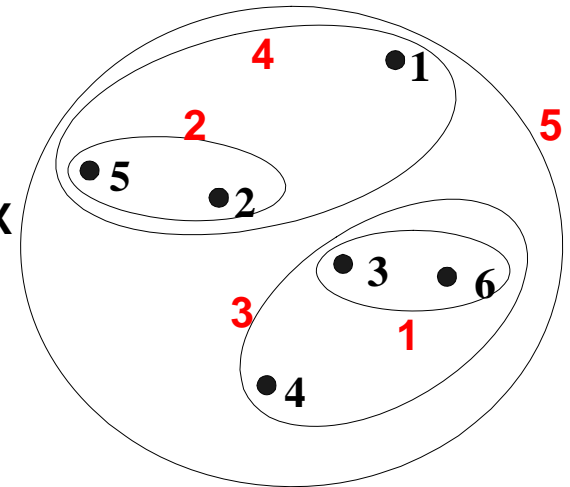
E.g. Ward linkage:

- Similarity based on the increase in SSE when two clusters are merged
- Similar to group average (uses all points)
- HC equivalent to k-means

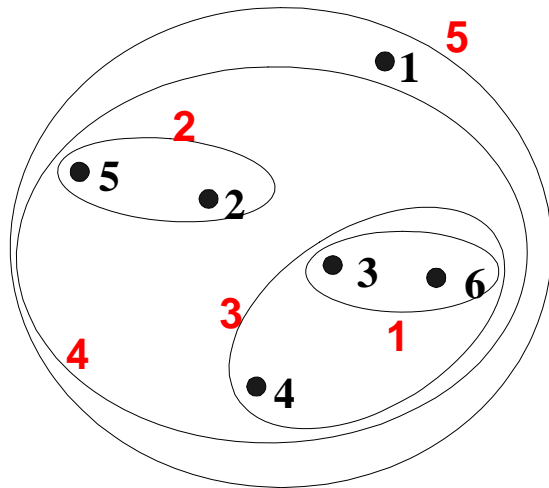
Hierarchical Clustering: Comparison



MIN

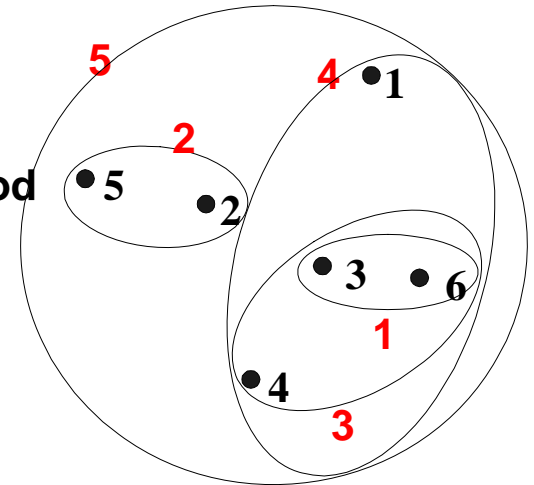


MAX

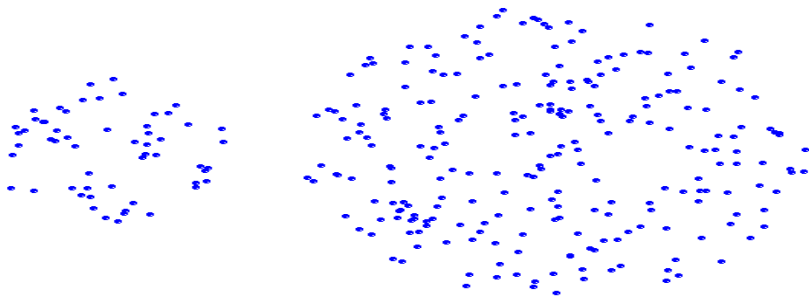


Group Average

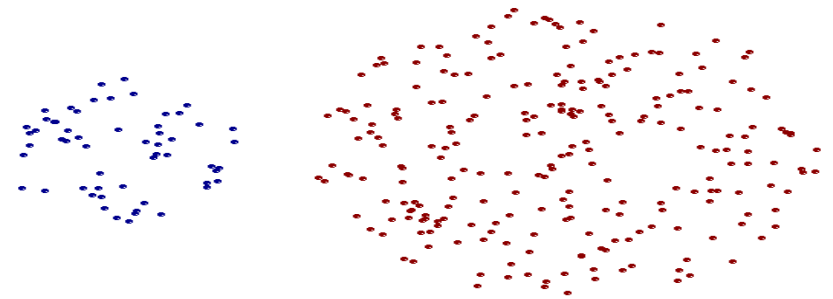
Ward's Method



Strength of MIN



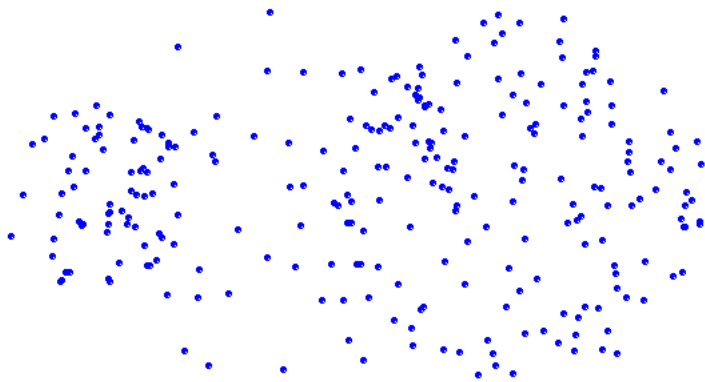
Original Points



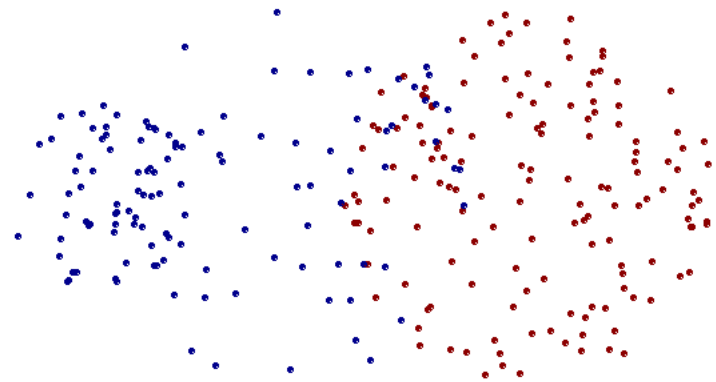
Two Clusters

- **Can handle non-elliptical shapes**

Limitations of MIN



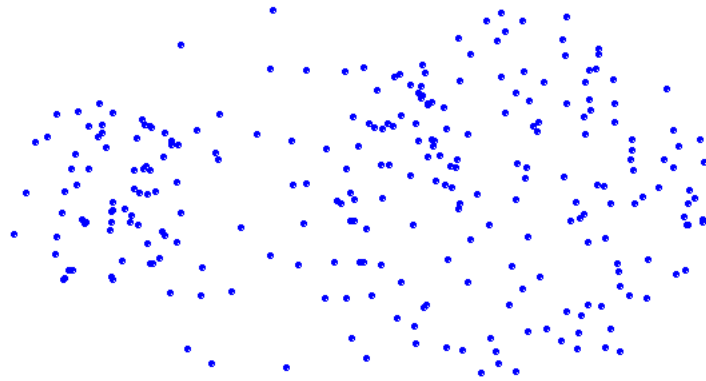
Original Points



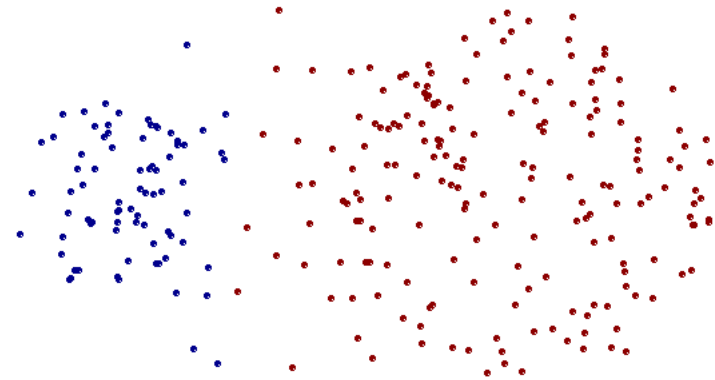
Two Clusters

- **Sensitive to noise and outliers**

Strength of MAX



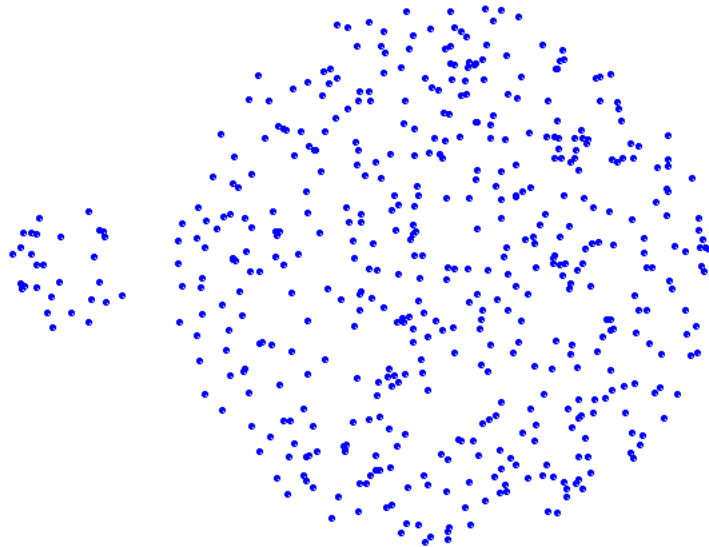
Original Points



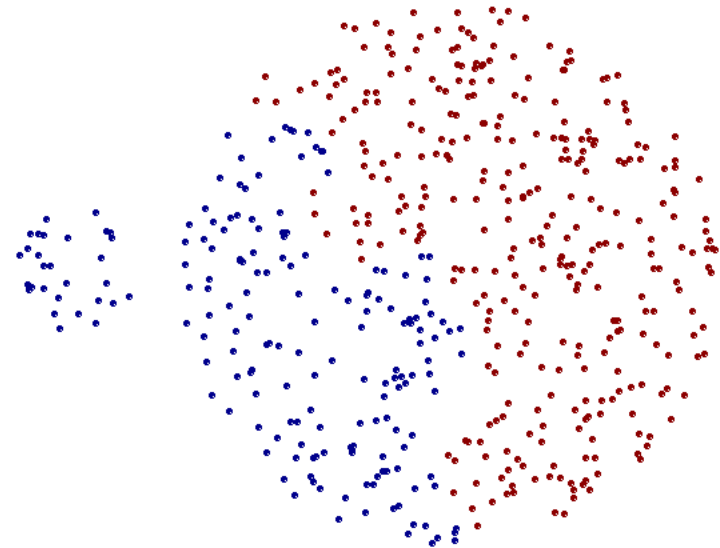
Two Clusters

- **Less susceptible to noise and outliers**

Limitations of MAX



Original Points



Two Clusters

- Tends to break large clusters
- Biased towards globular clusters

Hierarchical Clustering: Group Average + Ward

- o Compromise between Single and Complete Link
- o Strengths: Less susceptible to noise and outliers
- o Limitations: Biased towards globular clusters



Hierarchical Clustering: Time and Space requirements

- $O(N^2)$ space since it uses the proximity matrix.
