Trimming Approach to Cluster Analysis

Pan Charoensuk
Songwan Joun

Stat 454/556 - Robust Statistics

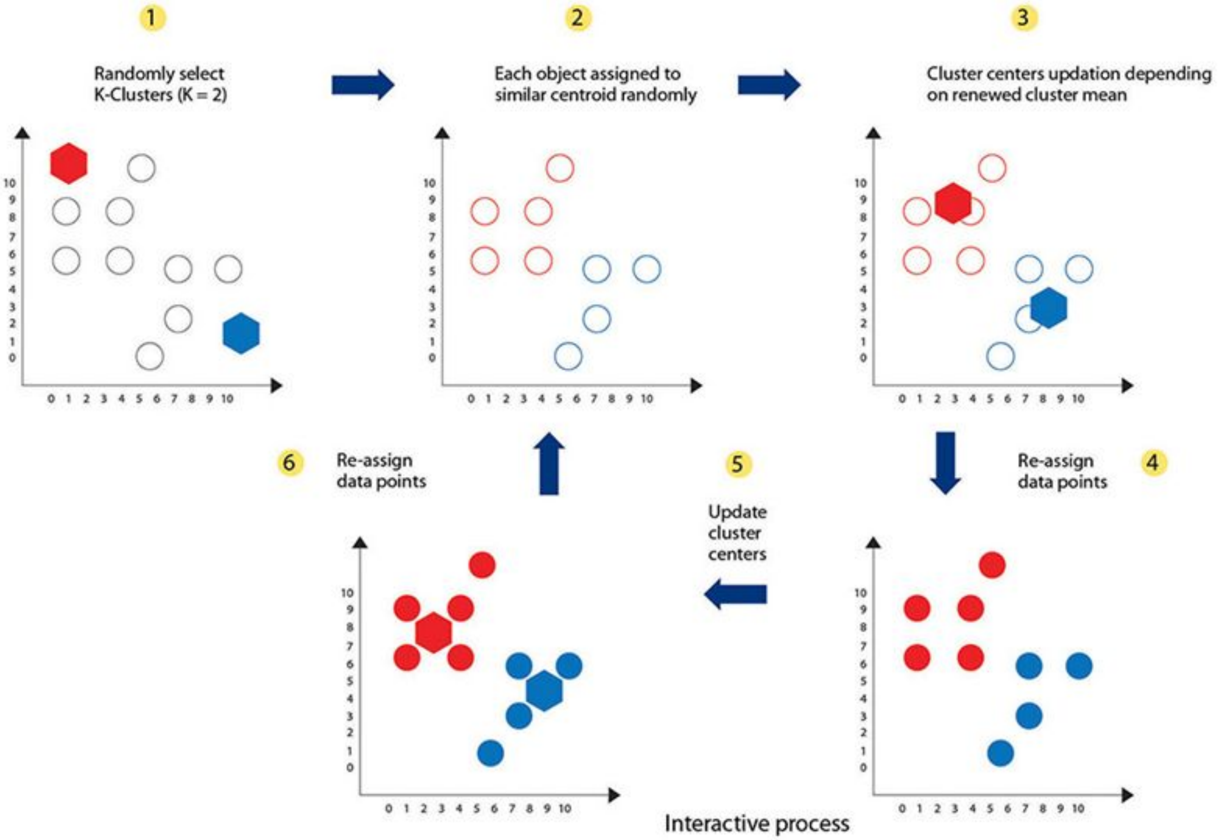
Department of Mathematics and Statistics
University of Victoria

Clustering

Clustering

- What is clustering?
- Unsupervised learning (Having no answer to the output)
- Algorithms: K-means, Trimmed K-means, TCLUST
- How about classification? (supervised learning)
- Example: Clustering customers by their shopping transaction data

K-means (Algorithm)

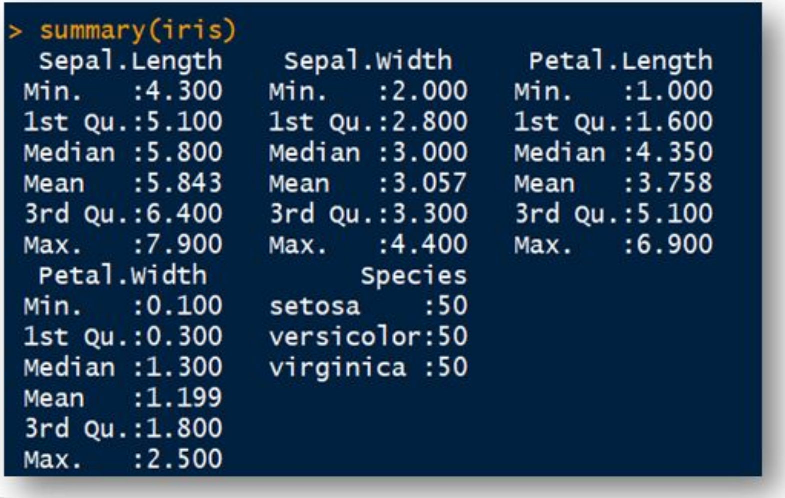

1
Randomly select K-Clusters (K = 2)
2
Each object assigned to similar centroid randomly
3
Cluster centers updation depending on renewed cluster mean
4
Re-assign data points
5
Update cluster centers
6
Re-assign data points
Interactive process
0 1 2 3 4 5 6 7 8 9 10
0 1 2 3 4 5 6 7 8 9 10

How it works?

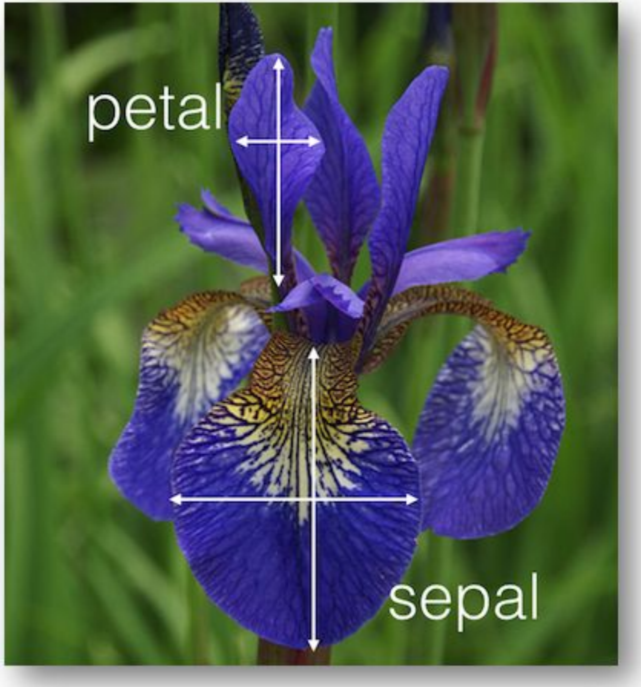
Iris dataset

dataset

- 150 observations
- 5 variables (Sepal.Length, Sepal.Width, Petal.Length, Petal.Width, Species)



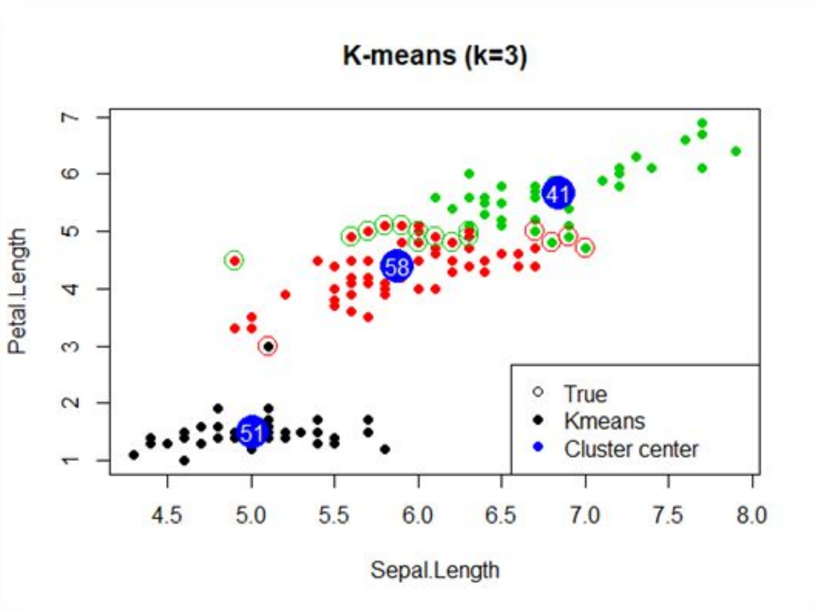
```
> summary(iris)
  Sepal.Length    Sepal.Width     Petal.Length  
 Min.   :4.300   Min.   :2.000   Min.   :1.000  
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600  
 Median :5.800   Median :3.000   Median :4.350  
 Mean   :5.843   Mean   :3.057   Mean   :3.758  
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100  
 Max.   :7.900   Max.   :4.400   Max.   :6.900  
  Petal.Width          Species  
 Min.   :0.100   setosa    :50  
 1st Qu.:0.300   versicolor:50  
 Median :1.300   virginica :50  
 Mean   :1.199                  
 3rd Qu.:1.800                  
 Max.   :2.500                  
```



K-means on Iris dataset

Result

- k: number of clusters = 3
- Empty circle : True Species
- Filled circle : k-means cluster results
- number of points within each cluster is similar
- A lot of misclassification within green cluster and red cluster (bridge region)
- misclassification rate = 0.12



Things to consider

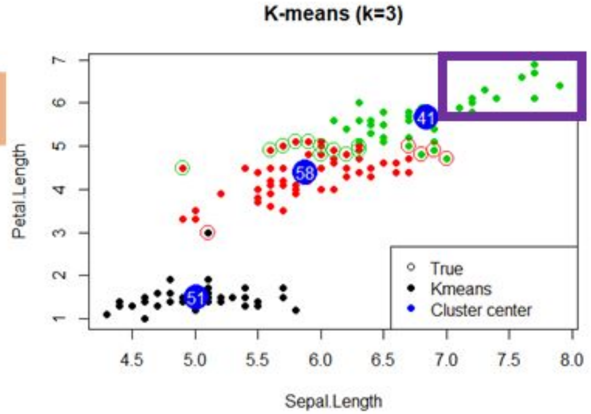
2 points to consider

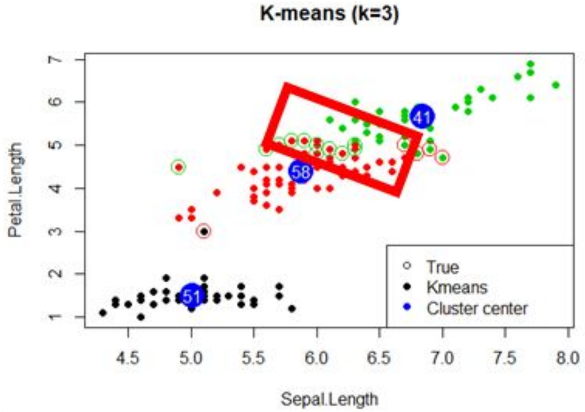
1. How about **points** that are far from clusters?

- Should we treat them as an outlier? or assign point to the closest region?
- What if there are group of far points which has some meanings?

