Melbourne Bioinformatics

BIOINFORMATICS + DATA SERVICES + INFRASTRUCTURE, FOR LIFE SCIENCES TODAY

THE UNIVERSITY OF MELBOURNE

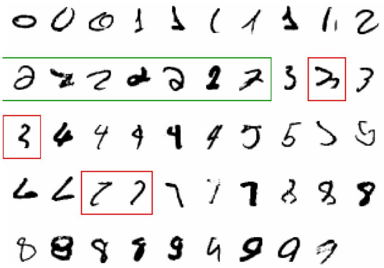
COMP90014

Algorithms for Bioinformatics

Week 10A: Unsupervised Learning - Clustering I

Machine learning: clustering

Learning

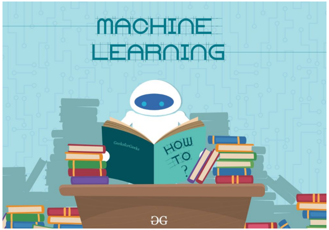


Learning is any process by which a system improves performance from experience.

- Herbert Simon

- we want computers to learn when the problem is too difficult or too expensive to program
- get the computer to program itself by showing examples of inputs and outputs.

Machine learning



Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed.

- Arthur Samuel, 1959.

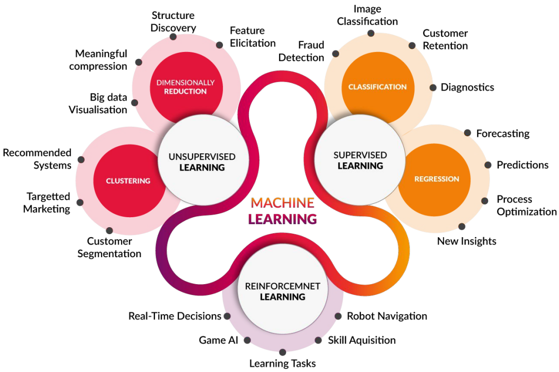
Machine Learning is the study of algorithms that:

- *Improve their performance P*
- *at some task T*
- *with experience E.*

A well-defined learning task is given by $\langle P, T, E\rangle$

- Tom Mitchell, 1998.

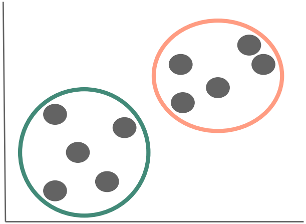
Machine learning



Supervised: Infers a mapping function from labelled training data

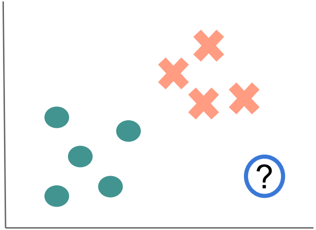
Unsupervised: Finds implicit/hidden patterns in data without pre-existing labels

Machine Learning



Unsupervised learning

- Algorithms operate on unlabelled examples
- Related to data description
- Task: Clustering

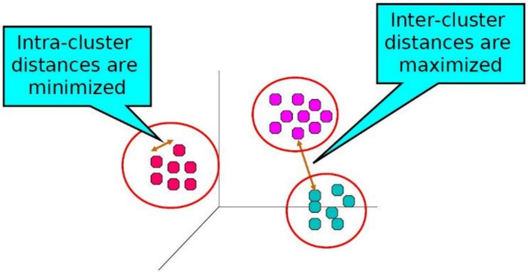


Supervised learning

- Algorithms are trained on labelled examples
- Related to function approximation
- Tasks: Classification & Regression

Machine learning: clustering

What is clustering?

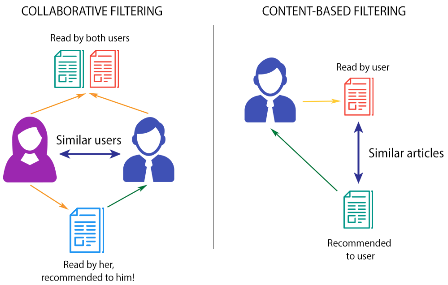


Cluster: a collection of items which are 'similar', and 'dissimilar' to items in other clusters

Clustering: organization of unlabeled data into similarity groups.

