What is Cluster Analysis?

- Cluster: a collection of data objects
  - Similar to one another within the same cluster
  - Dissimilar to the objects in other clusters
- Cluster analysis
  - Grouping a set of data objects into clusters
- Clustering is unsupervised classification: no predefined classes
- Typical applications
  - As a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms

Examples of Clustering Applications

- Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- Land use: Identification of areas of similar land use in an earth observation database
- Insurance: Identifying groups of motor insurance policy holders with a high average claim cost
- City-planning: Identifying groups of houses according to their house type, value, and geographical location
- Earth-quake studies: Observed earth quake epicenters should be clustered along continent faults

What Is Good Clustering?

- A good clustering method will produce high quality clusters with
  - high intra-class similarity
  - low inter-class similarity
- The quality of a clustering result depends on both the similarity measure used by the method and its implementation.
- The quality of a clustering method is also measured by its ability to discover some or all of the hidden patterns.

Measure the Quality of Clustering

- Dissimilarity/Similarity metric: Similarity is expressed in terms of a distance function, which is typically metric: $d(i, j)$
- There is a separate "quality" function that measures the "goodness" of a cluster.
- The definitions of distance functions are usually very different for interval-scaled, boolean, categorical, and ordinal variables.
- Weights should be associated with different variables based on applications and data semantics.
- It is hard to define "similar enough" or "good enough"
  - the answer is typically highly subjective.

Spoofing of the Sum of Squares Error Criterion

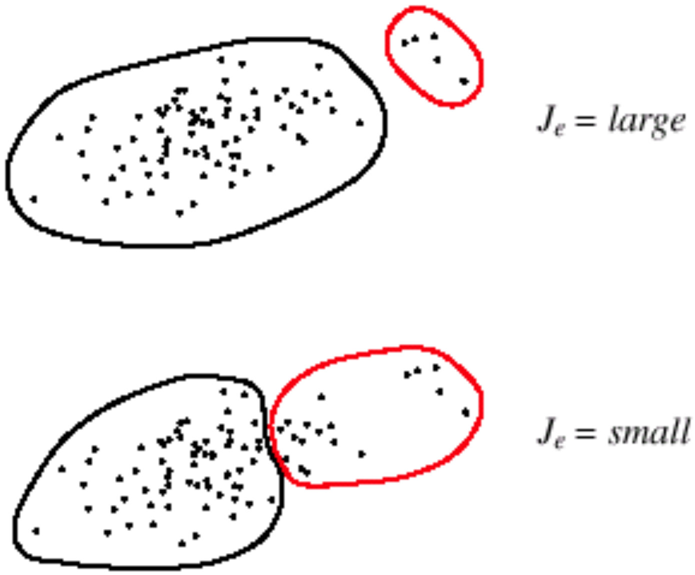


FIGURE 10.10. When two natural groupings have very different numbers of points, the clusters minimizing a sum-squared-error criterion J_e of Eq. 54 may not reveal the true underlying structure. Here the criterion is smaller for the two clusters at the bottom than for the more natural clustering at the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

Major Clustering Approaches

- Partitioning algorithms: Construct various partitions and then evaluate them by some criterion
- Hierarchy algorithms: Create a hierarchical decomposition of the set of data (or objects) using some criterion
- Density-based: based on connectivity and density functions
- Grid-based: based on a multiple-level granularity structure
- Model-based: A model is hypothesized for each of the clusters and the idea is to find the best fit of that model to each other

Partitioning Algorithms: Basic Concept

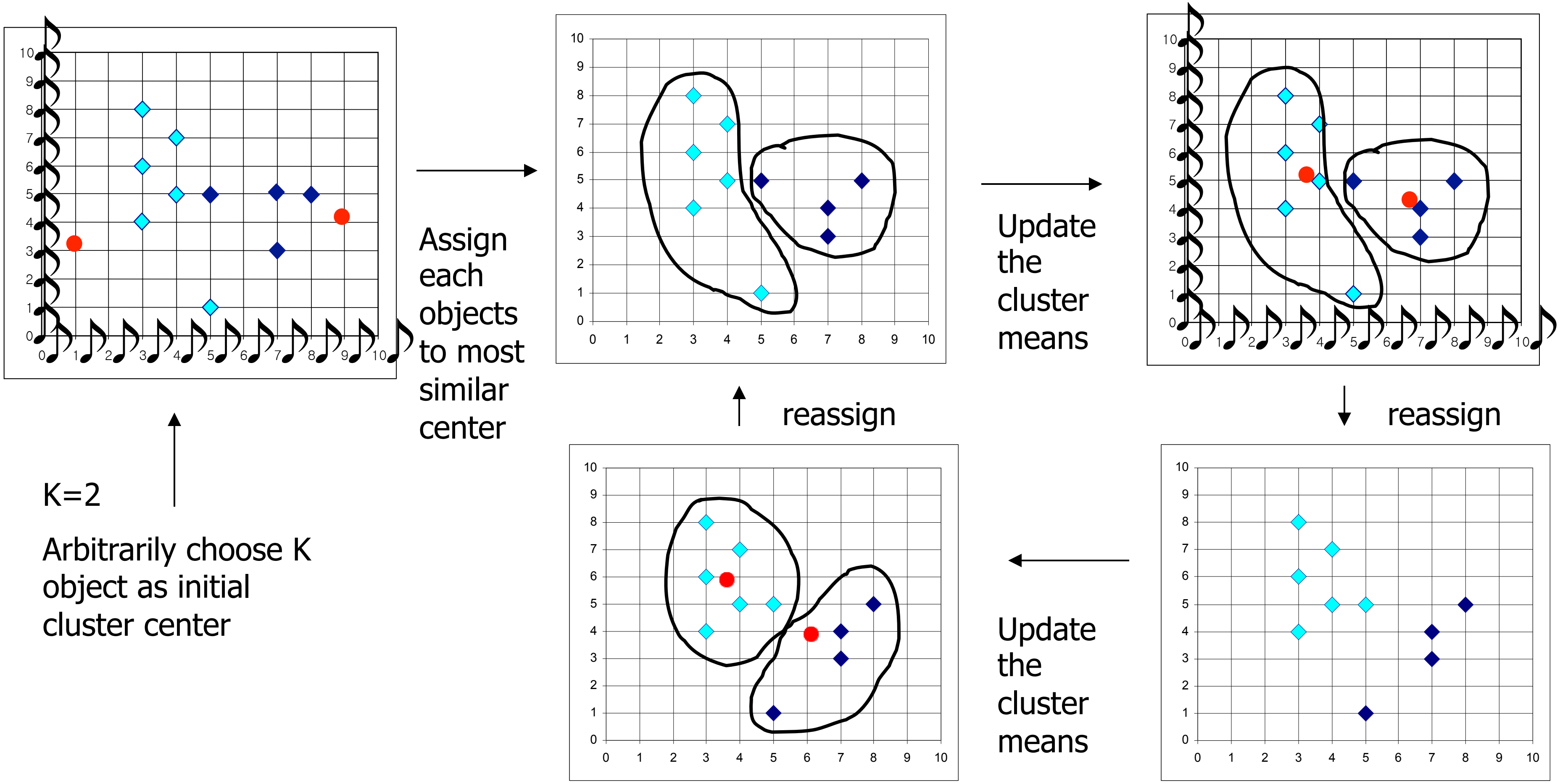
- Partitioning method: Construct a partition of a database *D* of *n* objects into a set of *k* clusters
- Given a *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion
  - Global optimal: exhaustively enumerate all partitions
  - Heuristic methods: *k-means* and *k-medoids* algorithms
  - *k-means* (MacQueen' 67): Each cluster is represented by the center of the cluster
  - *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw' 87): Each cluster is represented by one of the objects in the cluster

The K-Means Algorithm

```
for $k = 1, \ldots, K$ let $\mathbf{r}(k)$ be a randomly chosen point from $D$;
while changes in clusters $C_k$ happen do
    form clusters:
    for $k = 1, \ldots, K$ do
        $C_k = \{\mathbf{x} \in D \mid d(\mathbf{r}_k, \mathbf{x}) \leq d(\mathbf{r}_j, \mathbf{x})$ for all $j = 1, \ldots, K, j \neq k\}$
    end;
    compute new cluster centers:
    for $k = 1, \ldots, K$ do
        $\mathbf{r}_k =$ the vector mean of the points in $C_k$
    end;
end;
```

The *K-Means* Clustering Method

- Example



K=2

Arbitrarily choose K object as initial cluster center

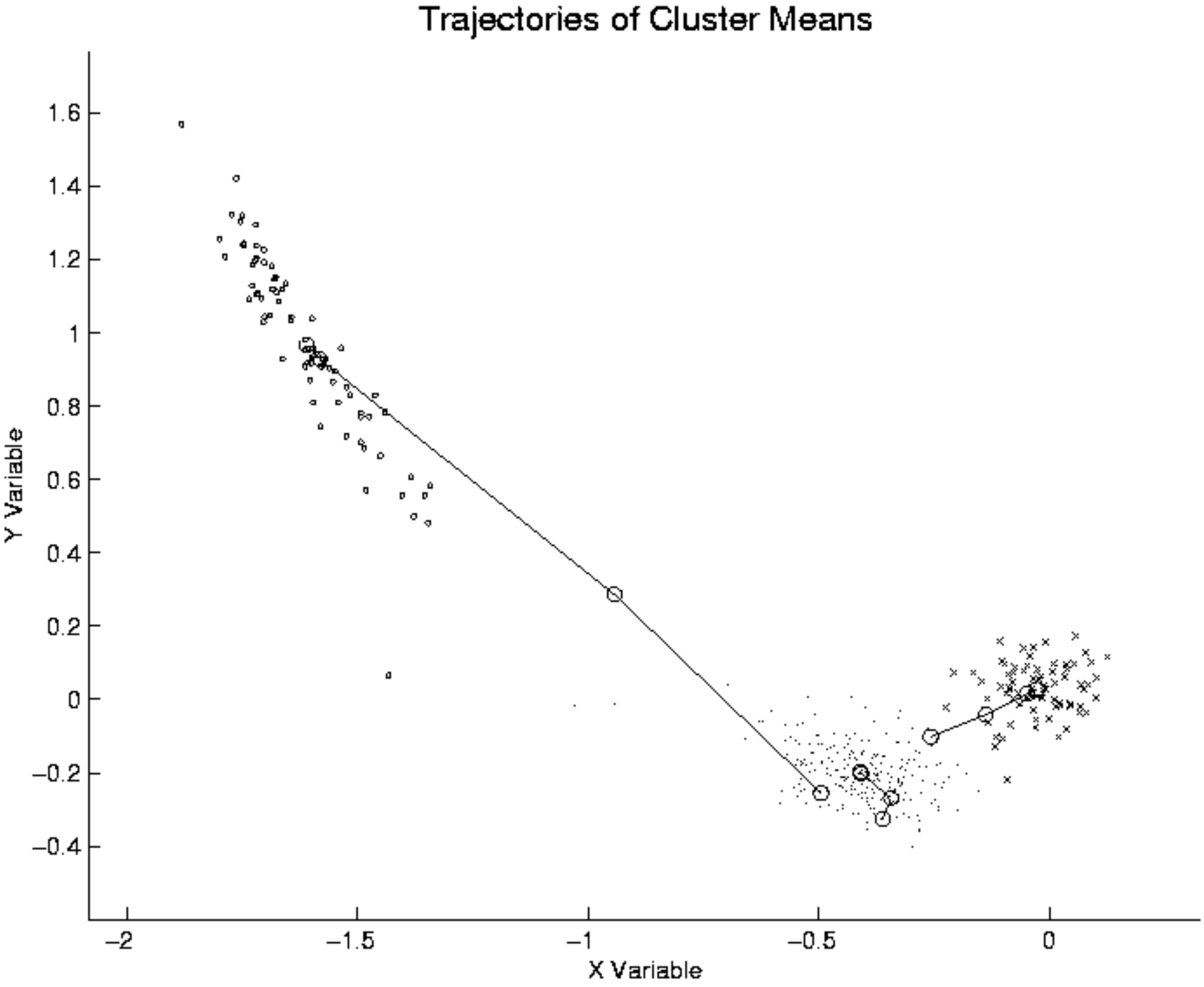
Assign each objects to most similar center

Update the cluster means

reassign

reassign

Update the cluster means


Trajectories of Cluster Means
Y Variable
X Variable
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−2
−1.5
−1
−0.5
0

Comments on the *K-Means* Method

- Strength: *Relatively efficient*: $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, k, $t << n$.
    - Comparing: PAM: $O(k(n-k)^2)$, CLARA: $O(ks^2 + k(n-k))$
- Comment: Often terminates at a *local optimum*. The *global optimum* may be found using techniques such as: *deterministic annealing* and *genetic algorithms*
- Weakness
    - Applicable only when *mean* is defined, then what about categorical data?
    - Need to specify *k,* the *number* of clusters, in advance
    - Unable to handle noisy data and *outliers*
    - Not suitable to discover clusters with *non-convex shapes*

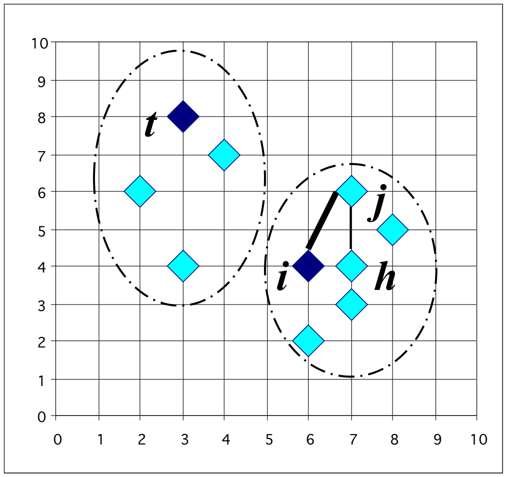
The *K-Medoids* Clustering Method

- Find *representative* objects, called medoids, in clusters
- *PAM* (Partitioning Around Medoids, 1987)
  - starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering
  - *PAM* works effectively for small data sets, but does not scale well for large data sets
- *CLARA* (Kaufmann & Rousseeuw, 1990)
- *CLARANS* (Ng & Han, 1994): Randomized sampling
- Focusing + spatial data structure (Ester et al., 1995)

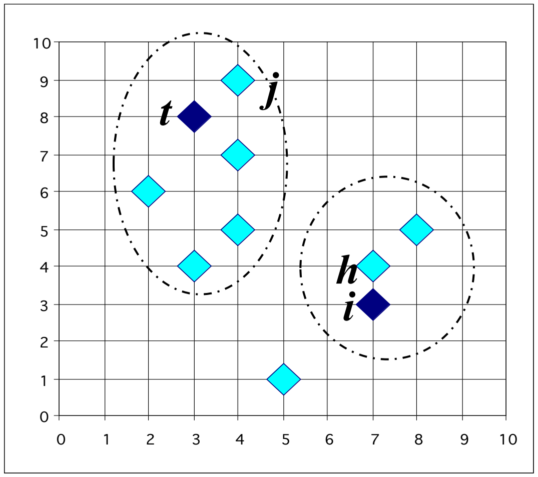
PAM (Partitioning Around Medoids) (1987)

- PAM (Kaufman and Rousseeuw, 1987), built in Splus
- Use real object to represent the cluster
  - Select *k* representative objects arbitrarily
  - For each pair of non-selected object *h* and selected object *i*, calculate the total swapping cost TC_{ih}
  - For each pair of *i* and *h*,
    - If $TC_{ih} < 0$, *i* is replaced by *h*
    - Then assign each non-selected object to the most similar representative object
  - repeat steps 2-3 until there is no change

PAM Clustering: Total swapping cost $TC_{ih}=\sum_j C_{jih}$

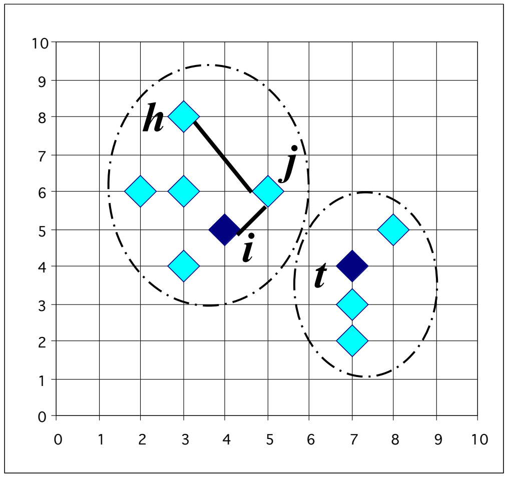


$C_{jih} = d(j, h) - d(j, i)$

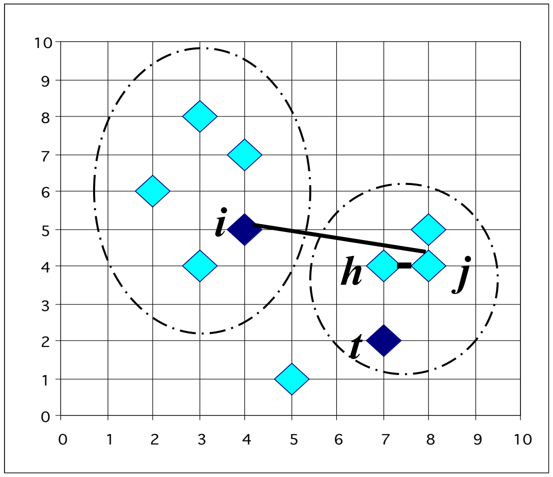


$C_{jih} = 0$

i & *t* are the current mediods



$C_{jih} = d(j, t) - d(j, i)$



$C_{jih} = d(j, h) - d(j, t)$

What is the problem with PAM?

