CSE 564
Visualization & Visual Analytics

Cluster Analysis & Dimension reduction

Klaus Mueller

Computer Science Department
Stony Brook University and SUNY Korea

Lecture	Topic	Projects
1	Intro, schedule, and logistics	
2	Applications of visual analytics	
3	Basic tasks, data types	Project #1 out
4	Data assimilation and preparation	
5	Introduction to D3	
6	Bias in visualization	
7	Data reduction and dimension reduction	
8	Data reduction and dimension reduction	Project #2(a) out
9	Visual perception and cognition	
10	Visual design and aesthetics	
11	High-dimensional data visualization: linear methods	
12	High-dimensional data visualization: non-linear methods	Project #2(b) out
13	Cluster analysis: numerical data	
14	Cluster analysis: categorical data	
15	Principles of interaction	
16	Midterm #1	
17	Visual analytics	Final project proposal call out
18	The visual sense making process	
19	Maps	
20	Visualization of hierarchies	Final project proposal due
21	Visualization of time-varying and time-series data	
22	Foundations of scientific and medical visualization	
23	Volume rendering	Project 3 out
24	Scientific and medical visualization	Final Project preliminary report due
25	Visual analytics system design and evaluation	
26	Memorable visualization and embellishments	
27	Infographics design	
28	Midterm #2	

When to Use 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 analyses

- use as a preprocessing step for classification and outlier detection
- use it for sampling and data reduction

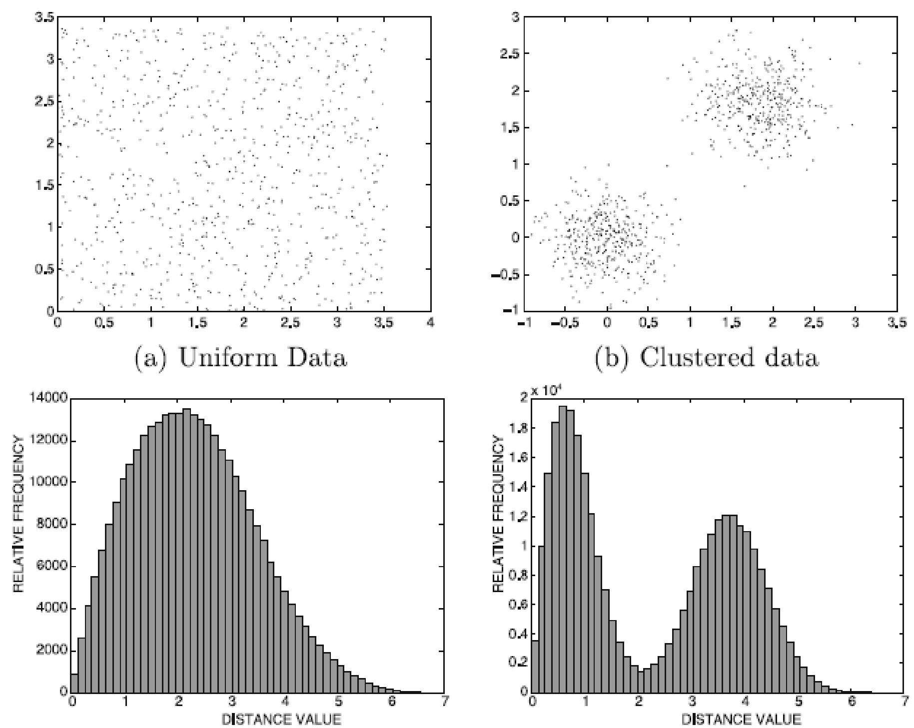
Attribute Selection

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

avoid

keep

histogram of pairwise distances in N-D space



(a) Uniform Data (b) Clustered data

(c) Distance distribution (uniform) (d) Distance distribution (clustered)

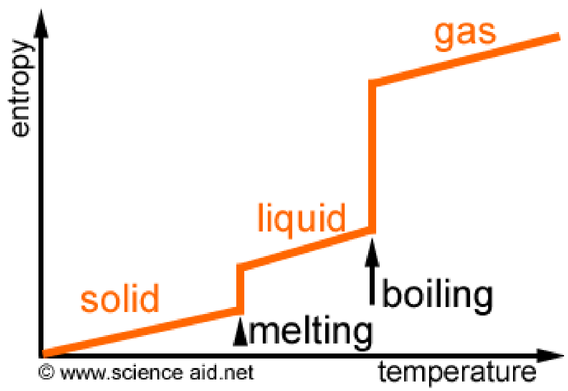
Attribute Selection

How to measure attribute "worthiness"

- use entropy

Entropy

- originates in thermodynamics
- measures lack of order or predictability



Entropy in statistics and information theory

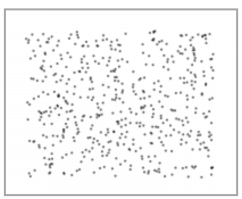
- has a value of 1 for uniform distributions (not predictable)
- knowing the value has a lot of information (high surprise)
- has a value of 0 for a constant signal (fully predicable)
- knowing the value has zero information (low surprise)

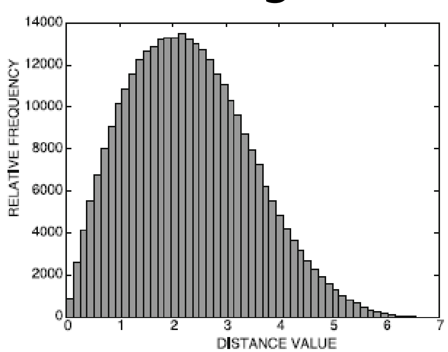
Entropy

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

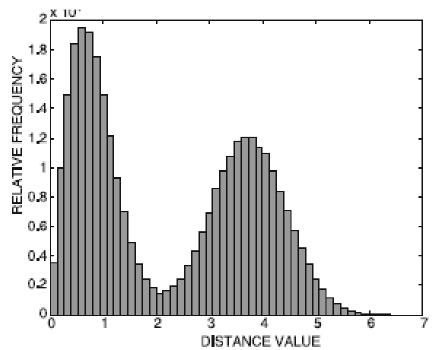
E high

E low

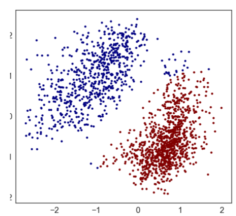


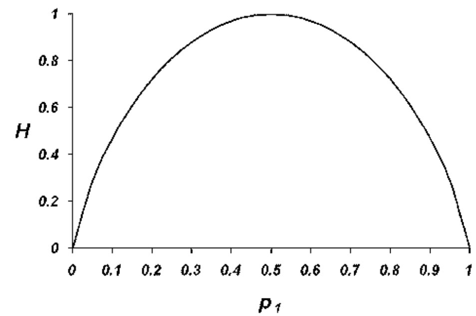


(c) Distance distribution (uniform)



(d) Distance distribution (clustered)



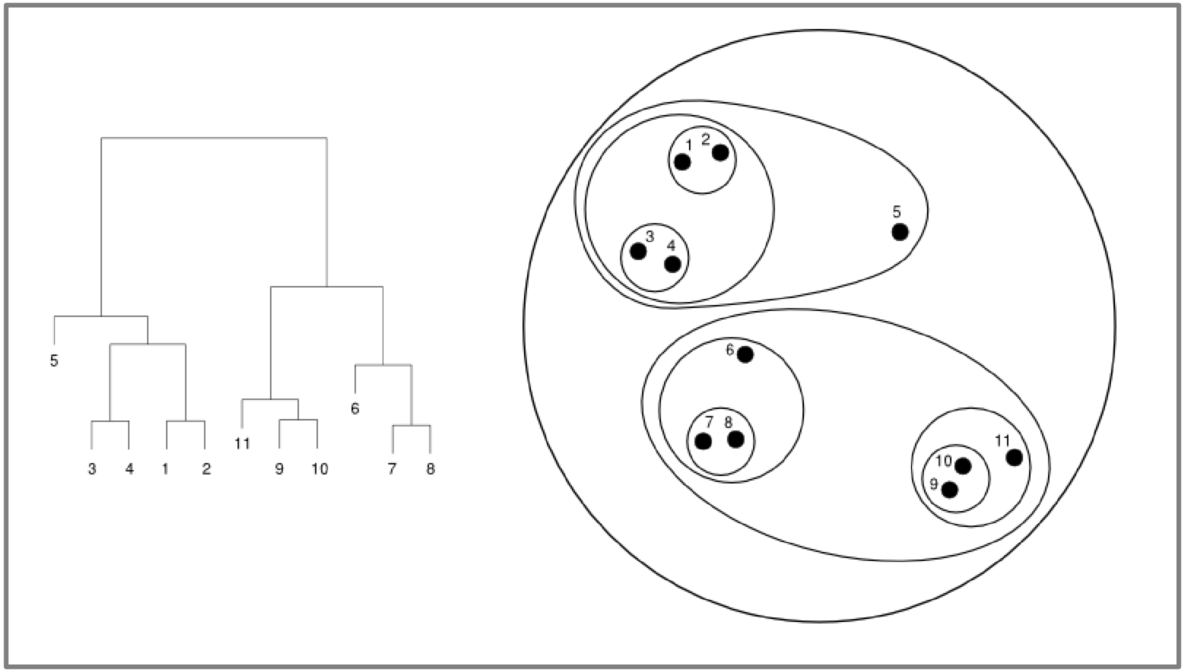


Binary source
(e.g. coin)

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

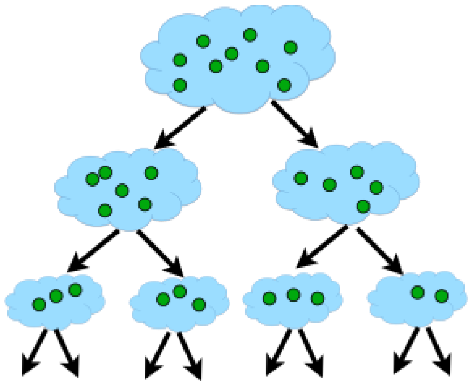


Two options for building the dendrogram on the left

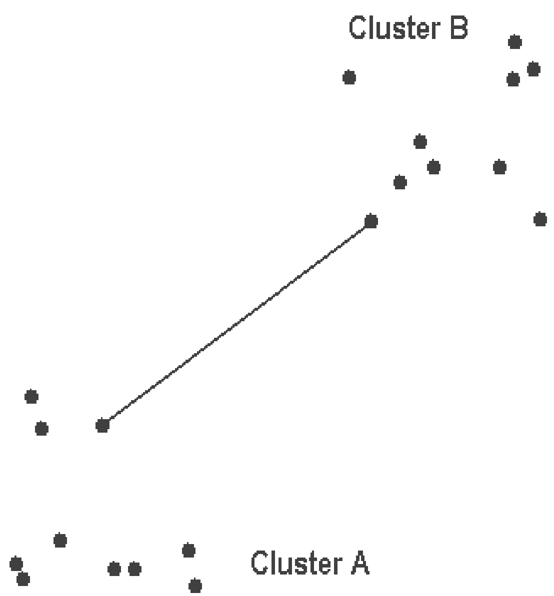
- top down (divisive)
- bottom up (agglomerative)

Bottom-Up Agglomerative Methods

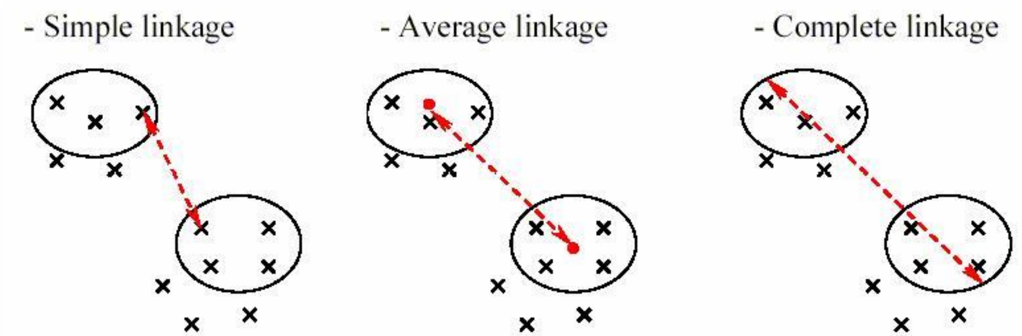
Algorithm *AgglomerativeMerge*(Data: $\mathcal{D}$)
begin
  Initialize $n \times n$ distance matrix M using $\mathcal{D}$;
  repeat
    Pick closest pair of clusters i and j using M;
    Merge clusters i and j;
    Delete rows/columns i and j from M and create
      a new row and column for newly merged 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 (best-case) linkage

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

Complete (worst-case) 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:

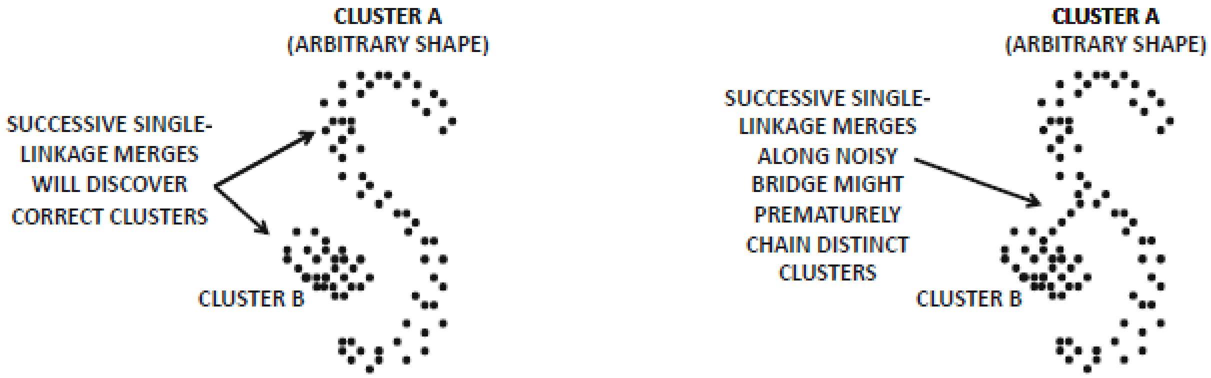
- closest centroid, variance-minimization, Ward's method

