Lecture 17: Unsupervised Learning

COMP90049

Semester 2, 2020

Lida Rashidi, CIS

Acknowledgement: Jeremy Nicholson, Tim Baldwin & Karin Verspoor

Clustering

A possible clustering of the weather dataset

Outlook	Temperature	Humidity	Windy	Cluster
sunny	**hot**	**high**	FALSE	?
sunny	**hot**	**high**	TRUE	?
overcast	**hot**	**high**	FALSE	?
rainy	mild	high	FALSE	?
rainy	**cool**	**normal**	FALSE	?
rainy	**cool**	**normal**	TRUE	?
overcast	**cool**	**normal**	TRUE	?
sunny	mild	**high**	FALSE	?
sunny	**cool**	**normal**	FALSE	?
rainy	**mild**	**normal**	FALSE	?
sunny	**mild**	**normal**	TRUE	?
overcast	**mild**	high	TRUE	?
overcast	**hot**	normal	FALSE	?
rainy	**mild**	high	TRUE	?

Clustering over the weather dataset (cf. outputs)

Outlook	Temperature	Humidity	Windy	Cluster	Play
sunny	hot	high	FALSE	0	no
sunny	hot	high	TRUE	0	no
overcast	hot	high	FALSE	0	yes
rainy	mild	high	FALSE	1	yes
rainy	cool	normal	FALSE	1	yes
rainy	cool	normal	TRUE	1	no
overcast	cool	normal	TRUE	1	yes
sunny	mild	high	FALSE	0	no
sunny	cool	normal	FALSE	1	yes
rainy	mild	normal	FALSE	1	yes
sunny	mild	normal	TRUE	1	yes
overcast	mild	high	TRUE	1	yes
overcast	hot	normal	FALSE	0	yes
rainy	mild	high	TRUE	1	no

Clustering

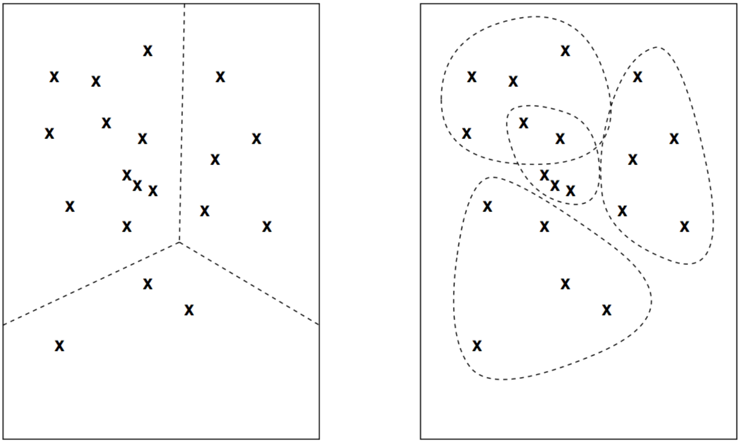
- Clustering is ***unsupervised***
- The class of an example is not known (or at least not used)
- Finding groups of items that are *similar*
- Success often measured subjectively
- Applications in pattern recognition, spatial data analysis, medical diagnosis, . . .

Clustering, basic contrasts

- Exclusive vs. overlapping clustering
  - Can an item be in more than one cluster?
- Deterministic vs. probabilistic clustering (Hard vs. soft clustering)
  - Can an item be partially or weakly in a cluster?
- Hierarchical vs. partitioning clustering
  - Do the clusters have subset relationships between them? e.g. nested in a tree?
- Partial vs. complete
  - In some cases, we only want to cluster some of the data
- Heterogenous vs. homogenous
  - Clusters of widely different sizes, shapes, and densities
- Incremental vs. batch clustering
  - Is the whole set of items clustered in one go?

Exclusive vs. overlapping clustering

- Can an item be in more than one cluster?



