Hands on Session | Clustering

Tamojit Maiti
Masters in Applied Statistics and Operations Research, ISI Kolkata
Data Scientist at Sixt R&D, previously at Rapido & AB InBev

Agenda

- Clustering
  - What?
  - Why?
- Clustering Algorithms Recap
  - K-Means
  - K-Modes
  - Hierarchical
- Playground
  - K-Means on the wine dataset
  - K-Modes on the bank transaction dataset
  - Hierarchical Clustering on the wine dataset
- Advanced Topics
  - Market Basket Analysis on a toy dataset
  - DBSCAN on the wine dataset
  - GMM for clustering on the iris dataset

Clustering

Clustering | What

- One of the most common unsupervised learning techniques
- Groups entities such that
  - similar entities belong to the same cluster
  - dissimilar entities belong to different clusters
- Similarity or dissimilarity between entities is measured by some notion of distance

Clustering | What -> Distance Measures

- Minkowski Distance $$D_p(x, y) = \left\{ \sum_i |x_i - y_i|^p \right\}^{\frac{1}{p}}$$
  - Euclidean Distance $p = 2$
  - Manhattan Distance $p = 1$
- As you increase p, you increase the influence that extreme points have on your distance measure, and by consequence, on your clustering algorithm

Clustering | What -> Distance Measures

- Hamming Distance

P1	1	0	1	1
P2	1	1	1	0

$$D_H(P1, P2) = 2$$

- This is used when you want to find the distance between entities described by categorical variables

- Cosine Distance

P1	1	2	5	2
P2	3	4	2	0

$$|P1| = \sqrt{34}$$
$$|P2| = \sqrt{29}$$
$$\langle P1, P2 \rangle = 21$$
$$D_C(P1, P2) = \frac{21}{\sqrt{34 \cdot 29}}$$

- This is used when you do not want the magnitude of your entities to affect your distance calculations

Clustering | Why

- Retail
  - Customers can be segmented based on purchase patterns and transaction history and be targeted with custom offers that result in better response and higher revenue

- Ride Hailing
  - Customers are segmented based on ride history, behavior attributes and are given special exclusive offers and discounts that yield higher conversions and improve the customer experience

Clustering Algorithms Recap

Clustering Algorithm Recap | K-Means

- Given a set of N points $\{x_i\}$ and the number of clusters k, it finds the location of the cluster centroids μ_k
- Each point $\{x_i\}$ would belong to that cluster whose cluster centroid it is closest to
- Each point can only belong to one cluster
- The final cluster centroids after convergence strongly depend on the initial centroid initialization

Clustering Algorithm Recap | K-Means -> Pseudo-code

- Initialization of cluster centroids is performed, strategically
- Repeat until convergence criteria is fulfilled
  - Assigning Each point $\{x_i\}$ is assigned to the cluster whose centroid it is closest to
  - Updating Estimation of cluster centroids is done by calculating the mean of all points $\{x_i\}$ belonging to that cluster
- Convergence criteria is usually either
  - Number of iterations
  - No appreciable change in cluster centroid locations

Clustering Algorithm Recap | K-Means -> Cost Function

- The cost function of the algorithm is

$$WCSS = \sum_{k \in C} \sum_{x_j \in k} \left| x_j - \mu_k \right|_2^2$$

- This means that
  - For each cluster
    - For each point in that cluster
      - Calculate the distance of the point to its cluster center
    - Sum all the distances of the points to its cluster center
  - Sum all such distances for all clusters

