The Math Behind the K-means and Hierarchical Clustering Algorithm!

Understanding Clustering:

Rohit Batra · Follow

13 min read · Feb 11, 2022

Clustering is a method used in the unsupervised learning technique of machine learning where you find patterns based on similarities in the data.

As we all know, machine learning has two main techniques, Supervised and Unsupervised learning used widely in the industry to make predictions on historical data. Supervised learning is similar to providing guidance or supervision available for learning the independent and dependent variables in the historical data. It starts with identifying the relationship between the variables followed by creating a model and then predicting. The model here is defining rules that the algorithm has derived after completing the learning from the data.

On the other hand, in unsupervised learning, there is no supervision or guidance available for learning, whereas the objective is to discover patterns in the data i.e. any subgroups among the data. There is no target variable provided to you while solving the business problem hence no evaluation is possible for the results generated by the algorithm. In such a case, you implement the concept of segmentation that is widely used in marketing companies. Under segmentation, you can bifurcate the data into different segments, based on various features and similar characteristics. These segments eventually formed are known as clusters which are built using the algorithm called “Clustering”. These clusters are also formed based on the mathematical methods and your business understanding.

Cluster Profiling: As we learned, clustering is used to place the data elements into related groups without any prior knowledge of the group definitions. Also, it does not require the bifurcation of data into dependent and independent variables. Hence, post clustering you will be required to perform EDA on each segment in order to understand the profile of the clusters to evaluate the algorithm.

Applications of Clustering in the Real-World:

Customer Segmentation for targeted marketing is one of the most vital applications of the clustering algorithm. In such a scenario, you do not have any label in mind, such as a good customer or a bad customer. You just want to look at the patterns and then try and find the segments. This is where clustering techniques can help you with segmenting the customers. The technique uses raw data to form clusters based on common factors among various data points and differences between them. As a manager, you would have to decide what is the important business criteria are on which you would want to segment the customers. So, you would need a method or an algorithm that itself decides which customer to group together based on these criteria.

Clustering and Segmentation are two uncommon methods. Segmentation is a business case in which we use clustering as an analytical technique to form segments. We can segment people, products, markets, etc. How to segment them and in which manner is the part of clustering. After segmenting is done, elements falling in the same bucket should remain in the same bucket and the segments should not change dynamically. This is quite important to assure the stability of a successful segmentation.

> *The three most used segmentation methods are:*

- *Behavioral segmentation:* This is based on the actual patterns displayed by the customers. Basically, how does a customer explicitly exhibit its action?
- *Attitudinal segmentation:* This is based on the intents of a customer, which may not translate into actions. Basically, intent vs actual patterns of the observations.

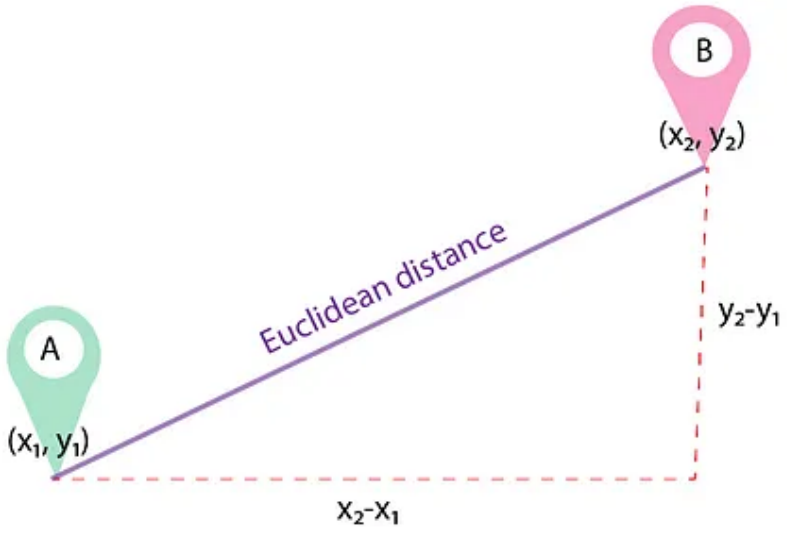
- *Demographic segmentation:* This is based on a customer's profile which uses information such as age, gender, location, income, etc.

