1. What is your definition of clustering? What are a few clustering algorithms you might think of?

Clustering algorithms [51] are used to establish pattern similarities so that data that exhibit similar characteristics can be classified into their corresponding target groups. Popular examples of clustering algorithms include hierarchical, expectation maximization, k-medians and k-means clustering approaches. Clustering algorithms are most suited for identifying linear correlations between data classes but their applications can be highly restricted by the non-linearities, noise, multi-dimensionality and significant variabilities often associated with real-life data. This is perhaps why several studies have explored alternative approaches for determining cluster compositions and efficiency

1. What are some of the most popular clustering algorithm applications?

In real world, not every data we work upon has a target variable. This kind of data cannot be analyzed using supervised learning algorithms. We need the help of unsupervised algorithms. One of the most popular type of analysis under unsupervised learning is [Cluster analysis](https://www.geeksforgeeks.org/data-mining-cluster-analysis/). When the goal is to group similar data points in a dataset, then we use cluster analysis. In practical situations, we can use cluster analysis for [customer segmentation](https://www.geeksforgeeks.org/customer-segmentation-using-unsupervised-machine-learning-in-python/) for targeted advertisements, or in medical imaging to find unknown or new infected areas and many more use cases that we will discuss further in this article.

**Table of Content**

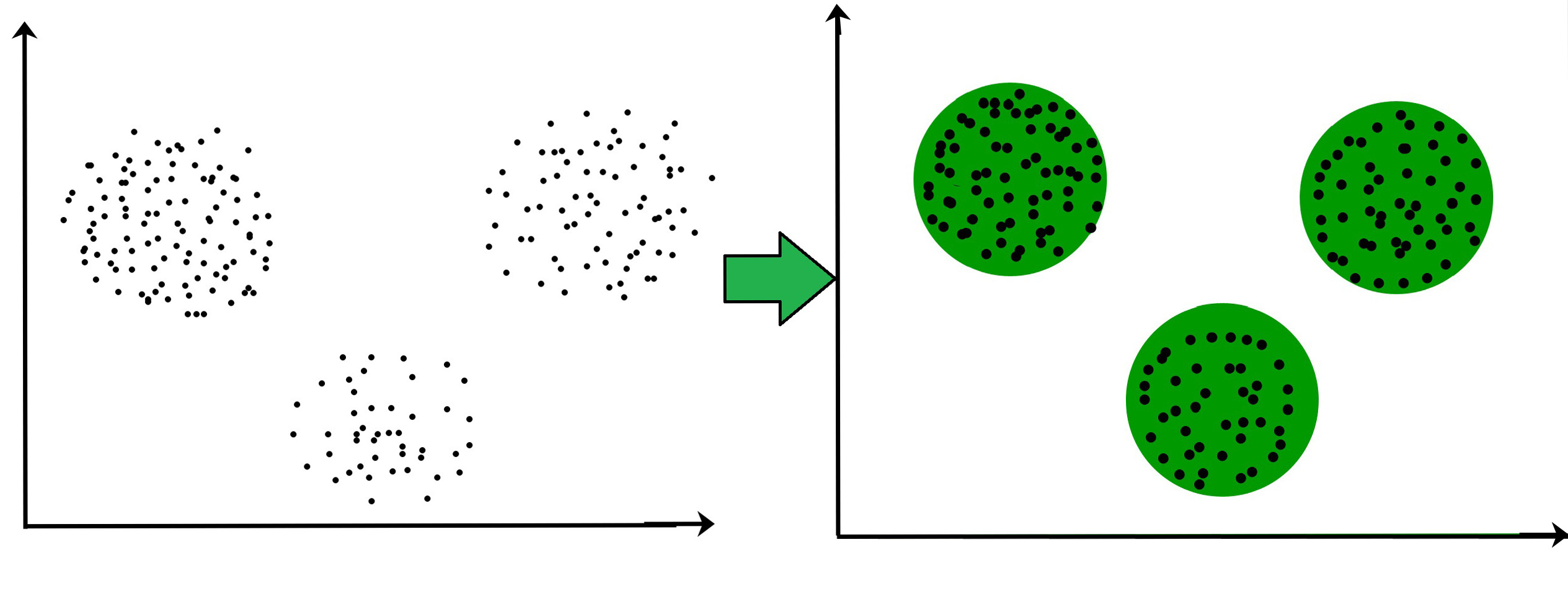
* [What is Clustering ?](https://www.geeksforgeeks.org/clustering-in-machine-learning/#what-is-clustering-)
* [Types of Clustering](https://www.geeksforgeeks.org/clustering-in-machine-learning/#types-of-clustering)
* [Uses of Clustering](https://www.geeksforgeeks.org/clustering-in-machine-learning/#uses-of-clustering)
* [Types of Clustering Algorithms](https://www.geeksforgeeks.org/clustering-in-machine-learning/#types-of-clustering-algorithms)
* [Applications of Clustering in different fields:](https://www.geeksforgeeks.org/clustering-in-machine-learning/#applications-of-clustering-in-different-fields)
* [Frequently Asked Questions (FAQs) on Clustering](https://www.geeksforgeeks.org/clustering-in-machine-learning/#frequently-asked-questions-faqs-on-clustering)