- Pam is more robust than k-means in the presence of noise and outliers because a medoid is less influenced by outliers or other extreme values than a mean
- Pam works efficiently for small data sets but does not **scale well** for large data sets.
  - O(k(n-k)^2) for each iteration

    where n is # of data,k is # of clusters

➔ Sampling based method,

CLARA(Clustering LARge Applications)

K-Means Clustering in R

```
kmeans(x, centers, iter.max=10)

x  A numeric matrix of data, or an object that can be coerced
   to   such a matrix (such as a numeric vector or a data
   frame with  all numeric columns).

centers
   Either the number of clusters or a set of initial cluster
   centers. If the first, a random set of rows in x are chosen
   as the initial centers.

iter.max
```

Hartigan's Rule

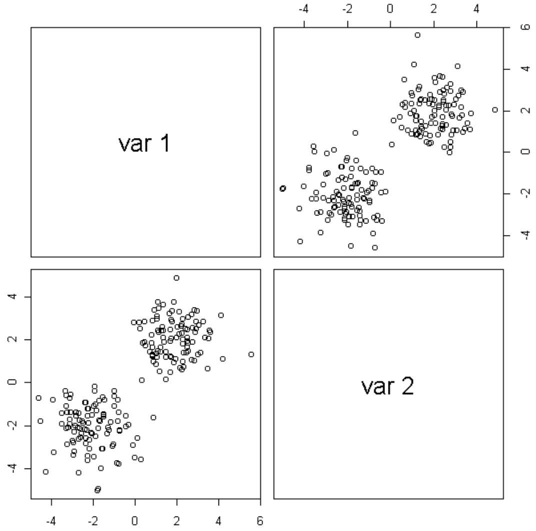
When deciding on the number of clusters, Hartigan (1975, pp 90-91) suggests the following rough rule of thumb. If *k* is the result of *k*-means with *k* groups and *k*plus1 is the result with *k*+1 groups, then it is justifiable to add the extra group when:

```
(sum(k$withinss)/sum(kplus1$withinss)-1)*(nrow(x)-k-1)
```

is greater than 10.

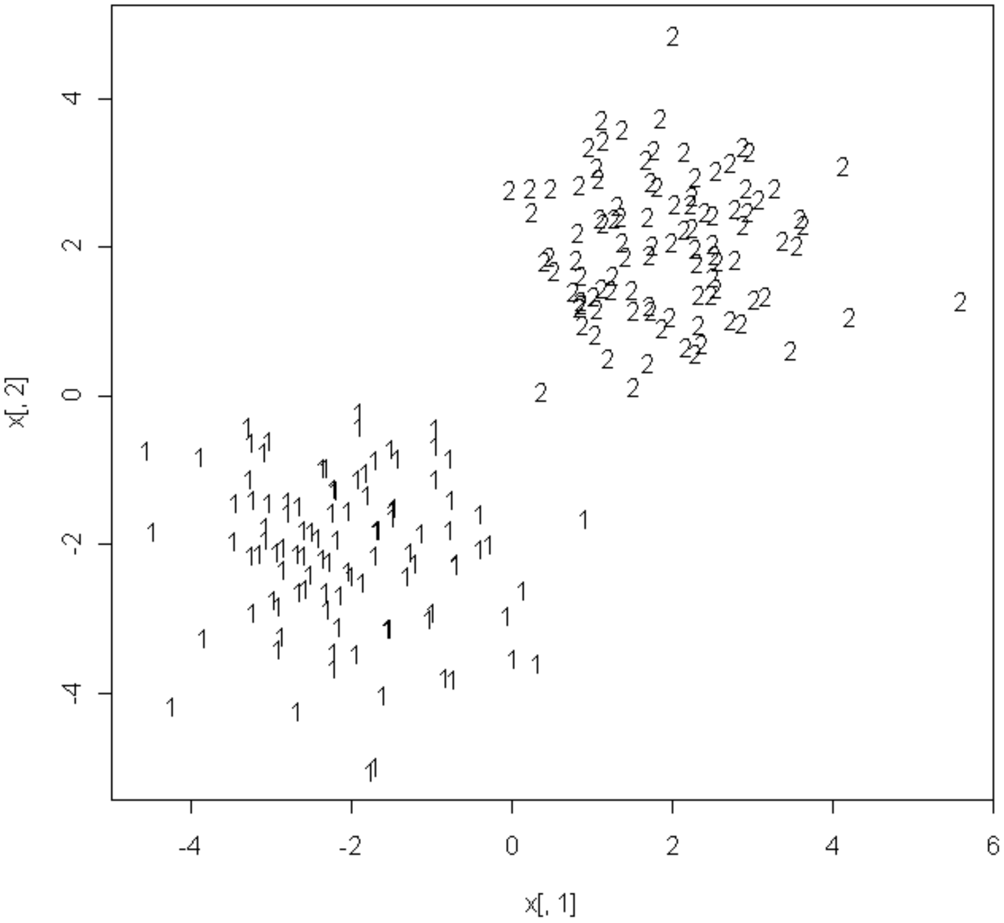
Example Data Generation

```
library(MASS)
x1<-mvrnorm(100, mu=c(2,2), Sigma=matrix(c(1,0,0,1),
   2))
x2<-mvrnorm(100, mu=c(-2,-2), Sigma=matrix(c(1,0,0,1),
   2))
x<-matrix(nrow=200,ncol=2)
x[1:100,]<-x1
x[101:200,]<-x2
pairs(x)
```

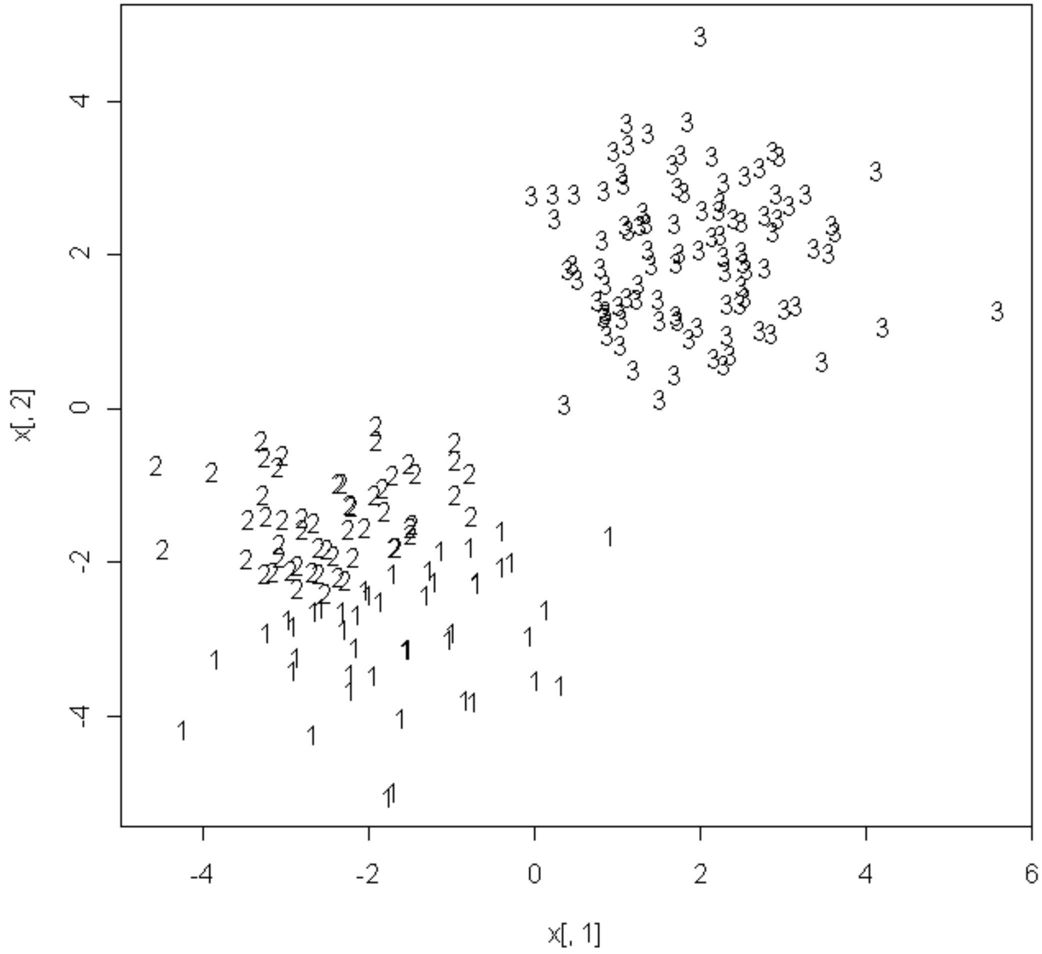


k-means Applied to our Data Set

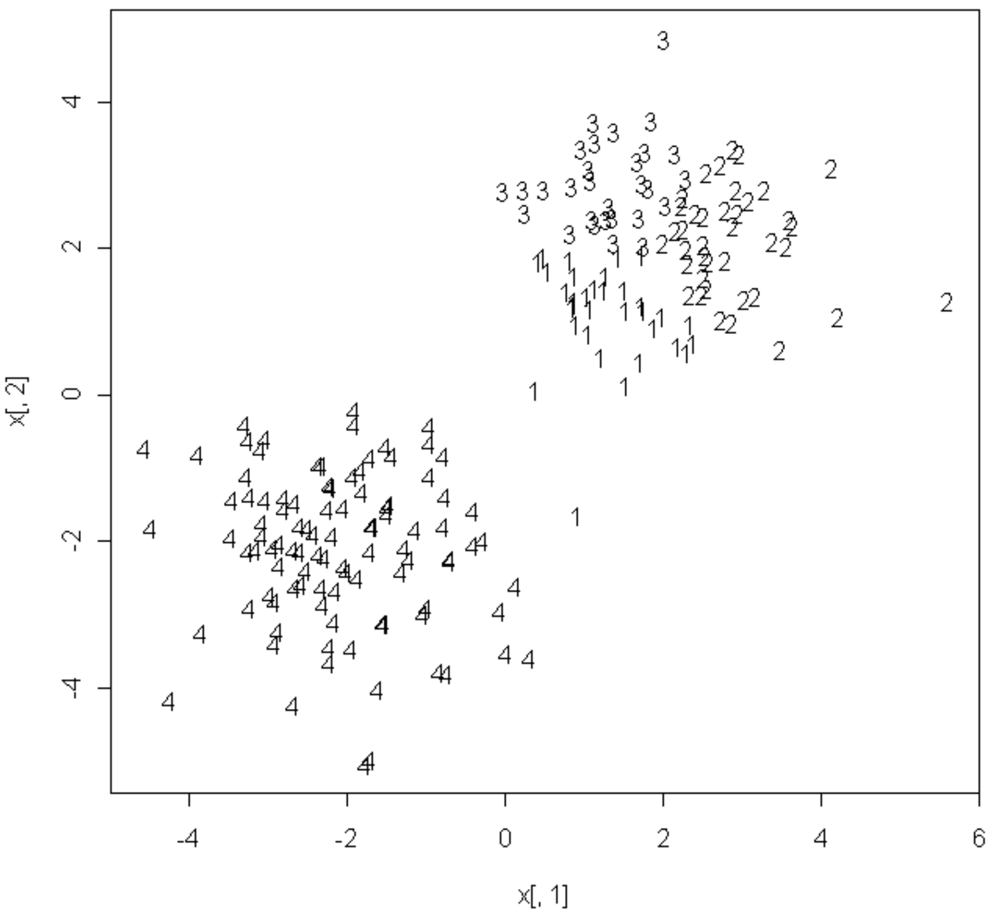
```
#Here we perform k=means clustering for a sequence
  of model
#sizes
x.km2<-kmeans(x,2)
x.km3<-kmeans(x,3)
x.km4<-kmeans(x,4)

plot(x[,1],x[,2],type="n")
text(x[,1],x[,2],labels=as.character(x.km2$cluster))
```


x[, 2]
4
2
0
-2
-4
-4
-2
0
2
4
6
x[, 1]

The 3 term *k*-means solution



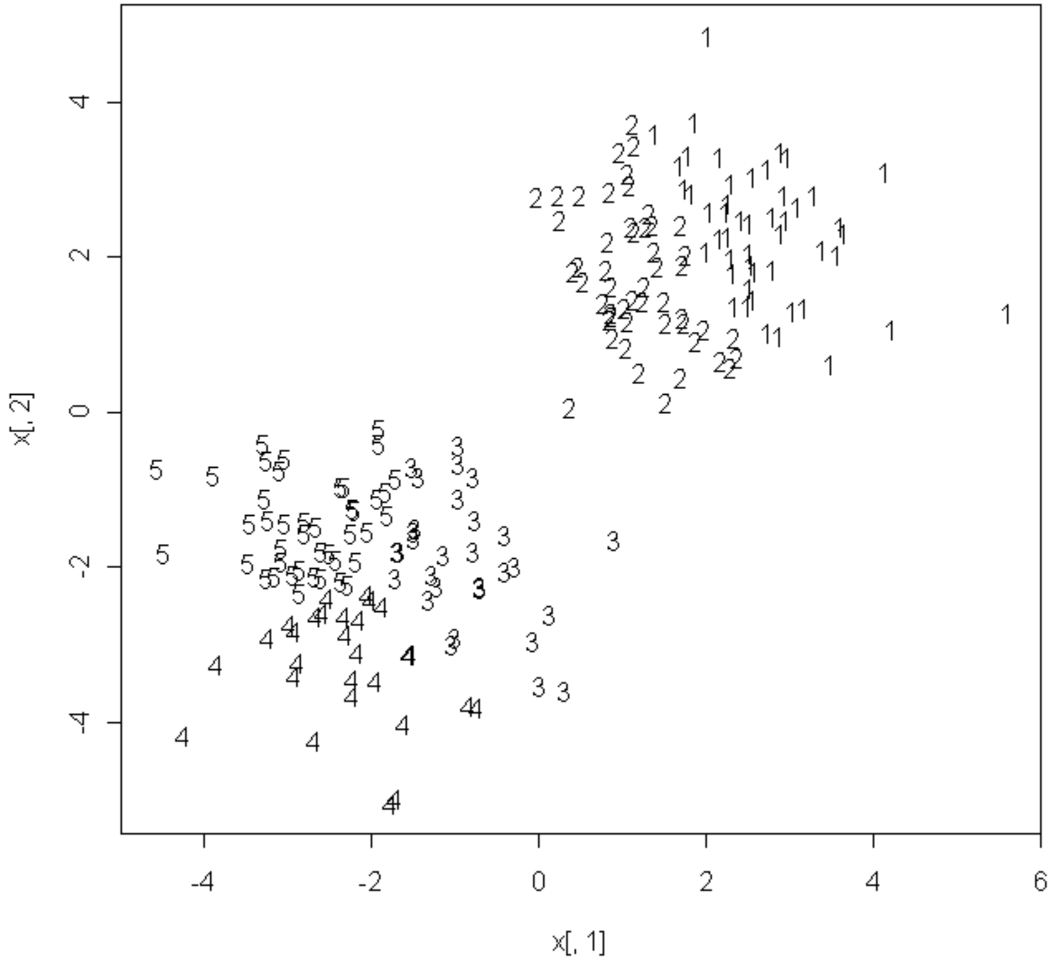
The 4 term *k*-means Solution



Determination of the Number of Clusters Using the Hartigan Criteria

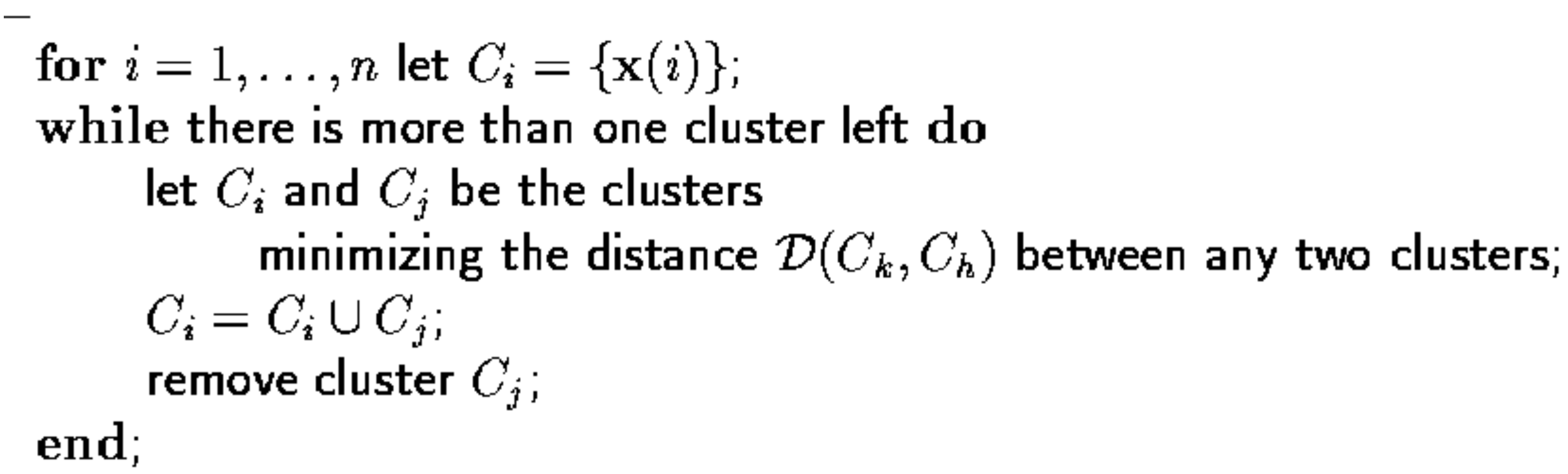
```
> (sum(x.km3$withinss)/sum(x.km4$withinss)-1)*(200-3-1)
[1] 23.08519
> (sum(x.km4$withinss)/sum(x.km5$withinss)-1)*(200-4-1)
[1] 75.10246
> (sum(x.km5$withinss)/sum(x.km6$withinss)-1)*(200-5-1)
[1] -6.553678
> plot(x[,1],x[,2],type="n")
> text(x[,1],x[,2],labels=as.character(x.km5$cluster))
```

k=5 Solution



Hierarchical Clustering

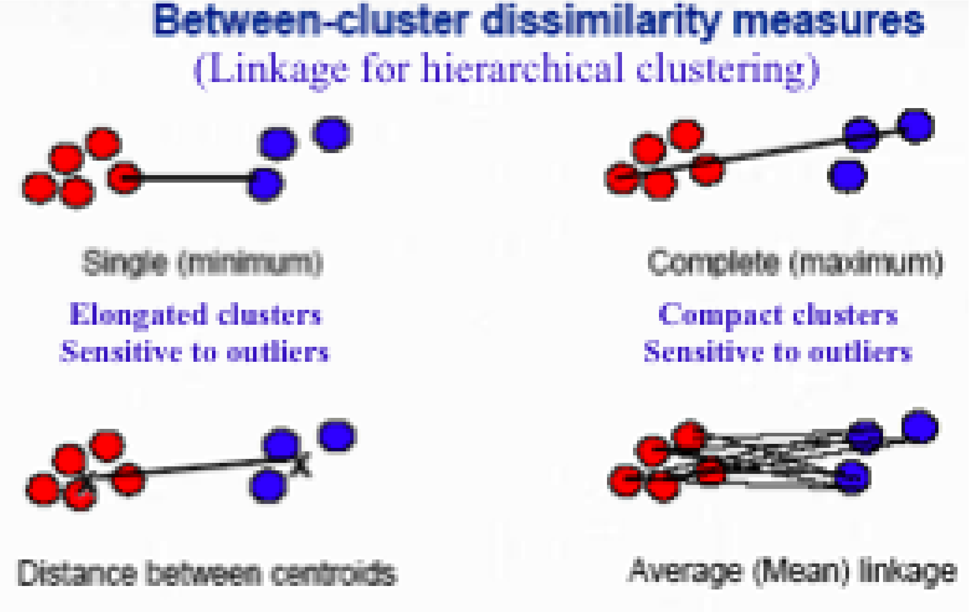
- Agglomerative versus divisive
- Generic Agglomerative Algorithm:



```
for $i = 1, \ldots, n$ let $C_i = \{\mathbf{x}(i)\}$;
while there is more than one cluster left do
    let $C_i$ and $C_j$ be the clusters
        minimizing the distance $\mathcal{D}(C_k, C_h)$ between any two clusters;
    $C_i = C_i \cup C_j$;
    remove cluster $C_j$;
end;
```

