

Data Mining

Lecture 3: Discovering Groups

Jo Houghton

ECS Southampton

February 22, 2019

Discovering Groups - Introduction

Understanding large datasets is hard, especially if it has high dimensional features

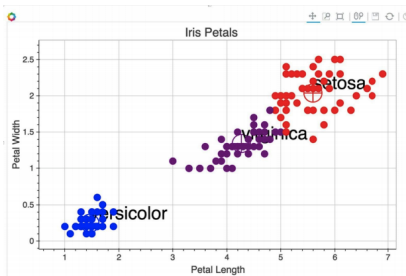
To help understand a dataset:

- ▶ Find similar data items
- ▶ Find similar features

Discovering Groups - Clustering

Grouping data, just using the feature vectors

- ▶ Unsupervised
- ▶ Similar feature vectors grouped together
- ▶ Can be
 - ▶ Soft (allow overlapping groups)
 - ▶ Hard (each item assigned to one group)



Discovering Groups - K Means

Algorithm 1: K Means clustering

Data: X , K

initialise K centroids;

while *positions of centroids change* **do**

for *each data point* **do**

 | assign to nearest centroid;

end

for *each centroid* **do**

 | move to average of assigned data points

end

end

return centroids, assignments;

A special case of Expectation Maximisation

[K Means ipynb demo](#)

[K Means Java Demo](#)

Discovering Groups - Hierarchical Clustering

Hierarchical Clustering:

Creates a binary tree that recursively groups pairs of similar items or clusters

Can be:

- ▶ Agglomerative (bottom up)
- ▶ Divisive (top down)

Discovering Groups - Hierarchical Clustering

Algorithm 2: Hierarchical Agglomerative Clustering

Data: N data points with feature vectors X_i $i = 1 \dots N$

$numClusters = N$;

while $numClusters > 1$ **do**

 cluster1, cluster2 = FindClosestClusters();

 merge(cluster1, cluster2);

end

The distance between the clusters is evaluated using a linkage criterion.

If each merge is recorded, a binary tree structure linking the clusters can be formed.

This gives a **dendrogram**

Discovering Groups - Hierarchical Clustering

Linkage criterion: A measure of dissimilarity between clusters

Centroid Based:

- ▶ Dissimilarity is equal to distance between centroids
- ▶ Needs numeric feature vectors

Distance-Based:

- ▶ Dissimilarity is a function of distance between items in clusters
- ▶ Only needs precomputed measure of similarity between items

We could compute a distance matrix between points

Discovering Groups - Hierarchical Clustering

Centroid based linkage:

- ▶ WPGMC: Weighted Pair Group Method with Centroids
When two clusters are combined into a new cluster, the average of the two centroids is the new centroid
- ▶ UPGMC: Unweighted Pair Group Method with Centroids
When two clusters are combined into a new cluster, the new centroid is recalculated based on the positions of the items

Discovering Groups - Hierarchical Clustering

Distance based linkage:

- ▶ **Minimum**, or **single-linkage clustering** Distance between two closest members

$$\min d(a, b) : a \in A, b \in B$$

Produces long, thin clusters

- ▶ **Maximum**, or **complete-linkage clustering** Distance between two most distant members

$$\max d(a, b) : a \in A, b \in B$$

Finds compact clusters, approximately equal diameter

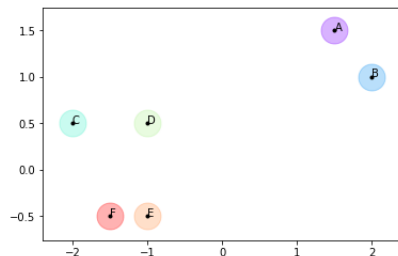
- ▶ **Mean** or **Average Linkage Clustering (UPGMA:**
Unweighted Pairwise Group Method with Arithmetic Mean):

$$\frac{1}{|A||B|} \sum_{a \in A} \sum_{b \in B} d(a, b)$$

Discovering Groups - Hierarchical Clustering

With sample data:

