# Clustering

Jonas Haslbeck

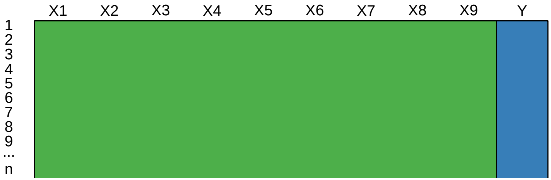
Nov 23th, 2016

# Overview

1. Supervised vs. Unsupervised Learning
2. Definition Clustering
3. Two Clustering methods
   - Partition based clustering (k-means)
   - Hierarchical Clustering
4. What is a good clustering?
   - And how many clusters are there?
5. Discussion

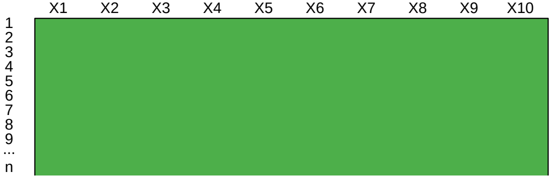
# Supervised Learning

- Goal: Estimate a conditional probability distribution $P(X|Y)$
- $Y$ often univariate, $X$ mostly multivariate
- Often only the location parameter $\mu = \mathbb{E}_P[P(X|Y)]$ of interest
- Trying to find function $f(X)$ that predicts $Y$ as well as possible
- 'well' defined by the loss function

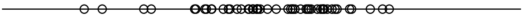


# Unsupervised Learning

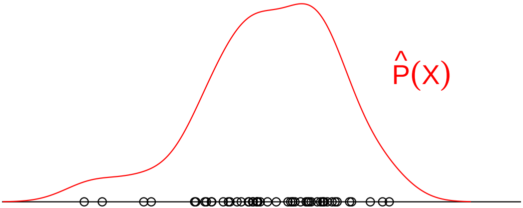
- Goal: Estimate a probability distribution $P(X)$
- $X$ mostly multivariate, possibly huge $p$
- Different characteristics of interest
- Trying to find a density function $\hat{P}(X)$ that is close to $P(X)$



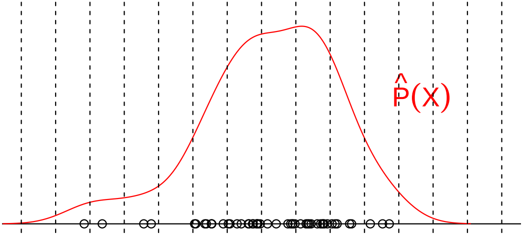
# Density Estimation



# Density Estimation

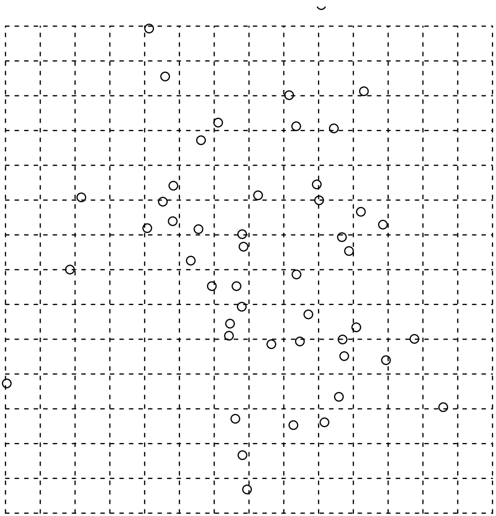


# Density Estimation



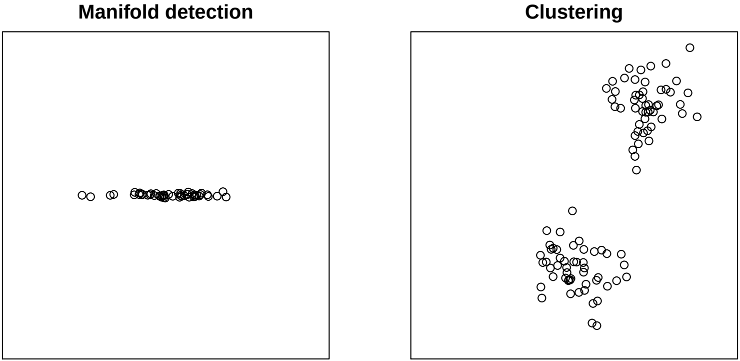
$15^1$ bins

# Density Estimation



$15^p = 15^2 = 225$ bins; curse of dimensionality

# Feasible characterizations of the density

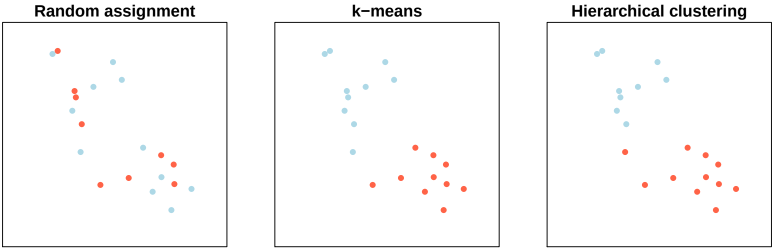


Manifold detection

Clustering

# Definition: Clustering

- A clustering $\psi(X)$ is a mapping from a configuration $X_i \in \mathbb{R}^p$ to a cluster assignment $k \in \{1, \ldots, K\}$
- A *clustering algorithm* $\Psi(X, K)$ learns such a mapping $\psi(X)$ from the data
- Examples: random number generator, k-means, hierarchical clustering



Random assignment

k−means

Hierarchical clustering

# k-means algorithm

Goal:

- Find clustering in $K$ clusters that minimizes within-cluster-variance

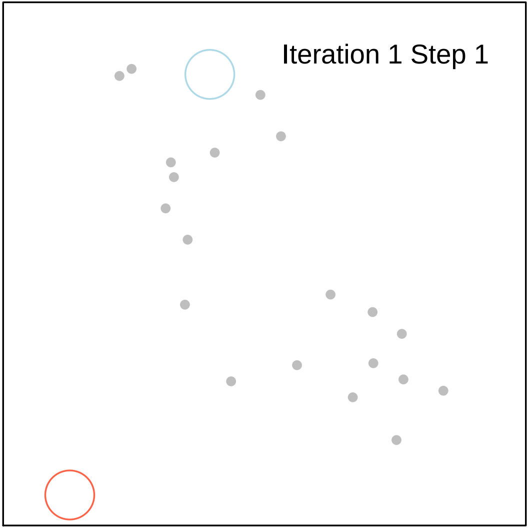
Intuition:

1. Define $k$ points (or centers) in $\mathbb{R}^p$ and assign each object to the closest of the $k$ points
2. Find locations for the $k$ points such that the within-cluster variance is minimal

