R·I·T

Clustering

DHS 10.6-10.7, 10.9-10.10, 10.4.3-10.4.4

Clustering

Definition

A form of unsupervised learning, where we identify groups in feature space for an unlabeled sample set

- *Define* class regions in feature space using unlabeled data
- *Note:* the classes identified are abstract, in the sense that we obtain 'cluster 0' ... 'cluster n' as our classes (e.g. clustering MNIST digits, we may not get 10 clusters)

Applications

Clustering Applications Include:

- Data reduction: represent samples by their associated cluster
- Hypothesis generation
  - Discover possible patterns in the data: validate on *other* data sets
- Hypothesis testing
  - Test assumed patterns in data
- Prediction based on groups
  - e.g. selecting medication for a patient using clusters of previous patients and their reactions to medication for a given disease

Kuncheva:
Supervised vs. Unsupervised Classification

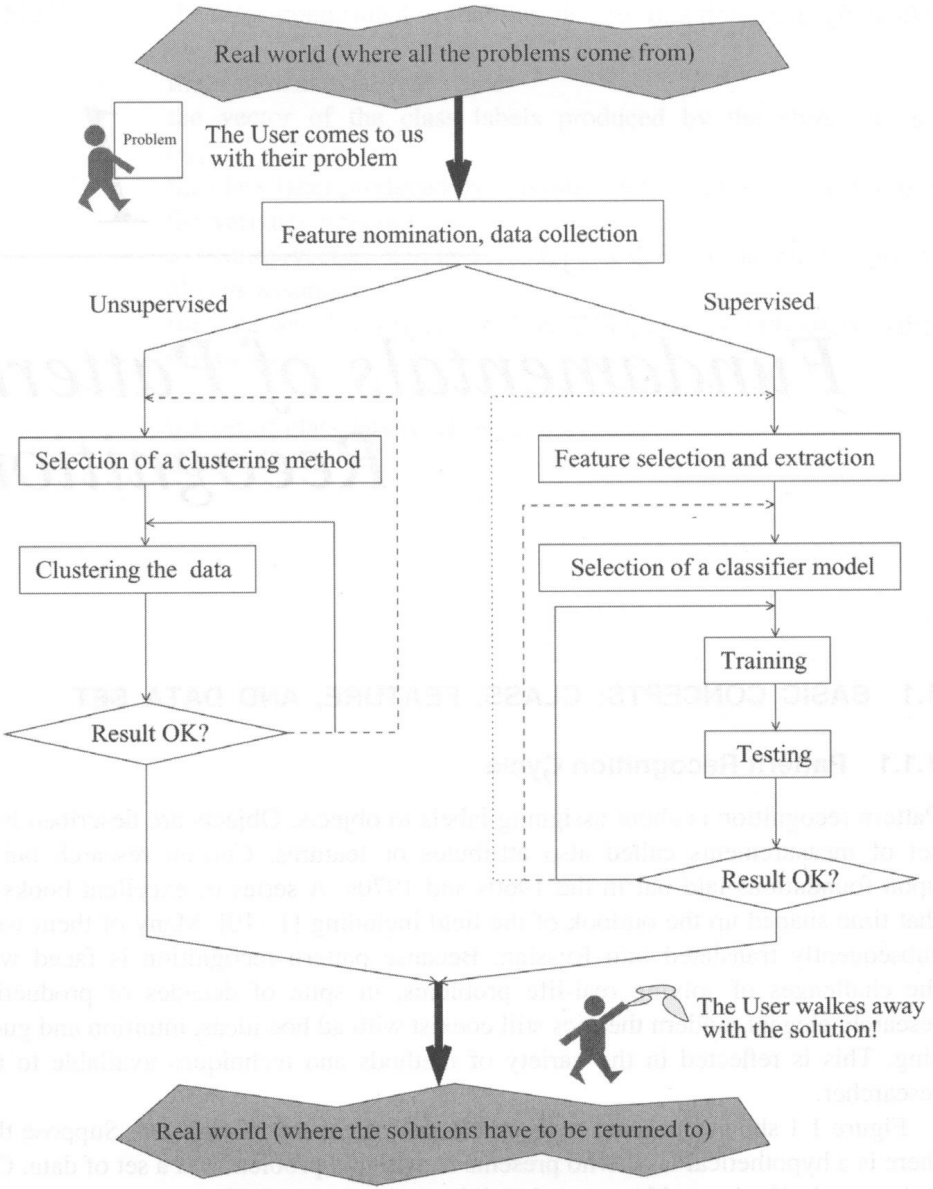


Fig. 1.1 *The pattern recognition cycle.*

A Simple Example

Assume Class Distributions Known to be Normal

Can define clusters by mean and covariance matrix

However...