- does not require labels
- good for discovering patterns or structure in the data

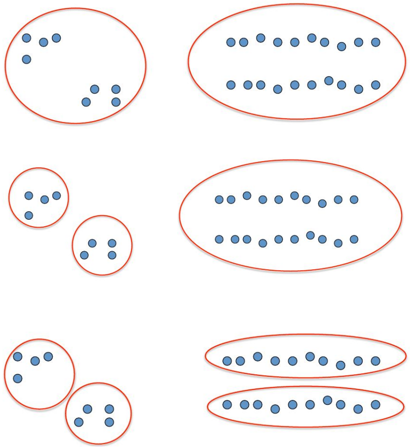
Applications



- phylogenetic tree reconstruction (agglomerative clustering)
- clustering gene expression data
- exploratory analysis and data visualization
- data compression
- recommender systems

Images: Robert Bear via Khan Academy; Tothill *et al.*, 2008 (10.1158/1078-0432.CCR-08-0196); Peter Cock via warwick.ac.uk; Amy Ma via rpubs.com; Sanket Doshi via towardsdatascience.com.

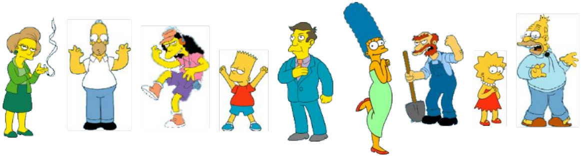
Ingredients for clustering



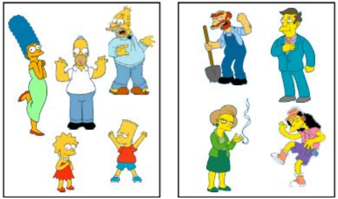
1. similarity metric
   - *e.g.* Euclidean distance
2. a function to evaluate the quality of the clusters
3. clustering algorithm

🤖 clustering is subjective
e.g. how many clusters?
- fixed k clusters
- find the best k to optimize a function

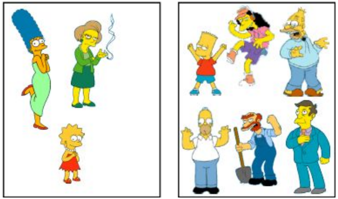
Clustering is subjective



How would you cluster this set of objects?



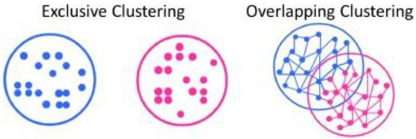
Simpsons family *vs.* school employees

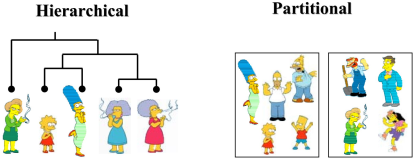


Male and female characters

Eamonn Keogh, Brandeis University

Clustering approaches





Exclusive

- Data points belong to only one cluster

Overlapping

- Data points may belong to many clusters

Hierarchical

- Assign points to “nested” clusters
- Get all possible clusters for given metric

Partitional

- Split points into “flat” independent clusters
- How many?

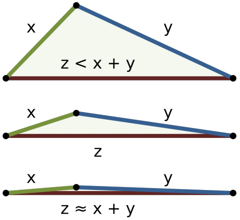
Images: Yohoshiva Basaraboyina via medium.com; Eamonn Keogh, Brandeis University

Machine learning: clustering

Ingredients for clustering: Distances

Distance matrix:

$$D(x_i, x_j) = \begin{pmatrix} 0 & 1.1 & 7.6 & 3.4 \\ 1.1 & 0 & 3.2 & 2.1 \\ 7.6 & 3.2 & 0 & 4.5 \\ 3.4 & 2.1 & 4.5 & 0 \end{pmatrix}$$



- Many hierarchical and partitional clustering use pairwise similarity/dissimilarity

- Formally, a distance metric satisfies the following properties:

Non-negativity:	$d(a, b) \geq 0$
Identity:	$d(a, a) = 0$
Symmetry:	$d(a, b) = d(b, a)$
Triangle inequality:	$d(a, c) \leq d(a, b) + d(b, c)$

Clustering depend on the distance metric

Euclidean distance

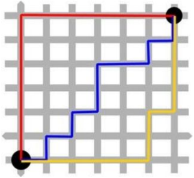
- The 'ordinary' distance used in Euclidean space.
- d: dimensions
- x: data point in d-dimensional space

