CSE 332
Introduction to Visualization

Cluster Analysis

Klaus Mueller

Computer Science Department
Stony Brook University

Lecture	Topic	Projects
1	Intro, schedule, and logistics	
2	Applications of visual analytics, data, and basic tasks	
3	Data preparation and reduction	Project 1 out
4	Data preparation and reduction	
5	Data reduction and similarity metrics	
6	Dimension reduction	
7	Introduction to D3	Project 2 out
8	Bias in visualization	
9	Perception and cognition	
10	Visual design and aesthetics	
11	Cluster and pattern analysis	
12	High-Dimensional data visualization: linear methods	
13	High-D data vis.: non-linear methods, categorical data	Project 3 out
14	Computer graphics and volume rendering	
15	Techniques to visualize spatial (3D) data	
16	Scientific and medical visualization	
17	Scientific and medical visualization	
18	Non-photorealistic rendering	Project 4 out
19	Midterm	
20	Principles of interaction	
21	Visual analytics and the visual sense making process	
22	Visualization of graphs and hierarchies	
23	Visualization of text data	Project 5 out
24	Visualization of time-varying and time-series data	
25	Memorable visualizations, visual embellishments	
26	Evaluation and user studies	
27	Narrative visualization and storytelling	
28	Data journalism	

Finding The Needle – Cluster Analysis

Data summarization

- data reduction
- cluster centers, shapes, and statistics

Customer segmentation

- collaborative filtering

Social network analysis

- find similar groups of friends (communities)

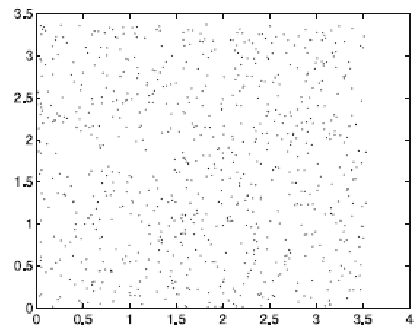
Precursor to other analysis

- use as a preprocessing step for classification and outlier detection

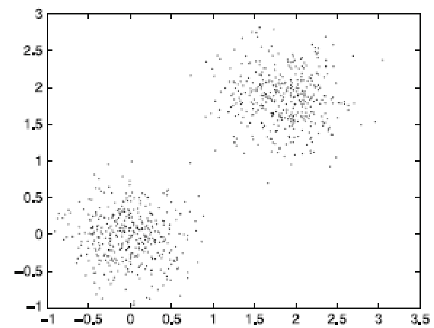
Attribute Selection

With 1,000s of attributes (dimensions) which ones are relevant and which one are not?

avoid



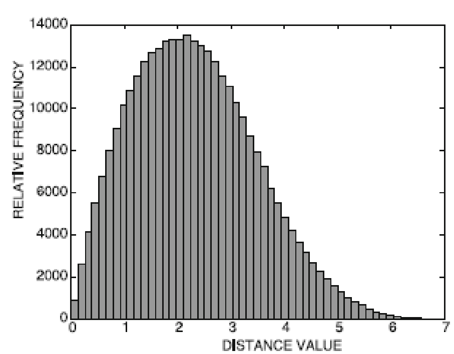
(a) Uniform Data



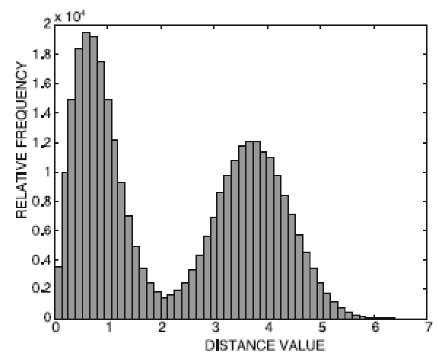
(b) Clustered data

keep

histogram of
pairwise distances
in N-D space



(c) Distance distribution (uniform)

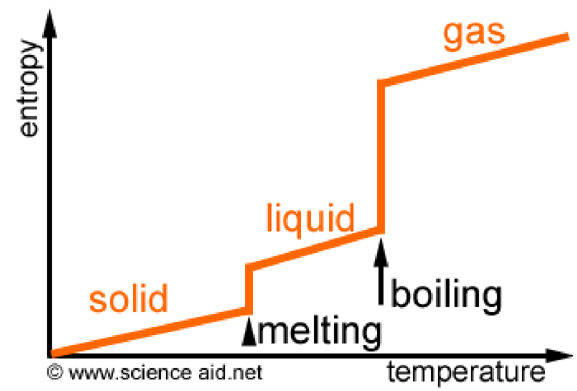


(d) Distance distribution (clustered)

Attribute Selection

How to measure attribute "worthiness"

- use entropy



Entropy

- originates in thermodynamics
- measures lack of order or predictability

Entropy in statistics and information theory

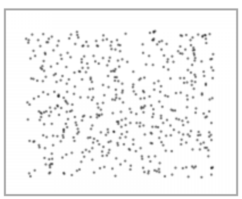
- has a value of 1 for uniform distributions (not predictable)
- knowing the value has a lot of information (high surprise)
- a value of 0 for a constant value (fully predicable)
- knowing the value has zero information (low surprise)

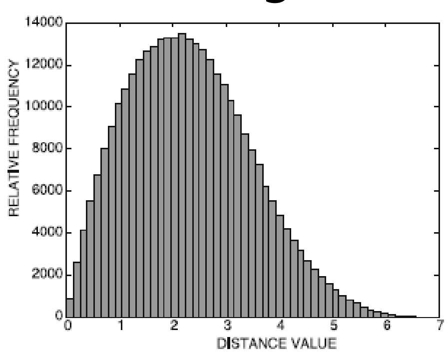
Entropy

Assume m bins, $1 \le i \le m$: $$E = -\sum_{i=1}^{m}[p_i \log(p_i) + (1-p_i)\log(1-p_i)].$$

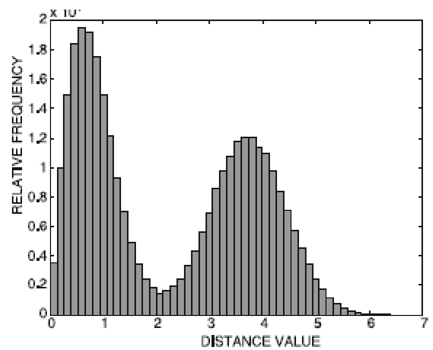
E high

E low

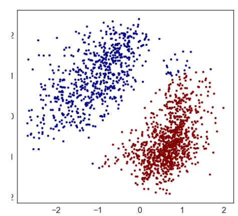


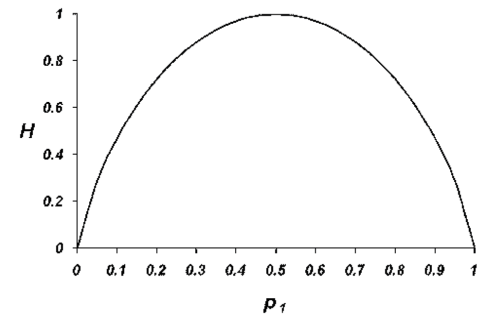


(c) Distance distribution (uniform)



(d) Distance distribution (clustered)