 - N is the number of points.
- $O(N^3)$ time in many cases
 - There are N steps and at each step the size, N^2 , proximity matrix must be updated and searched
 - Complexity can be reduced to $O(N^2 \log(N))$ time for some approaches



Hierarchical Clustering: Problems and Limitations

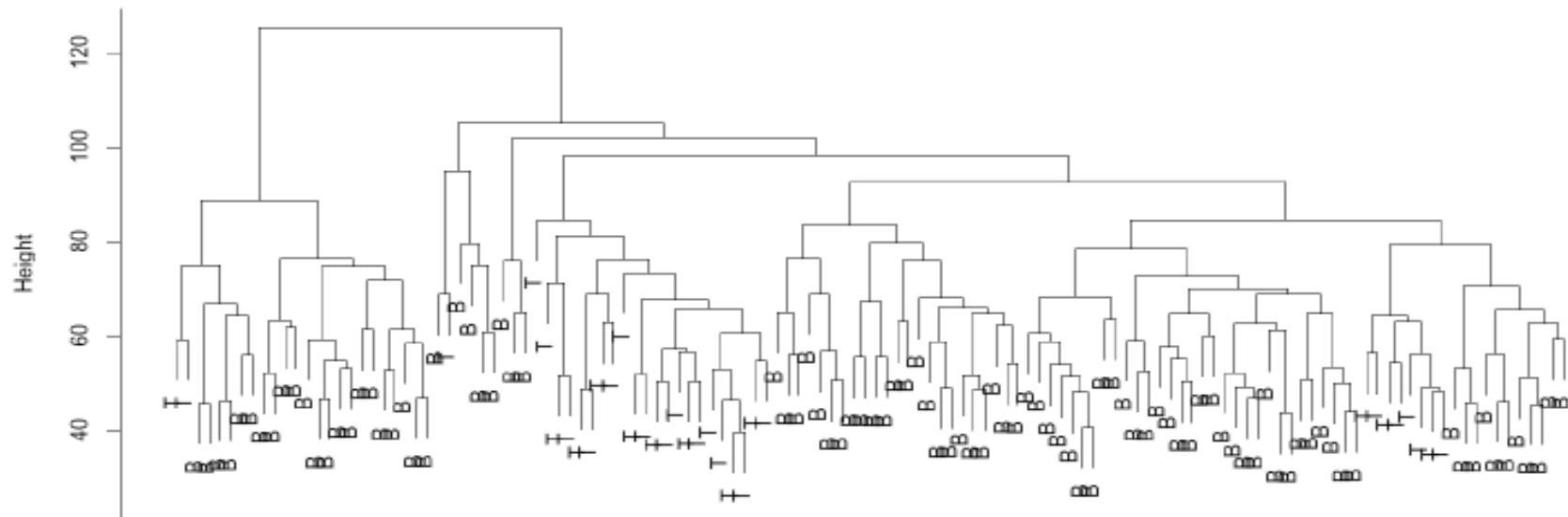
- Once a decision is made to combine two clusters, it cannot be undone
- No objective function is directly minimized
- Different schemes have problems with one or more of the following:
 - Sensitivity to noise and outliers
 - Difficulty handling different sized clusters and convex shapes
 - Breaking large clusters



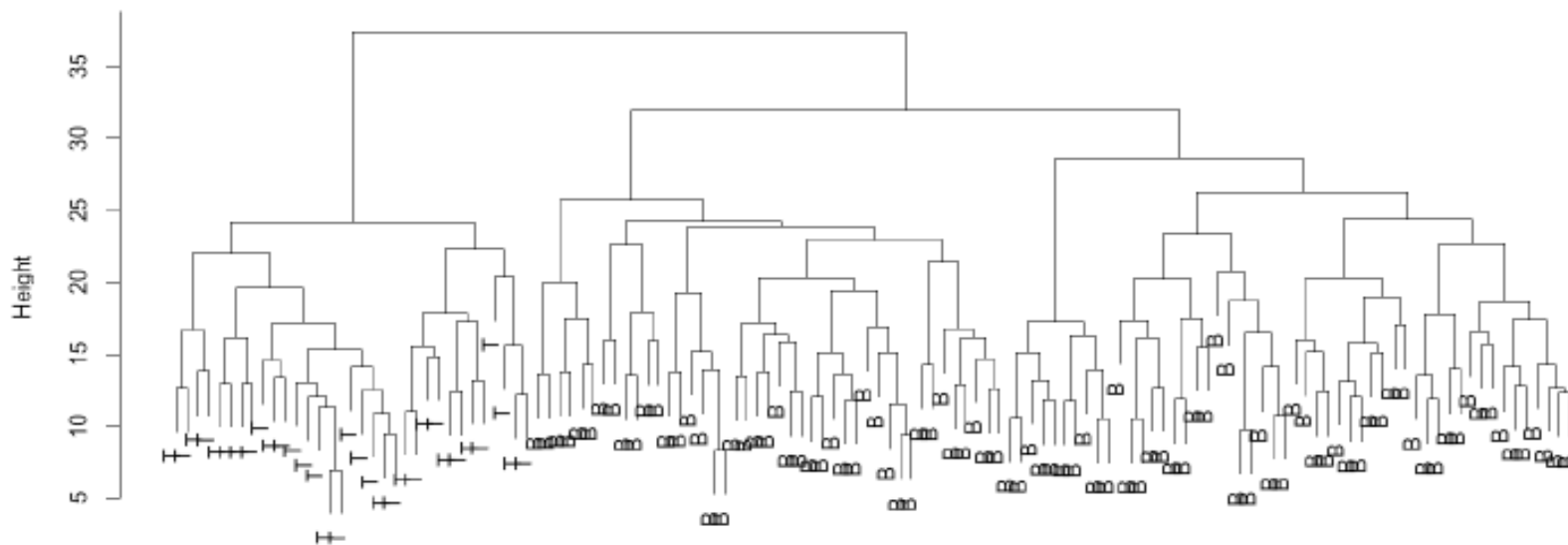
Feature (Gene) Selection as Preprocessing

- Various approaches for gene selection, especially in *supervised* learning.
- Practical unsupervised filtering procedure: choose the top 100-200 genes with respect to variance (across samples).
- Decreases noise and computation time.





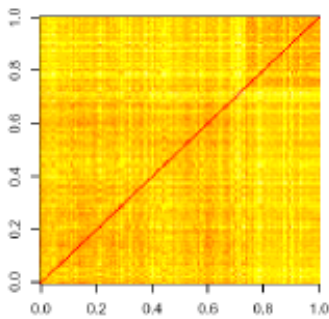
Dendrogram for clustering Leukemia patients (Chiaretti et al., 2004)
without gene selection



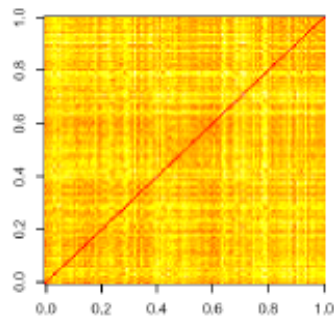
Dendrogram for clustering Leukemia patients (Chiaretti et al., 2004)
with 100 top variance genes

Clustering random genes vs selected genes

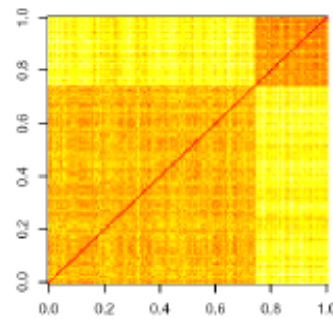
Distance matrices for clustering Leukemia patients (Chiaretti et al., 2004)



