

INTRODUCTION TO DATA SCIENCE

This lecture is
based on course by E. Fox and C. Guestrin, Univ of Washington

5/12/2017

WFAiS UJ, Informatyka Stosowana
II stopień studiów

Hierarchical clustering

Why hierarchical clustering

3

- Avoid choosing # clusters beforehand

- **Dendrograms** help visualize different clustering **granularities**
 - No need to rerun algorithm



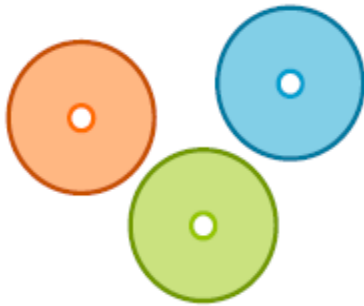
- Most algorithms allow user to **choose any distance metric**
 - k-means restricted us to Euclidean distance

Why hierarchical clustering

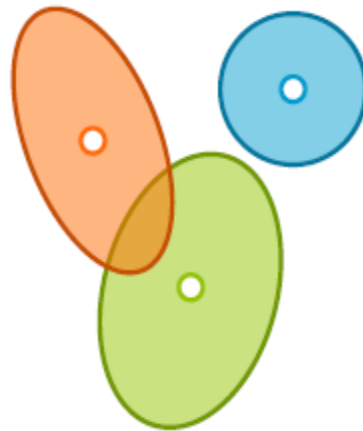
4

Can often find more **complex shapes** than k-means or Gaussian mixture models

k-means: spherical clusters



Gaussian mixtures: ellipsoids

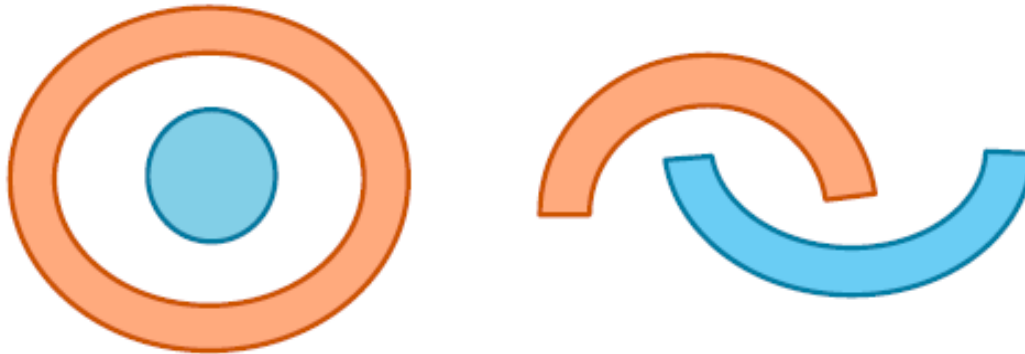


Why hierarchical clustering

5

Can often find more **complex shapes** than k-means or Gaussian mixture models

What about these?



Two main types of algorithms

6

Divisive, *a.k.a top-down*: Start with all data in one big cluster and recursively split.

- Example: **recursive k-means**

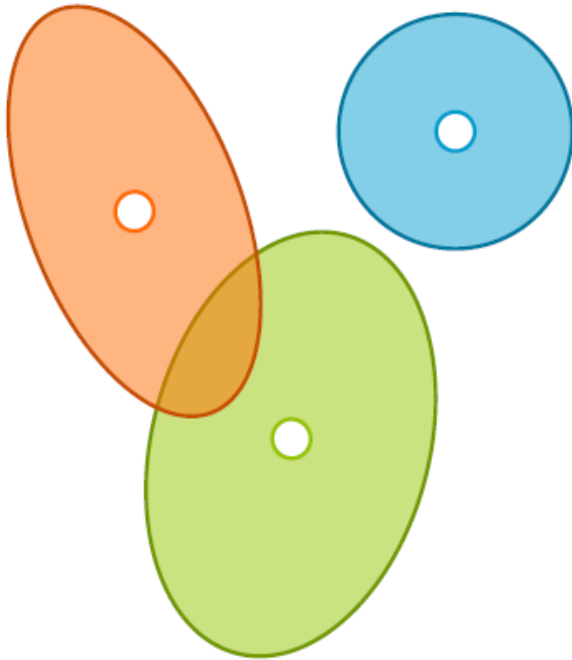
Agglomerative *a.k.a. bottom-up*: Start with each data point as its own cluster. Merge clusters until all points are in one big cluster.

