UNIVERSITY OF BIRMINGHAM

Unsupervised Learning

Lecturer: Dr. Sharu Theresa Jose

Learning Outcomes

- Differentiate between supervised and unsupervised learning
- Applications of unsupervised learning in real-world
- Fundamentals of clustering algorithms

Overview of Lecture

- Introduction to Unsupervised Learning
  - Real world applications
- Clustering – Basic Principles
  - Measures of similarity
  - Normalization of data
  - Distance matrix
- Clustering Algorithms - Introduction

From Supervised to Unsupervised Learning

Notation

- $\boldsymbol{x}^{(i)} = (x_1^{(i)}, x_2^{(i)}, \dots, x_m^{(i)})$ denotes the ith **feature vector** consisting of m **feature attributes** $x_j^{(i)}$ for $j = 1, \dots, m$.
- Lower case letter y_i denotes the corresponding output label.

attributes

Class labels	Sepal length	Sepal width	Petal length	Petal width
Iris setosa	5.1	3.5	1.4	0.2
Iris versicolor	4.9	3	1.4	0.2
Iris virginica	4.7	3.2	1.3	0.2

y_1

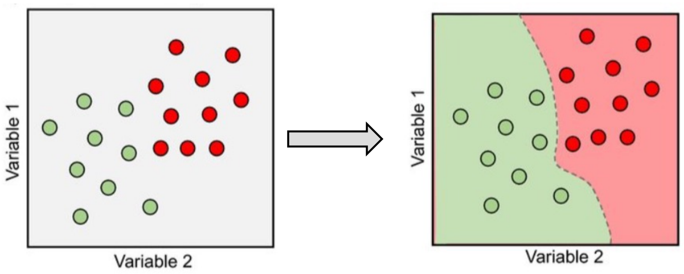
$\boldsymbol{x}^{(1)}$

Supervised Learning

- Labeled observations: Each observation is a tuple $(\boldsymbol{x}, y)$ of feature vector $\boldsymbol{x}$ and output label y which are related according to an unknown function $f(\boldsymbol{x}) = y$.
- During training: Learn the relationship between $\boldsymbol{x}$ and y, i.e., find a function (or model) $h(\boldsymbol{x})$ that best fits the observations
- Goal: Learned model accurately predicts the output label of a previously unseen, test feature input (generalization)
- Labels : 'Teachers' during training, and 'validator' of results during testing

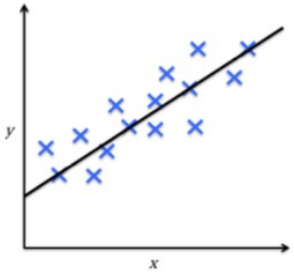
Classification

- Predict categorical labels, i.e., $y \in \{1,2,..K\}$ is discrete
- Example: multi-class handwritten digits