All genes

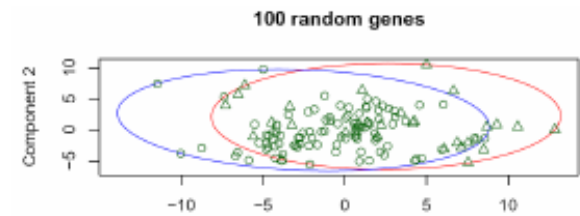


100 random genes

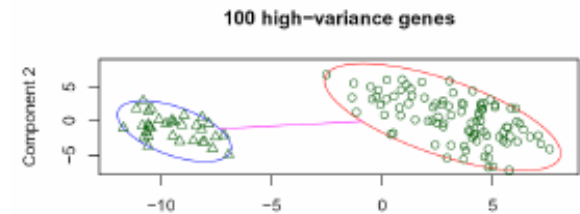


100 high-variance genes

Plot of sample types in first two principal components



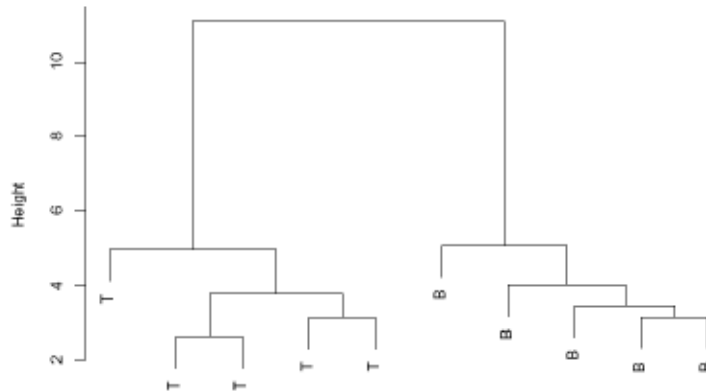
Component 1
These two components explain 30.39 % of the point variability.



Component 1
These two components explain 44.08 % of the point variability.

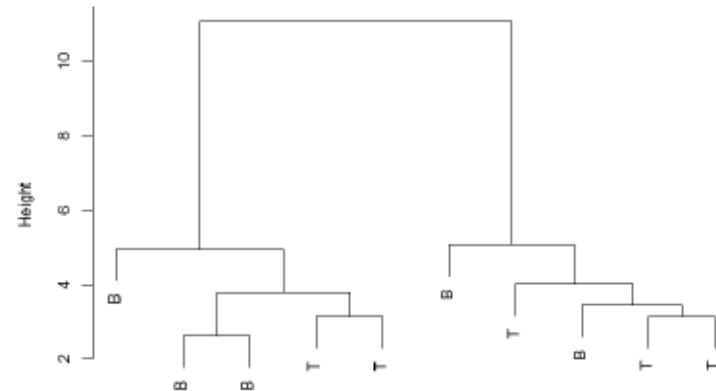
Gene selection and clustering: pitfall

- Do not first select genes based on the outcome of some covariable (e.g. tumor type) and then look at the clustering.
- You will ALWAYS find difference w.r.t. your covariable, since this is how you selected the genes!



Left dendrogram obtained by

1. Random assignment of sample labels
2. Selection of best discriminating genes
3. Clustering with selected genes



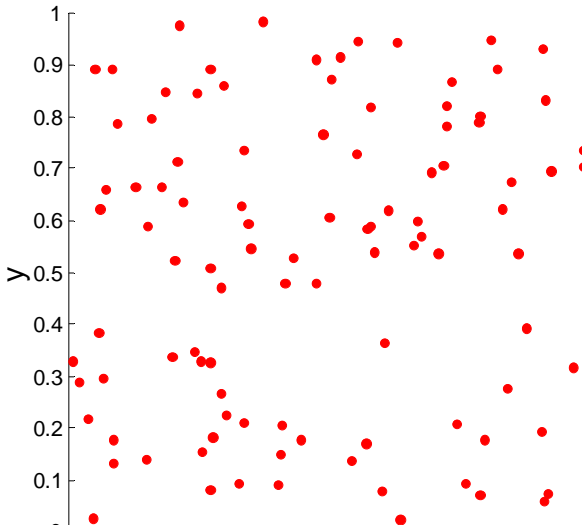
Right plot shows **original labels**

Cluster Validity

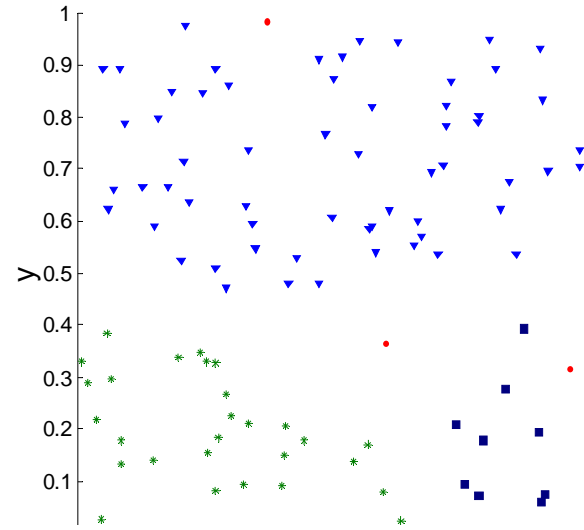
- For supervised classification we have a variety of measures to evaluate how good our model is
 - Accuracy, precision, recall
- For cluster analysis, how to evaluate the “goodness” of the resulting clusters?
- Why do we want to evaluate them?
 - To avoid finding patterns in noise
 - To find the optimal number of clusters
 - To compare clustering algorithms.
 - To compare two sets of clusters (two partitions)

