Clustering:

1. What is an unsupervised learning approach? Why is it needed?

To find structures and data insights, patterns in the data where labels are not there.To find associations and structures in data that are hard to find manually but that can be relevant and useful to find patterns and predict trends.

​

1. What is clustering?

it’s the task of partitioning the dataset into groups, called clusters. the goal is to split up the data ins such a way that points within a single cluster are very similar and points in a different cluster are different

​

1. How do clustering and classification differ?

Classification is supervised learning method predicts the unseen un labelled data based on the labelled data

​

1. What are the various applications of clustering?

​ Clustering methods are usually used in biology, medicine, social sciences, archaeology, marketing, characters recognition, management systems and so on.

* Recommendation engines
* Market segmentation
* Social network analysis
* Search result grouping
* Medical imaging
* Image segmentation
* Anomaly detection

1. How does clustering play a role in supervised learning?

​ **Clustering is** considered unsupervised **learning**, because there's no labeled target variable in **clustering**. **Clustering** algorithms try to, well, **cluster** data points into similar groups (or… ... However, **Clustering is** a valuable input into **supervised** models because **clusters** may have a preponderance of a particular class.

1. What are the requirements to be met by a clustering algorithm?

​ The main requirements that a clustering algorithm should satisfy are:

* scalability;
* dealing with different types of attributes;
* discovering clusters with arbitrary shape;
* minimal requirements for domain knowledge to determine input parameters;
* ability to deal with noise and outliers;
* insensitivity to order of input records;
* high dimensionality;
* interpretability and usability.

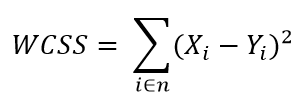
1. Discuss the different approaches for clustering.

Broadly speaking, clustering can be divided into two subgroups :

* **Hard Clustering:** In hard clustering, each data point either belongs to a cluster completely or not. For example, in the above example each customer is put into one group out of the 10 groups.
* **Soft Clustering**: In soft clustering, instead of putting each data point into a separate cluster, a probability or likelihood of that data point to be in those clusters is assigned. For example, from the above scenario each costumer is assigned a probability to be in either of 10 clusters of the retail store.
* ​ **Connectivity models:** As the name suggests, these models are based on the notion that the data points closer in data space exhibit more similarity to each other than the data points lying farther away. These models can follow two approaches. In the first approach, they start with classifying all data points into separate clusters & then aggregating them as the distance decreases. In the second approach, all data points are classified as a single cluster and then partitioned as the distance increases. Also, the choice of distance function is subjective. These models are very easy to interpret but lacks scalability for handling big datasets. Examples of these models are hierarchical clustering algorithm and its variants.
* **Centroid models:** These are iterative clustering algorithms in which the notion of similarity is derived by the closeness of a data point to the centroid of the clusters. K-Means clustering algorithm is a popular algorithm that falls into this category. In these models, the no. of clusters required at the end have to be mentioned beforehand, which makes it important to have prior knowledge of the dataset. These models run iteratively to find the local optima.
* **Distribution models:** These clustering models are based on the notion of how probable is it that all data points in the cluster belong to the same distribution (For example: Normal, Gaussian). These models often suffer from overfitting. A popular example of these models is Expectation-maximization algorithm which uses multivariate normal distributions.
* **Density Models:**These models search the data space for areas of varied density of data points in the data space. It isolates various different density regions and assign the data points within these regions in the same cluster. Popular examples of density models are DBSCAN and OPTICS.