We may need more information to cluster well

- Many different distributions can share a mean and covariance matrix
- ....*number* of clusters?

R·I·T

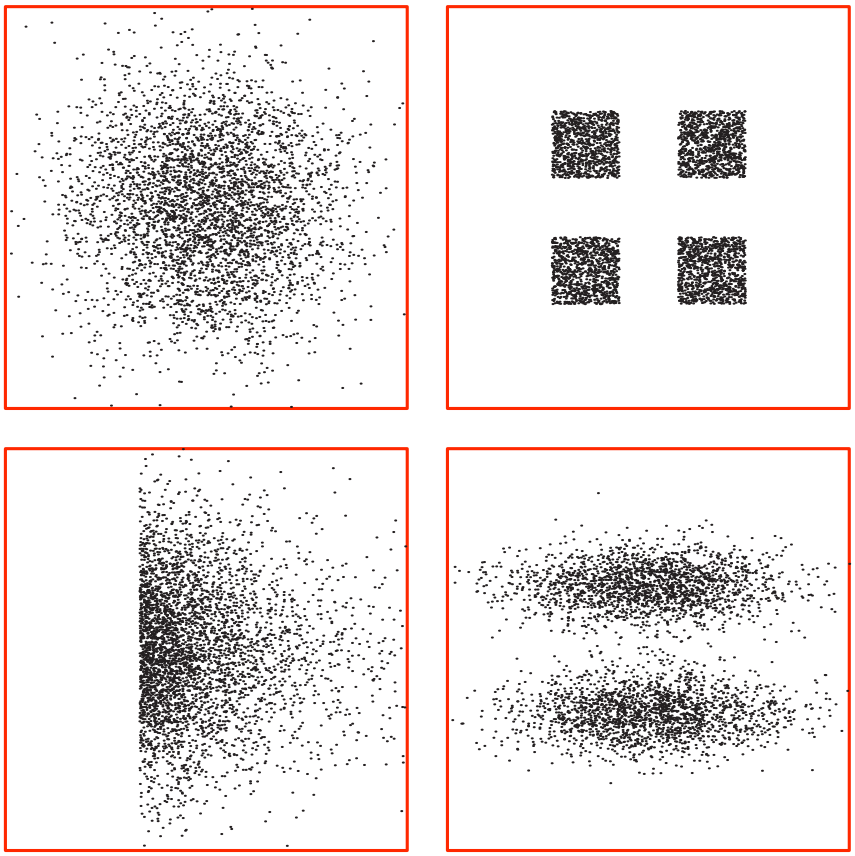


FIGURE 10.6. These four data sets have identical statistics up to second-order—that is, the same mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. In such cases it is important to include in the model more parameters to represent the structure more completely. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Steps for Clustering

1. Feature Selection

- Ideal: small number of features with little redundancy

2. Similarity (or Proximity) Measure

- Measure of similarity or *dissimilarity*

Red: defining 'cluster space'

3. Clustering Criterion

- Determine how distance patterns determine cluster likelihood (e.g. preferring circular to elongated clusters)

4. Clustering Algorithm

- Search method used with the clustering criterion to identify clusters

5. Validation of Results

- Using appropriate tests (e.g. statistical)

6. Interpretation of Results

- Domain expert interprets clusters (clusters are *subjective*)

Choosing a Similarity Measure

Most Common: Euclidean Distance

Roughly speaking, want distance between samples in a cluster to be smaller than the distance between samples in different clusters

- Example (next slide): define clusters by a maximum distance d_0 between a point and a point in a cluster
- Rescaling features can be useful (transform the space)
  - Unfortunately, normalizing data (e.g. by setting features to zero mean, unit variance) may eliminate subclasses
  - One might also choose to rotate axes so they coincide with eigenvectors of the covariance matrix (i.e. apply PCA)