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

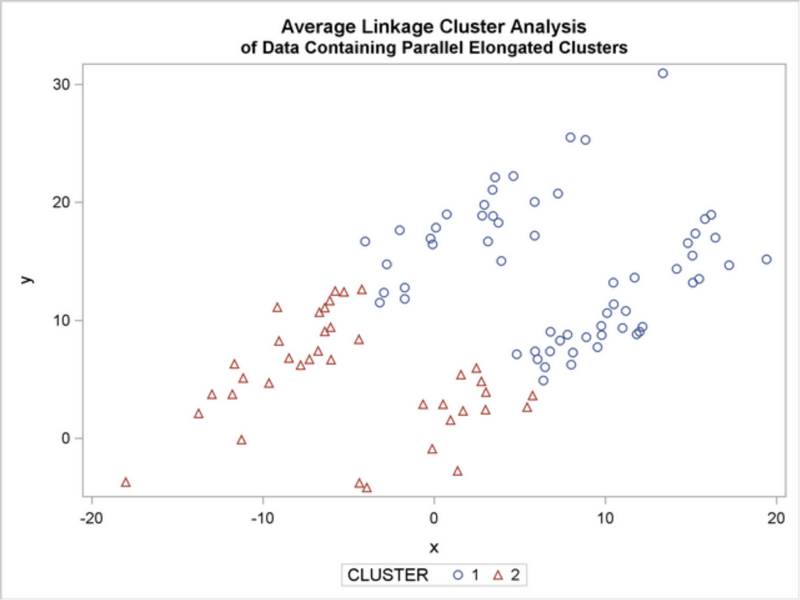


(a) Good case with no noise (b) Bad case with noise

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

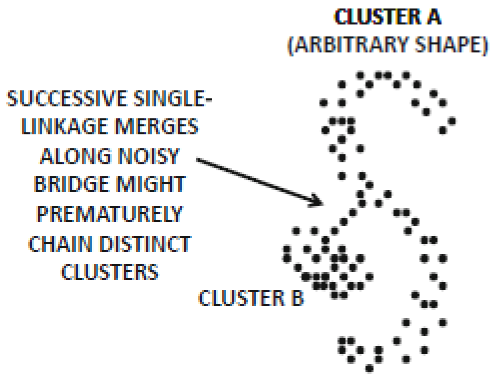


Comparison

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)

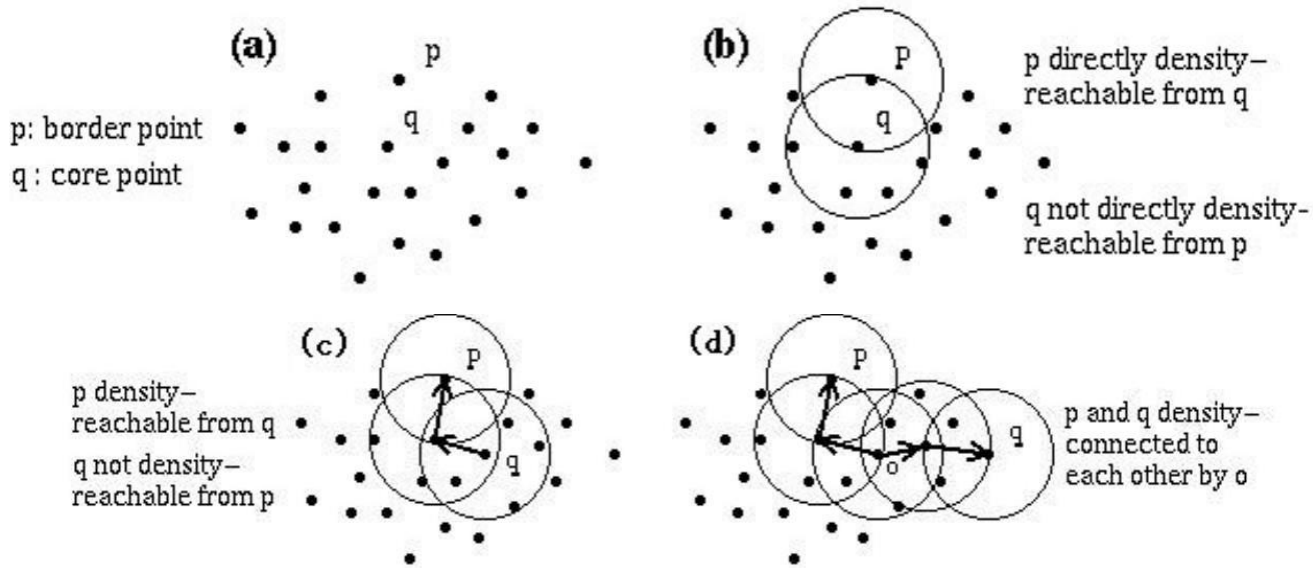
  $N_{Eps}(p) = \{q \in D \mid dist(p,q) \leq Eps\}$
- minimum point cluster membership and core point (MinPts)

  $|N_{Eps}(q)| \geq MinPts$
- 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

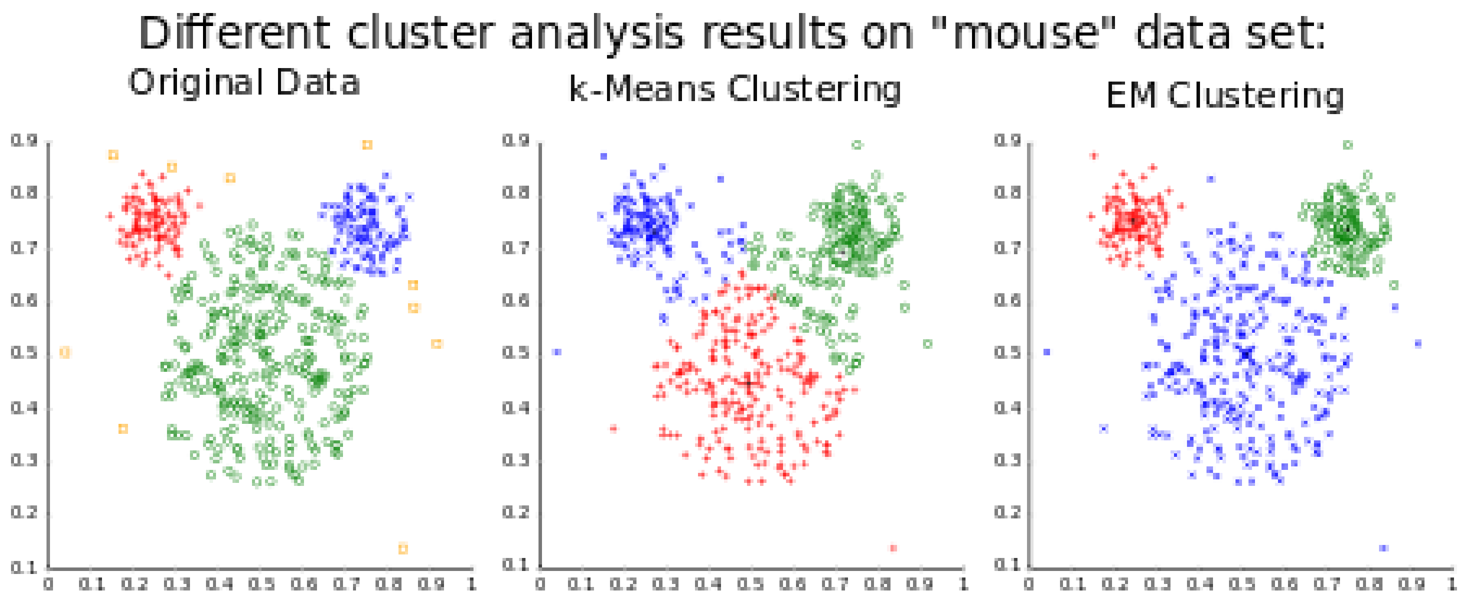
  $|N_{Eps}(q)| \geq MinPts$ (core point condition)

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

First a comparison:



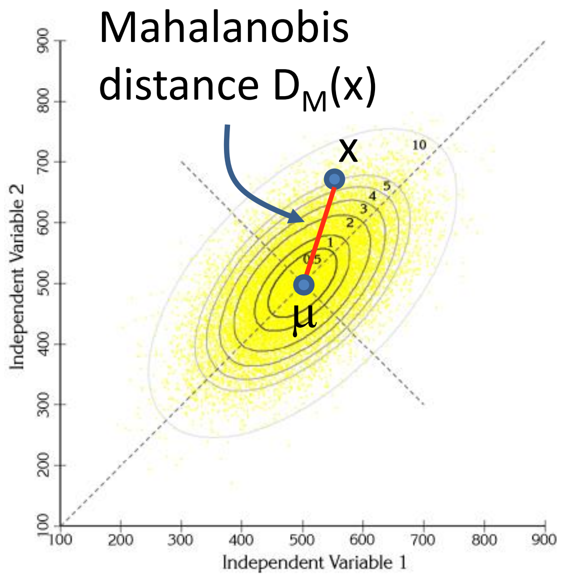
Mahalanobis Distance

The distance between a point X and a distribution D

- measures how many standard deviations X is away from the mean μ 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 the N×N covariance matrix
- the outcome $D_M(x)$ is a single-dimensional number



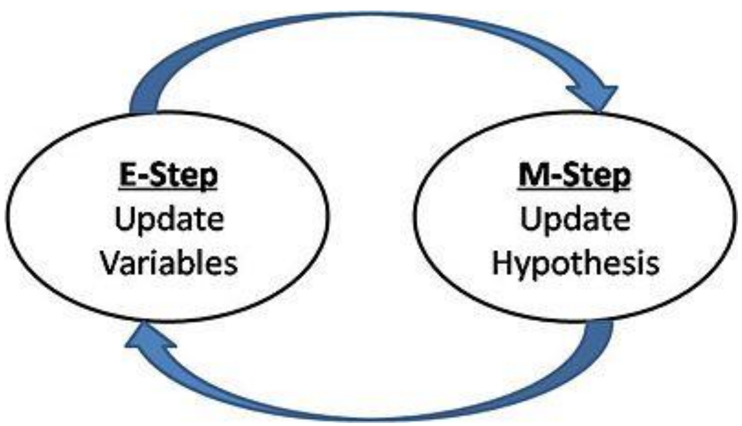
Probabilistic Clustering

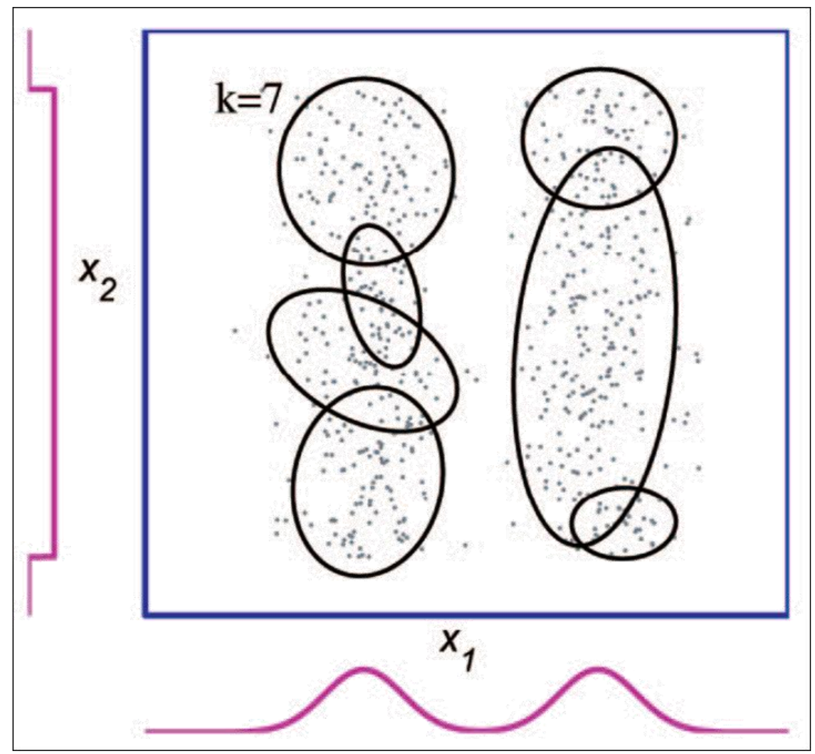
Is a better match for point distributions

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

Need a probabilistic algorithm

- Expectation-Maximization





EM Algorithm (Mixture Model)

probability that data point 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 points to *m* clusters/classes