- Computing complexity $O(n^2)$

Distance Between Clusters



Between-cluster dissimilarity measures

(Linkage for hierarchical clustering)

Single (minimum)

Elongated clusters
Sensitive to outliers

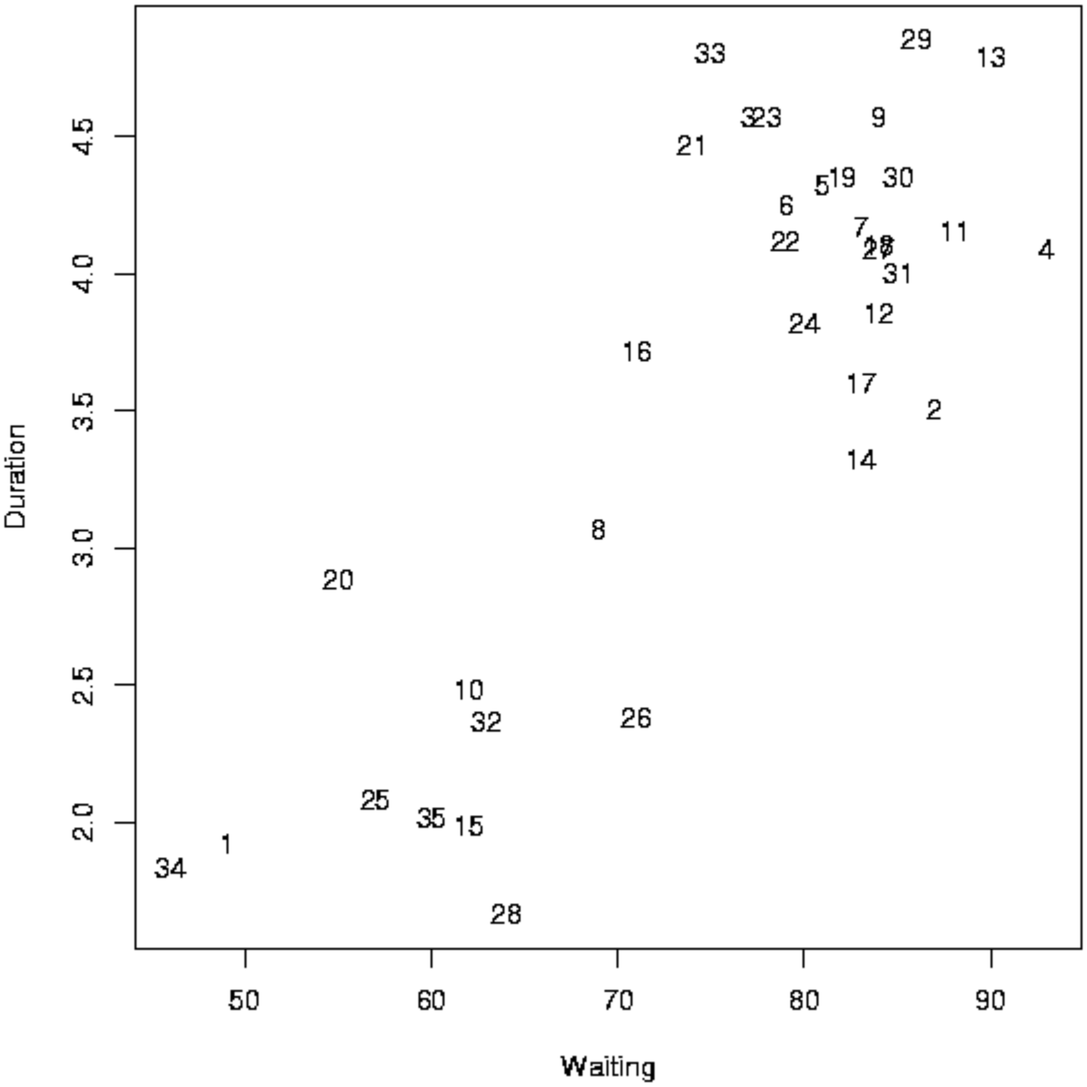
Complete (maximum)

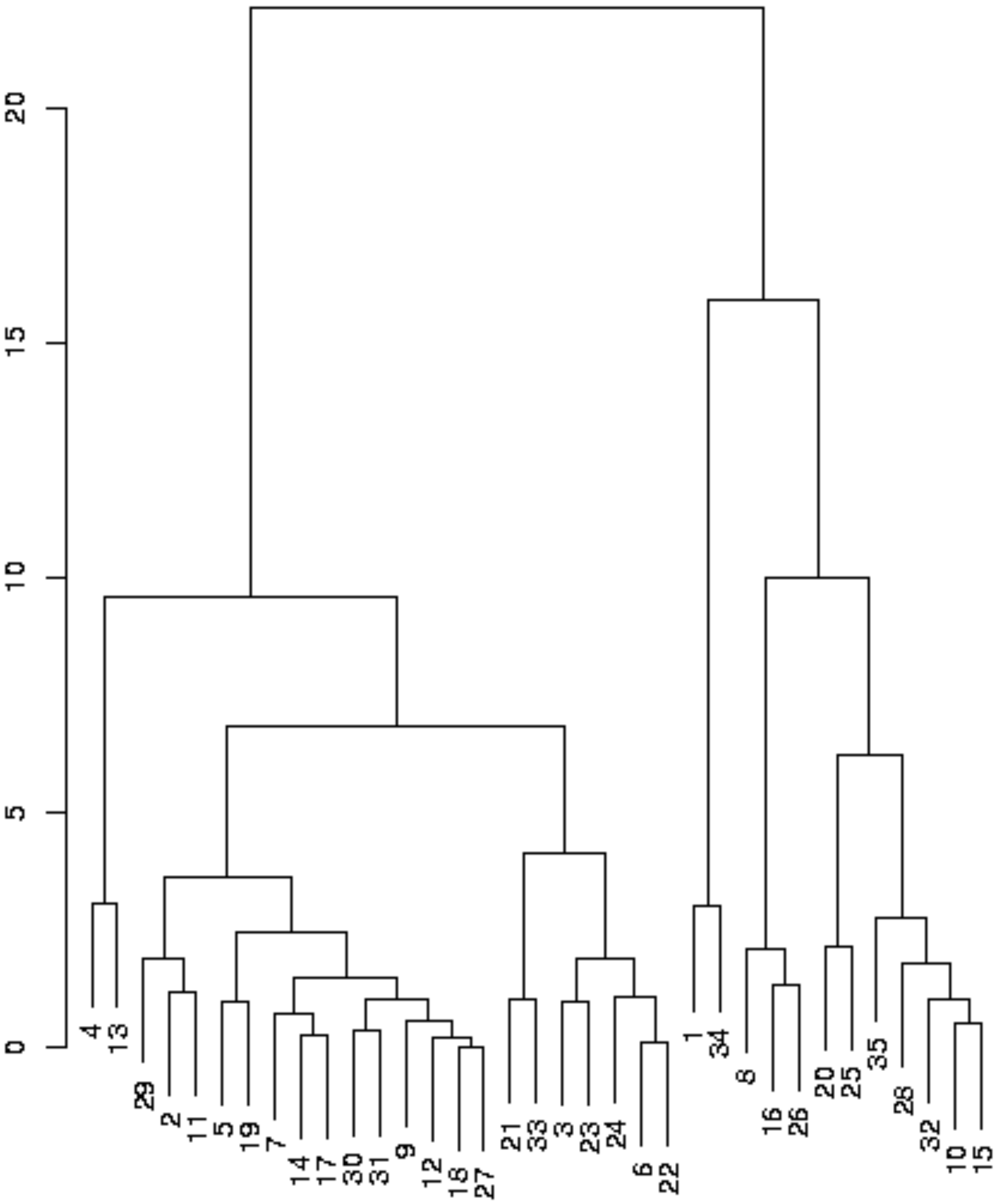
Compact clusters
Sensitive to outliers

Distance between centroids

Average (Mean) linkage

In between
Less sensitive to outliers


Duration
4.5
4.0
3.5
3.0
2.5
2.0
Waiting
50
60
70
80
90
33
29
13
3
23
9
21
5
19
30
6
7
11
22
18
27
4
31
12
24
16
17
2
14
8
20
10
26
32
25
35
15
1
34
28



Height of the cross-bar shows the change in within-cluster SS

Agglomerative

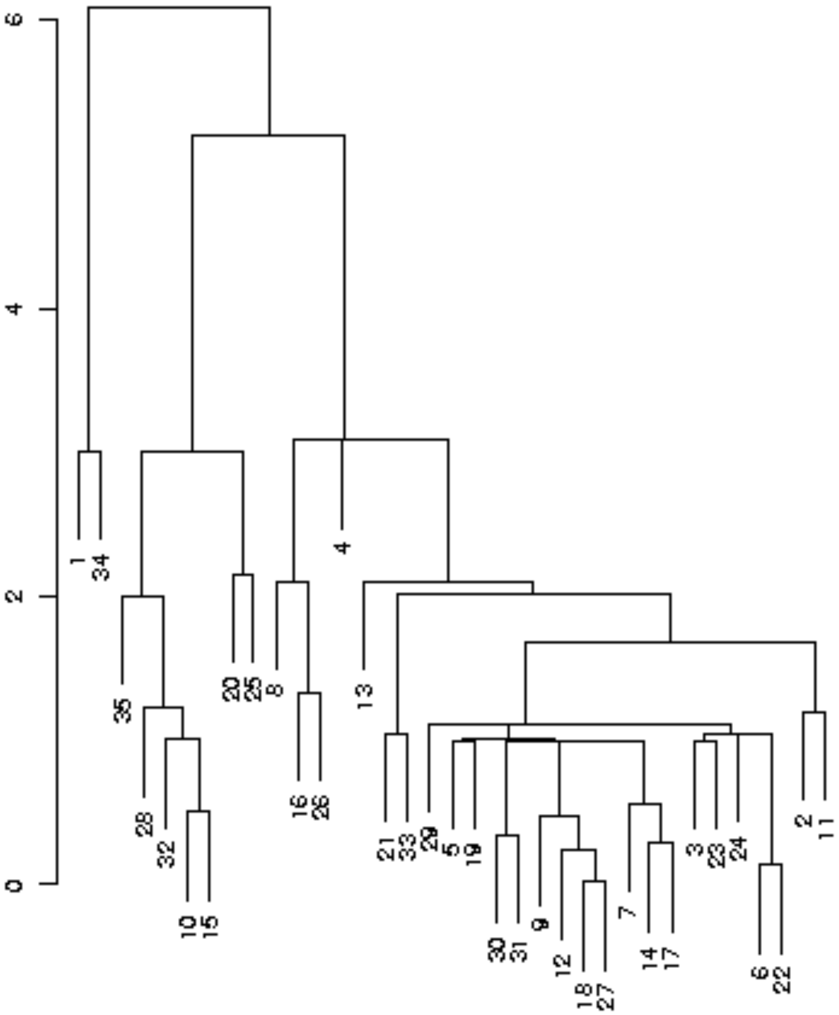


Figure 9.9: Dendrogram of the single link method applied to the data in figure 9.7.

Hierarchical Clustering in R

- Assuming that you have read your data into a matrix called `data.mat` then first you must compute the interpoint distance matrix using the dist function

```
library(mva)
data.dist<- dist(data.mat)
```

- Next hierarchical clustering is accomplished with a call to hclust

`hclust`

- It computes complete linkage clustering by default
- Using the `method="connected"` we obtain single linkage clustering
- Using the `method = "average"` we obtain average clustering

plclust and cutree

- plot is used to plot our dendrogram

- cutree is used to examine the groups that are given at a given cut level

Computing the Distance Matrix

```
dist(x, metric = "euclidean")
```

metric = character string specifying the distance metric to be used.

The currently available options are "euclidean", "maximum",

"manhattan", and "binary". Euclidean distances are root sum-of-squares of differences, "maximum" is the maximum difference, "manhattan" is the of absolute differences, and "binary" is the proportion of non-that two vectors do not have in common (the number of occurrences of a zero and a one, or a one and a zero divided by the number of times at least one vector has a one).

Example Distance Matrix Computation

```
> x.dist<-dist(x)

> length(x.dist)
[1] 19900
```

hclust

```
hclust(d, method = "complete", members=NULL)
```

d   a dissimilarity structure as produced by dist.

method   the agglomeration method to be used. This should be (an unambiguous abbreviation of) one of "ward", "single", "complete", "average", "median" or "centroid".

Values Returned by hclust

merge an *n-1* by 2 matrix. Row *i* of merge describes the merging of clusters at step *i* of the clustering. If an element *j* in the row is negative, then observation *-j* was merged at this stage. If *j* is positive then the merge was with the cluster formed at the (earlier) stage *j* of the algorithm. Thus negative entries in merge indicate agglomerations of singletons, and positive entries indicate agglomerations of non-singletons.

height aset of *n-1* non-decreasing real values. The clustering *height*: that is, the value of the criterion associated with the clustering method for the particular agglomeration.

order a vector giving the permutation of the original observations suitable for plotting, in the sense that a cluster plot using this ordering and matrix merge will not have crossings of the branches.

labels labels for each of the objects being clustered.

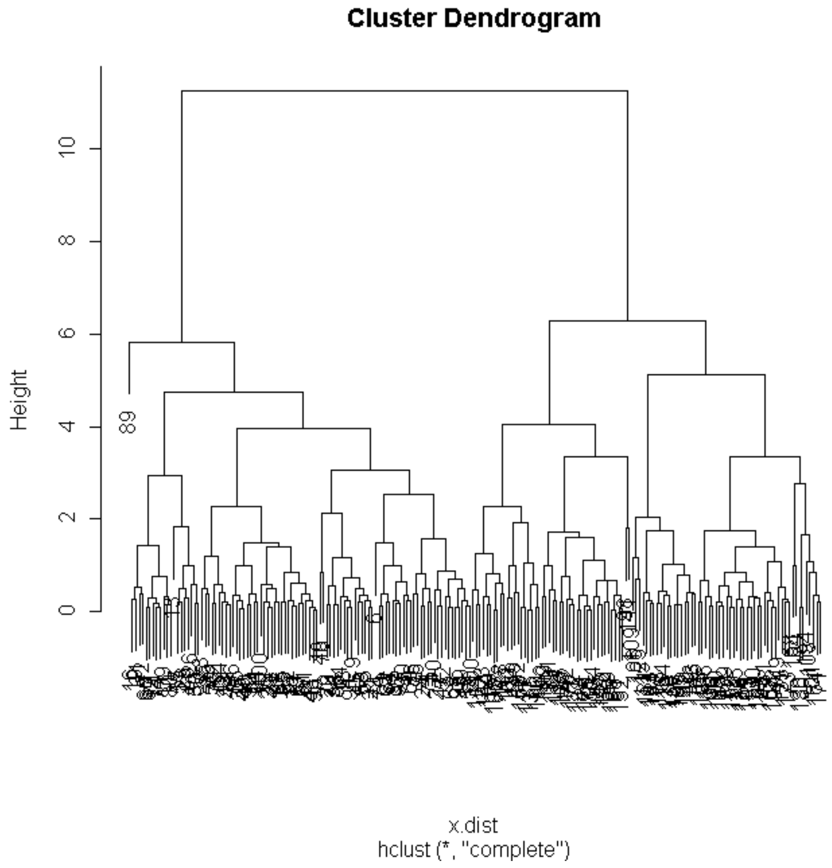
call the call which produced the result.

method the cluster method that has been used.

dist.method the distance that has been used to create d (only returned if the distance object has a "method" attribute).