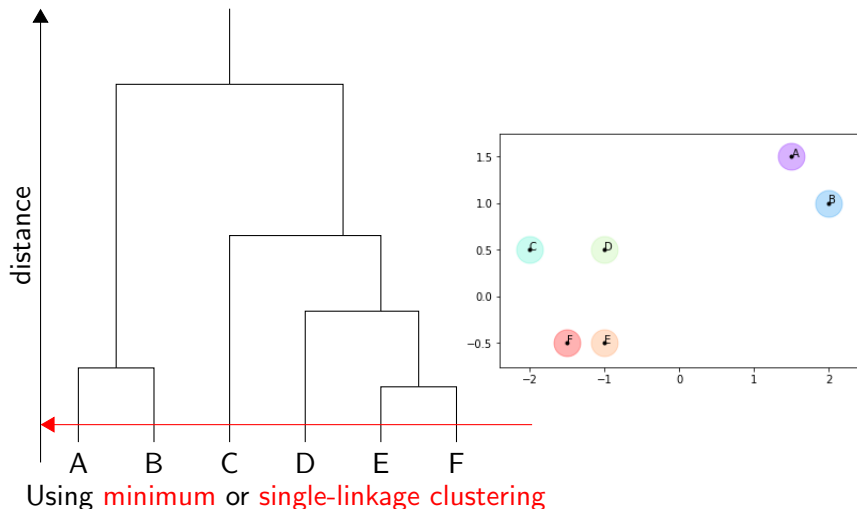
$$X = \begin{bmatrix} 1.5 & 1.5 \\ 2.0 & 1.0 \\ 2.0 & 0.5 \\ -1.0 & 0.5 \\ -1.5 & -0.5 \\ -1 & 0.5 \end{bmatrix}$$



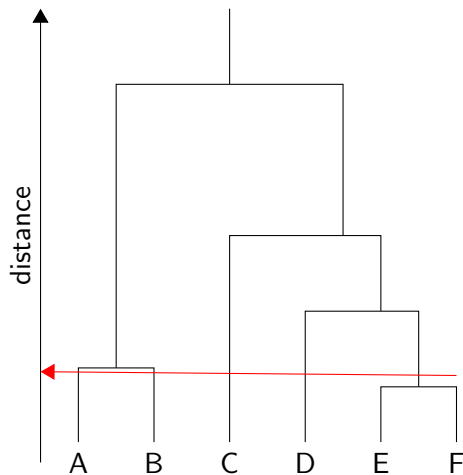
Distance matrix:

| | A | B | C | D | E | F |
|---|-----|-----|-----|-----|-----|---|
| A | 0 | 0.7 | 2.7 | 1.8 | ... | |
| B | 0.7 | 0 | ... | | | |
| C | 2.7 | | 0 | ... | | |
| D | 1.8 | | | 0 | ... | |
| E | ... | | | | 0 | |

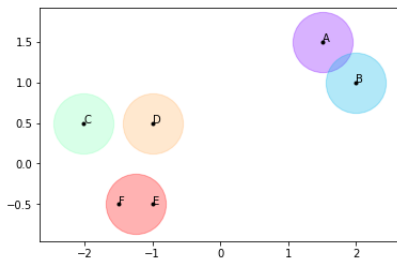
Discovering Groups - Centroid Clustering



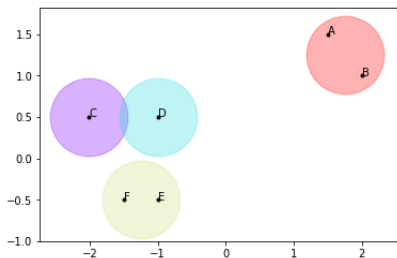
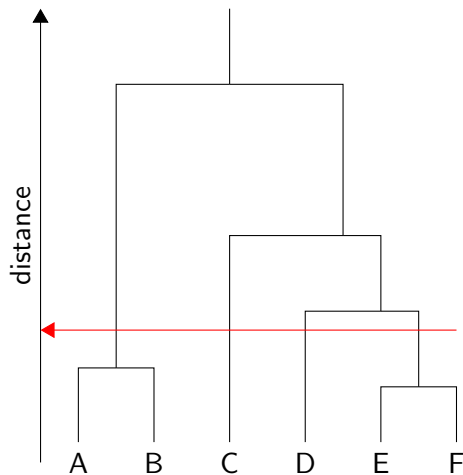
Discovering Groups - Centroid Clustering