**What is Clustering ?**

The task of grouping data points based on their similarity with each other is called Clustering or Cluster Analysis. This method is defined under the branch of [Unsupervised Learning](https://www.geeksforgeeks.org/supervised-unsupervised-learning/), which aims at gaining insights from unlabelled data points, that is, unlike [supervised learning](https://www.geeksforgeeks.org/supervised-unsupervised-learning/) we don’t have a target variable.

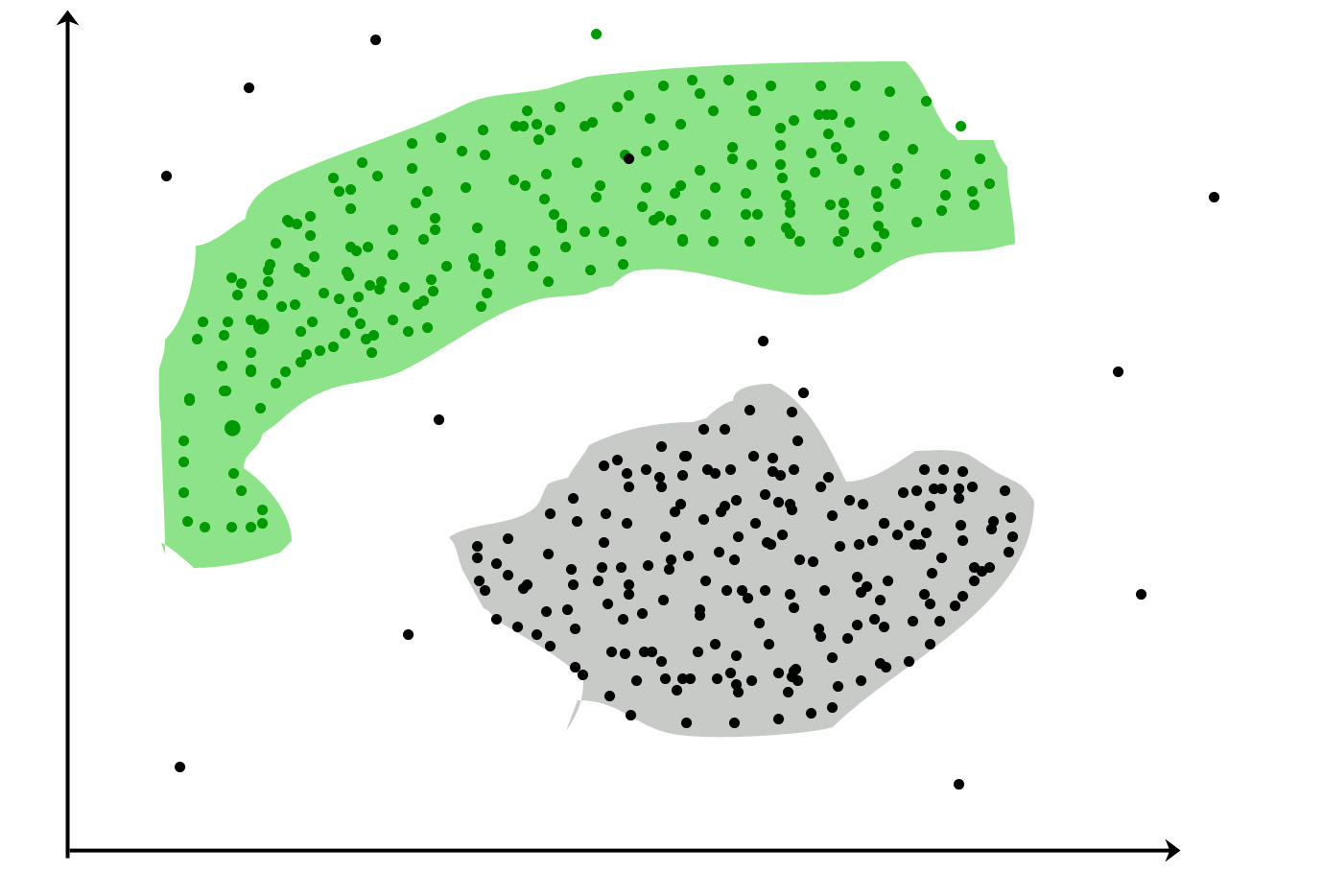
Clustering aims at forming groups of homogeneous data points from a heterogeneous dataset. It evaluates the similarity based on a metric like Euclidean distance, Cosine similarity, Manhattan distance, etc. and then group the points with highest similarity score together.

