Data Mining

Prof. Dr. Nizamettin AYDIN

naydin@yildiz.edu.tr

http://www3.yildiz.edu.tr/~naydin

1

1

Data Mining

Cluster Analysis

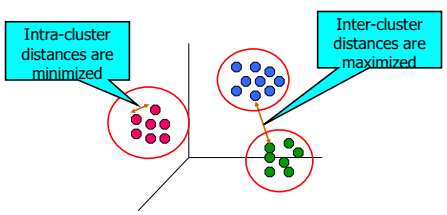
- Outline
  - Overview
  - K-means
  - Agglomerative Hierarchical Clustering
  - DBSCAN
  - Cluster Evaluation
  -

2

2

What is Cluster Analysis?

- Cluster analysis divides data into groups (clusters) that are meaningful, useful,or both.
  - Given a set of objects, place them in groups such that the objects in a group are similar (or related) to one another and different from (or unrelated to) the objects in other groups



3

3

Applications of Cluster Analysis

- Clustering for Understanding
  - classes, or conceptually meaningful groups of objects that share common characteristics
    - In the context of understanding data, clusters are potential classes and cluster analysis is the study of techniques for automatically finding classes
    - Examples:
      - Biology
      - Information Retrieval
      - Climate
      - Psychology and Medicine
      - Business

4

4

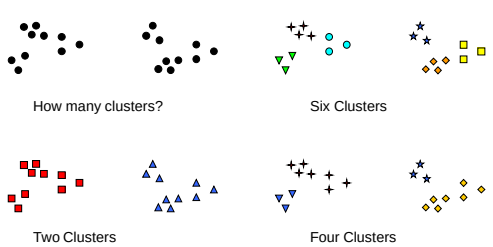
Applications of Cluster Analysis

- Clustering for Utility
  - Cluster analysis provides an abstraction from individual data objects to the clusters in which those data objects reside
    - In the context of utility, cluster analysis is the study of techniques for finding the most representative cluster prototypes
    - Examples:
      - Summarization
      - Compression
      - Efficiently Finding Nearest Neighbors

5

5

Notion of a Cluster can be Ambiguous



6

6

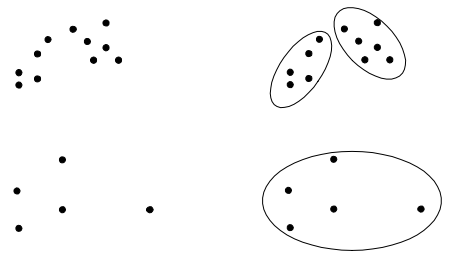
Types of Clusterings

- A clustering is a set of clusters
- Important distinction between hierarchical and partitional sets of clusters
  - Partitional (unnested) Clustering
    - A division of data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset
  - Hierarchical (nested) clustering
    - A set of nested clusters organized as a hierarchical tree
    - Each node (cluster) in the tree (except for the leaf nodes) is the union of its children (subclusters), and the root of the tree is the cluster containing all the objects

7

7

Partitional Clustering



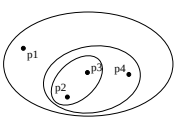
Original Points

A Partitional Clustering

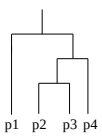
8

8

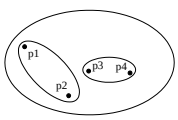
Hierarchical Clustering



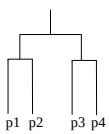
Traditional Hierarchical Clustering



Traditional Dendrogram



Non-traditional Hierarchical Clustering



Non-traditional Dendrogram

9

9

Other Distinctions Between Sets of Clusters

- Exclusive versus non-exclusive
  - In non-exclusive clusterings, points may belong to multiple clusters.
    - Can belong to multiple classes or could be 'border' points
  - Fuzzy clustering (one type of non-exclusive)
    - In fuzzy clustering, a point belongs to every cluster with some weight between 0 and 1
    - Weights must sum to 1
    - Probabilistic clustering has similar characteristics
- Partial versus complete
  - In some cases, we only want to cluster some of the data

10

10

Types of Clusters

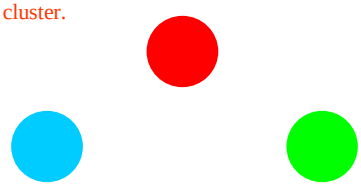
- Well-separated clusters
- Prototype-based clusters
- Contiguity-based clusters
- Density-based clusters
- Described by an Objective Function

11

11

Types of Clusters

- Well-Separated Clusters:
  - A cluster is a set of points such that any point in a cluster is closer (or more similar) to every other point in the cluster than to any point not in the cluster.



3 well-separated clusters

12

12

Types of Clusters

- Prototype-based (center-based) Clusters
  - A cluster is a set of objects such that an object in a cluster is closer (more similar) to the prototype or "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

13

13

Types of Clusters

- Graph-Based (Contiguity-Based) Clusters
  - A cluster is a set of points such that a point in a cluster is closer (or more similar) to one or more other points in the cluster than to any point not in the cluster.

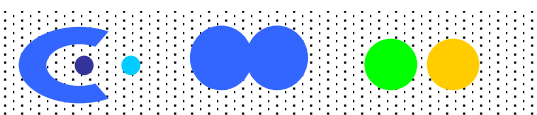
8 contiguous clusters

14

14

Types of Clusters

- Density-based Clusters
  - 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.

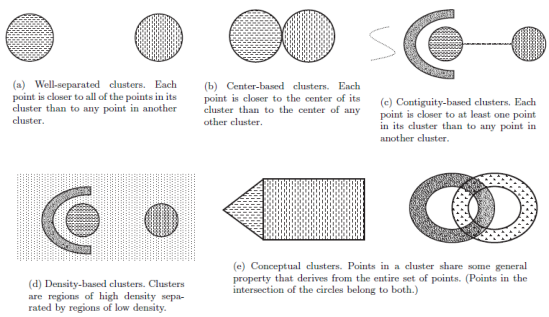


6 density-based clusters

15

15

Types of Clusters



(a) Well-separated clusters. Each point is closer to all of the points in its cluster than to any point in another cluster.