2. There are many misclassification on **bridge region**

- Should we build a complex model for this region?
- (e.g, mixture model)
- Or should we treat them as outliers?





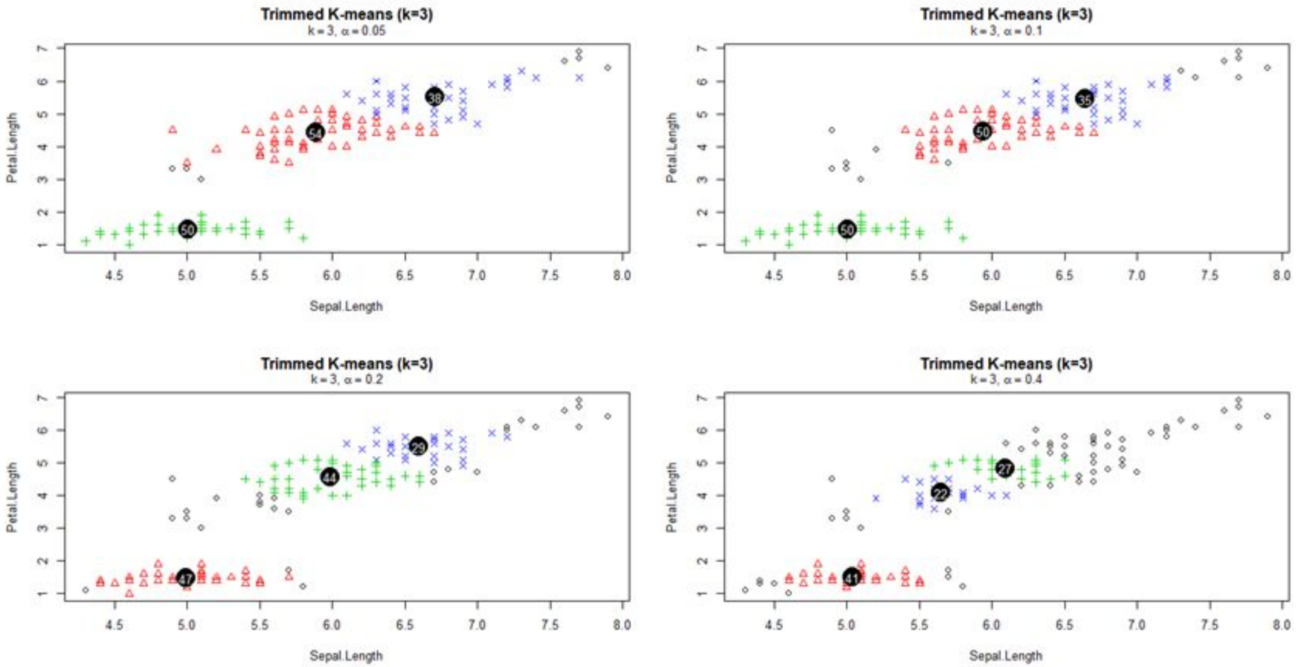
Decisions

Decisions

- Bridge points" lying between clusters ought to be trimmed.
- Take the whole data structure into account (by likelihood function) and decide which parts of the sample should be discarded.
- Define outliers among spurious("non-regular") observations
- Assume certain sensible assumptions for the "non-regular" distributions.
- The idea is that we want to maximize the likelihood model considering 'regular observations' and 'non-regular' observations.

Trimming α%

Iris dataset, trimmed K-means



Limitations to K-means and Trimmed K-means

Limitations

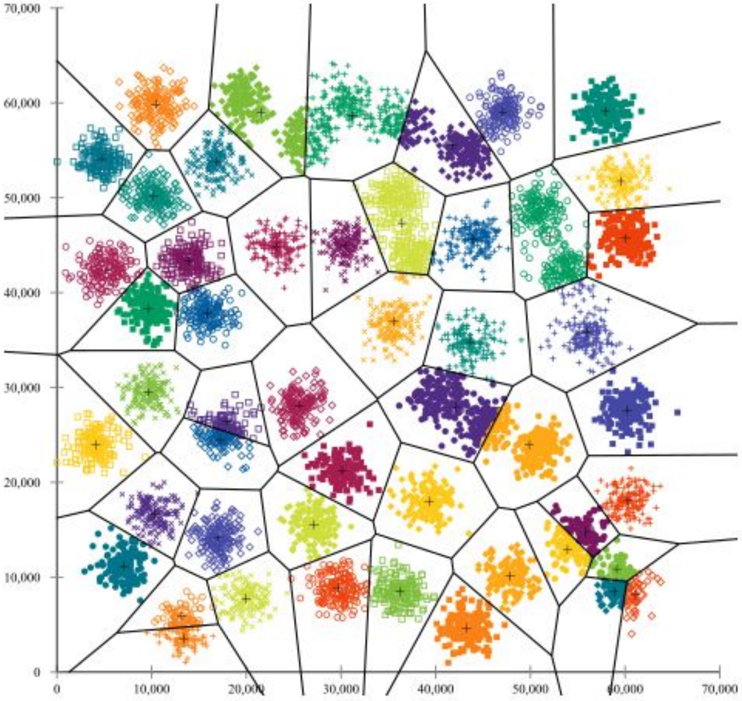
- Number of clusters, k
- SSE is the right objective to minimize
- Every cluster has the same shape
- Every observation is equally important for each cluster

- Note: Even with nice data for K-means, eg. all the assumptions hold. The classical algorithm could get stuck in local minima

Example

Local minima

- K-means stuck in local minima



Example taken from:
https://stats.stackexchange.com/questions/133656/how-to-understand-the-drawbacks-of-k-means

Problem

Problem and our approach

- Could we improve on the classical K-means and Trimmed K-means algorithms so that the clusters are approximately optimal?

Our approach:

- We will apply TCLUST algorithm on multidimensional data, following multivariate normal distribution and compare with K-means and Trimmed K-means algorithm using the misclassification rates.

TCLUST

TCLUST algorithm

TCLUST allows us to modify the scatter matrix of the cluster by putting a constraint on

- Relative size of the axes of clusters
- Relative volumes of clusters

- Scatter matrices with roughly equal volumes are also achievable, using TCLUST function. Note that this will give us our classical K-means and Trimmed K-means.

TCLUST

constraints on the scatter matrices

- TCLUST implements different algorithms aimed at approximately maximizing the likelihood function under different types of constraints applied on the scatter matrices.

- We need these constraints since maximizing the likelihood function without any restriction is not a well-defined problem. An almost degenerated scatter matrix would cause maximized log-likelihood to go to infinity. The algorithm would, then, end up finding spurious clusters almost lying in lower-dimensional subspaces

Strength of constraints

restr.fact

- restr.fact is fixed value greater than 0 which determines the strength of the constraint in TCLUST function.
  - The larger restr.fact, the looser is the restriction on the scatter matrices, allowing more heterogeneity among clusters.
  - The closer restr.fact to 1, the more equally scattered are the clusters

- The usage of TCLUST function is

  R > tclust(x, k, alpha, restr = c("eigen", "deter", "sigma"), restr.fact, equal.weights)

Types of constraint

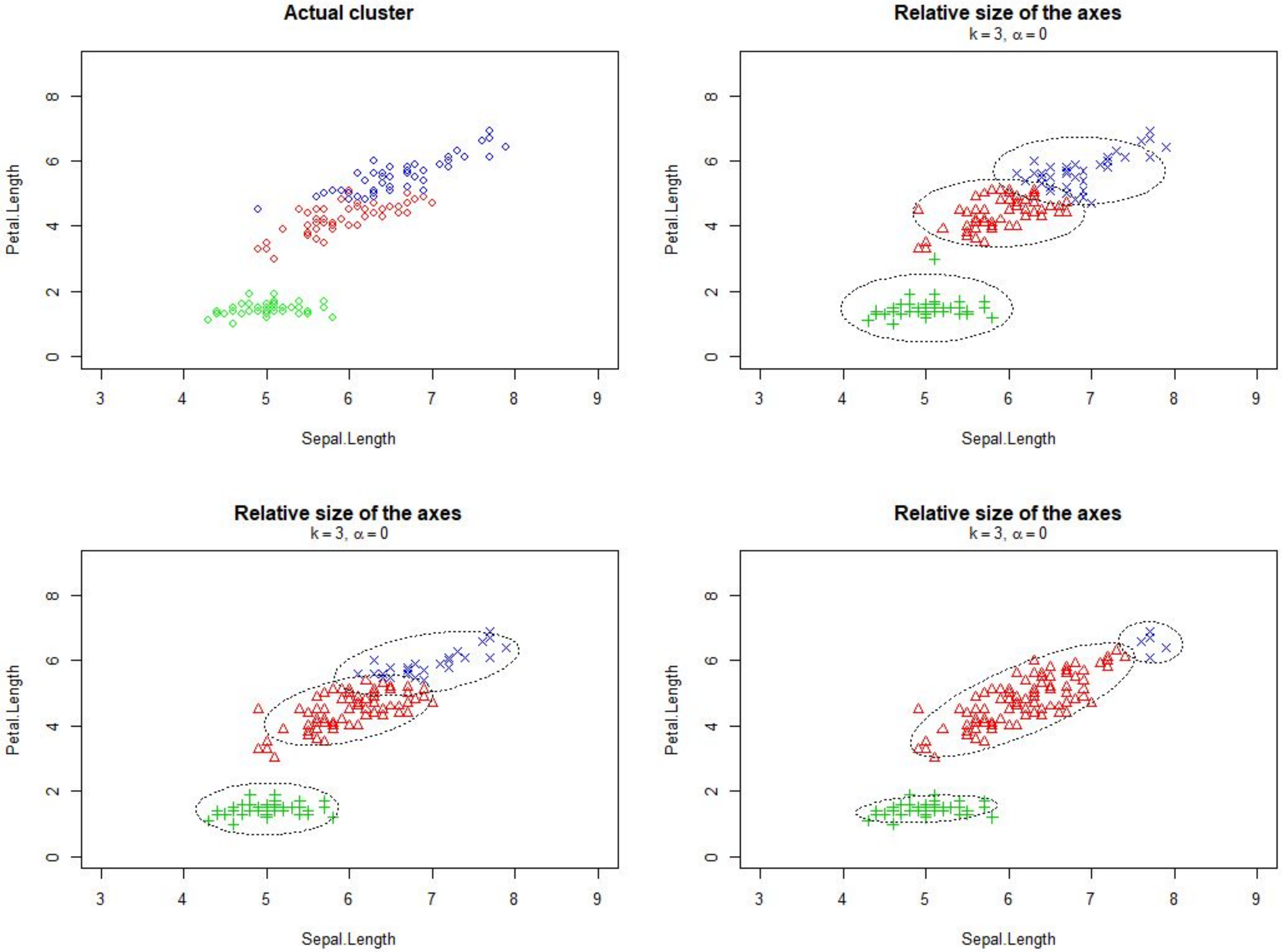
Constraints on Eigenvalues

- Let $\lambda_l(\boldsymbol{\Sigma}_j)$ be the eigenvalues of the cluster scatter matrices $\boldsymbol{\Sigma}_j$

- Let $M_n = \max_{j=1,\dots,k} \max_{l=1,\dots,p} \lambda_l(\boldsymbol{\Sigma}_j)$ and $m_n = \min_{j=1,\dots,k} \min_{l=1,\dots,p} \lambda_l(\boldsymbol{\Sigma}_j)$. Then,

$$\mathrm{M_n}/\mathrm{m_n} \leq restr.fact$$

- This constraint is achieved when we set restr = “eigen”.
- Constraining the eigenvalues allows us to simultaneously control the relative group sizes and also the deviation from sphericity in each cluster.