For Example, In the graph given below, we can clearly see that there are 3 circular clusters forming on the basis of distance.



Now it is not necessary that the clusters formed must be circular in shape. The shape of clusters can be arbitrary. There are many algortihms that work well with detecting arbitrary shaped clusters.

For example, In the below given graph we can see that the clusters formed are not circular in shape.



**Types of Clustering**

Broadly speaking, there are 2 types of clustering that can be performed to group similar data points:

* **Hard Clustering:**In this type of clustering, each data point belongs to a cluster completely or not. For example, Let’s say there are 4 data point and we have to cluster them into 2 clusters. So each data point will either belong to cluster 1 or cluster 2.

| **Data Points** | **Clusters** |
| --- | --- |
| A | C1 |
| B | C2 |
| C | C2 |
| D | C1 |

* **Soft Clustering:**In this type of clustering, instead of assigning each data point into a separate cluster, a probability or likelihood of that point being that cluster is evaluated. For example, Let’s say there are 4 data point and we have to cluster them into 2 clusters. So we will be evaluating a probability of a data point belonging to both clusters. This probability is calculated for all data points.

| Data Points | Probability of C1 | Probability of C2 |
| --- | --- | --- |
| A | 0.91 | 0.09 |
| B | 0.3 | 0.7 |
| C | 0.17 | 0.83 |
| D | 1 | 0 |

**Uses of Clustering**

Now before we begin with types of clustering algorithms, we will go through the use cases of Clustering algorithms. Clustering algorithms are majorly used for:

* [Market Segmentation](https://www.geeksforgeeks.org/customer-segmentation-using-unsupervised-machine-learning-in-python/) – Businesses use clustering to group their customers and use targeted advertisements to attract more audience.
* [Market Basket Analysis](https://www.geeksforgeeks.org/market-basket-analysis-in-data-mining/) – Shop owners analyze their sales and figure out which items are majorly bought together by the customers. For example, In USA, according to a study diapers and beers were usually bought together by fathers.
* [Social Network Analysis](https://www.geeksforgeeks.org/social-network-analysis-using-r-programming/) – Social media sites use your data to understand your browsing behaviour and provide you with targeted friend recommendations or content recommendations.
* Medical Imaging – Doctors use Clustering to find out diseased areas in diagnostic images like X-rays.
* [Anomaly Detection](https://www.geeksforgeeks.org/machine-learning-for-anomaly-detection/) – To find outliers in a stream of real-time dataset or forecasting fraudulent transactions we can use clustering to identify them.
* Simplify working with large datasets – Each cluster is given a cluster ID after clustering is complete. Now, you may reduce a feature set’s whole feature set into its cluster ID. Clustering is effective when it can represent a complicated case with a straightforward cluster ID. Using the same principle, clustering data can make complex datasets simpler.

There are many more use cases for clustering but there are some of the major and common use cases of clustering. Moving forward we will be discussing Clustering Algorithms that will help you perform the above tasks.

**Types of Clustering Algorithms**

At the surface level, clustering helps in the analysis of unstructured data. Graphing, the shortest distance, and the density of the data points are a few of the elements that influence cluster formation. Clustering is the process of determining how related the objects are based on a metric called the similarity measure. Similarity metrics are easier to locate in smaller sets of features. It gets harder to create similarity measures as the number of features increases. Depending on the type of clustering algorithm being utilized in data mining, several techniques are employed to group the data from the datasets. In this part, the clustering techniques are described. Various types of clustering algorithms are:

1. Centroid-based Clustering (Partitioning methods)
2. Density-based Clustering (Model-based methods)
3. Connectivity-based Clustering (Hierarchical clustering)
4. Distribution-based Clustering

We will be going through each of these types in brief.