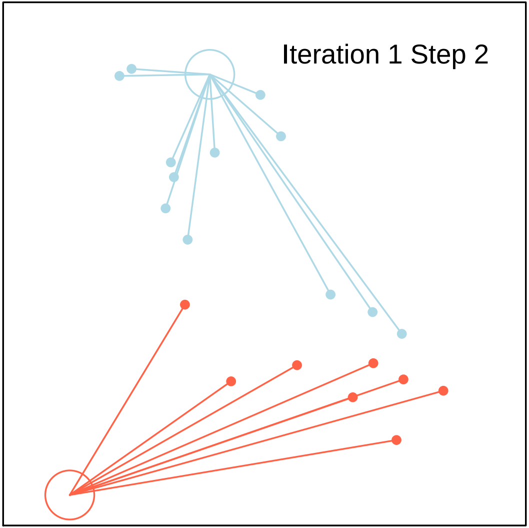
The actual algorithm:

1. start with $K$ randomly chosen centers $m_1, \ldots, m_K$
2. Determine assignment $\psi(X_i) = \arg\min_{k \in \{1,\ldots,K\}} \left\{ ||X_i - m_k||_2^2 \right\}$
3. Recompute centers: $m_k = |S_k|^{-1} \sum_{i=1}^n \mathbb{I}_{\psi(X_i)=k} X_i$, where $S_k$ is the set of objects with assignment $k$
4. repeat steps (2) and (3) until the sets $S_k$ do not change anymore