1. What is WCSS?

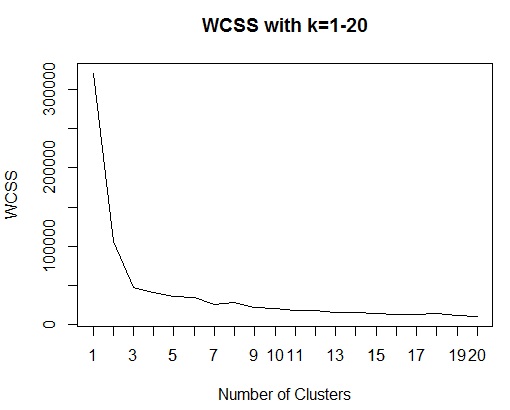
​ **objective function in k-Means** measures sum of distances of observations from their cluster centroids, called Within-Cluster-Sum-of-Squares (WCSS). This is computed as



where Yi is centroid for observation Xi. By definition, this is geared towards maximizing number of clusters, and in limiting case each data point becomes its own cluster centroid. This is, naturally, neither practical nor desirable. Fig. 2 plots WCSS for k=1.20 and we can see that it continuously drops, indicating more clusters the better!

1. Discuss the elbow method.

​ **Elbow-Method using WCSS** – This is one of the most common and technically robust methods. This is based on principle that while clustering performance as measured by WCSS increases (i.e. WCSS decreases) with increase in k, rate of increase is usually decreasing. So performance improvement for increasing number of cluster from, say, 3 to 4 is higher than that for increasing from 4 to 5. Plotting WCSS against increasing k can show an ‘elbow’ which demarks significant drop in rate of increase. Selecting number of clusters corresponding to elbow point achieves reasonable performance without having too many clusters.



1. What is the significance of ‘K’ in K-Means and how is it calculated?

​

|  |  |  |
| --- | --- | --- |
| K-Means clustering intends to partition *n* objects into *k* clusters in which each object belongs to the cluster with the nearest mean. This method produces exactly *k* different clusters of greatest possible distinction. The best number of clusters *k* leading to the greatest separation (distance) is not known as a priori and must be computed from the data. The objective of K-Means clustering is to minimize total intra-cluster variance, or, the squared error function: |  |  |
|  |  |  |
| https://www.saedsayad.com/images/Clustering_kmeans_c.png |  |  |

1. Discuss the step by step implementation of K-Means Clustering.

​

|  |  |  |
| --- | --- | --- |
| **Algorithm** |  |  |
| 1. Clusters the data into *k* groups where *k*  is predefined. 2. Select *k* points at random as cluster centers. 3. Assign objects to their closest cluster center according to the *Euclidean distance* function. 4. Calculate the centroid or mean of all objects in each cluster. 5. Repeat steps 2, 3 and 4 until the same points are assigned to each cluster in consecutive rounds. |  |  |

1. What are the challenges with K-Means?

k-means assume the variance of the distribution of each attribute (variable) is spherical;

all variables have the same variance;

the prior probability for all k clusters are the same, i.e. each cluster has roughly equal number of observations; If any one of these 3 assumptions is violated, then k-means will fail.

There are two challenges that need to handled wisely in order to get the most out of the k-means clustering algorithm:

* Defining the number of clusters
* Determining the initial centroids

|  |  |  |
| --- | --- | --- |
| a large *k* probably decreases the error but increases the risk of overfitting. |  |  |
|  |  |  |

**k*-means++****is an algorithm for choosing the initial values (or “seeds”) for the*k*-means clustering algorithm.*

​

1. Discuss the various improvements in K-Means.

Currently, research about improved K-means algorithm [5] is mainly concentrated in the number of clusters determined, selecting of the cluster center and improving of the clustering and other criteria.

K-means [clustering algorithm](https://www.sciencedirect.com/topics/computer-science/clustering-algorithm) can be significantly improved by using a better [initialization](https://www.sciencedirect.com/topics/engineering/initialization) technique, and by repeating (re-starting) the algorithm.

•

When the data has overlapping clusters, k-means can improve the results of the initialization technique.

•

When the data has well separated clusters, the performance of k-means depends completely on the goodness of the initialization.

•

Initialization using simple furthest point heuristic (Maxmin) reduces the clustering error of k-means from 15% to 6%, on average.