(b) Center-based clusters. Each point is closer to the center of its cluster than to the center of any other cluster.

(c) Contiguity-based clusters. Each point is closer to at least one point in its cluster than to any point in another cluster.

(d) Density-based clusters. Clusters are regions of high density separated by regions of low density.

(e) Conceptual clusters. Points in a cluster share some general property that derives from the entire set of points. (Points in the intersection of the circles belong to both.)

16

16

Types of Clusters

- Clusters Defined by an Objective Function
  - Finds clusters that minimize or maximize an objective function.
  - Enumerate all possible ways of dividing the points into clusters and evaluate the `goodness' of each potential set of clusters by using the given objective function. (NP Hard)
  - Can have global or local objectives.
    - Hierarchical clustering algorithms typically have local objectives
    - Partitional algorithms typically have global objectives
  - A variation of the global objective function approach is to fit the data to a parameterized model.
    - Parameters for the model are determined from the data.
    - Mixture models assume that the data is a 'mixture' of a number of statistical distributions.

17

17

Characteristics of the Input Data Are Important

- Type of proximity or density measure
  - Central to clustering
  - Depends on data and application
- Data characteristics that affect proximity and/or density are
  - Dimensionality
    - Sparseness
  - Attribute type
  - Special relationships in the data
    - For example, autocorrelation
  - Distribution of the data
- Noise and Outliers
  - Often interfere with the operation of the clustering algorithm
- Clusters of differing sizes, densities, and shapes

18

18

Clustering Algorithms

- Techniques to introduce many of the concepts involved in cluster analysis:
  - K-means and its variants
    - a prototype-based, partitional clustering technique that attempts to find a user-specified number of clusters (K), which are represented by their centroids
  - Agglomerative Hierarchical clustering
    - a collection of closely related clustering techniques that produce a hierarchical clustering by starting with each point as a singleton cluster and then repeatedly merging the two closest clusters until a single, all encompassing cluster remains
  - Density-based clustering

19

19

K-means Clustering

- Prototype-based clustering techniques create a one-level partitioning of the data objects.
  - K-means
    - defines a prototype in terms of a centroid, which is usually the mean of a group of points, and is typically applied to objects in a continuous n-dimensional space.
  - K-medoid
    - defines a prototype in terms of a medoid, which is the most representative point for a group of points, and can be applied to a wide range of data since it requires only a proximity measure for a pair of objects.
  - While a centroid almost never corresponds to an actual data point, a medoid, by its definition, must be an actual data point

20

20

K-means Clustering

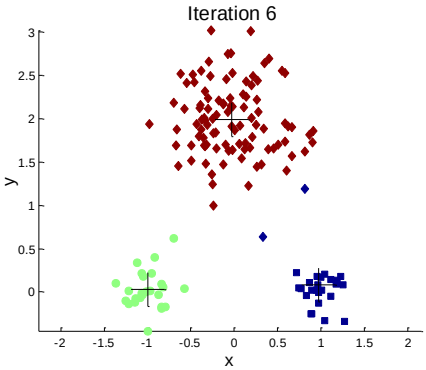
- Partitional clustering approach
- Number of clusters, K, must be specified
- Each cluster is associated with a centroid (center point)
- Each point is assigned to the cluster with the closest centroid
- The basic algorithm is very simple:

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

21

21

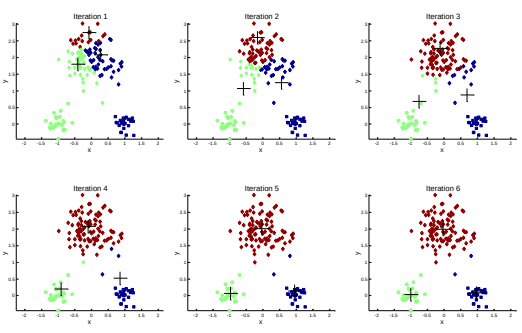
Example of K-means Clustering



22

22

Example of K-means Clustering



23

23

K-means Clustering – Details

- Simple iterative algorithm.
  - Choose initial centroids;
  - repeat {assign each point to a nearest centroid; re-compute cluster centroids}
  - until centroids stop changing.
- Initial centroids are often chosen randomly.
  - Clusters produced can vary from one run to another
- The centroid is (typically) the mean of the points in the cluster, but other definitions are possible (see Table 7.2).
- K-means will converge for common proximity measures with appropriately defined centroid (see Table 7.2)
- 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

24

24

K-means Objective Function

- A common objective function (used with Euclidean distance measure) is Sum of Squared Error (SSE) , which is also known as scatter
  - For each point, the error is the distance to the nearest cluster center
  - To get SSE, we square these errors and sum them.

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

  x is a data point in cluster C_i , c_i is the centroid for cluster C_i , m_i is the number of objects in the i^{th} cluster
  - SSE improves in each iteration of K-means until it reaches a local or global minima.