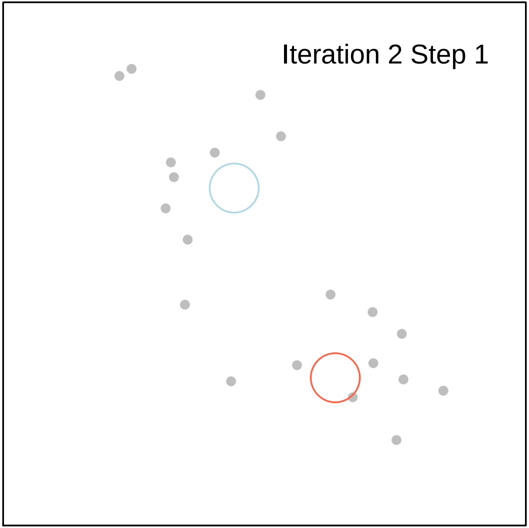
# k-means in action



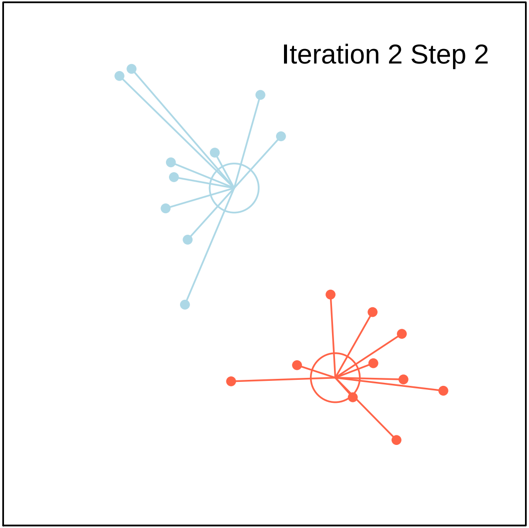
# k-means in action



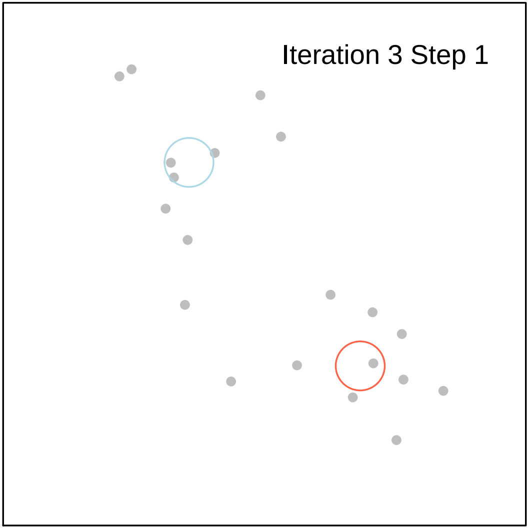
# k-means in action



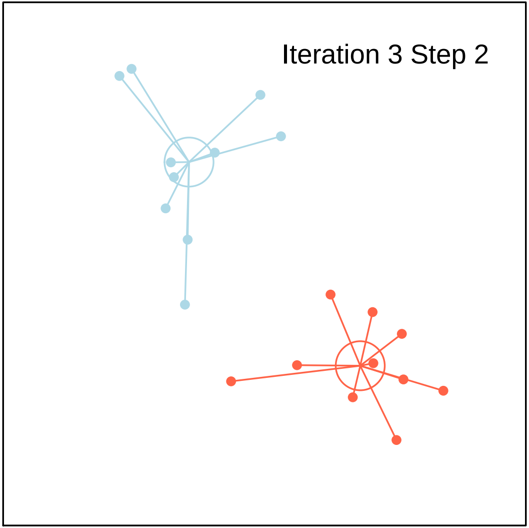
# k-means in action



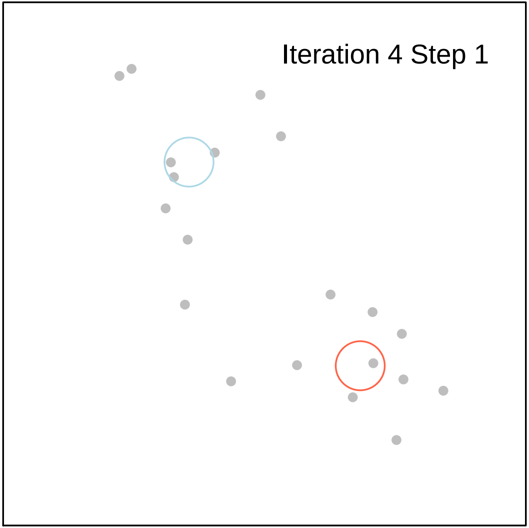
# k-means in action



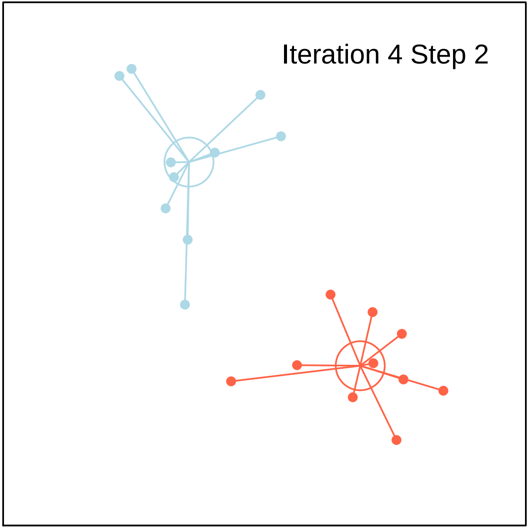
# k-means in action



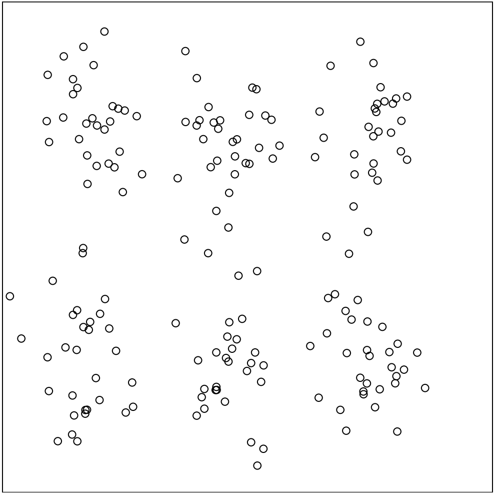
# k-means in action



# k-means in action

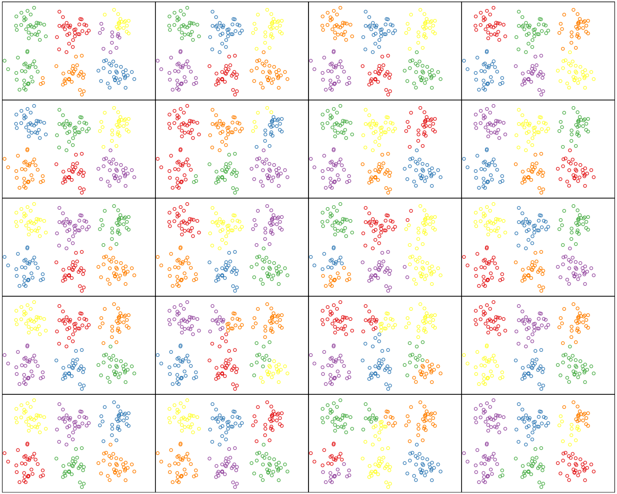


# Problem: local minima



Example: Mixture of 6 Gaussians

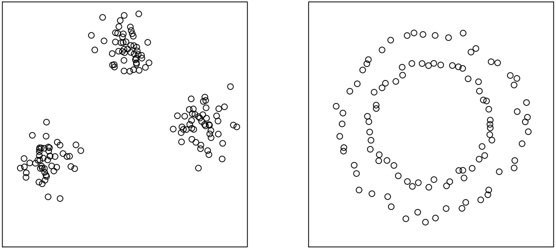
# Problem: local minima



Example: 20 restarts with different initial centers

# Recap k-means: Choices made

1. Euclidean distance metric (SS of deviations from centers is equal to sum of squared pairwise Euclidean distances divided by $n$)
2. We specified $K = 6$ clusters
3. Regions of high density have the shape of centroids:



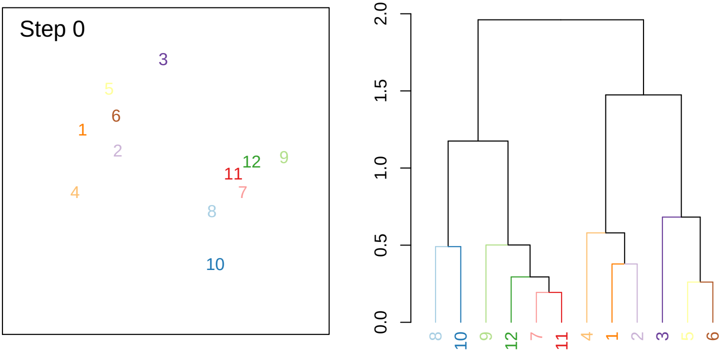
# Hierarchical Clustering

Two types:

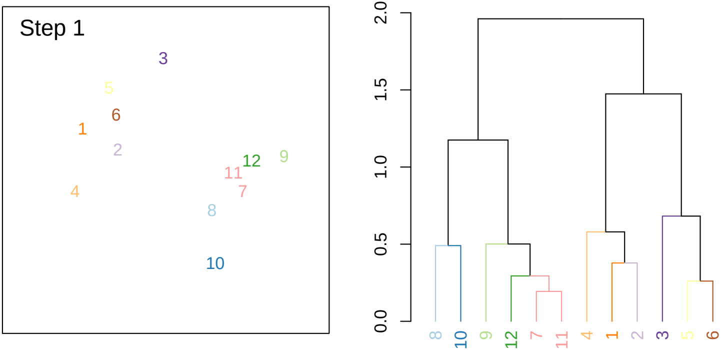
1. **Divisive**: all observations start in one cluster, then make binary splits that maximize the distance between the two new clusters; complexity: $\mathcal{O}(2^n)$
2. **Agglomerative**: start each observation in its own cluster, then combine clusters with smallest distance between them; complexity: $\mathcal{O}(n^2 \log n)$

$n = 20$  
$2^n = 1,048,576$  
$n^2 \log n \approx 1198$  
From $n = 300$ on divisive clustering needs more operations than there are particles in the universe ($10^{80}$).