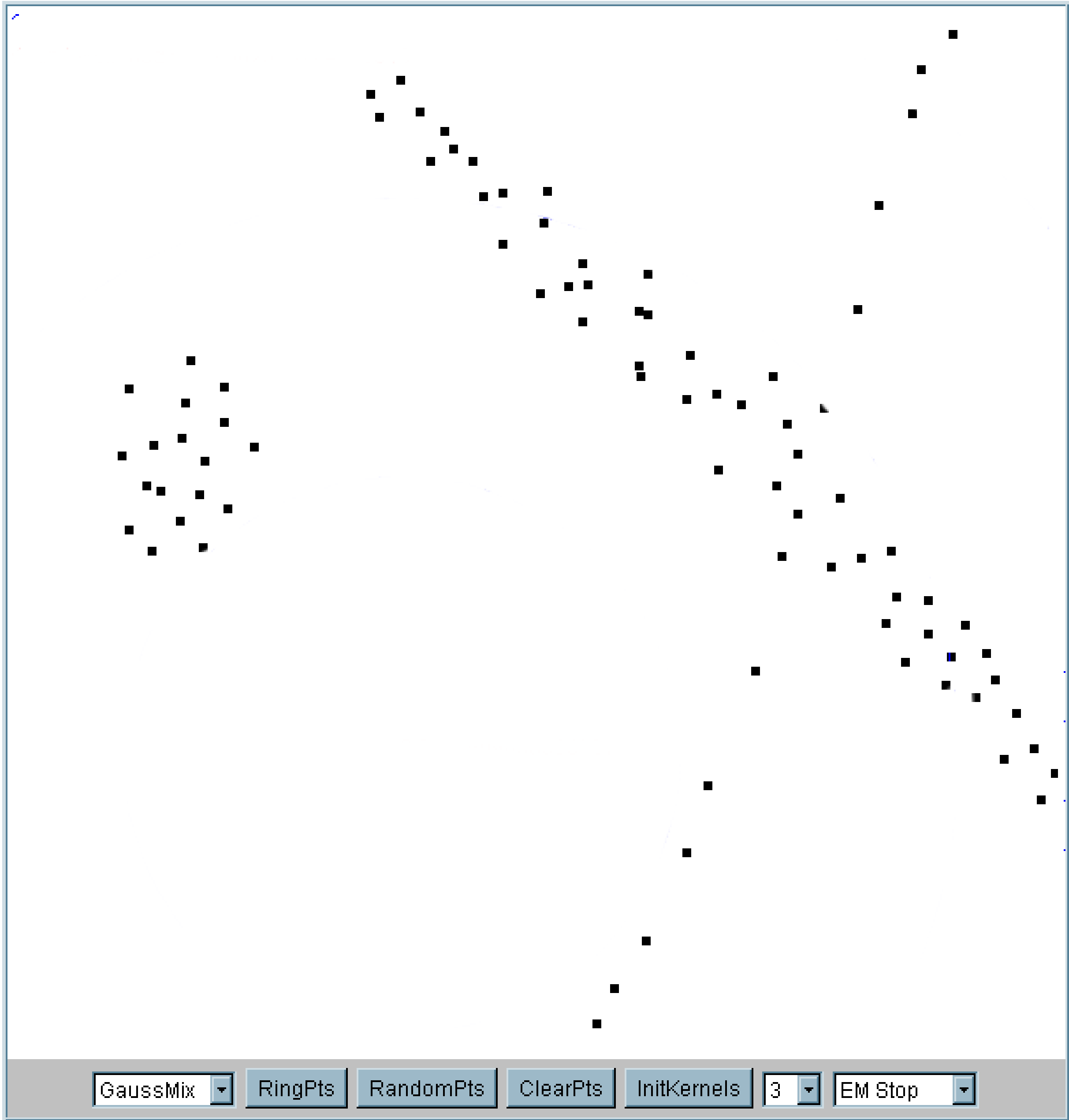
$$P(d_i \in c_k) = w_k \Pr(d_i \mid c_k) \Big/ \sum_j w_j \Pr(d_i \mid c_j)$$

