Date :

Unsupervised Learning

* Unsupervised learning is the training of a machine using information that is neither classified nor labelled & allowing algorithm to act on the information without guidance

* Unsupervised algorithm finds out patterns, similarities & differences ~~that~~ in the dataset. It is just shown the i/p data & asked to extract knowledge from the data.

Why unsupervised learning?

(1) It is easier to get unlabelled data from a computer than labelled data which needs manual intervention.

(2) It ~~takes place in real time~~, so all the ~~input~~ data ~~to be~~ Unsupervised ML ~~methods~~ finds all kinds of unknown patterns in data

(3) Unsupervised ML methods find features which can be useful for categorization.

Date :

Types of unsupervised learning

(1) Clustering:- It is the process of grouping a set of objects or data points in such a way that objects in the same group are more similar to each other than to those in other groups.

Common clustering algorithms.

(1) K-means clustering:- Divides a set of samples in to disjoint clusters, each described by mean of samples in the cluster.

(2) Hierarchical clustering:- Builds a multilevel hierarchy of clusters by creating a cluster tree.

(3) DBSCAN (Density-based spatial clustering of applications with noise):- Defines clusters as areas of high density allowing it to find arbitrarily shaped clusters & to be more robust to outliers than K-means.

(4) Gaussian mixture models:- Points are probabilistically assigned to clusters (soft clustering)

Date :

② Association:- Find interesting relations bet? variables in large databases.
Used in market basket analysis
Algorithms:- Apriori algorithm
FP-Growth

③ Dimensionality Reduction:- Reduce no. of features in a dataset while preserving as much relevant information as possible.
Algorithms:- Principal Component Analysis (PCA)
t-Distributed Stochastic Neighbor Embedding (t-SNE)
Autoencoders.

④ Anomaly detection:- Identification of rare items, events or observations which are significantly different or deviate from the majority of data.
Algorithms:- Isolation forest
One class SVM

⑤ Neural n/ws:- Neural n/w models are used in an unsupervised manner to learn features from unlabeled data.

Date :

Algorithms:- ① Self-organizing maps
② Generative adversarial n/ws.

Clustering:- Clustering is the task of dividing data points into a no. of groups such that (on the basis of chosen or attrib) data points in same group are similar to other data points in same group & dissimilar to data points in other group.

* The distance bet? points in a cluster is less than the " " a point in the cluster & any pt. outside it.

* Database segmentation:- It is the term closely aligned with clustering

* Clustering doesn't require labeled data & is used to discover patterns hidden in the data without prior knowledge of data's structure.

Note:- Clustering is somewhere similar to classification algorithm but difference is the type of dataset that we are using. (Labelled-Classification & unlabelled-clustering).

Date :

Examples of clustering

1. Customer segmentation in Marketing :- Based on purchasing behavior, demographics or other attributes. This will be used to plan marketing strategies & product offerings to meet needs & preferences of each group.

2. Image segmentation in medical imaging :- Based on similarities in pixel intensity, color or texture. Useful in tumor detection & tissue classification in medical diagnostics.

3. Anomaly detection in n/w security :- N/w intrusions or Malicious activities.

4. Recommendation systems in e-commerce

Importance of clustering

1. Pattern discovery
2. Data exploration
3. Segmentation
4. Anomaly detection
5. Feature engineering
6. Data preprocessing
7. Decision-making

Date :

Appl?s of clustering

1. Customer segmentation
2. Image segmentation
3. Anomaly detection → Fraud detection, n/w security & quality control
4. Document clustering
5. Recommendation systems
6. Genomics & Bioinformatics
7. Market research.
8. Healthcare analytics.

Clustering attributes :- Features of data that are used to group similar data points together.

Types of clustering methods.

1. Hard clustering :- Data points belong to only one group.
2. Soft clustering :- Data points can belong to another group also.

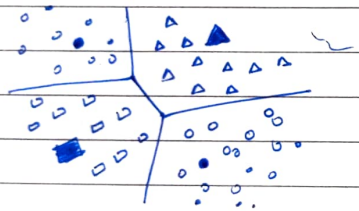
Other approaches.

1. Partitioning clustering
2. Density based clustering
3. Distribution model-based clustering
4. Hierarchical clustering
5. Fuzzy clustering

Date :

① Partitioning clustering :-

* Divides dataset into non-overlapping clusters.
* It partitions data in to set of clusters, where each cluster contains data points that are similar to each other & dissimilar to data points in other clusters.
* User determines no. of clusters & algorithm finds best partition of data into specified no. of clusters.
* Ex: K-means clustering algorithm



*. Cluster center is created in such a way that distance bet? data points of a cluster is minimum as compared to another cluster centroid.