Using minimum or single-linkage clustering

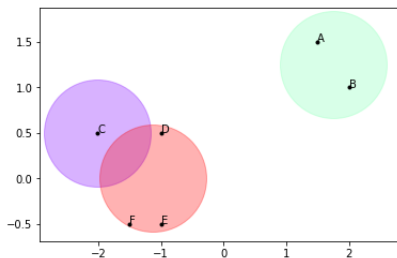
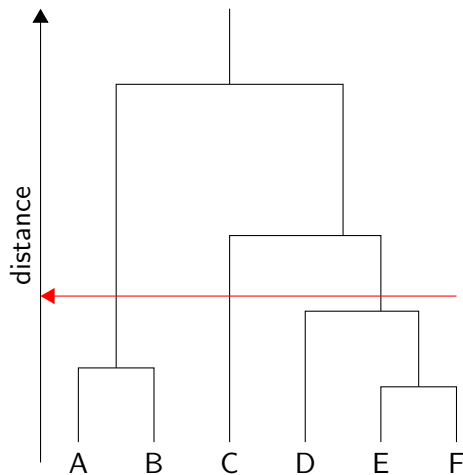


Discovering Groups - Centroid Clustering



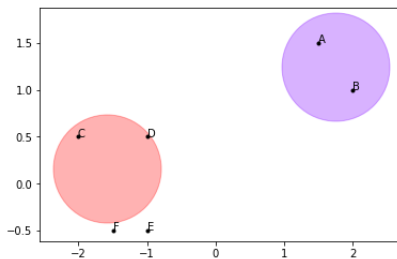
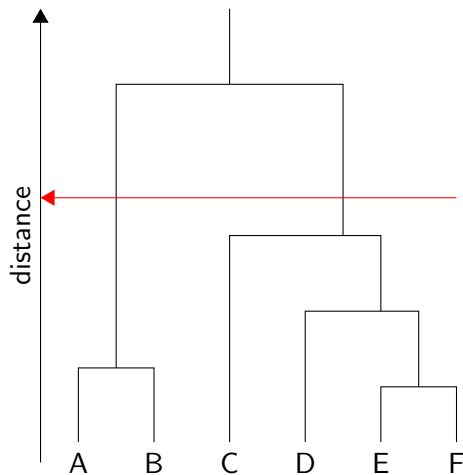
Using minimum or single-linkage clustering

Discovering Groups - Centroid Clustering



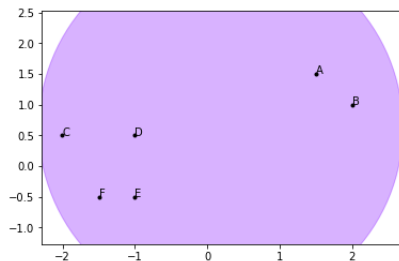
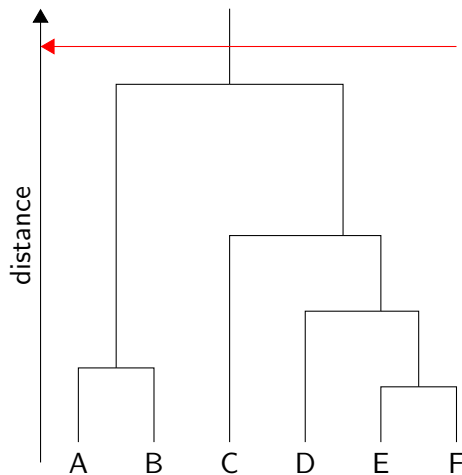
Using **minimum** or **single-linkage** clustering

Discovering Groups - Centroid Clustering



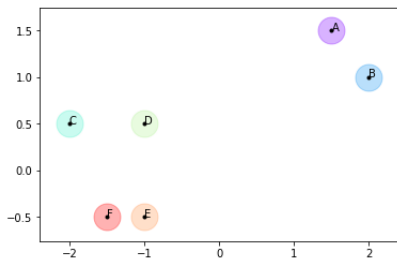
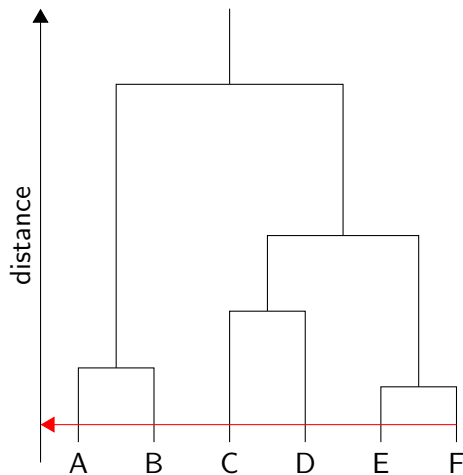
Using minimum or single-linkage clustering