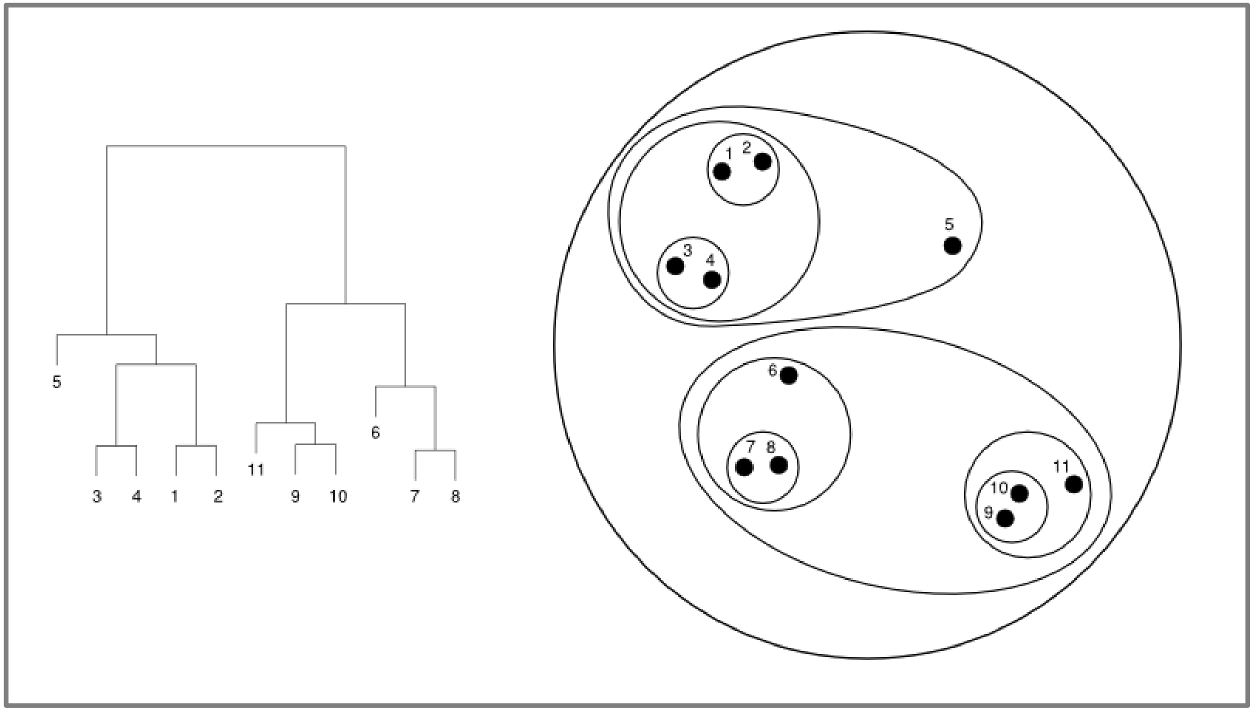


Binary source (e.g. coin)

Algorithm:

- start with all attributes and compute distance entropy
- greedily eliminate attributes that reduce the entropy the most
- stop when entropy no longer reduces or even increases

Hierarchical Clustering

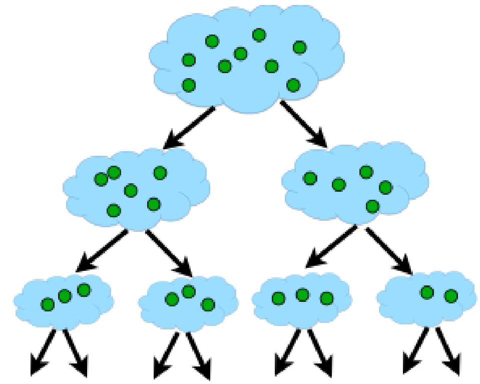


Two options:

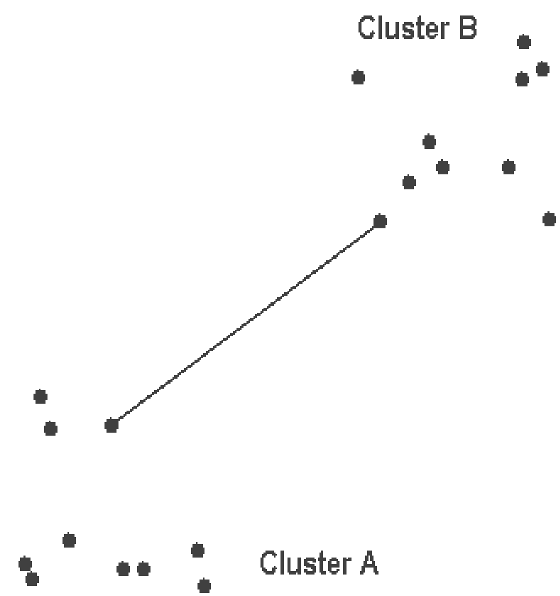
- top down (divisive)
- bottom up (agglomerative)

Bottom-Up Agglomerative Methods

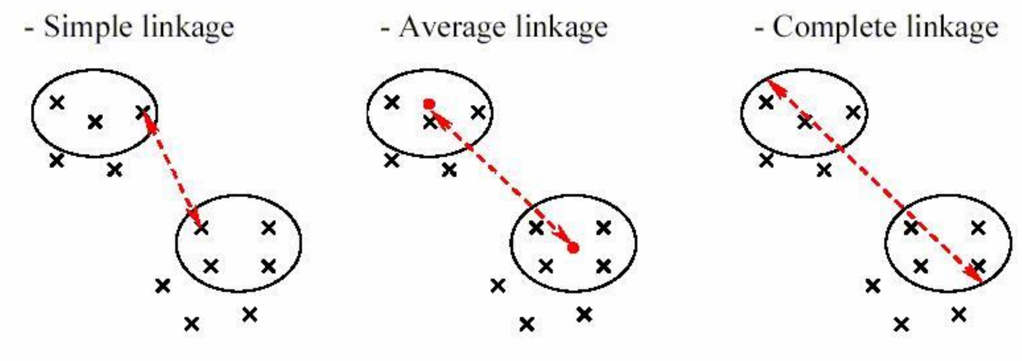
Algorithm *AgglomerativeMerge*(Data: $\mathcal{D}$)
begin
  Initialize $n \times n$ distance matrix M using $\mathcal{D}$;
  repeat
    Pick closest pair of clusters i and j using M;
    Merge clusters i and j;
    Delete rows/columns i and j from M and create
      a new row and column for newly merged cluster;
    Update the entries of new row and column of M;
  until termination criterion;
  return current merged cluster set;
end



How to merge?



Merge Criteria



Single linkage

- distance = minimum distance between all $m_i \cdot m_j$ pairs of objects
- joins the closest pair

Worst (complete) linkage

- distance = maximum distance between all $m_i \cdot m_j$ pairs of objects
- joins the pair furthest apart

Group-average linkage

- distance = average distance between all object pairs in the groups

Other methods:

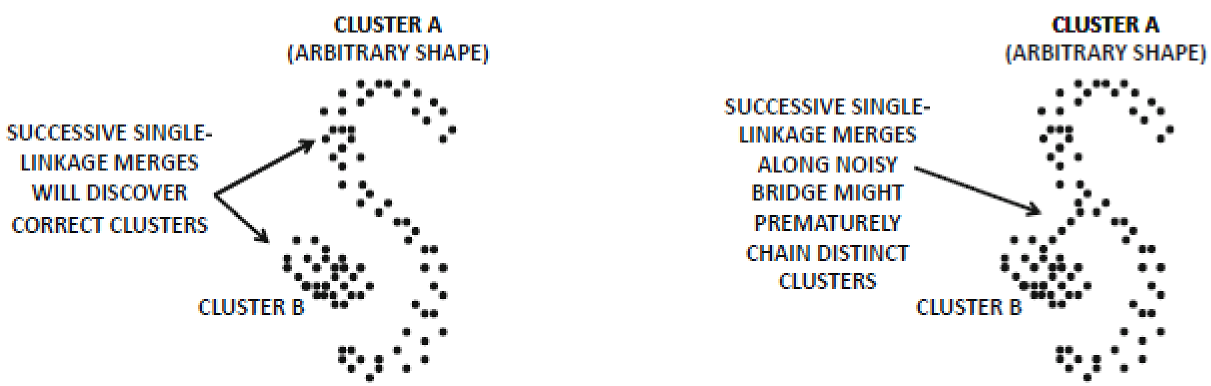
- closest centroid, variance-minimization, Ward's method