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2}$$

Manhattan distance

- City-block distance or taxicab distance

$$d(x_i, x_j) = \sum_{k=1}^{d} |x_{i,k} - x_{j,k}|$$

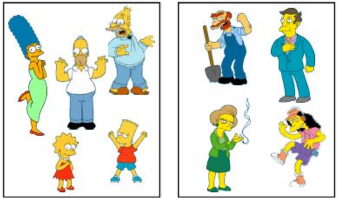


Edit distance

- Hamming distance (single-letter substitutions)
- Levenshtein distance (single-letter insertions, deletions or substitutions)
- Longest common substring (LCS) distance (single-letter insertions or deletions)

Machine learning: clustering

Clustering approaches



Images: Eamonn Keogh, Brandeis University

Exclusive

- Data points belong to only one cluster

Overlapping

- Data points may belong to many clusters

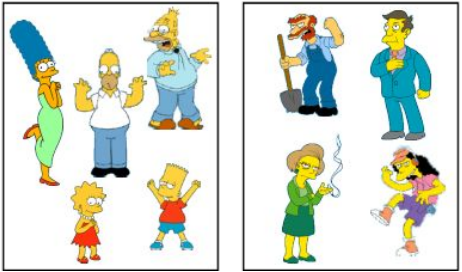
Hierarchical

- Assign points to “nested” clusters
- Get all possible clusters for given metric

Partitional

- Split points into “flat” independent clusters
- How many?

Partitional clustering



- ***nonhierarchical***
- each item is placed in exactly one of K non-overlapping clusters
- we have to decide the desired number of clusters K in advance

1. similarity metric
2. a function to evaluate the quality of the clusters
3. clustering algorithm

k-means algorithm

Input:

- data in a Euclidean space (Euclidean distance)
- parameter K (number of clusters)
- the algorithm starts with randomly located cluster centers (***centroids***)

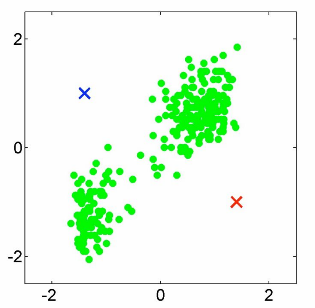
The algorithm alternates between two steps

- assignment:
  - assign each datapoint to the closest cluster
- refitting:
  - move each centroid to the ***center of gravity*** of the data assigned to it
- stopping criteria:
  - minimize an objective function
  - when no point-cluster assignments change

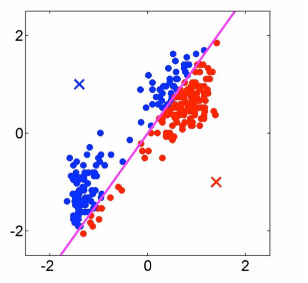
$$L = \sum_i \left\| x_i - \mu_{z_i} \right\|^2$$

k-means: MacQueen, 1967, in *Some methods for classification and analysis of multivariate observations*.

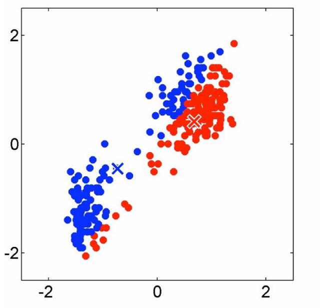
k-means algorithm ($k = 2$)



Randomly-assigned centroids

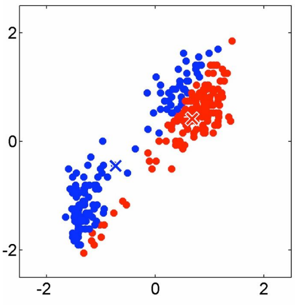


Find closest cluster

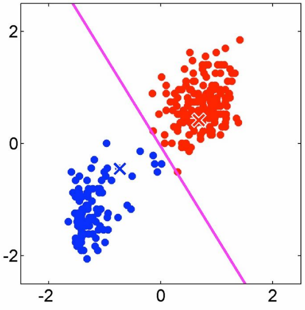


Refit centroids

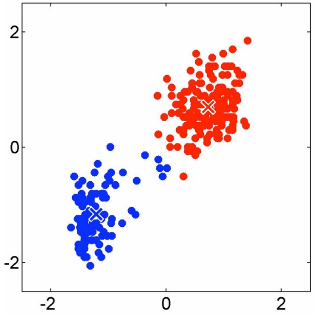
k-means algorithm ($k = 2$)