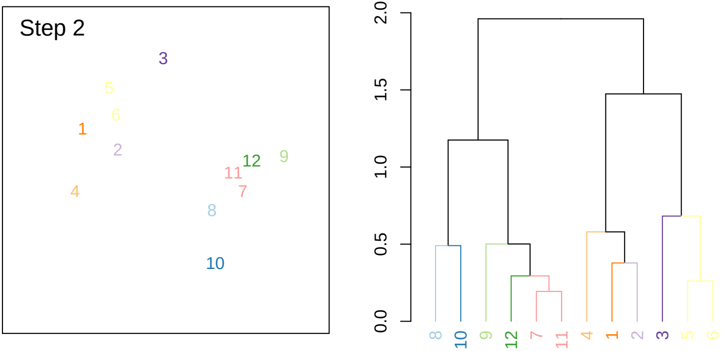
# Hierarchical Clustering in action



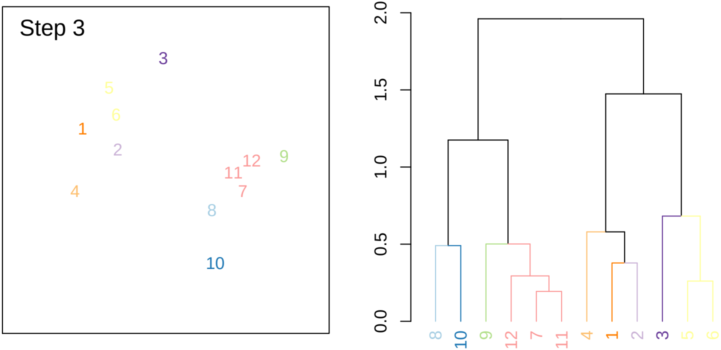
# Hierarchical Clustering in action



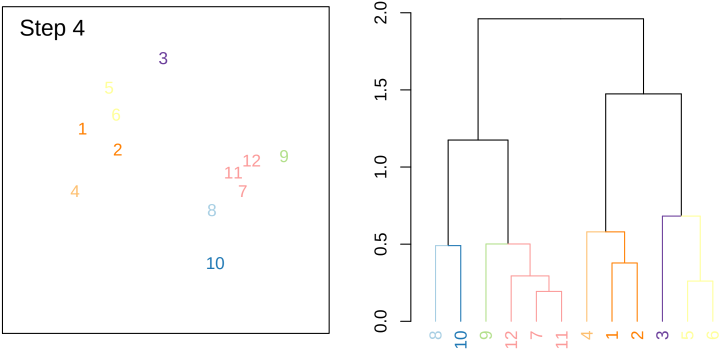
# Hierarchical Clustering in action



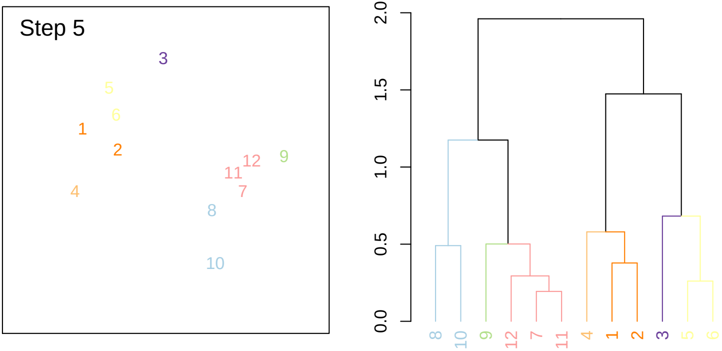
# Hierarchical Clustering in action



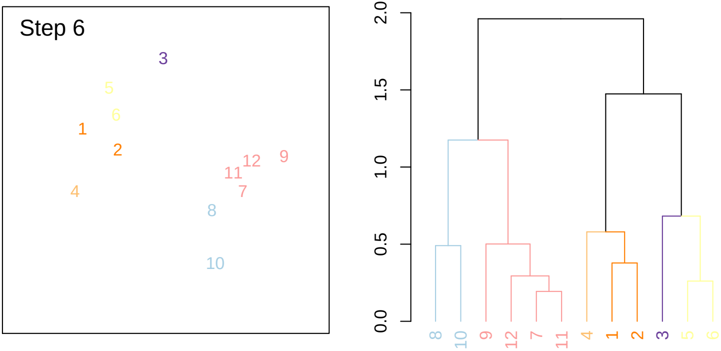
# Hierarchical Clustering in action



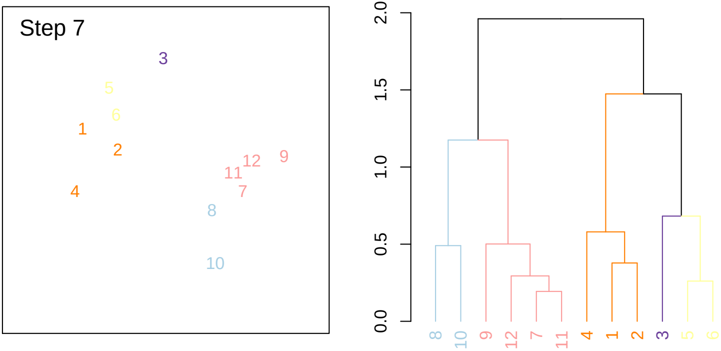
# Hierarchical Clustering in action



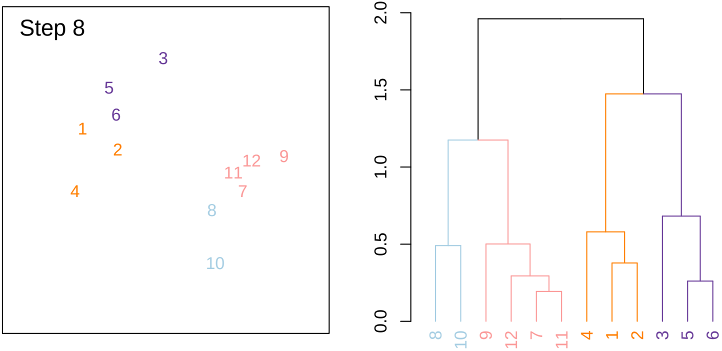
# Hierarchical Clustering in action



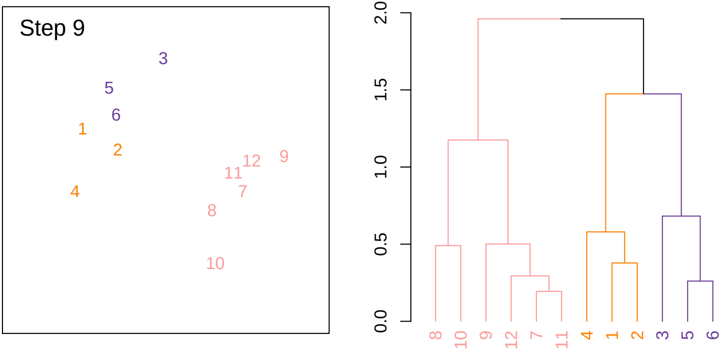
# Hierarchical Clustering in action



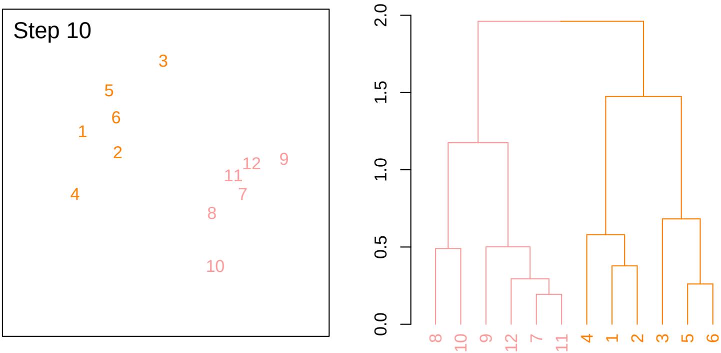
# Hierarchical Clustering in action



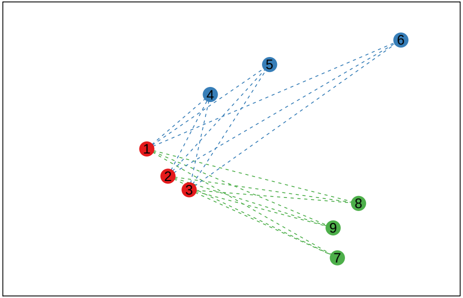
# Hierarchical Clustering in action



# Hierarchical Clustering in action



# How to define distance between clusters?



Two steps:

1. Define distance function for all pairs of objects $d(a, b)$
2. Define 'linkage criterion', to define distance between clusters

# Distance Function

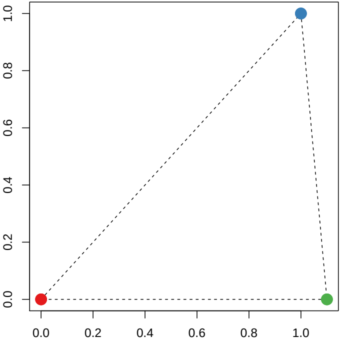
Let $a, b \in \mathbb{R}^p$. Examples for distance function $d(a, b)$:

- Euclidean distance: $\sqrt{\sum_i^p (a_i - b_i)^2}$
- Manhatten distance: $\sum_i^p |a_i - b_i|$
- Maximum distance: $\max_i \{a_i - b_i : i = 1, \ldots, p\}$
- Mahalonobis distance: $\sqrt{(a - b)^\top S^{-1} (a - b)}$, where $S$ is the covariance matrix

Or any other *distance metric*.