​

Hierarchical Clustering:

1. Discuss the agglomerative and divisive clustering approaches.

​ **Agglomerative Clustering:**Also known as bottom-up approach or hierarchical agglomerative clustering (HAC). A structure that is more informative than the unstructured set of clusters returned by flat clustering. This clustering algorithm does not require us to prespecify the number of clusters. Bottom-up algorithms treat each data as a singleton cluster at the outset and then successively agglomerates pairs of clusters until all clusters have been merged into a single cluster that contains all data.

**Divisive clustering :** Also known as top-down approach. This algorithm also does not require to prespecify the number of clusters. Top-down clustering requires a method for splitting a cluster that contains the whole data and proceeds by splitting clusters recursively until individual data have been splitted into singleton cluster.

given a dataset (d1, d2, d3, ....dN) of size N

at the top we have all data in one cluster

the cluster is split using a flat clustering method eg. K-Means etc

**repeat**

choose the best cluster among all the clusters to split

split that cluster by the flat clustering algorithm

**untill** each data is in its own singleton cluster

1. What are dendrograms?

​ **Dendograms** are used to divide a given cluster into many different clusters.

number of clusters you find appropriate in hierarchical clustering by interpreting the dendrogram

1. Discuss the Hierarchical clustering in detail.

In data mining and statistics, hierarchical clustering analysis is a method of cluster analysis which seeks to build a hierarchy of clusters i.e. tree type structure based on the hierarchy.

There are two types of hierarchical clustering, *Divisive* and *Agglomerative*.

​

1. Discuss the various linkage methods for clustering.

* Single Linkage: cluster distance = smallest pairwise distance
* Complete Linkage: cluster distance = largest pairwise distance
* Average Linkage: cluster distance = average pairwise distance
* Centroid Linkage: cluster distance= distance between the centroids of the clusters
* Ward’s Linkage: cluster criteria= Minimize the variance in the cluster

<https://support.minitab.com/en-us/minitab/18/help-and-how-to/modeling-statistics/multivariate/how-to/cluster-observations/methods-and-formulas/linkage-methods/>

The decision of merging two clusters is taken on the basis of closeness of these clusters. There are multiple metrics for deciding the closeness of two clusters :

* Euclidean distance: ||a-b||2 = √(Σ(ai-bi))
* Squared Euclidean distance: ||a-b||22 = Σ((ai-bi)2)
* Manhattan distance: ||a-b||1 = Σ|ai-bi|
* Maximum distance:||a-b||INFINITY = maxi|ai-bi|
* Mahalanobis distance: √((a-b)T S-1 (-b))   {where, s : covariance matrix}

1. Discuss the differences between K-Means and Hierarchical clustering.

​

* Hierarchical clustering can’t handle big data well but K Means clustering can. This is because the time complexity of K Means is linear i.e. O(n) while that of hierarchical clustering is quadratic i.e. O(n2).
* In K Means clustering, since we start with random choice of clusters, the results produced by running the algorithm multiple times might differ. While results are reproducible in Hierarchical clustering.
* K Means is found to work well when the shape of the clusters is hyper spherical (like circle in 2D, sphere in 3D).
* K Means clustering requires prior knowledge of K i.e. no. of clusters you want to divide your data into. But, you can stop at whatever number of clusters you find appropriate in hierarchical clustering by interpreting the dendrogram

DBSCAN

1. Discuss the basic terms used in DBSCAN.

​ To understand DBSCAN in more detail, let’s dive into it. **The main concept of DBSCAN algorithm is to locate regions of high density that are separated from one another by regions of low density.** So, how do we measure density of a region ? Below are the 2 steps —