Clusters found in Random Data

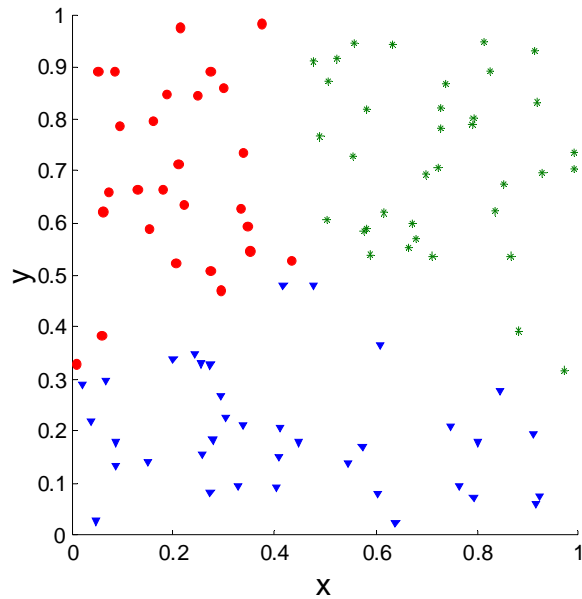
Random
Points



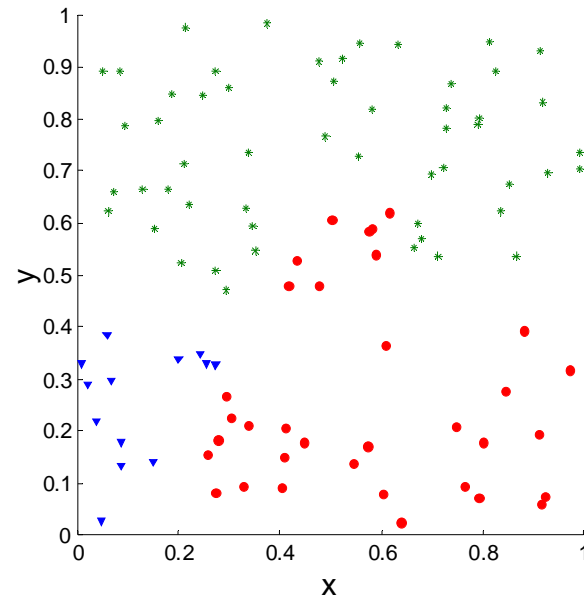
DBSCAN



K-means



Complete
Link



Framework for Cluster Validity

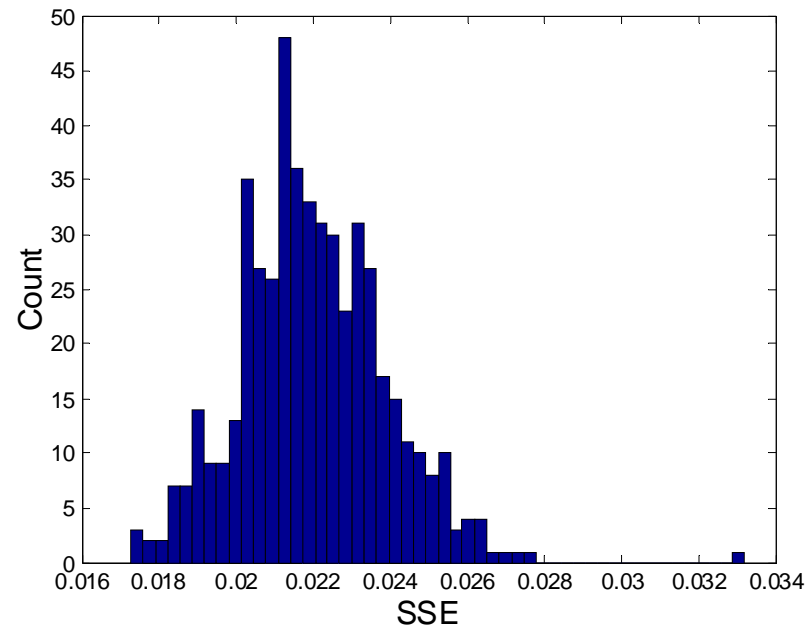
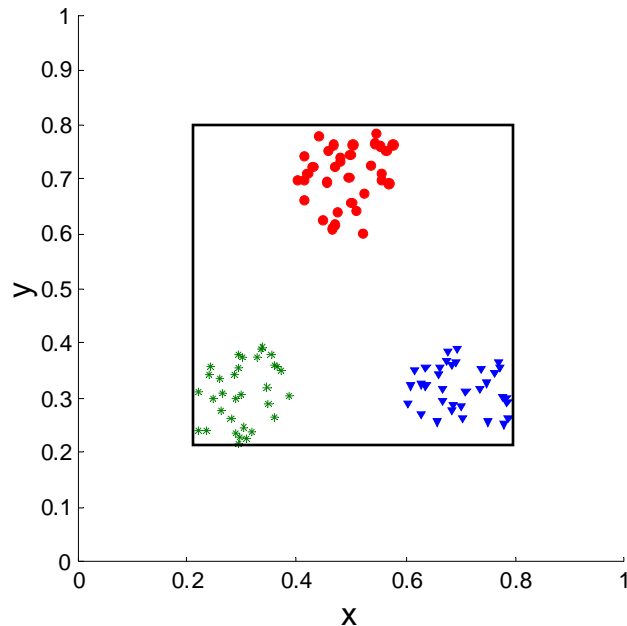
- Need a framework to interpret any measure.
 - For example, if our measure of evaluation has the value, 10, is that good, fair, or poor?
- Statistics provide a framework for cluster validity
 - The more “atypical” a clustering result is, the more likely it represents valid structure in the data
 - Can compare the values of an index that result from random data or clusterings to those of a clustering result.
 - If the value of the index is unlikely, then the cluster results are valid
 - These approaches are more complicated and harder to understand.
- For comparing the results of two different sets of cluster analyses, a framework is less necessary.
 - However, there is the question of whether the difference between two index values is significant



Statistical Framework for SSE

o Example

- Compare SSE of 0.005 against three clusters in random data
- Histogram shows SSE of three clusters in 500 sets of random data points of size 100 distributed over the range 0.2 – 0.8 for x and y values



External criteria

- Compare a given partition with an external gold standard or compare two experimental partitions
- If you have external gold standard indices don't have to be symmetric.
- Gold standard only useful for benchmark studies.



Jaccard and Rand indices

- Given two partitions C and C'
 - N_{11} = # pairs that cluster together in both C and C'
 - N_{00} = # pairs that are in separate clusters in both C and C
 - N_{10} = # pairs that cluster together in C, but not C'
 - N_{01} = # pairs that cluster together in C', but not C
- The two partitions can come from different clusterings or one may be an external truth
- Jaccard index: $N_{11} / (N_{11} + N_{10} + N_{01})$
i.e. fraction of pairs that cluster together in both C and C' relative to those pairs that cluster together in at least one clustering.
- Rand index: $(N_{11} + N_{00}) / (N_{11} + N_{10} + N_{01} + N_{00})$
- Corrected indices introduce a normalization in order to yield values close to zero for random partitions (also attenuate dependence from number of clusters)



External Measures of Cluster Validity: Entropy and Purity

Table 5.9. K-means Clustering Results for LA Document Data Set

Cluster	Entertainment	Financial	Foreign	Metro	National	Sports	Entropy	Purity
1	3	5	40	506	96	27	1.2270	0.7474
2	4	7	280	29	39	2	1.1472	0.7756
3	1	1	1	7	4	671	0.1813	0.9796
4	10	162	3	119	73	2	1.7487	0.4390
5	331	22	5	70	13	23	1.3976	0.7134
6	5	358	12	212	48	13	1.5523	0.5525
Total	354	555	341	943	273	738	1.1450	0.7203

entropy For each cluster, the class distribution of the data is calculated first, i.e., for cluster j we compute p_{ij} , the ‘probability’ that a member of cluster j belongs to class i as follows: $p_{ij} = m_{ij}/m_j$, where m_j is the number of values in cluster j and m_{ij} is the number of values of class i in cluster j . Then using this class distribution, the entropy of each cluster j is calculated using the standard formula $e_j = \sum_{i=1}^L p_{ij} \log_2 p_{ij}$, where the L is the number of classes. The total entropy for a set of clusters is calculated as the sum of the entropies of each cluster weighted by the size of each cluster, i.e., $e = \sum_{i=1}^K \frac{m_i}{m} e_j$, where m_j is the size of cluster j , K is the number of clusters, and m is the total number of data points.

purity Using the terminology derived for entropy, the purity of cluster j , is given by $purity_j = \max p_{ij}$ and the overall purity of a clustering by $purity = \sum_{i=1}^K \frac{m_i}{m} purity_j$.

Corrected indices

- Main problem: on two random partitions the indices do not take a constant value, say zero.
 - difficult to establish, given two partitions, how significant (distant from randomness) is the correlation between the two partitions.
- Adjusted version of an index:
$$(index - expected\ index) / (maximum\ index - expected\ index);$$
where *index* is the formula for the index, *maximum index* is its maximum value and *expected index* is its expected value derived under a suitably chosen model of randomly correlated partitions, i.e., null hypothesis.
- *ARI* uses the generalized hypergeometric distribution as the null hypothesis and the RI expected value can be computed by exact statistical methods

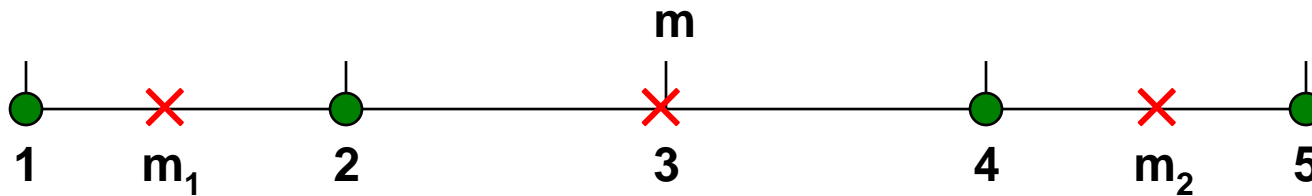
Internal validation criteria/1

- Connectivity: observations are placed in the same cluster as their nearest neighbors?
- Compactness (cohesion): assesses cluster homogeneity, usually by intra-cluster variance (within SSE)
- Separation: quantifies separation between clusters, usually by distance between cluster centroids (between SSE).
- Compactness and separation show opposing trends (compactness increases with the number of clusters, separation decreases) → combine the two measures into a single score
 - Mean
 - Silhouette Width
 - Dunn Index



Cohesion and Separation: example

o Example: SSE



K=1 cluster:

$$WSS = (1 - 3)^2 + (2 - 3)^2 + (4 - 3)^2 + (5 - 3)^2 = 10$$

$$BSS = 4 \times (3 - 3)^2 = 0$$

$$Total = 10 + 0 = 10$$

$$WSS = (1 - 1.5)^2 + (2 - 1.5)^2 + (4 - 4.5)^2 + (5 - 4.5)^2 = 1$$

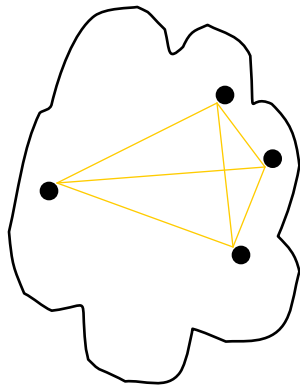
K=2 clusters:

$$BSS = 2 \times (3 - 1.5)^2 + 2 \times (4.5 - 3)^2 = 9$$

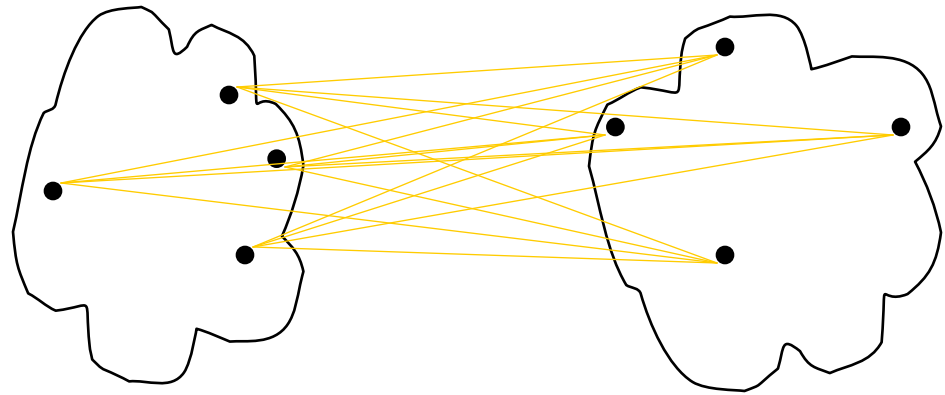
$$Total = 1 + 9 = 10$$

Cohesion and Separation: 2nd definition

- A proximity graph based approach can also be used for cohesion and separation.
 - Cluster cohesion is the sum of the weight of all links within a cluster.
 - Cluster separation is the sum of the weights between nodes in the cluster and nodes outside the cluster.