25

25

K-means Objective Function

- To illustrate, the centroid of a cluster containing the three two-dimensional points,
  (1,1), (2,3), and (6,2),
- Centroid is

$$c_i = \frac{1}{m_i} \sum_{x \in C_i} x$$

((1 + 2 + 6) / 3, (1 + 3 + 2) / 3) = (3, 2)

26

26

K-means Objective Function

- K-means is not restricted to data in Euclidean space
- Consider document data and the cosine similarity measure.
  - Assume that the document data is represented as a document-term matrix
- Objective is to maximize the similarity of the documents in a cluster to the cluster centroid;
  - this quantity is known as the cohesion of the cluster.
- The analogous quantity to the total SSE is the total cohesion, which is given by

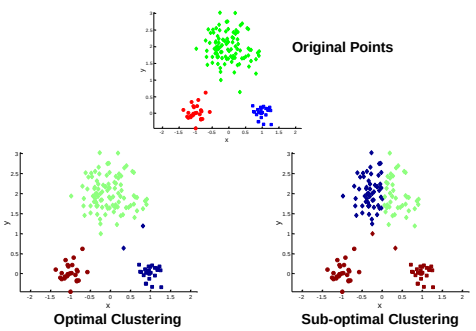
$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} cosine(c_i, x)$$

x is a data point in cluster C_i , c_i is the centroid for cluster C_i

27

27

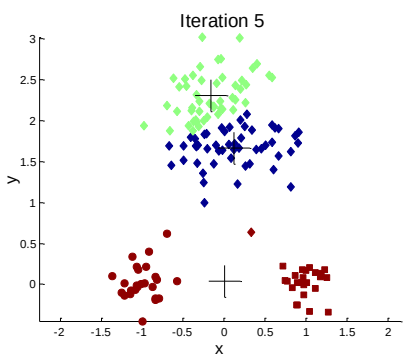
Two different K-means Clusterings



28

28

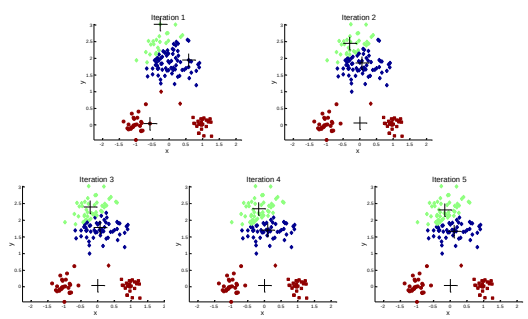
Importance of Choosing Initial Centroids …



29

29

Importance of Choosing Initial Centroids …

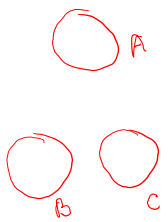


30

30

Importance of Choosing Initial Centroids

- Depending on the choice of initial centroids, B and C may get merged or remain separate



31

31

Problems with Selecting Initial Points

- If there are K 'real' clusters then the chance of selecting one centroid from each cluster is small.
  - Chance is relatively small when K is large
  - If clusters are the same size, n, then

$$P = \frac{\text{number of ways to select one centroid from each cluster}}{\text{number of ways to select } K \text{ centroids}} = \frac{K!n^K}{(Kn)^K} = \frac{K!}{K^K}$$

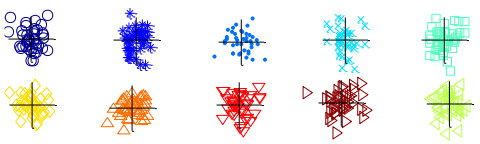
For example, if $K = 10$, then probability $P = 10!/10^{10} = 0.00036$

  - Sometimes the initial centroids will readjust themselves in 'right' way, and sometimes they don't consider an example of five pairs of clusters

32

32

10 Clusters Example

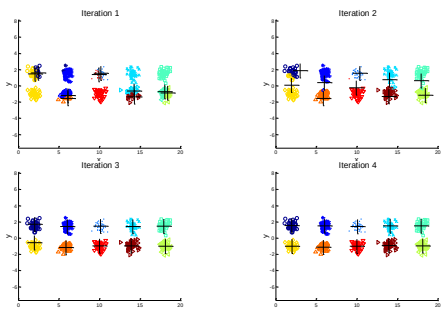


Starting with two initial centroids in one cluster of each pair of clusters

33

33

10 Clusters Example

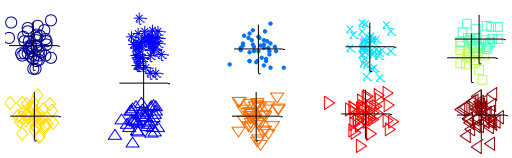


Starting with two initial centroids in one cluster of each pair of clusters

34

34

10 Clusters Example

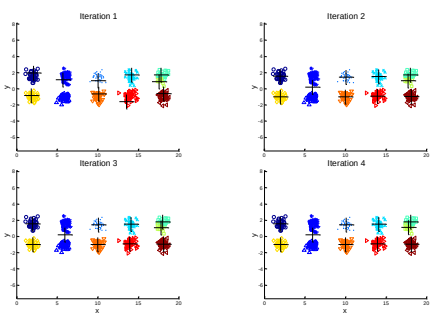


Starting with some pairs of clusters having three initial centroids, while other have only one.

35

35

10 Clusters Example



Starting with some pairs of clusters having three initial centroids, while other have only one.

36

36

Solutions to Initial Centroids Problem

- Multiple runs
  - Helps, but probability is not on your side
- Use some strategy to select the k initial centroids and then select among these initial centroids
  - Select most widely separated
    - K-means++ is a robust way of doing this selection
  - Use hierarchical clustering to determine initial centroids
- Bisecting K-means
  - Not as susceptible to initialization issues

37

37

K-means++

- This approach can be slower than random initialization, but very consistently produces better results in terms of SSE
  - The k-means++ algorithm guarantees an approximation ratio O(log k) in expectation, where k is the number of centers
- To select a set of initial centroids, *C*, perform the following

1. Select an initial point at random to be the first centroid
2. For k – 1 steps
3. For each of the N points, x_i, $1 \leq i \leq N$, find the minimum squared distance to the currently selected centroids, C_1, …, C_j, $1 \leq j < k$, i.e., $\min_j d^2(C_j, x_i)$
4. Randomly select a new centroid by choosing a point with probability proportional to $\frac{\min_j d^2(C_j, x_i)}{\sum_i \min_j d^2(C_j, x_i)}$
5. End For