Discovering Groups - Centroid Clustering



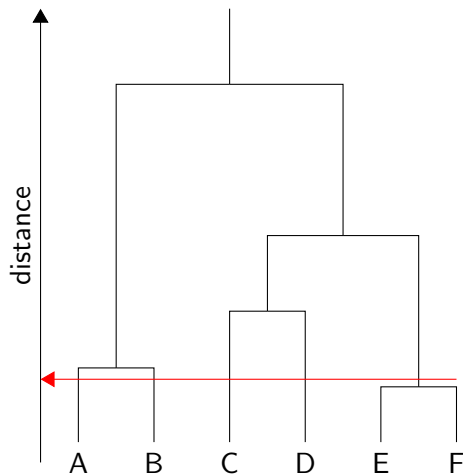
Using **minimum** or **single-linkage** clustering

Discovering Groups - Centroid Clustering

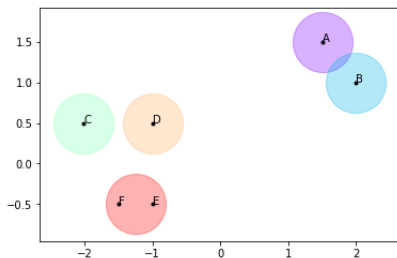


Using **maximum** or **complete-linkage** clustering

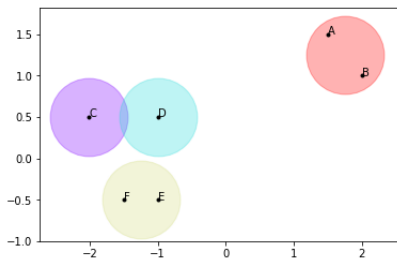
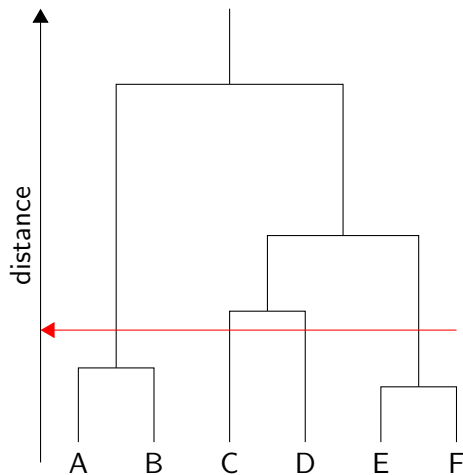
Discovering Groups - Centroid Clustering



Using maximum or complete-linkage clustering

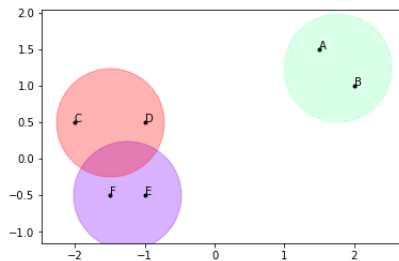
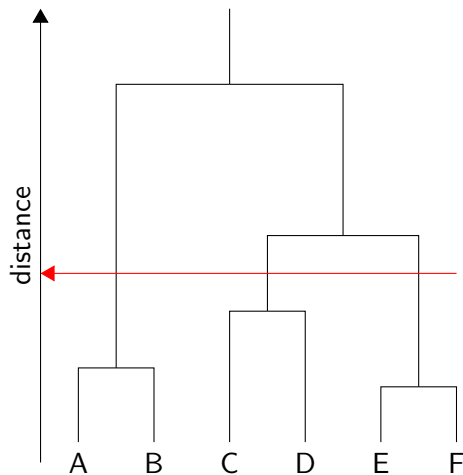


Discovering Groups - Centroid Clustering



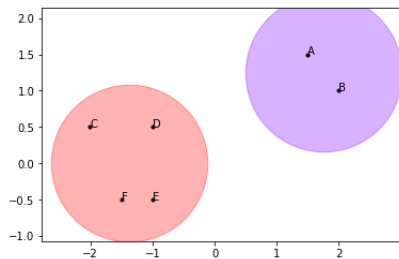
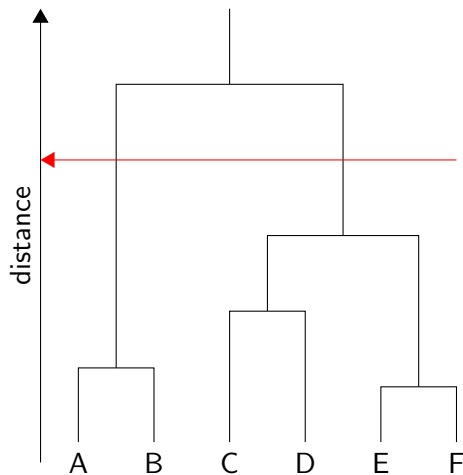
Using **maximum** or **complete-linkage** clustering