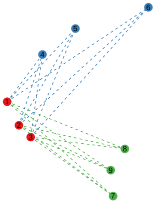
# Distance Function: Euclidean vs. Maximum

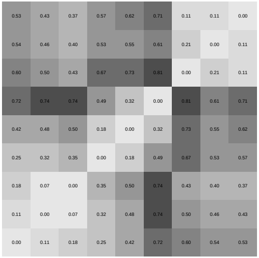
- Euclidean distance: $\sqrt{\sum_i^p (a_i - b_i)^2}$
- Maximum distance: $\max_i\{a_i - b_i : i = 1, \ldots, p\}$



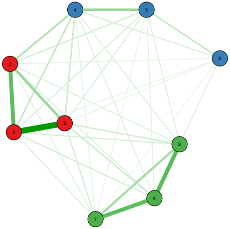
Which point has the larger distance to the red point if we use (a) *Euclidean distance* or (b) *Maximum distance*?

# Distance Function: Visualizations





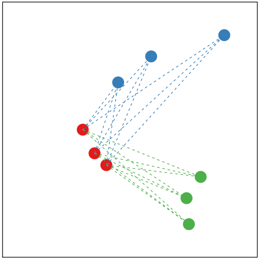
| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0.53 | 0.43 | 0.37 | 0.57 | 0.62 | 0.71 | 0.11 | 0.11 | 0.00 |
| 0.54 | 0.46 | 0.40 | 0.53 | 0.55 | 0.61 | 0.21 | 0.00 | 0.11 |
| 0.60 | 0.50 | 0.43 | 0.67 | 0.73 | 0.81 | 0.00 | 0.21 | 0.11 |
| 0.72 | 0.74 | 0.74 | 0.49 | 0.32 | 0.00 | 0.81 | 0.61 | 0.71 |
| 0.42 | 0.48 | 0.50 | 0.18 | 0.00 | 0.32 | 0.73 | 0.55 | 0.62 |
| 0.25 | 0.32 | 0.35 | 0.00 | 0.18 | 0.49 | 0.67 | 0.53 | 0.57 |
| 0.18 | 0.07 | 0.00 | 0.35 | 0.50 | 0.74 | 0.43 | 0.40 | 0.37 |
| 0.11 | 0.00 | 0.07 | 0.32 | 0.48 | 0.74 | 0.50 | 0.46 | 0.43 |
| 0.00 | 0.11 | 0.18 | 0.25 | 0.42 | 0.72 | 0.60 | 0.54 | 0.53 |



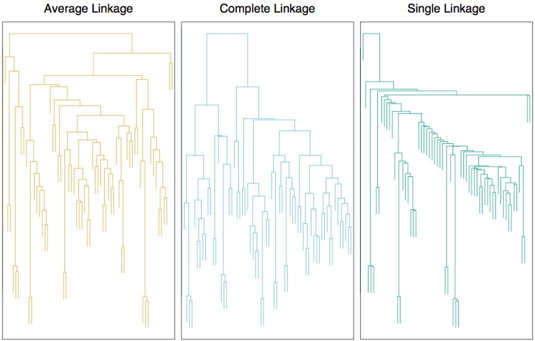
# Linkage function

Let $A, B$, be non-overlapping clusters at some step $j$:

- Maximum:
  $\max\{d(a,b) : a \in A_j, b \in B_j\}$
- Minimum:
  $\min\{d(a,b) : a \in A_j, b \in B_j\}$
- Average: $\frac{1}{|A_j||B_j|} \sum_{a \in A_j} \sum_{b \in B_j} d(a,b)$
- Many others

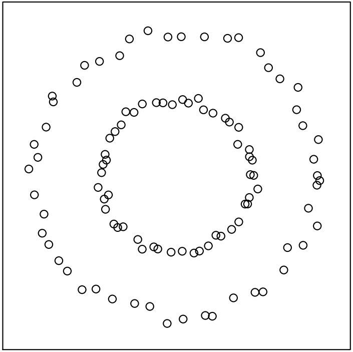


# Linkage function: Different behavior

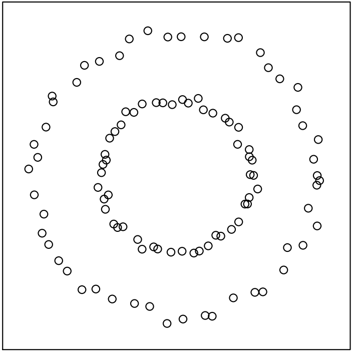


- Maximum (Complete): Produces compact clusters with small diameters
- Minimum (Single): Combines clusters linked by close objects, results in 'chaining'
- Average: Compromise between both