cohesion



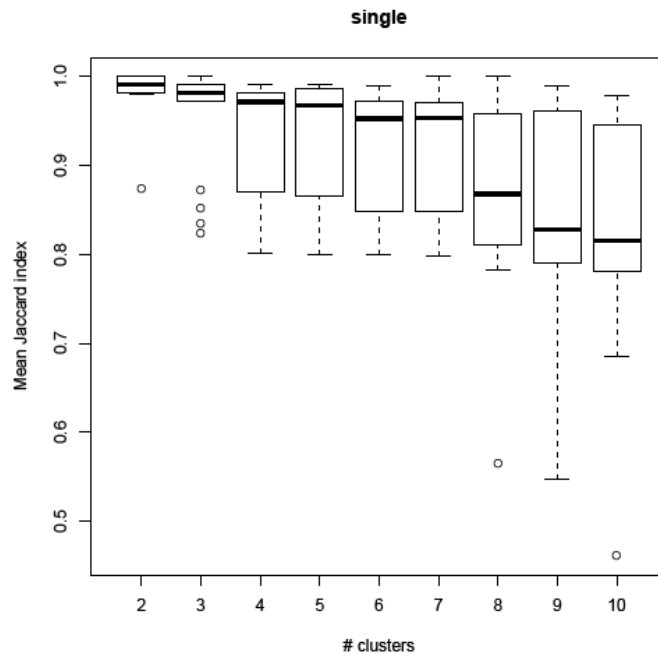
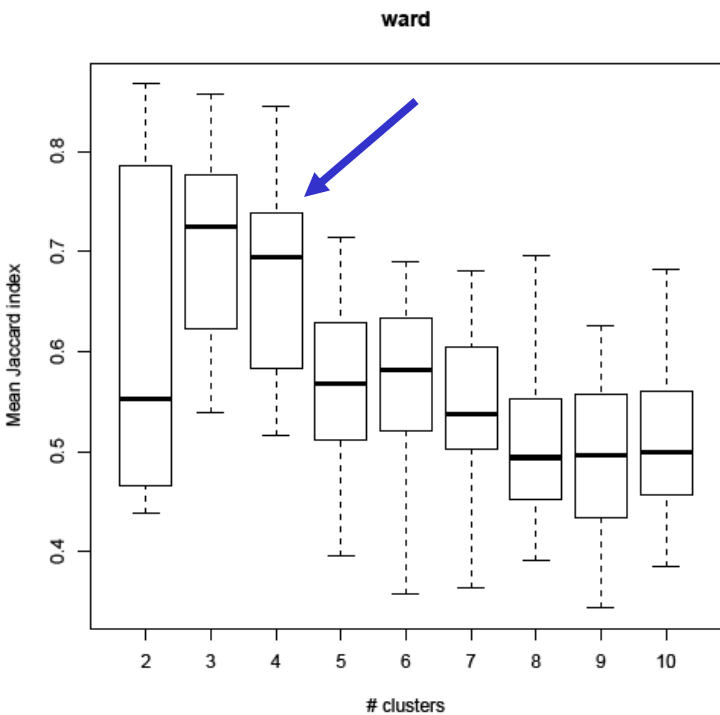
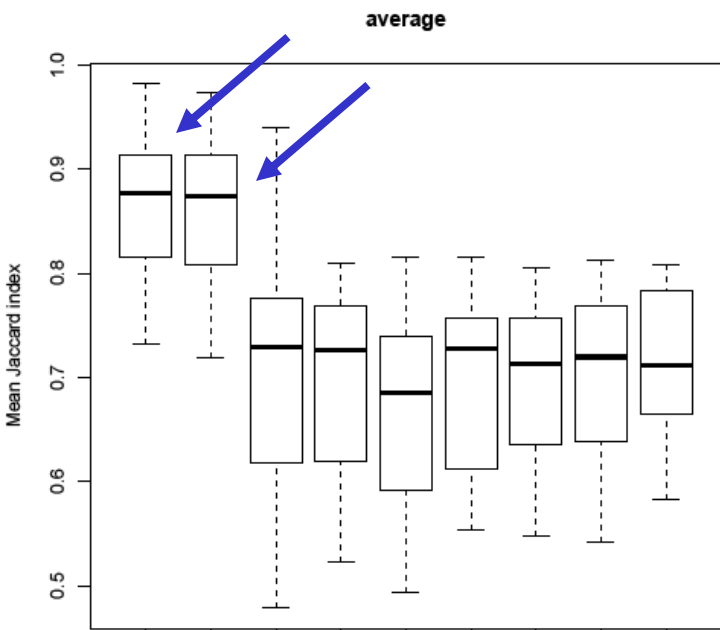
separation

Internal validation criteria/2: Stability indices

- Good clustering solutions should be robust to perturbation in the input data
- cluster stability criterion (Jaccard index): how similar are partitions from perturbed data?
- Type of perturbation:
 - Bootstrap observations
 - (Eliminate features)

Cluster Stability

- Cluster Chip-Seq data
- Unscaled data
- Hierarchical clustering with different linkage functions
- Result: 2, 3, or 4 clusters
- Single link does not give indications (usually good for connectivity)



Connectivity

Let N denote the total number of observations (rows) in a dataset and M denote the total number of columns, which are assumed to be numeric (e.g., a collection of samples, time points, etc.). Define $nn_{i(j)}$ as the j th nearest neighbor of observation i , and let $x_{i,nn_{i(j)}}$ be zero if i and j are in the same cluster and $1/j$ otherwise. Then, for a particular clustering partition $\mathcal{C} = \{C_1, \dots, C_K\}$ of the N observations into K disjoint clusters, the connectivity is defined as

$$Conn(\mathcal{C}) = \sum_{i=1}^N \sum_{j=1}^M x_{i,nn_{i(j)}} .$$

The connectivity has a value between zero and ∞ and should be minimized.



Dunn index

The Dunn Index is the ratio of the smallest distance between observations not in the same cluster to the largest intra-cluster distance. It is computed as

$$D(\mathcal{C}) = \frac{\min_{C_k, C_l \in \mathcal{C}, C_k \neq C_l} \left(\min_{i \in C_k, j \in C_l} dist(i, j) \right)}{\max_{C_m \in \mathcal{C}} diam(C_m)},$$

where $diam(C_m)$ is the maximum distance between observations in cluster C_m . The Dunn Index has a value between zero and ∞ , and should be maximized.

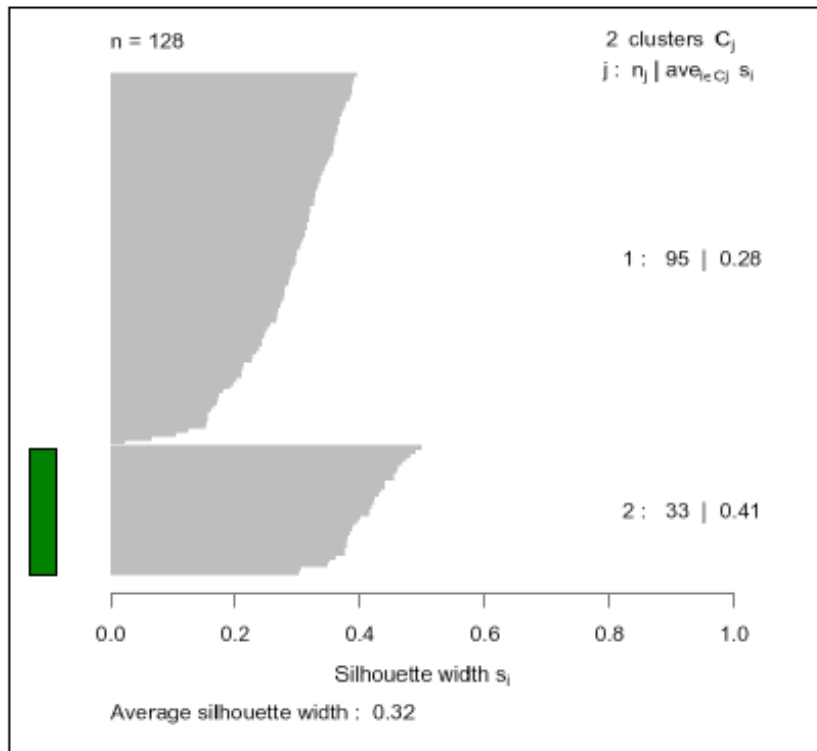
Silhouette: clustering strength for each observation

- Given:
 - $a(i)$ = average dissimilarity between i and all other points of the cluster to which i belongs
 - $d(i,C)$ = average dissimilarity of i to all observations of C .
 - $b(i) := \min_C d(i,C)$ dissimilarity between i and its "neighbor" cluster
- Silhouette width $s(i) := (b(i) - a(i)) / \max(a(i), b(i))$.
- large $s(i)$ (almost 1): observation well clustered, $s(i) \sim 0$: lies between two clusters, negative $s(i)$: probably placed in wrong cluster.
- Mean silhouette across all observations used to validate clustering



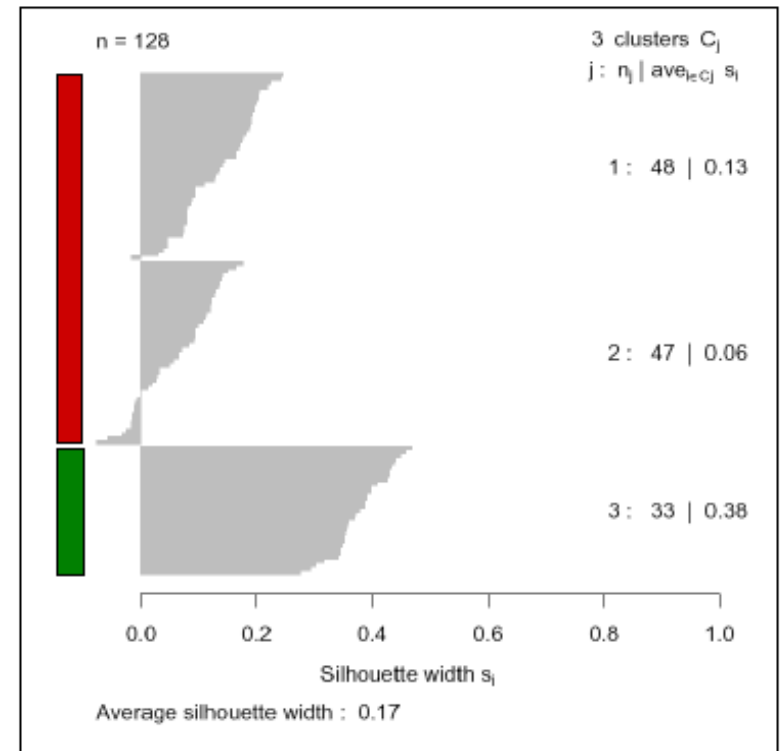
Silhouette plots for Leukemia patients (Chiaretti et al., 2004)