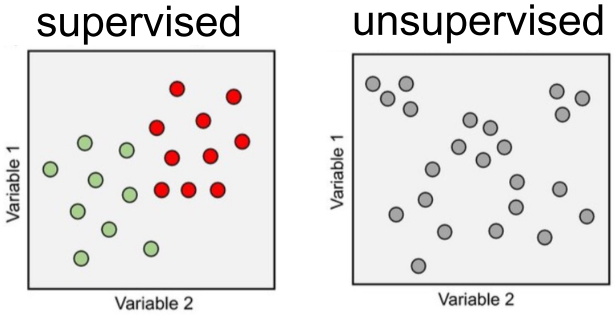
Regression

- Predict continuous-valued labels
- Example: predict students' scores

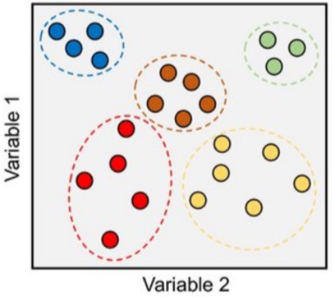


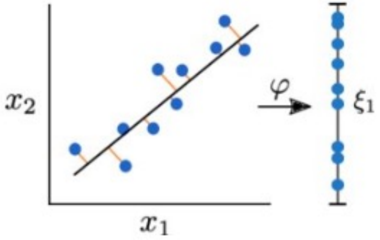
Unsupervised Learning

- Unlabeled data set of feature vectors



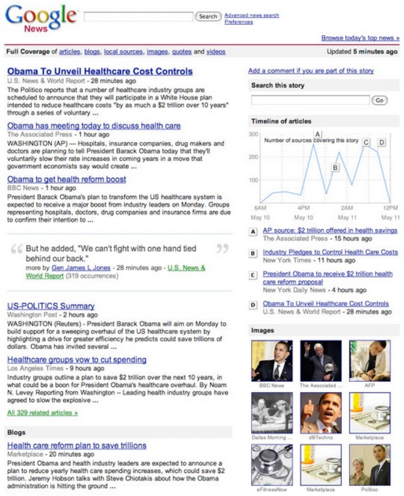
- What can we deduce?
  - find sub-groups (or clusters) among observations with *similar* traits (clustering)
  - find patterns within feature vector to identify a lower dimensional representation (dimensionality reduction)



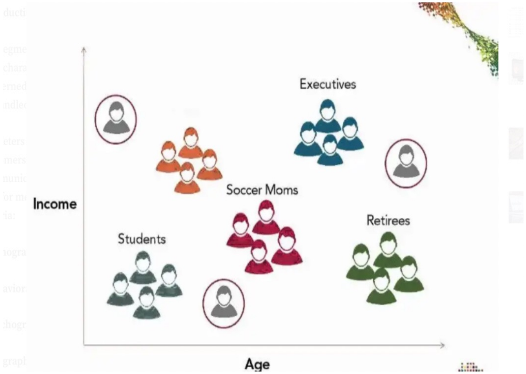


Clustering: Real World Applications

Google News



Market Segmentation

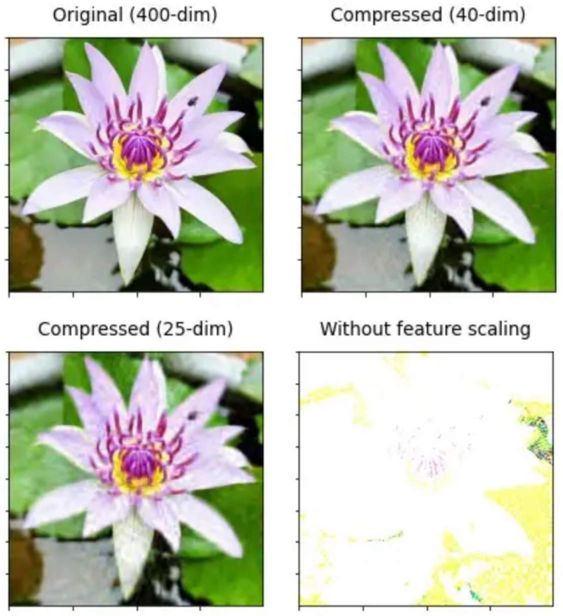


Social Network Analysis

Clusters are potential 'classes'; clustering algorithms automatically find 'classes'

Dimensionality Reduction: Application

Image Compression



Techniques for dimensionality reduction:

- Principal component analysis (PCA)
- Non-negative matrix factorization (NMF)
- Linear discriminant analysis (LDA)

Challenges

- No simple goal as in supervised learning
- Validation of results is subjective
- Often more used in exploratory data analysis

Why unsupervised learning?

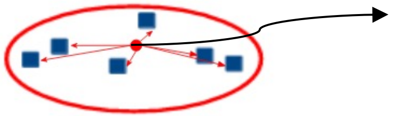
- Labeled data expensive and difficult to collect; unlabeled data cheap and abundant
- Compressed representation saves on storage and computation
- Reduce noise, irrelevant attributes in high dimensional data
- Pre-processing step for supervised learning

UNIVERSITY OF BIRMINGHAM

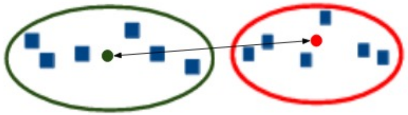
Clustering: Basic Principles

What is clustering?