# Example: Average vs. Single



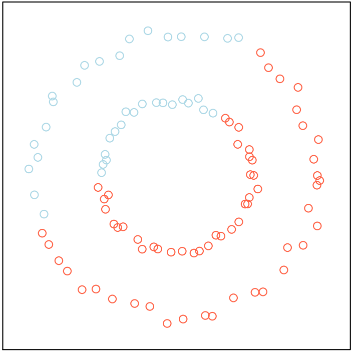
Average linkage



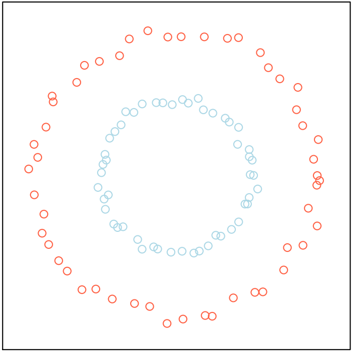
Single linkage

# Example: Average vs. Single

**Average linkage**



**Single linkage**

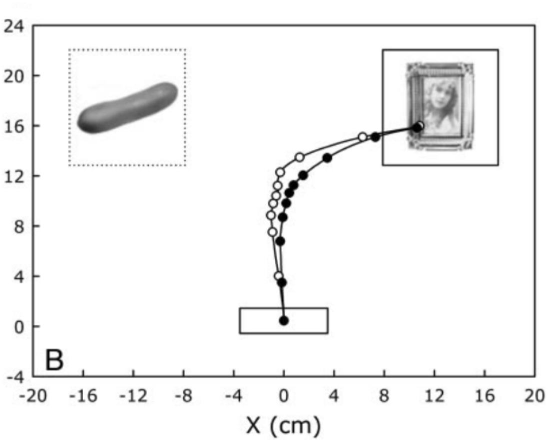


# Recap Hierarchical Clustering: Choices made

1. Distance measure
2. Linkage function
3. Number of clusters $k$
4. Shape of high density regions?

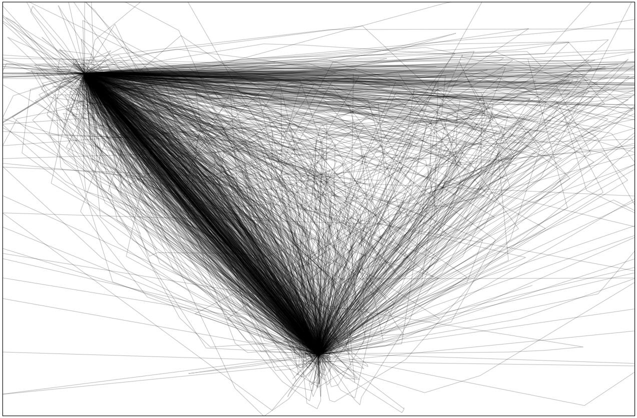
# Optimality: Biased by Expectation?

Example: Mouse-tracking experiments



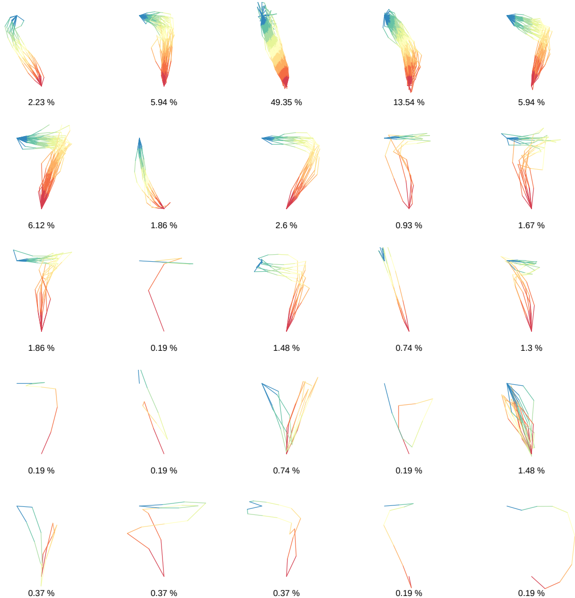
(Spivey et al., 2005)

# Optimality: Biased by Expectation?



Raw data; 1038 trials of 31 participants
(Dale et al. 2006)

# Optimality: Biased by Expectation?



2.23 %

5.94 %

49.35 %

13.54 %

5.94 %

6.12 %

1.86 %

2.6 %

0.93 %

1.67 %

1.86 %

0.19 %

1.48 %

0.74 %

1.3 %

0.19 %

0.19 %

0.74 %

0.19 %

1.48 %

0.37 %

0.37 %

0.37 %

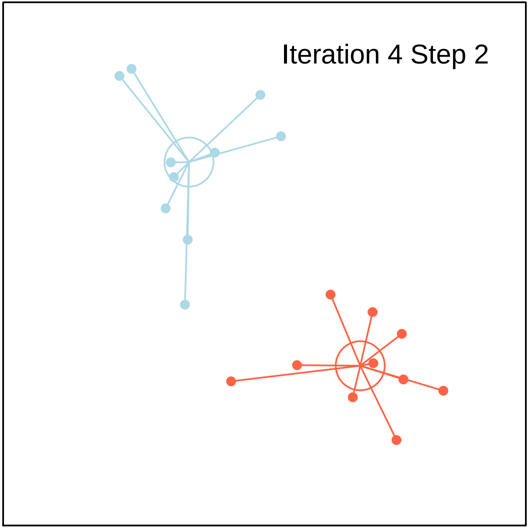
0.19 %

0.19 %

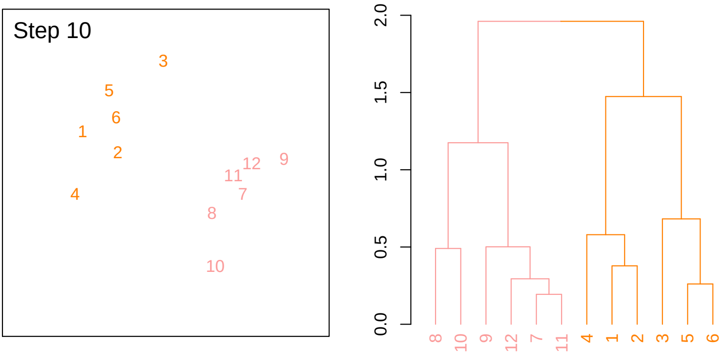
# Clustering: Part 2

December 7th

# Recap: k-means



# Recap: Hierarchical Clustering



# Recap: Design Choices

Definition of a 'good clustering' depends on many choices:

1. Data preprocessing
2. Algorithm
3. Distance measure
4. Linkage function
5. ...

# One more problem: How many clusters?

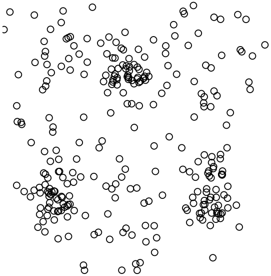
Some 'state-of-the-art' approaches:

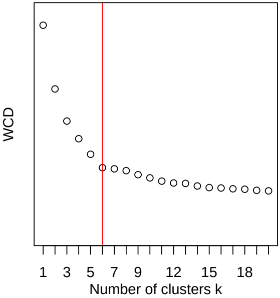
1. Gap-statistic (WCD based)
2. Jump-statistic (WCD based)
3. Silhouette Statistic (Separation based)
4. Stability based methods

Differ in how they define a 'good clustering'!