38

38

Bisecting K-means

- Bisecting K-means algorithm
  - Variant of K-means that can produce a partitional or a hierarchical clustering

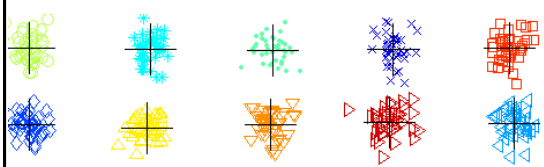
1: Initialize the list of clusters to contain the cluster containing all points.
2: **repeat**
3: Select a cluster from the list of clusters
4: **for** $i = 1$ to *number_of_iterations* **do**
5: Bisect the selected cluster using basic K-means
6: **end for**
7: Add the two clusters from the bisection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

CLUTO: http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview

39

39

Bisecting K-means Example



40

40

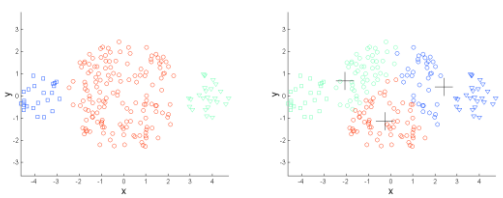
Limitations of K-means

- K-means has problems when clusters are of differing
  - Sizes
  - Densities
  - Non-globular shapes
- K-means has problems when the data contains outliers.
  - One possible solution is to remove outliers before clustering

41

41

Limitations of K-means: Differing Sizes



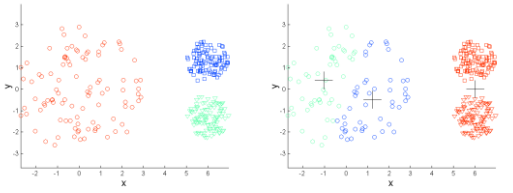
Original Points

K-means (3 Clusters)

42

42

Limitations of K-means: Differing Density

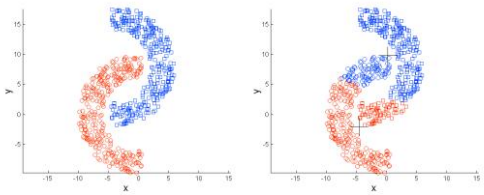


Original Points | K-means (3 Clusters)

43

43

Limitations of K-means: Non-globular Shapes

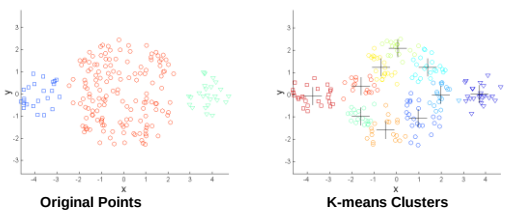


Original Points | K-means (2 Clusters)

44

44

Overcoming K-means Limitations



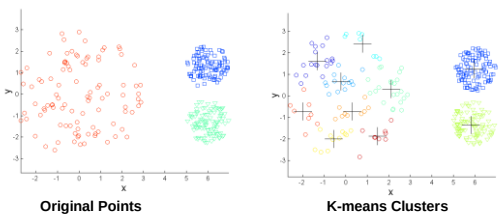
Original Points | K-means Clusters

- One solution is to find a large number of clusters such that each of them represents a part of a natural cluster.
- But these small clusters need to be put together in a post-processing step.

45

45

Overcoming K-means Limitations



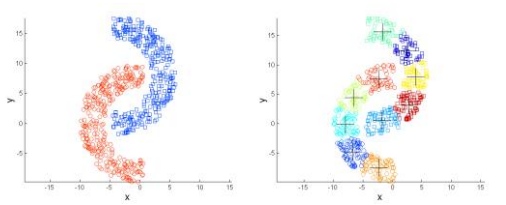
Original Points | K-means Clusters

- One solution is to find a large number of clusters such that each of them represents a part of a natural cluster.
- But these small clusters need to be put together in a post-processing step.

46

46

Overcoming K-means Limitations



Original Points | K-means Clusters

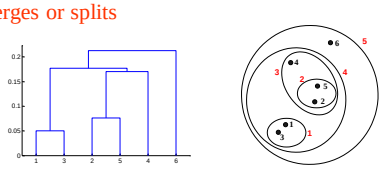
- One solution is to find a large number of clusters such that each of them represents a part of a natural cluster.
- But these small clusters need to be put together in a post-processing step.

47

47

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



48

48

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, …)

49

49

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 an individual point (or there are k clusters)
- Traditional hierarchical algorithms use a similarity or distance matrix
  - Merge or split one cluster at a time

50

50

Agglomerative Clustering Algorithm

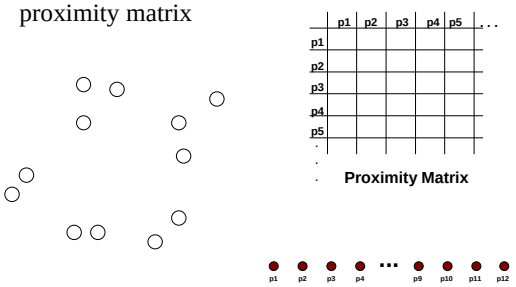
- Key Idea:
  - Successively merge closest clusters
- Basic algorithm
  1. Compute the proximity matrix
  2. Let each data point be a cluster
  3. **Repeat**
  4. Merge the two closest clusters
  5. Update the proximity matrix
  6. **Until** only a single cluster remains
- Key operation is the computation of the proximity of two clusters
  - Different approaches to defining the distance between clusters distinguish the different algorithms