**1.**[Centroid-based Clustering (Partitioning methods)](https://www.geeksforgeeks.org/partitioning-method-k-mean-in-data-mining/)

Partitioning methods are the most easiest clustering algorithms. They group data points on the basis of their closeness. Generally, the similarity measure chosen for these algorithms are Euclidian distance, Manhattan Distance or Minkowski Distance. The datasets are separated into a predetermined number of clusters, and each cluster is referenced by a vector of values. When compared to the vector value, the input data variable shows no difference and joins the cluster.

The primary drawback for these algorithms is the requirement that we establish the number of clusters, “k,” either intuitively or scientifically (using the Elbow Method) before any clustering machine learning system starts allocating the data points. Despite this, it is still the most popular type of clustering. [K-means](https://www.geeksforgeeks.org/k-means-clustering-introduction/) and [K-medoids](https://www.geeksforgeeks.org/ml-k-medoids-clustering-with-example/) clustering are some examples of this type clustering.

**2.**[Density-based Clustering (Model-based methods)](https://www.geeksforgeeks.org/dbscan-clustering-in-ml-density-based-clustering/)

Density-based clustering, a model-based method, finds groups based on the density of data points. Contrary to centroid-based clustering, which requires that the number of clusters be predefined and is sensitive to initialization, density-based clustering determines the number of clusters automatically and is less susceptible to beginning positions. They are great at handling clusters of different sizes and forms, making them ideally suited for datasets with irregularly shaped or overlapping clusters. These methods manage both dense and sparse data regions by focusing on local density and can distinguish clusters with a variety of morphologies.

In contrast, centroid-based grouping, like k-means, has trouble finding arbitrary shaped clusters. Due to its preset number of cluster requirements and extreme sensitivity to the initial positioning of centroids, the outcomes can vary. Furthermore, the tendency of centroid-based approaches to produce spherical or convex clusters restricts their capacity to handle complicated or irregularly shaped clusters. In conclusion, density-based clustering overcomes the drawbacks of centroid-based techniques by autonomously choosing cluster sizes, being resilient to initialization, and successfully capturing clusters of various sizes and forms. The most popular density-based clustering algorithm is [DBSCAN](https://www.geeksforgeeks.org/dbscan-clustering-in-ml-density-based-clustering/).

**3.**[Connectivity-based Clustering (Hierarchical clustering)](https://www.geeksforgeeks.org/ml-hierarchical-clustering-agglomerative-and-divisive-clustering/)

A method for assembling related data points into hierarchical clusters is called hierarchical clustering. Each data point is initially taken into account as a separate cluster, which is subsequently combined with the clusters that are the most similar to form one large cluster that contains all of the data points.

Think about how you may arrange a collection of items based on how similar they are. Each object begins as its own cluster at the base of the tree when using hierarchical clustering, which creates a dendrogram, a tree-like structure. The closest pairings of clusters are then combined into larger clusters after the algorithm examines how similar the objects are to one another. When every object is in one cluster at the top of the tree, the merging process has finished. Exploring various granularity levels is one of the fun things about hierarchical clustering. To obtain a given number of clusters, you can select to cut the [dendrogram](https://www.geeksforgeeks.org/scipy-cluster-hierarchy-dendrogram/)at a particular height. The more similar two objects are within a cluster, the closer they are. It’s comparable to classifying items according to their family trees, where the nearest relatives are clustered together and the wider branches signify more general connections. There are 2 approaches for Hierarchical clustering:

* [**Divisive Clustering**](https://www.geeksforgeeks.org/difference-between-agglomerative-clustering-and-divisive-clustering/)**:**It follows a top-down approach, here we consider all data points to be part one big cluster and then this cluster is divide into smaller groups.
* [**Agglomerative Clustering**](https://www.geeksforgeeks.org/difference-between-agglomerative-clustering-and-divisive-clustering/)**:**It follows a bottom-up approach, here we consider all data points to be part of individual clusters and then these clusters are clubbed together to make one big cluster with all data points.

**4. Distribution-based Clustering**

Using distribution-based clustering, data points are generated and organized according to their propensity to fall into the same probability distribution (such as a Gaussian, binomial, or other) within the data. The data elements are grouped using a probability-based distribution that is based on statistical distributions. Included are data objects that have a higher likelihood of being in the cluster. A data point is less likely to be included in a cluster the further it is from the cluster’s central point, which exists in every cluster.