Refit centroids

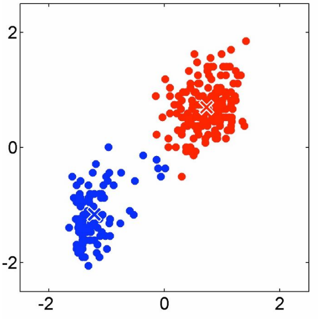


Find closest cluster

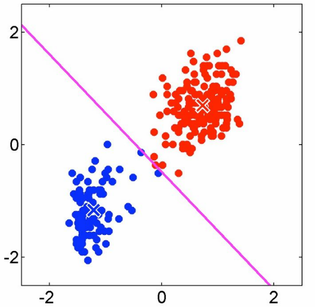


Refit centroids

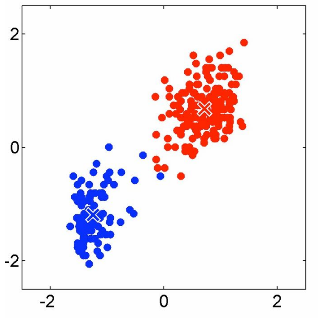
k-means algorithm ($k = 2$)



Refit centroids

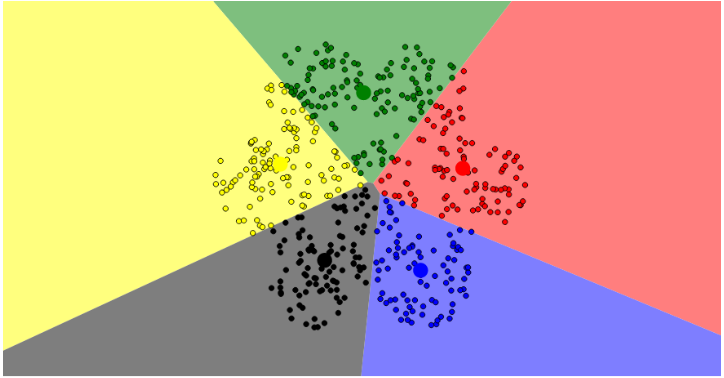


Find closest cluster



Stopping criterion

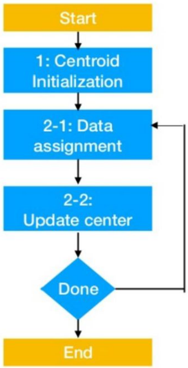
More *k*-means visualisations



naftaliharris.com

k-means algorithm

- D: items
  K: number of clusters
  ε: stopping criteria



Billy Yang via [slideshare]()

K-means (D, k, ε)

$t = 0$

Randomly initialize k centroids: $\mu_1^t, \mu_2^t, \ldots, \mu_k^t \in \mathbb{R}^d$

Repeat until ε *# criteria - centroids don't move*

For $d = 1$ to D do *# assign data points to nearest centroid*

$z_d \leftarrow \text{Argmin}_k \, ||\mu_k - x_d||$

end for

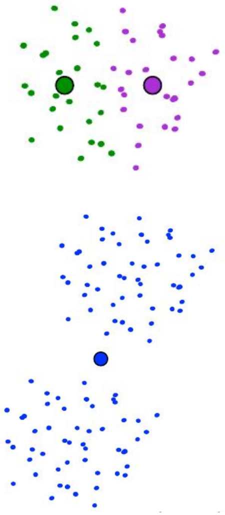
For $k = 1$ to K do *# recalculate centroids*

$\mu_k \leftarrow \textit{Mean}(\{ x_d : z_d = k \})$ *# mean(data points) for given cluster*

end for

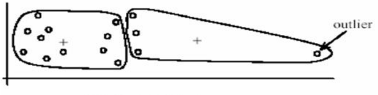
Return z *# return cluster assignments*

k-means is a heuristic

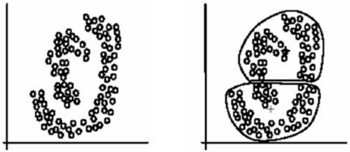


- assigning points to the closest centroid
- the algorithm is guaranteed to converge, **but** it may not converge to the optimal solution
- may reach a local minimum instead
- depends on initialisation of centroid

k-means



(A): Undesirable clusters
(B): Ideal clusters


(A): Two natural clusters
(B): k-means clusters

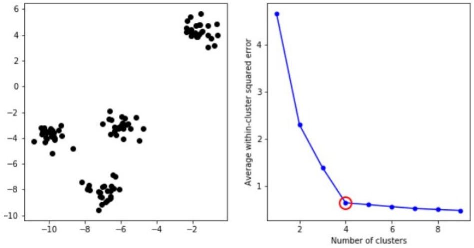
- Time complexity: $O(tkn)$
  - n is the number of data points
  - k is thenumber of clusters
  - t is the number of iterations
- Strengths
  - relatively efficient
  - simple and widely used
- Limitations
  - need to specify k in advance
  - applicable only when mean is defined
  - sensitive to outliers
  - not suitable for clusters with non-convex shapes

Anil Yadav via slideshare

k-means

How do we choose the number of clusters k?