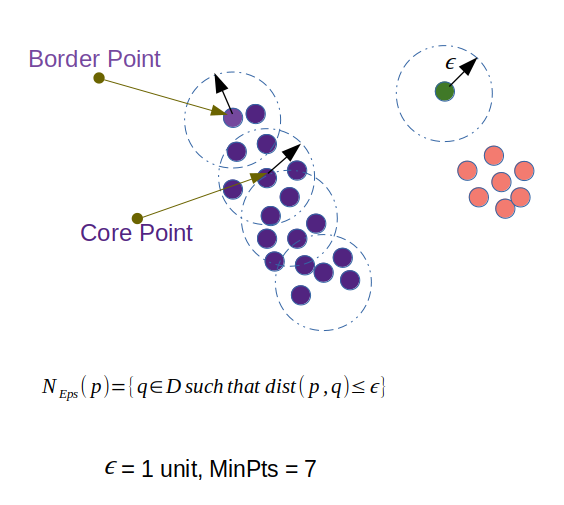
* Density at a point P: Number of points within a circle of Radius Eps (ϵ) from point P.
* Dense Region: For each point in the cluster, the circle with radius ϵ contains at least minimum number of points (MinPts).

The Epsilon neighborhood of a point P in the database D is defined as (following the definition from Ester et.al.)

N (p) = {q ∈ D | dist(p, q) ≤ ϵ}…. (1)

Following the definition of dense region, a point can be classified as a **Core Point** if |N (p)|≥ MinPts. The Core Points, as the name suggests, lie usually within the interior of a cluster. A **Border Point** has fewer than MinPts within its ϵ-neighborhood (N), but it lies in the neighborhood of another core point. **Noise** is any data point that is neither core nor border point. See the picture below for better understanding.

https://miro.medium.com/max/60/1*ejlV2WryiH4zGFP_KohEeA.png?q=20



1. Discuss the step by step implementation of DBSCAN.

With the definitions above, we can go through the steps of DBSCAN algorithm as below —

1. The algorithm starts with an arbitrary point which has not been visited and its neighborhood information is retrieved from the ϵ parameter.
2. If this point contains MinPts within ϵ neighborhood, cluster formation starts. Otherwise the point is labeled as noise. This point can be later found within the ϵ neighborhood of a different point and, thus can be made a part of the cluster. Concept of density reachable and density connected points are important here.
3. If a point is found to be a core point then the points within the ϵ neighborhood is also part of the cluster. So all the points found within ϵ neighborhood are added, along with their own ϵ neighborhood, if they are also core points.
4. The above process continues until the density-connected cluster is completely found.
5. The process restarts with a new point which can be a part of a new cluster or labeled as noise.

From the definitions and algorithm steps above, you can guess two of the biggest drawbacks of DBSCAN algorithm.

* If the database has data points that form clusters of varying density, then DBSCAN fails to cluster the data points well, since the clustering depends on ϵ and MinPts parameter, they cannot be chosen separately for all clusters.
* If the data and features are not so well understood by a domain expert then, setting up ϵ and MinPts could be tricky and, may need comparisons for several iterations with different values of ϵ and MinPts.

​

Kmeans vs DB SCAN:

In short, KMeans is a distance based clustering technique where depending on the distance between the data points your initialization(usually kmeans++) and clustering works. In kmeans, you initialize cluster centers and then find distance between each point and each of the cluster and then you cluster points to their nearest centers. Here the optimization problem we solve is to find the no of clusters such that sum of distances from each point and its nearest cluster is minimized.

Disadvantages of kmeans are

* Kmeans tries to create same sized cluster no matter how the data is scattered
* Kmeans doesnt work well for non-[globular](https://en.wikipedia.org/wiki/Globular_cluster) structures
* Kmeans doesnt care about how dense the data is present
* Curse of dimensionality affects kmeans at high dimension since it uses distance measure

DBSCAN solves some of the problems of kmeans by working with the density of points. This is a density based method. The main assumption of DBSCAN is two dense regions are seperated by one sparse region. Since DBSCAN works with density, it can easily model non-globular structures. This is a high level overview of DBSCAN, I can go into details but thats a separate blog in itself. Hope its helps

The main difference is that they work completely differently and solve different problems.

Kmeans is a least-squares optimization, whereas DBSCAN finds density-connected regions.

Which technique is appropriate to use depends on your data and objectives. If you want to minimize least squares, use k-means. If you want to find density-connected regions use DBSCAN.