Comparison

Centroid-based methods tend to merge large clusters

Single linkage method can merge chains of closely related points to discover clusters of arbitrary shape

- but can also (inappropriately) merge two unrelated clusters, when the chaining is caused by noisy points between two clusters

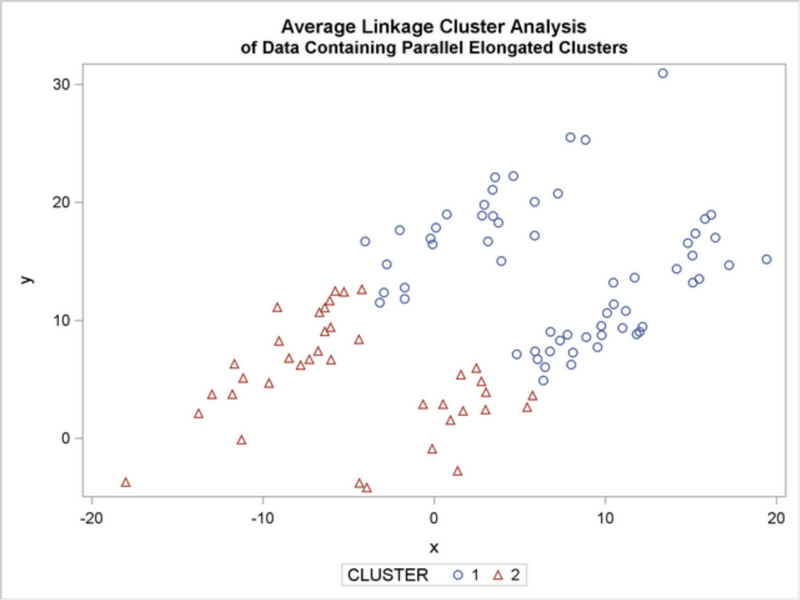


(a) Good case with no noise (b) Bad case with noise

Comparison

Complete (worst-case) linkage method tends to create spherical clusters with similar diameter

- will break up the larger odd-shaped clusters into smaller spheres
- also gives too much importance to data points at the noisy fringes of a cluster

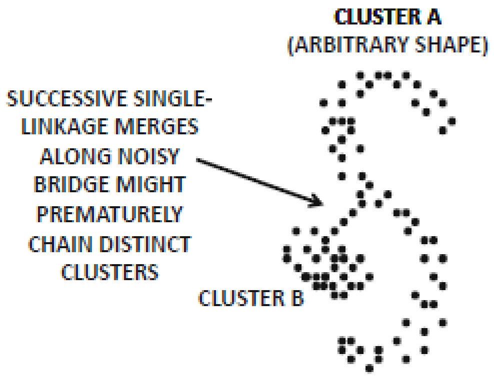


Comparison

The group average, variance, and Ward's methods are more robust to noise due to the use of multiple linkages in the distance computation

Hierarchical methods are sensitive to a small number of mistakes made during the merging process

- can be due to noise
- no way to undo these mistakes



(b) Bad case with noise

DBSCAN

Highly-cited density-based hierarchical clustering algorithm (Ester et al. 1996)

- clusters are defined as density-connected sets
- epsilon-distance neighbor criterion (Eps)

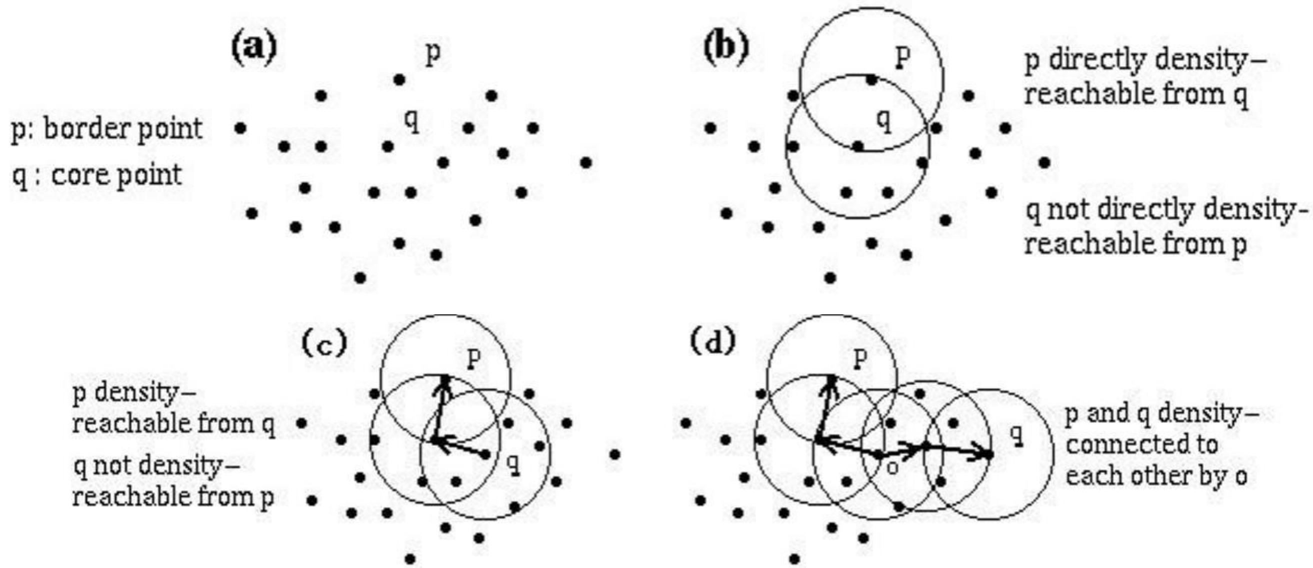
  $N_{Eps}(p) = \{q \in D \mid dist(p,q) \leq Eps\}$
- minimum point cluster membership and core point (MinPts)

  $|N_{Eps}(q)| \geq MinPts$
- notions of density-connected & density-reachable (direct, indirect)
- a point p is directly density-reachable from a point q wrt. Eps, MinPts if