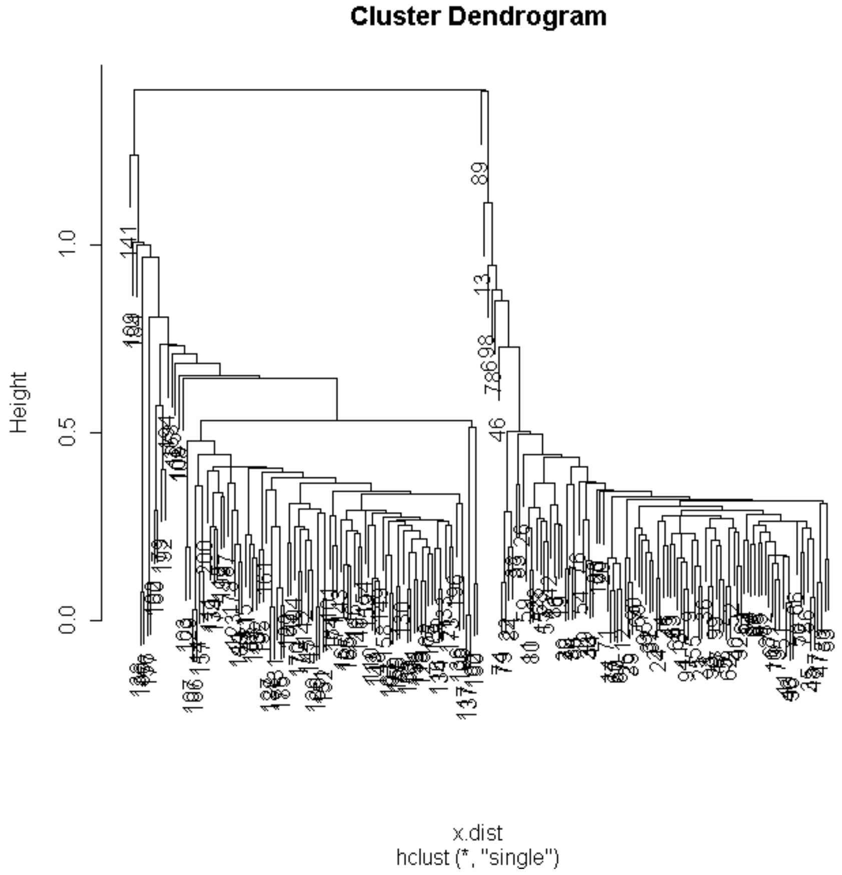
Complete Linkage Clustering with `hclust`

```
> plot(hclust(x.dist))
```



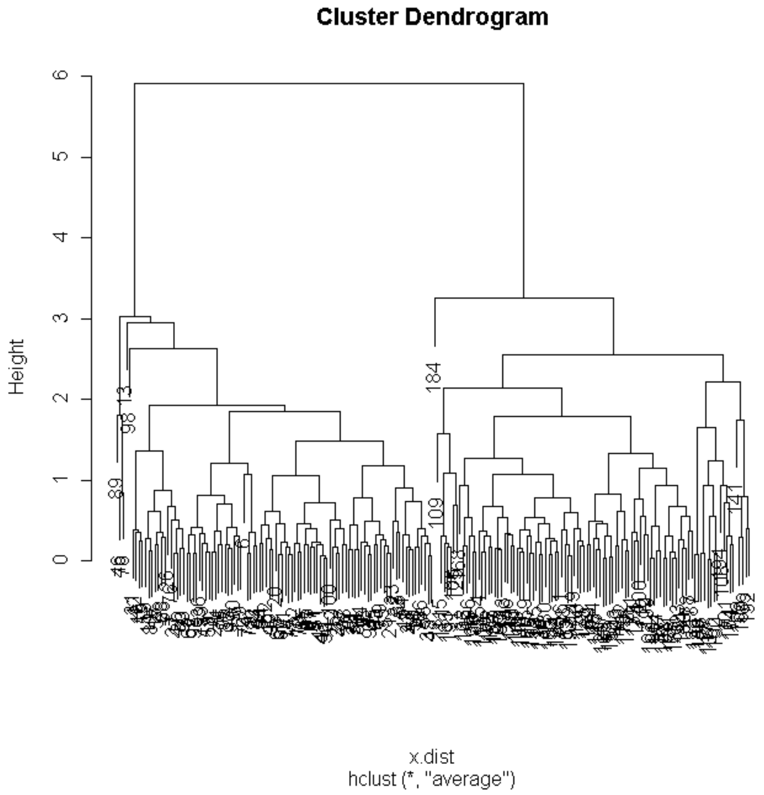
Single Linkage Clustering with `hclust`

```
> plot(hclust(x.dist,method=“single"))
```



Average Linkage Clustering with `hclust`

```
plot(hclust(x.dist,method="average"))
```



Pruning Our Tree

```
cutree(tree, k = NULL, h = NULL)
```

tree a tree as produced by hclust. cutree() only expects a list with components merge, height, and labels, of appropriate content each.

k an integer scalar or vector with the desired number of groups

h numeric scalar or vector with heights where the tree should be cut.

At least one of k or h must be specified, k overrides h if both are given.

Values Returned

cutree returns a vector with group memberships if k or h are scalar, otherwise a matrix with group meberships is returned where each column corresponds to the elements of k or h, respectively (which are also used as column names).

Example Pruning

```
> x.cl2<-cutree(hclust(x.dist),k=2)

> x.cl2[1:10]
 [1] 1 1 1 1 1 1 1 1 1 1

> x.cl2[190:200]
 [1] 2 2 2 2 2 2 2 2 2 2 2
```

Identifying the Number of Clusters

- As indicated previously we really have no way of identify the true cluster structure unless we have divine intervention

- In the next several slides we present some well-known methods

Method of Mojena

- Select the number of groups based on the first stage of the dendogram that satisfies

$$\alpha_{j+1} > \overline{\alpha} + ks_{\alpha}$$

- The a_0,a_1,a_2,... a_{n-1} are the fusion levels corresponding to stages with n, n-1, …,1 clusters. $\overline{\alpha}$ and s_{α} are the mean and unbiased standard deviation of these fusion levels and k is a constant.

- Mojena (1977) 2.75 < k < 3.5

- Milligan and Cooper (1985) k=1.25

Method of Mojena Applied to Our Data Set - I

```
> x.clfl<-hclust(x.dist)$height
#assign the fusion levels

> x.clm<-mean(x.clfl)
#compute the means

> x.cls<-sqrt(var(x.clfl))
#compute the standard deviation

> print((x.clfl-x.clm)/x.cls)
#output the results for comparison with k
```

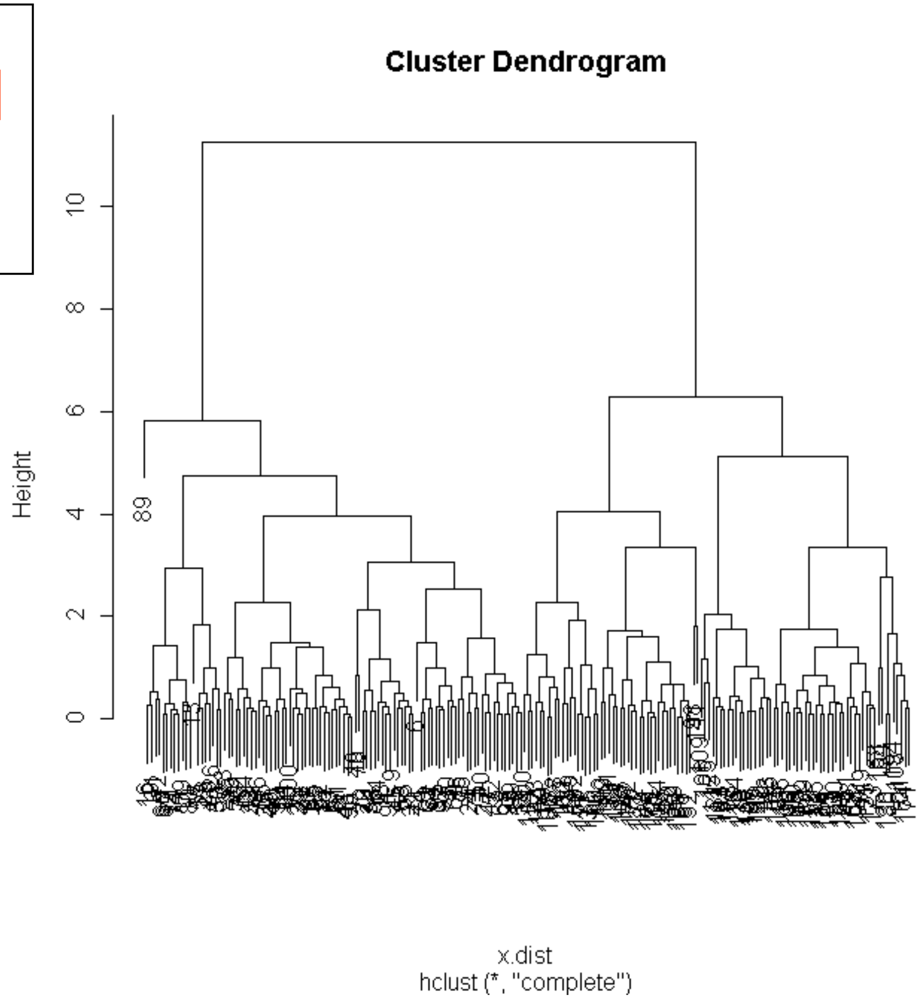
Method of Mojena Applied to Our Data Set - II

```
> print((x.clfl-x.clm)/x.cls)
```

```
 [1] -0.609317763 -0.595451243 -0.591760600 -0.590785339 -0.590132779
  [6] -0.587620192 -0.574381404 -0.570288225 -0.560984067 -0.559183861
.
.
.
[186]  1.189406923  1.391764160  1.582611713  1.731697165  1.817821995
[191]  2.056156268  2.057782017  2.534517541  2.606030029  3.157604485
[196]  3.473036668  4.028366785  4.385419127  8.368682725
```

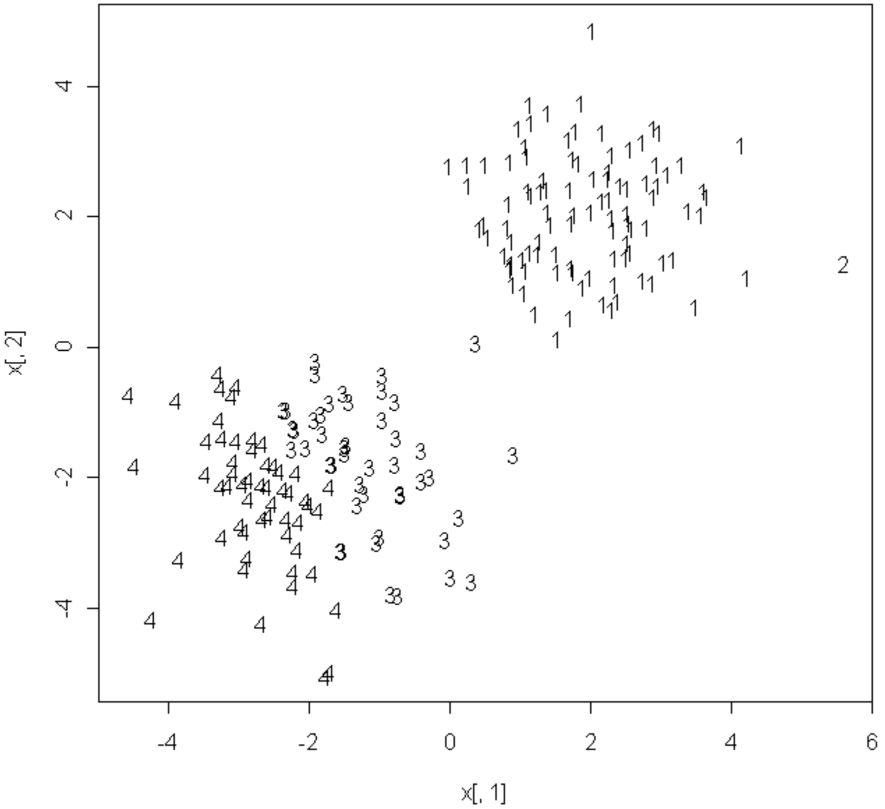
Method of Mojena Applied to Our Data Set - III

```
> print(x.clfl[196]
[1] 5.131528
```



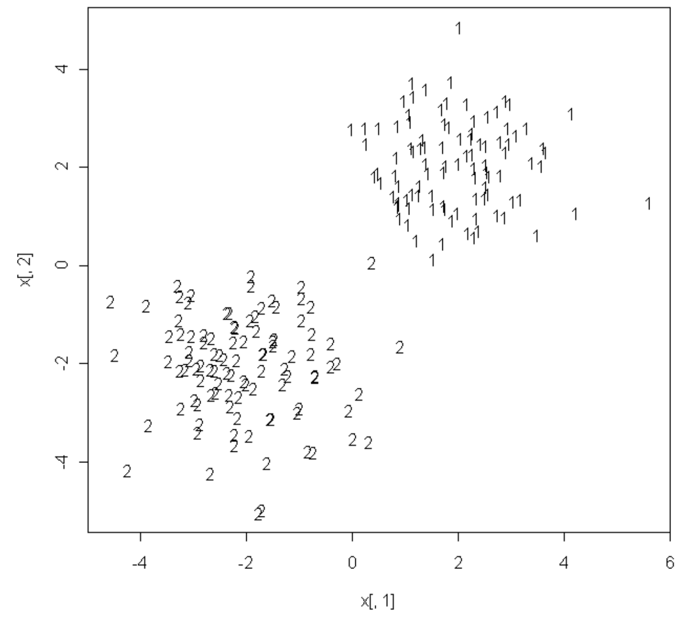
Visualizing Our Cluster Structure

```
> x.clmojena<-cutree(hclust(x.dist),h=x.clfl[196])
> plot(x[,1],x[,2],type="n")
> text(x[,1],x[,2], labels=as.character
  (x.clmojena))
```

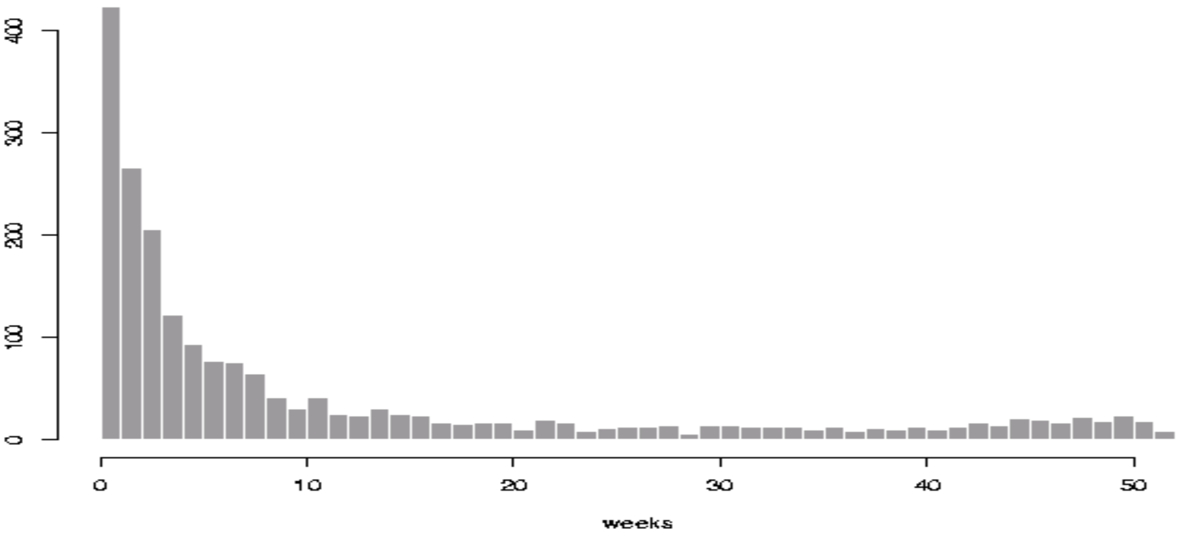


Visualizing Our Cluster Structure (Cutting the Tree Higher)

```
> x.cllastsplit<-cutree(hclust
  (x.dist),h=x.clfl[198])
```



Mixture Models



$$f(x) = p\frac{(\lambda_1)^x e^{-\lambda_1}}{x!} + (1-p)\frac{(\lambda_2)^{52-x} e^{-\lambda_2}}{(52-x)!}$$

"Two-stage model"

$$f(x) = \sum_{k=1}^{K} \pi_k f_k(x;\theta_k)$$

Mixture Models and EM

•No closed-form for MLE' s

•EM widely used - flip-flop between estimating parameters assuming class mixture component is known and estimating class membership given parameters.