Deterministic vs. probabilistic clustering

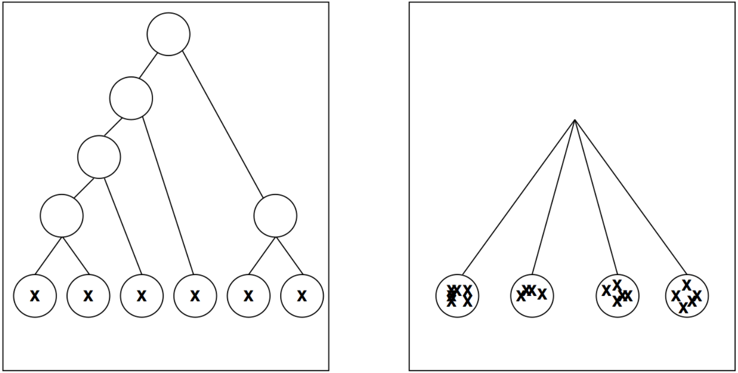
- Can an item be partially or weakly in a cluster?

Instance	*Cluster*
1	2
2	3
3	2
4	1
5	2
6	2
7	4
⋮	⋮

	Cluster			
Instance	1	2	3	4
1	0.01	0.87	0.12	0.00
2	0.05	0.25	0.67	0.03
3	0.00	0.98	0.02	0.00
4	0.45	0.39	0.08	0.08
5	0.01	0.99	0.00	0.00
6	0.07	0.75	0.08	0.10
7	0.23	0.10	0.20	0.47
⋮	⋮			

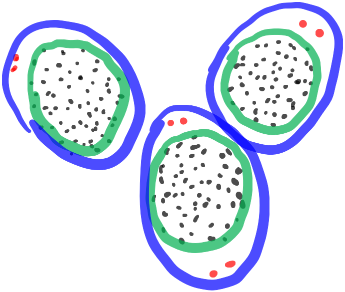
Hierarchical vs. partitioning clustering

- Do the clusters have subset relationships between them? e.g. nested in a tree?



Partial vs. complete

- In some cases, we only want to cluster some of the data

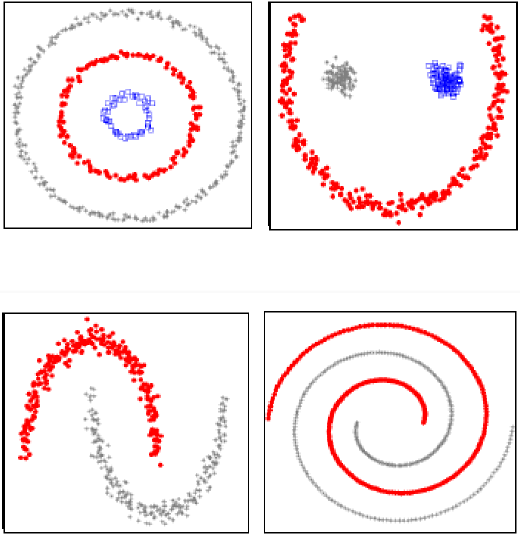


– partial

– Complete

Heterogenous vs. homogenous

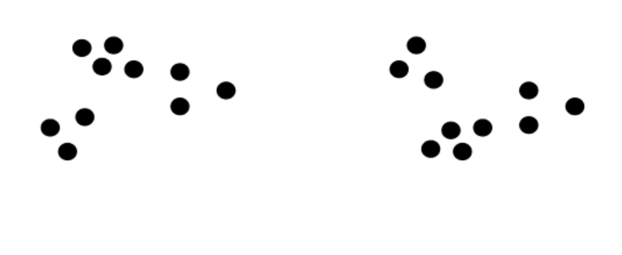
- Clusters of widely different sizes, shapes, and densities



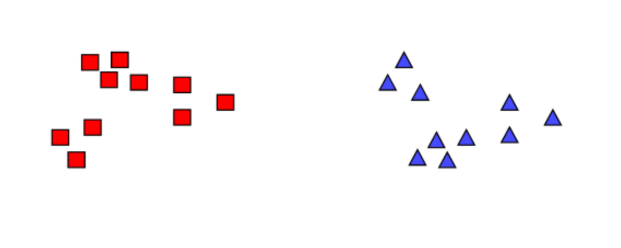
Clustering, Desiderata

- Scalability; high dimensionality
- Ability to deal with different types of attributes
- Discovery of clusters with arbitrary shape
- Able to deal with noise and outliers
- Insensitive to order of input records

What is a good clustering?