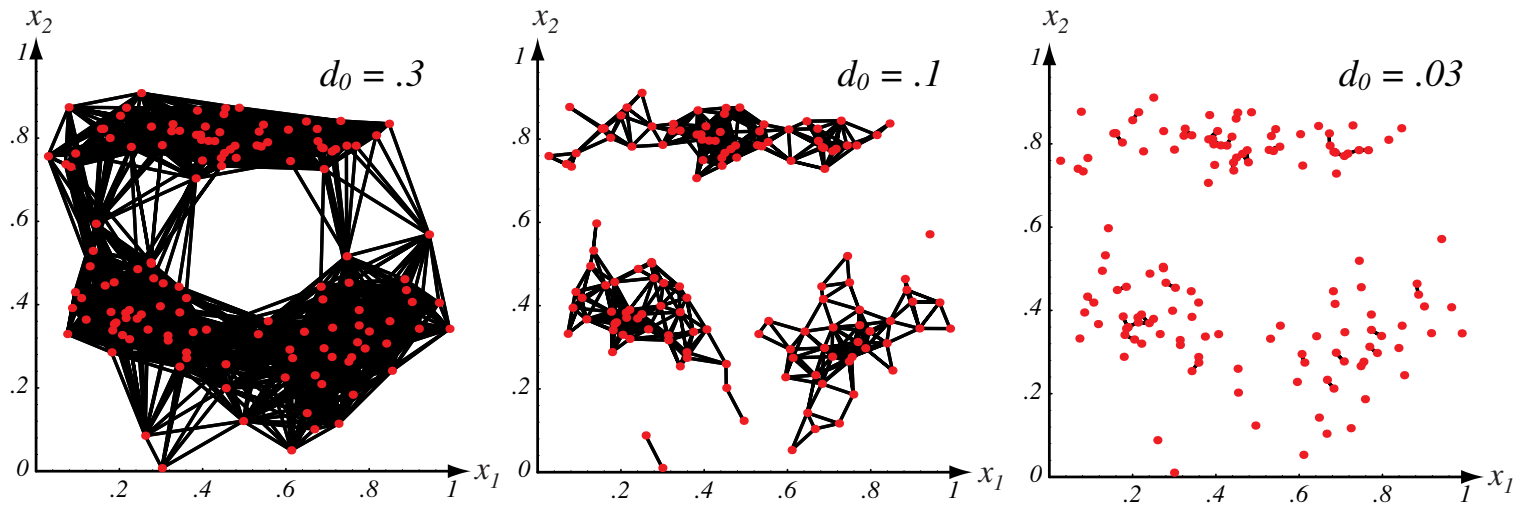


FIGURE 10.7. The distance threshold affects the number and size of clusters in similarity based clustering methods. For three different values of distance d_0, lines are drawn between points closer than d_0—the smaller the value of d_0, the smaller and more numerous the clusters. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