•Time complexity $O(Kp^2n)$; space complexity $O(Kn)$

•Can be slow to converge; local maxima

Mixture-model example: Binomial Mixture

Market basket: $x_j(i) = \begin{cases} 1, \text{ if person } i \text{ purchased item } j \\ 0, \text{ otherwise} \end{cases}$

For cluster *k*, item *j*: $p_k(x_j;\theta_{kj}) = \theta_{kj}^{x_j}(1-\theta_{kj})^{1-x_j}$

Thus for person *i*: $p(x(i)) = \sum_{k=1}^{K} \pi_k \prod_j \theta_{kj}^{x_j}(1-\theta_{kj})^{1-x_j}$

Probability that person *i* is in cluster *k*:

$$p(k \mid i) = \frac{\pi_k \prod_j \theta_{kj}^{x_j(i)}(1-\theta_{kj})^{1-x_j(i)}}{p(x(i))}$$

E-step

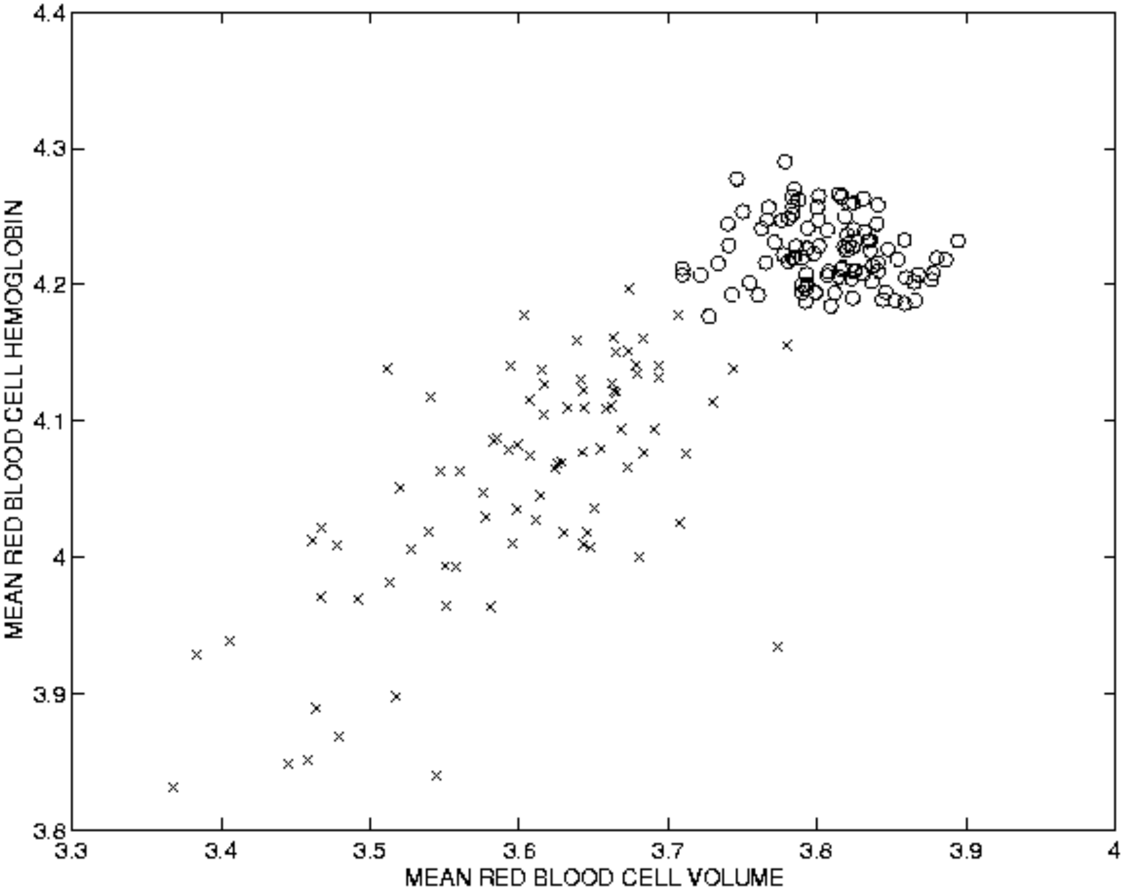
Update within-cluster parameters:

$$\theta_{kj}^{new} = \frac{\sum_{i=1}^{n} p(k \mid i) x_j(i)}{\sum_{i=1}^{n} p(k \mid i)}$$

M-step

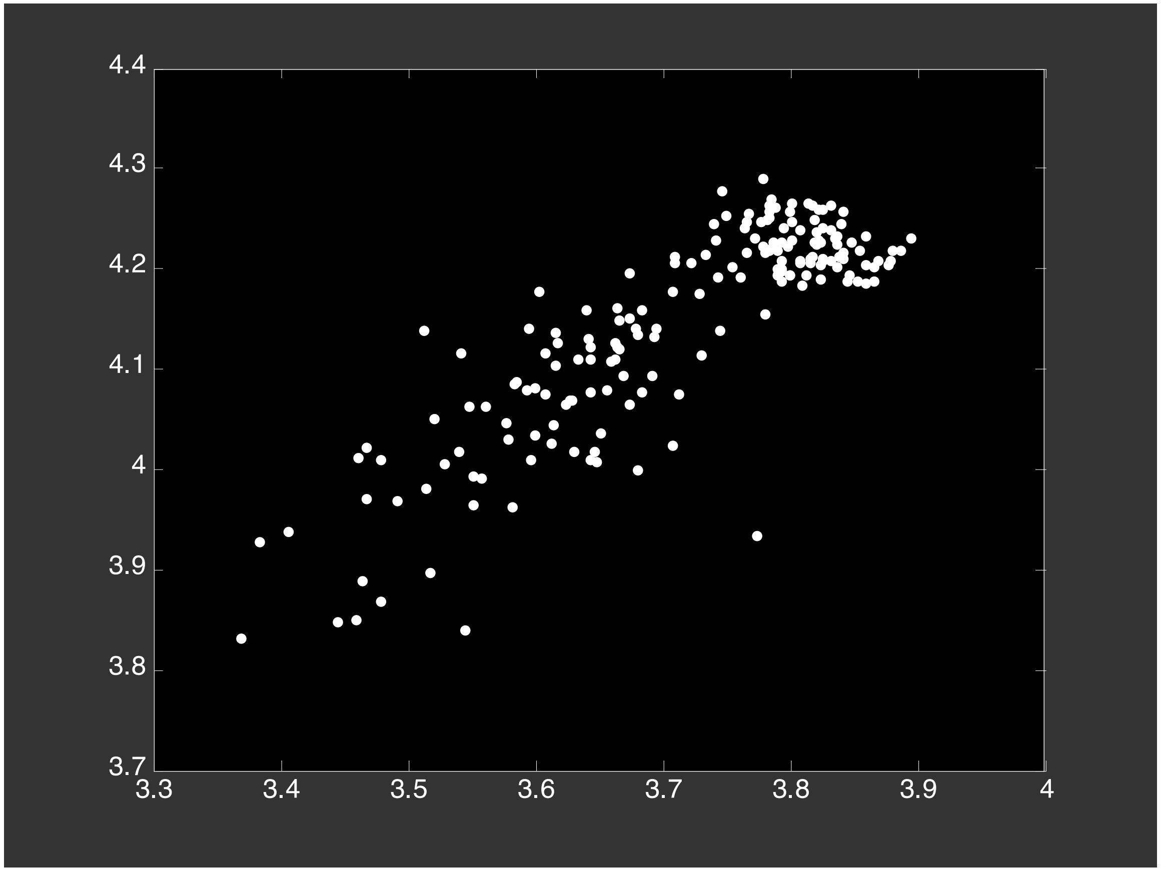
Model-based Clustering

$$f(x) = \sum_{k=1}^{K} \pi_k f_k(x;\theta_k)$$

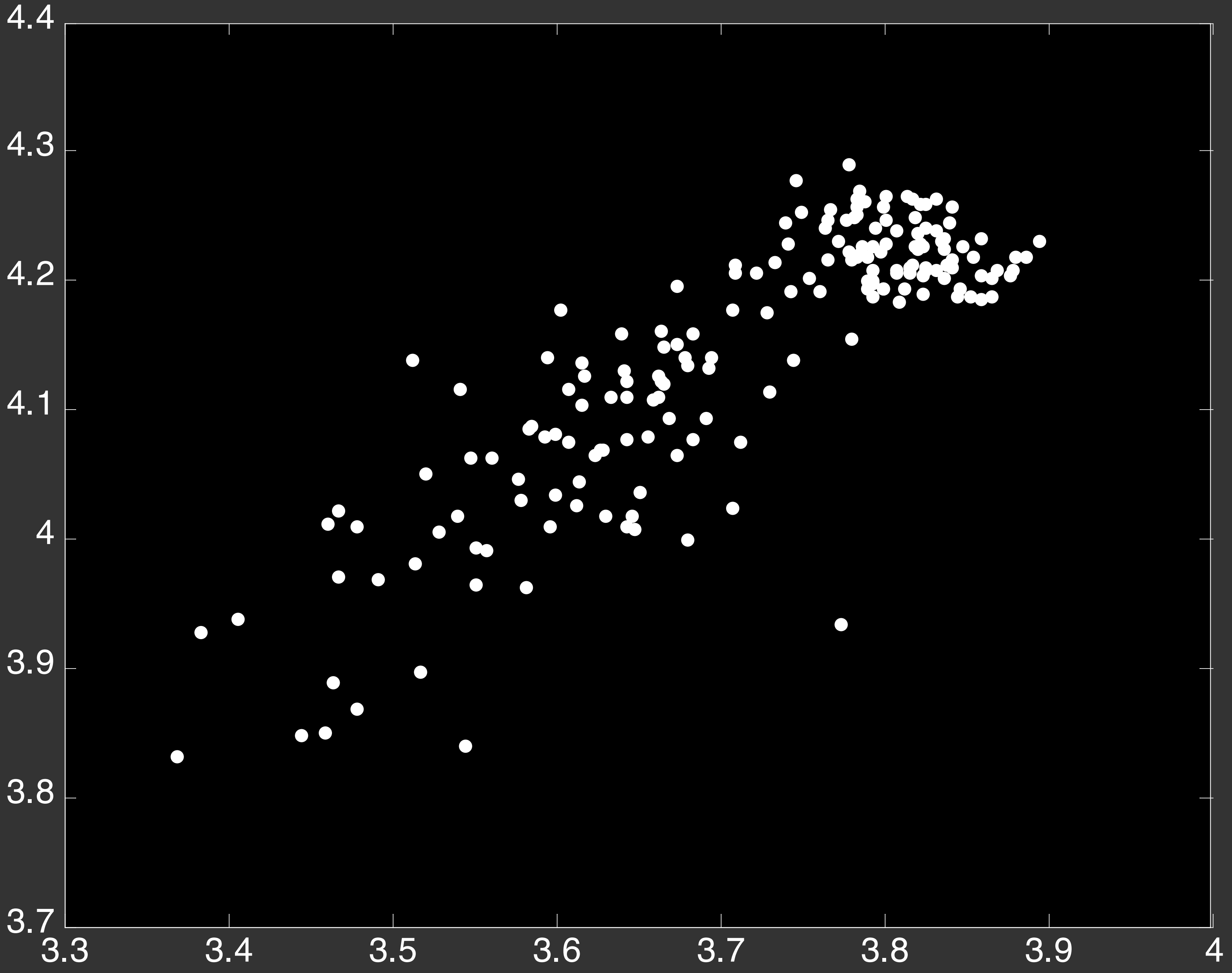


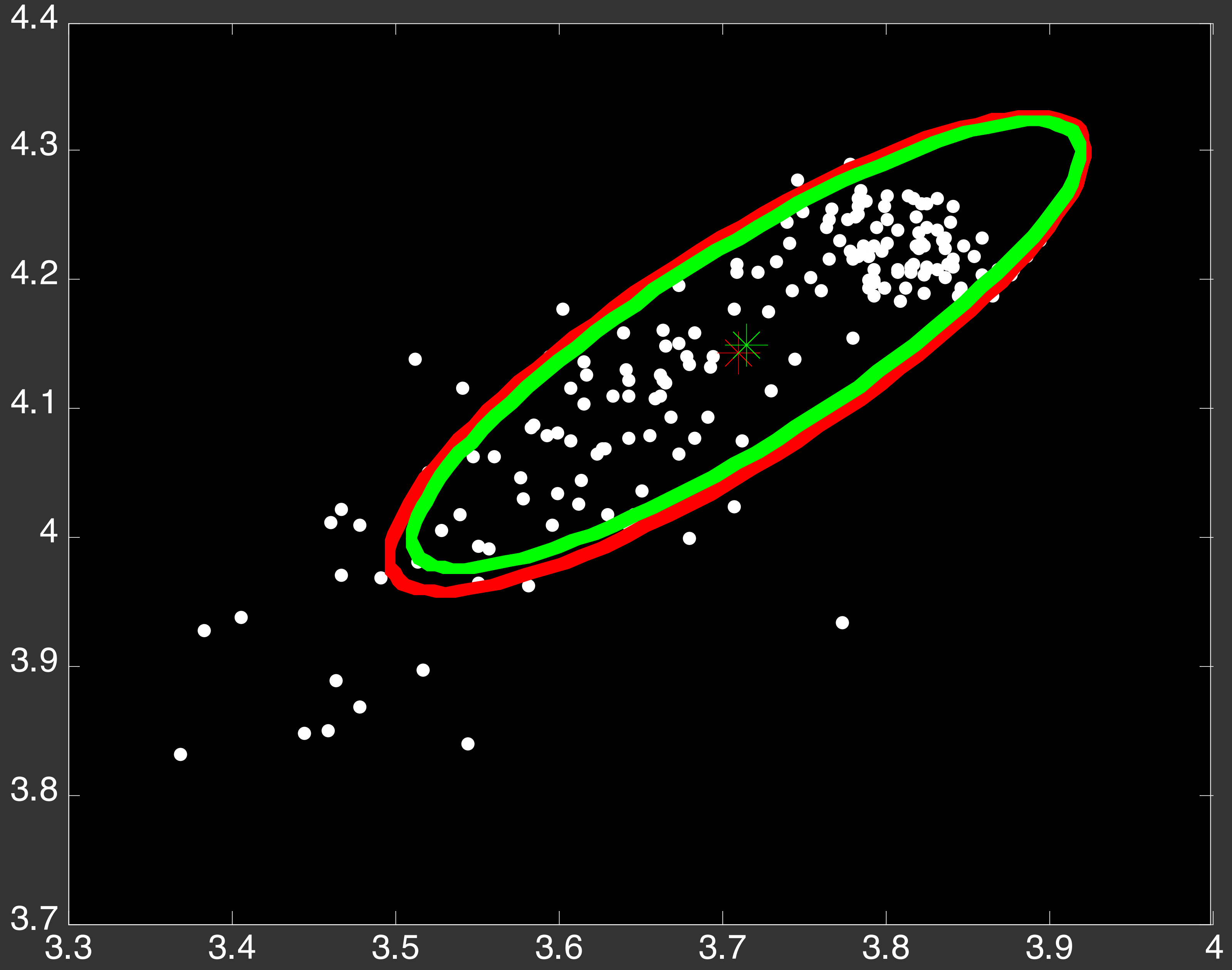
Model-based Clustering

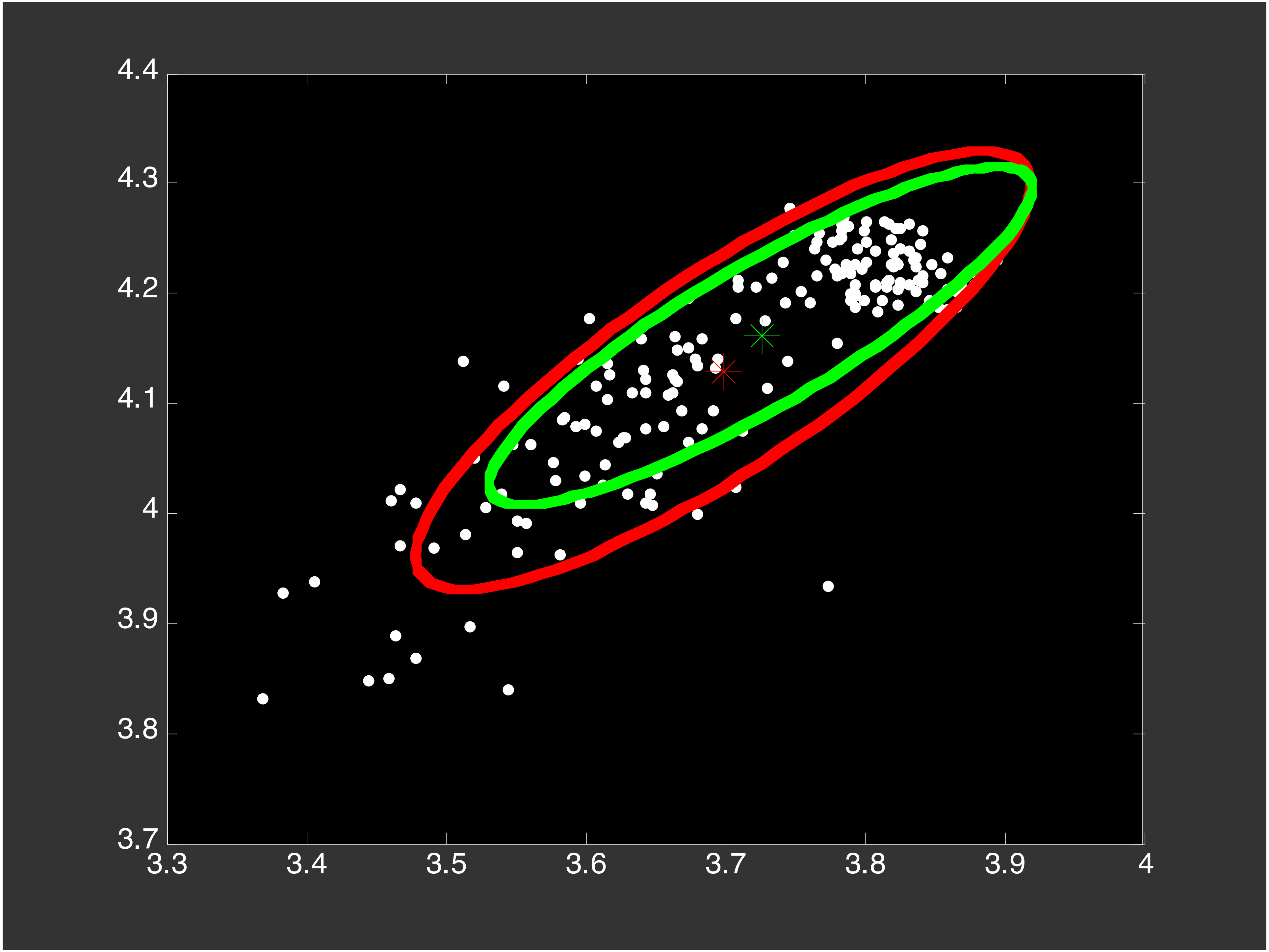
$$f(x) = \sum_{k=1}^{K} \pi_k f_k(x;\theta_k)$$