The two most commonly used clustering algorithms are K-means clustering and hierarchical clustering. Let's learn more about them in detail.

K-means clustering

As we have seen about how clustering works — it groups the objects on the basis of their similarity or closeness to each other. But, what does this exactly mean?

In more understandable terms, the clustering algorithm needs to find data points whose characteristics are similar to each other and therefore these points would then belong to the same cluster. The method in which any clustering algorithm goes about doing that is through the method of finding something called a "distance measure". The distance measure that is used in K-means clustering is called the *Euclidean Distance* measure. Let's look at the below equation to understand how this value is calculated. The idea behind this is, how do we quantify that two things are similar to each other?



the Euclidean Distance between the 2 points is measured as follows:

Observation A = (X1, Y1)

Observation B = (X2, Y2),

The Euclidean distance of two observations is given by:

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

But what If there are 2 observations X and Y having n dimensions,

X = (X1, X2, X3, …., Xn)

Y = (Y1, Y2, Y3, …., Yn)

Then the *Euclidean Distance D* is given as,

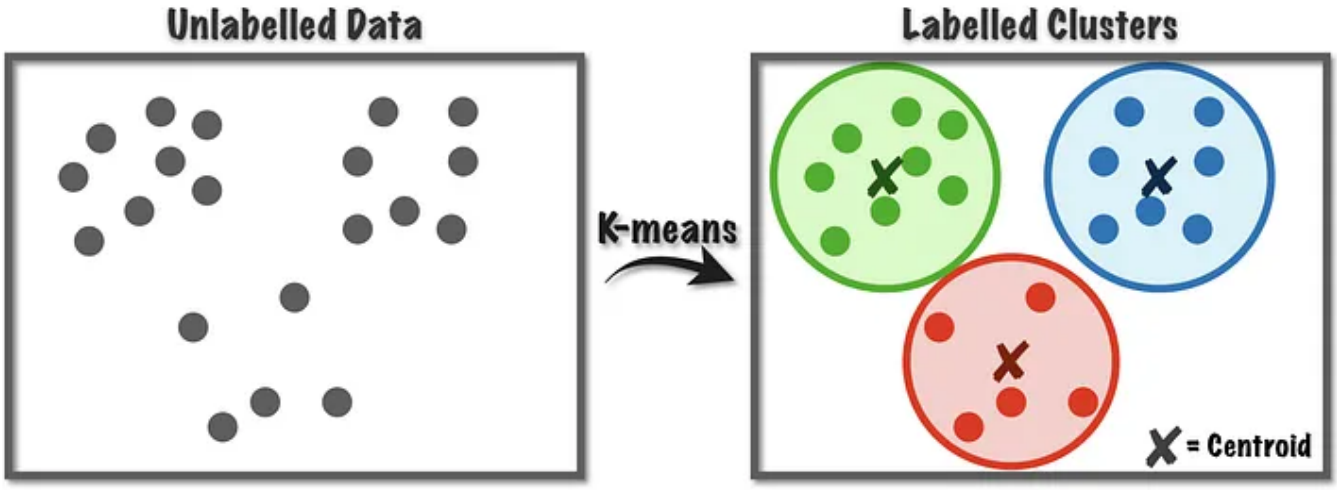
$$d(x, y) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2}$$

The idea of distance measure is quite intuitive. Essentially, the observations which are closer or more similar to each other would have a low Euclidean

distance and the observations which are farther or less similar to each other would have a higher Euclidean distance. Now once you've computed the Euclidean distance, the next step is pretty straightforward for the Clustering Algorithm. All it has to do is compute these distances and then find out which observations or points have a low Euclidean distance between them, i.e. are closer to each other, and then cluster them together.

Centroid:

How clustering generally works is the idea of centroids. Centroids are essentially the center points of triangles. Similarly, in the case of clustering, the centroids are essentially the cluster centers of a group of observations that help us in summarising the cluster's properties. Thus, the centroid value in the case of clustering is essentially the mean of all the observations that belong to a particular cluster.