Discovering Groups - Centroid Clustering



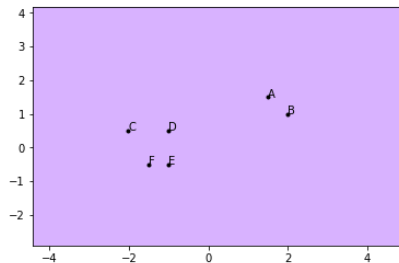
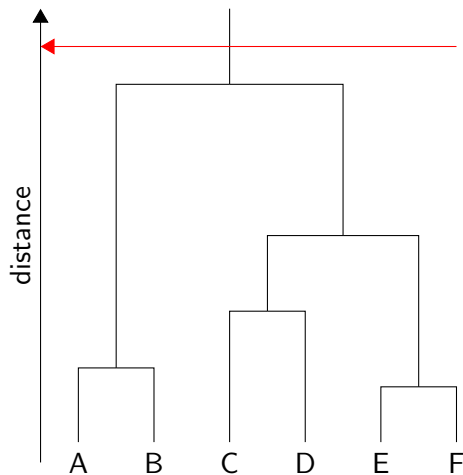
Using **maximum** or **complete-linkage** clustering

Discovering Groups - Centroid Clustering



Using **maximum** or **complete-linkage** clustering

Discovering Groups - Centroid Clustering



Using **maximum** or **complete-linkage** clustering

Discovering Groups - Hierarchical Agglomerative Clustering

Java HAC Demo

Minimum distance linkage tends to give long thin clusters
maximum distance linkage tends to give rounded clusters

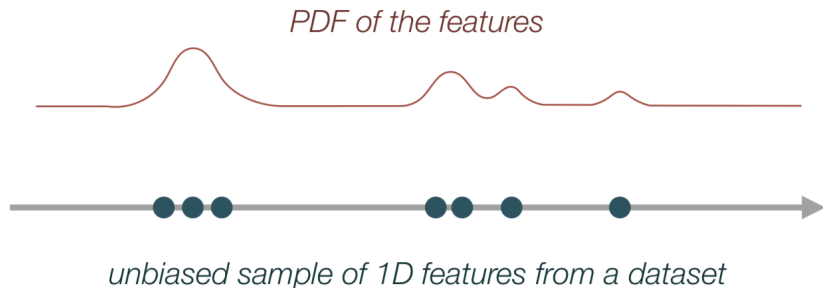
Discovering Groups - Mean Shift Clustering

Mean shift finds the *modes* of a probability density function.

This means it finds the points in feature space with the highest feature density, i.e. are the most likely given the dataset
Needs a kernel and a kernel bandwidth.

It is a hill climbing algorithm that

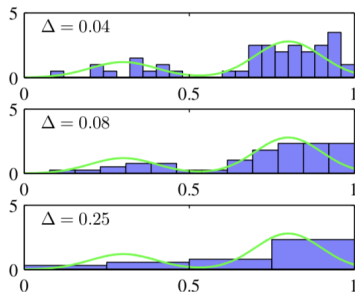
Discovering Groups - Mean Shift Clustering



Discovering Groups - Mean Shift Clustering

How can we estimate the PDF?

Could use a histogram, need to guess number of bins



Changing bin size affecting accuracy of probability density estimation¹

Can be too crude

¹C. Bishop, Pattern Recognition and Machine Learning

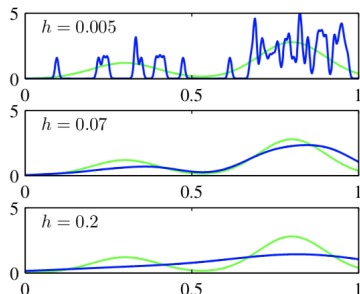
Discovering Groups - Mean Shift Clustering

Kernel Density Estimation (aka Parzen Window)

Gives a smooth continuous estimate

$$f(x) = \frac{1}{nh^d} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right)$$

Where nh is the number of items, d is the dimensionality of the feature space, K is the kernel function, x is an arbitrary position in feature space, h is the kernel bandwidth

