CSE 590
Data Science Fundamentals

Cluster Analysis I

Klaus Mueller

Computer Science Department
Stony Brook University and SUNY Korea

Lecture	Topic	Projects
1	Intro, schedule, and logistics	
2	Data Science components and tasks	
3	Data types	Project #1 out
4	Introduction to R, statistics foundations	
5	Introduction to D3, visual analytics	
6	Data preparation and reduction	
7	Data preparation and reduction	Project #1 due
8	Similarity and distances	Project #2 out
9	Similarity and distances	
10	Cluster analysis	
11	Cluster analysis	
12	Pattern miming	Project #2 due
13	Pattern mining	
14	Outlier analysis	
15	Outlier analysis	Final Project proposal due
16	Classifiers	
17	Midterm	
18	Classifiers	
19	Optimization and model fitting	
20	Optimization and model fitting	
21	Causal modeling	
22	Streaming data	Final Project preliminary report due
23	Text data	
24	Time series data	
25	Graph data	
26	Scalability and data engineering	
27	Data journalism	
	Final project presentation	Final Project slides and final report due

Purpose

Data summarization

- data reduction
- cluster centers, shapes, and statistics

Customer segmentation

- collaborative filtering

Social network analysis

- find similar groups of friends (communities)

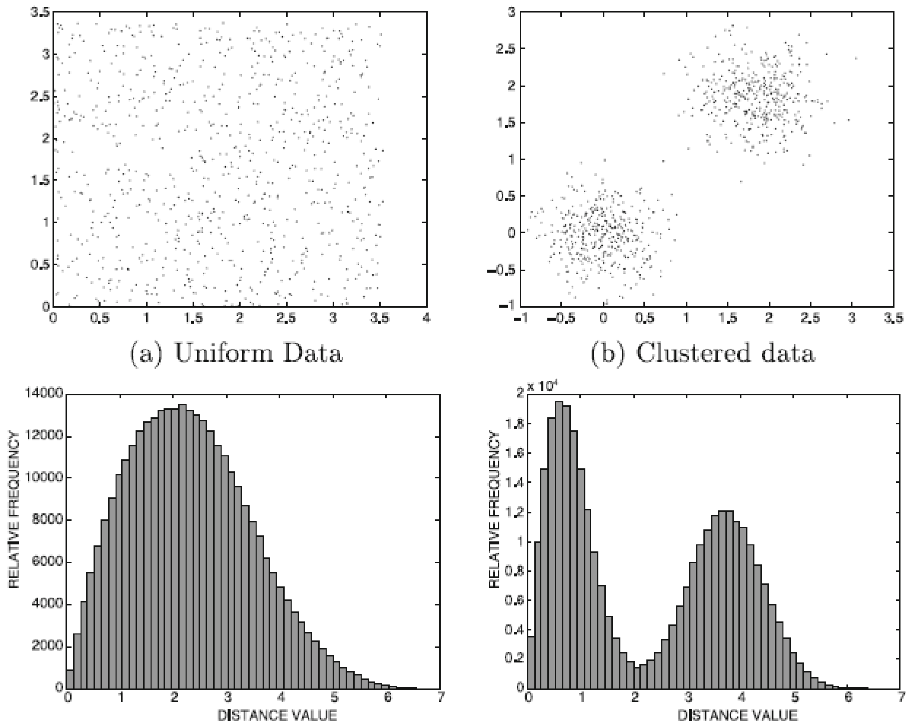
Precursor to other analysis

- use as a preprocessing step for classification and outlier detection

Attribute Selection

With 1,000s of attributes (dimensions) which ones are relevant and which one are not?