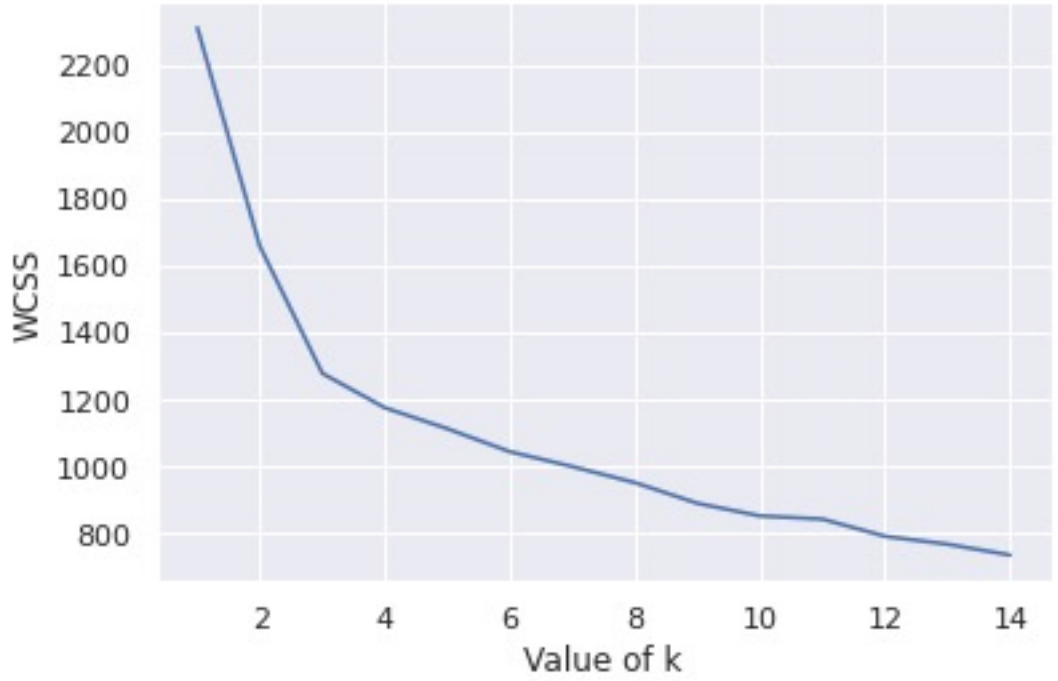
Clustering Algorithm Recap | K-Means -> Caveats

- The cost function involves the calculation of Euclidean distance, hence extreme points have an appreciable influence on clustering outcome
- Only used when all features are continuous valued, and not discrete or categorical
- The points to be clustered need to be scaled, else one feature will dominate the clustering result
- Choosing the number of clusters is of utmost importance
  - Needs to be statistically sound
  - Needs to align with business needs and actionability

Clustering Algorithm Recap | K-Means -> Choosing k

- Two methods for choosing k
  - Steepest decline of WCSS or Elbow Method

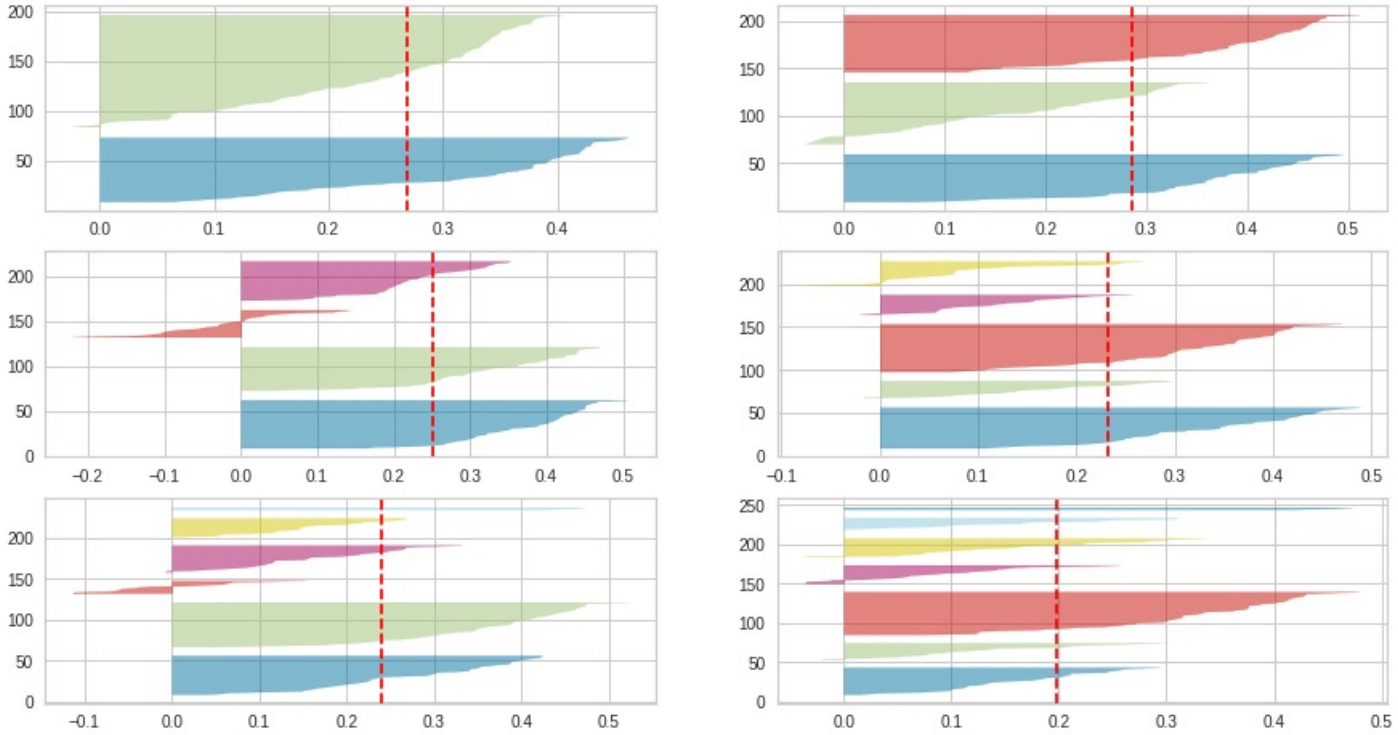
Choose that value of k from which the WCSS vs k graph begins to flatten