51

51

Steps 1 and 2

- Start with clusters of individual points and a proximity matrix

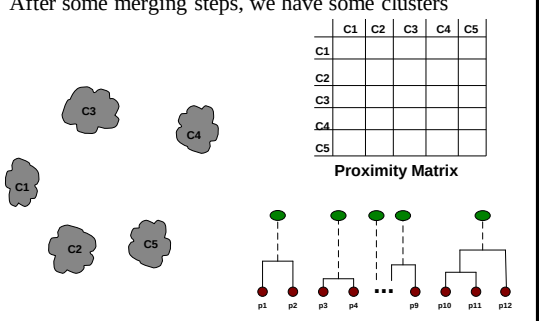


52

52

Intermediate Situation

- After some merging steps, we have some clusters

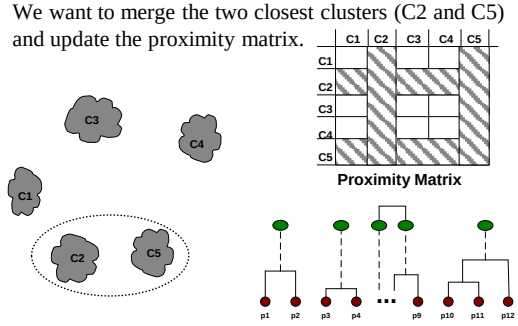


53

53

Step 4

- We want to merge the two closest clusters (C2 and C5) and update the proximity matrix.

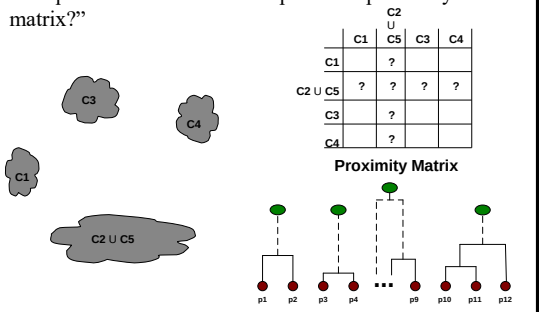


54

54

Step 5

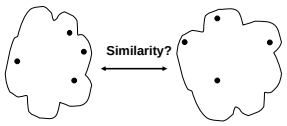
- The question is "How do we update the proximity matrix?"



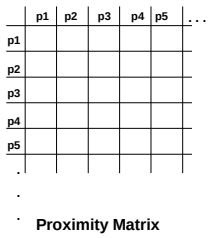
55

55

How to Define Inter-Cluster Distance



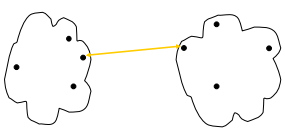
- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error



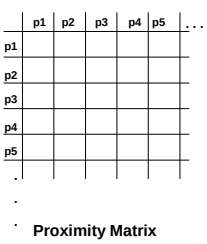
56

56

How to Define Inter-Cluster Similarity



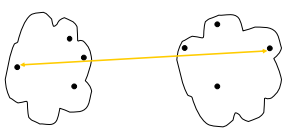
- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error



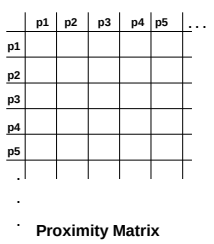
57

57

How to Define Inter-Cluster Similarity



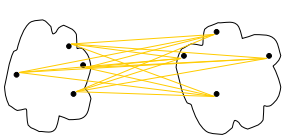
- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error



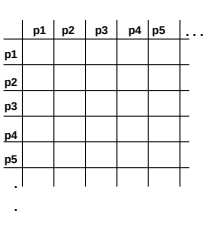
58

58

How to Define Inter-Cluster Similarity



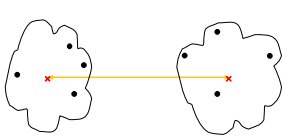
- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error



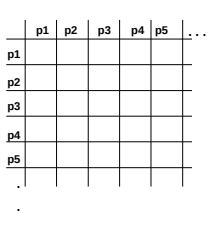
59

59

How to Define Inter-Cluster Similarity



- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error

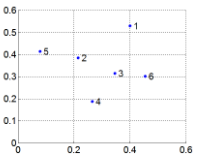


60

60

MIN or Single Link

- Proximity of two clusters is based on the two closest points in the different clusters
  - Determined by one pair of points, i.e., by one link in the proximity graph
- Example:



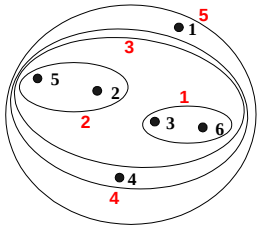
Distance Matrix:

	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

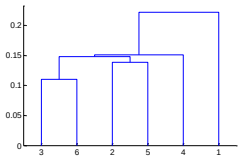
61

61

Hierarchical Clustering: MIN



Nested Clusters



Dendrogram

62

62

Strength of MIN

Original Points

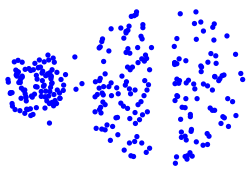
Six Clusters

• Can handle non-elliptical shapes

63

63

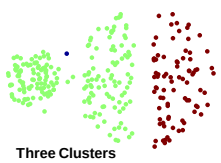
Limitations of MIN



Original Points

• Sensitive to noise

Two Clusters



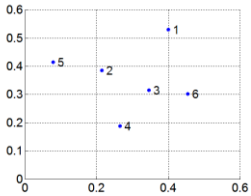
Three Clusters

64

64

MAX or Complete Linkage

- Proximity of two clusters is based on the two most distant points in the different clusters
  - Determined by all pairs of points in the two clusters