Actual cluster

Relative size of the axes

k = 3, α = 0

Relative size of the axes

k = 3, α = 0

Relative size of the axes

k = 3, α = 0

Types of constraint

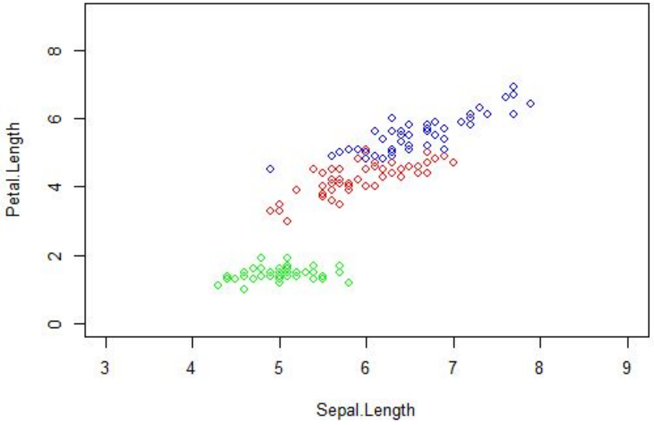
Constraints on Determinants

- Let $M_n = \max_{j=1,\ldots,k} |\boldsymbol{\Sigma}_j|$ and $m_n = \underset{j=1,\ldots,k}{min} |\boldsymbol{\Sigma}_j|$. Then,

$$\mathrm{M_n}/\mathrm{m_n} \leq restr.fact$$

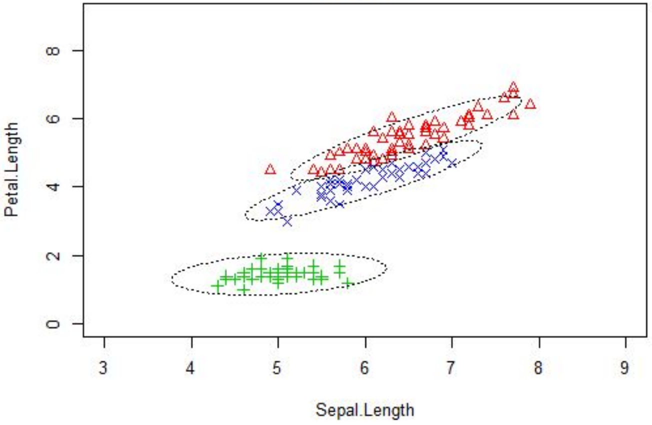
- This type of constraint is done by setting restr = “deter”
- Contrainting determinants limits the relative volumes of the clusters
- The use of this type of constraint is particularly advisable when affine equivariance is required.

Actual cluster



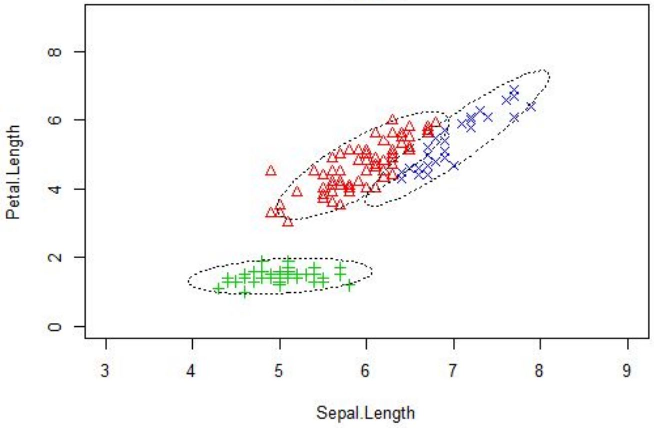
Relative volumnes of the axes

$k = 3, \alpha = 0$



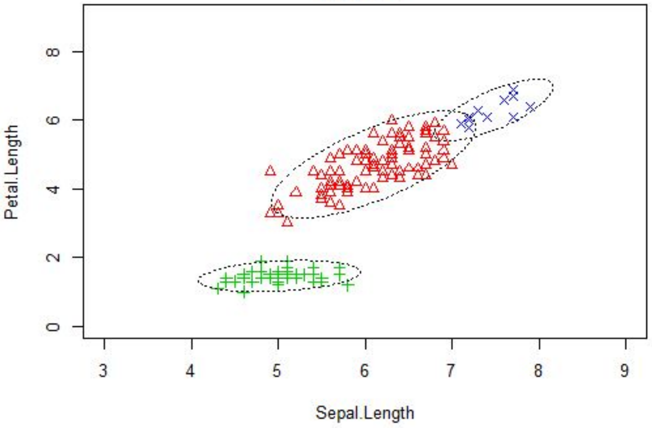
Relative volumnes of the axes

$k = 3, \alpha = 0$



Relative volumnes of the axes

$k = 3, \alpha = 0$

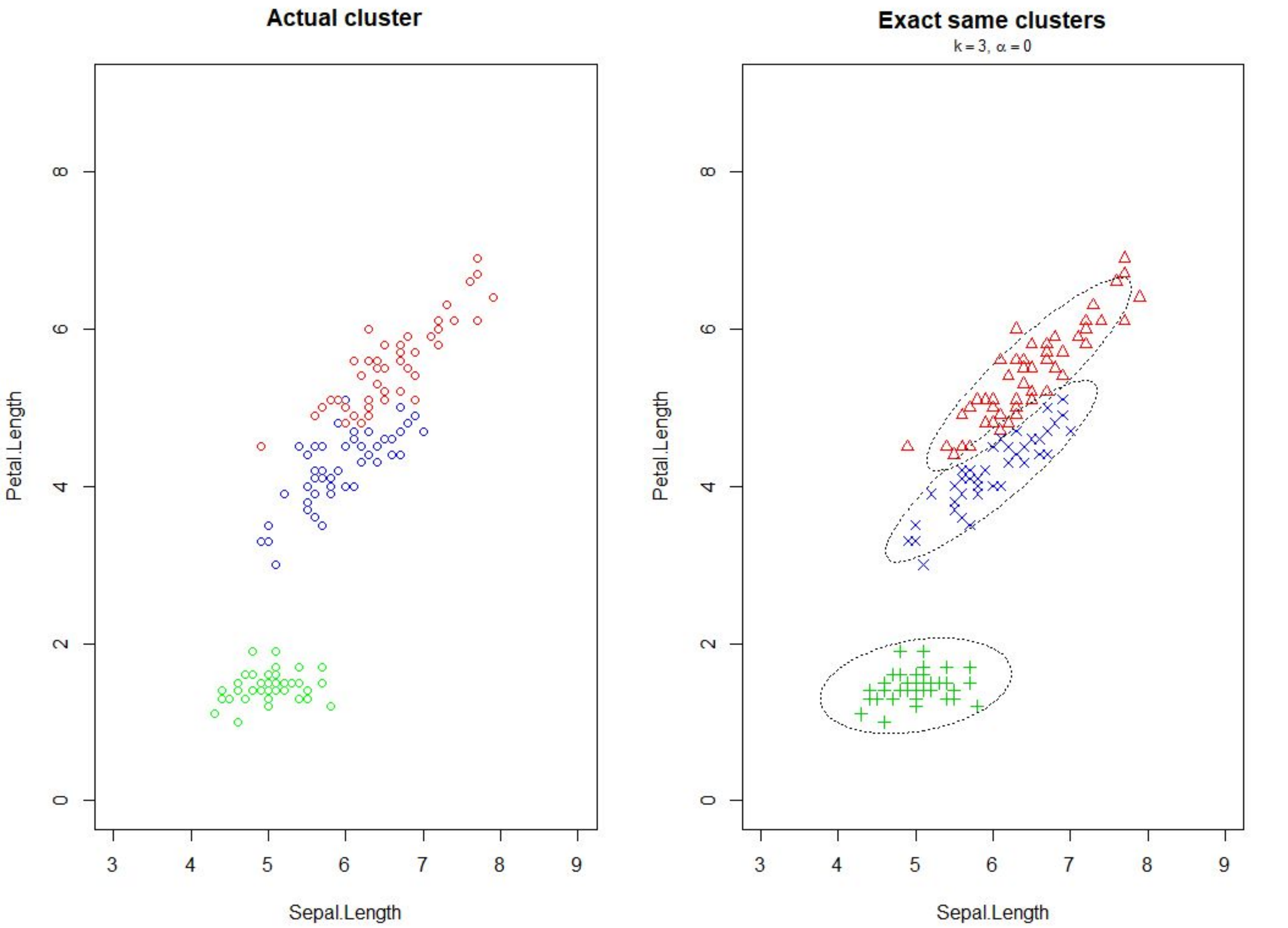


Same scatter matrices

- Equal scatter matrices
- Setting restr = sigma forces all cluster scatter matrices to be the same.

$$\boldsymbol{\Sigma}_1 = \ldots = \boldsymbol{\Sigma}_k$$

- restr.fact is ignored when applying this type of constraint


Actual cluster
Exact same clusters
k = 3, α = 0
Petal.Length
Sepal.Length
0
2
4
6
8
3
4
5
6
7
8
9

Questions

Questions

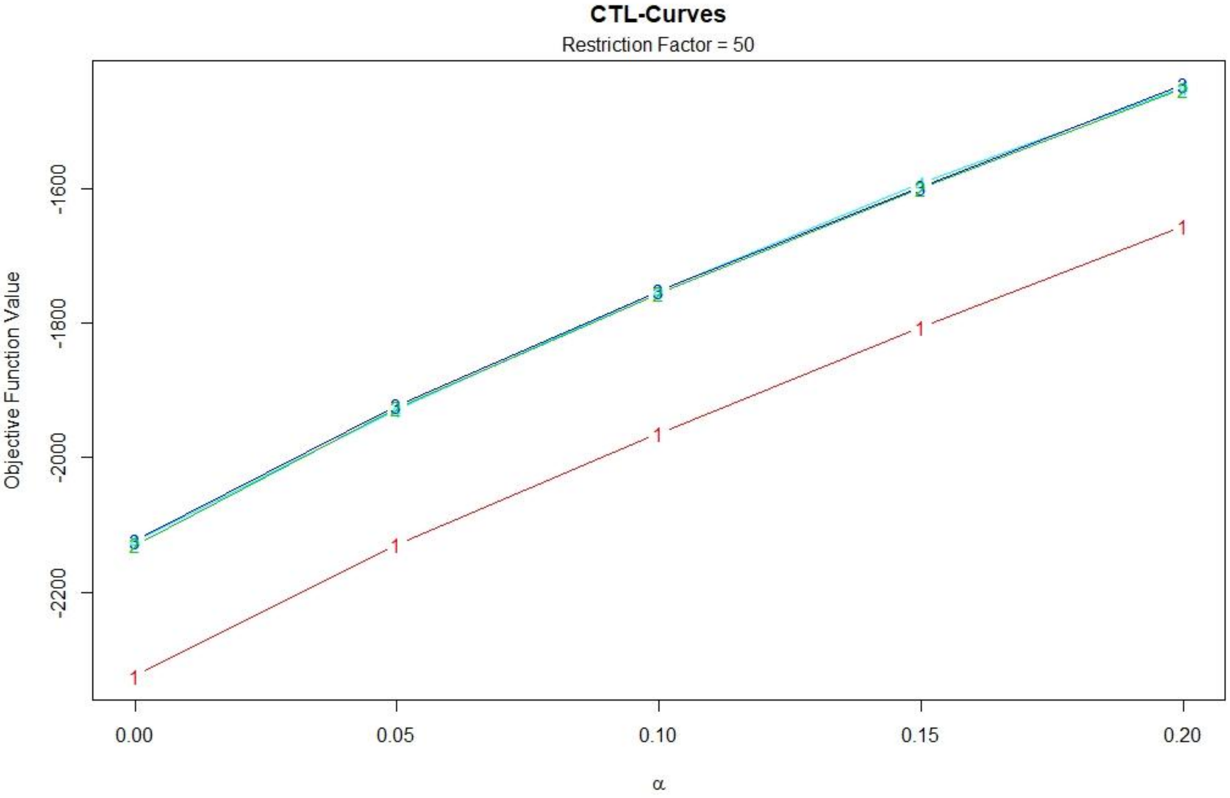
- How do we find optimal k value?
- How do we set optimal α?