Clustering Algorithm Recap | K-Means -> Choosing k

- Two methods for choosing k
  - Silhouette Score

Choose that k which yields a higher silhouette score for all clusters



Clustering Algorithm Recap | K-Means -> K-Means++

- The final clustering result is strongly dependent on the cluster centroid initialization
- Random initialization often leads to sub-optimal clusters
- An improvement is to initialize clusters such that each subsequent cluster centroid is the farthest from all other clusters
- This improves rate of convergence and leads to a steeper decrease in WCSS with k

Clustering Algorithm Recap | K-Means -> Disadvantages

- Assumes that cluster shapes are spherical, real world data points rarely form spherical clusters
- Only continuous features can be used, fails for categorical features
- May not converge for distances other than Euclidean distance
- Each point can belong to only one cluster, no soft clustering possible, but real-world entities can lie in equal proximity to more than one cluster

Clustering Algorithm Recap | K-Modes

- Applied when you have categorical features in data
- The distance measure is some version of the total number of mismatches between data points, e.g. Hamming Distance
- The pseudo-code is exactly like K-Means, except mode is used to assign cluster centroids instead of mean
- Clustering outcome is not that sensitive to initialization strategy since categorical variables have much lesser number of unique values, hence less chances of yielding sub-optimal results

Clustering Algorithm Recap | K-Modes | Pseudo-Code

- The cost function is

$$WCSS_H = \sum_{k \in C} \sum_{x_j \in k} H(x_i, \mu_k)$$

The term $H(x_i, \mu_k)$ calculates the Hamming distance between x_i and μ_k

- This means that
  - For each cluster
    - For each point in that cluster
      - Calculate the H - distance of the point to its cluster center
    - Sum all the distances of the points to its cluster center
  - Sum all such distances for all clusters

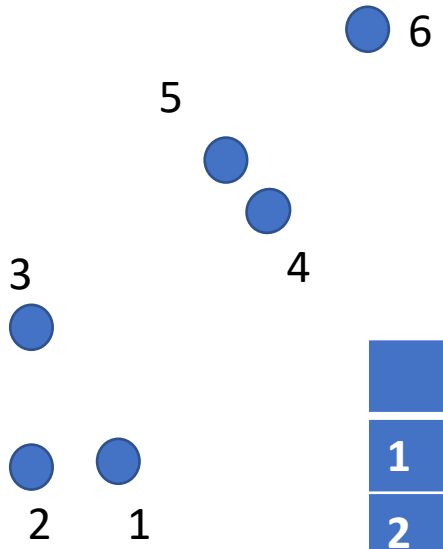
Clustering Algorithm Recap | Hierarchical Clustering