A notable drawback of density and boundary-based approaches is the need to specify the clusters a priori for some algorithms, and primarily the definition of the cluster form for the bulk of algorithms. There must be at least one tuning or hyper-parameter selected, and while doing so should be simple, getting it wrong could have unanticipated repercussions. Distribution-based clustering has a definite advantage over proximity and centroid-based clustering approaches in terms of flexibility, accuracy, and cluster structure. The key issue is that, in order to avoid [overfitting](https://www.geeksforgeeks.org/underfitting-and-overfitting-in-machine-learning/), many clustering methods only work with simulated or manufactured data, or when the bulk of the data points certainly belong to a preset distribution. The most popular distribution-based clustering algorithm is [Gaussian Mixture Model](https://www.geeksforgeeks.org/gaussian-mixture-model/).

**Applications of Clustering in different fields:**

1. **Marketing:** It can be used to characterize & discover customer segments for marketing purposes.
2. **Biology:** It can be used for classification among different species of plants and animals.
3. **Libraries:** It is used in clustering different books on the basis of topics and information.
4. **Insurance:** It is used to acknowledge the customers, their policies and identifying the frauds.
5. **City Planning:**It is used to make groups of houses and to study their values based on their geographical locations and other factors present.
6. **Earthquake studies:**By learning the earthquake-affected areas we can determine the dangerous zones.
7. **Image Processing**: Clustering can be used to group similar images together, classify images based on content, and identify patterns in image data.
8. **Genetics:** Clustering is used to group genes that have similar expression patterns and identify gene networks that work together in biological processes.
9. **Finance:** Clustering is used to identify market segments based on customer behavior, identify patterns in stock market data, and analyze risk in investment portfolios.
10. **Customer Service:** Clustering is used to group customer inquiries and complaints into categories, identify common issues, and develop targeted solutions.
11. **Manufacturing**: Clustering is used to group similar products together, optimize production processes, and identify defects in manufacturing processes.
12. **Medical diagnosis:** Clustering is used to group patients with similar symptoms or diseases, which helps in making accurate diagnoses and identifying effective treatments.
13. **Fraud detection:** Clustering is used to identify suspicious patterns or anomalies in financial transactions, which can help in detecting fraud or other financial crimes.
14. **Traffic analysis:** Clustering is used to group similar patterns of traffic data, such as peak hours, routes, and speeds, which can help in improving transportation planning and infrastructure.
15. **Social network analysis:** Clustering is used to identify communities or groups within social networks, which can help in understanding social behavior, influence, and trends.
16. **Cybersecurity:** Clustering is used to group similar patterns of network traffic or system behavior, which can help in detecting and preventing cyberattacks.
17. **Climate analysis:** Clustering is used to group similar patterns of climate data, such as temperature, precipitation, and wind, which can help in understanding climate change and its impact on the environment.
18. **Sports analysis:**Clustering is used to group similar patterns of player or team performance data, which can help in analyzing player or team strengths and weaknesses and making strategic decisions.
19. **Crime analysis:**Clustering is used to group similar patterns of crime data, such as location, time, and type, which can help in identifying crime hotspots, predicting future crime trends, and improving crime prevention strategies.

3. When using K-Means, describe two strategies for selecting the appropriate number of clusters.

**Introduction To Elbow Method**

A fundamental step for any unsupervised algorithm is to determine the optimal number of clusters into which the data may be clustered. Since we do not have any predefined number of clusters in unsupervised learning. We tend to use some method that can help us decide the best number of clusters.  In the case of K-Means clustering, we use Elbow Method for defining the best number of clustering

**What Is the Elbow Method in K-Means Clustering**

As we know in the k-means clustering algorithm we randomly initialize k clusters and we iteratively adjust these k clusters till these k-centroids riches in an equilibrium state. However, the main thing we do before initializing these clusters is that determine how many clusters we have to use.

For determining  K(numbers of clusters) we use Elbow method. **Elbow Method**is a technique that we use to determine the number of centroids(k) to use in a k-means clustering algorithm.  In this method to determine the k-value we continuously iterate for k=1 to k=n (Here n is the [hyperparameter](https://www.geeksforgeeks.org/hyperparameter-tuning/) that we choose as per our requirement). For every value of k, we calculate the within-cluster sum of squares (WCSS) value.

WCSS - It is defined as the sum of square distances between the centroids and

each points.

Now For determining the best number of clusters(k) we plot a graph of k versus their WCSS value. Surprisingly the graph looks like an elbow (*which we will see later*). Also, When k=1 the WCSS has the highest value but with increasing k value WCSS value starts to decrease. We choose that value of k from where the graph starts to look like a straight line.

4. What is mark propagation and how does it work? Why would you do it, and how would you do it?

There are 3 components that you have to consider while throwing darts.