- Example: **single linkage**

Divisive clustering

7

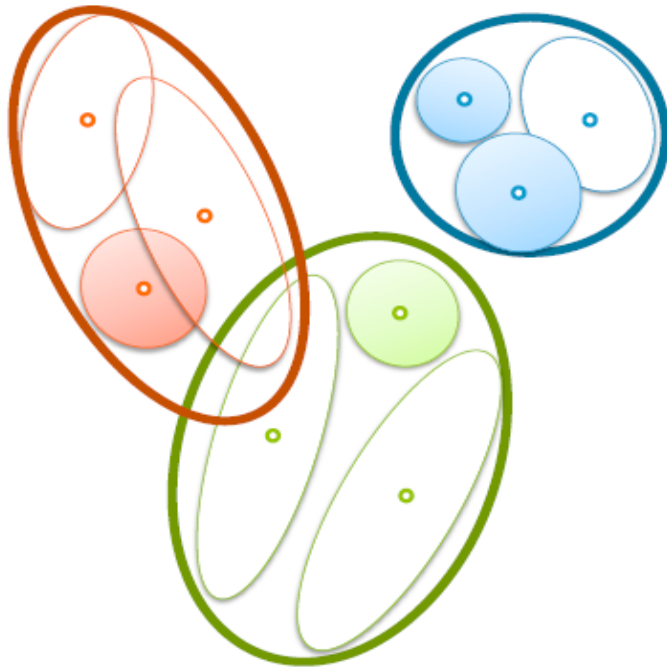
Divisive in pictures – level 1



Divisive clustering

8

Divisive in pictures – level 2



Divisive clustering

9

Divisive: Recursive k-means

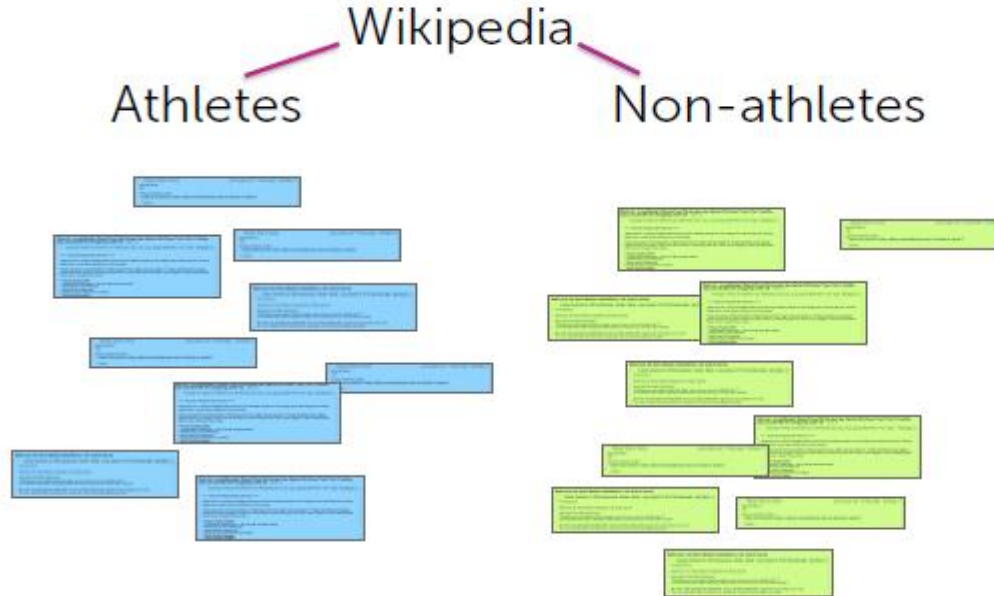
Wikipedia



Divisive clustering

10

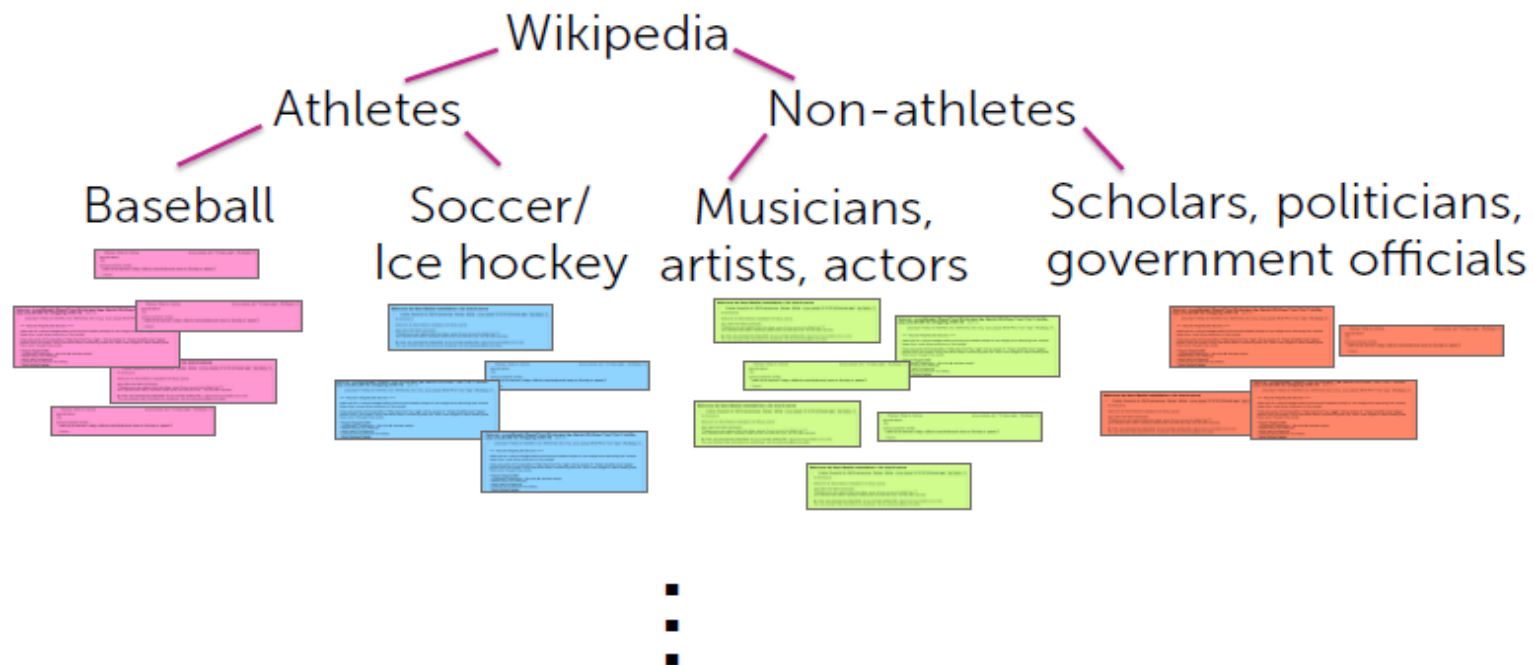
Divisive: Recursive k-means



Divisive clustering

11

Divisive: Recursive k-means



Divisive clustering

12

Divisive choices to be made

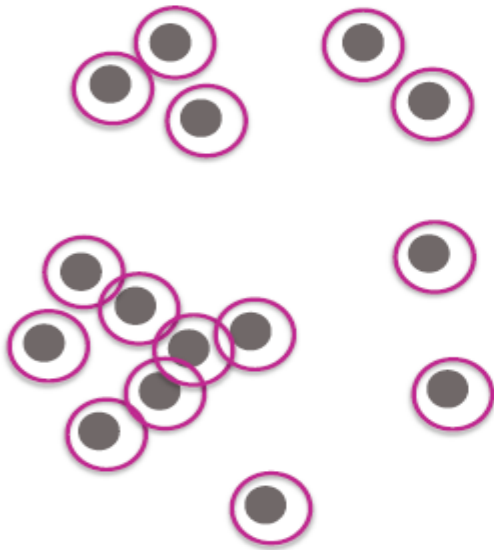
- Which algorithm to recurse
- How many clusters per split
- When to split vs. stop
 - Max cluster size:
number of points in cluster falls below threshold
 - Max cluster radius:
distance to furthest point falls below threshold
 - Specified # clusters:
split until pre-specified # clusters is reached