- Clustering without pre-specified k, the number of clusters
- The distance measure between points needs to be specified
- Can be either top down (divisive) or bottom up (agglomerative)
- Allows flexibility in choosing number of clusters
- The output is a dendogram, highlighting the extent of proximity between individual points in a hierarchical manner
- Each level in a dendogram represents a clustering output

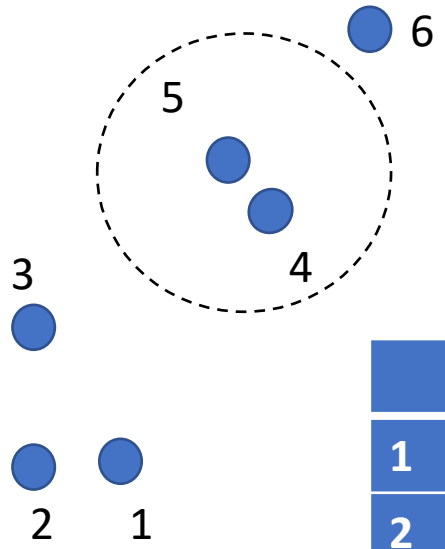
Recap | Hierarchical Clustering | Pseudo-Code

- For all the N points {x_i}, we consider them as individual clusters
- We calculate a N \times N distance matrix to compute distance between the points
- The points with the least distance between them become the first cluster
- The N-1 \times N-1 distance matrix is recomputed for the remaining points
- The points/clusters with the subsequent distance between them become the next cluster
- The above two steps are repeated until all points are used up

Recap | Hierarchical Clustering | Walkthrough

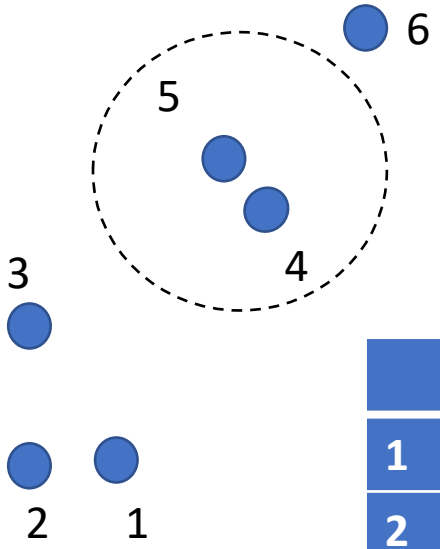


	1	2	3	4	5	6
1	0					
2	4	0				
3	6	5	0			
4	12	14	10	0		
5	13	16	10	2	0	
6	21	23	18	8	7	0