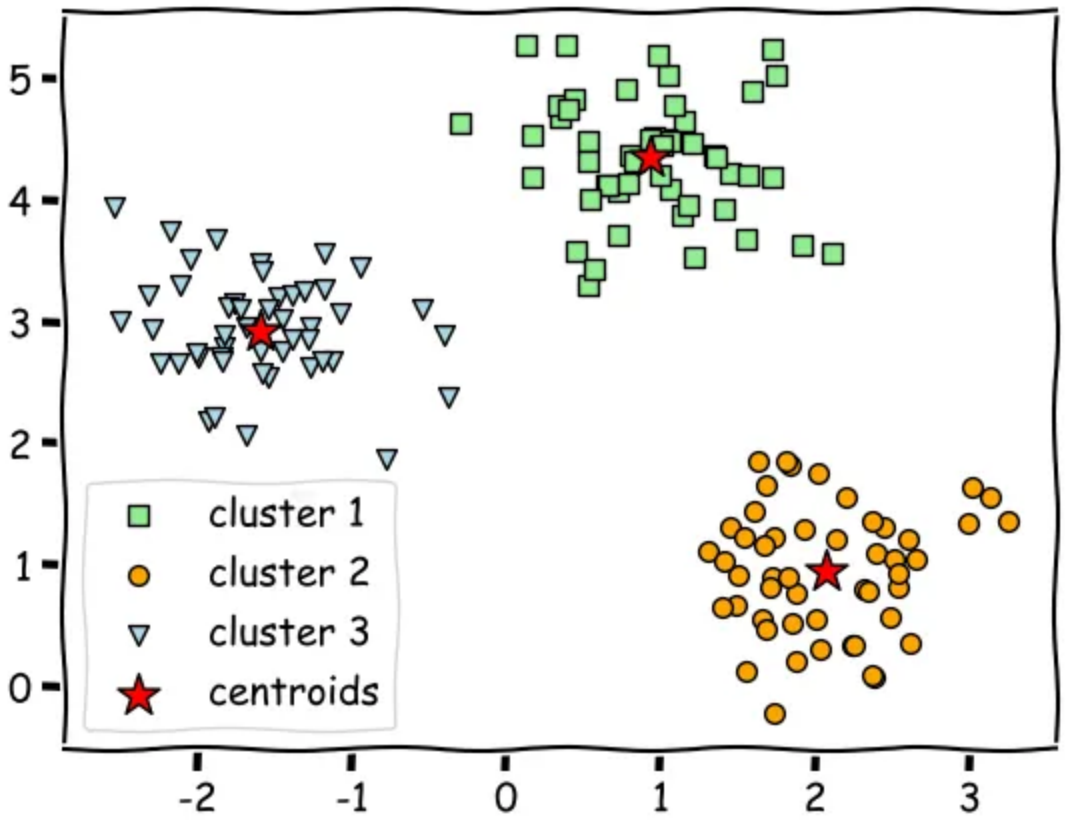
How its algorithm works:

The K-means algorithm uses the concept of centroid to create the clusters. In simple terms, a centroid of n points on an X — Y plane is another point having its own x and y coordinates and is often referred to as the geometric center of the n points. For example, consider three points having coordinates (X1, Y1), (X2, Y2), and (X3, Y3). The centroid of these three points is the average of the x and y coordinates of the three points, i.e.

(X1 + X2 + X3 / 3, Y1 + Y2 + Y3 / 3).

Similarly, if you have n points, the formula (coordinates) of the centroid will be:

(X1+X2.....+Xn / n, Y1+Y2.....+Yn / n).



Suppose we have a few data points to be divided into 3 clusters based on their Euclidean distance from their centroid.

Here, n = number of data points, and

k = number of clusters.

We start by choosing random k initial centroids.

Step-1 = Here, we first calculate the distance of each data point to the two cluster centers (initial centroids) and allocate the data points to the closet centroid individually with the least distance using Euclidean distance. This step is known as the "Assignment Step".