Cluster Evaluation

1. What are the aspects of cluster validation?

##### ​ *Aspects of cluster validation*

* **External:** Compare your cluster to the ground truth.
* **Internal:** Evaluating the cluster without reference to external data.
* **Reliability:** The clusters are not formed by chance(randomly)- some statistical framework can be used.

1. What is a confusion matrix?

Prediction vs actual comparisions (TP+TN+FP+FN)

1. ​ 𝐶𝑖𝑗=𝑃𝑖𝑗=1Cij=Pij=1 --> both the points belong to the same cluster for both our algorithm and ground truth(Agree)--- **SS**
2. 𝐶𝑖𝑗=𝑃𝑖𝑗=0Cij=Pij=0 --> both the points don’t belong to the same cluster for both our algorithm and ground truth(Agree)--- **DD**
3. 𝐶𝑖𝑗=1𝑏𝑢𝑡𝑃𝑖𝑗=0Cij=1butPij=0 --> The points belong in the same cluster for our algorithm but in different clusters for the ground truth (Disagree)---- **SD**
4. 𝐶𝑖𝑗=0𝑏𝑢𝑡𝑃𝑖𝑗=1Cij=0butPij=1 --> The points don’t belong in the same cluster for our algorithm but in same clusters for the ground truth (Disagree)----**DS**
5. What is Jaccard’s coefficient?

**Jaccard Coeeficient**=𝑆(𝑆𝑆+𝑆𝐷+𝐷𝑆)

​

1. What is Rand Index?

**Rand Index**= 𝑇𝑜𝑡𝑎𝑙𝐴𝑔𝑟𝑒𝑒𝑇𝑜𝑡𝑎𝑙𝐷𝑖𝑠𝑎𝑔𝑟𝑒𝑒=(𝑆𝑆+𝐷𝐷)(𝑆𝑆+𝐷𝐷+𝐷𝑆+𝑆𝐷)

**SIMILAR TO ACCURACY of classification models**

​

1. What is the entropy of a cluster?

**Entropy** Entropy of Cluster i, given by 𝑒𝑖=−∑𝑝𝑖𝑗𝑙𝑜(𝑝𝑖𝑗)ei=−∑pijlog(pij)

For the entire clustering algorithm, the entropy can be given as: 𝑒=∑𝑚𝑖𝑛𝑒𝑖

​

1. Discuss the purity of a cluster.

​ **Purity** The purity is the total percentage of data points clustered correctly.

The purity of cluster i, given by 𝑝𝑖=𝑚𝑎(𝑝𝑖𝑗)pi=max(pij)

And for the entire cluster it is: (𝐶)=∑𝑚𝑖𝑛𝑝𝑖p(C)=∑minpi

The Scikit-Learn Package hasen't yet implemented the Purity metrics. Hence, we'll write our custom code to implememt that.

1. What are cohesion and compression?

#### Internal Measures

These are the methods use to measure the quality of clusters without external references. There are two aspects to it.

* **Cohesion:** How closely the objects in the same cluster are related to each other. It is the within-cluster sum of squared distances. It is the same metric that we used to calculate for the K-Means algorithm. 𝑊𝐶𝑆𝑆=∑∑(𝑥−𝑚𝑖)2WCSS=∑∑(x−mi)2
* **Separation:** How different the objects in different clusters are and how distinct a well-separated cluster is from other clusters. It is the between cluster sum of squared distances. 𝐵𝑆𝑆=∑𝐶𝑖(𝑚−𝑚𝑖)2BSS=∑Ci(m−mi)2

Where C is the size of the individual cluster and m is the centroid of all the data points.

**Note:** BSS+WSS is always a constant.

The silhouse co-efficient

Where a(x) is the avarage distance of x from all the other points in the same cluster and b(x) is the avarage distance of x from all the other points in the other clusters.

And the Silhoeutte coefficient is given by:

𝑆𝐶=1𝑁∑(𝑥)

​

1. What are the steps for AWS deployment?

​

1. What difficulties did you face while deploying to AWS?

No