* Force (f)
* Angle (a)
* Wind (w)

You decide randomly to throw the first dart with (f1, a1, w1).

You see that the dart has missed the bulls eye by a big margin. Then you start thinking, was it because the angle was too much, or the force was too little, or the wind caused a massive change in what your anticipated direction of dart movement should have been.

So, we compute errors based on each of these factors.

* the dart was below the target, thus I should marginally increase my force, so that gravity has less time to push my dart downwards.
* since dart hit below the target, I can also consider adjusting my angle marginally, to aim higher.
* with my first throw I saw the dart go to the right of the target due to wind, so I will adjust my throw more towards the left.

With these adjustments, I make a new estimate and throw the dart again with (f2, a2, w2). I repeat this process till I hit the target or till my mom starts shouting at me and questioning why I am wasting so much **time** on throwing darts.

Notice I used the word **marginally** in all cases. If I make drastic changes, I am likely to move further away from the target. Taking small steps usually results in better learning. (Recall concepts about learning rate)

So, what I have described here is the fundamental gradient descent which is used in back prop (BP). The 3 components are effectively 3 neurons of the output layer.

Now let's try to understand BP. Let's draw a parallel to the "hidden" layer. This layer is essentially different parts of our hand. The error made by our hand is propagated back to the brain. In this context, the error is of slightly different format than the obvious error. The error here is technically derived from the initial error.

There are two components here

* Elbow (e)
* Fingers (fi)

Initially we threw it randomly with (e1, fi1). After the first throw, we received feedback that our force, angle and wind were not correct. From this error we computed what changes were required in elbow position and how to hold the dart in our fingers. and computed new (e2, fi2) using the brain.

Thus we have a back prop network, where error is propagated backwards, and the brain serving as the input layer (through observations) does the computation of the forward propagation here.

5. Provide two examples of clustering algorithms that can handle large datasets. And two that look for high-density areas?

There’s a very efficient version of K-Means known as “Online K-Means” which is actually not very well known, I have no idea why.

This version can be extremely efficient for very large datasets so it may very well suit your case.

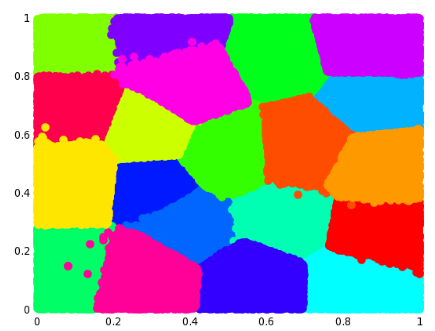
The idea is very simple, you start with k random centroids, which can be k random points sampled from your dataset. Then points are processed one by one, for each point you assign it to the closest centroid (k comparisons). And then move the centroid towards the point proportionally to the number of points in the cluster. So you also keep a k-length vector indicating the number of points in each cluster. The principle is very simple, if a cluster has a lot of points a new point should not move its centroid much.

Here’s some pseudo-code

1. while(more\_points\_to\_process)
2. get point X
3. compute closest centroid to X: mi
4. assign X to mi
5. increment ni, the number of points in cluster i
6. mi = mi + (1/ni) \* (X-mi)

To improve results you can do more than one iteration, remembering to remove the point from the old cluster if it gets reassigned to a new cluster.

A Voronoi diagram after one iteration in 2D can look like this:



We can see some points that are assigned to the wrong cluster but those are only a fraction and a second iteration will correct them, the important thing here is that even after just one iteration we get a reasonable clustering result.

6. Can you think of a scenario in which constructive learning will be advantageous? How can you go about putting it into action?

When I taught adult education many students had problems understanding fraction math. One day, I brought a cake to school and used it to help students figure out adding and subtracting fractions. One of the students had a brain wave and started demonstrating multiplying fractions with the cake.

My granddaughter attended a multi grade, elementary farm school once a week. At the beginning of the day, the students listed all the tasks that needed to be done (feeding animals, planting, harvesting, cooking lunch etc.) and picked their choice. Children learned so much from this process from concrete measurement and division skills to cooperation and empathy.