Agglomerative clustering

13

Agglomerative: Single linkage

1. Initialize each point to be its own cluster

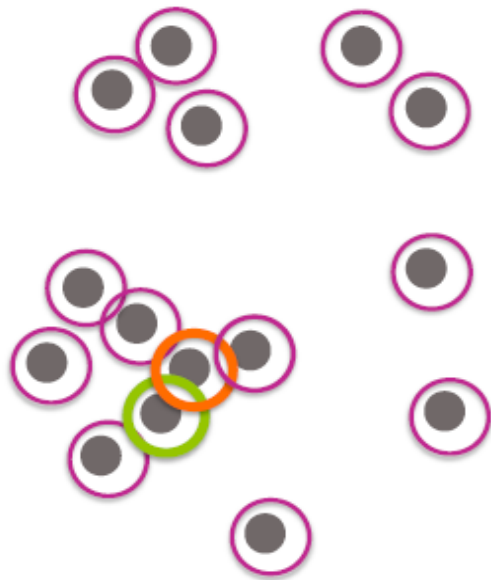


Agglomerative clustering

14

Agglomerative: Single linkage

2. Define distance between clusters to be:



$$\text{distance}(C_1, C_2) =$$

$$\min_{\substack{x_i \in C_1, \\ x_j \in C_2}} d(x_i, x_j)$$

specified pairwise
distance function

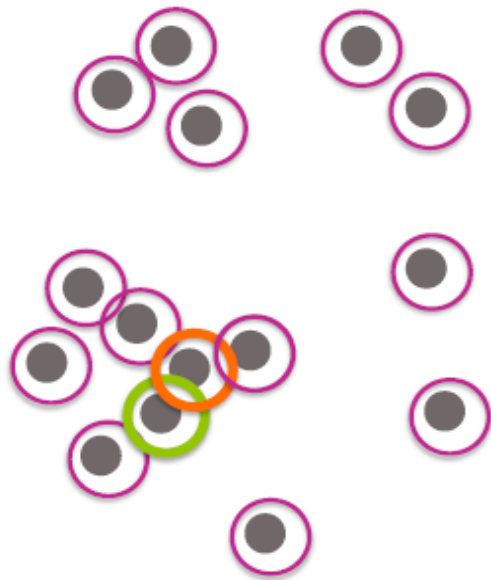
Linkage criteria

Agglomerative clustering

15

Agglomerative: Single linkage

2. Define distance between clusters to be:



$$\text{distance}(C_1, C_2) =$$

$$\min_{\substack{x_i \in C_1, \\ x_j \in C_2}} d(x_i, x_j)$$

specified pairwise
distance function

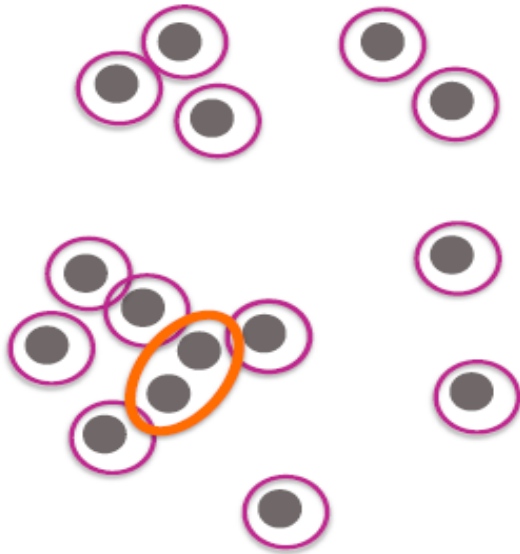
Linkage criteria

Agglomerative clustering

16

Agglomerative: Single linkage

3. Merge the two closest clusters



Agglomerative clustering

17

Agglomerative: Single linkage

4. Repeat step 3 until all points are in one cluster



Agglomerative clustering

18

Agglomerative: Single linkage

4. Repeat step 3 until all points are in one cluster

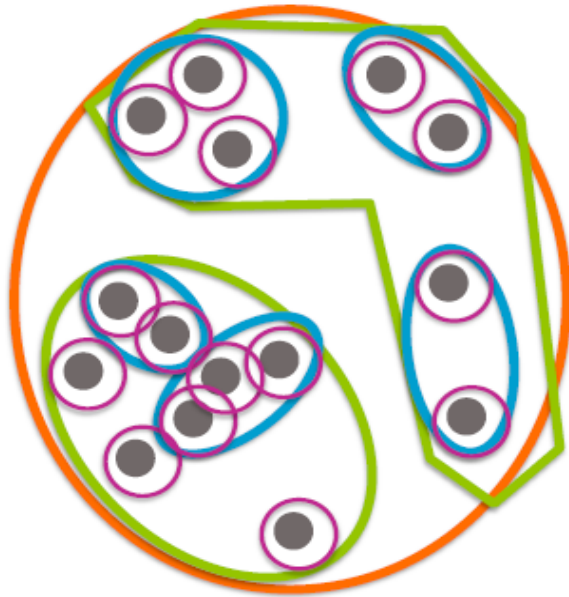


Agglomerative clustering

19

Clusters of clusters

Just like our picture for divisive clustering...