Choosing k

Which k should we use?

- One of the most difficult problem in clustering is choosing the number of clusters, k.
- k is often unknown.
- Trimming proportion α is dependent on k
- CTL-Curve(classification trimmed likelihood curve)

CTL-Curve example



Warning

Warning

- The obtained values for k and α and their associated clustering solutions must be explored carefully.
- Algorithm gives a warning if the ratio exceeds the upper bound. In which case, the upper bound may be increased stepwise until the warning disappears.
- TCLUST outputs point out which solutions are artificial. This allows us to easily search for clustering solutions which are not artificially restricted, if desired.

Methodology for eigenvalue ratio constraint

Initial settings

- Sample of observations : $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ in $\boldsymbol{R}^p$
- pdf of p-variate normal distribution : $\phi(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$
- trimming level : α
- number of clusters : k

Goal

- Goal : search for a partition $R_0, R_1, \ldots, R_k$ of the indices {1, …, n} with $\#R_0 = [n\alpha]$, centers $\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k$ in $\boldsymbol{R}^p$, symmetric positive semidefinite pxp scatter matrices $\boldsymbol{S}_1, \ldots, \boldsymbol{S}_k$ and weights $p_1, \ldots, p_k$ with $p_i \in [0,1]$ and $\sum_{j=1}^{k} p_j = 1$ which maximizes the objective function

- Objective function : $\sum_{j=1}^{k} \sum_{i \in R_j} \log(p_j \phi(\boldsymbol{x}_i; \boldsymbol{\mu}_j, \boldsymbol{S}_j))$

  - Sum of weighted log normal pdf for each observation, giving same weight if cluster is same

Example

Example

- (Given) Sample of observations : $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_{10}\}$ in $\boldsymbol{R}^2$

x1	9	2	3	4	6	1	7	8	5	10
x2	0	2	1	5	9	4	6	7	3	10

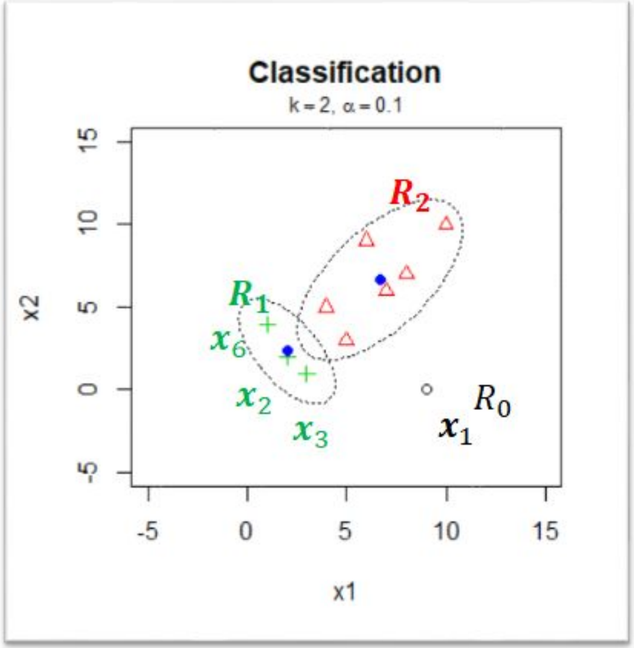
- (Given) pdf of 2-variate normal distribution : $\phi(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$
- trimming level : $\alpha = 0.1$ → $[n\alpha] = [10*0.1] = 1$ (trim one observation, $\#R_0 = 1$)
- number of clusters : $k = 2$

- Object function : $\sum_{i \in R_1} \log(p_1 \phi(\boldsymbol{x}_i; \boldsymbol{\mu}_1, \boldsymbol{S}_1)) + \sum_{i \in R_2} \log(p_2 \phi(\boldsymbol{x}_i; \boldsymbol{\mu}_2, \boldsymbol{S}_2))$

→ Goal : find best R_0(for outliers), R_1, R_2 partition of the observation, centers $\boldsymbol{m}_1$, $\boldsymbol{m}_2$, 2x2 symmetric positive semidefinite scatter matrices $\boldsymbol{S}_1, \boldsymbol{S}_2$, and weights p_1, p_2.

Example

Example



- Through TCLUST algorithm, we can assign 9 observations to 2 clusters with α = 0.1
- TCLUST assigned $\boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_6 \sim \phi_1(\cdot; \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ to green cluster (R_1) and $\boldsymbol{x}_4, \boldsymbol{x}_5, \boldsymbol{x}_7, \boldsymbol{x}_8, \boldsymbol{x}_9 \sim \phi_2(\cdot; \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ to red cluster (R_2)
- $\boldsymbol{x}_1$ is selected as an outlier (R_0)
- This cluster assignment maximizes the objective function similar to

$$\sum_{i \in R_1} \log(p_1 \phi(\boldsymbol{x}_i; \boldsymbol{\mu}_1, \boldsymbol{S}_1)) + \sum_{i \in R_2} \log(p_2 \phi(\boldsymbol{x}_i; \boldsymbol{\mu}_2, \boldsymbol{S}_2))$$

- Here, $\boldsymbol{\mu}_1 = (2.3, 2.0)', \boldsymbol{\mu}_2 = (6.6, 6.6)', p_1 = 0.3, p_2 = 0.7$
- $\boldsymbol{S}_1 = \begin{bmatrix} 1.0 & -0.8 \\ -0.8 & 1.7 \end{bmatrix}$ and $\boldsymbol{S}_2 = \begin{bmatrix} 2.8 & 2.1 \\ 2.1 & 3.9 \end{bmatrix}$

Problems

Problems of objective function : $\sum_{j=1}^{k}\sum_{i\in R_j}\log(p_j\phi(\boldsymbol{x}_i;\boldsymbol{\mu}_j,\boldsymbol{S}_j))$

- Maximization of objective function $\sum_{j=1}^{k}\sum_{i\in R_j}\log(p_j\phi(\mathbf{x}_\mathrm{i};\boldsymbol{\mu}_\mathrm{j},\mathbf{S}_\mathrm{j}))$ without any constraint on scatter matrices($\boldsymbol{S}_j$) is not a well defined problem
- Example: if $\boldsymbol{\mu}_j=\boldsymbol{x}_i$ and $\det(\boldsymbol{S}_j)\rightarrow 0$, then $\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_j,\boldsymbol{S}_j\right)$ is not defined

$$\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_j,\boldsymbol{S}_j\right)=\frac{ex\ p\left\{-\frac{1}{2}(\mathbf{x}_\mathrm{i}-\boldsymbol{\mu}_\mathrm{j})'\boldsymbol{S}_j^{-1}(\mathbf{x}_\mathrm{i}-\boldsymbol{\mu}_\mathrm{j})\right\}}{\sqrt{2\pi}^{\,p}\ \det(\boldsymbol{S}_j)^{\frac{1}{2}}} \rightarrow 1/0=\infty$$

- Note that $(\mathbf{x}_\mathrm{i}-\boldsymbol{\mu}_\mathrm{j})'\boldsymbol{S}_j^{-1}(\mathbf{x}_\mathrm{i}-\boldsymbol{\mu}_\mathrm{j})$ is called the Mahalanobis distance

Solutions

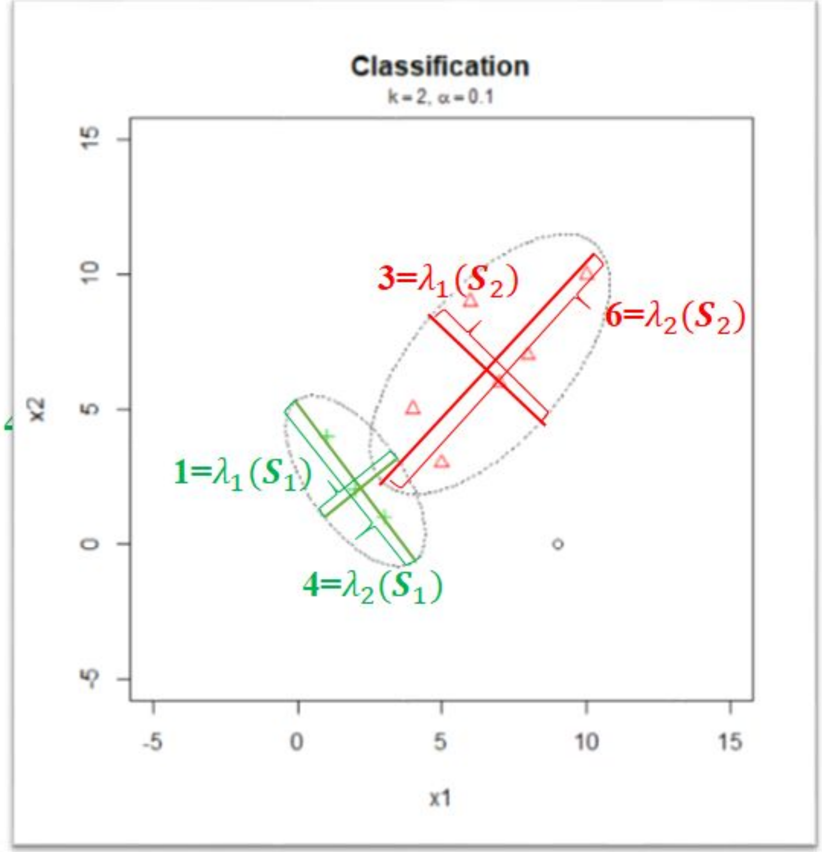
Solution (well defined problem)

- In order to make the maximization of the objective function well defined problem, consider an eigenvalue ratio constraint on the scatter matrices : $\boldsymbol{S}_1, \dots, \boldsymbol{S}_k$

$$\frac{\max_{j,l} \lambda_l(\boldsymbol{S}_j)}{\min_{j,l} \lambda_l(\boldsymbol{S}_j)} \leq c$$