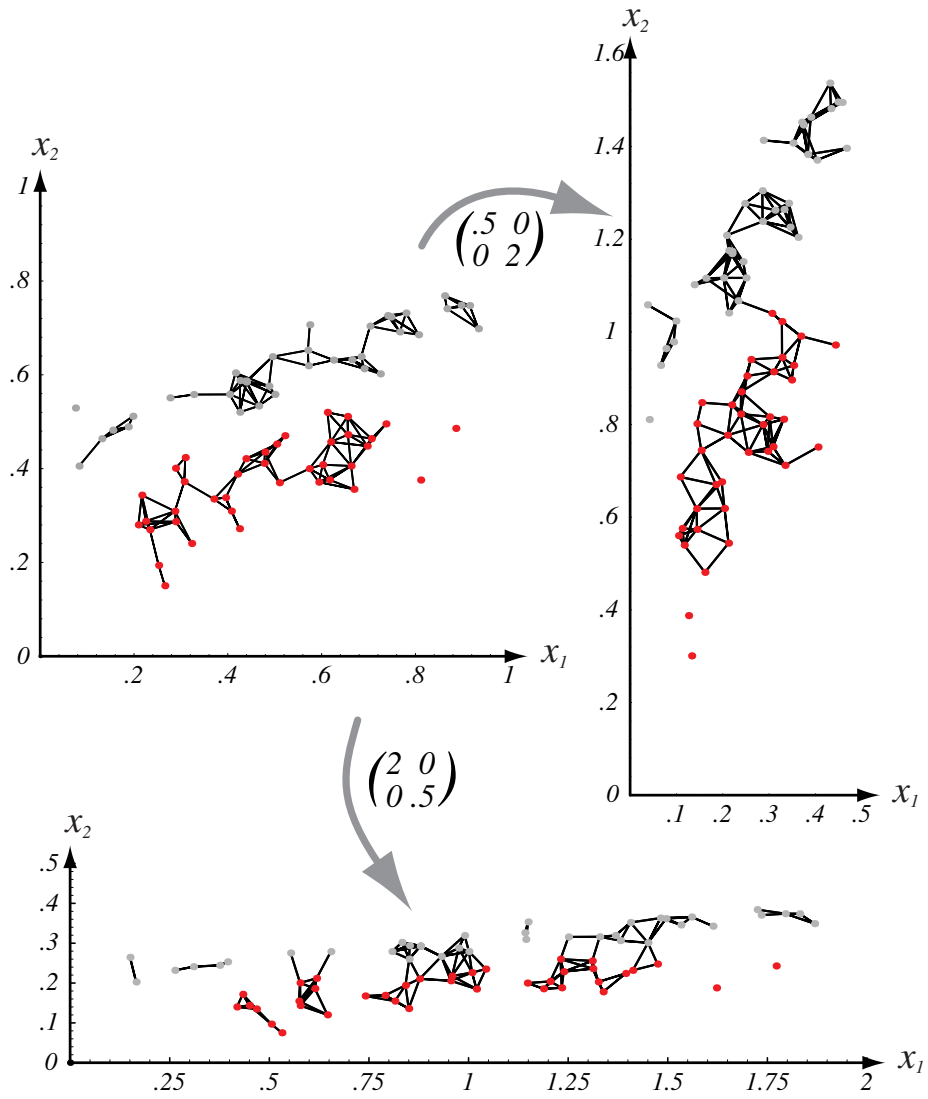


FIGURE 10.8. Scaling axes affects the clusters in a minimum distance cluster method. The original data and minimum-distance clusters are shown in the upper left; points in one cluster are shown in red, while the others are shown in gray. When the vertical axis is expanded by a factor of 2.0 and the horizontal axis shrunk by a factor of 0.5, the clustering is altered (as shown at the right). Alternatively, if the vertical axis is shrunk by