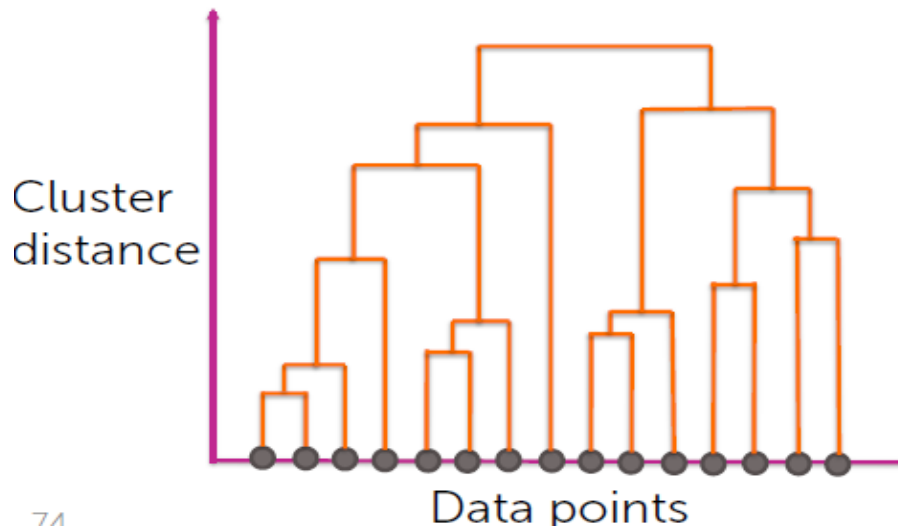


Agglomerative clustering

20

The dendrogram

- x axis shows data points (carefully ordered)
- y-axis shows distance between pair of clusters



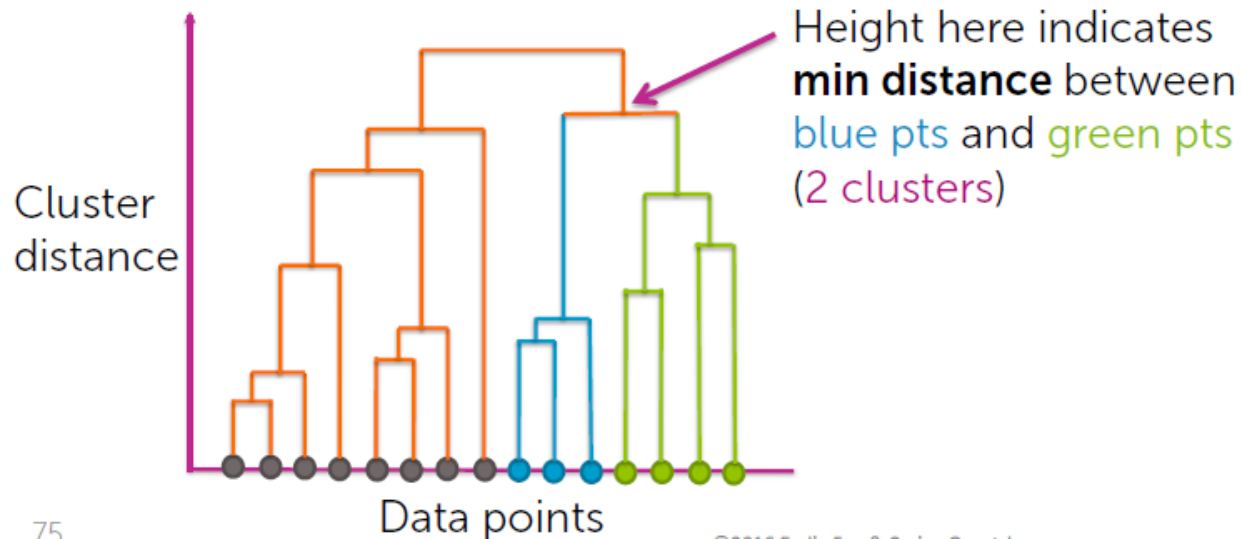
71

Agglomerative clustering

21

The dendrogram

- x axis shows data points (carefully ordered)
- y-axis shows distance between pair of clusters