avoid



(c) Distance distribution (uniform) (d) Distance distribution (clustered)

keep

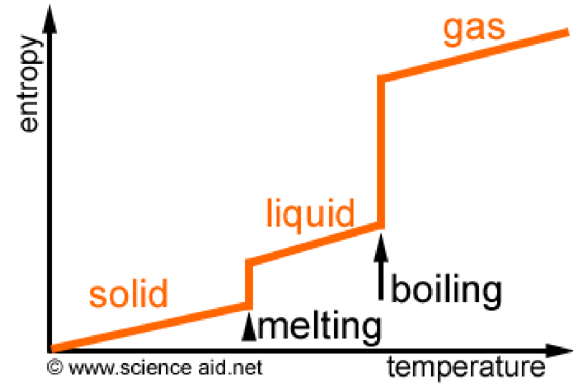
Attribute Selection

How to measure attribute "worthiness"

- use entropy

Entropy

- originates in thermodynamics
- measures lack of order or predictability



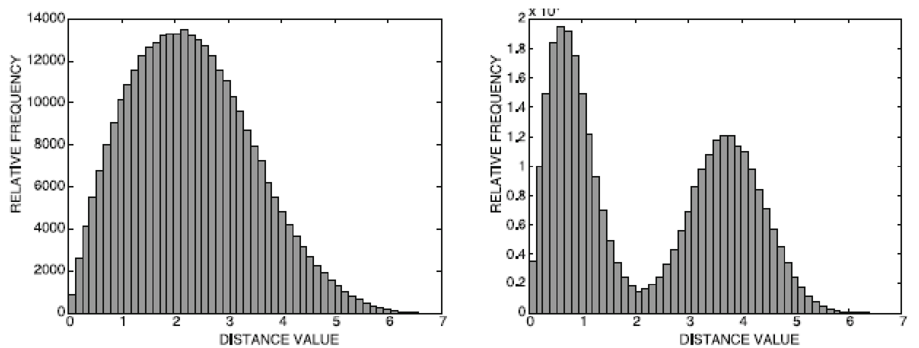
Entropy in statistics and information theory

- has a value of 1 for uniform distributions (not predictable)
- knowing the value has a lot of information (high surprise)
- a value of 0 for a constant value (fully predicable)
- knowing the value has zero information (low surprise)

ENTROPY

Assume m bins, $1 \leq i \leq m$: $$E = -\sum_{i=1}^{m}[p_i \log(p_i) + (1-p_i)\log(1-p_i)].$$

E low



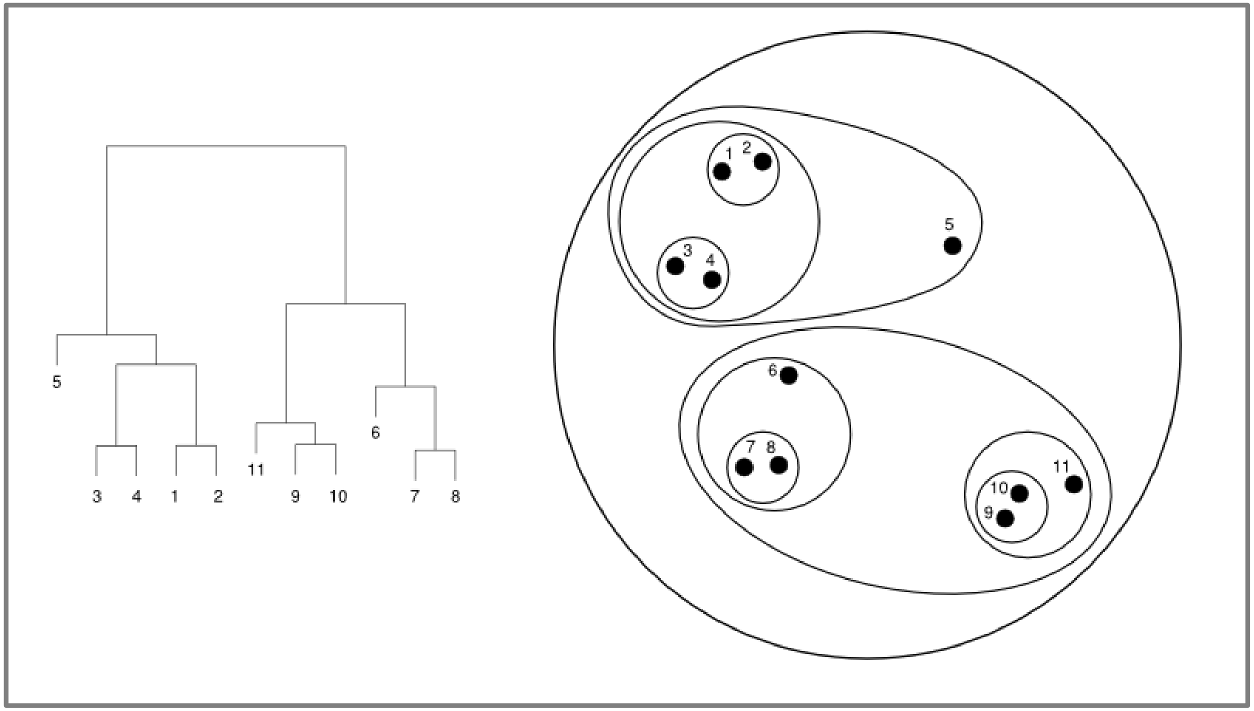
(c) Distance distribution (uniform) (d) Distance distribution (clustered)