  $p \in N_{Eps}(q)$ and

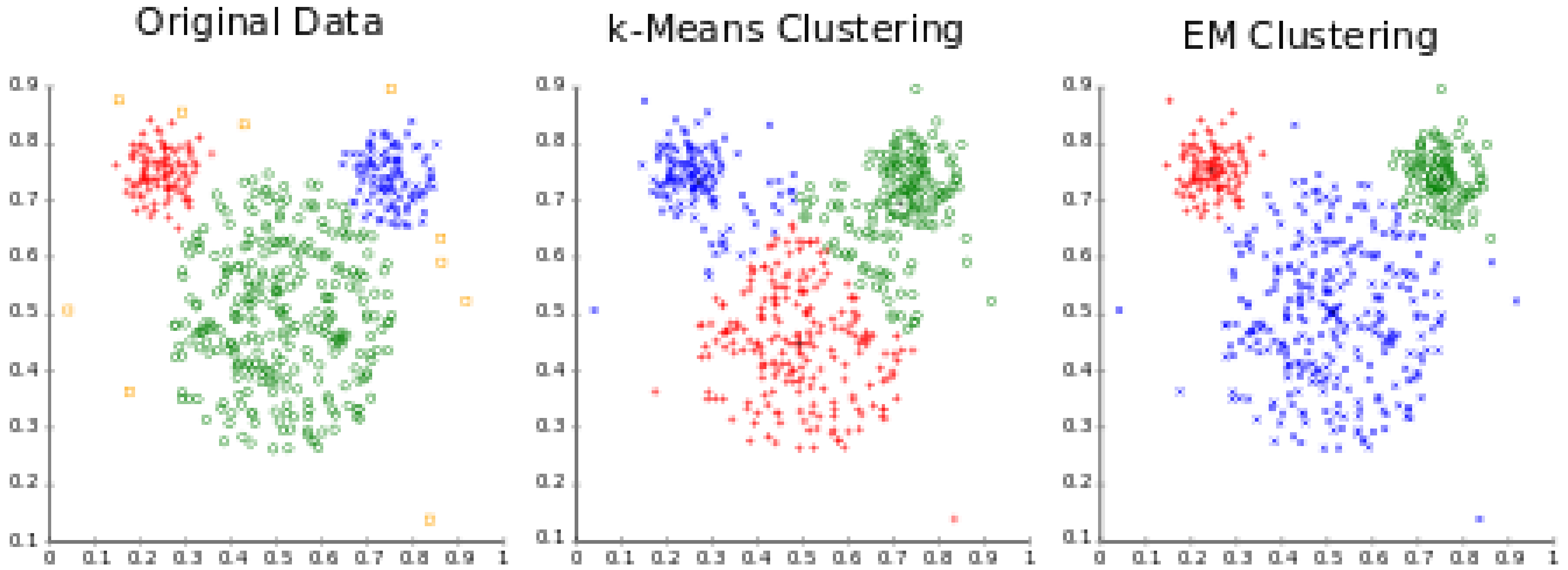
  $|N_{Eps}(q)| \geq MinPts$ (core point condition)

DBSCAN


(a)
p
q
p: border point
q : core point
(b)
p
q
p directly density–
reachable from q
q not directly density-
reachable from p
(c)
p
q
p density–
reachable from q
q not density–
reachable from p
(d)
p
q
o
p and q density–
connected to
each other by o

Probabilistic Extension to K-Means

Different cluster analysis results on "mouse" data set:



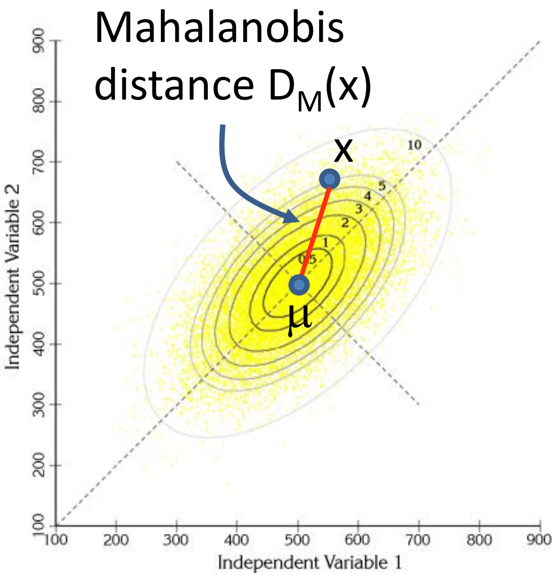
Mahalanobis Distance

The distance between a point P and a distribution D

- measures how many standard deviations P is away from the mean of D
- S is the covariance matrix of the distribution D
- the Mahanalobis distance D_M of a point x to a cluster center μ is

$$D_M(x) = \sqrt{(x-\mu)^T S^{-1} (x-\mu)}.$$

- x and μ are N-dimensional vectors
- S is a N×N matrix
- the outcome $D_M(x)$ is a single-dimensional number (a scalar)



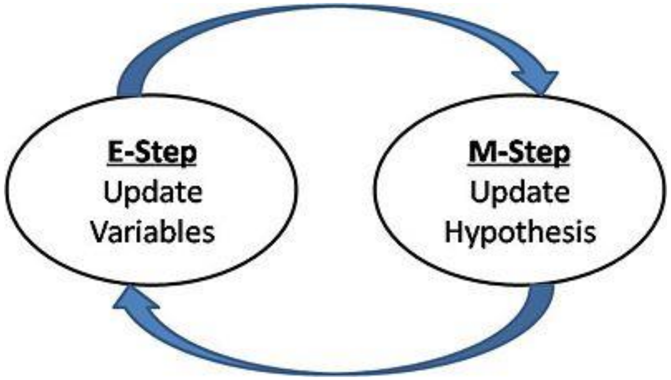
Probabilistic Clustering

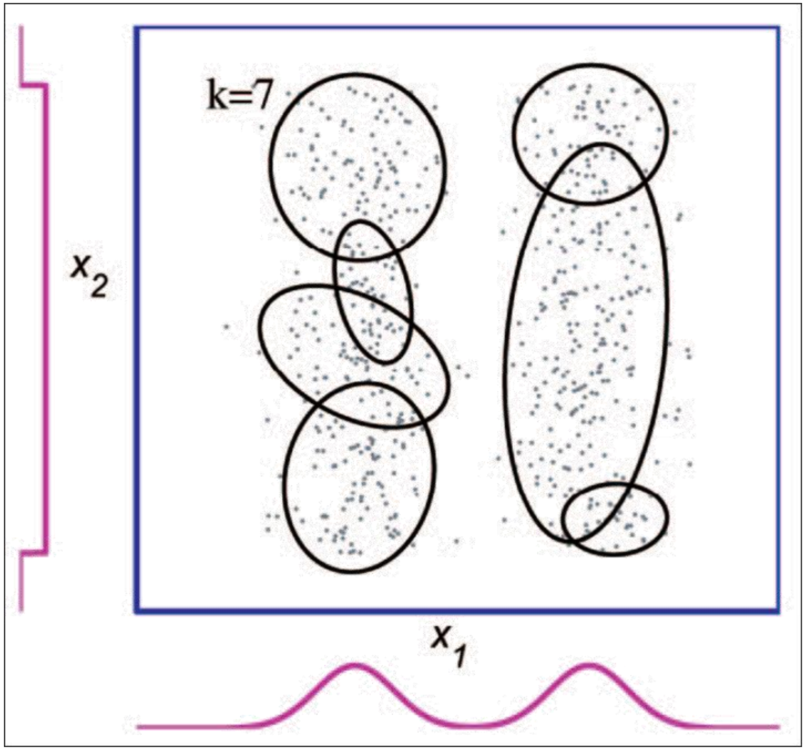
Better match for point distributions

- overlapping clusters are now possible
- better match with real world?
- Gaussian mixtures

Need a probabilistic algorithm

- Expectation-Maximization





EM Algorithm (Mixture Model)

probability P that d_i is in class c_j
(Mahanalobis distance of d_i to c_j)

- Initialize K cluster centers
- Iterate between two steps
  - **E**xpectation step: assign n points to m clusters/classes