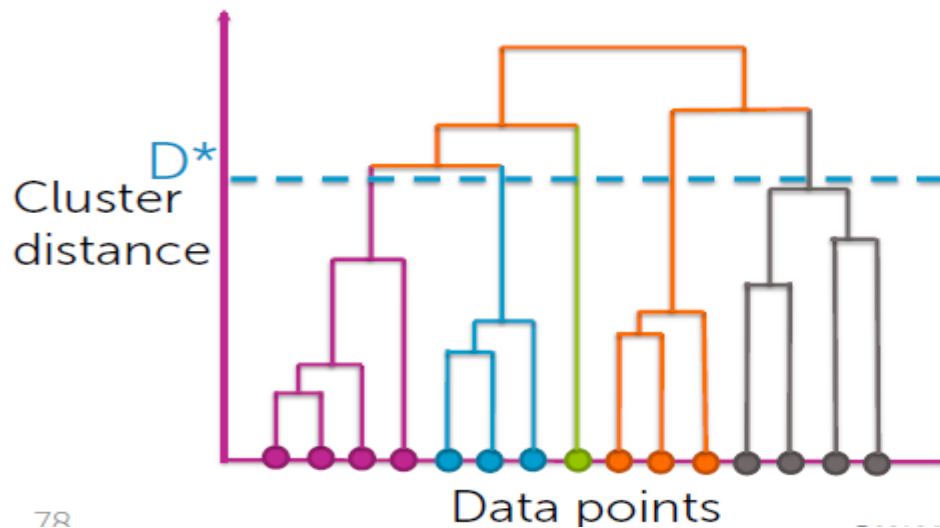
75

Agglomerative clustering

22

Extracting a partition

Every branch that crosses D^*
becomes a separate cluster



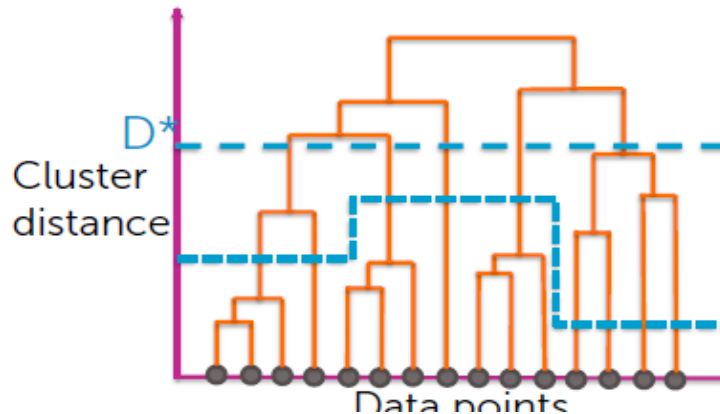
7R

Agglomerative clustering

23

Agglomerative choices to be made

- Distance metric: $d(\mathbf{x}_i, \mathbf{x}_j)$
- Linkage function: e.g., $\min_{\substack{\mathbf{x}_i \in C_1, \\ \mathbf{x}_j \in C_2}} d(\mathbf{x}_i, \mathbf{x}_j)$
- Where and how to cut dendrogram

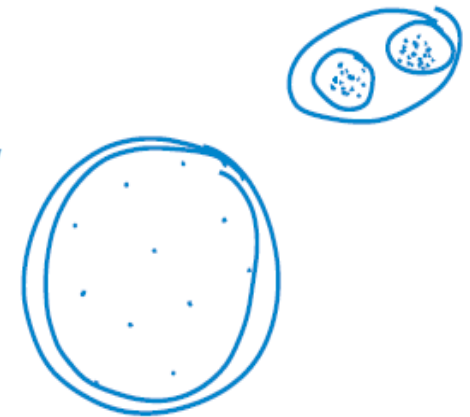


Agglomerative clustering

24

More on cutting dendrogram


- For visualization, smaller # clusters is preferable
- For tasks like outlier detection, cut based on:
 - Distance threshold
 - Inconsistency coefficient
 - Compare height of merge to average merge heights below
 - If top merge is substantially higher, then it is joining two subsets that are relatively far apart compared to the members of each subset internally
 - Still have to **choose a threshold** to cut at, but now in terms of "inconsistency" rather than distance
- No cutting method is "incorrect", some are just more useful than others



Agglomerative clustering

25

Computational considerations

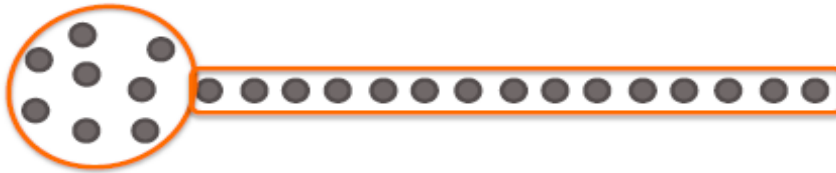
- Computing all pairs of distances is **expensive**
 - Brute force algorithm is $O(N^2 \log(N))$
 -  # datapoints
- Smart implementations use triangle inequality to **rule out candidate pairs**
- Best known algorithm is $O(N^2)$

Agglomerative clustering

26

Statistical issues

Chaining: Distant points clustered together if there is a chain of pairwise close points between



Other **linkage functions** can be more robust, but **restrict the shapes** of clusters that can be found