E high

Algorithm:

- start with all attributes and compute distance entropy
- greedily eliminate attributes that reduce the entropy the most
- stop when entropy no longer reduces or even increases

Hierarchical Clustering



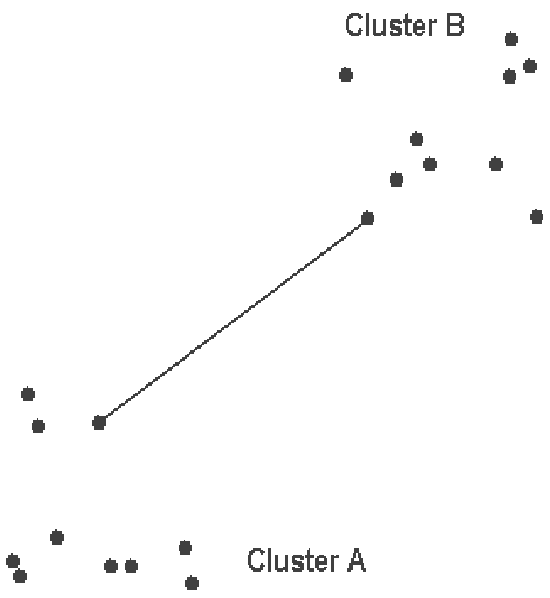
Two options:

- top down (divisive)
- bottom up (agglomerative)

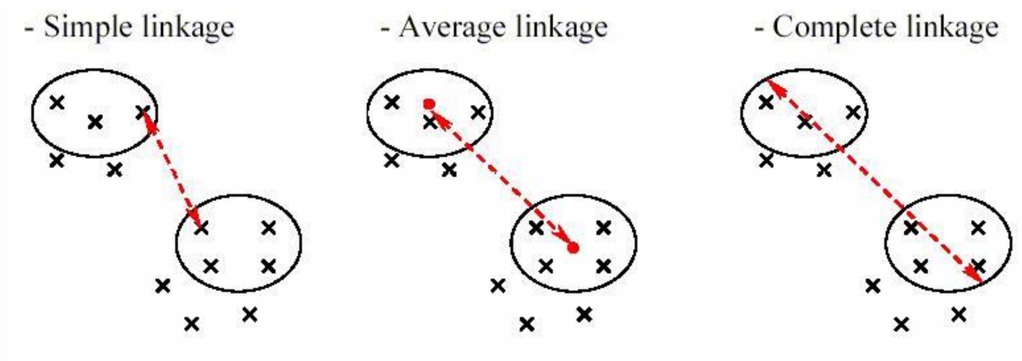
Bottom-Up Agglomerative Methods

Algorithm *AgglomerativeMerge*(Data: $\mathcal{D}$)
begin
  Initialize $n \times n$ distance matrix M using $\mathcal{D}$;
  repeat
    Pick closest pair of clusters i and j using M;
    Merge clusters i and j;
    Delete rows/columns i and j from M and create
      a new row and column for newly merged cluster;
    Update the entries of new row and column of M;
  until termination criterion;
  return current merged cluster set;
end

How to merge?