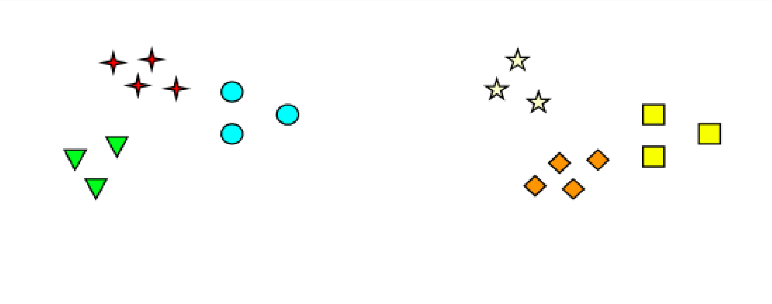


Two clusters?



Four clusters?

Six clusters?



Unsupervised.

- Measures the goodness of a clustering structure without respect to external information. Includes measures of **cluster cohesion** (compactness, tightness), and measures of **cluster separation** (isolation, distinctiveness).

Supervised.

- Measures the extent to which the clustering structure discovered by a clustering algorithm matches some external structure. For instance, *entropy* can measure how well cluster labels match externally supplied class labels.

Unsupervised Evaluation I

A "good" cluster should have one or both of:

- **High cluster cohesion**: instances in a given cluster should be closely related to each other

$$cohesion((C_i) = \frac{1}{\sum_{x,y \in C_i} Distance(x, y)}$$

- **High Cluster Separation** instances in different clusters should be distinct from each other

$$seperation(C_i, C_j) = \sum_{x \in C_i, y \in C_{j \neq i}} Distance(x, y)$$

Unsupervised Evaluation II

Most common measure for evaluating cluster quality is **Sum of Squared Error (SSE)** or *Scatter*

- For each point, the error is the distance to the nearest cluster
- To get SSE, we square these errors and sum them.

  $\sum_{i=1}^{k} \sum_{x \in C_i} dist^2(m_i, x)$
- x is a data point in cluster C_i and m_i is the representative point for cluster C_i
- Can show that the m_i that minimises SSE corresponds to the center (mean) of the cluster
- Given two clusters, we can choose the one with the smallest error
- One easy way to reduce SSE is to increase k, the number of clusters
- However, a good clustering with smaller k can have a lower SSE than a poor clustering with higher k

Sum of squared errors: Example

Cluster 1 centroid: sunny	mild	high	no	
sunny	hot	high	no	$1^2 = 1$
sunny	hot	high	yes	$2^2 = 4$
overcast	hot	high	no	$2^2 = 4$
rainy	mild	high	no	$1^2 = 1$
sunny	mild	high	no	0
overcast	mild	high	yes	$2^2 = 4$
rainy	mild	high	yes	$2^2 = 4$

$SSE_1 = 18$

Cluster 2 centroid: overcast	cool	normal	yes
rainy	cool	normal	yes
overcast	cool	normal	yes
sunny	cool	normal	no
overcast	mild	normal	no
sunny	mild	normal	yes
overcast	hot	normal	no
rainy	cool	normal	no