- Find natural groupings among observations
- Segment observations into clusters/groups such that
  - Objects within a cluster have high similarity (high intra-cluster similarity)



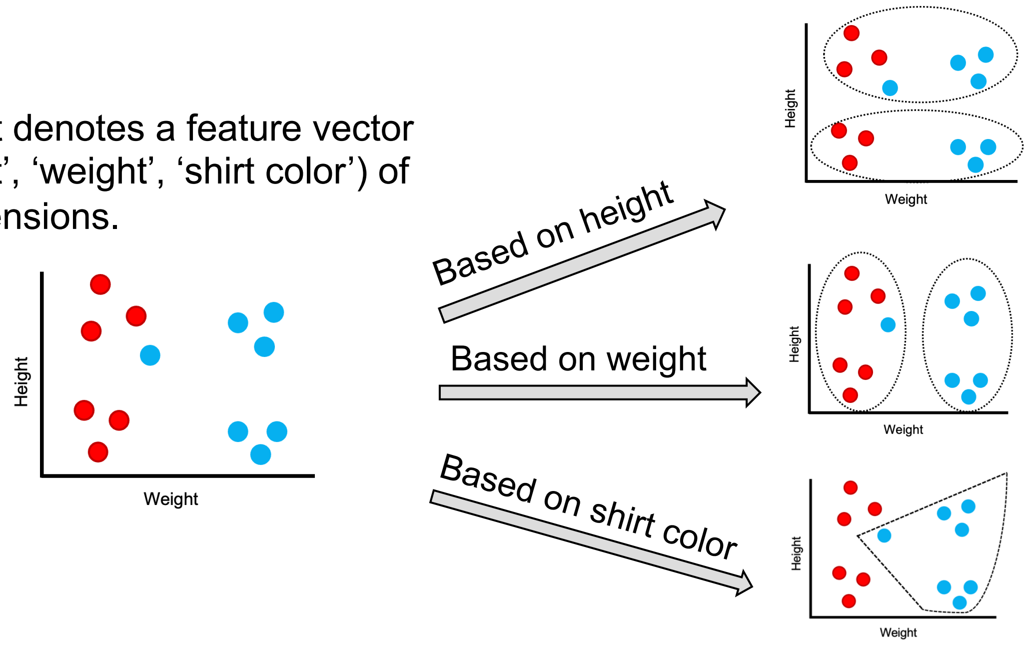
  - Objects across clusters have low similarity (low inter-cluster similarity)



UNIVERSITY OF BIRMINGHAM

Example 1: How do you cluster the following points?

Each point denotes a feature vector $\boldsymbol{x}$ =('height', 'weight', 'shirt color') of three dimensions.



Clustering is **subjective**: clusters are formed based on a user-specified measure of similarity that depends on domain knowledge.

UNIVERSITY OF BIRMINGHAM

Clustering as Unsupervised Classification

- Supervised classification: labeled observations available
- Clustering creates a labeling of observations with cluster labels
- Labels are derived only from the observations
- Clustering = unsupervised classification

Example 2: Clustering of Mammals

- Problem: group mammals into three clusters (herbivores, carnivores, omnivores) based on the feature attributes.
- How do we compute similarity between mammals?

Data Matrix

	Incisor (top)	Canine (top)	Molar (top)	Pre-molar (top)	Weight (pounds)	
Badger	3	1	3	1	10	$\boldsymbol{x}^{(1)}$
Bear	3	1	4	2	278	
Cow	0	0	3	3	400	
Dog	3	1	4	2	20	
Fox	3	1	4	2	5	$\boldsymbol{x}^{(5)}$