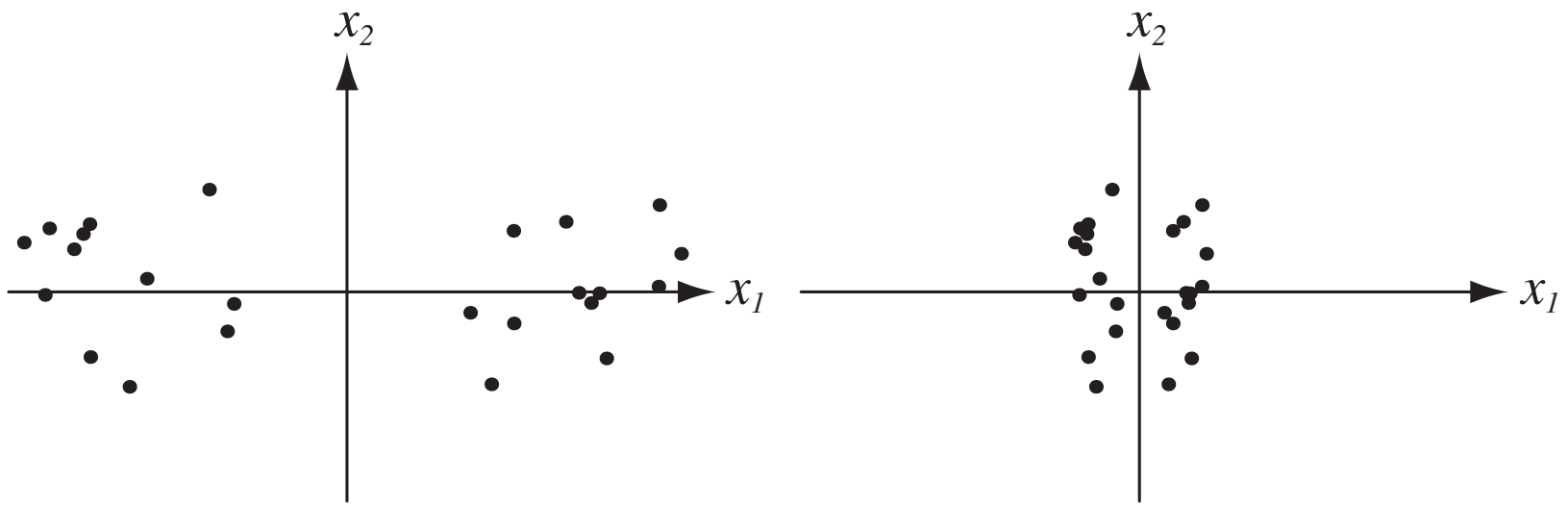


FIGURE 10.9. If the data fall into well-separated clusters (left), normalization by scaling for unit variance for the full data may reduce the separation, and hence be undesirable (right). Such a normalization may in fact be appropriate if the full data set arises from a single fundamental process (with noise), but inappropriate if there are several different processes, as shown here. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Other Similarity Measures

Minkowski Metric (*Dissimilarity*)

Change the exponent q:

$$d(\mathbf{x}, \mathbf{x}') = \left(\sum_{k=1}^{d} |x_k - x'_k|^q \right)^{1/q}$$

- q = 1: Manhattan (city-block) distance
- q = 2: Euclidean distance (only form invariant to translation and rotation in feature space)

Cosine *Similarity*

$$s(\mathbf{x}, \mathbf{x}') = \frac{\mathbf{x}^{\mathbf{T}} \mathbf{x}'}{||\mathbf{x}||\ ||\mathbf{x}'||}$$

Characterizes similarity by the cosine of the angle between two feature vectors (in [0,1])

- Ratio of inner product to vector magnitude product
- Invariant to rotations and dilation (*not* translation)

More on Cosine Similarity

If features binary-valued: $s(\mathbf{x}, \mathbf{x}') = \frac{\mathbf{x}^{\mathbf{T}}\mathbf{x}'}{||\mathbf{x}||\ ||\mathbf{x}'||}$

- Inner product is sum of shared feature values
- Product of magnitudes is geometric mean of number of attributes in the two vectors

Variations

Frequently used for Information Retrieval

- Ratio of shared attributes (identical lengths): $s(\mathbf{x}, \mathbf{x}') = \frac{\mathbf{x}^{\mathbf{T}}\mathbf{x}'}{d}$
- Tanimoto distance: ratio of shared attributes to attributes in x or x'

$$s(\mathbf{x}, \mathbf{x}') = \frac{\mathbf{x}^{\mathbf{T}}\mathbf{x}'}{\mathbf{x}^{\mathbf{T}}\mathbf{x} + \mathbf{x}'^{\mathbf{T}}\mathbf{x}' - \mathbf{x}^{\mathbf{T}}\mathbf{x}'}$$

Cosine Similarity: Tag Sets for YouTube Videos (Example by K. Kluever)

Let A and B be binary vectors of the same length (represent all tags in A&B)

Tag Set	Occ. Vector	dog	puppy	funny	cat
A_t	A	1	1	1	0
R_t	B	1	1	0	1

$$\text{SIM}(A, B) = \cos\theta = \frac{A \cdot B}{\|A\|\|B\|} \qquad \cos\theta = \frac{|A_t \cap R_t|}{\sqrt{|A_t|}\sqrt{|R_t|}}$$

Here SIM(A, B) is 2/3.

Additional Similarity Metrics

Theodoridis Text

Defines a *large* number of alternative distance metrics, including:

- Hamming distance: number of locations where two vectors (usually bit vectors) disagree
- Correlation coefficient
- Weighted distances...