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

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

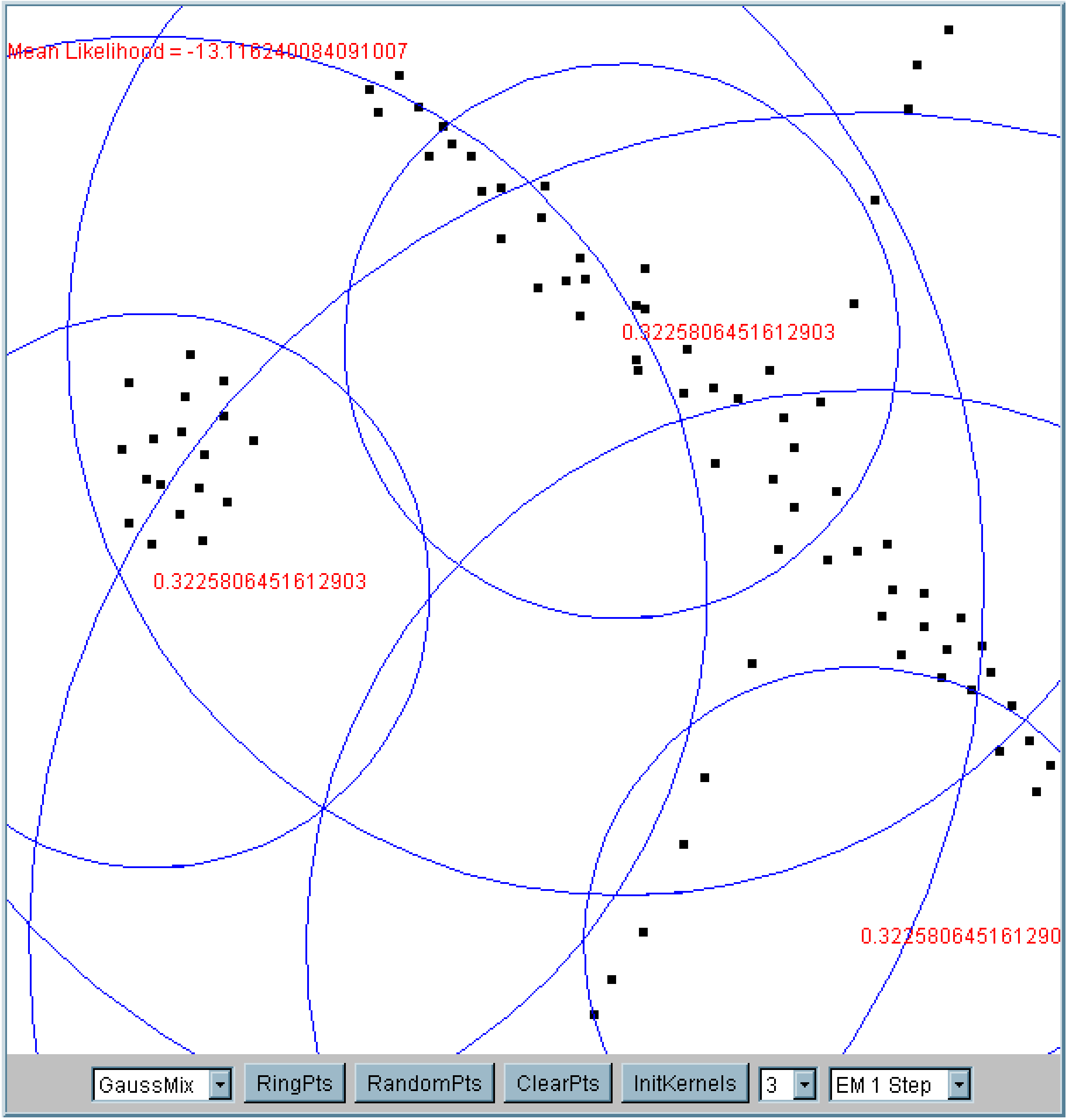
do similar also for
covariance matrix S

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

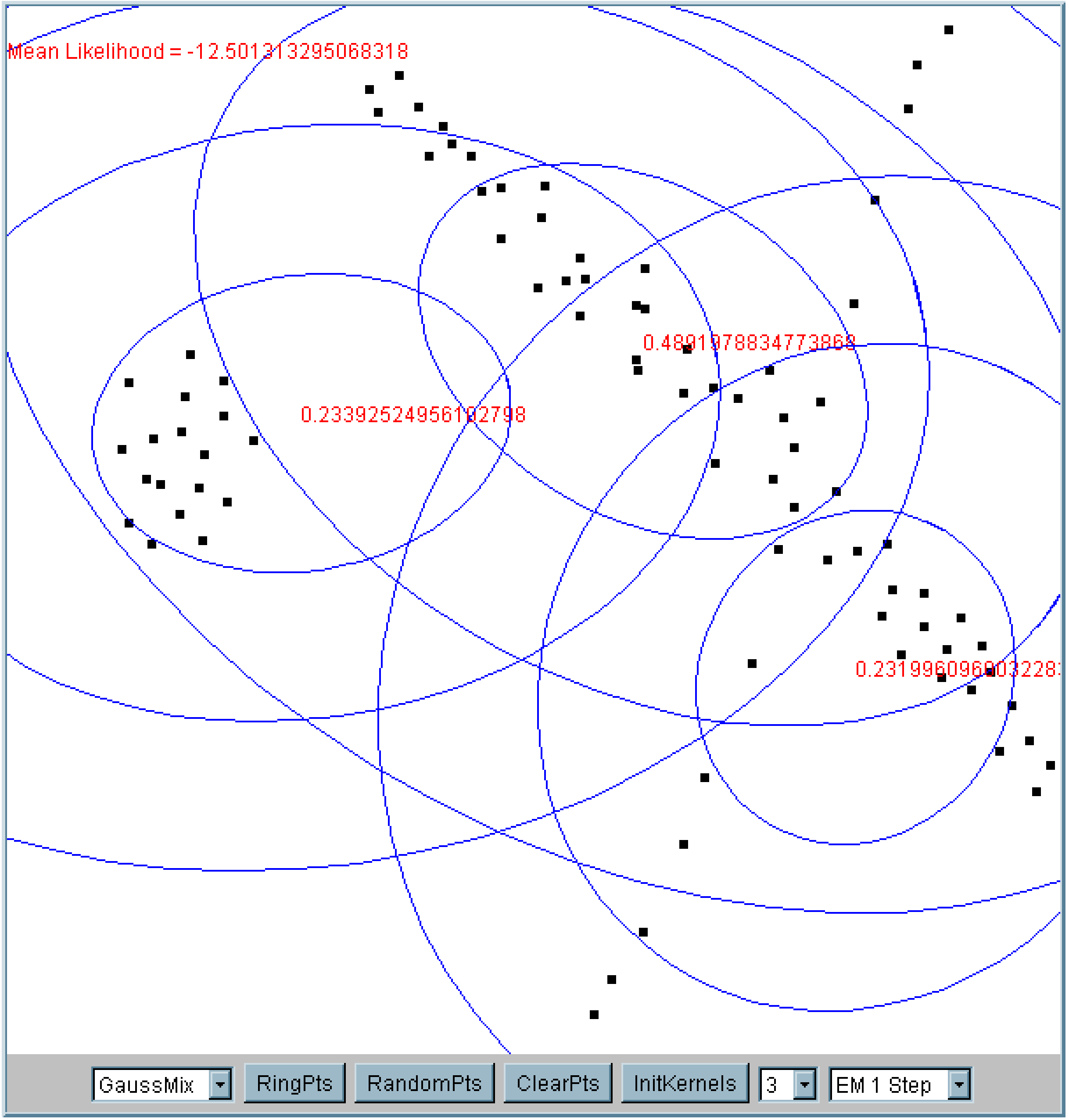

GaussMix
RingPts
RandomPts
ClearPts
InitKernels
3
EM Stop

Iteration 1

The cluster means are randomly assigned

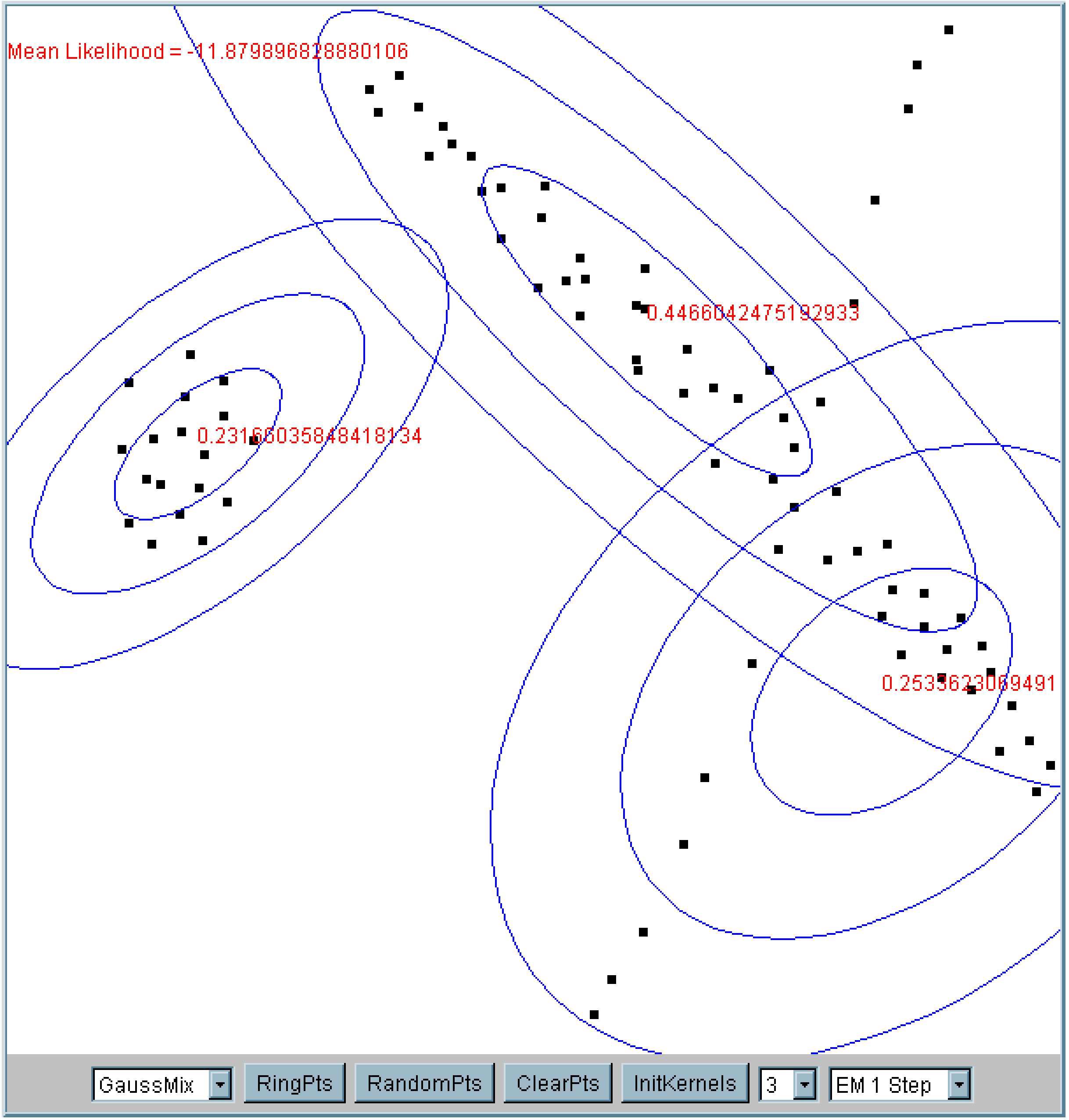


Iteration 2



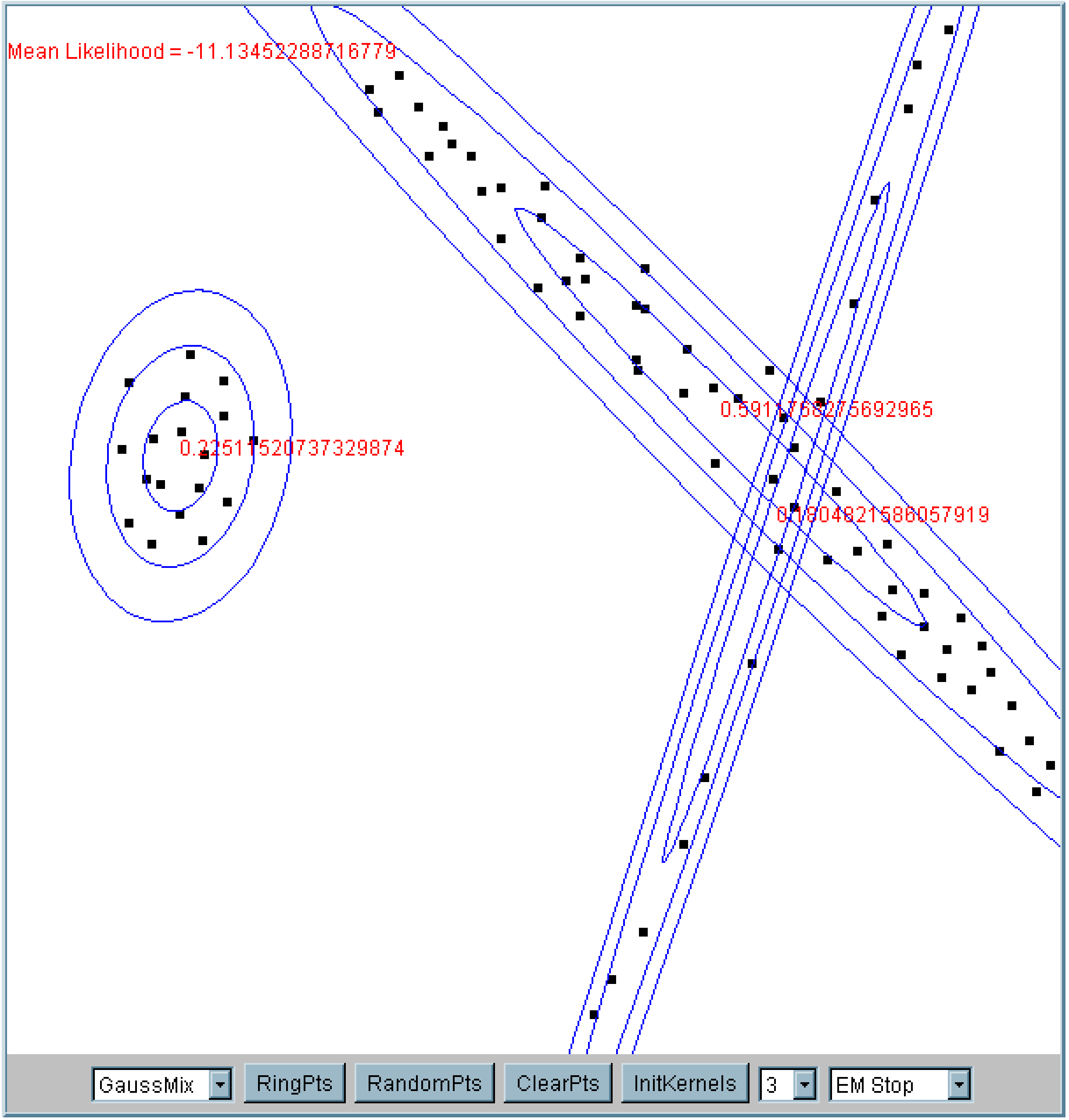
Mean Likelihood = -12.501313295068318
0.4891978834773868
0.23392524956102798
GaussMix
RingPts
RandomPts
ClearPts
InitKernels
3
EM 1 Step

Iteration 5



Mean Likelihood = -11.879896828880106
0.4466042475192933
0.23166035848418134
0.2536230069491
GaussMix
RingPts
RandomPts
ClearPts
InitKernels
3
EM 1 Step

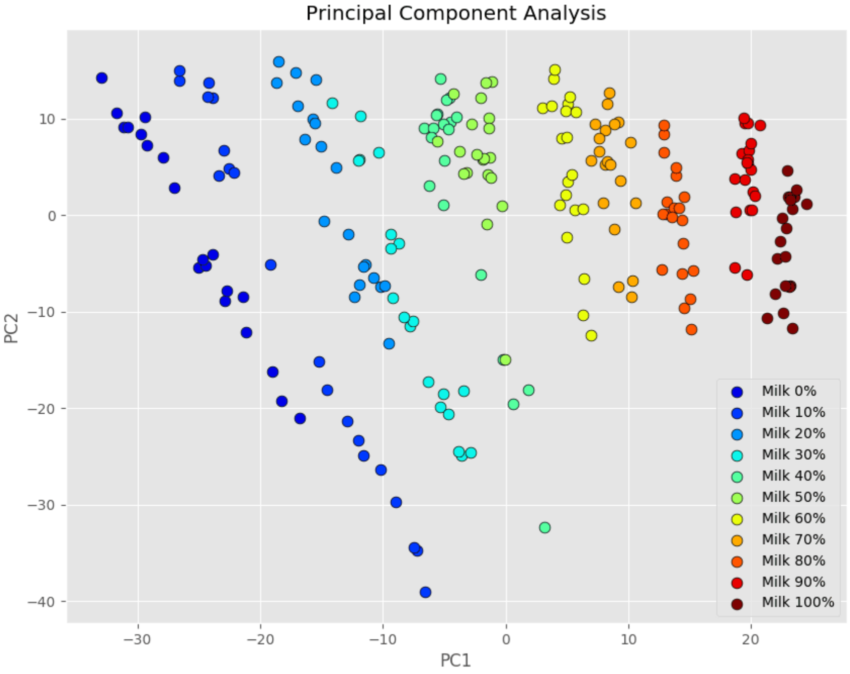
Iteration 25



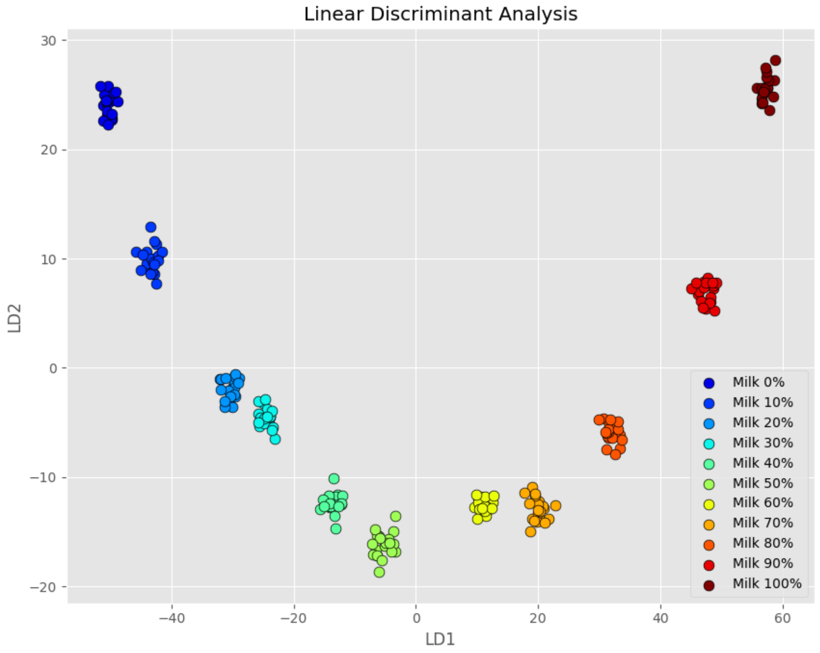
Linear Discriminate Analysis (LDA)

LDA requires class labels, PCA does not

- having class labels enables better segmentation



PCA



LDA

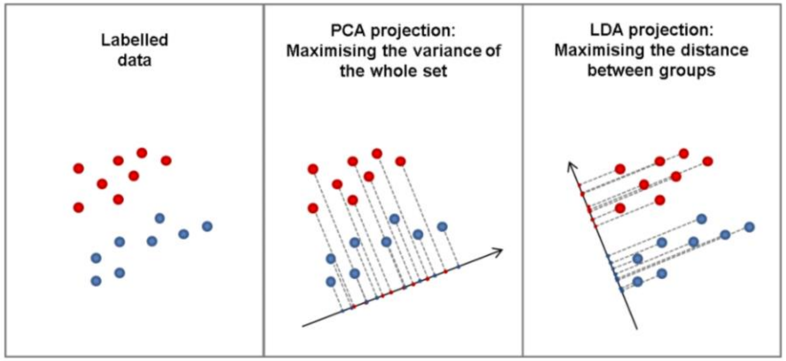
Linear Discriminate Analysis (LDA)

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 $P_{lda} = \arg\max_{P} \frac{|P^T S_b P|}{|P^T S_w P|}$ P is low-Dim projection

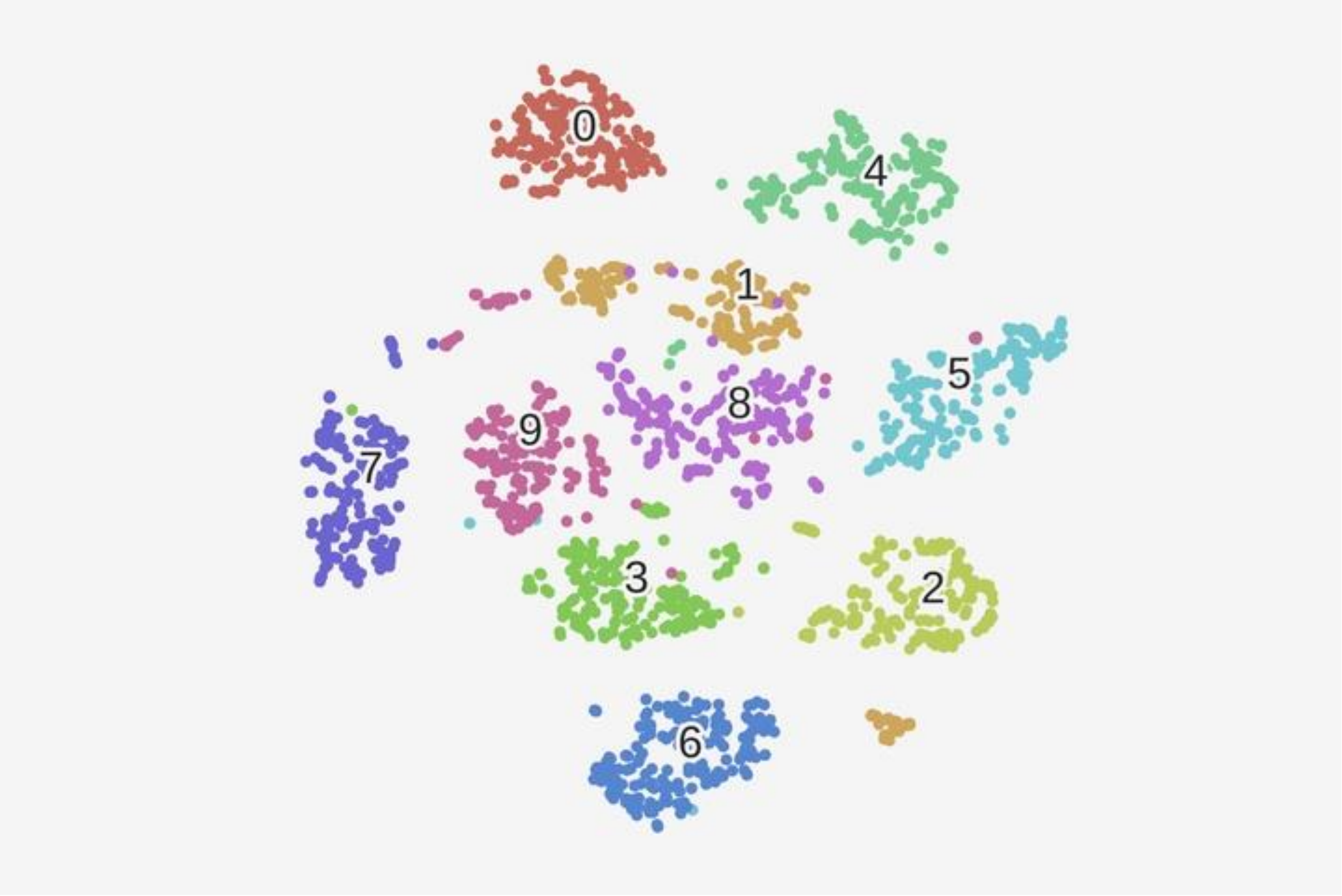
Fisher Criterion

- 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:

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

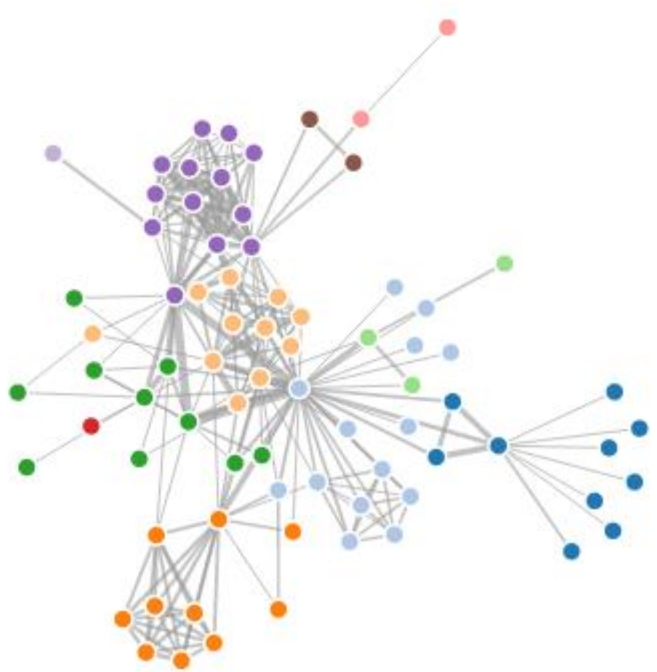
Similarity (distance) metric for mapped points:

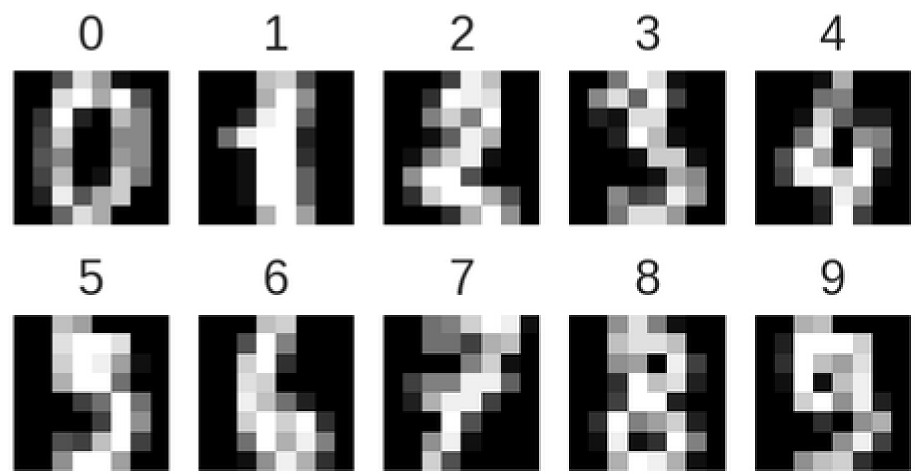
$$q_{ij} = \frac{f(|x_i - x_j|)}{\sum_{k \neq i} f(|x_i - x_k|)} \quad \text{with} \quad f(z) = \frac{1}{1+z^2}$$

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

9

More information

See this webpage

Shortcomings of t-SNE

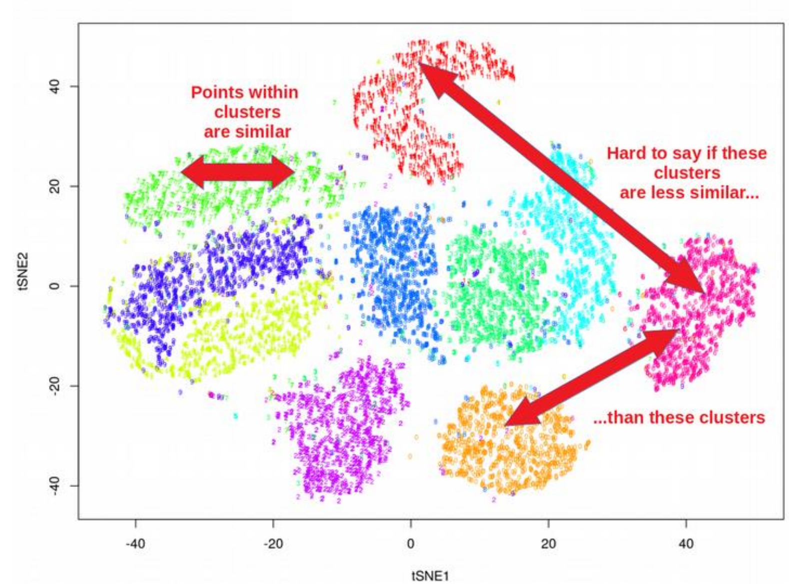
t-SNE does not preserve global data structure

- only within cluster distances are meaningful
- between cluster similarities are not guaranteed

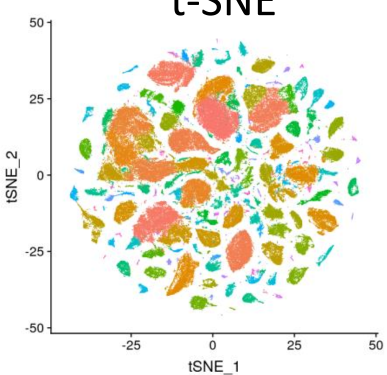
More recently introduced: U-MAP

- follows the philosophy of t-SNE
- but introduces many improvements
- more info, for example, [here]()

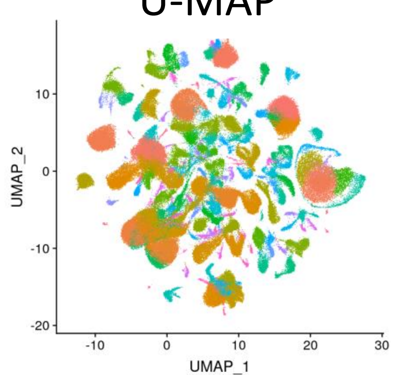
t-SNE MINST



t-SNE



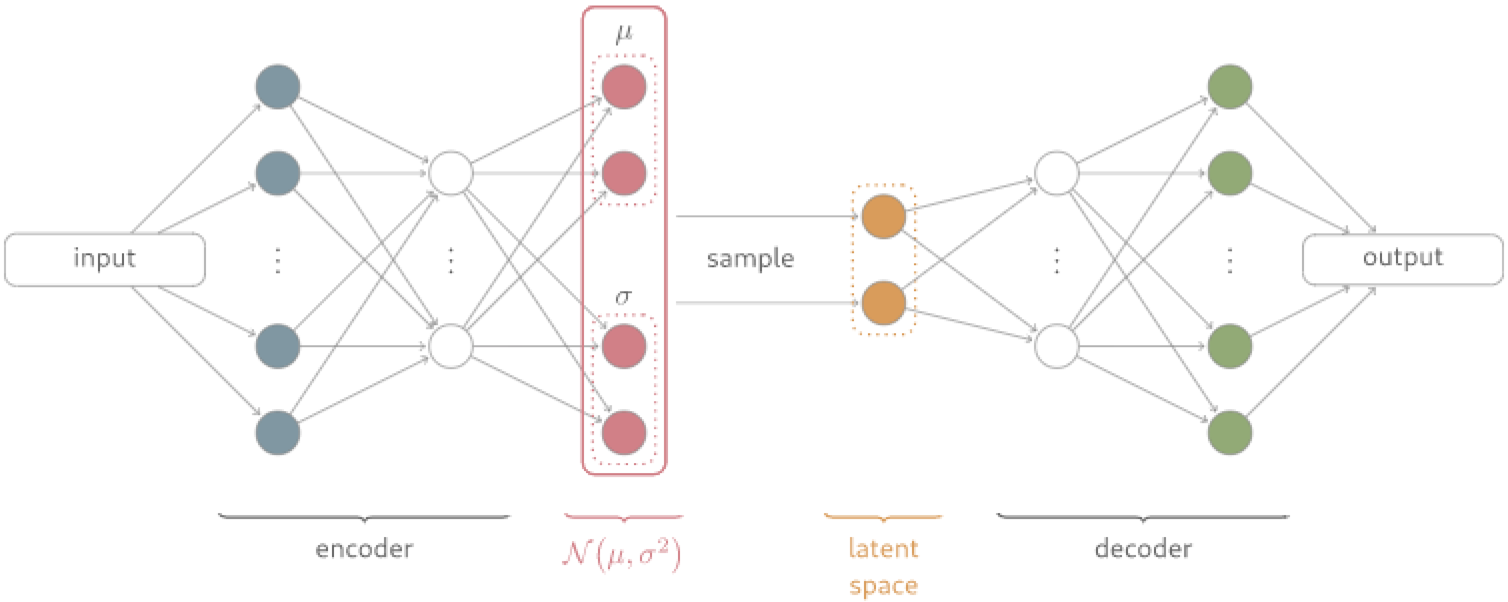
U-MAP



Reduction via Neural Network

Train a Variational Autoencoder (VAE)

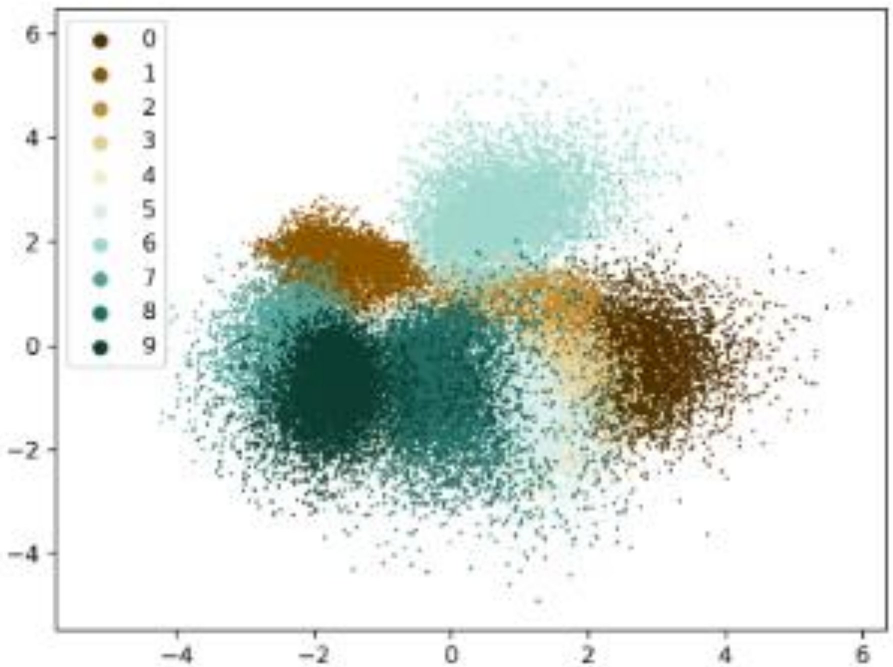
- optimize the output reconstruction loss of the input
- also optimize the latent distribution to be standard normal



Reduction via NN: Results

Dataset: 60,000 images of handwritten digits (MINST)

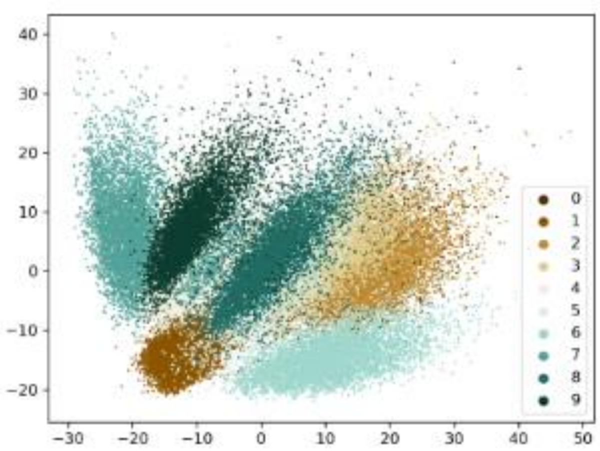
- each image is 28 × 28 → 784 D space



PCA projection of its 4D latent space

Reduction via NN: Results

Result when not assuring a standard normal distribution in the latent space



$$l_i(\theta, \phi) = -\mathbb{E}_{z \sim q_\theta(z \mid x_i)}[\log p_\phi(x_i \mid z)] + \mathbb{KL}(q_\theta(z \mid x_i) \mid\mid p(z))$$

Reconstruction loss

Kullback-Leibler divergence

Interpolation in Latent Space

What's the advantage of it?

- latent space allows easy interpolation
- move between samples in latent space and reconstruct novel instances by the decoder
- not easily possible using other non-linear layouts like MDS, T-SNE

See example here

Another application: :Deep clustering

- provides a convenient dimension reduction for k-means and other clustering algorithms
- linearizes non-linear data manifolds in high-D space which often appear in computer vision tasks

Cluster Analysis and Embedding of Categorical data

Text Processing

Let's look at application in text processing

Assume you are given a large corpus of documents and you wish to get an overview about what they contain

What can you do?