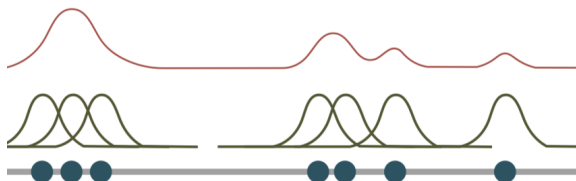


Changing bandwidth affecting accuracy of probability density estimation

Discovering Groups - Mean Shift Clustering

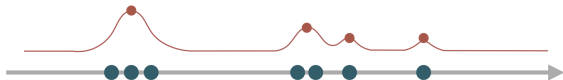
Usually use a Gaussian kernel with $\sigma = 1$

If kernel is radially symmetric, then only need profile of kernel, $k(x)$ that satisfies $K(x) = C_{k,d}k(\|x\|^2)$



Discovering Groups - Mean Shift Clustering

Find the modes of the probability density function (PDF), i.e. where the gradient is zero. $\Delta f(x) = 0$



Discovering Groups - Mean Shift Clustering

$$f(x) = \frac{1}{nh^d} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right)$$

$$K(x) = c_{k,d} k(\|x\|^2)$$

Where $c_{k,d}$ is a normalisation constant

$$f(x) = \frac{c_{k,d}}{nh^d} \sum_{i=1}^n k\left(\left\|\frac{x - x_i}{h}\right\|^2\right)$$

Assuming a radially symmetric kernel:

$$\Delta f(x) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^n (x - x_i) g\left(\left\|\frac{x - x_i}{h}\right\|^2\right) \quad g(x) = -k'(x)$$

$$\Delta f(x) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^n g\left(\left\|\frac{x - x_i}{h}\right\|^2\right) \frac{\sum_{i=1}^n x_i g\left(\left\|\frac{x - x_i}{h}\right\|^2\right)}{\sum_{i=1}^n g\left(\left\|\frac{x - x_i}{h}\right\|^2\right)} - x$$

Discovering Groups - Mean Shift Clustering

$$\Delta f(x) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^n g\left(\left\|\frac{x-x_i}{h}\right\|^2\right) \frac{\sum_{i=1}^n x_i g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)}{\sum_{i=1}^n g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)} - x$$

The first part is a probability density estimate with kernel
 $G(x) = x_{g,d} g(\|x\|^2)$

Discovering Groups - Mean Shift Clustering

$$\Delta f(x) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^n g\left(\left\|\frac{x - x_i}{h}\right\|^2\right) \frac{\sum_{i=1}^n x_i g\left(\left\|\frac{x - x_i}{h}\right\|^2\right)}{\sum_{i=1}^n g\left(\left\|\frac{x - x_i}{h}\right\|^2\right)} - x$$

The first part is a probability density estimate with kernel $G(x) = x_{g,d} g(\|x\|^2)$

The second part is the mean shift, the vector that always points in the direction of maximum density

Discovering Groups - Mean Shift Clustering

Mean shift algorithm:

Algorithm 3: Mean Shift Procedure

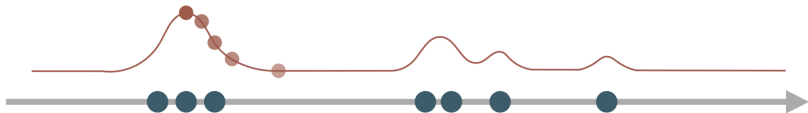
Data: N data points with feature vectors X_i $i = 1 \dots N$

while $x_t \text{ not } = x_{t+1}$ **do**

$m_h(x_t) = \text{computeMeanShiftVect}();$

$x_{t+1} = x_t + m_h(x_t);$

end

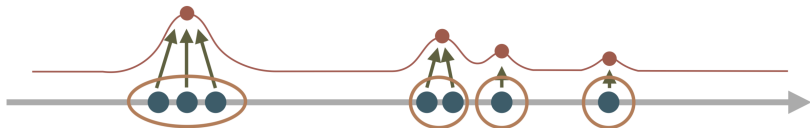


Discovering Groups - Mean Shift Clustering

For each feature vector:

- ▶ apply mean shift procedure until convergence
- ▶ store resultant mode

Set of feature vectors that converge to the same mode define the basin of attraction of that mode



Discovering Groups - Summary

Clustering is a key way to understand your data.

There are many different approaches

- ▶ K Means - Need to chose K
- ▶ Hierarchical Agglomerative Clustering -
- ▶ Mean Shift Clustering

They are a very good way to start exploring a dataset

Coursework 2 is Set!