# k-selection using WCD

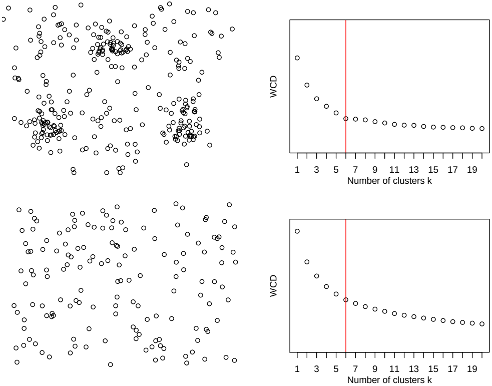
- Problem: WCD is a decreasing function of $k$:





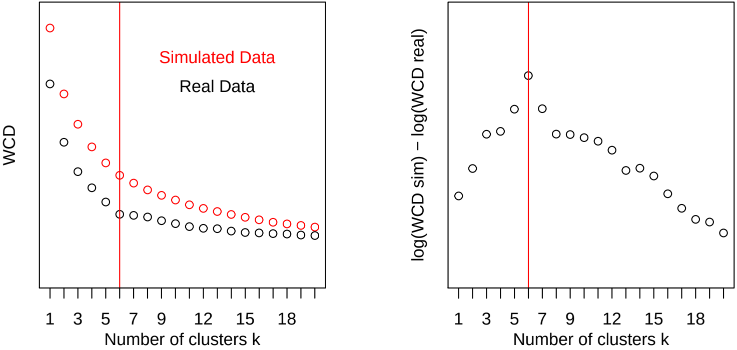
# k-selection: Gap statistic

1. Compute WCD-function of data
2. Generate uniform data of same dimensionality
3. Compute WCD-function of simulated data
4. Select k for which the difference is largest



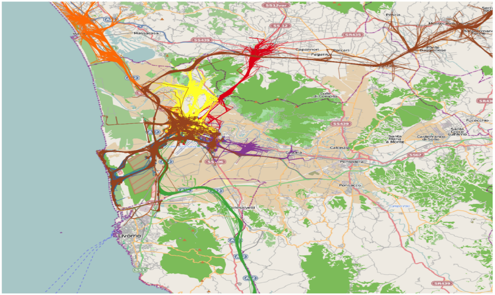
(Tibshirani et al., 2001)

# k-selection: Gap statistic



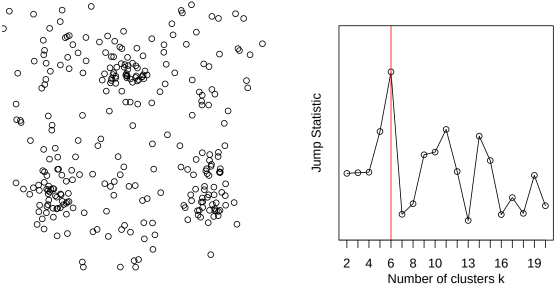
# k-selection: Gap statistic - Problems

- Only works well if clustering is easy / when it is also relatively easy to visually spot the 'kink'
- Large computational cost to fill high dimensional space
- Not always straightforward to generate uniform data:



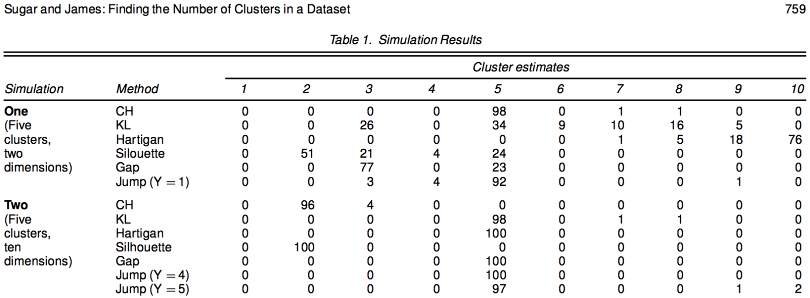
# k-selection: Jump statistic

- Calculate WCB for each time step $WCD_j, j = 1, \ldots, K$
- Choose a power $Y$, suggested $p/2$
- Calculate Jump Statistic: $J_k = WCD_j^{-Y} - WCD_{j-1}^{-Y}$
- Choose largest Jump statistic: $\arg\max_k \{J_k : j = 2, \ldots, K\}$



(Sugar & James, 2003)

# Jump statistic - Performance



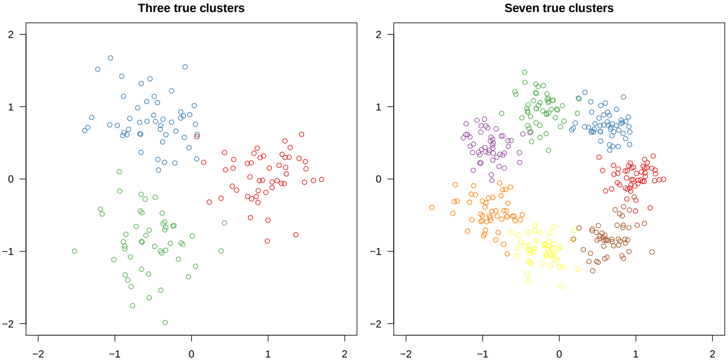
Sugar and James: Finding the Number of Clusters in a Dataset 759

*Table 1. Simulation Results*

| | | *Cluster estimates* | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| *Simulation* | *Method* | *1* | *2* | *3* | *4* | *5* | *6* | *7* | *8* | *9* | *10* |
| **One** | CH | 0 | 0 | 0 | 0 | 98 | 0 | 1 | 1 | 0 | 0 |
| (Five | KL | 0 | 0 | 26 | 0 | 34 | 9 | 10 | 16 | 5 | 0 |
| clusters, | Hartigan | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 5 | 18 | 76 |
| two | Silouette | 0 | 51 | 21 | 4 | 24 | 0 | 0 | 0 | 0 | 0 |
| dimensions) | Gap | 0 | 0 | 77 | 0 | 23 | 0 | 0 | 0 | 0 | 0 |
| | Jump ($Y = 1$) | 0 | 0 | 3 | 4 | 92 | 0 | 0 | 0 | 1 | 0 |
| **Two** | CH | 0 | 96 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| (Five | KL | 0 | 0 | 0 | 0 | 98 | 0 | 1 | 1 | 0 | 0 |
| clusters, | Hartigan | 0 | 0 | 0 | 0 | 100 | 0 | 0 | 0 | 0 | 0 |
| ten | Silhouette | 0 | 100 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| dimensions) | Gap | 0 | 0 | 0 | 0 | 100 | 0 | 0 | 0 | 0 | 0 |
| | Jump ($Y = 4$) | 0 | 0 | 0 | 0 | 100 | 0 | 0 | 0 | 0 | 0 |
| | Jump ($Y = 5$) | 0 | 0 | 0 | 0 | 97 | 0 | 0 | 0 | 1 | 2 |

(Sugar & James, 2003)

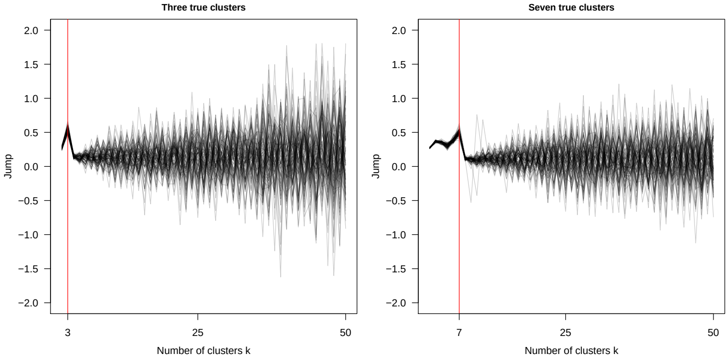
# Jump statistic - Performance?