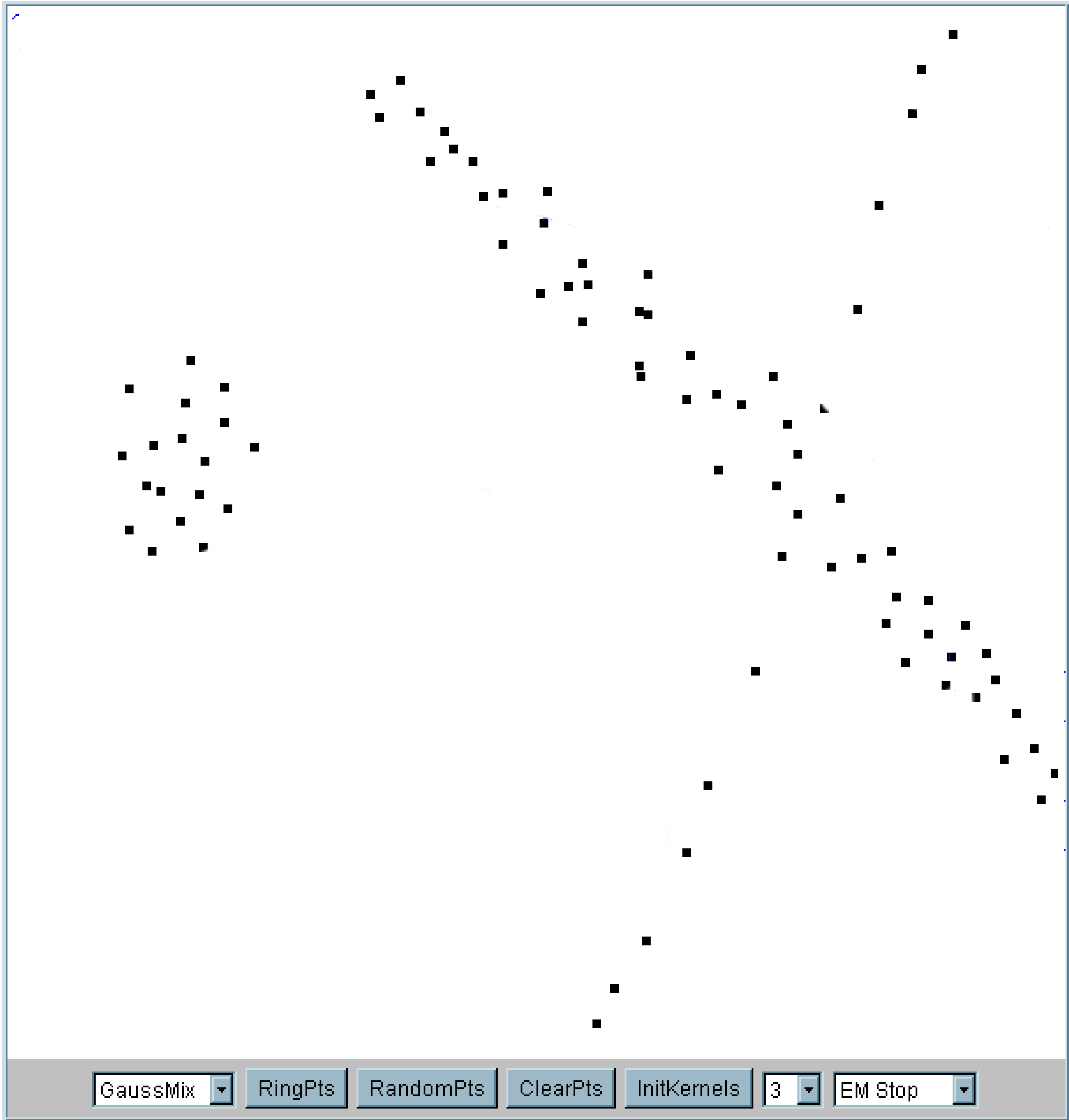
$$P(d_i \in c_k) = w_k P(d_i \mid c_k) \Big/ \sum_j w_j P(d_i \mid c_j)$$

$$w_k = \frac{\sum_i \Pr(d_i \in c_k)}{n}$$ = probability of class c_k

  - **M**aximation step: estimate model parameters

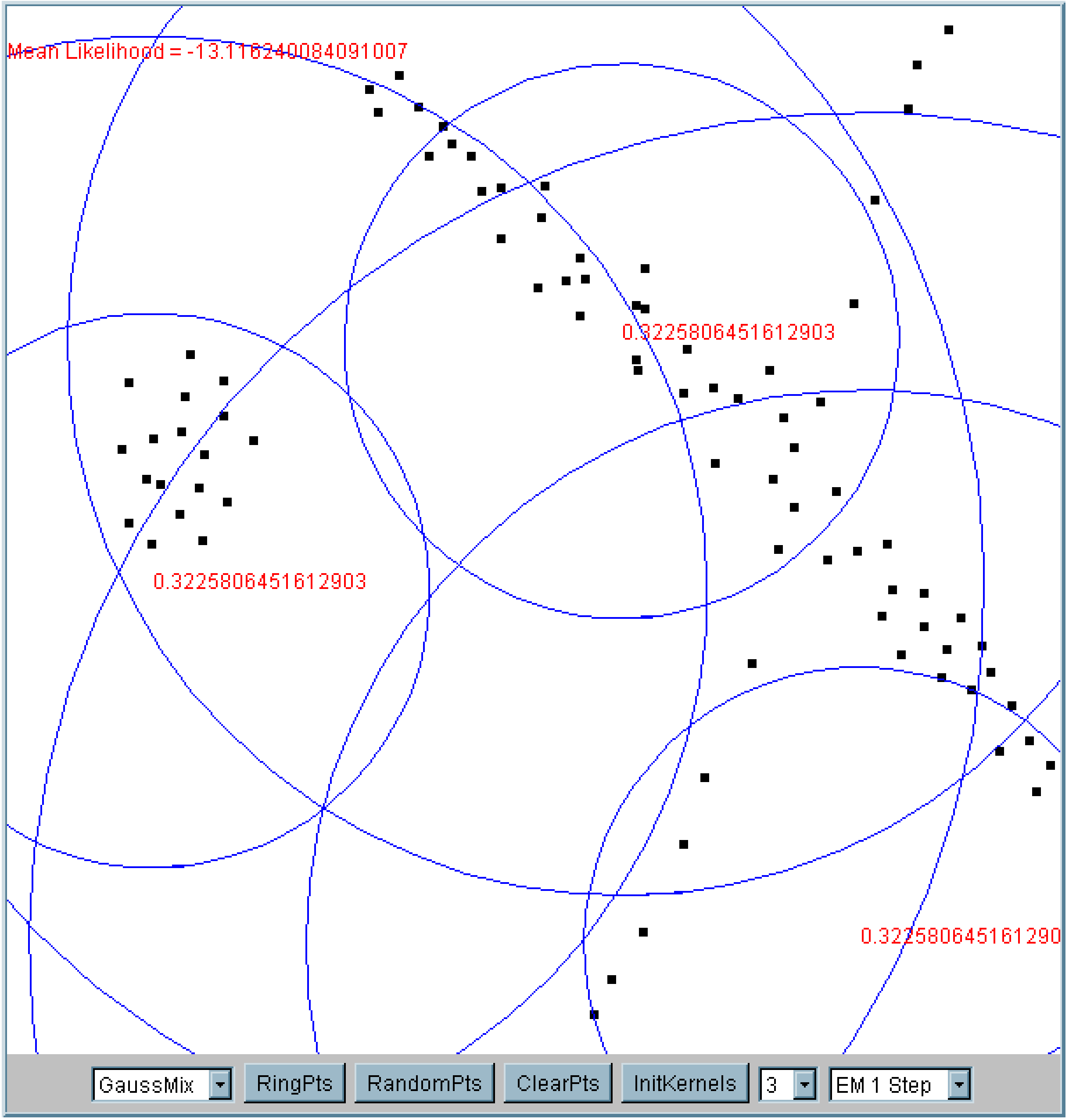
do similar also for covariance matrix S

$$\mu_k = \frac{1}{n} \sum_{i=1}^{n} \frac{d_i P(d_i \in c_k)}{\sum_j P(d_i \in c_j)}$$