Step-2 = The next step is to re-compute the centroid which will simply be the mean of individual points in each of the clusters. Then we will get our new cluster centers (next optimal centroids). This is known as the "Optimization Step"

After computing two new centroids, we will again go back to Step-1, again it will assign each data point to the nearest cluster (those new optimal clusters) using the same method i.e. computing the Euclidean distance from a data point to the centroid and assigning it to the nearest centroid. Post that it will again perform the optimization step for both the clusters which will update the position of the centers of the clusters. The algorithm keeps iterating through this process of Assignment and Optimization till the centroids no longer update and reach convergence. At this point, the algorithm has reached an optimal grouping and we have got out three clusters. Thus essentially, you can see that the K-means method is a clustering algorithm

that takes n points and group them into k clusters. The grouping is done in a way:

1. To maximize the tightness/closeness of individual clusters.
2. While maximizing the distance between different clusters.

K-Means++ algorithm

To choose the cluster centers smartly, we will compute the K-Mean++ algorithm. K-means++ is just an initialization procedure for K-means. In K-means++ you pick the initial centroids using an algorithm that tries to initialize centroids that are far apart from each other.

In K-Means++ algorithm:

- We choose one data point out of all data points (Xi) on the x-y plane as the cluster center at random.
- For each data point Xi, We compute the distance (di) between Xi and the nearest center that had already been chosen and square that distance (di)2.
- Now, we choose the next cluster center using the weighted probability distribution where a point X is chosen with probability proportional to d(Xi)2. This means the data point is farthest to the first cluster center.
- Repeat Steps 2 and 3 until K centers have been chosen.

Upon trying the different options, you may have noticed that the final clusters that you obtain vary depending on many factors, such as the choice

of the initial cluster centers and the value of K, i.e. the number of clusters that you want.

> *the major practical considerations involved in K-Means clustering are:*

Impact of the initial centroids: The number of clusters that you want to divide your data points into, i.e. the value of K has to be pre-determined.

Choosing the number of clusters in advance: The choice of the initial cluster centers can have an impact on the final cluster formation.

Impact of Outliers: The clustering process is very sensitive to the presence of outliers in the data.

Standardization of data: Since the distance metric used in the clustering process is the Euclidean distance, you need to bring all your attributes on the same scale. This can be achieved through "standardization".

The K-Means algorithm **does not work with categorical data**.

The process may not converge in the given number of iterations. You should always check for convergence.

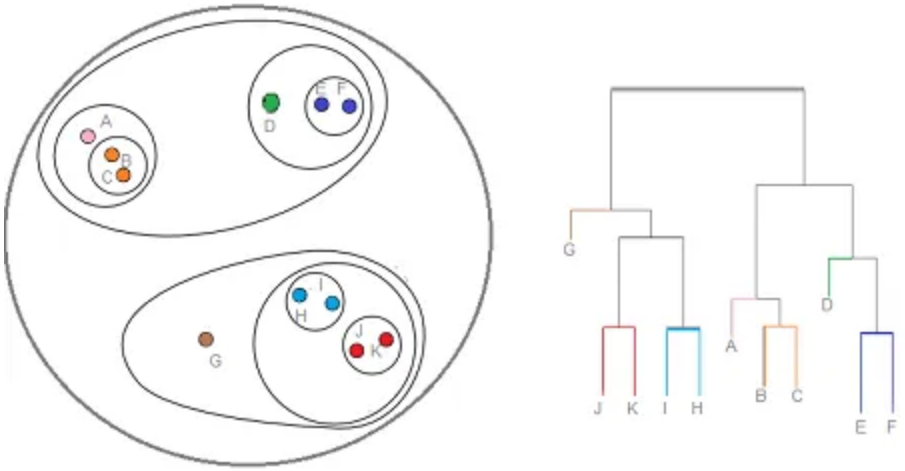
Hierarchical clustering:

This is another algorithm to achieve unsupervised clustering. Here, instead of pre-defining the number of k, you first have to visually describe the similarity or dissimilarity between the different data points and then decide the appropriate number of clusters based on these similarities or dissimilarities.