Three true clusters

Seven true clusters

# Jump statistic - Performance?!



# k-selection: via Clustering Instability

General idea:

- A good clustering is a clustering that is *stable* under small perturbations of the data.

Informal definition Clustering Instability:

- The probability that a pair of observations is not in the same cluster in two perturbed datasets $X_\alpha, X_\beta$.

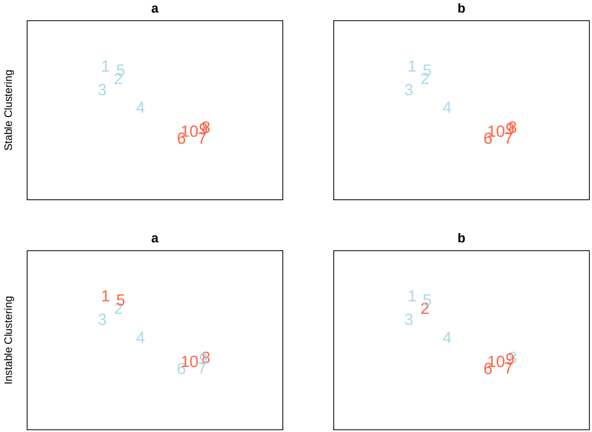
Formal definition:

- $d(\psi_a(X_1), \psi_b(X_2)) =$
  $\mathbb{E}\left[|\mathbb{I}\{\psi_a(X_1) = \psi_a(X_2)\} - \mathbb{I}\{\psi_b(X_1) = \psi_b(X_2)\}|\right]$

Perturbations:

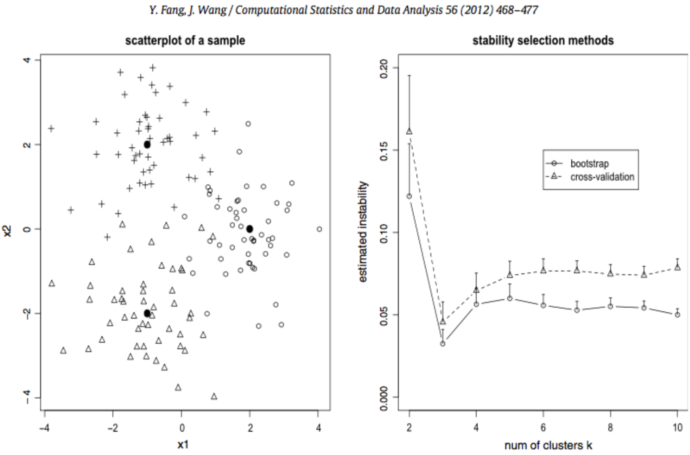
- Cross-validation or (mostly) subsampling

# k-selection: via Clustering Instability



$$\mathbb{E}\left[\left|\mathbb{I}\{\psi_a(X_1)=\psi_a(X_2)\}-\mathbb{I}\{\psi_b(X_1)=\psi_b(X_2)\}\right|\right]$$

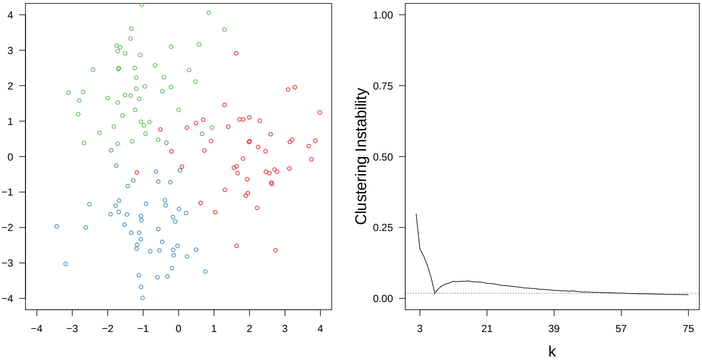
# Clustering Instability - Performance?



Y. Fang, J. Wang / Computational Statistics and Data Analysis 56 (2012) 468–477

(Fang & Wang, 2011)

# Clustering Instability - Performance?!



# Discussion: k-selection

WCD-based methods:

- Basically only work really well if we are able to spot the 'kink' anyway

In General:

- In most situations we sequentially use *two different* definitions of what a good clustering is to obtain a clustering with some $k$
- Example: k-means algorithm + k-selection via clustering instability
- This is weird!

# Discussion

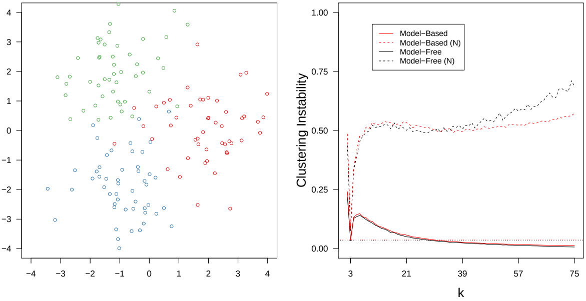
*It is difficult to ascertain the validity of inferences drawn from the output of most unsupervised learning algorithms. One must resort to heuristic arguments not only for motivating the algorithms, as is often the case in supervised learning as well, but also for judgments as to the quality of the results. This uncomfortable situation has led to heavy proliferation of proposed methods, since effectiveness is a matter of opinion and cannot be verified directly.* ESL, p. 487

- How much can we *discover* if we judge the outcome by how close it is to what we expect?

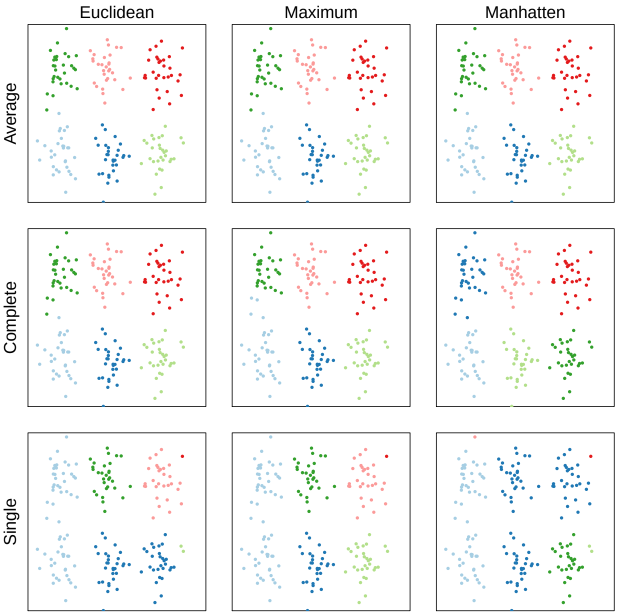
# Outlook

- Connection to Mixture Modeling
- Connection to Manifold detection
- Connection to Quantization
- Spectral Clustering
- Sparse Clustering
- Connection to GGM estimation
- Many many other things

# Fixing the Instability Problem



# Design choices: It makes a difference


Euclidean
Maximum
Manhatten
Average
Complete
Single