MERGE CRITERIA



Single linkage

- distance = minimum distance between all $m_i \cdot m_j$ pairs of objects
- joins the closest pair

Worst (complete) linkage

- distance = maximum distance between all $m_i \cdot m_j$ pairs of objects
- joins the pair furthest apart

Group-average linkage

- distance = average distance between all object pairs in the groups

Other methods:

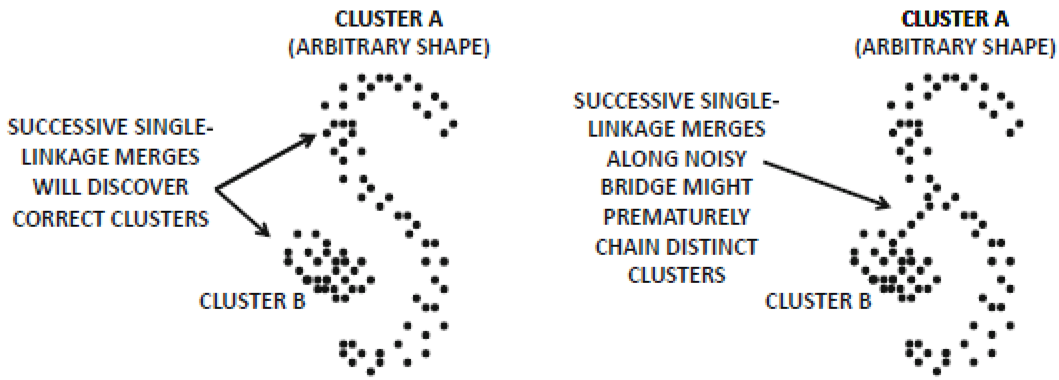
- closest centroid, variance-minimization, Ward's method

Comparison

Centroid-based methods tend to merge large clusters

Single linkage method can merge chains of closely related points to discover clusters of arbitrary shape

- but can also (inappropriately) merge two unrelated clusters, when the chaining is caused by noisy points between two clusters



(a) Good case with no noise

(b) Bad case with noise

Comparison

Complete (worst-case) linkage method tends to create spherical clusters with similar diameter

- will break up the larger clusters into smaller spheres
- also gives too much importance to data points at the noisy fringes of a cluster

The group average, variance, and Ward's methods are more robust to noise due to the use of multiple linkages in the distance computation

Hierarchical methods are sensitive to a small number of mistakes made during the merging process

- can be due to noise
- no way to undo these mistakes

Computation

Needs a heap of sorted distances

- needs $O(n^2 \cdot d)$ time and $O(n^2)$ space
- heap maintenance is $O(n^2 \cdot \log(n))$
- overall time is $O(n^2 \cdot d + n^2 \cdot \log(n))$
- problematic for large n and d

The CURE clustering algorithm improves on this

- makes use of the concept of well-scattered points
- carefully chosen representative points from clusters to approximately compute the single-linkage criterion

The DBSCAN algorithm overcomes problems with single-linkage

- excludes the noisy points between clusters from the merging process to avoid undesirable chaining effects

Top-Down Divisive Methods

Algorithm *GenericTopDownClustering*(Data: $\mathcal{D}$, Flat Algorithm: $\mathcal{A}$)
begin
  Initialize tree $\mathcal{T}$ to root containing $\mathcal{D}$;
  repeat
    Select a leaf node L in $\mathcal{T}$ based on pre-defined criterion;
    Use algorithm $\mathcal{A}$ to split L into $L_1 \dots L_k$;
    Add $L_1 \dots L_k$ as children of L in $\mathcal{T}$;
  until termination criterion;
end

Use a generic clustering method as algorithm *A*

- 2-means algorithm
- select heaviest node and split
- or try to balance the tree

K-Means and EM Clustering

see separate slides by Eamonn Keogh