Date :

Types of partitioning clustering algorithms :-

① Minimum spanning tree :- Tree-like structure that connects all data points in a dataset while minimizing total distance bet? them. The tree is then cut at certain level to form clusters.

② Squared error clustering :- Aim is to minimize sum of squared distances bet? data points & their assigned cluster centres. It starts by randomly selecting initial cluster centers & then iteratively updates them until convergence.

③ K-means clustering :- It is similar to squared error clustering but uses a different distance metric. Simple & efficient.

④ Nearest neighbor algorithm :- This algorithm assigns each data point to the nearest cluster center based on distance metric. It also starts by randomly selecting initial cluster centers & then assigns each data point to nearest center. Process is repeated until convergence.

Ex:- Similar purchasing behavior customers are grouped in to segments.

Date

(2) Density-based clustering:

In density-based clustering algorithms like DBSCAN, clusters are formed by grouping together data points that belong to high-density regions while separating them from low-density regions.

Algorithm identifies:
1. Core points (high-density pts)
2. Border " (edge of high density regions)
3. Noise " (isolated pt.s in low-density areas)

Ex:- Clusters of customers → Frequently buy in certain areas of a store (high density)
→ Shop less frequently or in different area (low-density)

Algorithm: DBSCAN: Identifies core points, border points & noise points in a dataset

Date

(3) Distribution model-based clustering:

Data is divided based on the probability of how a dataset belongs to a particular distribution. It use gaussian distribution to calculate probability distribution.

Algorithm: Expectation-Maximization (EM)
Gaussian mixture models (GMM)

Ex:- Amount spent on groceries, electronics & clothing by each customer. By running EM algorithm with GMM on this data, algorithm can partition customers into segments based on their spending preferences.

(4) Hierarchical clustering:

Data is divided into clusters to create a tree-like structure which is also called a dendrogram. The algorithm starts by considering each data point as a separate cluster & then iteratively merges closest pairs of clusters based on their similarity.

This process continues until all data points belong to a single cluster at top level of hierarchy.

Ex:- Library

Date :

Types: ① Agglomerative ② Divisive

③ Fuzzy clustering: In fuzzy clustering, data objects are allowed to belong to multiple clusters simultaneously with each object having membership coefficients indicating the degree to which it belongs to each cluster.

Ex: Fuzzy k-means algorithm.

Similarity & distance measures.

① Similarity measures:

(i) If 2 data points are identical, their similarity should be 1.

(ii) If 2 data points are completely dissimilar, their similarity should be 0.

(iii) If 2 data points are somewhat similar, their similarity should be bet^n 0 & 1.

Date

② Distance measures:

(i) If 2 points are identical, their distance should be 0.

(ii) If 2 data points are completely dissimilar, their distance should be high.

(iii) If 2 data points are somewhat dissimilar, their distance should be bet^n 0 & high.

Date

K-Means Clustering algorithm

It is an iterative algorithm that divides the unlabeled dataset into k different clusters in such a way that each dataset belongs only one group that has similar properties.

- The value of k is predetermined in the algorithm.
- The algorithm takes the unlabeled dataset as i/p, divides the dataset into k-no. of clusters until it does not find the best clusters.

How K-means clustering works?

Step 1: Initialization:

1. Choose no. of clusters, k.
2. Initialize k cluster centroids (means) randomly or based on some heuristic

Step 2: Assign data points to clusters

For each data point in the dataset:

- Calculate the distance bet? each data point & each cluster centroid.
- Assign each data point to the cluster with the closest centroid (min distance)

Date :

Step 3 : Update cluster centroids :-

After assigning all data points to clusters, calculate new centroid for each cluster by taking the mean of all data points assigned to that cluster.

Step 4 : Convergence check :-

Repeat the assignment & centroid update steps until a stopping criterion is met, such as

1. No data point changes its cluster assignment.
2. The centroids do not change significantly bet^n iterations.
3. Max no. of iterations is reached

Step 5 : Algorithm Iteration :-

Iterate bet^n assigning data points to clusters & updating cluster centroids until convergence.

Step 6 : Final o/p

Once the algorithm converges.

↳ The final o/p is a set of k clusters where each cluster contains data pts that are more similar to each other than to data points in other clusters.