UNIVERSITY OF BIRMINGHAM

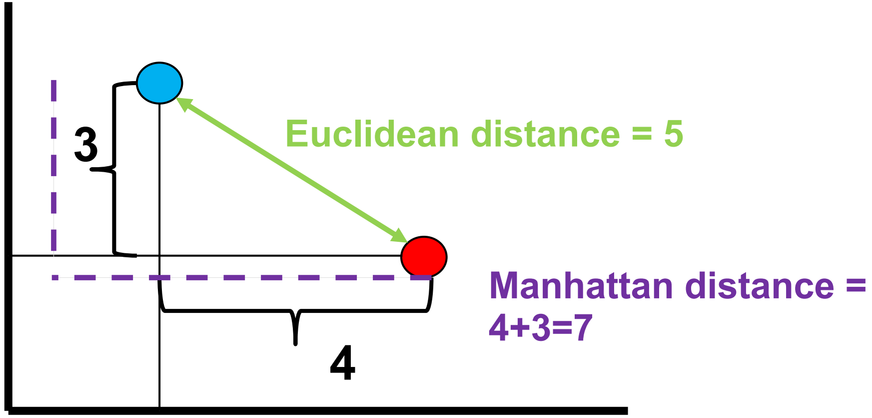
Measures of similarity: Distance functions

- Measures the strength of relationship between any two feature vectors.
- Examples of distance measures between real-valued feature vectors $\boldsymbol{x}^{(\mathbf{1})} = (x_1^{(1)}, \ldots, x_m^{(1)})$ and $\boldsymbol{x}^{(\mathbf{2})} = (x_1^{(2)}, \ldots, x_m^{(2)})$:

Euclidean	$d_{Euc}\left(\boldsymbol{x}^{(\mathbf{1})}, \boldsymbol{x}^{(\mathbf{2})}\right) = \sqrt{\underbrace{\left(x_1^{(1)} - x_1^{(2)}\right)^2} + \cdots + \left(x_m^{(1)} - x_m^{(2)}\right)^2}$
Manhattan	$d_{Man}\left(\boldsymbol{x}^{(\mathbf{1})}, \boldsymbol{x}^{(\mathbf{2})}\right) = \sum_{j=1,..m} \underbrace{\lvert x_j^{(1)} - x_j^{(2)} \rvert}$
Chebychev	$d_{Cheb}\left(\boldsymbol{x}^{(\mathbf{1})}, \boldsymbol{x}^{(\mathbf{2})}\right) = \max_j \underbrace{\lvert x_j^{(1)} - x_j^{(2)} \rvert}$

Inter-attribute similarity measure

UNIVERSITY OF BIRMINGHAM



Chebychev distance = max(4,3)=4

Properties of distance functions

- Distance between two points is always non-negative, i.e.,
$$d\left(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}\right) \geq 0.$$
- Distance between a point to itself is zero, i.e.,
$$d\left(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}\right) = 0.$$
- Distance is symmetric i.e.,
$$d\left(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}\right) = d\left(\boldsymbol{x}^{(2)}, \boldsymbol{x}^{(1)}\right).$$
- Distance satisfies a triangle inequality, i.e.,
$$d\left(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}\right) \leq d\left(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(3)}\right) + d\left(\boldsymbol{x}^{(3)}, \boldsymbol{x}^{(2)}\right).$$

Example 2: Revisited

- Compute the Euclidean distance between Badger and Cow

  Solution: $d_{Euc}(Badger, Cow) =$
  $\sqrt{(3-0)^2+(1-0)^2+(3-3)^2+(1-3)^2+(10-400)^2} = \sqrt{9+1+0+4+390^2} =$
  390.017