- Here, $l = 1, \dots, p$ and $\lambda_l(\boldsymbol{S}_j)$ is the set of p eigenvalues of the scatter matrix $\boldsymbol{S}_j$.
- $c(\geq 1)$: a constant which controls the strength of the constraint

Example



- Suppose we have cluster $\boldsymbol{S}_1$ with $\lambda_1 = 1$, $\lambda_2 = 4$ and cluster $\boldsymbol{S}_2$ with $\lambda_1 = 3$, $\lambda_2 = 6$
- Here, $\max_{j,l} \lambda_l(\mathbf{S}_j) = \lambda_2(\mathbf{S}_2) = 6$
- $\min_{j,l} \lambda_l(\mathbf{S}_j) = \lambda_1(\mathbf{S}_1) = 1$
- Therefore, $\dfrac{\max_{j,l} \lambda_l(\boldsymbol{S}_j)}{\min_{j,l} \lambda_l(\boldsymbol{S}_j)}$ implies the ratio of longest axis and the smallest axis across clusters
- This implies that we should modify $\mathbf{S}_j$'s to meet our eigenvalue ratio constraints

Algorithm

Overview

- Algorithm approximately maximizing objective function $\sum_{j=1}^{k}\sum_{i\in R_j}\log(p_j\phi(\mathbf{x}_\mathrm{i};\boldsymbol{\mu}_\mathrm{j},\mathbf{S}_\mathrm{j}))$ under

eigenvalue ratio constraint $$\frac{\max\limits_{j,l}\lambda_l(\boldsymbol{S}_j)}{\min\limits_{j,l}\lambda_l(\boldsymbol{S}_j)}\le c$$

- Can be seen as a Classification EM algorithm, and more generally, generalized k-means algorithm
- An implementation of the algorithm is available through the R package "tclust"
- Whole steps are consist of 3 steps

Expectation Step
Concentration Step
Maximization Step

Expectation step

E - step: calculating posterior probabilities

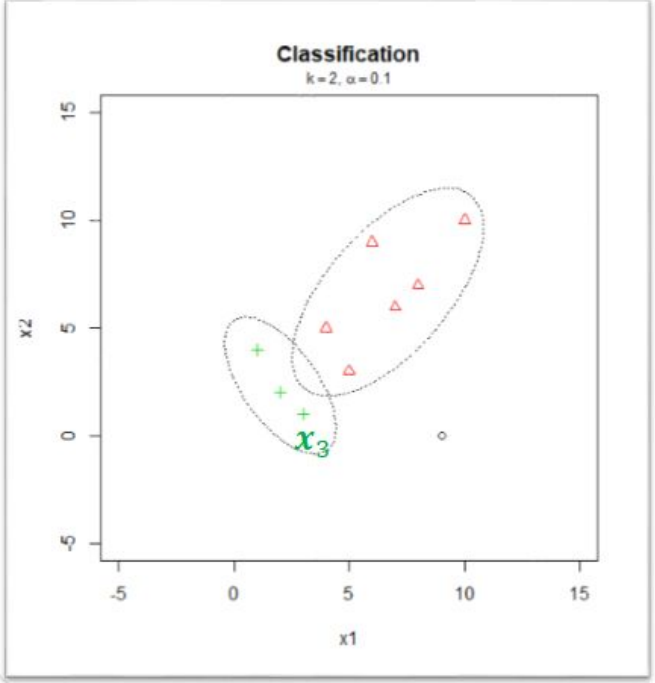
- For each observation $\mathbf{x}_\mathrm{i}$, and $D_j(\mathbf{x}_\mathrm{i};\theta) = p_j\phi(\mathbf{x}_\mathbf{i};\boldsymbol{\mu}_\mathrm{j},\mathbf{S}_\mathrm{j})$, the posterior probabilities:

$$\frac{D_j(\mathbf{x}_\mathrm{i};\theta)}{\sum_{j=1}^{k} D_j(\mathbf{x}_\mathrm{i};\theta)} \text{ for } j = 1, \dots, k,$$

- with $\theta = (p_1, \dots, p_k, \boldsymbol{m}_1, \dots, \boldsymbol{m}_k, \boldsymbol{S}_1, \dots, \boldsymbol{S}_k)$ as the set of cluster parameters in the current iteration of the algorithm
- Can think of posterior probabilities as normalized probabilities
- $D_j(\mathbf{x}_\mathrm{i};\theta)$: metric for the distance of an observation $\mathbf{x}_\mathrm{i}$ to the center of cluster j ($\boldsymbol{\mu}_\mathrm{j}$)
  - If $D_j(\mathbf{x}_\mathrm{i};\theta)$ small, distance of the $\mathbf{x}_\mathrm{i}$ to $\boldsymbol{\mu}_\mathrm{j}$ is large
  - Define an overall measure for outlyingness

Expectation step

Example



- For example, let's look at the observation $\boldsymbol{x}_3$
- Suppose p_1, p_2 are randomly chosen and $\boldsymbol{\mu}_1$, $\mathbf{S}_1$, $\boldsymbol{\mu}_2$, $\mathbf{S}_2$ are given
- Then, for each cluster, we can calculate $D_1(\mathbf{x}_3;\theta) = p_1\phi(\mathbf{x}_1;\boldsymbol{\mu}_1,\mathbf{S}_1)$ and $D_2(\mathbf{x}_3;\theta) = p_2\phi(\mathbf{x}_2;\boldsymbol{\mu}_2,\mathbf{S}_2)$
- Therefore, posterior probabilities are

$$P_{1,post}(\boldsymbol{x}_3) = \frac{D_1(\mathbf{x}_3;\theta)}{\sum_{j=1}^{2} D_j(\mathbf{x}_3;\theta)} = \frac{p_1\phi(\mathbf{x}_3;\boldsymbol{\mu}_1,\mathbf{S}_1)}{p_1\phi(\mathbf{x}_3;\boldsymbol{\mu}_1,\mathbf{S}_1)+p_2\phi(\mathbf{x}_3;\boldsymbol{\mu}_2,\mathbf{S}_2)} \text{ and}$$

$$P_{2,post}(\boldsymbol{x}_3) = \frac{D_2(\mathbf{x}_3;\theta)}{\sum_{j=1}^{2} D_j(\mathbf{x}_3;\theta)} = \frac{p_2\phi(\mathbf{x}_3;\boldsymbol{\mu}_2,\mathbf{S}_2)}{p_1\phi(\mathbf{x}_3;\boldsymbol{\mu}_1,\mathbf{S}_1)+p_2\phi(\mathbf{x}_3;\boldsymbol{\mu}_2,\mathbf{S}_2)} \text{ each}$$

- It is easy to see that $P_{j,post}(\boldsymbol{x}_i) \propto D_j(\mathbf{x}_\text{i};\theta)$

Concentration step & Maximization step

C - step: trimming [nα] observations

- Each non-trimmed observation $\mathbf{x}_\mathrm{i}$ will be assigned to the cluster which provides maximum posterior probability
- In order to implement the trimming procedure, the [nα] observations $\mathbf{x}_\mathrm{i}$ with smallest values of

$$\mathrm{D}(\mathbf{x}_\mathrm{i};\theta) = \max\{D_1(\mathbf{x}_\mathrm{i};\theta), \dots, D_k(\mathbf{x}_\mathrm{i};\theta)\}$$

are discarded as possible outliers

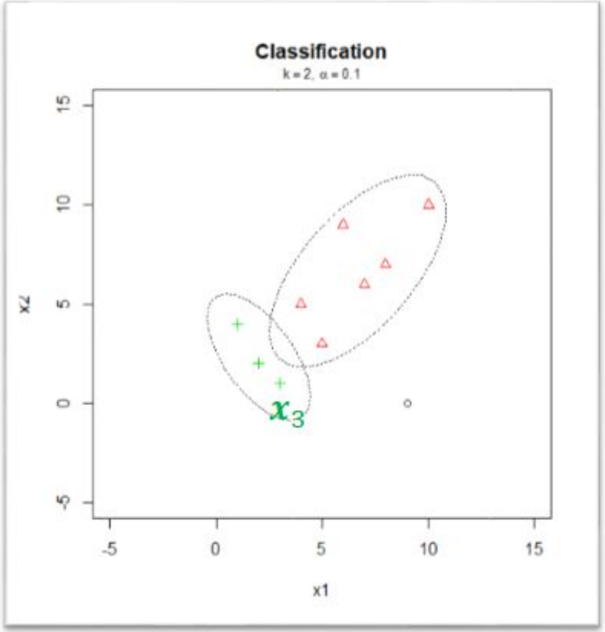
- If k=1, $\max\{D_1(\mathbf{x}_\mathrm{i};\theta), \dots, D_k(\mathbf{x}_\mathrm{i};\theta)\} = \mathrm{D}(\mathbf{x}_\mathrm{i};\theta)$
  = smallest Mahalanobis distances $(\mathbf{x}_\mathrm{i} - \boldsymbol{\mu}_\mathrm{j})' \boldsymbol{S}_j^{-1} (\mathbf{x}_\mathrm{i} - \boldsymbol{\mu}_\mathrm{j})$

M - step: update parameters

- The parameters are updated, based on non-discarded observations and their cluster assignments.
- It is crucial to properly enforce the constraints on the cluster scatter matrices

Concentration step & Maximization step

Example



- For the observation x_3, we have $D_1(\mathbf{x}_3;\theta) = p_1\phi(\mathbf{x_1};\boldsymbol{\mu}_1,\mathbf{S}_1)$ and $D_2(\mathbf{x}_3;\theta) = p_2\phi(\mathbf{x_2};\boldsymbol{\mu}_2,\mathbf{S}_2)$
- Here, $\mathrm{D}(\mathbf{x}_3;\theta) = \max\{D_1(\mathbf{x}_3;\theta), D_2(\mathbf{x}_3;\theta)\}$

$$\propto \max\{P_{1,post}(\boldsymbol{x}_3), P_{2,post}(\boldsymbol{x}_3)\}$$

- x_3 will be assigned to the cluster which provides maximum posterior probability
- After calculating $\mathrm{D}(\mathbf{x}_\mathrm{i};\theta)\,'s$ for every observation $\mathrm{i} = 1, \dots, \mathrm{n}$, $[\mathrm{n}\alpha]$ observations $\mathbf{x}_\mathrm{i}$ with smallest values of $\mathrm{D}(\mathbf{x}_\mathrm{i};\theta)$ will be discarded as possible outliers
- (M-step) Update parameters based on non-discarded observations and new cluster assignment

Detailed Steps

1. Initialization

- $k \times (p+1)$ observations were randomly selected
- compute k cluster centers $\boldsymbol{m}_j^{\mathbf{0}}$ and k scatter matrices $\boldsymbol{S}_j^{\mathbf{0}}$ from the chosen data points
- The cluster scatter matrix constraints are applied to these $\boldsymbol{S}_j^{\mathbf{0}}$
- Weights $p_1^0, \dots, p_k^0$ in the interval (0,1) and summing to 1 are also randomly chosen
- The procedure is initialized *nstart* times by selecting different $\theta^0 = \left(p_1^0, \dots, p_k^0, \boldsymbol{m}_1^{\mathbf{0}}, \dots, \boldsymbol{m}_k^{\mathbf{0}}, \boldsymbol{S}_1^{\mathbf{0}}, \dots, \boldsymbol{S}_k^{\mathbf{0}}\right)$