7. How do you tell the difference between anomaly and novelty detection?

An anomaly is an observation or event that is rare or very different from known or familiar observations or events, while a novelty is a new type of observation not discovered in the learning process. Recognizing, detecting and explaining anomalies and novelties have come to form an area of active research in many domains, including machine learning, computer vision, natural language processing, fraud prevention, cybersecurity and medicine. In these domains, successful and early detection of anomalies and novelties is essential. For example, anomaly detection methods can be applied to:

* Intrusion detection to protect computer networks and data centers from enemy attacks in the cyberspace.
* Fraud detection and anti-money laundering activities in order to prevent damages of millions of dollars.
* Abnormal event detection in video to enable immediate intervention by first responders in case of fights, traffic accidents, explosions, etc.
* Industrial inspection to identify defects and verify the conformity of products and parts.

8. What is a Gaussian mixture, and how does it work? What are some of the things you can do about it?

# Gaussian Mixture Model

Suppose there are a set of data points that need to be grouped into several parts or clusters based on their similarity. In Machine Learning, this is known as Clustering. There are several methods available for clustering:

* [K Means Clustering](https://www.geeksforgeeks.org/k-means-clustering-introduction/)
* [Hierarchical Clustering](https://www.geeksforgeeks.org/ml-hierarchical-clustering-agglomerative-and-divisive-clustering/)
* Gaussian Mixture Models

In this article, Gaussian Mixture Model will be discussed.

### Normal or Gaussian Distribution

In real life, many datasets can be modeled by Gaussian Distribution (Univariate or Multivariate). So it is quite natural and intuitive to assume that the clusters come from different Gaussian Distributions. Or in other words, it tried to model the dataset as a mixture of several Gaussian Distributions. This is the core idea of this model.  
In one dimension the probability density function of a Gaussian Distribution is given by

where  and  are respectively the mean and variance of the distribution. For Multivariate ( let us say d-variate) Gaussian Distribution, the probability density function is given by

Here  is a d dimensional vector denoting the mean of the distribution and  is the d X d covariance matrix.

### Gaussian Mixture Model

Suppose there are K clusters (For the sake of simplicity here it is assumed that the number of clusters is known and it is K). So  and  are also estimated for each k. Had it been only one distribution, they would have been estimated by the **maximum-likelihood method**. But since there are K such clusters and the probability density is defined as a linear function of densities of all these K distributions, i.e.

where  is the mixing coefficient for kth distribution. For estimating the parameters by the maximum log-likelihood method, compute p(X|, , ).

Now define a random variable such that =p(k|X).

From Bayes theorem,

Now for the log-likelihood function to be maximum, its derivative of with respect to , , and  should be zero. So equating the derivative of with respect to to zero and rearranging the terms, 

Similarly taking the derivative with respect to  and pi respectively, one can obtain the following expressions.  
 And

***Note:****denotes the total number of sample points in the kth cluster. Here it is assumed that there is a total N number of samples and each sample containing d features is denoted by .*

So it can be clearly seen that the parameters cannot be estimated in closed form. This is where the **Expectation-Maximization algorithm** is beneficial.

### Expectation-Maximization (EM) Algorithm

The Expectation-Maximization (EM) algorithm is an iterative way to find maximum-likelihood estimates for model parameters when the data is incomplete or has some missing data points or has some hidden variables. EM chooses some random values for the missing data points and estimates a new set of data. These new values are then recursively used to estimate a better first date, by filling up missing points, until the values get fixed.

In the Expectation-Maximization (EM) algorithm, the estimation step (E-step) and maximization step (M-step) are the two most important steps that are iteratively performed to update the model parameters until the model convergence.

#### Estimation Step (E-step):

* In the estimation step, we first initialize our model parameters like the mean (μk​), covariance matrix (Σk​), and mixing coefficients (πk​).
* For each data point, We calculate the posterior probabilities of data points belonging to each centroid using the current parameter values. These probabilities are often represented by the latent variables γk​.
* At the end Estimate the values of the latent variables  γ k ​ based on the current parameter values

#### ****Maximization Step****

* In the maximization step, we update parameter values ( i.e. , and) using the estimated latent variable γk.
* We will update the mean of the cluster point (μk​) by taking the weighted average of data points using the corresponding latent variable probabilities
* We will update the covariance matrix (Σk​) by taking the weighted average of the squared differences between the data points and the mean, using the corresponding latent variable probabilities.
* We will update the mixing coefficients (πk​) by taking the average of the latent variable probabilities for each component.

#### Repeat the E-step and M-step until convergence

* We iterate between the estimation step and maximization step until the change in the log-likelihood or the parameters falls below a predefined threshold or until a maximum number of iterations is reached.
* Basically, in the estimation step, we update the latent variables based on the current parameter values.
* However, in the maximization step, we update the parameter values using the estimated latent variables
* This process is iteratively repeated until our model converges.