K=2 clusters



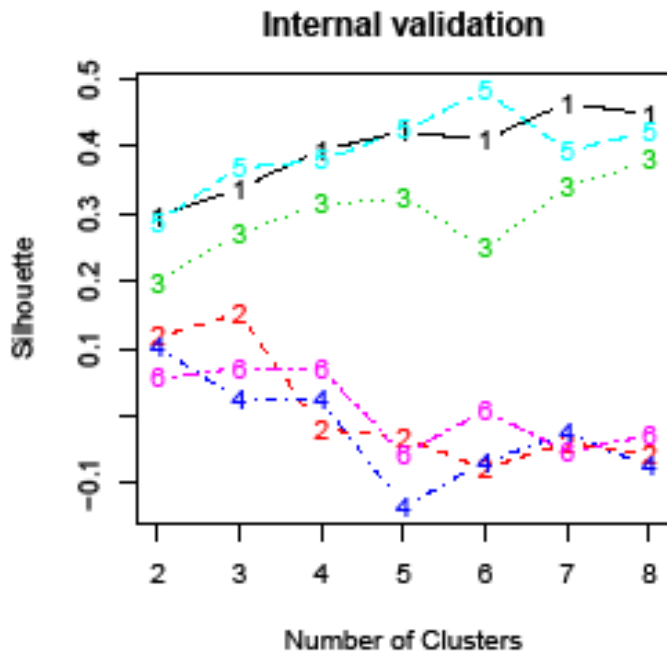
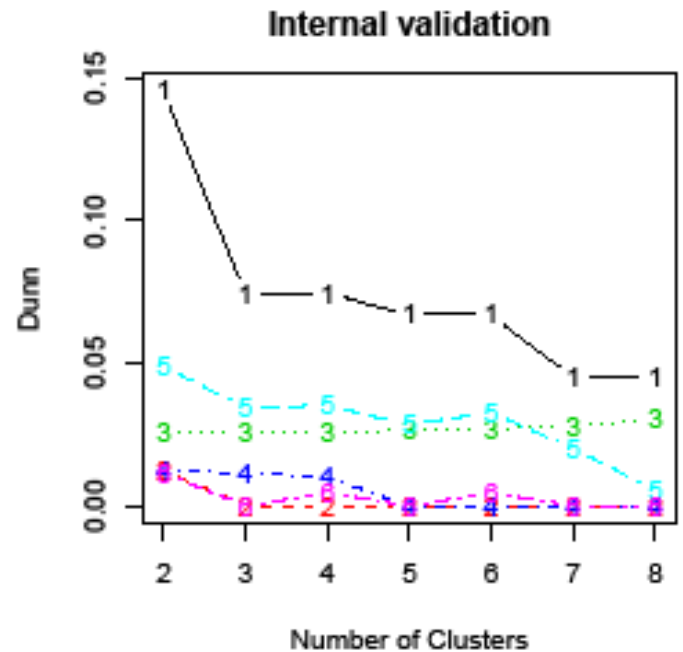
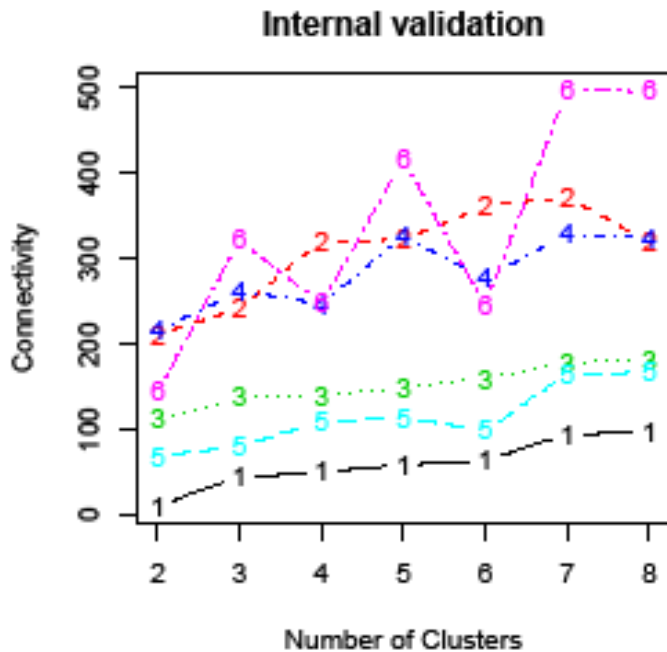
Green: Well separated cluster

K=3 clusters

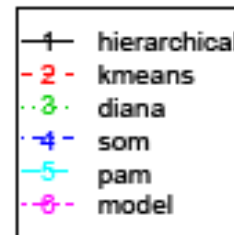


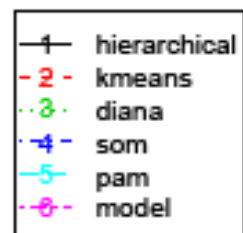
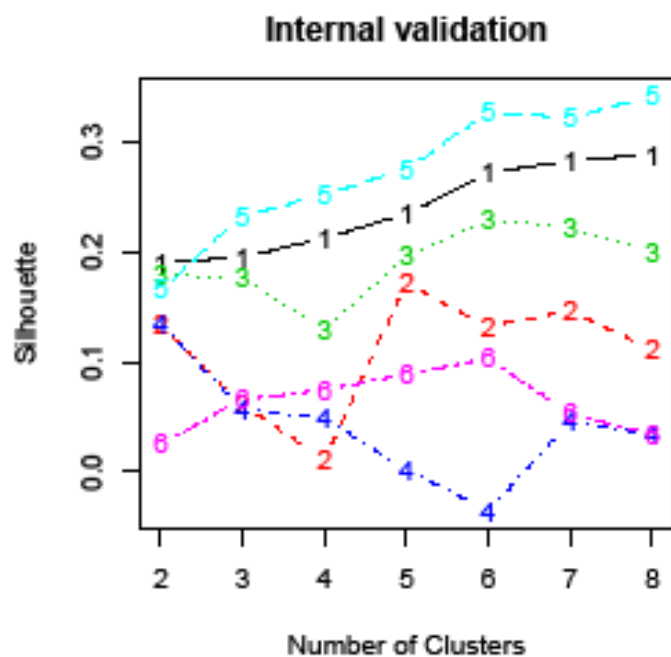
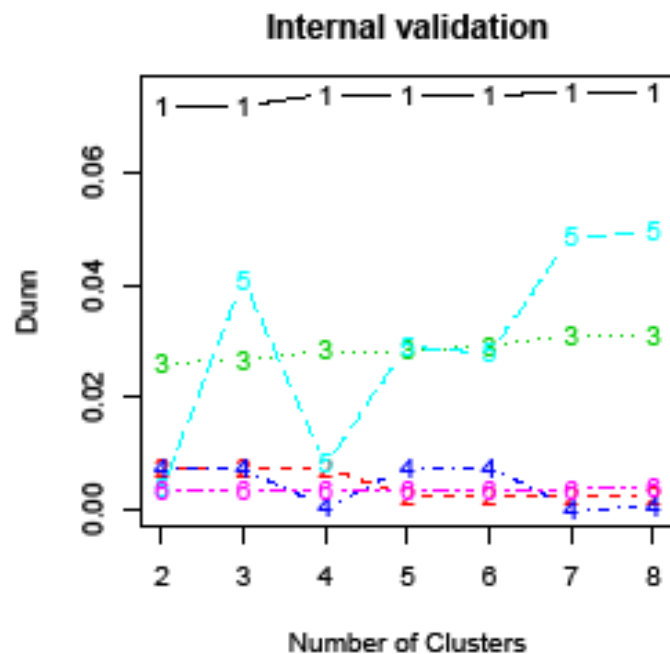
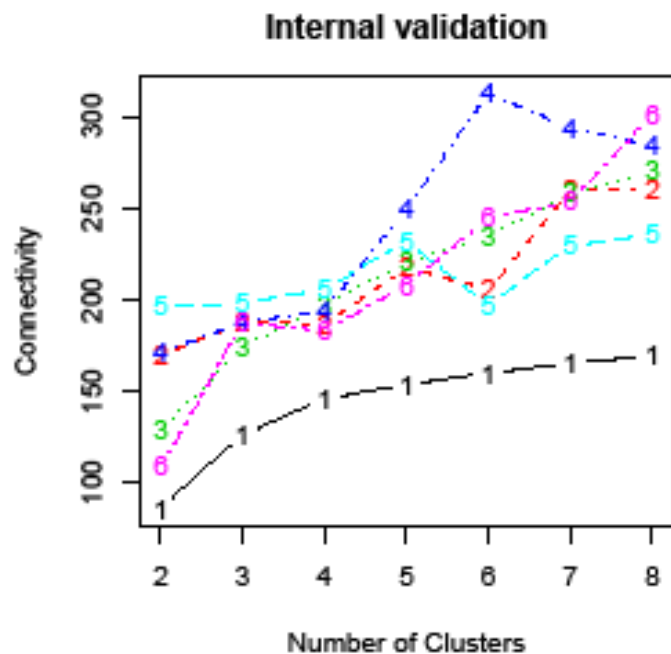
Red: No clear cluster structure