Criterion Functions for Clustering

Criterion Function

Quantifies 'quality' of a set of clusters

- Clustering task: partition data set *D* into *c* disjoint sets D_1 ... D_c
- Choose partition maximizing the criterion function

Criterion: Sum of Squared Error

$$J_e = \sum_{i=1}^{c} \sum_{\mathbf{x} \in D_i} ||\mathbf{x} - \mu_{D_i}||^2$$

Measures total squared 'error' incurred by choice of cluster centers (cluster means)

'Optimal' Clustering

Minimizes this quantity

Issues

- Well suited when clusters compact and well-separated
- Different # points in each cluster can lead to large clusters being split 'unnaturally' (next slide)
- Sensitive to outliers

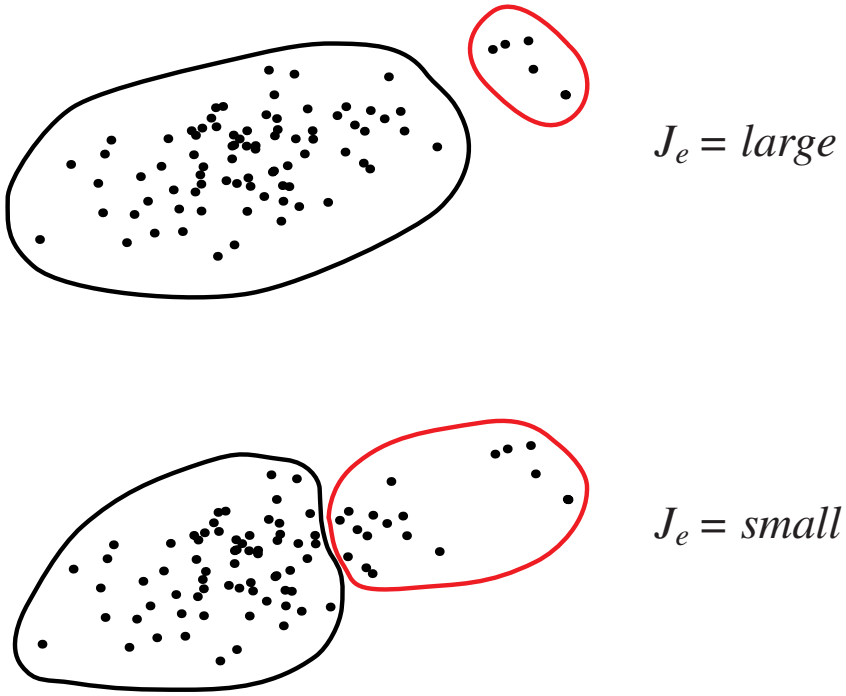


FIGURE 10.10. When two natural groupings have very different numbers of points, the clusters minimizing a sum-squared-error criterion J_e of Eq. 54 may not reveal the true underlying structure. Here the criterion is smaller for the two clusters at the bottom than for the more natural clustering at the top. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Related Criteria: Min Variance

$$J_e = \frac{1}{2}\sum_{i=1}^{c} n_i \bar{s}_i \qquad \bar{s}_i = \frac{1}{n_i^2}\sum_{\mathbf{x}\in D_i}\sum_{\mathbf{x}'\in D_i} ||\mathbf{x}-\mathbf{x}'||^2$$

An Equivalent Formulation for SSE

$\bar{s}_i$: mean squared distance between points in cluster i (variance)

- Alternative Criterions: use median, maximum, other descriptive statistic on distance for $\bar{s}_i$

Variation: Using Similarity (e.g. Tanimoto)

s may be any similarity function (in this case, *maximize)*

$$\bar{s}_i = \frac{1}{n_i^2}\sum_{\mathbf{x}\in D_i}\sum_{\mathbf{x}'\in D_i} s(\mathbf{x},\mathbf{x}') \qquad \bar{s}_i = \min_{\mathbf{x},\mathbf{x}'\in D_i} s(x,x')$$

R·I·T

Criterion: Scatter Matrix-Based

$$trace[S_w] = \sum_{i=1}^{c} \sum_{\mathbf{x} \in D_i} ||\mathbf{x} - \mu_\mathbf{i}||^\mathbf{2} = \mathbf{J_e}$$

$$S_w = \sum_{i=1}^{c} \sum_{\mathbf{x} \in D_i} (\mathbf{x} - \mu_\mathbf{i})(\mathbf{x} - \mu_\mathbf{i})^T$$

Minimize Trace of S_w (within-class)

Equivalent to SSE!