↳ The centroids represent mean of data pts within each cluster.

Date :

Example.

Suppose we have 4 data points :-

A(2, 3), B(3, 3), C(8, 6) & D(9, 5)

And we want to create 2 clusters ($k = 2$)

Step 1 : Initialization

- Randomly choose 2 intial clusters centroids

cluster 1 Centroids (C1) : (2, 3)

cluster 2 Centroids (C2) : (8, 6)

Step 2 : Assignment :

Calculate the distance from each data point to each cluster centroid & assign each point to nearest centroid using Euclidean distance

For data point A :

Distance from (2, 3) to (2, 3) = 0.

" " (2, 3) to (8, 6) = $\sqrt{40}$.

So A is assigned to cluster 1 (C1)

Date :

For data point B:

Distance from B(3,3) to C(2,3) = 1

" " B(3,3) to C(8,6) = $\sqrt{34}$.

So, B is assigned to cluster 1 (C1).

For data point C:

C(8,6) to C(2,3) = $\sqrt{58}$

C(8,6) to C2(8,6) = 0.

So, C is assigned to cluster 2.

For data point D:

D(9,5) to C1(2,3) = $\sqrt{58}$

D(9,5) to C2(8,6) = 1.

So, D is assigned to cluster 2.

Step 3: Update

↳ Recalculate the centroids for each cluster based on points assigned to them.

For cluster 1:- New C1 = Average of A & B

= $((2+3)/2, (3+3)/2) = (2.5, 3)$

For cluster 2 :- New C2 = Average of C & D

= $((8+9)/2, (6+5)/2) = (8.5, 5.5)$

Step 4 :- Convergence check:

Check whether cluster centroids have

Date :

stopped moving significantly. This is done by comparing old centroids with new centroids. If they are almost the same (within a same threshold), the algorithm converges. Otherwise return to step 2 for reassignment.

Step 5:- Output clusters:-

Cluster 1 → A & B

cluster 2 → C & D

Date :

For data point B:

Distance from B(3,3) to C(2,3) = 1

" " B(3,3) to C(8,6) = $\sqrt{34}$.

So, B is assigned to cluster 1 (C1).

For data point C:

C(8,6) to C1(2,3) = $\sqrt{58}$

C(8,6) to C2(8,6) = 0.

So, C is assigned to cluster 2.

for data point D:

D(9,5) to C1(2,3) = $\sqrt{58}$

D(9,5) to C2(8,6) = 1.

So, D is assigned to cluster 2.

Step 3: Update

↳ Recalculate the centroids for each cluster based on points assigned to them.

For cluster 1 → New C1 = Average of A & B

= $((2+3)/2, (3+3)/2) = (2.5, 3)$

For cluster 2 → New C2 = Average of C & D

= $((8+9)/2, (6+5)/2) = (8.5, 5.5)$

Step 4 → Convergence check:

Check whether cluster centroids have

Date :

stopped moving significantly. This is done by comparing old centroids with new centroids. If they are almost the same (within a same threshold), the algorithm converges. Otherwise return to step 2 for reassignment.

Step 5: Output clusters:

Cluster 1 → A & B

cluster 2 → C & D.

Using clustering for Image segmentation:

What is Image segmentation?

* Here the image is partitioned into meaningful parts that correspond to objects or areas of interest within image.

* Why segmentation → Extract import information, identify objects, boundaries (lines, curves, etc..) & textures & enable further analysis & processing tasks.

*. The result of image segmentation is a set of segments that collectively cover the entire image, or a set of boundaries extracted from the image.

* Each of the pixels in a region is similar with respect to color, intensity or texture.

Use Case:- (1) Tumor detection in MRI Images
(2) Autonomous Vehicles.

Date

How image segmentation works?

Image is divided into segments, each represented by labeled images.

2 approaches:-

(1) To detect abrupt changes in pixel values, which often correspond to edges bet? regions within the image. These edges serve as boundaries bet? areas with varying characteristics.
(2) Identifying similarities among regions in an image. Clustering & thresholding are utilized to group pixels with similar attributes.

K-Means clustering for image segmentation

Similar pixels on basis of color intensity, texture.

K-means clustering for Image segmentation
(1) Initialization :- K.
(2) Assignment step :- Each pixel in the image is assigned to the cluster whose centroid is closest to it in terms of feature similarity. Distance metric is used to measure similarity.