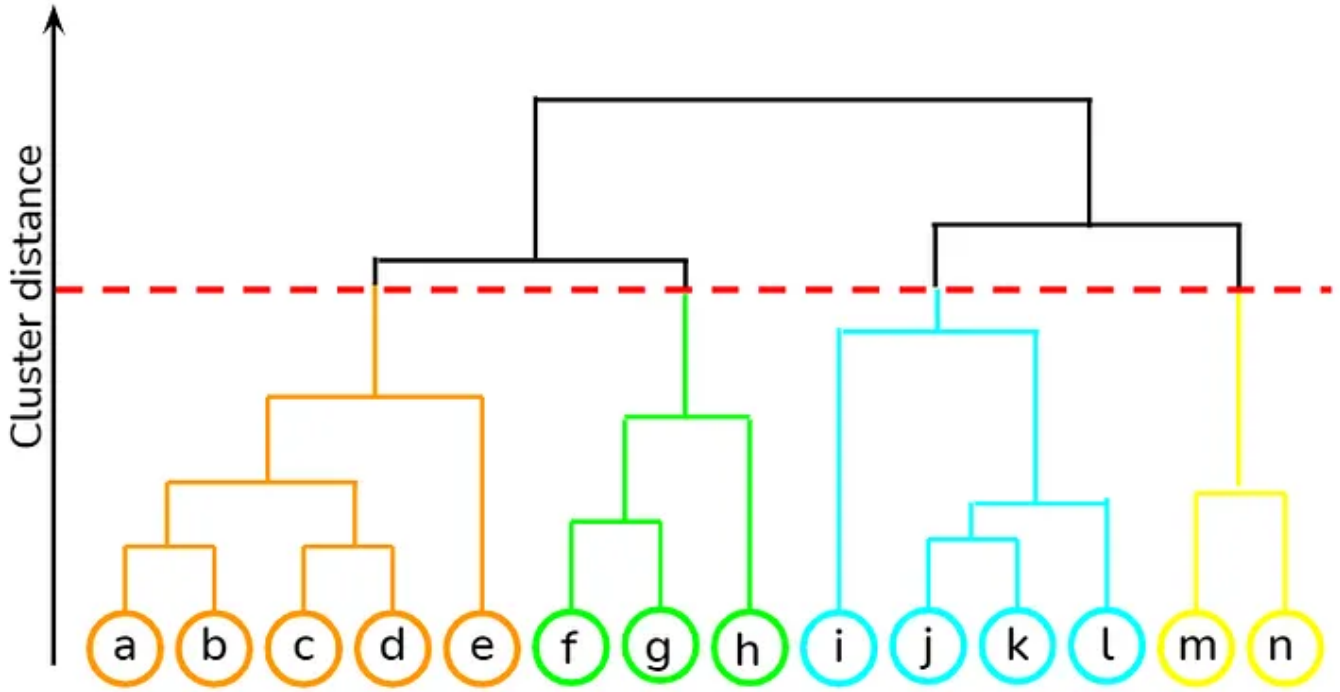
As we have learned in the previous section, the potential disadvantage of K-means is the fact that we have to choose the value of k or the number of clusters in advance. Whereas hierarchical clustering does not have this limitation.

Let's suppose we have a set of 10 points from A to K. We begin with specifying a measure of difference or a dissimilarity measure between the observations. As in most cases, Euclidean distance is the most common measure however other measures can be used. The points with the smaller value of the dissimilarity measure are more similar to each other. The algorithm also proceeds iteratively. We first consider each of the n observations as a separate individual cluster so in this case n = 11. We begin by having 11 clusters. In the next step, the two clusters that are close to each other are then fused to form a cluster. Since we have chosen Euclidean distance as a dissimilarity measure, we do this by calculating the distance of each point to each of the other points in the data set.

We then, fuse the two points with the smallest pairwise value between them. So, after this step, we are left with n-1 clusters i.e. 10 clusters in our case. These steps are repeated multiple times till we do not achieve 1 single cluster. Now the question that might come to your mind is, how do you calculate the distance between clusters that contain groups of observations. So in this case, how do you calculate the distance or dissimilarity between cluster B & C as 1 set and cluster A which is another set. This is achieved by defining the concept of linkage.