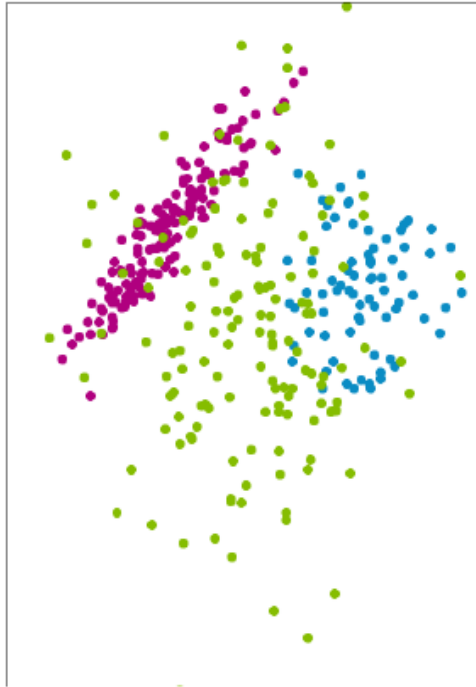
- **Complete linkage:**
max pairwise distance between clusters
- **Ward criterion:**
min within-cluster variance at each merge

Hidden Markov models (HMMs)

Hidden Markov models (HMMs)

28

So far, looked at clustering unordered data

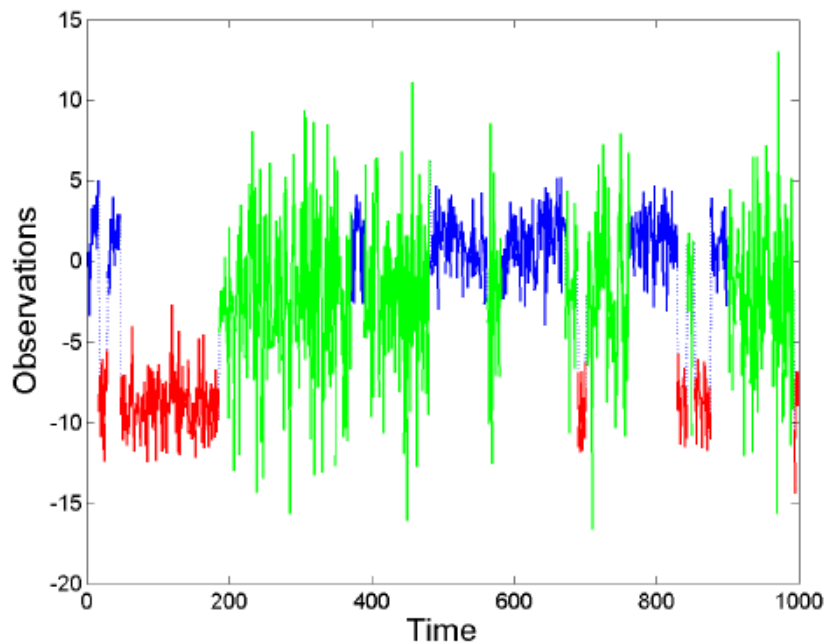


Data index (i.e., when observation was recorded) does not influence clustering

Hidden Markov models (HMMs)

29

What if we have time series data?

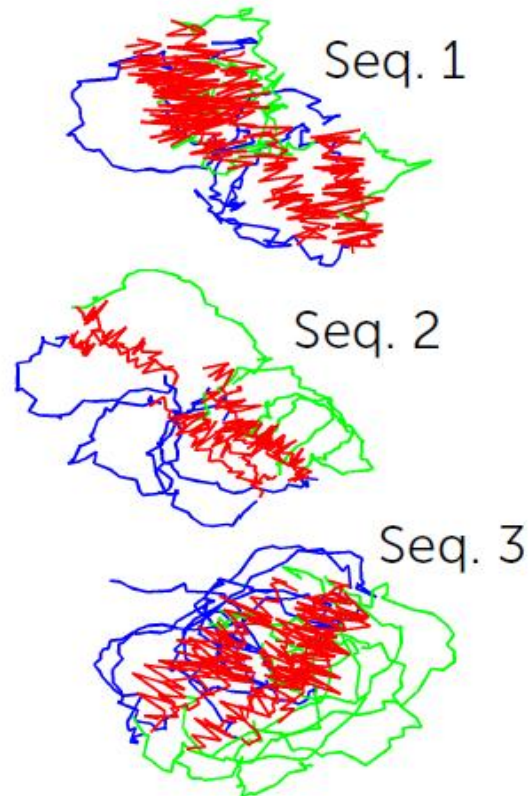


Would be hard to distinguish red, blue, and green clusters if we ignored order of data

Hidden Markov models (HMMs)