- ChIp-Seq data again
- Unscaled features
- Distance: correlation



Remember: for connectivity low is good, for Dunn and Silhouette high is good





Comments

unscaled.corr

- GLOBALLY (over all cluster numbers): hierarchical (average link.), pam and diana are best, in that order
- for connectivity and dunn: hierarchical best with 2 clusters; for silhouette: pam~hierarchical and best is 6 or 7 clusters

scaled.corr

- GLOBALLY: hierarchical best for connectivity and dunn, pam best for silhouette
- for connectivity 2 clusters are best for almost all algorithms. for dunn: unclear results, values rather flat. for silhouette: pam~hierarchical and best is 6 to 8

scaled.eucl (not shown)

- GLOBALLY: hierarchical and diana best
- for all measures 2 clusters are best. actually from PCA density plot, it can be seen that only one cluster is there → further reason to discard euclidean distance

THEREFORE: GO analysis possible for

- unscaled.corr
 - hierarchical (average linkage): 2,6 clusters
 - pam: 6
 - stability: 2,3,4
- scaled.corr
 - hierarchical: 2,6,7,8
 - pam: 6,7,8
- scaled.eucl: none



Cluster Validity - Comparative Study



- **Comparative study for tumor classification** with microarrays:
Comparison of hierarchical clustering, k-means, PAM and SOM's
- **Data sets:**
 - Golub et al: Leukemia dataset, <http://www.genome.wi.mit.edu/MPR>, 3 cancer classes: 25 acute myeloid leukemia (AML) and 47 acute lymphoblastic leukemia (ALL) (9 T-cell and 38 B-cell), Affymetrix.
 - Ross et al.: NCI60 cancer dataset, <http://genome-www.stanford.edu/nci60>, 9 cancer classes: 9 breast, 6 central nervous system, 7 colon, 8 leukemia, 8 melanoma, 9 lung, 6 ovarian, 2 prostate, 8 renal, cDNA microarray
- **Superiority of k-means** with repeated runs
(Similar for discriminant analysis: FLDA best, Dudoit et al. 2001)
- **Superiority of PAM** with Manhattan distance especially **for noisy data**
- Differences depend on the specific dataset
- Rahnenführer (2002): **Efficient clustering methods for tumor classification with gene expression arrays**, *Proceedings of '26th Annual Conference of the Gesellschaft für Klassifikation'*, Mannheim, July 2002.



Validation: conclusions

- If you want a clear cut results use one clustering method and (maximally) one validation procedure
- If you want some more reproducibility (truth), try more methods and you will have less neat results
- Indices should be corrected for biases with regard to the number of clusters (Handl, 2005): this is not usually the case for the R packages I looked at.
- Personally, have some doubts on Dunn index and stability indices where the perturbation is in the form of eliminating one feature



Landscape of Clustering Algorithms

Anil K. Jain, Alexander Topchy, Martin H.C. Law, and Joachim M. Buhmann[§]

Department of Computer Science and Engineering,

Michigan State University, East Lansing, MI, 48824, USA

[§]*Institute of Computational Science, ETH Zentrum, HRS F31*

Swiss Federal Institute of Technology ETHZ, CH-8092, Zurich, Switzerland

- Several sensible taxonomies exist, e.g. by considering:
 - input data representation, e.g. pattern-matrix or similarity-matrix,
 - data type, e.g. numerical, categorical, or special data structures, such as rank data, strings, graphs
 - output representation, e.g. a partition or a hierarchy of partitions,
 - probability model used (if any),
 - core search (optimization) process,
 - clustering direction, e.g. agglomerative or divisive.



Goal

- Characterization of the landscape of the clustering algorithms in the space of their objective functions.
- However:
 - different objective functions can take drastically different forms and it is very hard to compare them analytically.
 - Also, some clustering algorithms do not have **explicit** objective functions.



A posteriori analysis of the clusters

- alternative characterization of the landscape of the clustering algorithms by a direct comparative analysis of the clusters they detect.
- such an empirical view of the clustering landscape depends on the data sets
- two scenarios:
 - (i) average-case landscape of the variety of clustering algorithms over a number of real world data sets,
 - (ii) a landscape over artificial data sets generated by mixtures of Gaussian components.
- multidimensional scaling employed to visualize the landscape



Way out

- Derive the underlying objective function of known clustering algorithms and the corresponding general description of clustering solutions.
- For example, classical agglomerative algorithms, including single-link (SL), average-link (AL) and complete-link (CL), have quite complex underlying probability models.
 - SL algorithm \rightarrow mixture of branching random walks,
 - AL algorithm \rightarrow equivalent to finding the maximum likelihood estimate of the parameters of a stochastic process with Laplacian conditional probability densities.
- Too complex



Relationship between the clustering algorithms

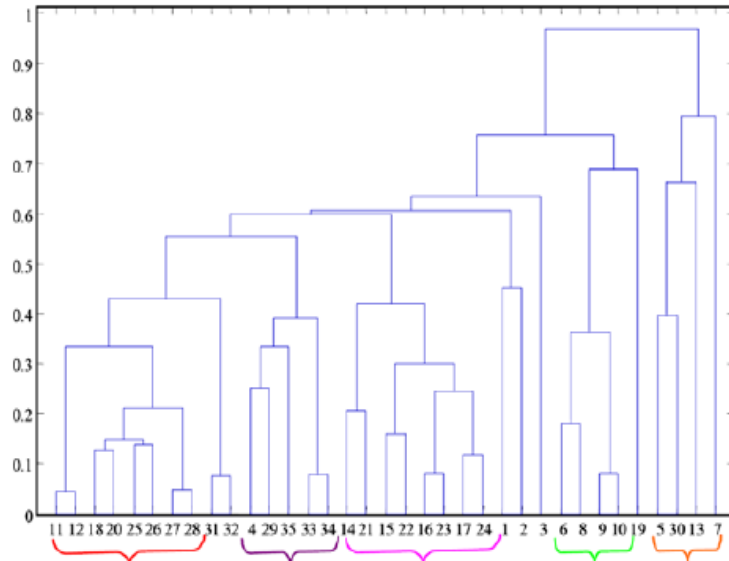
- Estimated by Rand's index of partition similarity
- Rand's index: $(n_{CC} + n_{CC'})/N$,
 n_{CC} : pairs of objects assigned to the same cluster;
 $n_{CC'}$: pairs of objects assigned to different clusters in both the partitions (N total number of pairs)

Experiments

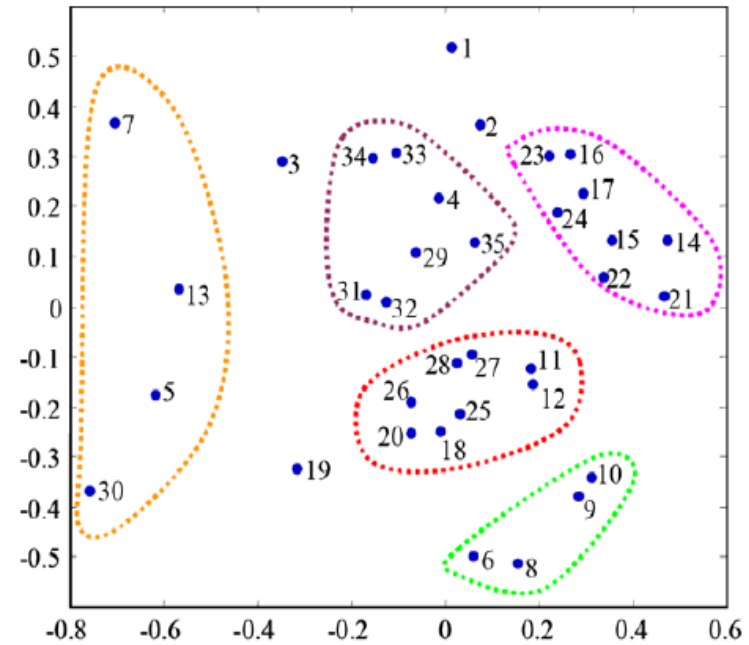
- o Rand's index averaged over 12 different UCI datasets for 35 different clustering criteria.
- o Generated 12 datasets with three 2-dimensional Gaussian clusters.
 - a) datasets differed in the degree of separation between clusters.
 - b) datasets differed in the degree of sparseness (density) of two of the clusters.



Cluster the clustering



(a)



(b)

Fig. 10. Clustering of clustering algorithms. (a) Hierarchical clustering of 35 different algorithms; (b) Sammon's mapping of the 35 algorithms into a two-dimensional space, with the clusters highlighted for visualization. The algorithms in the group (4, 29, 31–35) correspond to K-means, spectral clustering, Gaussian mixture models, and Ward's linkage. The algorithms in group (6, 8–10) correspond to CHAMELEON algorithm with different objective functions.

Results

- Five main families
- k -means and hierarchical clustering with Ward linkage placed in the center of the landscape.
- The majority of the clustering solutions have intrinsic aggregations (some algorithms like SL are “outliers”): Chameleon, Cure/graph partitioning, k -means/spectral/EM are representatives of the different groups.
- The parameters of the algorithms (other than the number of clusters) are of less importance.
- Landscape visualization suggests a simple recipe that includes k -means, graph-partitioning and linkage-based algorithms.