$SSE_2 = 20$

$$SSE = SSE_1 + SSE_2 = 38$$

Supervised Evaluation

- If labels are available, evaluate the degree to which class labels are consistent within a cluster and different across clusters:

$$purity = \sum_{i=1}^{k} \frac{|C_i|}{N} \max_j P_i(j)$$

$$entropy = \sum_{i=1}^{k} \frac{|C_i|}{N} H(x_i)$$

- where x_i is the distribution of class labels in cluster i

Supervised Evaluation Example I

- Calculate the entropy and purity of the following cluster output

Cluster	Play = yes	Play = no
1	4	0
2	4	4

$$entropy_1 = -1 \times log(1) - 0 \times log(0) = 0$$

$$entropy_2 = -0.5 \times log(0.5) - 0.5 \times log(0.5) = 1$$

$$purity_1 = max(1, 0) = 1$$

$$purity_2 = max(0.5, 0.5) = 0.5$$

THE UNIVERSITY OF MELBOURNE

Supervised Evaluation Example I

- Calculate the entropy and purity of the following cluster output

Cluster	Play = yes	Play = no	Entropy	Purity
1	4	0	0	1
2	4	4	1	0.5
Total:			**0.67**	**0.67**

$$entropy = \frac{4}{12} \times 0 + \frac{8}{12} \times 1 = 0.67$$

$$purity = \frac{4}{12} \times 1 + \frac{8}{12} \times 0.5 = 0.67$$

Supervised Evaluation Example II

- Calculate the entropy and purity of the following cluster output

Cluster	Play = yes	Play = no
1	2	0
2	6	4

$$entropy_1 = -1 \times log(1) - 0 \times log(0) = 0$$

$$entropy_2 = -0.6 \times log(0.6) - 0.4 \times log(0.4) = 0.97$$

$$purity_1 = max(1, 0) = 1$$

$$purity_2 = max(0.6, 0.4) = 0.6$$

THE UNIVERSITY OF MELBOURNE

Supervised Evaluation Example II

- Calculate the entropy and purity of the following cluster output

Cluster	Play = yes	Play = no	Entropy	Purity
1	2	0	0	1
2	6	4	0.97	0.6
Total:			**0.81**	**0.67**

$$entropy = \frac{2}{12} \times 0 + \frac{10}{12} \times 0.97 = 0.81$$

$$purity = \frac{2}{12} \times 1 + \frac{10}{12} \times 0.6 = 0.67$$

Methods

Similarity / Proximity / Closeness

- Clustering finds groups of instances in the dataset which are similar or close to each other within a group while being different or separated from other clusters/clusters.
- A key component of any clustering algorithm is a measurement of the distance between any points.

Measuring Similarity / Proximity / Closeness

- Data points in Euclidean space
  - Euclidean distance
  - Manhattan (L1) distance
- Discrete values
  - Hamming distance (discrepancy between the bit strings)

d	a	b	c
a	0	1	1
b	1	0	1
c	1	1	0

  For two bit strings, the number of positions at which the corresponding symbols are different
- Documents
  - Cosine similarity
  - Jaccard measure
- Other measures
  - Correlation
  - Graph-based measures

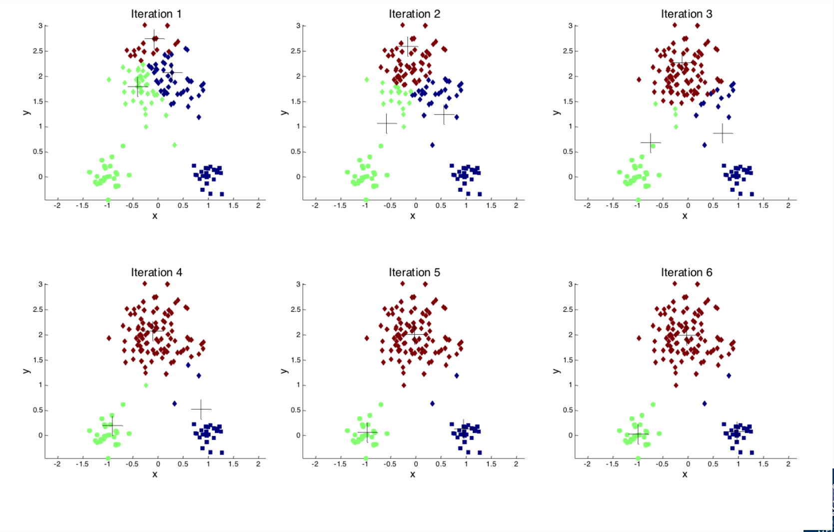
THE UNIVERSITY OF MELBOURNE

Given k, the *k*-means algorithm is implemented in four steps:

1. Select *k* points to act as seed cluster centroids
2. **repeat**
3. Assign each instance to the cluster with the **nearest centroid**
4. Recompute the centroid of each cluster
5. **until** the centroids don't change

- Exclusive, deterministic, partitioning, batch clustering method

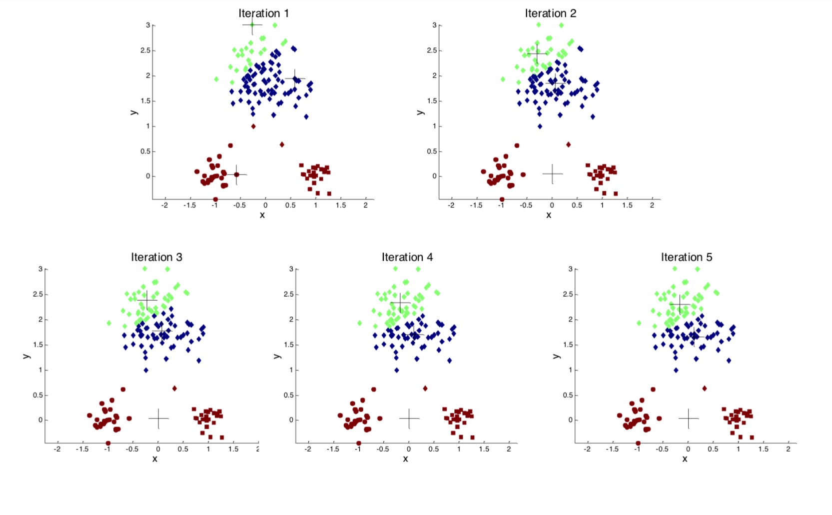
Example, Iterations



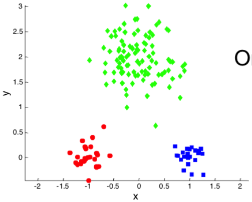
k-means Clustering – Details

- Initial centroids are often chosen randomly.
  - Clusters produced vary from one run to another.
- The centroid is (typically) the mean of the points in the cluster.
- 'Nearest' is based on proximity/similarity/etc. metric.
- K-means will converge for common similarity measures mentioned above.
  - Most of the convergence happens in the first few iterations.
  - Often the stopping condition is changed to
    'Until relatively few points change clusters'
    (this way the stopping criterion will not depend on the type of similarity or dimensionality)

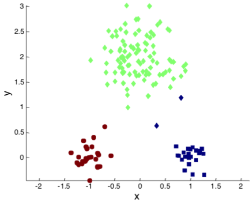
Example, Impact of initial seeds



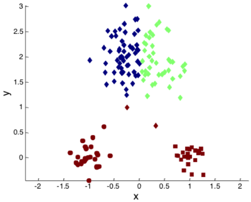
Example, Different outcomes



Original Points

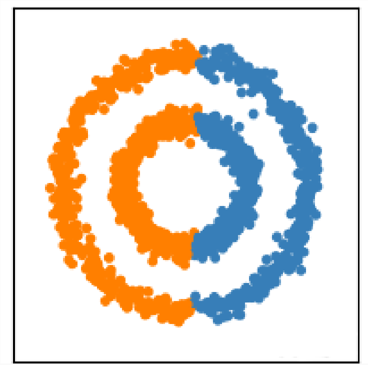


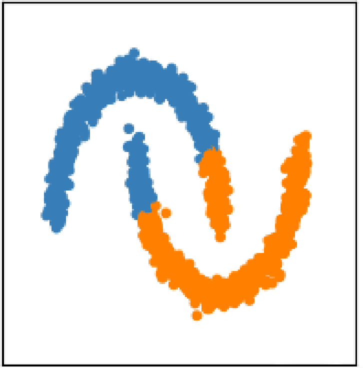
Optimal Clustering



Sub-optimal Clustering

Shortcomings of k-means





k-means, Pros and Cons

Strengths:

- relatively efficient:
  - $O(ndki)$, where n is no. instances, d is no. attributes, k is no. clusters, and i is no. iterations; normally $k, i \ll n$
  - Unfortunately we cannot a priori know the value of i!
- can be extended to hierarchical clustering

Weaknesses:

- tends to converge to local minimum; sensitive to seed instances (try multiple iterations with different seeds?)
- need to specify k in advance
- not able to handle non-convex clusters, or clusters of differing densities or sizes
- “mean” ill-defined for nominal or categorical attributes
- may not work well when the data contains outliers

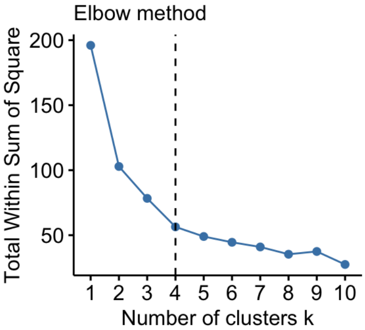
THE UNIVERSITY OF MELBOURNE

How to choose the number of clusters?

- calculate SSE for different number of clusters K

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} (x - m_i)^2$$

- As K increases, we will have a smaller number of instances in each cluster $\rightarrow$ SSE decreases
- Elbow method: K increases to $K+1$, the drop of SSE starts to diminish



Hierarchical Clustering

Bottom-up (= agglomerative) clustering

- Start with single-instance clusters
- At each step, join the two closest clusters (in terms of margin between clusters, distance between mean, ...)

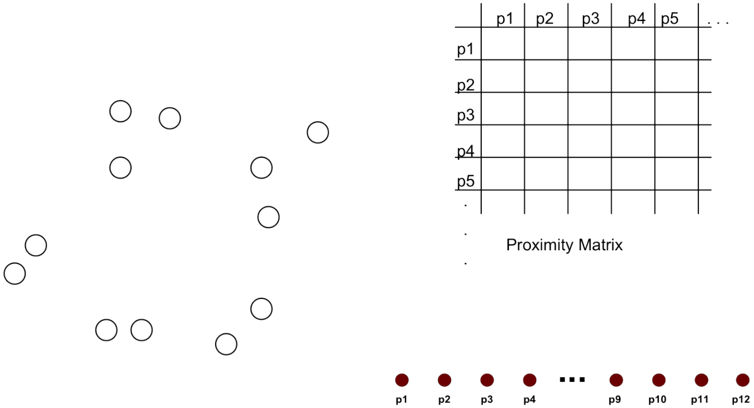
Top-down (= divisive) clustering

- Start with one universal cluster
- Find two partitioning clusters
- Proceed recursively on each subset
- Can be very fast

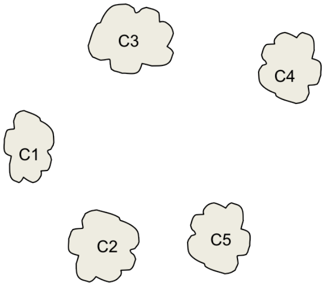
In contrast to k-means clustering, hierarchical clustering only requires a measure of similarity between *groups* of data points (no seeds, no k value).

1. Compute the proximity matrix, if necessary.
2. **repeat**
3. Merge the closest two clusters
4. Update the proximity matrix to reflect the proximity between the new cluster and the original clusters
5. **until** Only one cluster remains