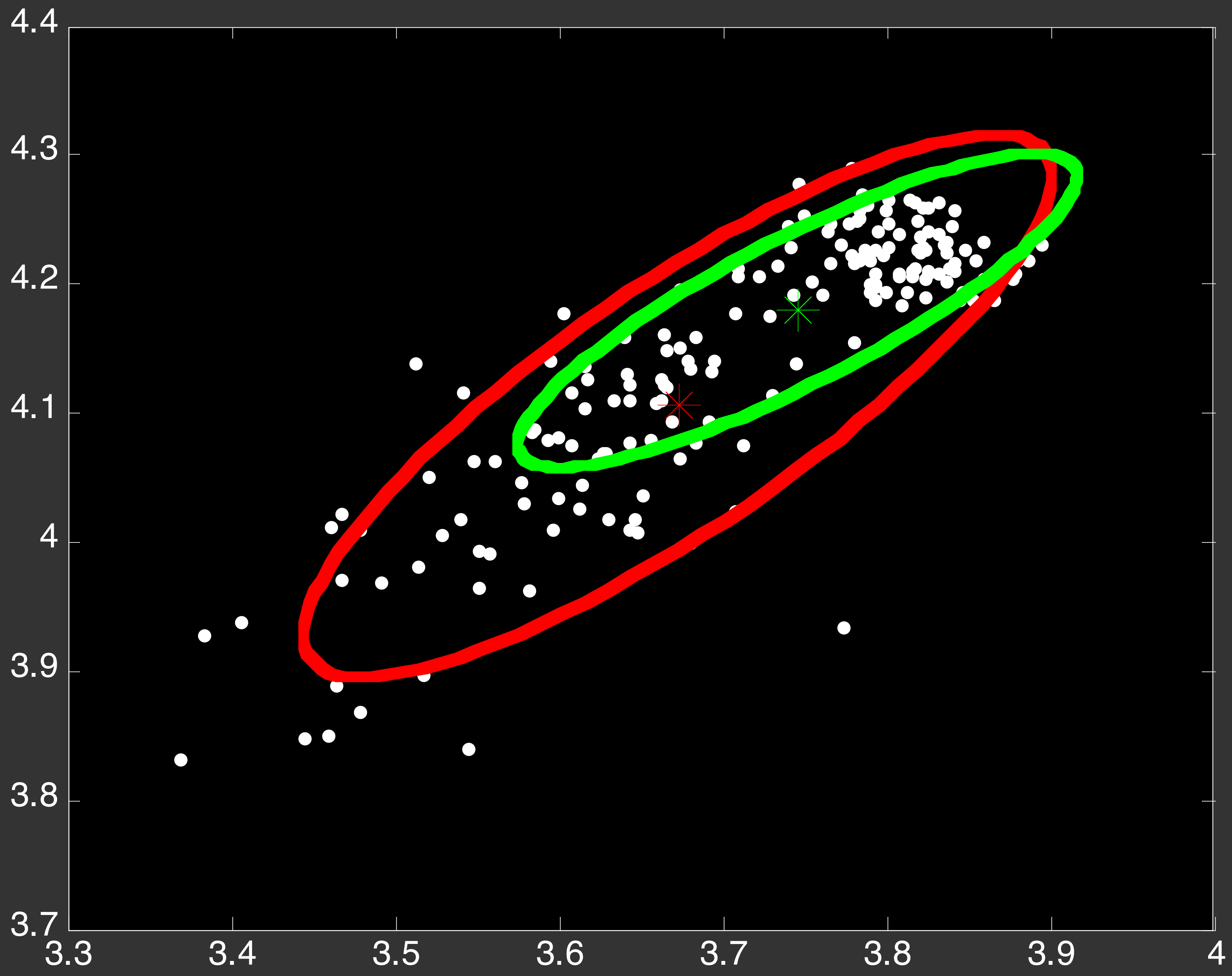


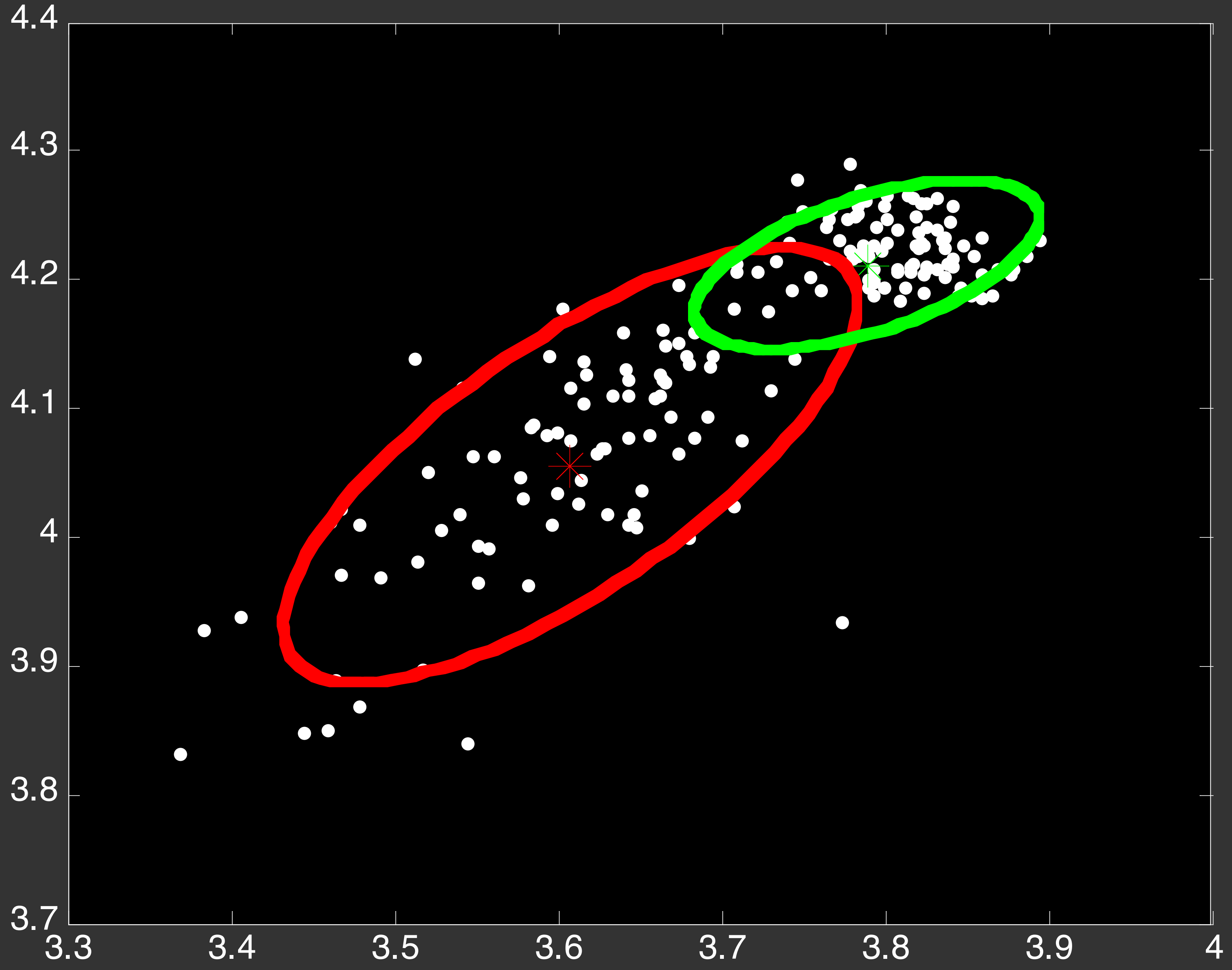
Padhraic Smyth, UCI

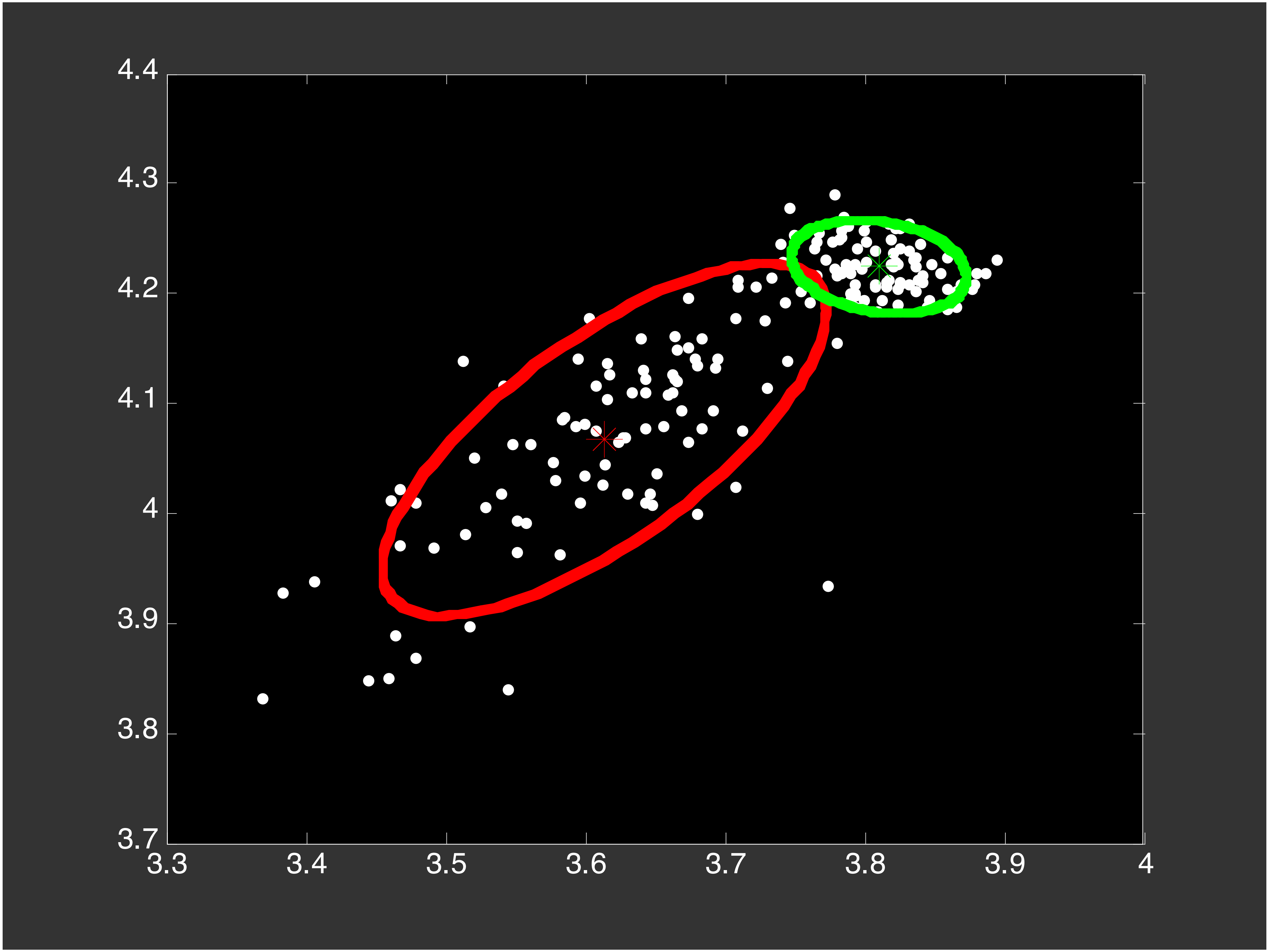

4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4

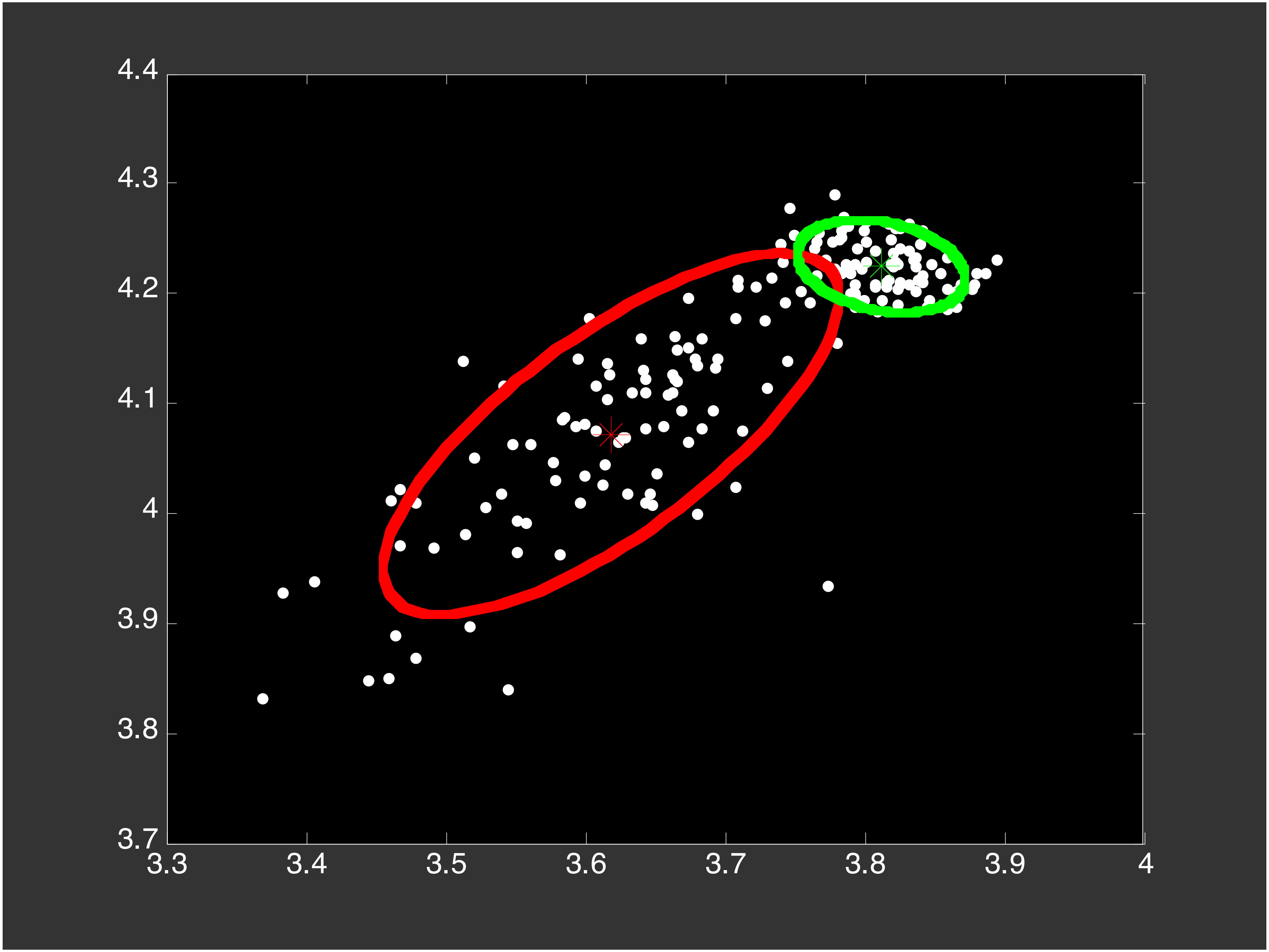

4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4


4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4

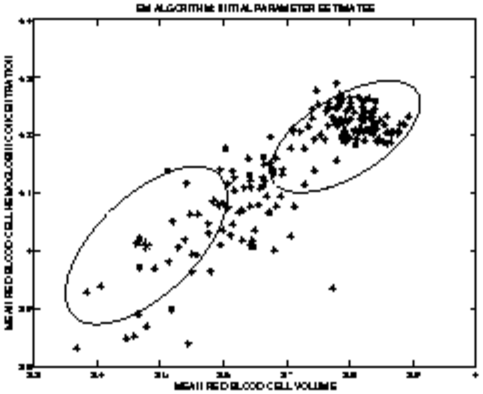

4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4


4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4

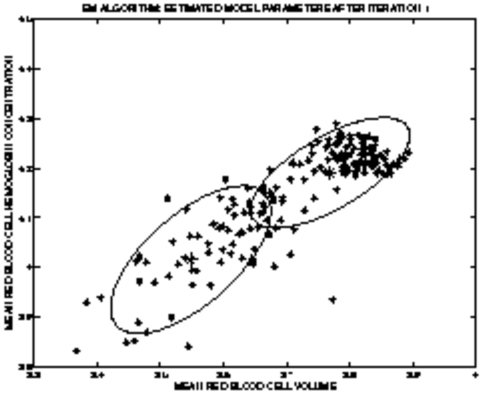

4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4


4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4

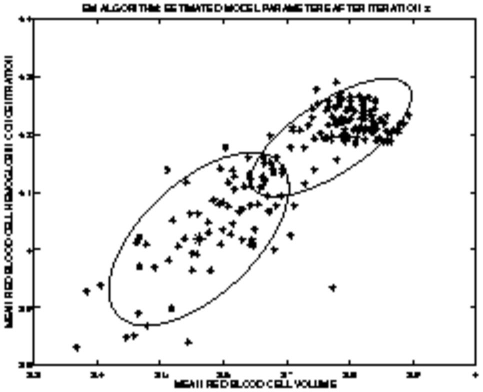
Iter: 0

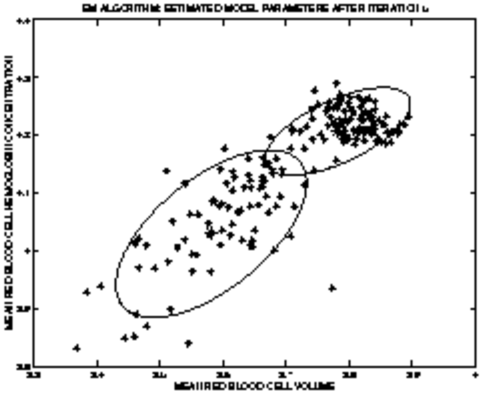


Iter: 1

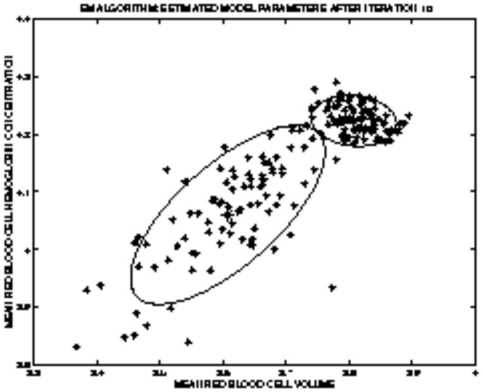


Iter: 2

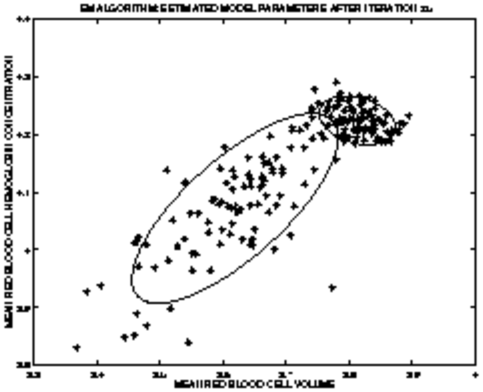




Iter: 5



Iter: 10



Iter: 25

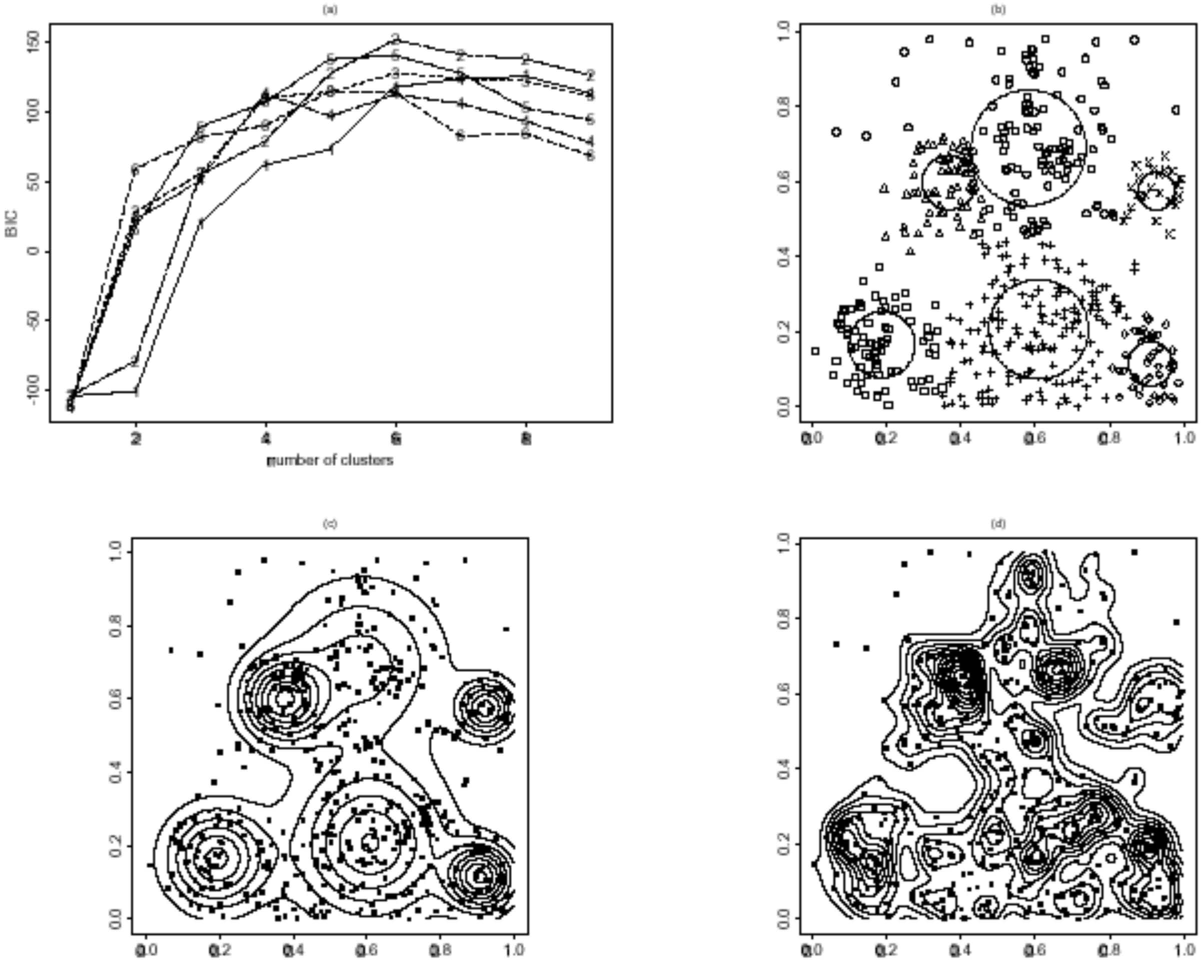


Figure 8: Density estimation for the Lansing Woods maples. (a) BIC from model-based clustering. The maximum-BIC model is a six-component nonuniform spherical mixture. (b) Model-based classification, with circles indicating the circles defined by the estimated covariance of each of the six groups. (c) Contours of the density as determined by model-based clustering, with the location of the maples superimposed. (d) Contours of a standard Gaussian kernel density estimate with bandwidth selected by cross-validation.

Fraley and Raftery (2000)

Advantages of the Probabilistic Approach

- Provides a distributional description for each component
- For each observation, provides a K-component vector of probabilities of class membership
- Method can be extended to data that are not in the form of p-dimensional vectors, e.g., mixtures of Markov models
- Can find clusters-within-clusters
- Can make inference about the number of clusters
- But... its computationally somewhat costly

Mixtures of {Sequences, Curves, …}

$$p(D_i) = \sum_{k=1}^{K} p(D_i \mid c_k)\alpha_k$$

Generative Model

- select a component c_k for individual i
- generate data according to $p(D_i \mid c_k)$
    - $p(D_i \mid c_k)$ can be very general
    - e.g., sets of sequences, spatial patterns, etc

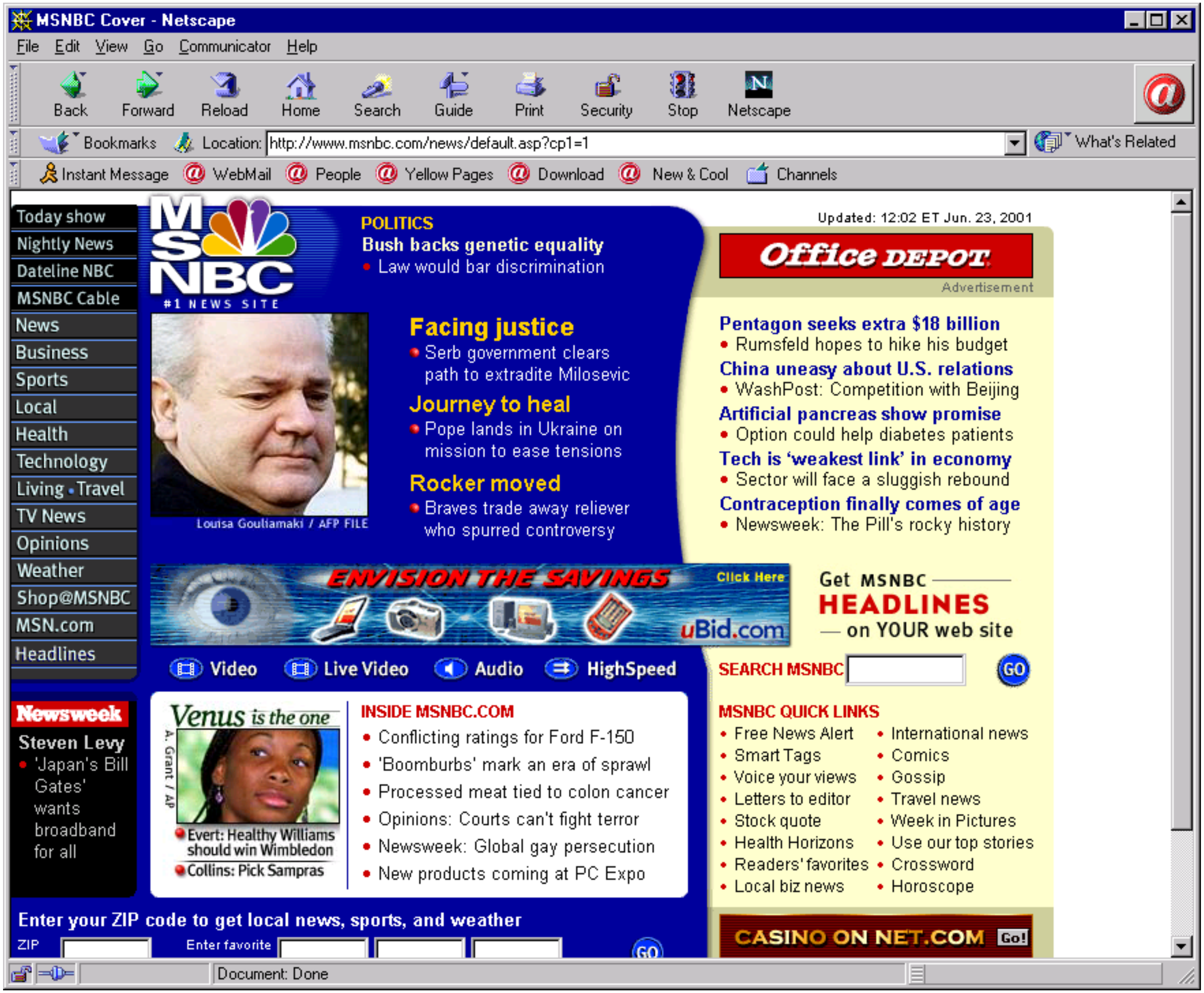
[Note: given $p(D_i \mid c_k)$, we can define an EM algorithm]

Application 1: Web Log Visualization

(Cadez, Heckerman, Meek, Smyth, KDD 2000)

- MSNBC Web logs
  - 2 million individuals per day
  - different session lengths per individual
  - difficult visualization and clustering problem

- WebCanvas
  - uses mixtures of SFSMs to cluster individuals based on their observed sequences