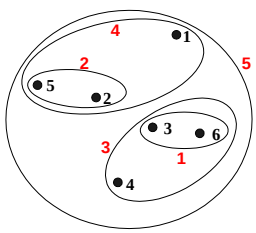
Distance Matrix:

	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

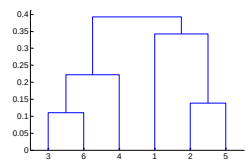
65

65

Hierarchical Clustering: MAX



Nested Clusters

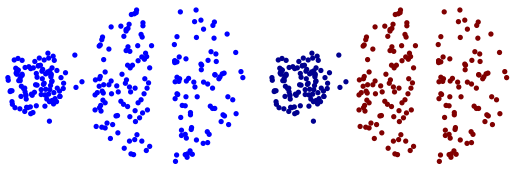


Dendrogram

66

66

Strength of MAX



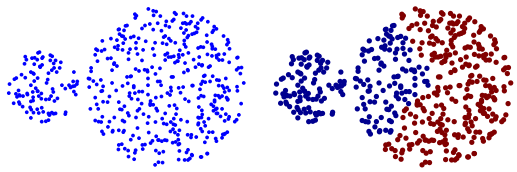
Original Points Two Clusters

- Less susceptible to noise

67

67

Limitations of MAX



Original Points Two Clusters

- Tends to break large clusters
- Biased towards globular clusters

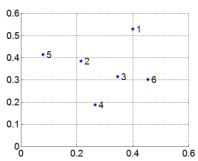
68

68

Group Average

- Proximity of two clusters is the average of pairwise proximity between points in the two clusters.

$$\text{proximity}(\text{Cluster}_i, \text{Cluster}_j) = \frac{\sum_{p_i \in \text{Cluster}_i, p_j \in \text{Cluster}_j} \text{proximity}(p_i, p_j)}{|\text{Cluster}_i| \times |\text{Cluster}_j|}$$



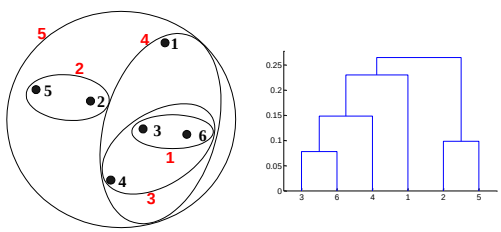
Distance Matrix:

	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

69

69

Hierarchical Clustering: Group Average



Nested Clusters Dendrogram

70

70

Hierarchical Clustering: Group Average

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

71

71

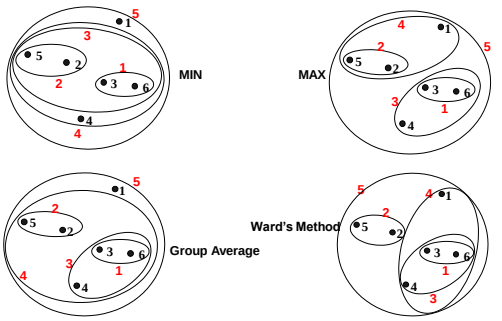
Cluster Similarity: Ward's Method

- Similarity of two clusters is based on the increase in squared error when two clusters are merged
  - Similar to group average if distance between points is distance squared
- Less susceptible to noise
- Biased towards globular clusters
- Hierarchical analogue of K-means
  - Can be used to initialize K-means

72

72

Hierarchical Clustering: Comparison



73

73

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 with some cleverness

74

74

Hierarchical Clustering: Problems and Limitations

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

75

75

Density Based Clustering

- Clusters are regions of high density that are separated from one another by regions on low density.

76

76

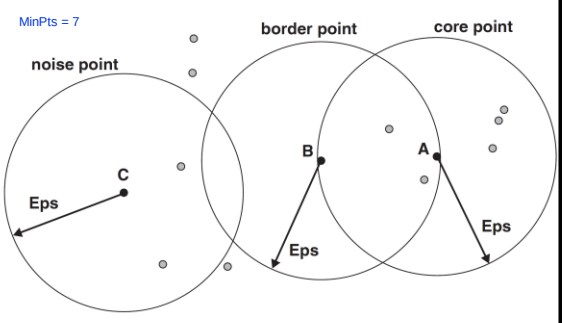
DBSCAN

- DBSCAN is a density-based algorithm.
  - Density = number of points within a specified radius (Eps)
  - A point is a core point if it has at least a specified number of points (MinPts) within Eps
    - These are points that are at the interior of a cluster
    - Counts the point itself
  - A border point is not a core point, but is in the neighborhood of a core point
  - A noise point is any point that is not a core point or a border point

77

77

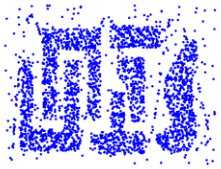
DBSCAN: Core, Border, and Noise Points



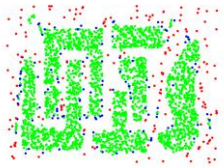
78

78

DBSCAN: Core, Border and Noise Points



Original Points



Point types: core, border and noise

Eps = 10, MinPts = 4

79

79

DBSCAN Algorithm

- Form clusters using core points, and assign border points to one of its neighboring clusters

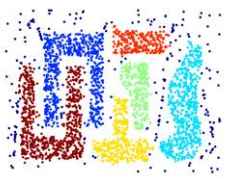
1: Label all points as core, border, or noise points.
2: Eliminate noise points.
3: Put an edge between all core points within a distance *Eps* of each other.
4: Make each group of connected core points into a separate cluster.
5: Assign each border point to one of the clusters of its associated core points

80

80

When DBSCAN Works Well

Original Points



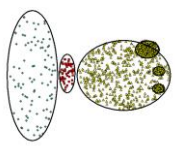
Clusters (dark blue points indicate noise)

- Can handle clusters of different shapes and sizes
- Resistant to noise

81

81

When DBSCAN Does NOT Work Well

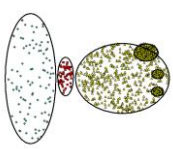


Original Points

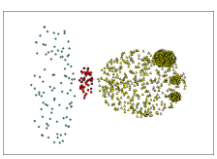
82

82

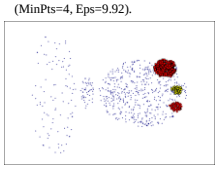
When DBSCAN Does NOT Work Well



Original Points



(MinPts=4, Eps=9.92).