Singular Value Decomposition (SVD)

The same as PCA when the mean of each attribute is zero

SVD does not subtract the mean

- appropriate if values close to zero should not be influential
- PCA puts them at in the extreme negative side

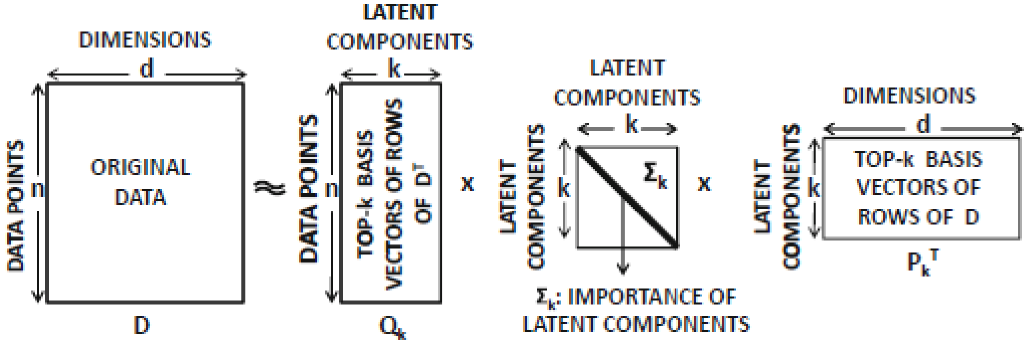
SVD often used for text analysis

- values close to zero are frequent and should not affect the analysis

Singular Value Decomposition (SVD)

Decomposes *C* into the matrix:

$$Q_k \Sigma_k P_k^T$$



q_i and p_i are two column vectors with significance σ_i

$$Q_k \Sigma_k P_k^T = \sum_{i=1}^{k} \overline{q_i} \sigma_i \overline{p_i}^T = \sum_{i=1}^{k} \sigma_i (\overline{q_i}\ \overline{p_i}^T)$$

Example: in a user-item ratings matrix we wish to determine:

- a reduced representation of the users
- a reduced representation of the items
- *SVD* has the basis vectors for both of these reductions

SVD Computation

Find the matrices **U**, **D,** and **V** such that:

$$\mathbf{C} = \mathbf{U}\ \mathbf{D}\ \mathbf{V}^T$$

U are the Eigenvectors of $\mathbf{CC}^T$
V are the Eigenvectors of $\mathbf{C}^T\mathbf{C}$
D a diagonal matrix of $\sqrt{\lambda_k}$ where λ^k are Eigenvalues of $\mathbf{CC}^T$
k=Rank(**C**)<Min(r-1,c-1)

Latent Semantic Analysis

Create an occurrence matrix (term-document matrix)

- words (terms *t*) are the rows
- paragraphs (documents *d*) are the columns
- uses the *term frequency–inverse document frequency (tf-idf)* metric
- $tf(t,d)$ = simplest form is frequency of t in d = $f(t,d)$

Index Words	Titles								
	T1	T2	T3	T4	T5	T6	T7	T8	T9
book			1	1					
dads						1			1
dummies		1						1	
estate							1		1
guide	1					1			
investing	1	1	1	1	1	1	1	1	1
market	1		1						
real							1		1
rich						2			1
stock	1		1					1	
value				1	1				

Latent Semantic Analysis

Create an occurrence matrix (term-document matrix)

- words (terms *t*) are the rows
- paragraphs (documents *d*) are the columns
- uses the *term frequency–inverse document frequency (tf-idf)* metric
- *tf(t,d)* = simplest form is frequency of *t* in *d* = *f(t,d)*
- idf(*t*,*d*) $\mathrm{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$
- *N* = number of docs = |*D*| , *D* is the corpus of documents
- idf is a measure of term rareness, it's 0 when term occurs in all of *D*
- important terms get a higher tf-idf

Use SVD to reduce the number of rows

- preserves similarity of columns

Co-Occurrence TF-IDT matrix

A

$$
\begin{array}{cc}
M & \begin{array}{cccccccc} D_1 & D_2 & D_3 & D_4 & D_5 & D_6 & \cdots & D_n \end{array} \\
\begin{array}{c} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \\ \vdots \\ T_m \end{array} &
\begin{pmatrix}
0.00060 & 0.00012 & 0.00003 & 0.00003 & 0.00333 & 0.00048 & \cdots & a_{1n} \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & a_{2n} \\
0 & 2.98862 & 0 & 0 & 0 & 1.49431 & \cdots & a_{3n} \\
0 & 0 & 0 & 13.32555 & 0 & 0 & \cdots & a_{4n} \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & a_{5n} \\
1.03442 & 1.03442 & 0 & 0 & 0 & 3.10326 & \cdots & a_{6n} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
a_{m1} & a_{m2} & a_{m3} & a_{m4} & a_{m5} & a_{m6} & \cdots & a_{mn}
\end{pmatrix}
\end{array}
$$

A

$$
\begin{array}{c|c}
\mathrm{M} & \begin{array}{cccccccc} D_1 & D_2 & D_3 & D_4 & D_5 & D_6 & \cdots & D_\mathrm{n} \end{array} \\
\begin{array}{c} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \\ \vdots \\ T_\mathrm{m} \end{array} &
\begin{pmatrix}
0.00060 & 0.00012 & 0.00003 & 0.00003 & 0.00333 & 0.00048 & \cdots & a_{1n} \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & a_{2n} \\
0 & 2.98862 & 0 & 0 & 0 & 1.49431 & \cdots & a_{3n} \\
0 & 0 & 0 & 13.32555 & 0 & 0 & \cdots & a_{4n} \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & a_{5n} \\
1.03442 & 1.03442 & 0 & 0 & 0 & 3.10326 & \cdots & a_{6n} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
a_{m1} & a_{m2} & a_{m3} & a_{m4} & a_{m5} & a_{m6} & \cdots & a_{mn}
\end{pmatrix}
\end{array}
$$

B

U = term-concept matrix

concept = latent (hidden) *topic*

U_k

$$
\mathrm{U} =
\begin{array}{c|c}
 & \begin{array}{ccccc} C_1 & C_2 & C_3 & \cdots & C_m \end{array} \\
\begin{array}{c} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \\ \vdots \\ T_m \end{array} &
\begin{pmatrix}
a_{11} & a_{12} & a_{13} & \cdots & a_{1m} \\
a_{21} & a_{22} & a_{23} & \cdots & a_{2m} \\
a_{31} & a_{32} & a_{33} & \cdots & a_{3m} \\
a_{41} & a_{42} & a_{43} & \cdots & a_{4m} \\
a_{51} & a_{52} & a_{53} & \cdots & a_{5m} \\
a_{61} & a_{62} & a_{63} & \cdots & a_{6m} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mm}
\end{pmatrix}
\end{array}
$$

sort and keep the k
most significant rows/columns

Σ_k

$$
\Sigma =
\begin{array}{c|c}
 & \begin{array}{ccccc} D_1 & D_2 & D_3 & \cdots & D_n \end{array} \\
\begin{array}{c} T_1 \\ T_2 \\ T_3 \\ T_4 \\ \vdots \\ T_m \end{array} &
\begin{pmatrix}
a_{11} & 0 & 0 & \cdots & 0 \\
0 & a_{22} & 0 & \cdots & 0 \\
0 & 0 & a_{33} & \cdots & 0 \\
0 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & a_{mm}
\end{pmatrix}
\end{array}
$$

V = concept-document matrix

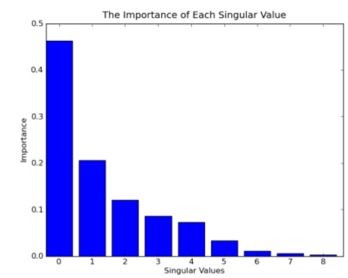
V_k^T

$$
\mathrm{V}^T =
\begin{array}{c|c}
 & \begin{array}{ccccc} D_1 & D_2 & D_3 & \cdots & D_n \end{array} \\
\begin{array}{c} C_1 \\ C_2 \\ C_3 \\ C_4 \\ \vdots \\ C_n \end{array} &
\begin{pmatrix}
a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\
a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\
a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\
a_{41} & a_{42} & a_{43} & \cdots & a_{4n} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn}
\end{pmatrix}
\end{array}
$$

Visualizing The Concept Space

How many concepts to use when approximating the matrix?

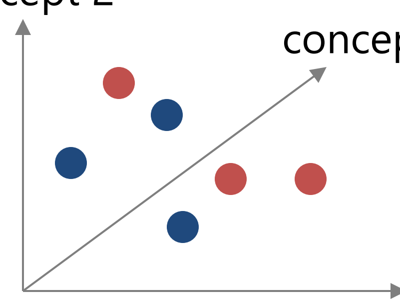
- if too few, important patterns are left out
- if too many, noise caused by random word choices will creep in
- can use the elbow method in the scree plot



Throw out the 1st dimension in U and V

- in U it is correlated with document length
- in V it correlates with the number of times a term was mentioned

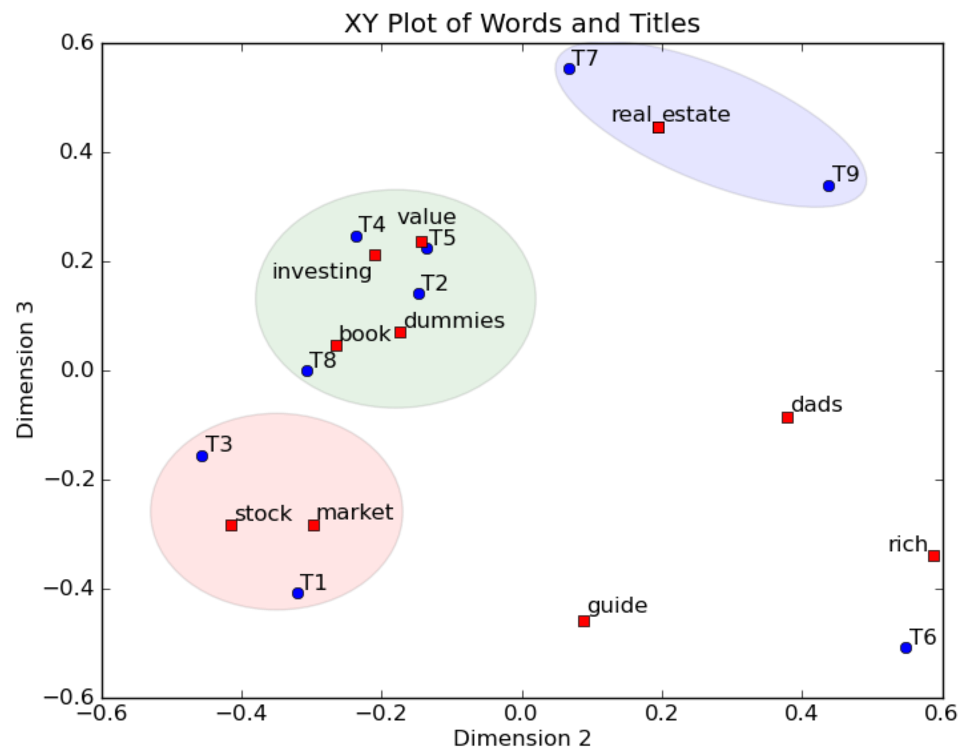
Now we have a k-D concept space shared by both terms and documents



Visualizing The Concept Space

Project the k-D concept space into 2D and visualize as a map

- can cluster the map
- the cluster of documents are then labeled by the terms
- provides map semantics



LSA Disadvantages

LSA assumes a Gaussian distribution and Frobenius norm

- this may not fit all problems

LSA cannot handle polysemy effectively

- need LDA (Latent Dirichlet Allocation) for this

LSA depends heavily on SVD

- computationally intensive
- hard to update as new documents appear
- but faster algorithms have emerged recently

What About Categorical Variables?

You will need to use correspondence analysis (CA)

- CA is PCA for categorical variables
- related to factor analysis

Makes use of the χ^2 test

- what's χ^2 ?

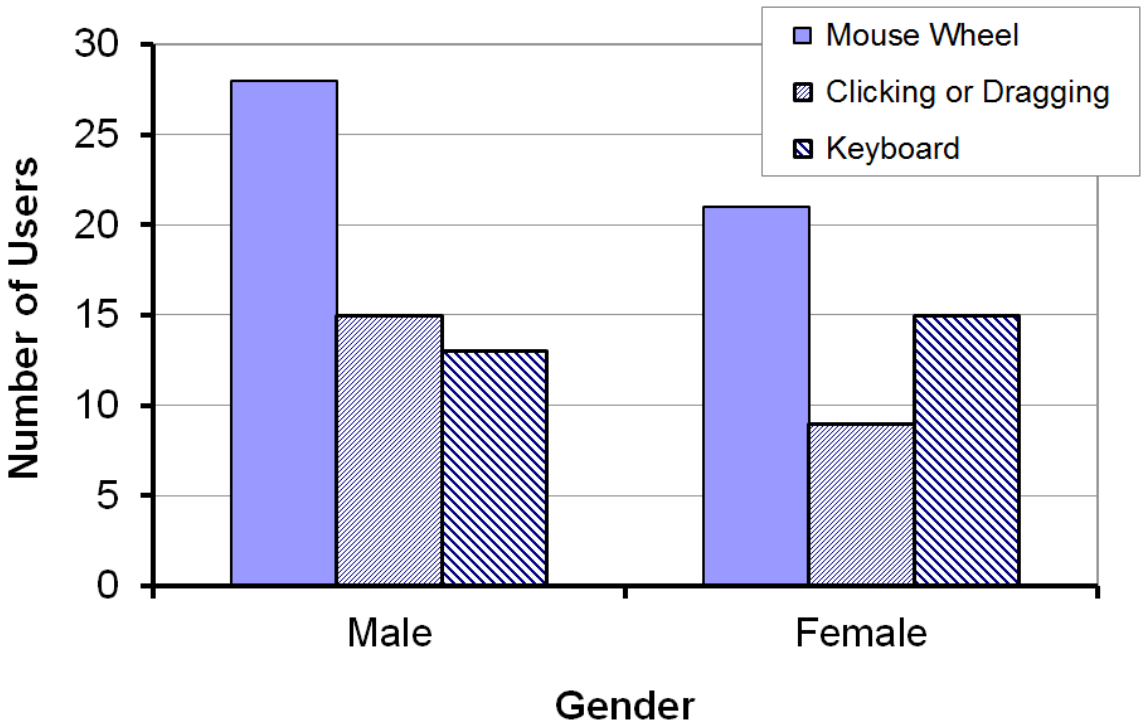
Chi-square Test (Nominal Data)