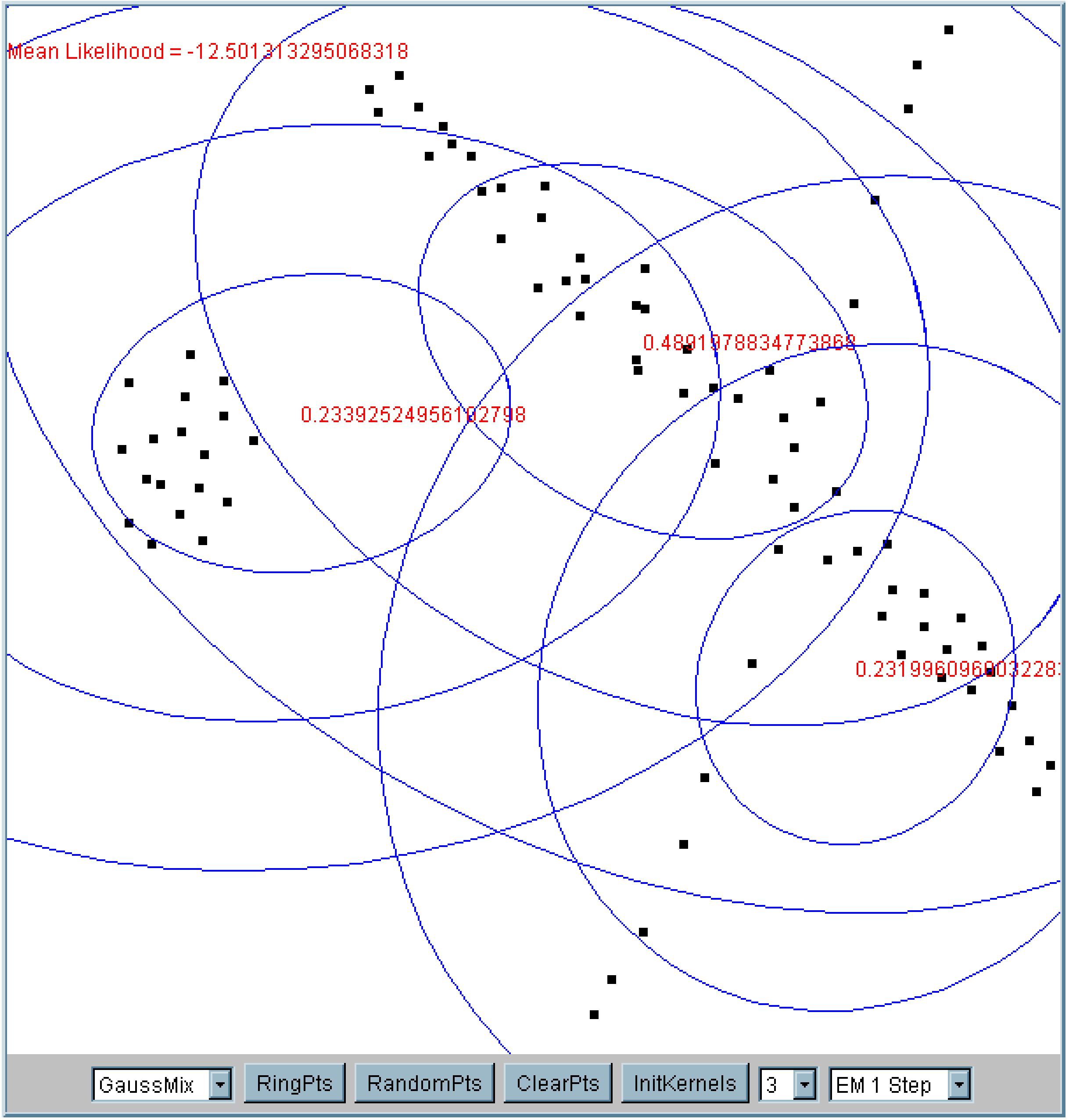

GaussMix
RingPts
RandomPts
ClearPts
InitKernels
3
EM Stop

Iteration 1

The cluster means are randomly assigned

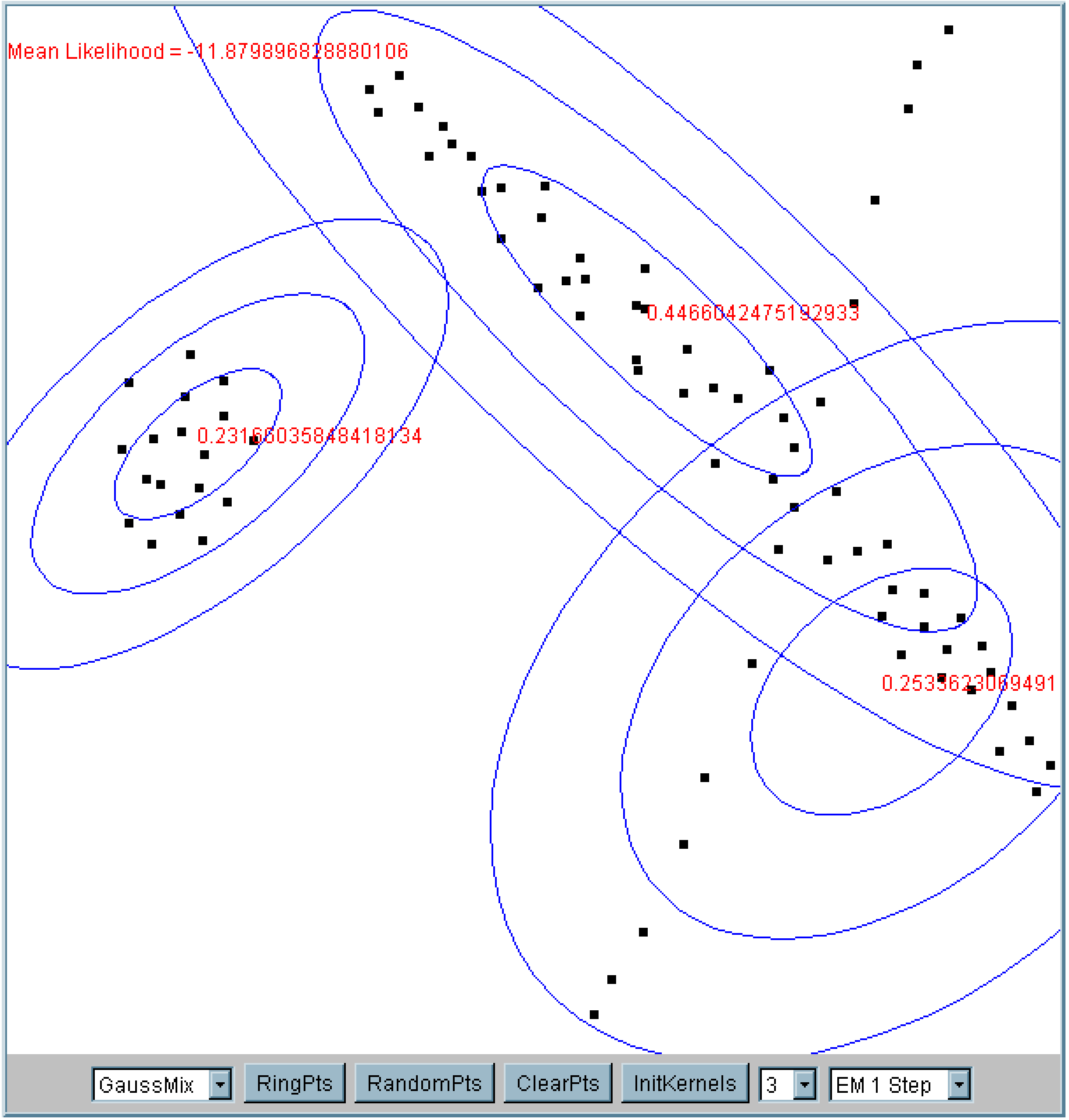


Iteration 2

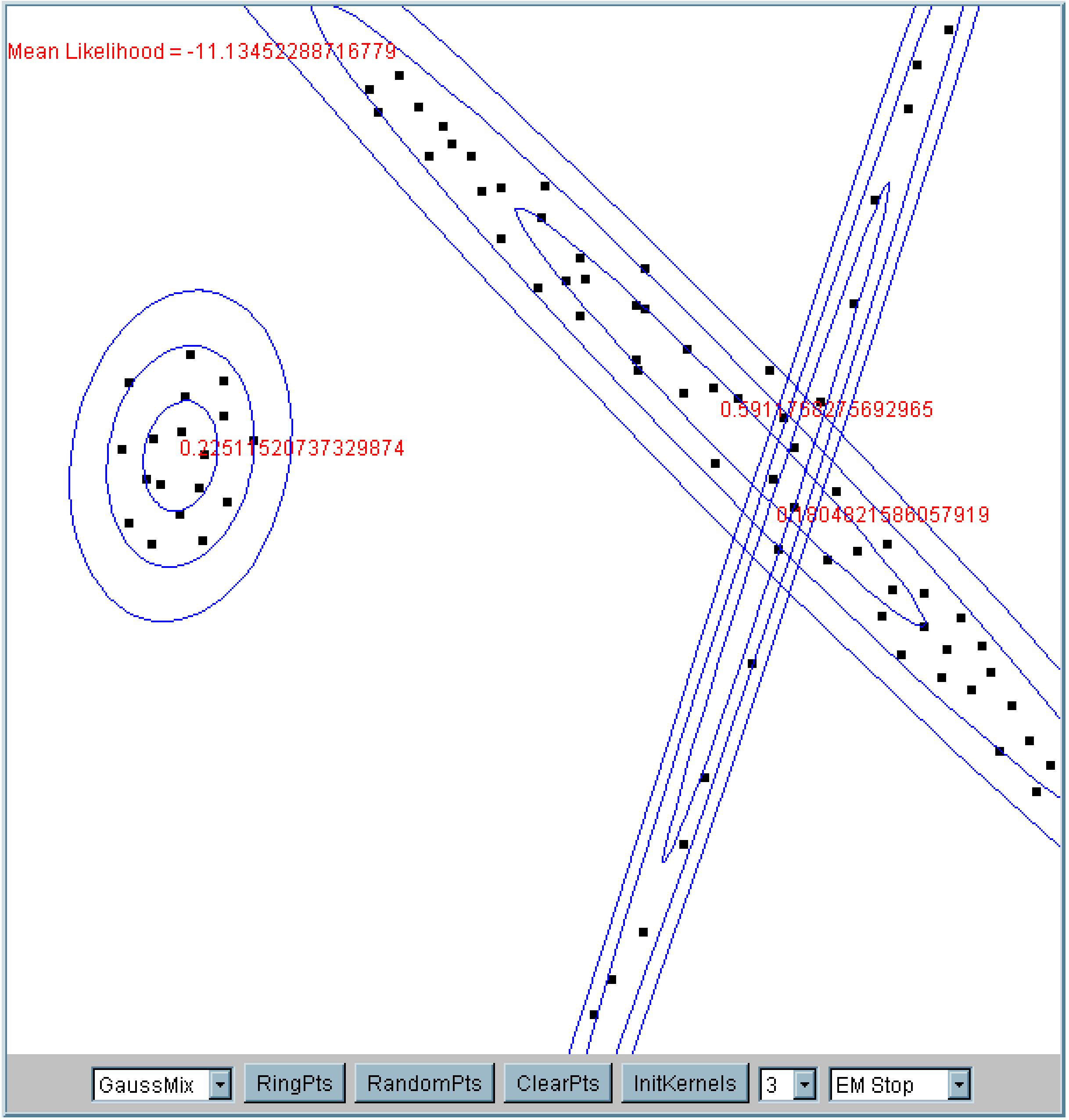


Mean Likelihood = -12.501313295068318
0.4891978834773868
0.23392524956102798