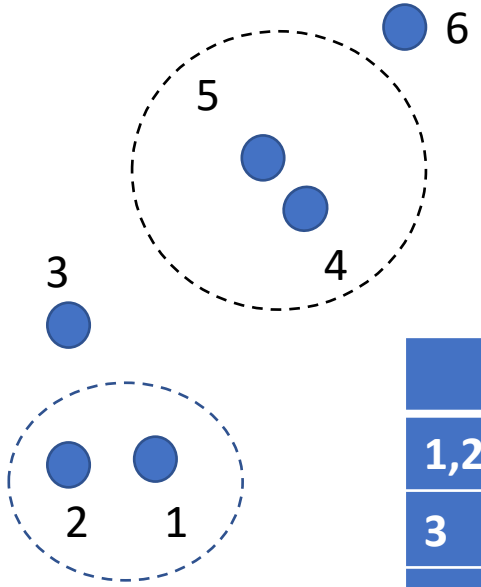


	1	2	3	4,5	6
1	0				
2	4	0			
3	6	5	0		
4,5	12	14	10	0	
6	21	23	18	8	0

Recap | Hierarchical Clustering | Walkthrough

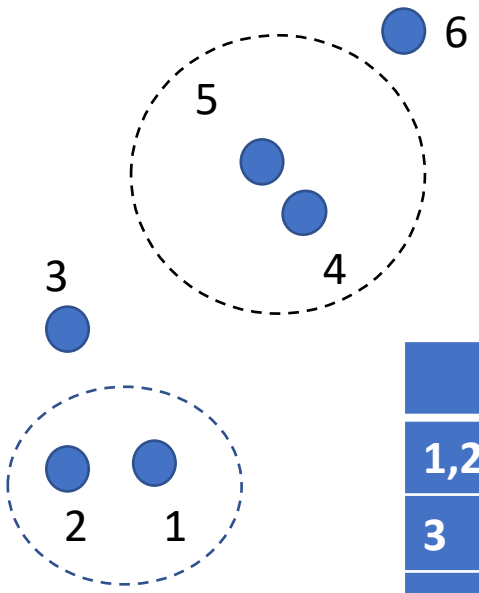


	1	2	3	4	6
1	0				
2	4	0			
3	6	5	0		
4,5	12	14	10	0	
6	21	23	18	8	0

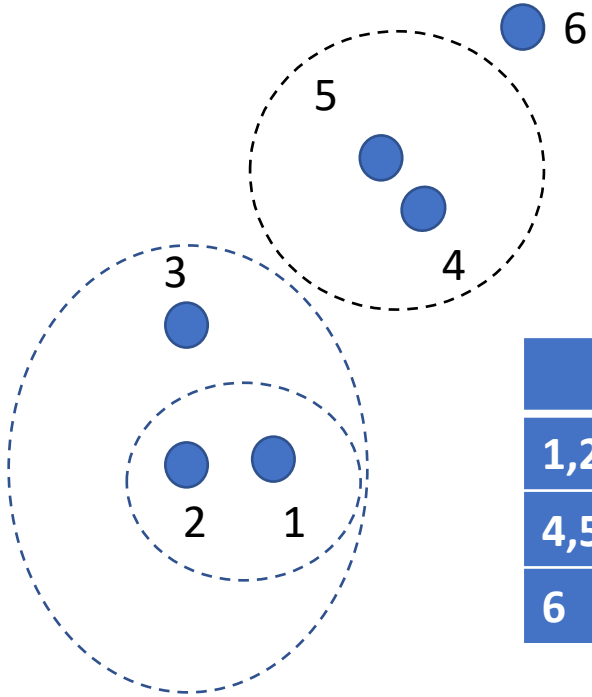


	1,2	3	4	6
1,2	0			
3	5	0		
4,5	12	10	0	
6	21	18	8	0

Recap | Hierarchical Clustering | Walkthrough

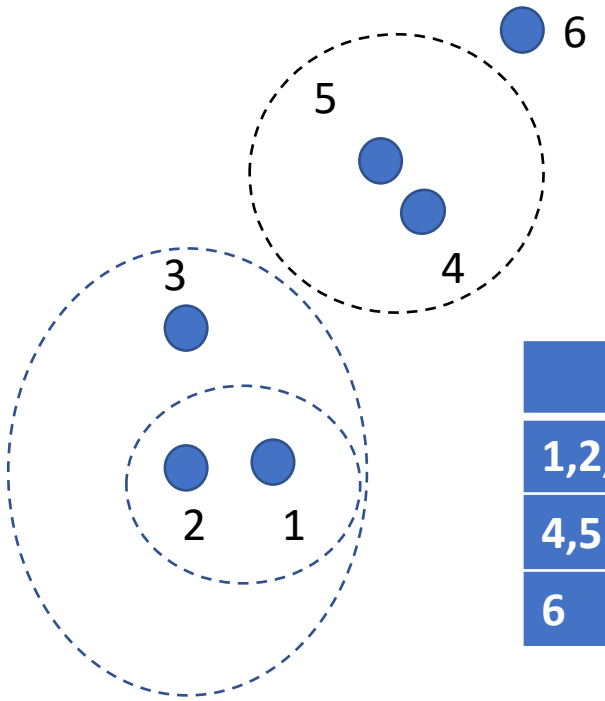


	1,2	3	4,5	6
1,2	0			
3	5	0		
4,5	12	10	0	
6	21	18	8	0

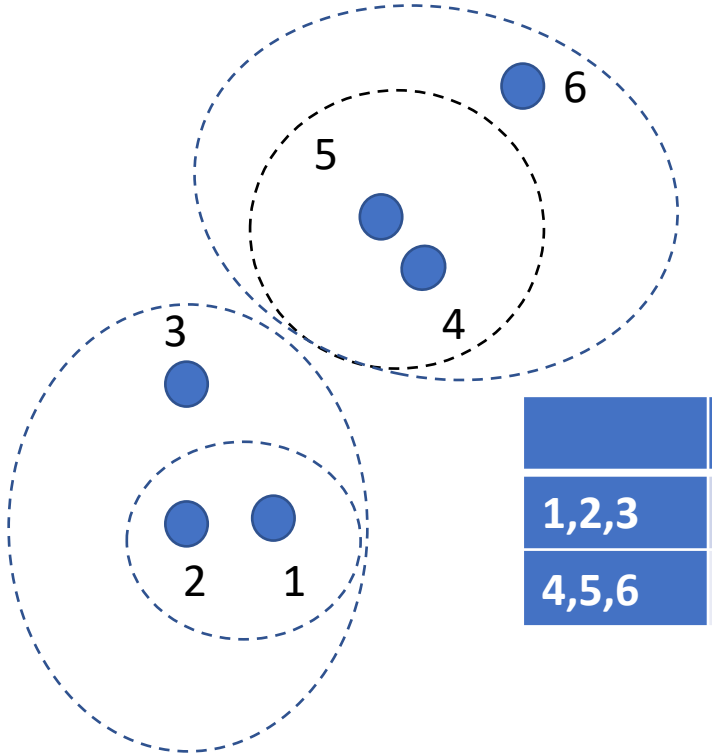


	1,2,3	4,5	6
1,2,3	0		
4,5	12	0	
6	21	8	0

Recap | Hierarchical Clustering | Walkthrough

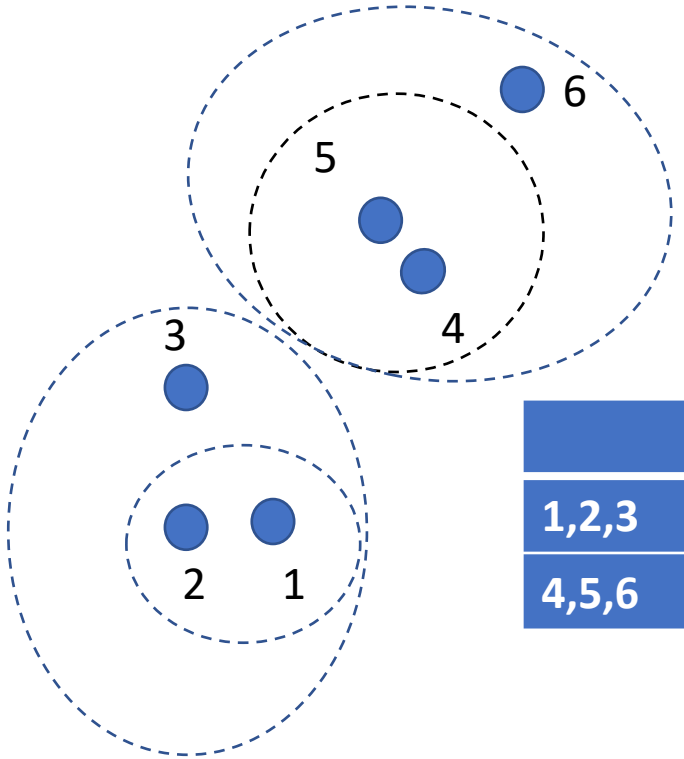


	1,2,3	4,5	6
1,2,3	0		
4,5	12	0	
6	21	8	0

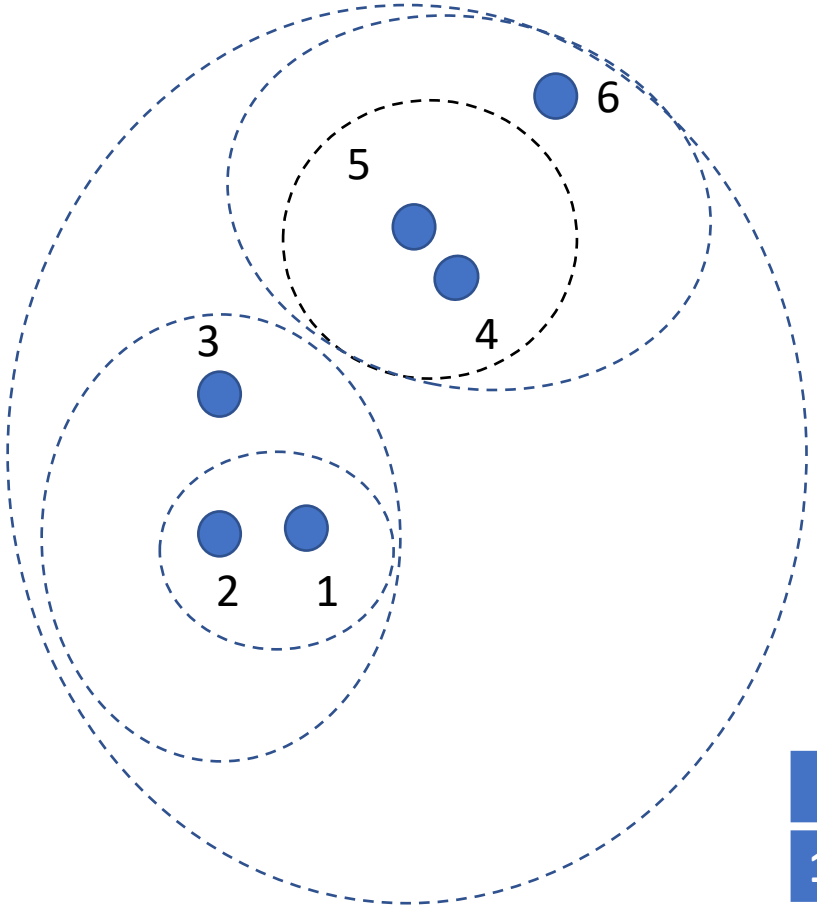


	1,2,3	4,5,6
1,2,3	0	
4,5,6	12	0

Recap | Hierarchical Clustering | Walkthrough



	1,2,3	4,5,6
1,2,3	0	
4,5,6	12	0



	1,2,3,4,5,6
1,2,3,4,5,6	0

Playground

- Colab notebook for KMeans
- Colab notebook for KModes
- Colab notebook for Hierarchical Clustering

Advanced Topics

Advanced Topics | Market Basket Analysis

- Typically used to find items that are bought together in a supermarket or online store