- A *chi-square test* is used to investigate relationships
- Relationships between categorical, or nominal-scale, variables representing attributes of people, interaction techniques, systems, etc.
- Data organized in a *contingency table* – cross tabulation containing counts (frequency data) for number of observations in each category
- A chi-square test compares the *observed values* against *expected values*
- Expected values assume “no difference”
- Research question:
  – *Do males and females differ in their method of scrolling on desktop systems?* (next slide)

Chi-square – Example #1

Observed Number of Users				
Gender	Scrolling Method			Total
	MW	CD	KB	
Male	28	15	13	56
Female	21	9	15	45
Total	49	24	28	101

MW = mouse wheel
CD = clicking, dragging
KB = keyboard



Chi-square – Example #1

56.0·49.0/101=27.2

Expected Number of Users				
Gender	Scrolling Method			Total
	MW	CD	KB	
Male	27.2	13.3	15.5	56.0
Female	21.8	10.7	12.5	45.0
Total	49.0	24.0	28.0	101

(Expected-Observed)^2/Expected = (28-27.2)^2/27.2

Chi Squares				
Gender	Scrolling Method			Total
	MW	CD	KB	
Male	0.025	0.215	0.411	0.651
Female	0.032	0.268	0.511	0.811
Total	0.057	0.483	0.922	**1.462**

Significant if it exceeds critical value (next slide)

$\chi^2 = 1.462$

Chi-square Critical Values

- Decide in advance on *alpha* (typically .05)
- Degrees of freedom
  – $df = (r - 1)(c - 1) = (2 - 1)(3 - 1) = 2$
  – r = number of rows, c = number of columns

Significance Threshold (α)	Degrees of Freedom							
	1	2	3	4	5	6	7	8
.1	2.71	4.61	6.25	7.78	9.24	10.65	12.02	13.36
.05	3.84	$\boxed{5.99}$	7.82	9.49	11.07	12.59	14.07	15.51
.01	6.64	9.21	11.35	13.28	15.09	16.81	18.48	20.09
.001	10.83	13.82	16.27	18.47	20.52	22.46	24.32	26.13

$\chi^2 = 1.462$ (< 5.99 $\therefore$ not significant)

Correspondence Analysis (CA)

[more info]

Example:

	Smoking Category				
Staff Group	(1) None	(2) Light	(3) Medium	(4) Heavy	Row Totals
(1) Senior Managers	4	2	3	2	11
(2) Junior Managers	4	3	7	4	18
(3) Senior Employees	25	10	12	4	51
(4) Junior Employees	18	24	33	13	88
(5) Secretaries	10	6	7	2	25
Column Totals	61	45	62	25	193

There are two high-D spaces

- 4D (column) space spanned by smoking habits – plot staff group
- 5D (row) space spanned by staff group – plot smoking habits

Are these two spaces (the rows and columns) independent ?

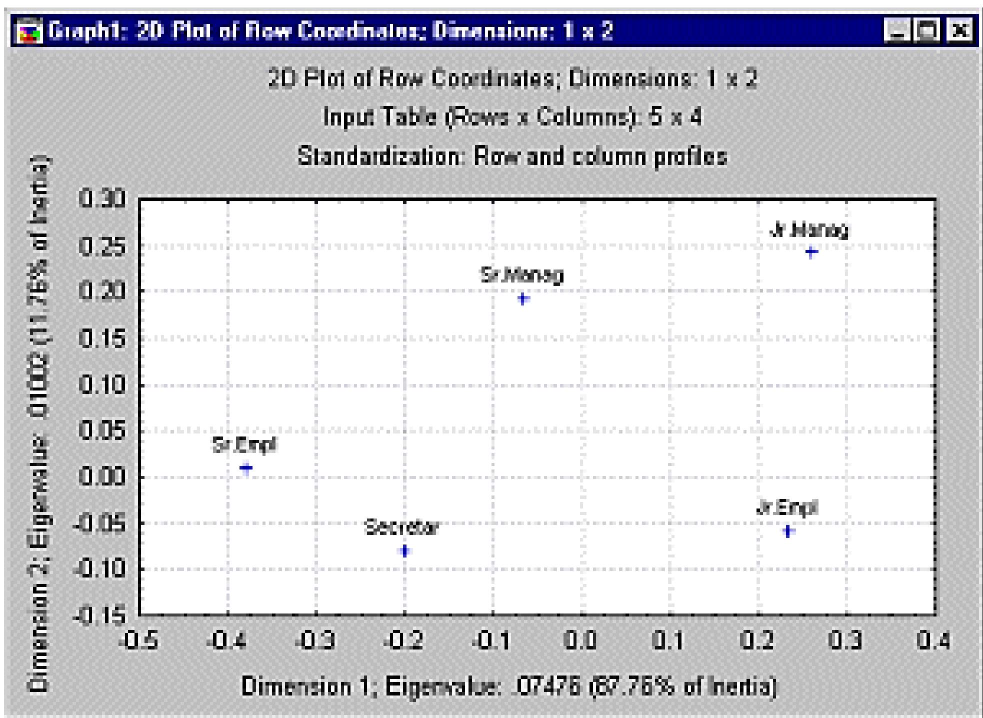
- this occurs when the χ^2 statistics of the table is insignificant

CA Eigen Analysis

Staff Group	Smoking Category (1) None	(2) Light	(3) Medium	(4) Heavy	Row Totals
(1) Senior Managers	4	2	3	2	11
(2) Junior Managers	4	3	7	4	18
(3) Senior Employees	25	10	12	4	51
(4) Junior Employees	18	24	33	13	88
(5) Secretaries	10	6	7	2	25
Column Totals	61	45	62	25	193

Let's do some plotting

- compute distance matrix of the rows $\mathbf{C}\mathbf{C}^T$
- compute Eigenvector matrix **U** and the Eigenvalue matrix **D**
- sort eigenvectors by values, pick two major vectors, create 2D plot



-- senior employees most similar to secretaries

Eigenvalues and Inertia for all Dimensions
Input Table (Rows x Columns): 5 x 4
Total Inertia = .08519 Chi² = 16.442

No. of Dims	Singular Values	Eigen-Values	Perc. of Inertia	Cumulatv Percent	Chi Squares
1	.273421	.074759	87.75587	87.7559	14.42851
2	.100086	.010017	11.75865	99.5145	1.93332
3	.020337	.000414	.48547	100.0000	.07982

CA Eigen Analysis

	Smoking Category				
Staff Group	(1) None	(2) Light	(3) Medium	(4) Heavy	Row Totals
(1) Senior Managers	4	2	3	2	11
(2) Junior Managers	4	3	7	4	18
(3) Senior Employees	25	10	12	4	51
(4) Junior Employees	18	24	33	13	88
(5) Secretaries	10	6	7	2	25
Column Totals	61	45	62	25	193

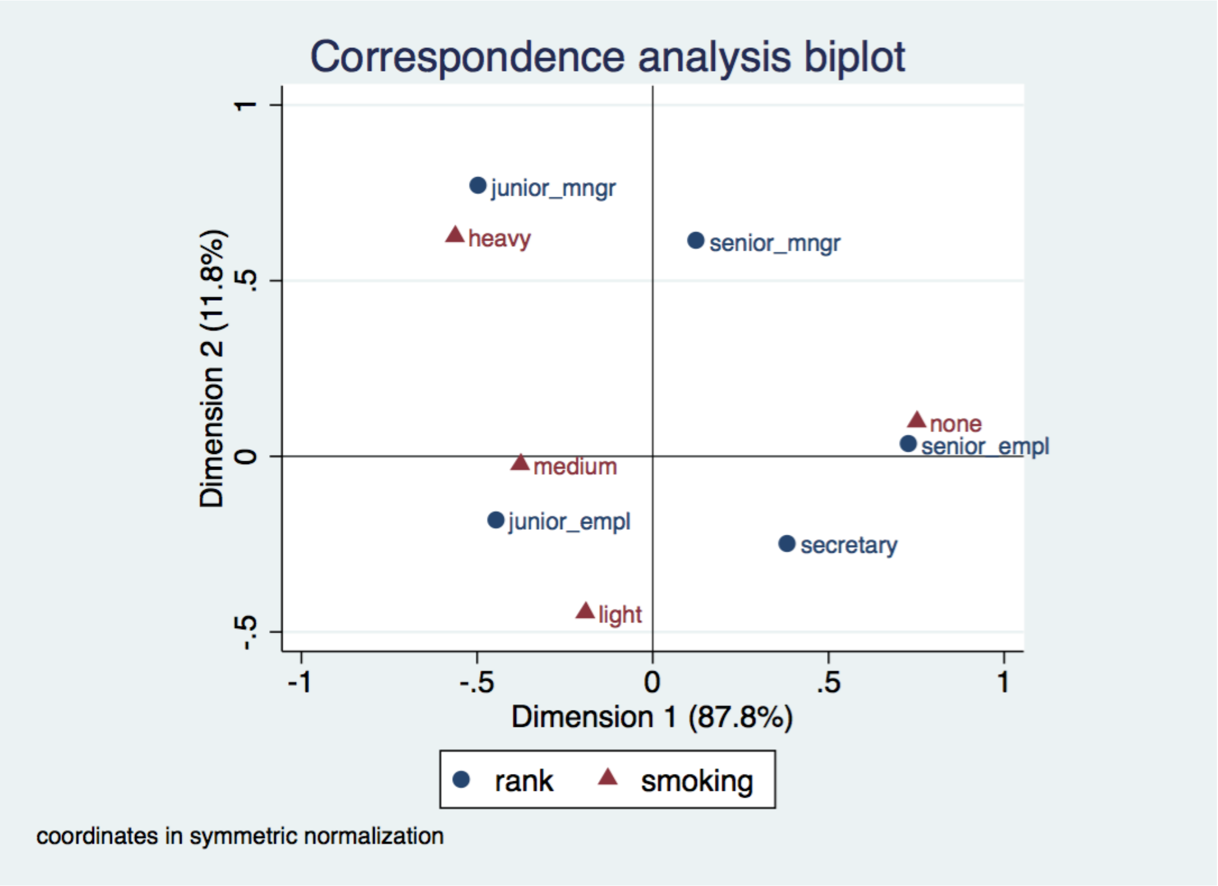
Next:

- compute distance matrix of the columns $\mathbf{C}^T\mathbf{C}$
- compute Eigenvector matrix **V** (gives the same Eigenvalue matrix **D**)
- sort eigenvectors by value
- pick two major vectors
- create 2D plot of smoking categories

Following (next slide):

- combine the plots of **U** and **V**
- if the χ^2 statistics was significant we should see some dependencies

Combined CA Plot



Interpretation sample (using the χ^2 frequentist mindset)

- *relatively speaking,* there are more non-smoking senior employees

Extending to Cases

Case Number	Senior Manager	Junior Manager	Senior Employee	Junior Employee	Secretary	None	Light	Medium	Heavy
1	1	0	0	0	0	1	0	0	0
2	1	0	0	0	0	1	0	0	0
3	1	0	0	0	0	1	0	0	0
4	1	0	0	0	0	1	0	0	0
5	1	0	0	0	0	0	1	0	0
...	.	.	.	.	.	.	.	.	.
...	.	.	.	.	.	.	.	.	.
...	.	.	.	.	.	.	.	.	.
191	0	0	0	0	1	0	0	1	0
192	0	0	0	0	1	0	0	0	1
193	0	0	0	0	1	0	0	0	1

Plot would now show 193 cases and 9 variables

Multiple Correspondence Analysis

Extension where there are more than 2 categorical variables

	SURVIVAL		AGE			LOCATION		
Case No.	NO	YES	LESST50	A50TO69	OVER69	TOKYO	BOSTON	GLAMORGN
1	0	1	0	1	0	0	0	1
2	1	0	1	0	0	1	0	0
3	0	1	0	1	0	0	1	0
4	0	1	0	0	1	0	0	1
...	.	.	.	.	.	.	.	.
...	.	.	.	.	.	.	.	.
...	.	.	.	.	.	.	.	.
762	1	0	0	1	0	1	0	0
763	0	1	1	0	0	0	1	0
764	0	1	0	1	0	0	0	1

Let's call it matrix X

Multiple Correspondence Analysis

Compute X'X to get the Burt Table

	SURVIVAL		AGE			LOCATION		
	NO	YES	<50	50-69	69+	TOKYO	BOSTON	GLAMORGN
SURVIVAL:NO	210	0	68	93	49	60	82	68
SURVIVAL:YES	0	554	212	258	84	230	171	153
AGE:UNDER_50	68	212	280	0	0	151	58	71
AGE:A_50TO69	93	258	0	351	0	120	122	109
AGE:OVER_69	49	84	0	0	133	19	73	41
LOCATION:TOKYO	60	230	151	120	19	290	0	0
LOCATION:BOSTON	82	171	58	122	73	0	253	0
LOCATION:GLAMORGN	68	153	71	109	41	0	0	221

Compute Eigenvectors and Eigenvalues

- keep top two Eigenvectors/values
- visualize the attribute loadings of these two Eigenvectors into the Burt table plot (the loadings are the coordinates)