0.231996096003228
GaussMix
RingPts
RandomPts
ClearPts
InitKernels
3
EM 1 Step

Iteration 5


Mean Likelihood = -11.879896828880106
0.4466042475192933
0.23166035848418134
0.2533623069491
GaussMix
RingPts
RandomPts
ClearPts
InitKernels
3
EM 1 Step

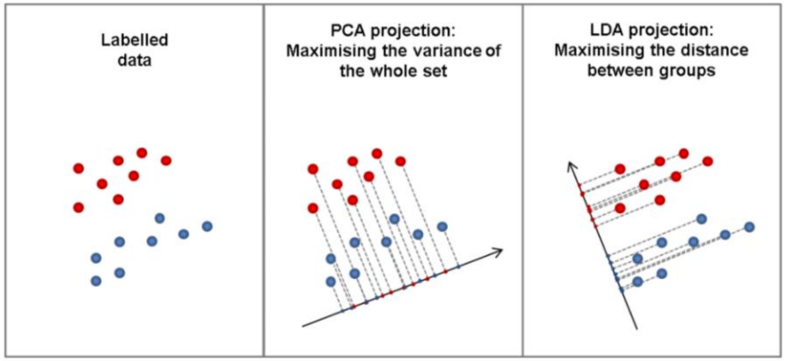
Iteration 25



Linear Discriminant Analysis (LDA)