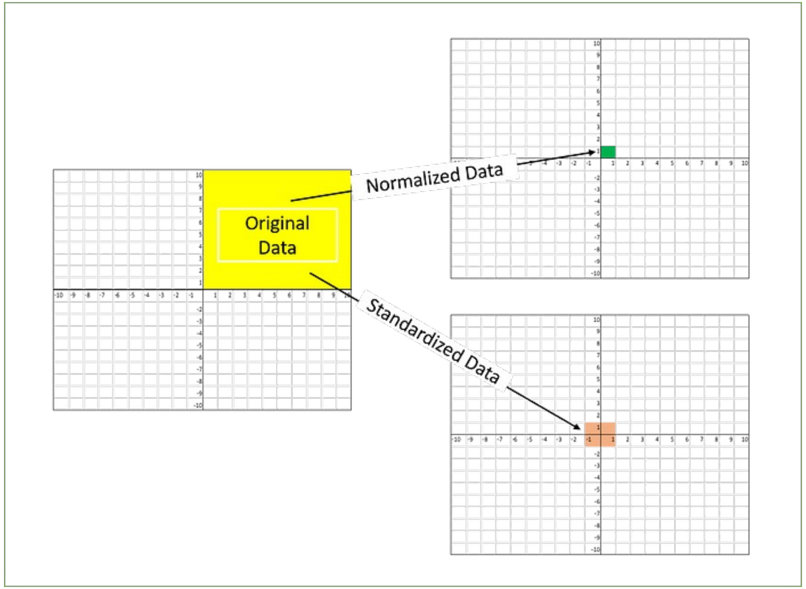
- Compute the Manhattan distance between Badger and Cow.

  Solution: $d_{Man}(Badger, Cow) = |3-0| + |1-0| + |3-3| + |1-3| + |10-400| =$
  $3+1+0+2+390 = 396$

- Takeaways:
  - Different choice of distance functions yields different measures of similarity.
  - Distance functions implicitly assign more weighting to features with large ranges than to those with small ranges.
  - Rule of thumb: when no a priori domain knowledge is available, clustering should follow the principle of equal weightings to each attribute [Mirkin, 2005]
  - This necessitates need for **normalization/data pre-processing/feature scaling** of feature vectors.

Normalization of Feature Vectors

- Normalization ensures that attributes contribute approximately equally to the similarity measure
- Two well studied approaches: **min-max normalization** and **z-score standardization**



- **Min-max normalization**: all feature attributes rescaled to **lie in the range [0,1].**

$$\begin{bmatrix} x_1^{(1)*} & x_2^{(1)*} & . & x_m^{(1)*} \\ x_1^{(2)*} & x_2^{(2)*} & . & x_m^{(2)*} \\ . & . & . & . \\ . & . & . & . \\ x_1^{(N)*} & x_2^{(N)*} & . & x_m^{(N)*} \end{bmatrix} \xrightarrow{\text{Min-max scaling}} \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & . & x_m^{(1)} \\ x_1^{(2)} & x_2^{(2)} & . & x_m^{(2)} \\ . & . & . & . \\ . & . & . & . \\ x_1^{(N)} & x_2^{(N)} & . & x_m^{(N)} \end{bmatrix}$$

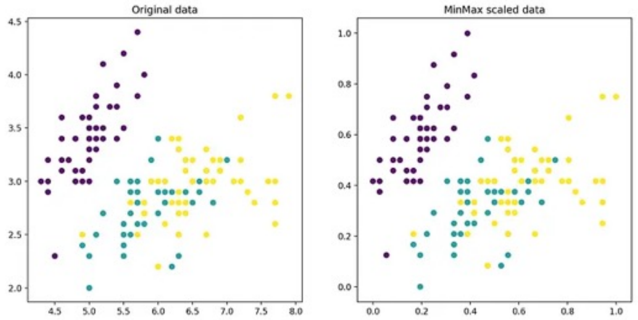
Asterisk denotes unnormalized feature entries.

Maximum of feature j: $x_{j,max} = \max_{i=1,..,N} x_j^{(i)}$

Minimum of feature j: $x_{j,min} = \min_{i=1,..,N} x_j^{(i)}$

Min-max rescaling of $x_j^{(i)*}$ results in entry:

$x_j^{(i)} = (x_j^{(i)*} - x_{j,min})/(x_{j,max} - x_{j,min})$



Drawback: sensitive to outliers

- **Z-score standardization**: all feature attributes have mean 0 and standard deviation 1.