- choose the k where objective function starts to sharply increase/decrease
- external validation set



How do we know whether we've found the global optimum, or some local minimum?

- run k-means multiple times
- check for stable convergence

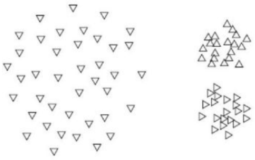
Billy Yang via slideshare

When *k*-means fails

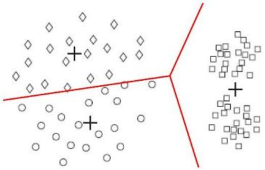
- Clusters of different densities
- Clusters of different sizes
- Non-convex clusters

Tan *et al*., Introduction to data mining. 9780273769224.

When k-means fails



(a) Original points.

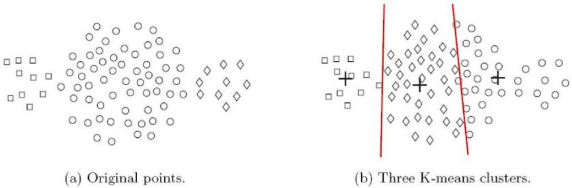


(b) Three K-means clusters.

- Clusters of different densities
- Clusters of different sizes
- Non-convex clusters

Tan *et al.*, Introduction to data mining. 9780273769224.

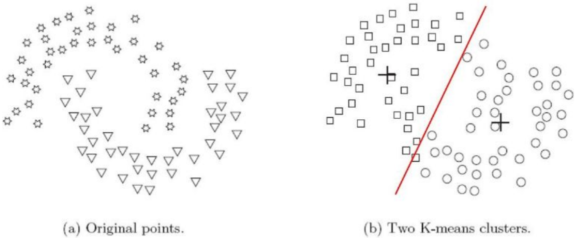
When *k*-means fails



- Clusters of different densities
- Clusters of different sizes
- Non-convex clusters

Tan *et al.*, Introduction to data mining. 9780273769224.

When *k*-means fails



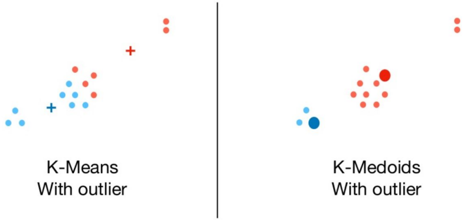
- Clusters of different densities
- Clusters of different sizes
- Non-convex clusters

Tan *et al.*, Introduction to data mining. 9780273769224.

k-medoids

Improvement to *k*-means:

- instead of the mean of the points, use the most central data point as the centroid
- we are optimising the same objective function as *k*-means



Billy Yang via slideshare

At each iterative step:

- assign points to the nearest centroid as for *k*-means
- update centroids by choosing the most central point (**medoid**)

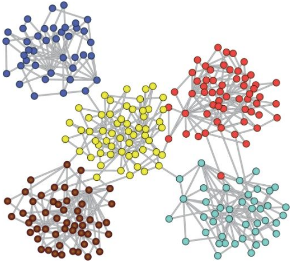
k-medoids:

- less sensitive to outliers than *k*-means
- requires extra calculation to find the most central point

Other clustering approaches

Density based

- points are part of same cluster if they are close and in a dense region
- finds non-convex, arbitrarily-shaped clusters
- uses distance between points



Probabilistic

- each cluster represented by a parametric distribution *e.g.* Gaussian
- Gaussian mixture models
- overlapping clusters

Network-based

- instead of distances we use a graph, with edges connecting nodes
- find clusters that are ***densely connected***

Benedek Rozemberczki, benedekrozemberczki/awesome-community-detection

Thank you!

Today: Unsupervised Learning - Clustering I

Next time: Unsupervised Learning - Clustering II