Detailed Steps

2. Concentration step

- The following steps are executed until convergence (i.e., $\theta^{l+1} = \theta^{l}$) or a maximum number of iterations *iter.max* is reached

2.1 Trimming and cluster assignments (E and C-steps)

- Based on the current parameters $\theta^{l} = \left(p_1^l, \dots, p_k^l, \boldsymbol{m}_1^l, \dots, \boldsymbol{m}_k^l, \boldsymbol{S}_1^l, \dots, \boldsymbol{S}_k^l\right)$, the [nα] observations with smallest values of $\mathrm{D}\left(\mathbf{x}_\mathrm{i}; \theta^{l}\right)$ are discarded
- Each remaining observation $\mathbf{x}_\mathrm{i}$ is then assigned to a cluster j such that $\mathrm{D}_\mathrm{j}\left(\mathbf{x}_\mathrm{i}; \theta^{l}\right) = \mathrm{D}\left(\mathbf{x}_\mathrm{i}; \theta^{l}\right)$
- This yields a partition $R_0, R_1, \dots, R_k$ of {1, …, n} holding
    - indexes of the trimmed observations in R_0
    - indexes of the observations belonging to cluster j in $R_1, \dots, R_k$

Detailed Steps

2. Concentration step

- The following steps are executed until convergence (i.e., $\theta^{l+1} = \theta^{l}$) or a maximum number of iterations *iter.max* is reached

2.2 Update parameters (M-step)

- Given $n_j = \#R_j$, the weights are updated by $p_j^{l+1} = n_j/[n(1-\alpha)]$
- Centers are updated by the sample means $\boldsymbol{m}_j^{l+1} = \frac{1}{n_j} \sum_{i \in R_j} \mathbf{x}_\mathrm{i}$
- Scatter matrices are not updated by sample covariance matrices $\boldsymbol{T}_j = \frac{1}{n_j} \sum_{i \in R_j} (\mathbf{x}_\mathrm{i} - \boldsymbol{m}_j^{l+1})(\mathbf{x}_\mathrm{i} - \boldsymbol{m}_j^{l+1})'$
  - Because $\boldsymbol{T}_j$ may not satisfy the specified eigenvalue-ratio constraint
  - Instead, we use scatter matrices that are updated by truncated eigenvalues and spectral decomposition

1. Initialization | 2.1 Trimming and cluster assignment | **2.2 Update parameters** | 3. Evaluate target function

Detailed Steps

* Updating scatter matrices

- Apply spectral decomposition of $\boldsymbol{T}_j = \mathbf{U}_\mathrm{j}'\mathbf{D}_\mathrm{j}\mathbf{U}_\mathrm{j}$ where $\mathbf{U}_\mathrm{j}$ being an orthogonal matrix and $\mathbf{D}_\mathrm{j} = diag(d_{j1}, d_{j2}, \dots, d_{jp})$ a diagonal matrix consist of eigenvalues
- Let us consider truncated eigenvalues

$$d_{\mathrm{jl}}^m = \begin{cases} d_{jl} & if\ d_{jl} \in [m, cm] \\ m & if\ d_{jl} < m \\ cm & if\, d_{jl} > cm \end{cases}$$

with m as some threshold value

- The scatter matrices are updated as $\boldsymbol{S}_j^{l+1} = \mathbf{U}_\mathrm{j}'\mathbf{D}_\mathrm{j}^*\mathbf{U}_\mathrm{j}$ with $\mathbf{D}_\mathrm{j}^* = diag(d_{j1}^{m_{opt}}, d_{j2}^{m_{opt}}, \dots, d_{jp}^{m_{opt}})$ and m_{opt} minimizing

$$m \mapsto \sum_{j=1}^{k} n_j \sum_{l=1}^{p} \left(\log(d_{jl}^m) + \frac{d_{jl}}{d_{jl}^m} \right)$$

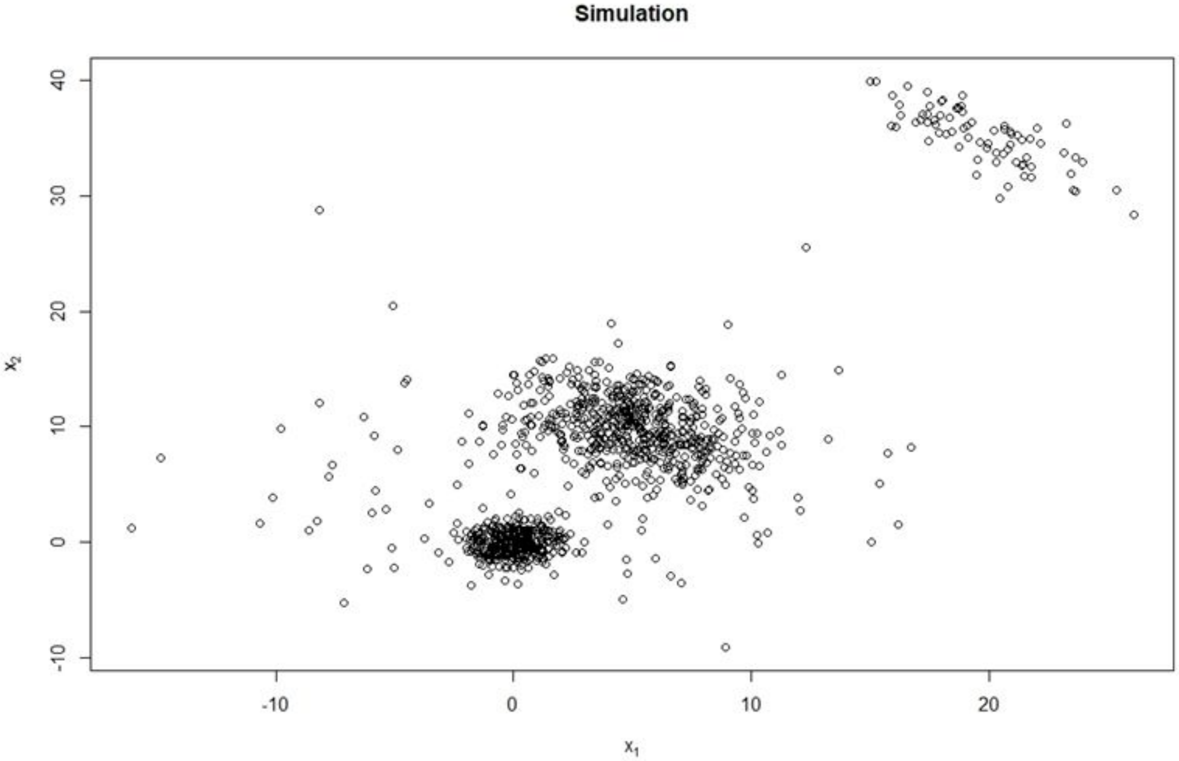
Detailed Steps

3. Evaluate target function

- After the concentration steps, the value of the target function $\sum_{j=1}^{k} \sum_{i \in R_j} \log(p_j \phi(\mathbf{x}_\mathrm{i}; \boldsymbol{\mu}_\mathrm{j}, \mathbf{S}_\mathrm{j}))$ is computed
- The parameters yielding the highest value of this target function are returned as the algorithm's output