Date :

(3) Update step :- After assigning all pixels to clusters, the centroids of the clusters are recalculated based on the mean of the pixel values within each cluster.

(4) Iteration :- Step (2) & (3) are repeated iteratively until convergence criteria are met such as maximum no. of iterations or minimal change in cluster assignments.

(5) Segmentation :- It is effectively segmented the image into distinct regions based on pixel similarity.

Using clustering for preprocessing :-

Clustering as a preprocessing step in a ML pipeline can enhance the performance of downstream algorithms.

(1) Outlier detection & handling :- Grouping data points in to clusters based on their similarity.
- ↳ Removed from dataset
- or
- ↳ assigning them to a specific cluster

Ex :- Fraudulent transactions can be assigned to separate cluster for further investigation or flagged for review

Date :

(2) Handling missing values :- Based on the values of other points in the same cluster.
Ex :- Income values missing can be estimated by clustering customers based on similar age, education level & occupation & inferring missing income values from cluster's income distribution.

(3) Dimension reduction :- Representing similar features by cluster centroids as new feature.

(4) Feature selection :- By identifying clusters of features that are highly correlated or redundant. Features within the same cluster may provide similar information & selecting representative features from each cluster can reduce redundancy.
Ex. In a dataset with numerous customer behavior features, clustering can group together features that provide similar information (eg :- purchase freq & total spend), allowing for the selection of representative features from each cluster for modeling.

Date :

(5) Data transformation:- Data transformation through clustering can help in creating new features, encoding categorical variables or normalizing data for better model performance.

Ex:- Dataset of product reviews → grouping similar reviews based on sentiment/topics → helps in sentiment analysis tasks.

(6) Anomaly detection:- Detecting potential cyber threats or anomalies in n/w activity

Date :

DBSCAN:- (Density based spatial clustering of applications with noise)

It is a clustering algorithm that groups together points that are closely packed based on density criterion.

It is particularly useful for identifying clusters of varying shapes & sizes in a dataset while also being robust to noise & outliers.

How DBSCAN works?

(1)

(1) Core points:- They are central hubs in a cluster. A point is considered a core point if it has minimum number of neighboring points within a specified distance

(2) Border points:- They are on the outskirts of a cluster. They are reachable from core points but do not have enough neighbors to be core points themselves.

(3) Noise points:- They are outliers that do not belong to any cluster.

Date : _____

(2) Parameters

Epsilon (eps) :- It is defined as the radius of each data point around which the density is considered. This defines maximum distance bet? 2 points for them to be considered neighbors.

Minimum Samples (min-samples) :-

It is the no. of points required within the radius so that data points becomes a core point.

(3) Algorithm Steps :-

(1) Initialization :- The algorithm begins by randomly selecting a point from the dataset that has not been visited. This initial pt. serves as the starting pt.. to form a cluster.

(2) Expand :-

(i) For each core point or border point, reachable from a core point the algorithm expands the cluster by adding neighboring points recursively.

(ii) It checks the neighboring points of the current point to determine if they should be included in the cluster.

Date :

(iii) If a neighboring point meets the criteria to be a core point or a border point, it is added to the cluster.

(iv) This process continues iteratively, expanding the cluster by adding points that are within the specified distance (epsilon) & have minimum no. of neighbors (min_samples)

(3) Termination:- The algorithm stops when all points have been visited.

(4) Output:

Clusters:- Points that belong to the same cluster based on density.

Noise:- Outliers or points that do not fit into any cluster.

Date :

Example Working of DBSCAN algorithm:-

Dataset of points representing customers in a shopping mall based on their spending score & annual income. We want to group these customers into clusters using DBSCAN.

(1) (i) Core points:- A core point could be a customer who has atleast 5 other customers within a distance of 10 units.

(ii) Border points:- Border points are customers who are reachable from core points but do not have enough neighbors to be core points themselves.

(iii) Noise points:- Noise points are customers who do not belong to any cluster because they are outliers in terms of spending score & income.

(2) Parameters:-

(i) Epsilon (eps):- Let's set epsilon to 10 unit, meaning points within a distance of 10 units are considered neighbors.

Date : _____

(ii) Minimum samples (min_samples):- We require atleast 5 points within the epsilon radius for a point to be considered a core point.

(3) Algorithm Steps:-

(i) Initialization

(ii) Expand:- Based on epsilon & min_samples criteria.
Check if neighboring customers meet the criteria to be core points or border points & add them to the cluster.

(iii) Termination after all customers have been visited & clustered.

(4) o/p.