$$\begin{bmatrix} x_1^{(1)*} & x_2^{(1)*} & . & x_m^{(1)*} \\ x_1^{(2)*} & x_2^{(2)*} & . & x_m^{(2)*} \\ . & . & . & . \\ . & . & . & . \\ x_1^{(N)*} & x_2^{(N)*} & . & x_m^{(N)*} \end{bmatrix} \xrightarrow{\text{Z-score}} \begin{bmatrix} (x_1^{(1)*} - \mu_1)/\sigma_1 & (x_2^{(1)*} - \mu_2)/\sigma_2 & . & (x_m^{(1)*} - \mu_m)/\sigma_m \\ (x_1^{(2)*} - \mu_1)/\sigma_1 & (x_2^{(2)*} - \mu_2)/\sigma_2 & . & (x_m^{(2)*} - \mu_m)/\sigma_m \\ . & . & . & . \\ . & . & . & . \\ (x_1^{(N)*} - \mu_1)/\sigma_1 & (x_2^{(N)*} - \mu_2)/\sigma_2 & . & (x_m^{(N)*} - \mu_m)/\sigma_m \end{bmatrix}$$

Mean of feature j: $\mu_j = \frac{1}{N}\sum_{i=1}^{N} x_j^{(i)*}$

Variance of feature j: $\sigma_j^2 = \frac{1}{N}\sum_{i=1}^{N} (x_j^{(i)*} - \mu_j)^2$

Drawback: not bounded range

- Choice of dissimilarity measure and normalization schemes depend on the specific problem. These are crucial factors that determine the performance of clustering algorithms.

UNIVERSITY OF BIRMINGHAM

Example 2: Z-score Standardization

Original data

Badger	**3**	**1**	**3**	**1**	**10**
Bear	3	1	4	2	278
Cow	0	0	3	3	400
Dog	3	1	4	2	20
Fox	3	1	4	2	5

$$\mu_1 = \frac{(3+3+0+3+3)}{5} = \frac{12}{5} = 2.4$$

$$\mu_2 = \frac{4}{5} = 0.8$$

$$\mu_3 = \frac{18}{5} = 3.6$$

$$\mu_4 = \frac{10}{5} = 2$$

$$\mu_5 = 142.6$$

$$\sigma_1^2 = \frac{(3-2.4)^2 + (3-2.4)^2 + (-2.4)^2 + (3-2.4)^2 + (3-2.4)^2)}{5} = 1.44$$

$\sigma_2^2 = 0.16$ $\sigma_3^2 = 0.24$ $\sigma_4^2 = 0.4$ $\sigma_5^2 = 27227$

Standardized data

Badger	0.5	0.5	-1.22	-1.58	-0.8
Bear	0.5	0.5	0.81	0	0.82
Cow	−2	-2	-1.22	1.58	1.56
Dog	0.5	0.5	0.81	0	-0.74
Fox	0.5	0.5	0.81	0	-0.83

$$\left(\frac{0.5-2}{2}, \frac{0.5-2}{2}, \frac{0.81-1.22}{2}, \frac{0+1.58}{2}, \frac{-0.74+1.56}{2}\right)$$

Distance Matrix (Proximity Matrix)

- Given: N observations $\boldsymbol{x}^{(\mathbf{1})}, \ldots, \boldsymbol{x}^{(\boldsymbol{N})}$ of feature vectors
- Distance matrix summarizes the similarity relationship among the N observations.
- Distance matrix D is a symmetric $N \times N$ matrix (matrix with N rows and N columns) whose entry in ith row and jth column is given by

$$D_{i,j} = d\left(\boldsymbol{x}^{(\boldsymbol{i})}, \boldsymbol{x}^{(\boldsymbol{j})}\right),$$

  where d is the chosen distance measure.
- Fed as input to clustering algorithms.

Example 2: Distance Matrix

Calculate the distance matrix based on Euclidean distance for the standardized data set in Example 2.

$$d_{Euc}(Badger, Bear)$$
$$= \sqrt{(0.5-0.5)^2 + (0.5-0.5)^2 + (-1.22-0.81)^2 + (-1.58)^2 + (0.82+0.8)^2} = 3.05$$