Example, Step 1

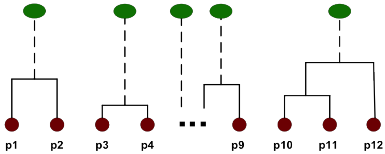


Example, Step 2

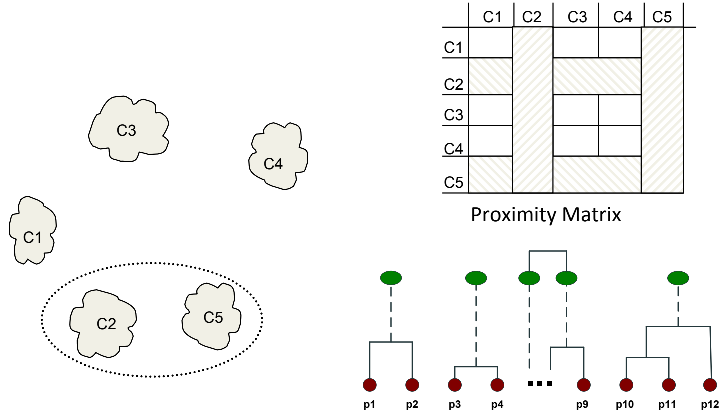


	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

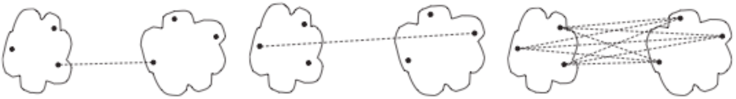
Proximity Matrix



Example, Step 3



Graph-based measure of Proximity



(a) MIN (single link.) (b) MAX (complete link.) (c) Group average.

Updating the proximity matrix:

- Single Link: *Minimum* distance between any two points in the two clusters. (most similar members)
- Complete Link: *Maximum* distance between any two points in the two clusters. (most dissimilar members)
- Group Average: *Average* distance between all points (pairwise).

Agglomerative Clustering Example

	1	2	3	4	5
1	1.00	0.90	0.10	0.65	0.20
2	0.90	1.00	0.70	0.60	0.50
3	0.10	0.70	1.00	0.40	0.30
4	0.65	0.60	0.40	1.00	0.80
5	0.20	0.50	0.30	0.80	1.00

What are the two closest points?

Agglomerative Clustering Example

	1	2	3	4	5
1	1.00	0.90	0.10	0.65	0.20
2	0.90	1.00	0.70	0.60	0.50
3	0.10	0.70	1.00	0.40	0.30
4	0.65	0.60	0.40	1.00	0.80
5	0.20	0.50	0.30	0.80	1.00

Merge points 1 & 2 into a new cluster: 6

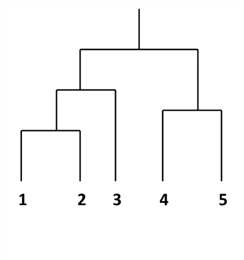
Update (single link):

	1	2	3	4	5	6
6	—	—	0.70	0.65	0.50	1.00

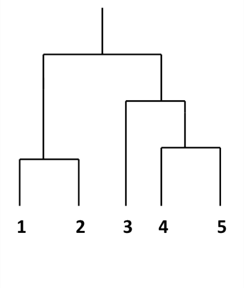
Update (complete link):

	1	2	3	4	5	6
6	—	—	0.10	0.60	0.20	1.00

	1	2	3	4	5
1	1.00	0.90	0.10	0.65	0.20
2	0.90	1.00	0.70	0.60	0.50
3	0.10	0.70	1.00	0.40	0.30
4	0.65	0.60	0.40	1.00	0.80
5	0.20	0.50	0.30	0.80	1.00



Single link



Complete link

Summary

- What basic contrasts are there in different clustering methods?
- How does *k*-means operate, and what are its strengths and weaknesses?
- What is hierarchical clustering, and how does it differ from partitioning clustering?
- What are some challenges we face when clustering data?

Resources:

Tan, Steinbach, Kumar (2006) Introduction to Data Mining. Chapter 8, Cluster Analysis `http://www-users.cs.umn.edu/~kumar/dmbook/ch8.pdf`

Jain, Dubes (1988) Algorithms for Clustering Data. `http://homepages.inf.ed.ac.uk/rbf/BOOKS/JAIN/Clustering_Jain_Dubes.pdf`

THE UNIVERSITY OF MELBOURNE