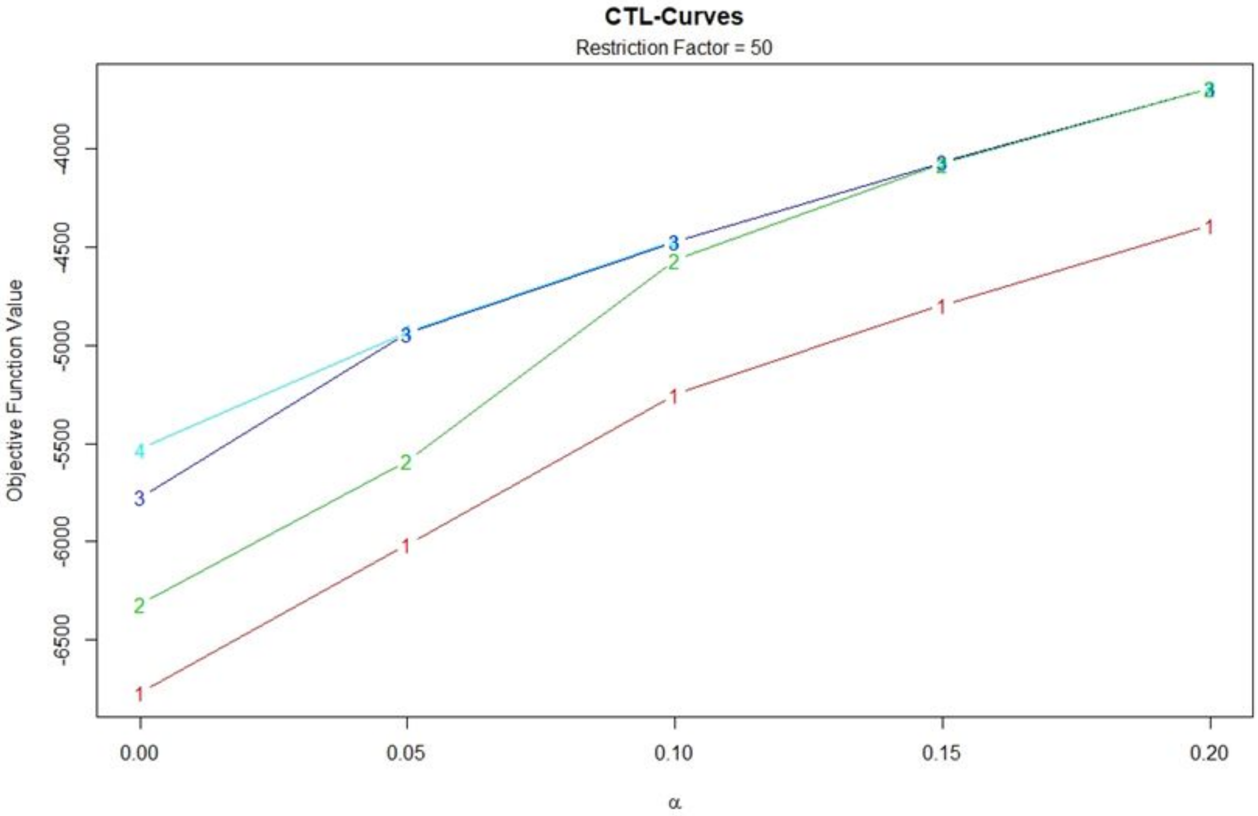
Simulation

- Consider the data randomly generated from a normal distribution



Simulation

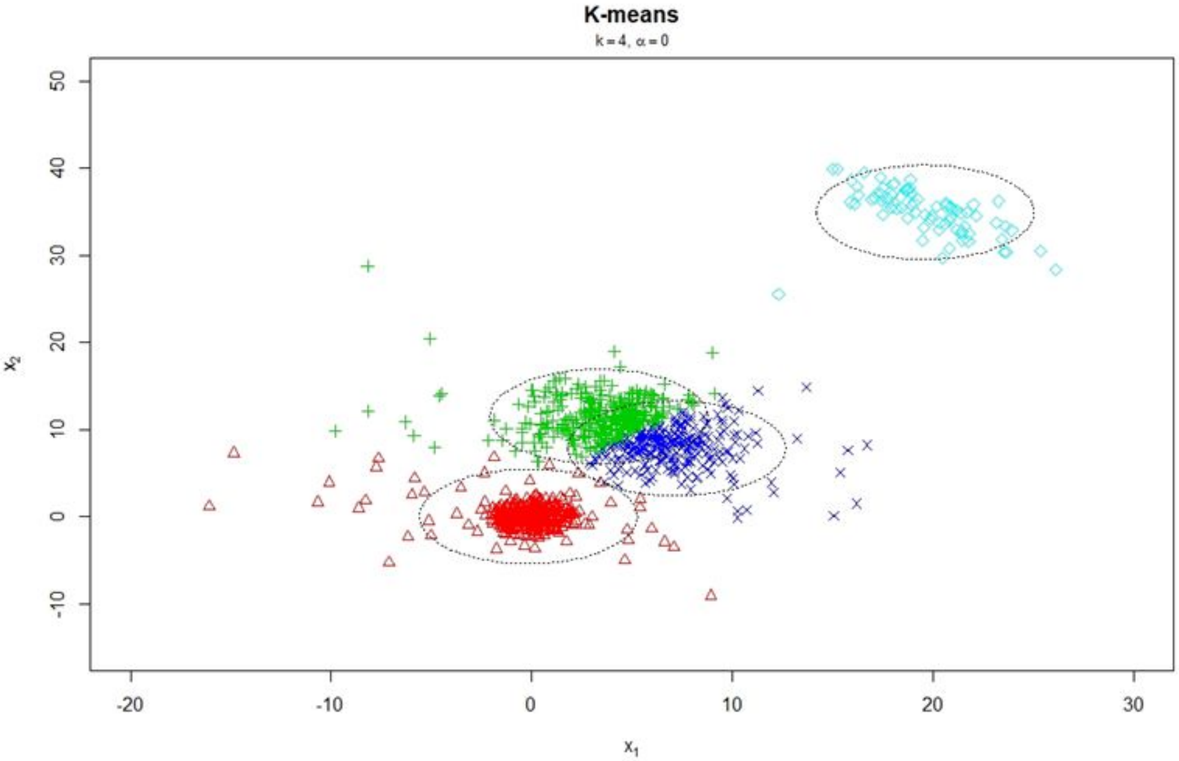
- Try to find reasonable k and α by looking at CTL-Curves



Simulation

- First, we will attempt to cluster our data using K-means.

- looks reasonable? improve?



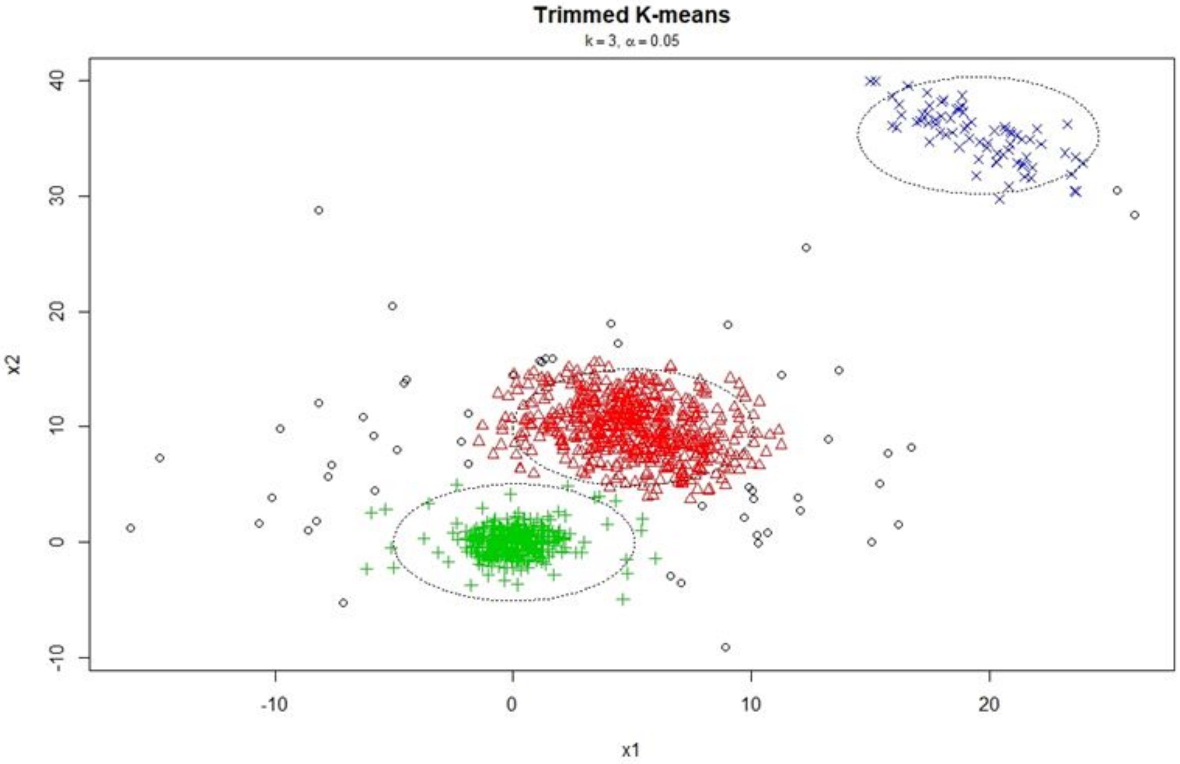
Simulation

Now, we will try trimming by

$\alpha = 0.05$.