(MinPts=4, Eps=9.75)

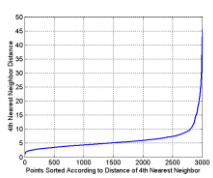
- Varying densities
- High-dimensional data

83

83

DBSCAN: Determining EPS and MinPts

- Idea is that for points in a cluster, their k^{th} nearest neighbors are at close distance
- Noise points have the k^{th} nearest neighbor at farther distance
- So, plot sorted distance of every point to its k^{th} nearest neighbor



84

84

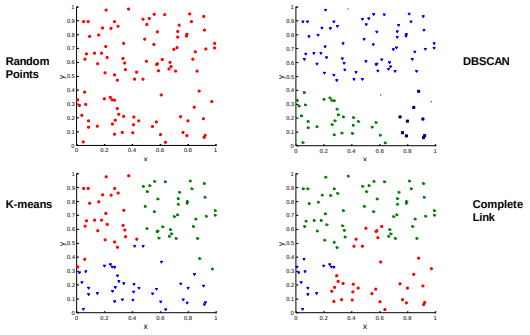
Cluster Validity

- For supervised classification we have a variety of measures to evaluate how good our model is
  - Accuracy, precision, recall
- For cluster analysis, the analogous question is how to evaluate the "goodness" of the resulting clusters?
- But "clusters are in the eye of the beholder"!
  - In practice the clusters we find are defined by the clustering algorithm
- Then why do we want to evaluate them?
  - To avoid finding patterns in noise
  - To compare clustering algorithms
  - To compare two sets of clusters
  - To compare two clusters

85

85

Clusters found in Random Data



86

86

Measures of Cluster Validity

- Numerical measures that are applied to judge various aspects of cluster validity, are classified into the following two types.
  - Supervised: Used to measure the extent to which cluster labels match externally supplied class labels.
    - Entropy
    - Often called *external indices* because they use information external to the data
  - Unsupervised: Used to measure the goodness of a clustering structure *without* respect to external information.
    - Sum of Squared Error (SSE)
    - Often called *internal indices* because they only use information in the data
- You can use supervised or unsupervised measures to compare clusters or clusterings

87

87

Unsupervised Measures: Cohesion and Separation

- Cluster Cohesion:
  - Measures how closely related are objects in a cluster
    - Example: SSE
  - Cohesion is measured by the within cluster sum of squares (SSE)

$$SSE = \sum_i \sum_{x \in C_i} (x - m_i)^2$$

- Cluster Separation:
  - Measures how distinct or well-separated a cluster is from other clusters
    - Example: Squared Error
  - Separation is measured by the between cluster sum of squares

$$SSB = \sum_i |C_i|(m - m_i)^2$$

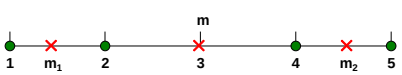
where $|C_i|$ is the size of cluster i

88

88

Unsupervised Measures: Cohesion and Separation

- Example: SSE
  - SSB + SSE = constant



K=1 cluster:

$$SSE = (1-3)^2 + (2-3)^2 + (4-3)^2 + (5-3)^2 = 10$$
$$SSB = 4 \times (3-3)^2 = 0$$
$$Total = 10 + 0 = 10$$

K=2 clusters:

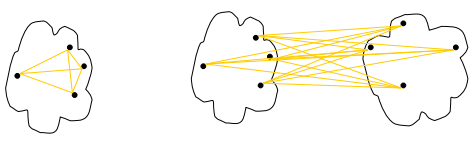
$$SSE = (1-1.5)^2 + (2-1.5)^2 + (4-4.5)^2 + (5-4.5)^2 = 1$$
$$SSB = 2 \times (3-1.5)^2 + 2 \times (4.5-3)^2 = 9$$
$$Total = 1 + 9 = 10$$

89

89

Unsupervised Measures: Cohesion and Separation

- 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.



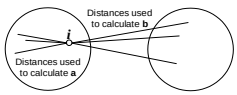
90

90

Unsupervised Measures: Silhouette Coefficient

- Silhouette coefficient combines ideas of both cohesion and separation, but for individual points, as well as clusters and clusterings
- For an individual point, ***i***
  - Calculate ***a*** = average distance of ***i*** to the points in its cluster
  - Calculate ***b*** = min (average distance of ***i*** to points in another cluster)
  - The silhouette coefficient for a point is then given by

    s = (b – a) / max(a,b)



  - Value can vary between -1 and 1
  - Typically ranges between 0 and 1.
  - The closer to 1 the better.
- Can calculate the average silhouette coefficient for a cluster or a clustering

91

91

Measuring Cluster Validity Via Correlation

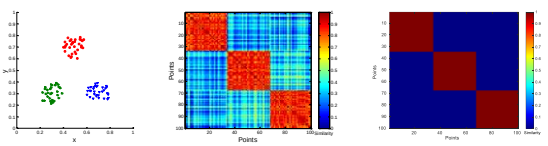
- Two matrices
  - Proximity Matrix
  - Ideal Similarity Matrix
    - One row and one column for each data point
    - An entry is 1 if the associated pair of points belong to the same cluster
    - An entry is 0 if the associated pair of points belongs to different clusters
- Compute the correlation between the two matrices
  - Since the matrices are symmetric, only the correlation between n(n-1) / 2 entries needs to be calculated.
- High magnitude of correlation indicates that points that belong to the same cluster are close to each other.
  - Correlation may be positive or negative depending on whether the similarity matrix is a similarity or dissimilarity matrix
- Not a good measure for some density or contiguity-based clusters.