The Expectation-Maximization (EM) algorithm is a general framework and can be applied to various models, including Gaussian Mixture Models (GMMs). The steps described above are specifically for GMMs, but the overall concept of the Estimization-step and Maximization-step remains the same for other models that use the EM algorithm.

9. When using a Gaussian mixture model, can you name two techniques for determining the correct number of clusters?

Suppose there are a set of data points that need to be grouped into several parts or clusters based on their similarity. In Machine Learning, this is known as Clustering. There are several methods available for clustering:

* [K Means Clustering](https://www.geeksforgeeks.org/k-means-clustering-introduction/)
* [Hierarchical Clustering](https://www.geeksforgeeks.org/ml-hierarchical-clustering-agglomerative-and-divisive-clustering/)
* Gaussian Mixture Models

In this article, Gaussian Mixture Model will be discussed.

### Normal or Gaussian Distribution

In real life, many datasets can be modeled by Gaussian Distribution (Univariate or Multivariate). So it is quite natural and intuitive to assume that the clusters come from different Gaussian Distributions. Or in other words, it tried to model the dataset as a mixture of several Gaussian Distributions. This is the core idea of this model.  
In one dimension the probability density function of a Gaussian Distribution is given by

where  and  are respectively the mean and variance of the distribution. For Multivariate ( let us say d-variate) Gaussian Distribution, the probability density function is given by

Here  is a d dimensional vector denoting the mean of the distribution and  is the d X d covariance matrix.

### Gaussian Mixture Model

Suppose there are K clusters (For the sake of simplicity here it is assumed that the number of clusters is known and it is K). So  and  are also estimated for each k. Had it been only one distribution, they would have been estimated by the **maximum-likelihood method**. But since there are K such clusters and the probability density is defined as a linear function of densities of all these K distributions, i.e.

where  is the mixing coefficient for kth distribution. For estimating the parameters by the maximum log-likelihood method, compute p(X|, , ).

Now define a random variable such that =p(k|X).

From Bayes theorem,

Now for the log-likelihood function to be maximum, its derivative of with respect to , ,and  should be zero. So equating the derivative of with respect to to zero and rearranging the terms, 

Similarly taking the derivative with respect to  and pi respectively, one can obtain the following expressions.  
And

***Note:****denotes the total number of sample points in the kth cluster. Here it is assumed that there is a total N number of samples and each sample containing d features is denoted by .*

So it can be clearly seen that the parameters cannot be estimated in closed form. This is where the **Expectation-Maximization algorithm** is beneficial.

### Expectation-Maximization (EM) Algorithm

The Expectation-Maximization (EM) algorithm is an iterative way to find maximum-likelihood estimates for model parameters when the data is incomplete or has some missing data points or has some hidden variables. EM chooses some random values for the missing data points and estimates a new set of data. These new values are then recursively used to estimate a better first date, by filling up missing points, until the values get fixed.

In the Expectation-Maximization (EM) algorithm, the estimation step (E-step) and maximization step (M-step) are the two most important steps that are iteratively performed to update the model parameters until the model convergence.

#### Estimation Step (E-step):

* In the estimation step, we first initialize our model parameters like the mean (μk​), covariance matrix (Σk​), and mixing coefficients (πk​).
* For each data point, We calculate the posterior probabilities of data points belonging to each centroid using the current parameter values. These probabilities are often represented by the latent variables γk​.
* At the end Estimate the values of the latent variables  γ k ​ based on the current parameter values

#### ****Maximization Step****

* In the maximization step, we update parameter values ( i.e. , and) using the estimated latent variable γk.
* We will update the mean of the cluster point (μk​) by taking the weighted average of data points using the corresponding latent variable probabilities
* We will update the covariance matrix (Σk​) by taking the weighted average of the squared differences between the data points and the mean, using the corresponding latent variable probabilities.
* We will update the mixing coefficients (πk​) by taking the average of the latent variable probabilities for each component.

#### Repeat the E-step and M-step until convergence

* We iterate between the estimation step and maximization step until the change in the log-likelihood or the parameters falls below a predefined threshold or until a maximum number of iterations is reached.
* Basically, in the estimation step, we update the latent variables based on the current parameter values.
* However, in the maximization step, we update the parameter values using the estimated latent variables
* This process is iteratively repeated until our model converges.

The Expectation-Maximization (EM) algorithm is a general framework and can be applied to various models, including Gaussian Mixture Models (GMMs). The steps described above are specifically for GMMs, but the overall concept of the Estimization-step and Maximization-step remains the same for other models that use the EM algorithm.