30

Example: Honey bee dances

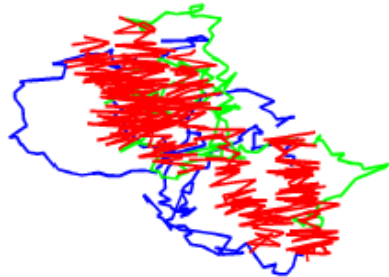


Hidden Markov models (HMMs)

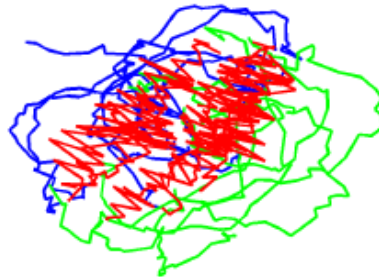
31

Repeated patterns of dance transitions

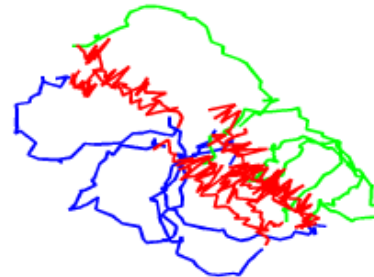
Sequence 1



Sequence 2



Sequence 3



Cluster labels over time

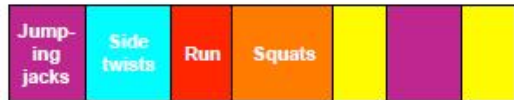


waggle dance
turn right
turn left

Hidden Markov models (HMMs)

32

Similar ideas appear in many applications



00

Hidden Markov models (HMMs)

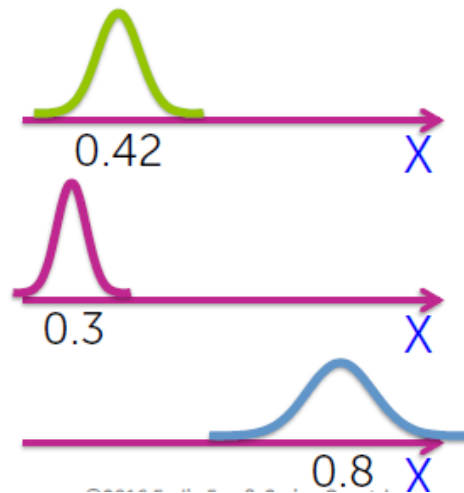
33

Hidden Markov model (HMM)

As in mixture model...

Every observation x_t is associated with cluster assignment variable z_t

Each cluster has a distribution over observed values



q1

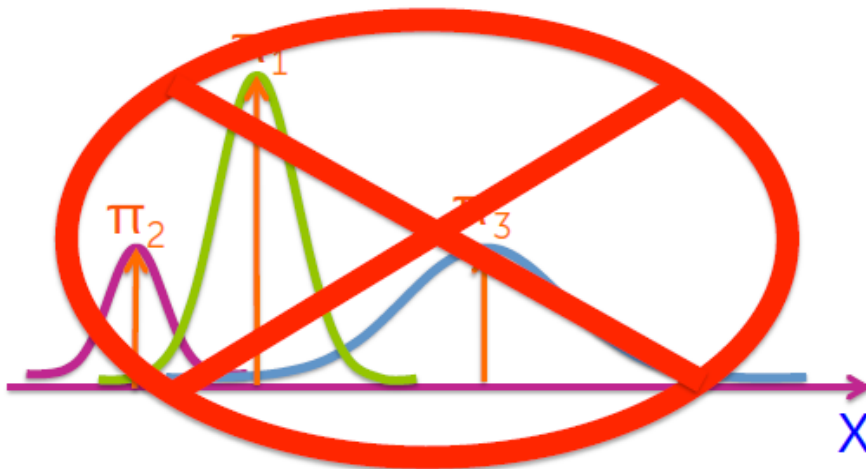
Hidden Markov models (HMMs)

34

Hidden Markov model (HMM)

Difference from mixture model:

Probability of ($z_t = k$) depends on previous cluster assignment z_{t-1}



Hidden Markov models (HMMs)

35

Inference in HMMs

- Learn MLE of HMM parameters using EM algorithm = **Baum Welch**
- Infer MLE of state sequence given fixed model parameters using dynamic programming = **Viterbi algorithm**
- Infer soft assignments of state sequence using dynamic programming = **forward-backward algorithm**

What was not covered

Other clustering+retrieval topics

37

Retrieval:

- Other distance metrics
- Distance metric learning

Clustering:

- Nonparametric clustering
- Spectral clustering

Related ideas:

- Density estimation
- Anomaly detection