Larger MCA Example

Results of a survey of car owners and car attributes

Burt Table

	American	European	Japanese	Large	Medium	Small	Family	Sporty	Work	1 Income	2 Incomes	Own	Rent	Married	Married with Kids	Single	Single with Kids	Female	Male
American	125	0	0	36	60	29	81	24	20	58	67	93	32	37	50	32	6	58	67
European	0	44	0	4	20	20	17	23	4	18	26	38	6	13	15	15	1	21	23
Japanese	0	0	165	2	61	102	76	59	30	74	91	111	54	51	44	62	8	70	95
Large	36	4	2	42	0	0	30	1	11	20	22	35	7	9	21	11	1	17	25
Medium	60	20	61	0	141	0	89	39	13	57	84	106	35	42	51	40	8	70	71
Small	29	20	102	0	0	151	55	66	30	73	78	101	50	50	37	58	6	62	89
Family	81	17	76	30	89	55	174	0	0	69	105	130	44	50	79	35	10	83	91
Sporty	24	23	59	1	39	66	0	106	0	55	51	71	35	35	12	57	2	44	62
Work	20	4	30	11	13	30	0	0	54	26	28	41	13	16	18	17	3	22	32
1 Income	58	18	74	20	57	73	69	55	26	150	0	80	70	10	27	99	14	47	103
2 Incomes	67	26	91	22	84	78	105	51	28	0	184	162	22	91	82	10	1	102	82
Own	93	38	111	35	106	101	130	71	41	80	162	242	0	76	106	52	8	114	128
Rent	32	6	54	7	35	50	44	35	13	70	22	0	92	25	3	57	7	35	57
Married	37	13	51	9	42	50	50	35	16	10	91	76	25	101	0	0	0	53	48
Married with Kids	50	15	44	21	51	37	79	12	18	27	82	106	3	0	109	0	0	48	61
Single	32	15	62	11	40	58	35	57	17	99	10	52	57	0	0	109	0	35	74
Single with Kids	6	1	8	1	8	6	10	2	3	14	1	8	7	0	0	0	15	13	2
Female	58	21	70	17	70	62	83	44	22	47	102	114	35	53	48	35	13	149	0
Male	67	23	95	25	71	89	91	62	32	103	82	128	57	48	61	74	2	0	185

more info see here

MCA Example (2)

Summary table:

Inertia and Chi-Square Decomposition					
Singular Value	Principal Inertia	Chi-Square	Percent	Cumulative Percent	4 8 12 16 20 ----+----+----+----+----+---
0.56934	0.32415	970.77	18.91	18.91	*******************
0.48352	0.23380	700.17	13.64	32.55	**************
0.42716	0.18247	546.45	10.64	43.19	***********
0.41215	0.16987	508.73	9.91	53.10	**********
0.38773	0.15033	450.22	8.77	61.87	*********
0.38520	0.14838	444.35	8.66	70.52	*********
0.34066	0.11605	347.55	6.77	77.29	*******
0.32983	0.10879	325.79	6.35	83.64	*******
0.31517	0.09933	297.47	5.79	89.43	******
0.28069	0.07879	235.95	4.60	94.03	*****
0.26115	0.06820	204.24	3.98	98.01	****
0.18477	0.03414	102.24	1.99	100.00	**
Total	1.71429	5133.92	100.00		

Degrees of Freedom = 324

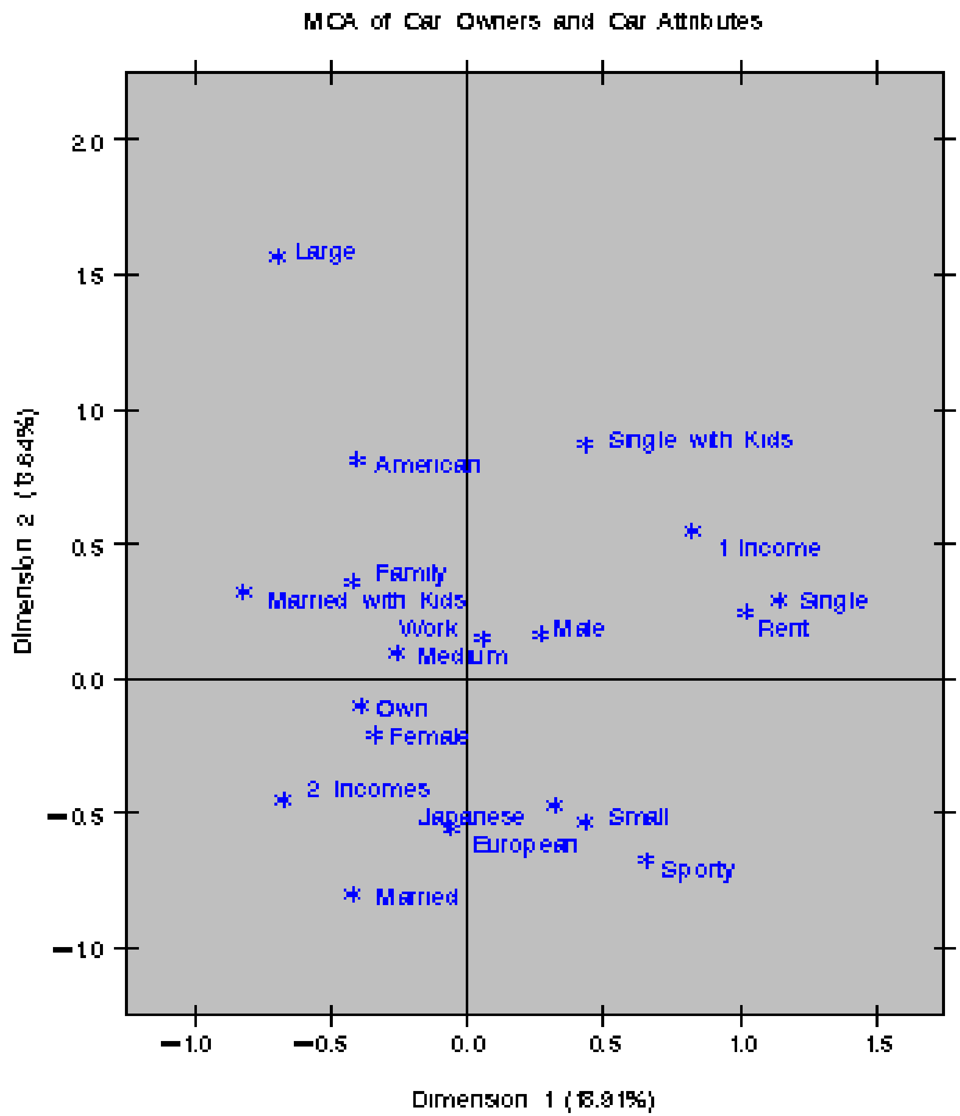
MCA Example (3)

Most influential column points (loadings):

Column Coordinates		
	Dim1	Dim2
American	-0.4035	0.8129
European	-0.0568	-0.5552
Japanese	0.3208	-0.4678
Large	-0.6949	1.5666
Medium	-0.2562	0.0965
Small	0.4326	-0.5258
Family	-0.4201	0.3602
Sporty	0.6604	-0.6696
Work	0.0575	0.1539
1 Income	0.8251	0.5472
2 Incomes	-0.6727	-0.4461
Own	-0.3887	-0.0943
Rent	1.0225	0.2480
Married	-0.4169	-0.7954
Married with Kids	-0.8200	0.3237
Single	1.1461	0.2930
Single with Kids	0.4373	0.8736
Female	-0.3365	-0.2057
Male	0.2710	0.1656

MCA Example (4)

Burt table plot:



Plot Observations

Top-right quadrant:

- categories single, single with kids, 1 income, and renting a home are associated

Proceeding clockwise:

- the categories sporty, small, and Japanese are associated
- being married, owning your own home, and having two incomes are associated
- having children is associated with owning a large American family car

Such information could be used in market research to identify target audiences for advertisements

Gartner Magic Quadrant

A Gartner Magic Quadrant is a culmination of research in a specific market, providing a wide-angle view of the relative positions of the market's competitors

This concept can be used for other dimension pairs as well

- essentially require to think of a segmentation of the 4 quadrants

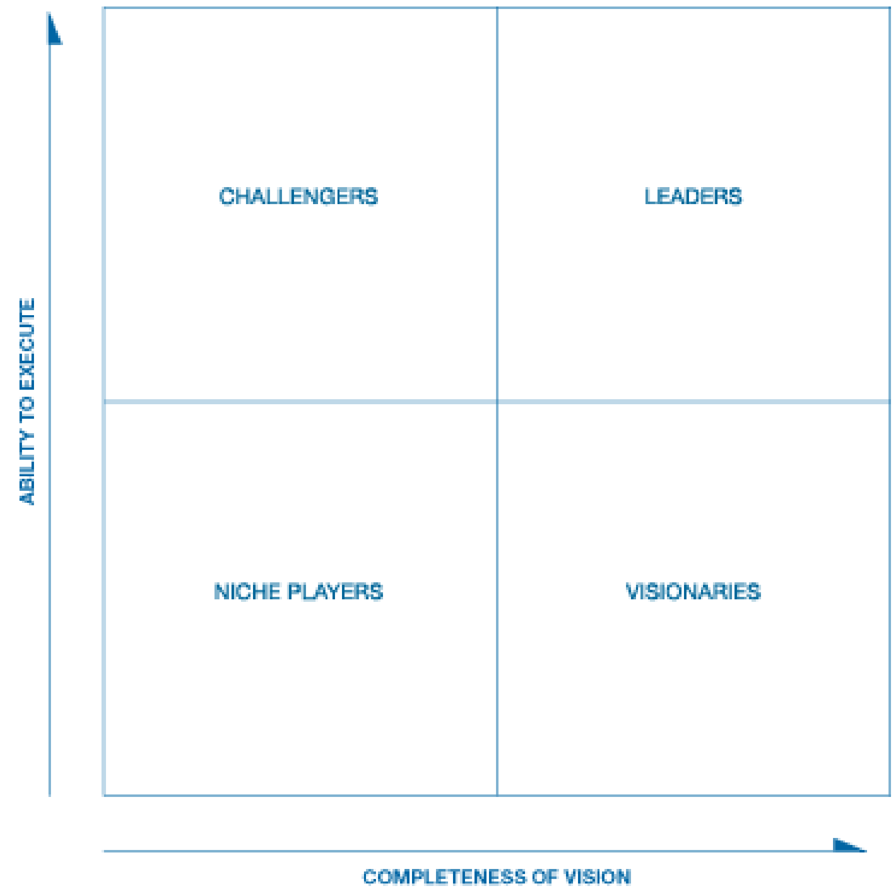
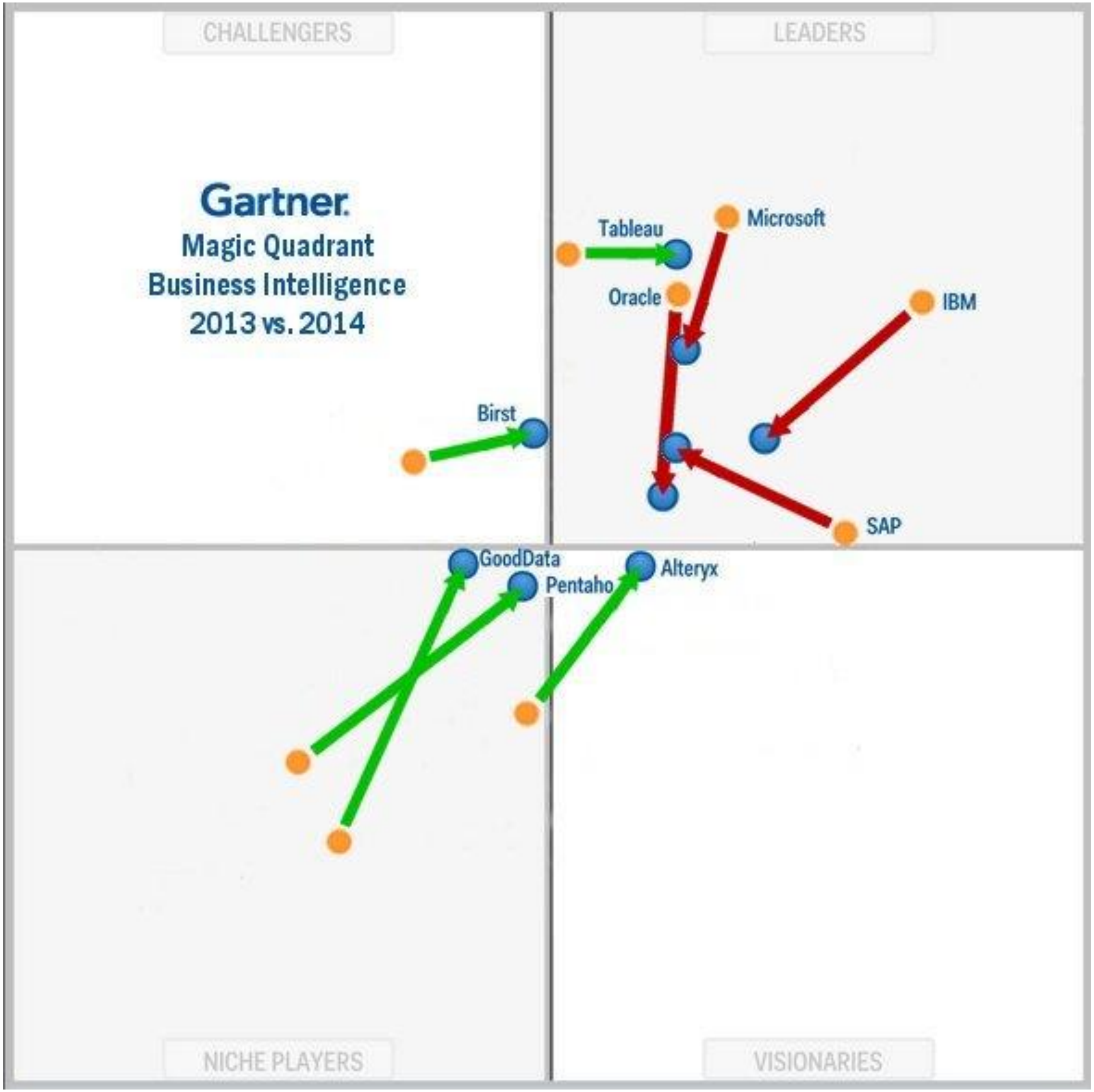


Figure 1. Magic Quadrant for Business Intelligence and Analytics Platforms

Source: Gartner (February 2014)


CHALLENGERS
LEADERS
Gartner
Magic Quadrant
Business Intelligence
2013 vs. 2014
Tableau
Microsoft
Oracle
IBM
Birst
SAP
GoodData
Pentaho
Alteryx
NICHE PLAYERS
VISIONARIES