Recall that total scatter is the sum of within and between-class scatter (Sm = Sw + Sb). This means that by minimizing the trace of Sw, we also maximize Sb (*as Sm is fixed*):

$$trace[S_b] = \sum_{i=1}^{c} n_i ||\mu_i - \mu_0||^2$$

Scatter-Based Criterions, Cont'd

$$J_d = |S_w| = \left| \sum_{i=1}^{c} \sum_{\mathbf{x} \in D_i} (\mathbf{x} - \mu_\mathbf{i})(\mathbf{x} - \mu_\mathbf{i})^T \right|$$

Determinant Criterion

Roughly measures square of the scattering volume; proportional to product of variances in principal axes (*minimize!*)

- Minimum error partition will not change with axis scaling, unlike SSE

Scatter-Based: Invariant Criteria

Invariant Criteria (Eigenvalue-based)

Eigenvalues: measure ratio of between to within-cluster scatter in direction of eigenvectors (*maximize!)*

- Trace of a matrix is sum of eigenvalues (here *d* is length of feature vector)
- Eigenvalues are *invariant* under non-singular linear transformations (rotations, translations, scaling, etc.)

$$trace[S_w^{-1} S_b] = \sum_{i=1}^{d} \lambda_i$$

$$J_f = trace[S_m^{-1} S_w] = \sum_{i=1}^{d} \frac{1}{1+\lambda_i}$$

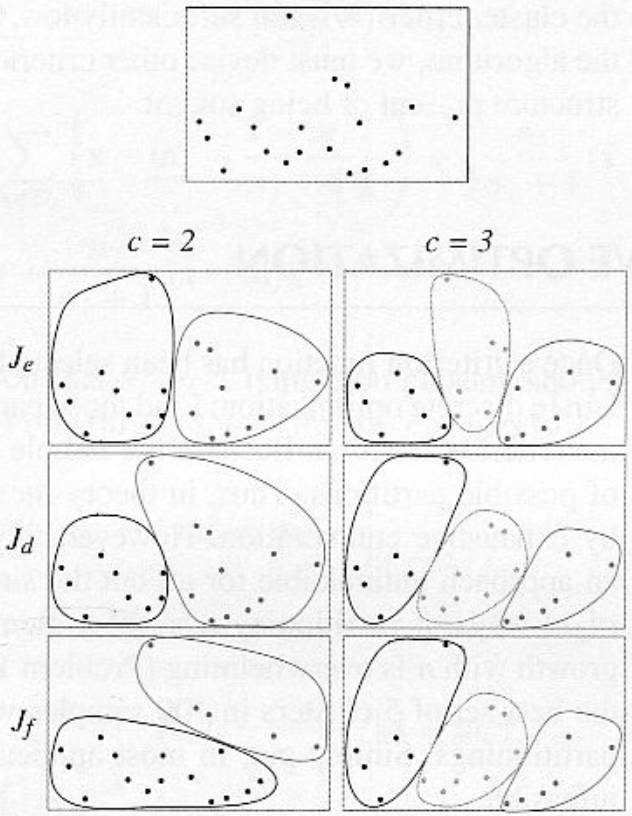
Clustering with a Criterion

Choosing Criterion

Creates a well-defined problem

- Define clusters so as to maximize the criterion function
- A search problem
  - Brute force solution: enumerate partitions of the training set, select the partition with maximum criterion value

Comparison: Scatter-Based Criteria



The raw data shown at the top does not exhibit any obvious clusters. The clusters found by minimizing a criterion depends upon the criterion function as well as the assumed number of clusters. The sum-of-squared-error criterion J_e (Eq. 54), the determinant criterion J_d (Eq. 68) and the more subtle trace criterion J_f (Eq. 70) were applied to the 20 points in the table with the assumption of $c = 2$ and $c = 3$ clusters. (Each point in the table is shown, with bounding boxes defined by $-1.8 < x_1 < 2.5$ and $-0.6 < x_2 < 1.9$.)

R·I·T