$d_{Euc}(Badger, Cow) = 5.3$, $d_{Euc}(Badger, Dog) = 2.58$, $d_{Euc}(Badger, Fox) = 2.582$,
$d_{Euc}(Bear, Cow) = 4.44$, $d_{Euc}(Bear, Dog) = 1.56$, $d_{Euc}(Bear, Fox) = 1.65$,
$d_{Euc}(Cow, Dog) = 4.95$, $d_{Euc}(cow, fox) = 4.99$, $d_{Euc}(Dog, Fox) = 0.09$

	Badger	Bear	Cow	Dog	Fox
Badger	0	3.05	5.3	2.58	2.582
Bear		0	4.44	1.56	1.65
Cow			0	4.95	4.99
Dog				0	0.09
Fox					0

Clustering Algorithms

UNIVERSITY OF BIRMINGHAM

Types of Clustering Algorithms

Clustering Algorithms

Partitional

- Generates a single partition of the data to recover natural clusters
- Input: Feature vectors
- Examples: K-means, K-medoids

Hierarchical

- Generates a sequence of nested partitions
- Input: Distance Matrix
- Example: agglomerative clustering, divisive clustering

Model-Based

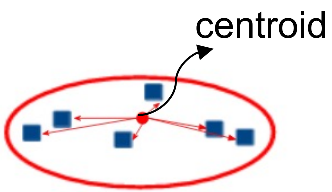
- Assumes that data is generated i.i.d. from a mixture of distributions, each of which determines a different cluster
- Example: Expectation-Maximization (EM)

Partitional Clustering

- Goal: assign N observations into K (K<N) clusters to ensure high intra-cluster similarity and low inter-cluster similarity
- Can be formulated as a combinatorial optimization problem.
- Notation:
  - $\boldsymbol{C}$ denotes a clustering structure with K clusters
  - $\mathrm{C} \in \boldsymbol{C}$ denotes a component cluster,
  - $e \in C$ denotes an example in cluster

Measure of intra-cluster similarity

Variability (or Inertia) of a cluster C:



$$variability(C) = \sum_{e \in C} d\big(e, centroid(C)\big).$$

- Commonly used distance measure: squared Euclidean distance, i.e., $d(\boldsymbol{a}, \boldsymbol{b}) = d_{Euc}(\boldsymbol{a}, \boldsymbol{b})^2$.
- Centroid of a cluster is usually taken as the average of all examples in the cluster i.e.,

$$centroid(C) = \frac{attribute\text{−}wise\ sum\ of\ examples\ in\ the\ cluster}{number\ of\ examples\ in\ the\ cluster}$$

- Variability determines how compact the cluster is.

UNIVERSITY OF BIRMINGHAM

- Dissimilarity within a clustering structure $\boldsymbol{C}$:

$$dissimilarity(\boldsymbol{C}) = \sum_{C \in \boldsymbol{C}} variability(C)$$

- Optimization problem: Find a clustering structure $\boldsymbol{C}$ of K clusters that minimizes the following objective:

$$\min_{\boldsymbol{C}} dissimilarity\ (\boldsymbol{C})$$

  - Larger clusters with high variability are penalized more than smaller clusters with high variability.
  - Under squared Euclidean distance, minimizing $dissimilarity(\boldsymbol{C})$ is equivalent to maximizing overall inter-cluster dissimilarity (will see this in detail later).

- Finding exact solution of the above problem is prohibitively hard.
    - Infeasible when large number of examples present
- Solution: Iterative Greedy Algorithms
    - Provide a sub-optimal approximate solution
    - Includes K-means, K-medoids

Example 2: Revisiting

Assume that clustering returns two clusters: C1: (Dog,Cow) and C2: (Badger, Bear, Fox). Use standardized data.

- Calculate the cluster centroids.
  - Centroid of cluster 1: $\left(\frac{0.5-2}{2}, \frac{0.5-2}{2}, \frac{0.81-1.22}{2}, \frac{0+1.58}{2}, \frac{-0.74+1.56}{2}\right)$

References

- Introduction to Computation and Programming Using Python with Application to Computational Modeling and Understanding Data third edition by John V. Guttag - Chapter 25
- Algorithms for clustering data – Jane and Dubes, Chapter 3
- On normalization, https://royalsocietypublishing.org/doi/epdf/10.1098/rspa.2011.0704