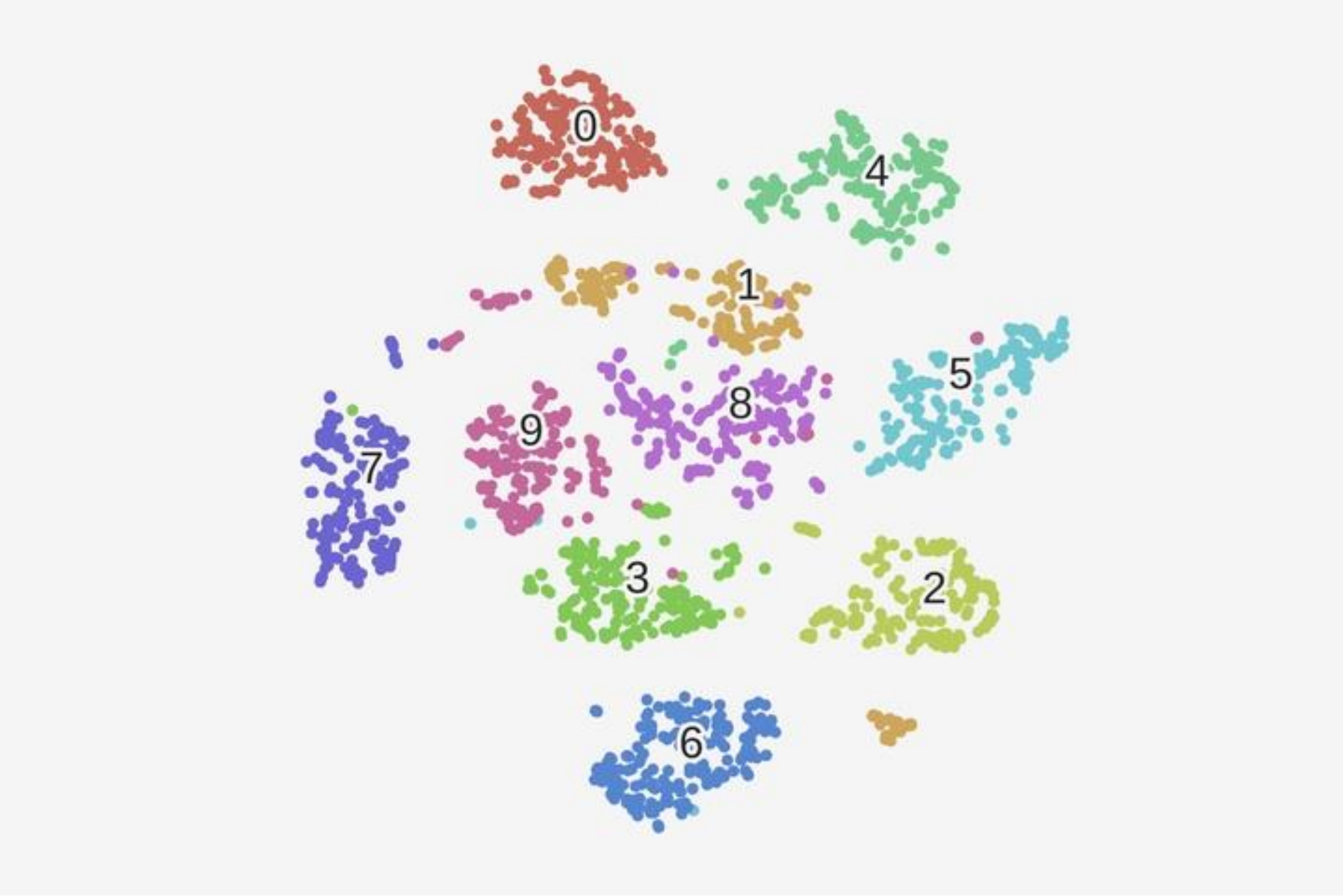
Procedure

- maximize inter-class variance $S_b = \sum_{i=1}^{g} N_i(\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T$
- minimize intra-class variance $S_w = \sum_{i=1}^{g}\sum_{j=1}^{N_i}(x_{i,j} - \bar{x}_i)(x_{i,j} - \bar{x}_i)^T$
- using this ratio $P_{lda} = \arg\max_{P} \frac{|P^T S_b P|}{|P^T S_w P|}$ Fisher Criterion P is low-Dim projection
- can be solved using Eigenvector decomposition
- finds a basis that maximally separates the classes
- Dim(P) is the # of classes *g*



T-SNE

t-distributed stochastic neighbor embedding



T-SNE Distance Metric

Uses the following density-based (probabilistic) distance metric

$$p_{j|i} = \frac{\exp\left(-|x_i - x_j|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-|x_i - x_k|^2 / 2\sigma_i^2\right)}$$

Measures how (relatively) close x_j is from x_i, considering a Gaussian distribution around x_i with a given variance σ^2_i.

- this variance is different for every point
- t is chosen such that points in dense areas are given a smaller variance than points in sparse areas

T-SNE Implementation

Use a symmetrized version of the conditional similarity:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}$$

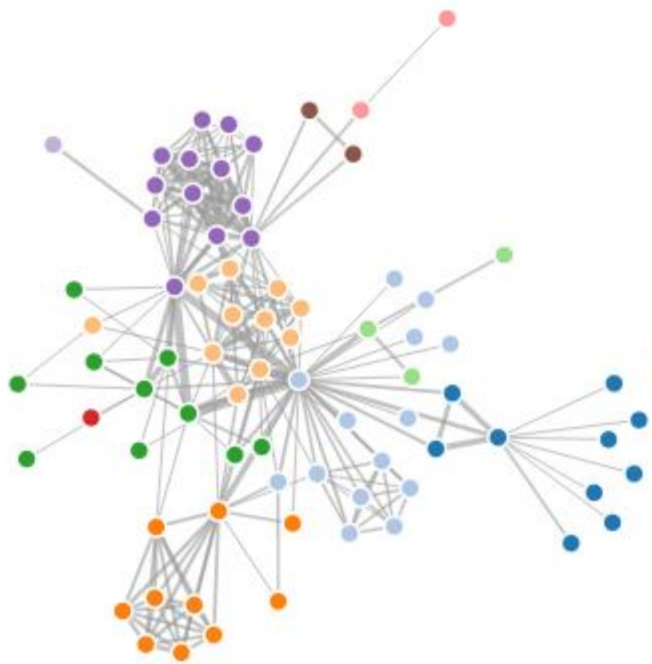
Similarity (distance) metric for mapped points:

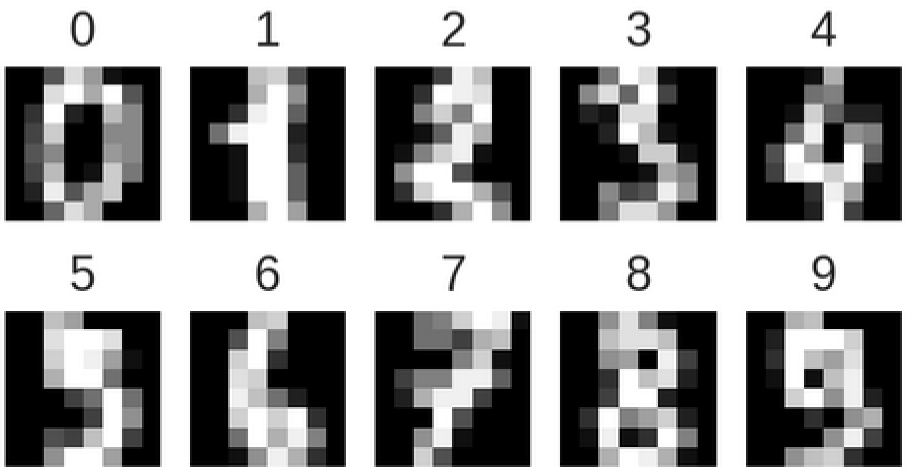
$$q_{ij} = \frac{f(|x_i - x_j|)}{\sum_{k \neq i} f(|x_i - x_k|)} \quad \text{with} \quad f(z) = \frac{1}{1+z^2}$$

This uses the t-student distribution with one degree of freedom, or Cauchy distribution, instead of a Gaussian distribution

Layout

Can use mass-spring system enforcing minimum of $|p_{ij} - q_{ij}|$





The classic *handwritten digits* datasets. It contains 1,797 images with 8∗8=64 pixels each.

Animated layout

More information

See [this webpage]

Summary

Cluster analysis

- detect and eliminate irrelevant (noisy) attributes using entropy
- build a cluster hierarchy bottom-up or top-down
- different metrics to join points and clusters
- the DBSCAN algorithm for more noise-robust clustering of arbitrary shapes
- EM-ML probabilistic clustering as an extension of k-means for less sensitivity to noise and overlapping clusters
- LDA to maximize separations of clusters (and as a tradeoff minimize intra-cluster spread)
- more sophisticated local density-based clustering and dimension reduction using t-SNE