Trends in data clustering (pointers)

- Clustering ensembles
- Semi-supervised clustering
- Large-scale clustering (up to millions of data points represented in thousands of features)
- Multi-way clustering (e.g. biclustering, or clustering using different sets of features)
- Heterogeneous data clustering (objects not naturally represented by a fixed length feature vector).



Clustering ensembles

- The new similarity between a pair of points is defined as the number of times the two points co-occur in the same cluster in N runs of a clustering algorithm.

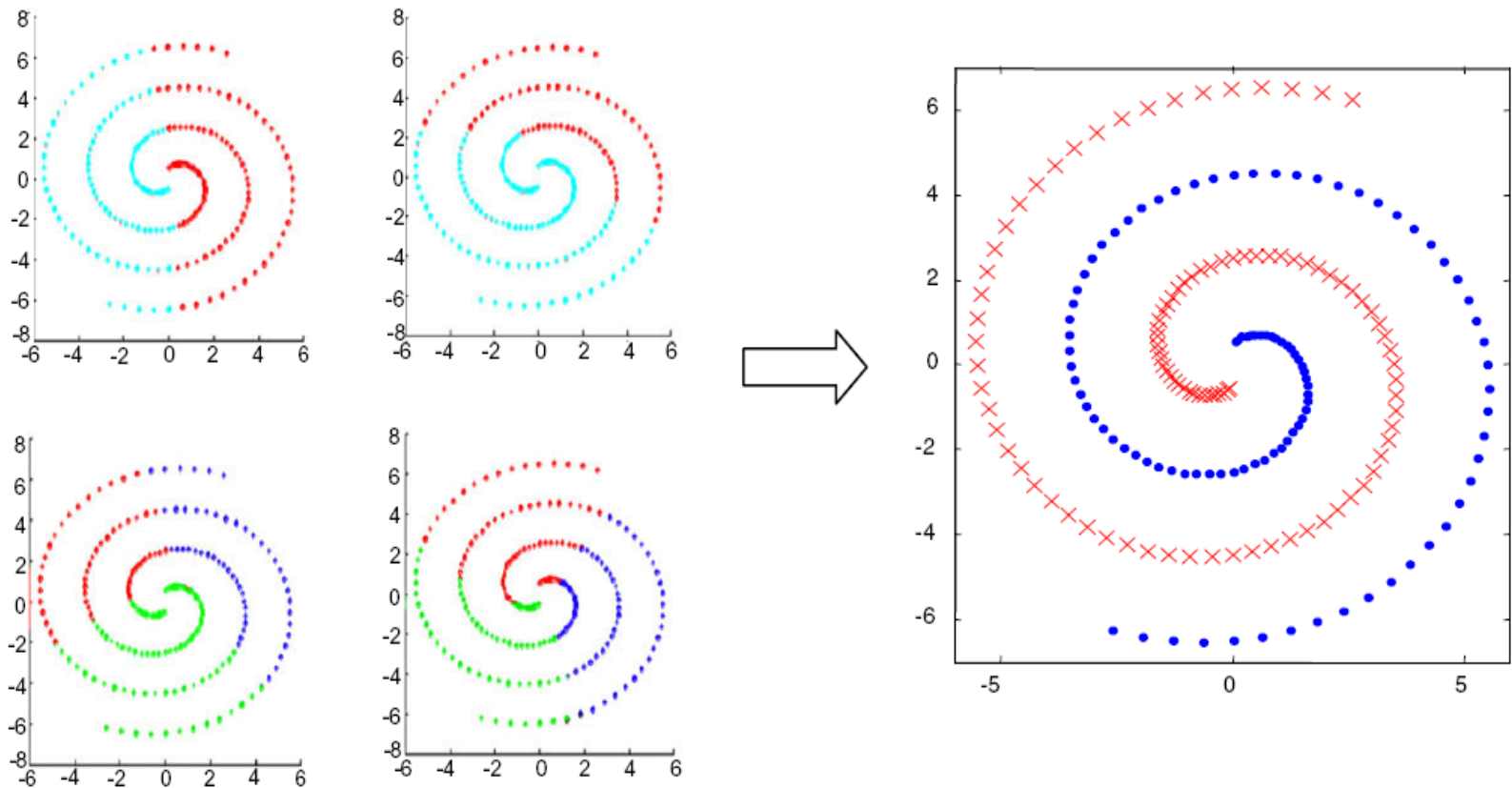


Fig. 11. Clustering ensembles. Multiple runs of K-means are used to learn the pair-wise similarity using the “co-occurrence” of points in clusters. This similarity can be used to detect arbitrary shaped clusters.

Semi-supervised clustering

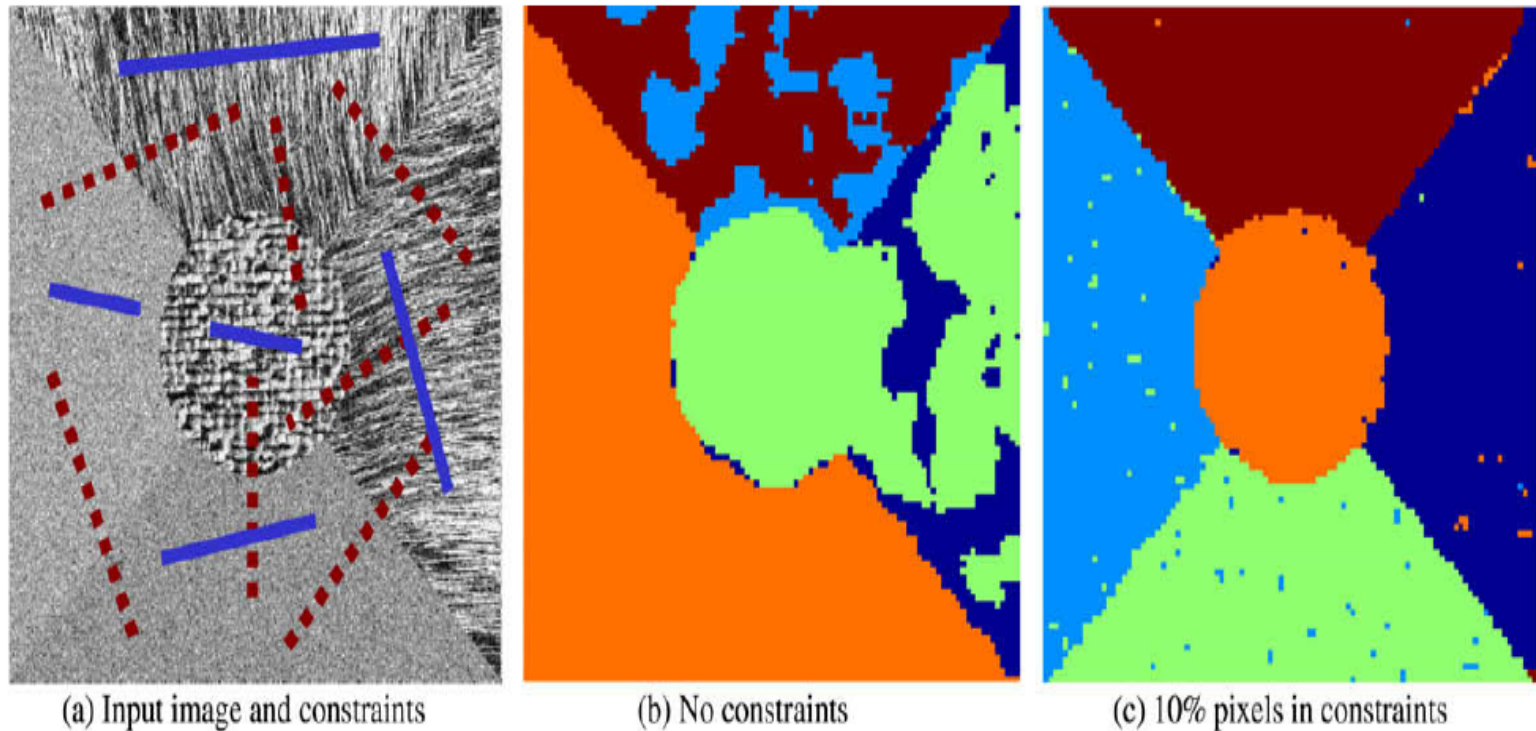


Fig. 12. Semi-supervised learning. (a) Input image consisting of five homogeneous textured regions; examples of must-link (solid blue lines) and must not link (broken red lines) constraints between pixels are indicated. (b) Clustering (segmentation) without constraints. (c) Clustering (segmentation) with five clusters (with 10% of the c

In the conclusions: "A fundamental issue related to clustering is its stability or consistency. A good clustering principle should result in a data partitioning that is stable with respect to perturbations in the data."

Summary

- There is a deluge of clustering methods, and each of them will produce clusters: **clustering cannot NOT work**
- A validation step is needed because:
 - Bias of clustering algorithms towards partitions that are in accordance with their own clustering criterion.
This explains fundamental discrepancies between the solutions produced by different algorithms.
 - Non-significance of results in the absence of natural clusters.
- In practice, results depend a lot on: distance measure and data normalization
- Distance measure and data normalization should really be dictated by biological knowledge/interest



Acknowledgements

- Jane Fridlyand, Jean Yee Hwa Yang (a few slides)
www.cbs.dtu.dk/courses/norfa2004/Extras/Jane-introduction.ppt
- Jörg Rahnenführer (a few more slides)
<http://compdiag.molgen.mpg.de/ngfn/docs/2007/sep/Clustering.pdf>
- Tan, Steinbach, Kumar (a lot more slides)
<http://www-users.cs.umn.edu/~kumar/dmbook/>