  - software tool: EM mixture modeling + visualization


MSNBC Cover - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Guide Print Security Stop Netscape
Bookmarks Location: http://www.msnbc.com/news/default.asp?cp1=1
What's Related
Instant Message WebMail People Yellow Pages Download New & Cool Channels
Document: Done

Today show
Nightly News
Dateline NBC
MSNBC Cable
News
Business
Sports
Local
Health
Technology
Living • Travel
TV News
Opinions
Weather
Shop@MSNBC
MSN.com
Headlines

MSNBC #1 NEWS SITE

POLITICS
Bush backs genetic equality
- Law would bar discrimination

Louisa Gouliamaki / AFP FILE

Facing justice
- Serb government clears path to extradite Milosevic

Journey to heal
- Pope lands in Ukraine on mission to ease tensions

Rocker moved
- Braves trade away reliever who spurred controversy

Updated: 12:02 ET Jun. 23, 2001

Pentagon seeks extra $18 billion
- Rumsfeld hopes to hike his budget

China uneasy about U.S. relations
- WashPost: Competition with Beijing

Artificial pancreas show promise
- Option could help diabetes patients

Tech is 'weakest link' in economy
- Sector will face a sluggish rebound

Contraception finally comes of age
- Newsweek: The Pill's rocky history

Get MSNBC HEADLINES on YOUR web site

Video Live Video Audio HighSpeed

SEARCH MSNBC GO

Newsweek
Steven Levy
- 'Japan's Bill Gates' wants broadband for all

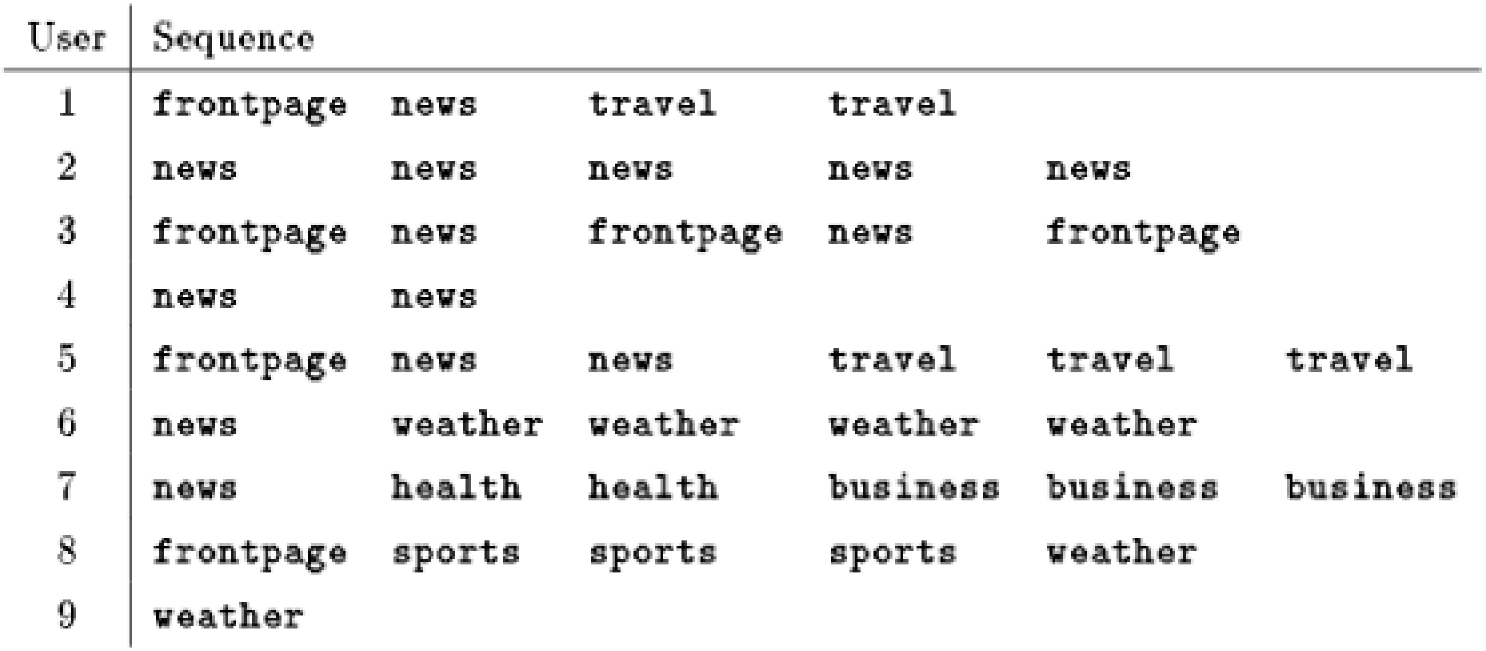
Venus is the one
A. Grant / AP
- Evert: Healthy Williams should win Wimbledon
- Collins: Pick Sampras

INSIDE MSNBC.COM
- Conflicting ratings for Ford F-150
- 'Boomburbs' mark an era of sprawl
- Processed meat tied to colon cancer
- Opinions: Courts can't fight terror
- Newsweek: Global gay persecution
- New products coming at PC Expo

MSNBC QUICK LINKS

• Free News Alert	• International news
• Smart Tags	• Comics
• Voice your views	• Gossip
• Letters to editor	• Travel news
• Stock quote	• Week in Pictures
• Health Horizons	• Use our top stories
• Readers' favorites	• Crossword
• Local biz news	• Horoscope

Enter your ZIP code to get local news, sports, and weather
ZIP Enter favorite GO



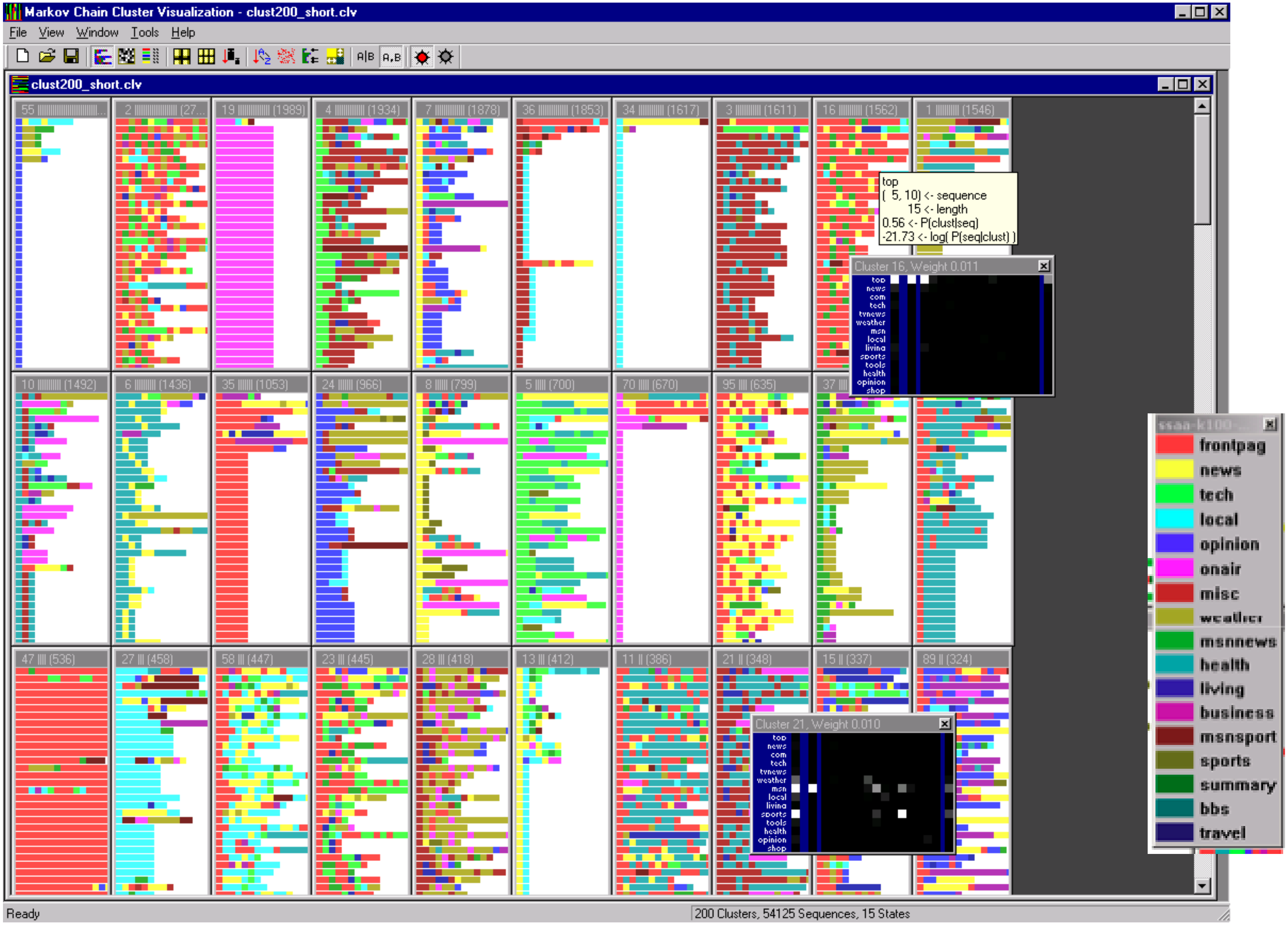
User	Sequence					
1	frontpage	news	travel	travel		
2	news	news	news	news	news	
3	frontpage	news	frontpage	news	frontpage	
4	news	news				
5	frontpage	news	news	travel	travel	travel
6	news	weather	weather	weather	weather	
7	news	health	health	business	business	business
8	frontpage	sports	sports	sports	weather	
9	weather					

Example: Mixtures of SFSMs

Simple model for traversal on a Web site
(equivalent to first-order Markov with end-state)

Generative model for large sets of Web users
- different behaviors <=> mixture of SFSMs