- Identifying items bought together will help retailers bundle offers, give discounts and recommend products to customers
- All these lead to improvement in customer experience and revenue

Advanced Topics | Market Basket Analysis | Terms

- Suppose there were 4000 transactions, of which 400 contain biscuits, and 600 contain chocolate, and these 600 transactions include 200 transactions that include biscuits and chocolates.

- Support is the default popularity of any product

Support of biscuit is 400/4000 = 10%

- Confidence refers to the conditional probability of buying a product given that they have bought another product

Confidence of chocolate given biscuit = 200/400 = 50%

- Lift is how much more likely is a product likely to be sold if it's coupled with another product vs standalone

Lift of chocolates given biscuit = 50/10 = 5

Advanced Topics | Market Basket Analysis | Apriori

- Given a Boolean transaction dataset, it outputs the most frequent items bought together
- The most frequent itemsets are characterized by their support, lift and confidence
- Itemsets with strong confidence and lift form stronger association rules
- Association rules are used by business for product recommendation or planning offer bundles
- Computationally intensive if # products is large

Advanced Topics | Gaussian Mixture Model

- Overcomes the constraint of spherical cluster shapes imposed by K-Means clustering, clusters modelled via GMM can have elliptical shapes
- Number of clusters need to be provided
- GMM does soft clustering hence points can belong to more than one cluster with certain probabilities, more accurate reflection of real-life phenomenon
- Can be used for clustering and outlier detection at the same time

Advanced Topics | Gaussian Mixture Model

- Points are assumed to be the superposition of more than one multivariate Gaussian distribution

$$X_i = \sum_j N(\mu_j, \Sigma_j)$$

- Each cluster has its own mean and variance, that is estimated for GMM via the EM algorithm

- Link: Colab Notebook

Advanced Topics | DBSCAN

- A Euclidean distance-based clustering technique that can perform simultaneous outlier detection
- Points are assigned to clusters if they are nearer to denser regions, while points in sparser regions are more likely to be classified as outliers
- Number of clusters does not need to be provided, however two other parameters, Epsilon and MinPts need to be specified
- Link: Colab Notebook for DBSCAN

Q & A