92

92

Measuring Cluster Validity Via Correlation

- Correlation of ideal similarity and proximity matrices for the K-means clusterings of the following well-clustered data set.



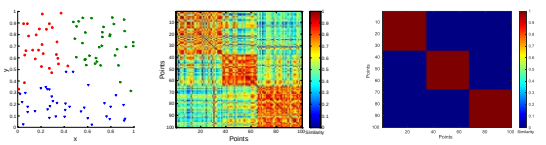
Corr = 0.9235

93

93

Measuring Cluster Validity Via Correlation

- Correlation of ideal similarity and proximity matrices for the K-means clusterings of the following random data set.



K-means

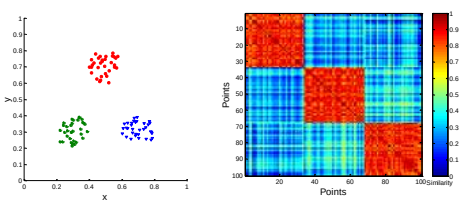
Corr = 0.5810

94

94

Judging a Clustering Visually by its Similarity Matrix

- Order the similarity matrix with respect to cluster labels and inspect visually.

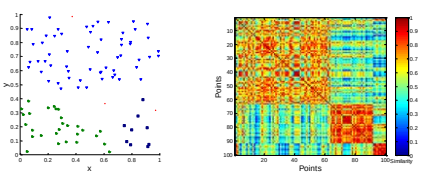


95

95

Judging a Clustering Visually by its Similarity Matrix

- Clusters in random data are not so crisp

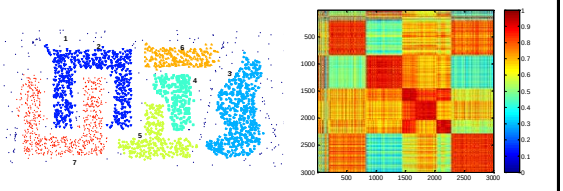


DBSCAN

96

96

Judging a Clustering Visually by its Similarity Matrix



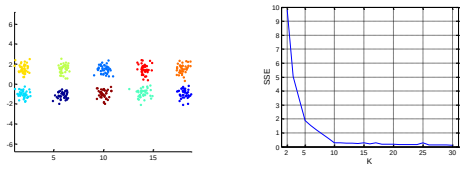
DBSCAN

97

97

Determining the Correct Number of Clusters

- SSE is good for comparing two clusterings or two clusters
- SSE can also be used to estimate the number of clusters

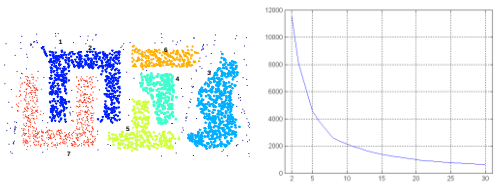


98

98

Determining the Correct Number of Clusters

- SSE curve for a more complicated data set



SSE of clusters found using K-means

99

99

Supervised 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., $e = \sum_{i=1}^{K} \frac{m_j}{m} e_j$, 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 $purity = \sum_{i=1}^{K} \frac{m_i}{m} purity_j$.

100

100

Assessing the Significance of Cluster Validity Measures

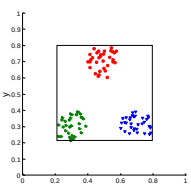
- Need a framework to interpret any measure.
  - For example, if our measure of evaluation has the value, 10, is that good, fair, or poor?
- Statistics provide a framework for cluster validity
  - The more "atypical" a clustering result is, the more likely it represents valid structure in the data
  - Compare the value of an index obtained from the given data with those resulting from random data.
    - If the value of the index is unlikely, then the cluster results are valid

101

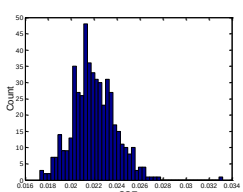
101

Statistical Framework for SSE

- Example
  - Compare SSE of three cohesive clusters against three clusters in random data



SSE = 0.005



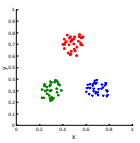
Histogram shows SSE of three clusters in 500 sets of random data points of size 100 distributed over the range 0.2 – 0.8 for x and y values

102

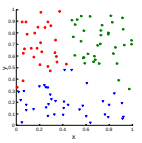
102

Statistical Framework for Correlation

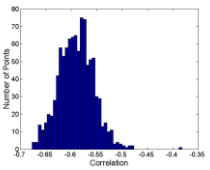
- Correlation of ideal similarity and proximity matrices for the K-means clusterings of the following two data sets.



Corr = -0.9235



Corr = -0.5810



Correlation is negative because it is calculated between a distance matrix and the ideal similarity matrix. Higher magnitude is better.

Histogram of correlation for 500 random data sets of size 100 with *x* and *y* values of points between 0.2 and 0.8.

103

103

Final Comment on Cluster Validity

- "The validation of clustering structures is the most difficult and frustrating part of cluster analysis.
- Without a strong effort in this direction, cluster analysis will remain a black art accessible only to those true believers who have experience and great courage."

***Algorithms for Clustering Data*, Jain and Dubes**

- H. Xiong and Z. Li. *Clustering Validation Measures*. In C. C. Aggarwal and C. K. Reddy, editors, Data Clustering: Algorithms and Applications, pages 571–605. Chapman & Hall/CRC, 2013.

104

104

105

105