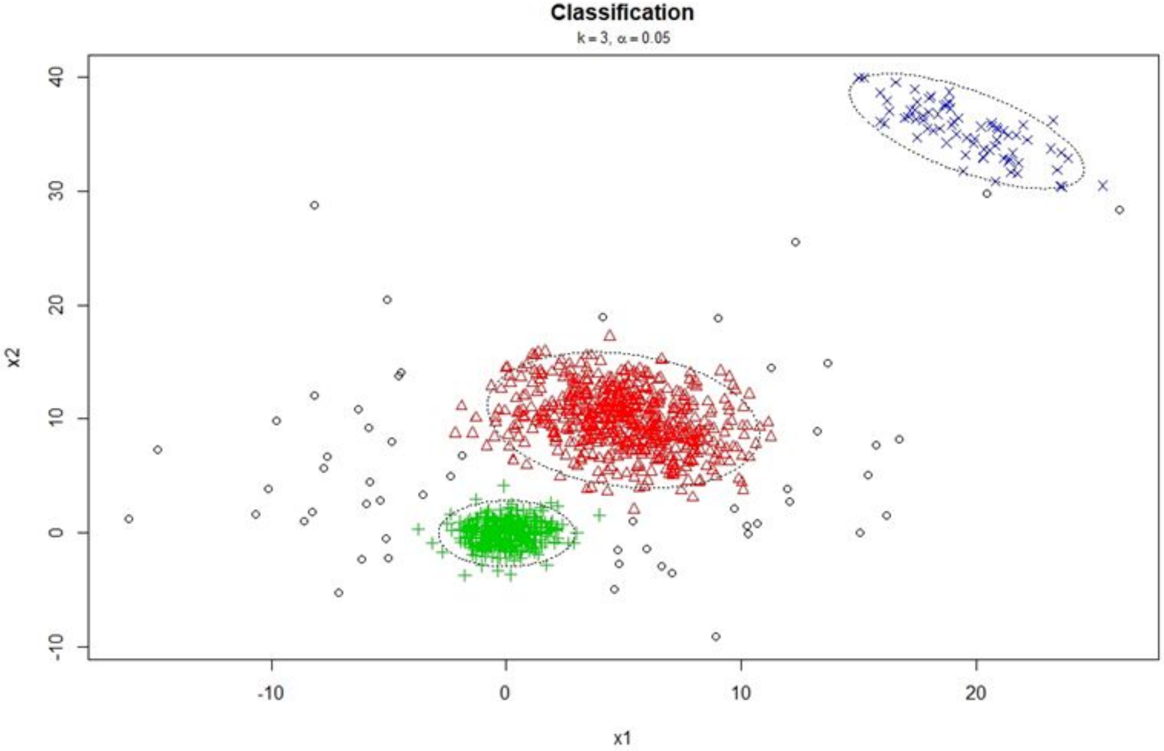
Note: CTL-Curves

suggest that we use k = 3



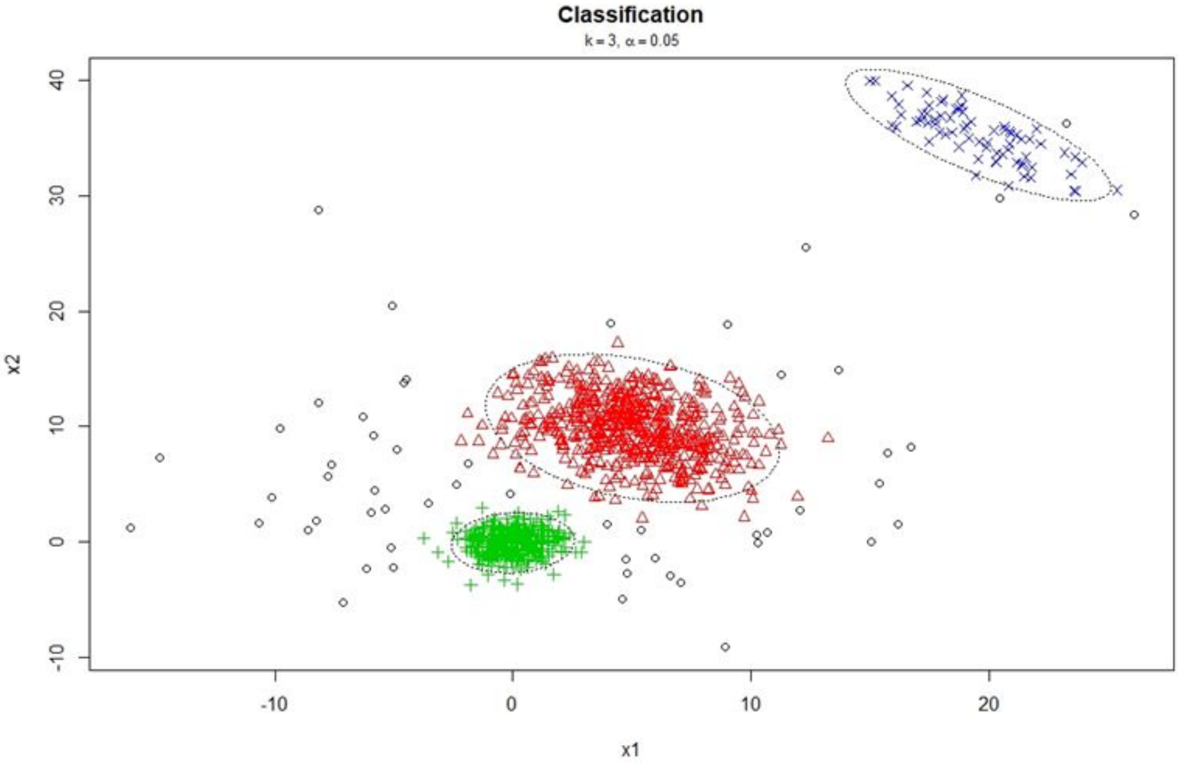
Simulation

And finally, we will try TCLUST, with restr = “eigen” and restr.fact = 5



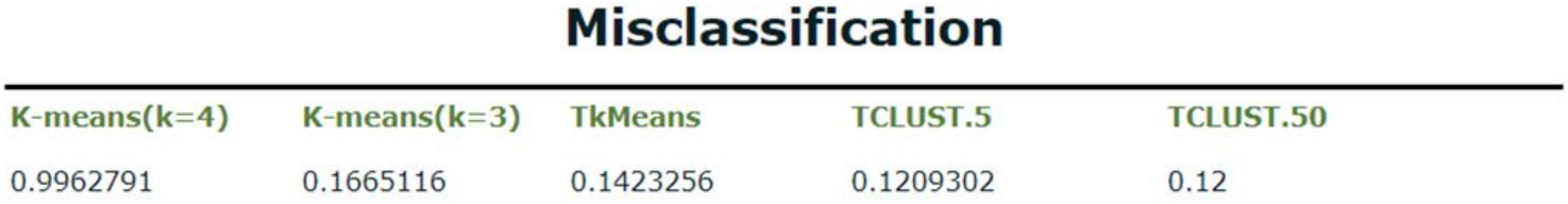
Simulation

Allowing the upper bound to be higher, giving flexibility to the clusters



Simulation

- We now compare the misclassification rates of each algorithm



Misclassification

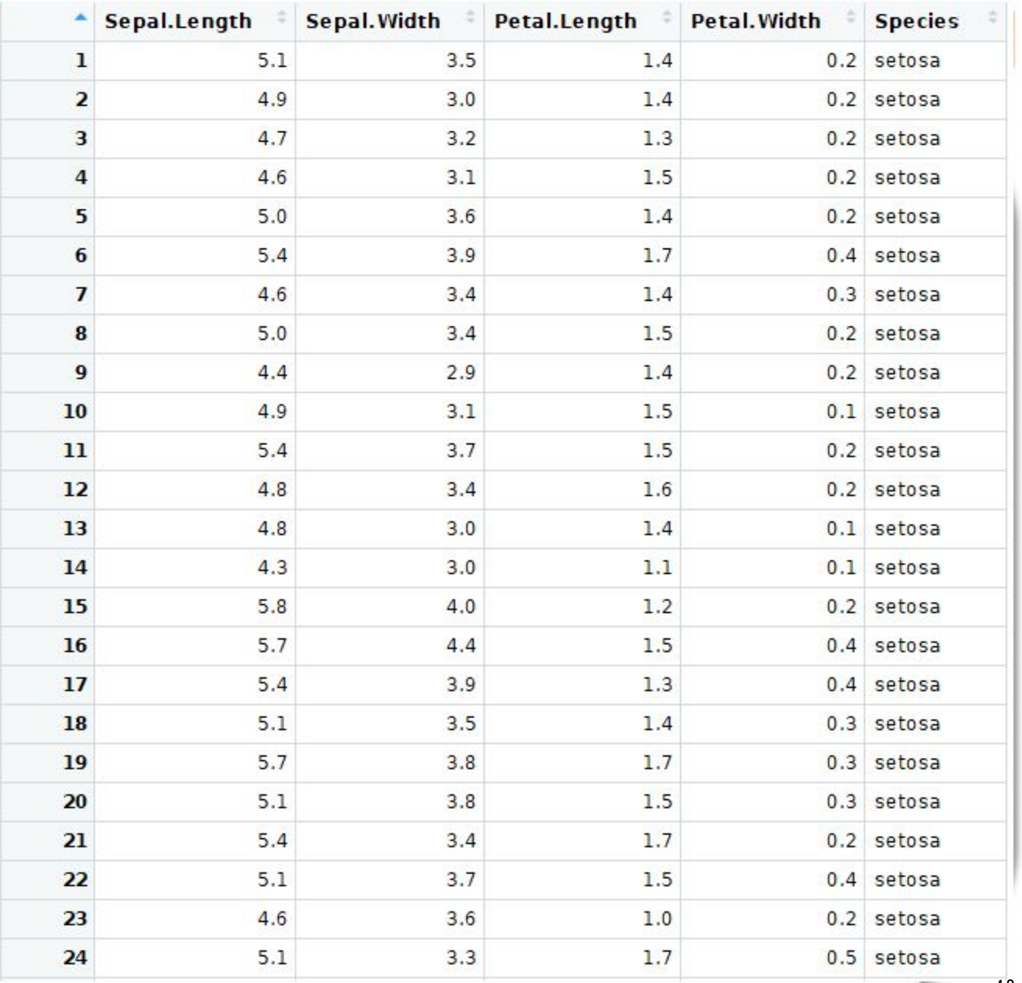
K-means(k=4)	K-means(k=3)	TkMeans	TCLUST.5	TCLUST.50
0.9962791	0.1665116	0.1423256	0.1209302	0.12

- TCLUST is the winner, and K-means is the sore loser!

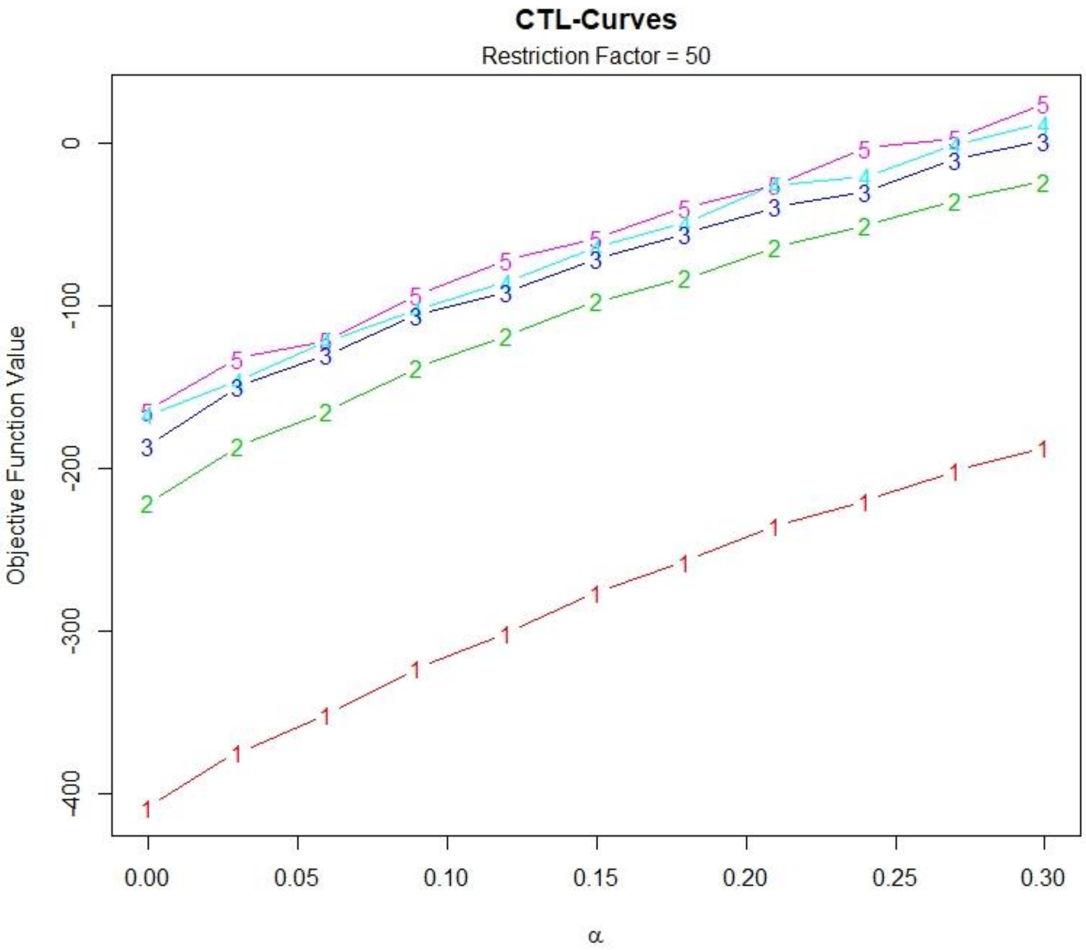
Application: Iris

- We now consider the full Iris dataset

- Here, we have 4 variables: sepal length, sepal width, petal length and petal width with 150 observations

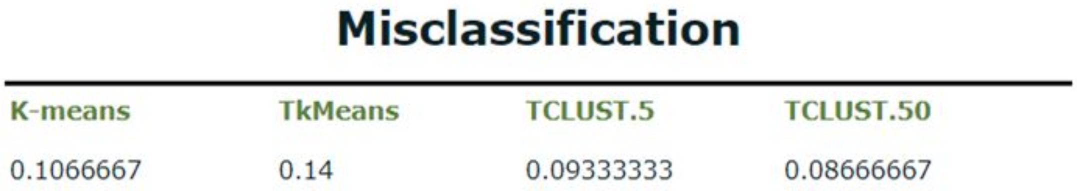


	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa
11	5.4	3.7	1.5	0.2	setosa
12	4.8	3.4	1.6	0.2	setosa
13	4.8	3.0	1.4	0.1	setosa
14	4.3	3.0	1.1	0.1	setosa
15	5.8	4.0	1.2	0.2	setosa
16	5.7	4.4	1.5	0.4	setosa
17	5.4	3.9	1.3	0.4	setosa
18	5.1	3.5	1.4	0.3	setosa
19	5.7	3.8	1.7	0.3	setosa
20	5.1	3.8	1.5	0.3	setosa
21	5.4	3.4	1.7	0.2	setosa
22	5.1	3.7	1.5	0.4	setosa
23	4.6	3.6	1.0	0.2	setosa
24	5.1	3.3	1.7	0.5	setosa


CTL-Curves
Restriction Factor = 50
Objective Function Value
0
-100
-200
-300
-400
0.00
0.05
0.10
0.15
0.20
0.25
0.30
α
1
2
3
4
5

Simulation

- We have that our misclassification rate table is



Misclassification

K-means	TkMeans	TCLUST.5	TCLUST.50
0.1066667	0.14	0.09333333	0.08666667

Trimming is unnecessary for this dataset!

Conclusion and suggestions

Conclusion

- Giving constraint on eigenvalue ratio of scatter matrices enables clustering to be robust
- Also, this allows clusters to be heterogeneous
- The presented algorithm is on the "tclust" package in R

Suggestions

- Mathematical derivation for determinant ratio of scatter matrices are also available by Friedman and Rubin (1967)
- The presented algorithm could also be adapted to develop an EM algorithm for the constrained univariate mixture fitting problem defined by Hathaway (1985) and for the multivariate extension by McLachlan and Peel (2000)

References

References

- Garc´ıa-Escudero, L.A., Gordaliza, A., Matran, C. and Mayo-Iscar, A. (2008). A general trimming approach to robust cluster analysis. The Annals of Statistics, 36, 1324–1345.
- Garc´ıa-Escudero, L.A., Fritz, H. and Mayo-Iscar, A. (2012). tclust: An R Package for a Trimming Approach to Cluster Analysis. Journal of Statistical Software, 47(12), DOI: 10.18637/jss.v047.i12