EM algorithm is quite simple: weighted counts



```
top
(  5, 10) <- sequence
        15 <- length
0.56 <- P(clust|seq)
-21.73 <- log( P(seq|clust) )
```

Ready

200 Clusters, 54125 Sequences, 15 States

WebCanvas: Cadez, Heckerman, et al, KDD 2000

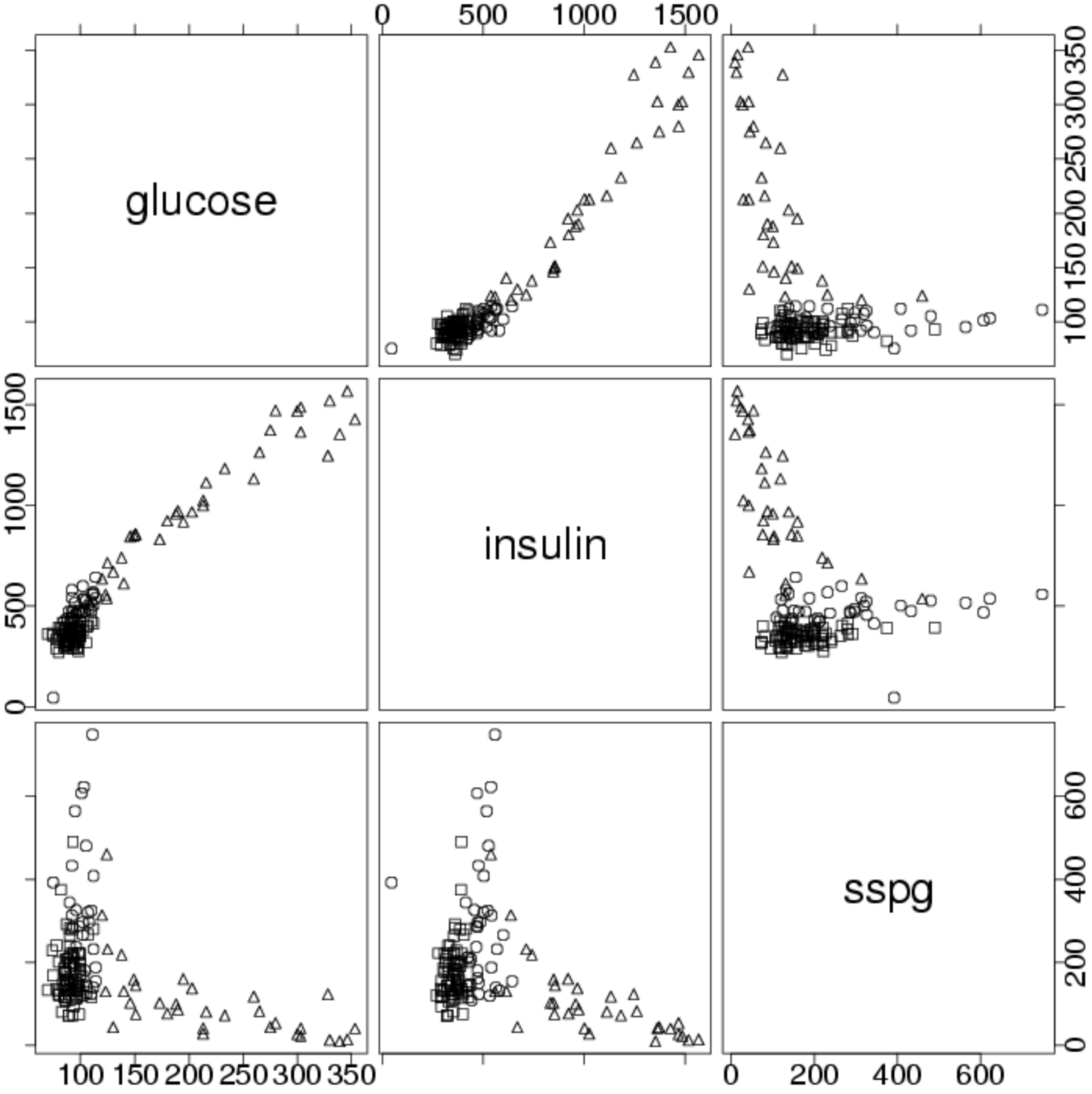


Figure 3: Pairs plot showing the clinical classification of the diabetes data. The symbols have the following interpretation: squares – normal; circles – chemical diabetes; triangles – overt diabetes.

`glucose` - plasma glucose response to oral glucose,
`insulin` - plasma insulin response to oral glucose,
`sspg` - degree of insulin resistance.

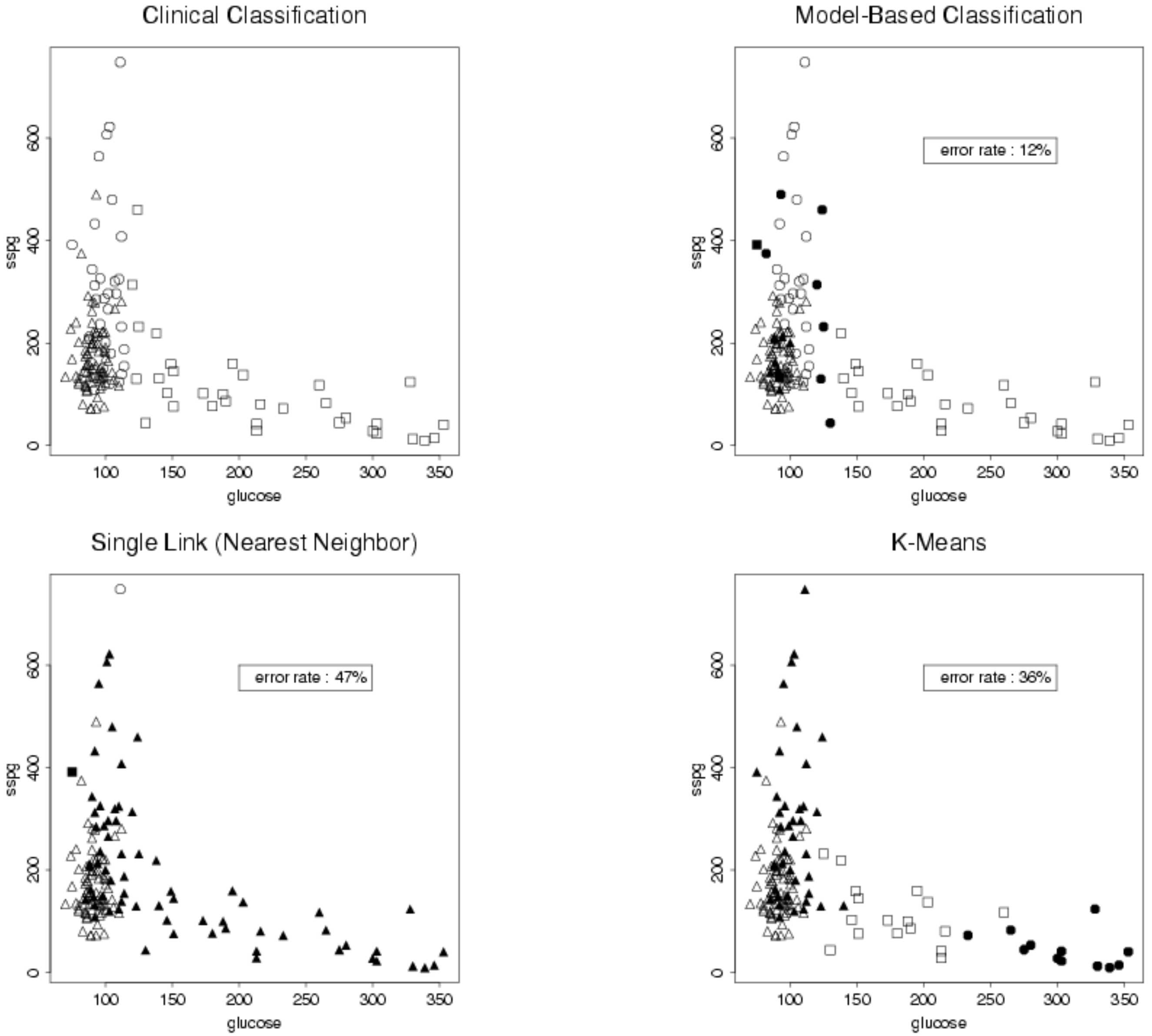


Figure 1: A projection of the three-group classification of the diabetes data from Reaven and Miller [56] using single link or nearest neighbor, standard k-means, and the unconstrained model-based approach. Filled symbols represent misclassified observations.

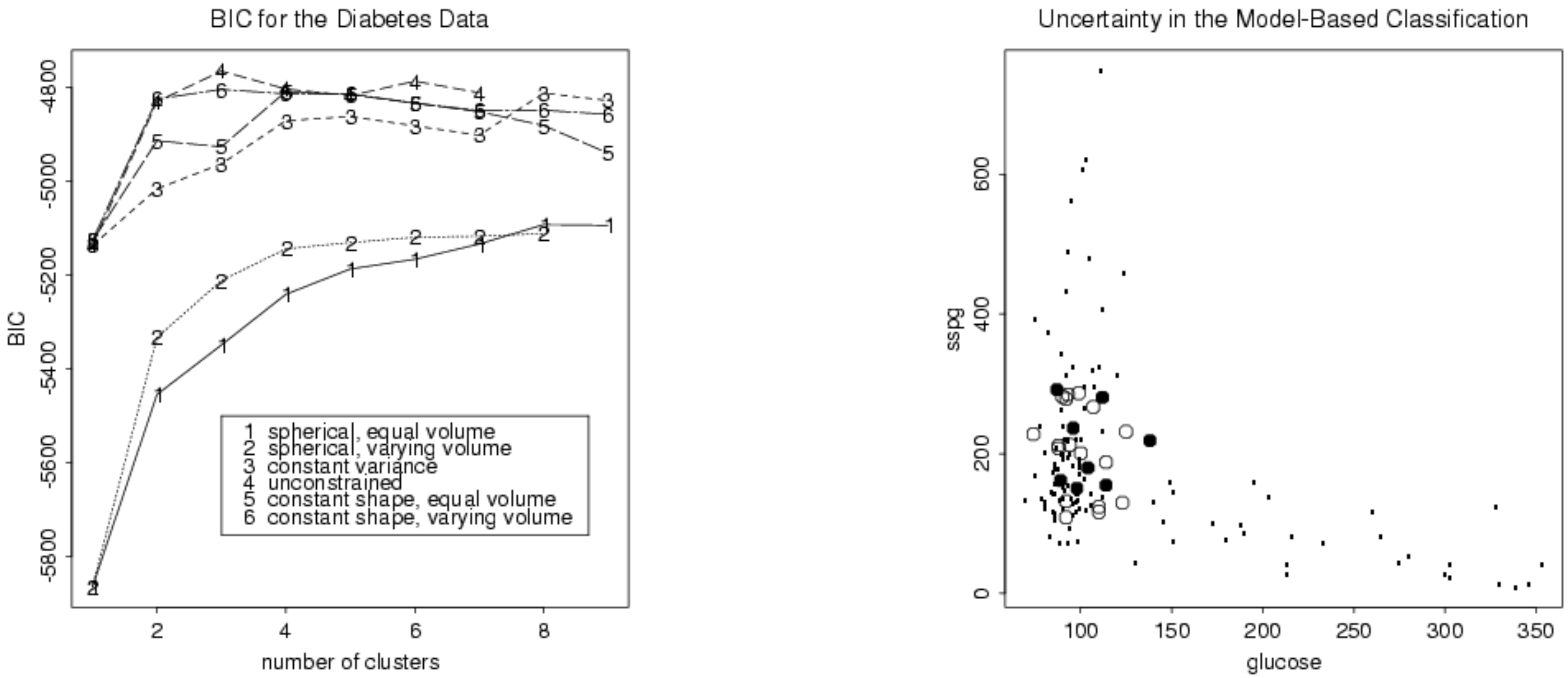


Figure 4: The plot on the left shows the Bayesian Information Criterion (BIC) for model-based methods applied to the diabetes data. The first local (also global) maximum occurs for the unconstrained model with three clusters. The plot on the right depicts the uncertainty of the classification produced by the best model (unconstrained, 3 clusters) indicated by the BIC. The symbols have the following interpretation: dots < 0.1; open circles ≥ 0.1 and < 0.2; filled circles ≥ 0.2.

MODEL-BASED CLUSTERING SOFTWARE

- R code can be downloaded:

  www.stat.washington.edu/mclust

- Also available at the CRAN site
- Documentation and other technical reports can be downloaded:

  http://www.stat.washington.edu/fraley/mclust/reps.shtml

- MBC Toolbox in MATLAB
  - Written by Angel & Wendy Martinez
  - Soon to be available on the mclust page and Statlib

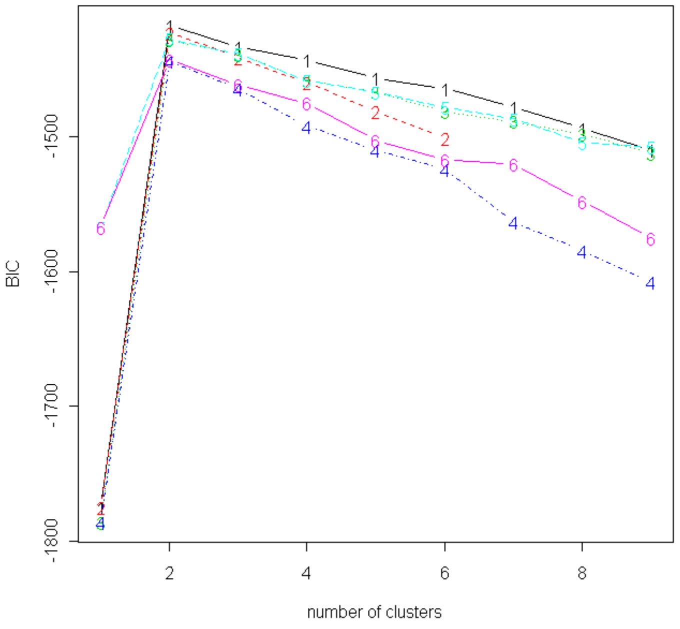
Model Based Clustering in R - Inputs

```
install.packages("mclust")


library(mclust)
```

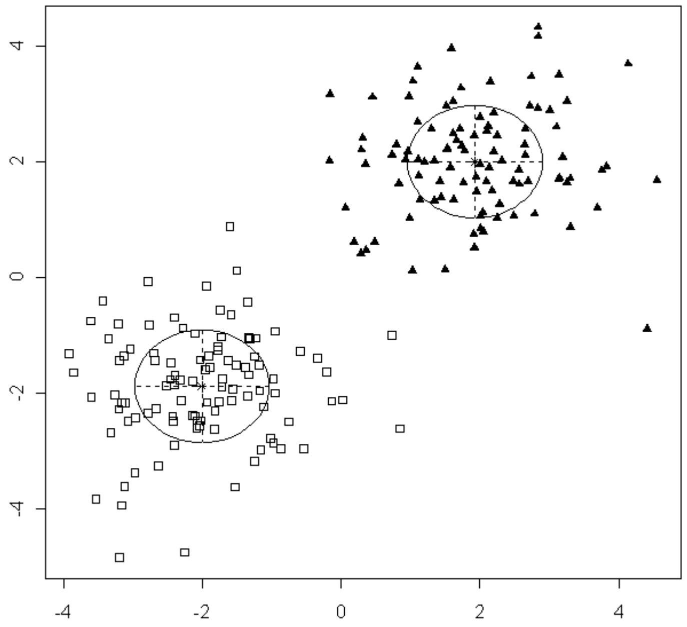
Mclust Applied to Our Data

```
> x.mclust = Mclust(x)
> summary(x.mclust)
```

Mclust Plots - I

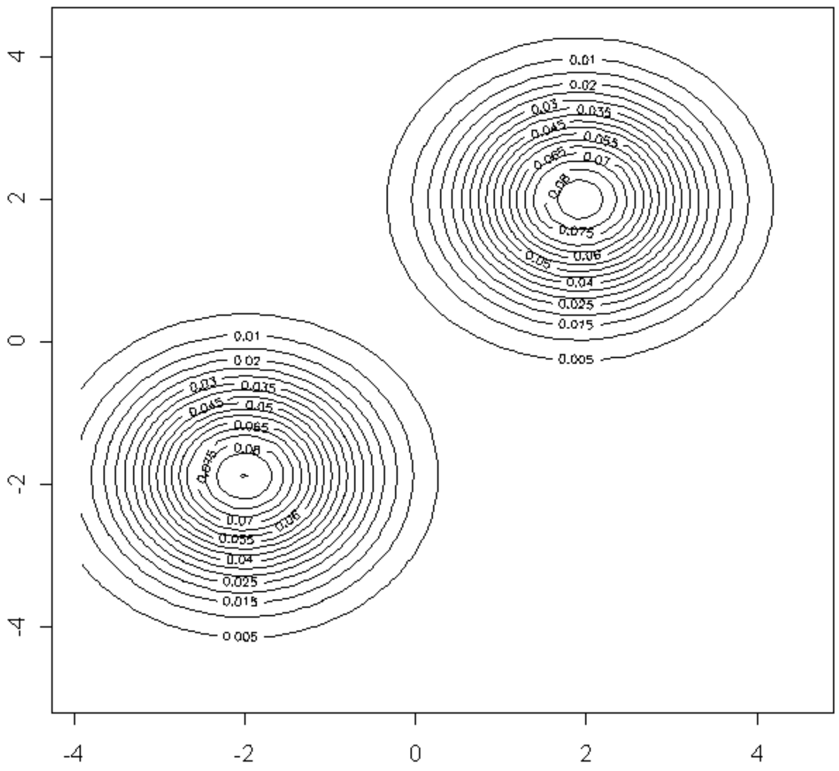


BIC



Model Fit Plots

Mclust Plots - II



4
2
0
-2
-4
-4
-2
0
2
4
0.01
0.02
0.03
0.035
0.045
0.055
0.065
0.07
0.08
0.075
0.06
0.05
0.04
0.025
0.015
0.005
0.01
0.02
0.03
0.035
0.045
0.05
0.065
0.08
0.075
0.07
0.06
0.055
0.04
0.025
0.015
0.005