Linkage: It is a measure of dissimilarity between clusters having multiple observations. There are different ways to measure this but the most common way is taking the minimum of the pair-wise distance between the points in the two clusters which is also called 'Single Linkage'. In our case, we would first calculate the distance between B & A and the A & C and then take the minimum of these two distances as a measure of dissimilarity between these two clusters. So, now we go back to our step and again fuse the two closet clusters which in this case are B & C and the with cluster A. Now we have left with n — 2 i.e. 9 clusters. The algorithm continues in this fashion till all points are fused together to form 1 single cluster. Now, you would notice that this had formed a tree-like structure as we iterate through steps. This structure is called a 'Dendrogram'. At each step of the algorithm we are reducing the number of clusters and building up this tree and this is why this algorithm is called hierarchical clustering algorithm. The height of the dendrogram at which different clusters are formed together represents the dissimilarity measure which in this case is the Euclidean distance between the clusters. The cluster fused together on the top of the tree are more dissimilar to each other than the ones fused together at the bottom.



How to interpret the dendrogram?

The result of the cluster analysis starts with all the data points as a separate cluster and indicates at what level of dissimilarity any two clusters were joined. This is shown by the dendrogram. Now let us try to understand how to identify clusters on the basis of the dendrogram that we have obtained from our set of points. To do this, we make a horizontal cut across the dendrogram. The distinct groups of observations beneath this cut represent the number of clusters. You can think of this as a number of distinct vertical lines intersecting the horizontal cut lines. So in this case, you can identify 4 distinct clusters where we have cut the dendrogram. So essentially once we get this dendrogram, we need to decide the optimal threshold value t which to cut the dendrogram and the number of clusters will depend on this threshold value. This algorithm does not have the limitation to predict the number of clusters between n to 1 depending on where you decide your threshold and cut the tree. This hierarchical clustering approach that goes bottom to top is called “Agglomerative Clustering”. There is another

approach called “Divisive Clustering” which goes top to bottom. In this case, all the observations together are first considered as 1 cluster, the splits are performed based on the farthest distance or least similarity.

> *Now let’s hear from our industry experts regarding the comparison between the K-Means algorithm and the Hierarchical clustering algorithm, before learning how to choose between the two based on your business problem.*

The most common question of all is where and when to use hierarchical segmentation and k-means segmentation? And the most common answer heard in the industry is, whenever there is a huge data, use k-means clustering and in the case of a small set of data, we should use hierarchical clustering. This is a perfectly relevant answer but we need to understand why is this notion true?

At every stage of the hierarchical clustering, you can either start from the top or you can start from the bottom and start building the tree. And then you try to conclude by different means at which layer of the tree I should cut it and those becomes my segments. In this process, it’s a linear method because one leaf once connected by a branch cannot fall into another branch, it can only be added up and there is no way the element can go in the other segment. This is also a rigorous and heavy process which is sometimes very difficult while handling a bigger size of the data. But size is not the constraint here, as this heavy process can be done in a cloud environment.

However, in K-means segmentation is a non-linear process. The only biggest challenge here is you have to mention what is k and to do that, you have to find the optimal value of k. Now suppose, somehow I know what is k, what it does is randomly it will select different points i.e equal to k as their starting

step, and then it will start looking for its optimal center. After which it will start assigning the data point based on their Euclidean distance. This process is repeated multiple times in which it recalculates its centroid and accordingly assigns the data points based on their closet distance. As this also happens in multiple iterations, you need to mention how many iterations sometimes might not match with your final convergence.

Hierarchical clustering generally produces better clusters but is more computationally intensive.

End Notes:

We understood that the algorithm's inner-loop iterates over two steps:

1. Assignment Step: Assign each observation Xi to the closest cluster centroid k
2. Optimization Step: Update each centroid to the mean of the points assigned to it.

All we need to know about clustering!!

This is an educational post made by the compilation of materials from my master's at IIIT Bangalore, that helped me in my journey. If you would like to put a glance at a complete Clustering project, a complete case study can be found in this GitHub repository.

— — — — — —

Thank you for reading!

Machine Learning | Clustering | K Means | K Means Clustering | Data Science

Written by Rohit Batra

Follow

81 Followers

👋 Hi, I'm Rohit Batra. 🌱 I'm pursuing a Master's in Data Science @ International University of Applied Science, Berlin. 📫to reach me...rohitbatra027@gmail.com

More from Rohit Batra

Rohit Batra

Multi-Class Text Classification with Scikit-Learn using TF-IDF model

Problem Formulation

Rohit Batra

Data Storytelling+Methodology Used-Airbnb, NYC Analysis using...

📄 Purpose: #Methodological Document for Data Analysis on Airbnb, NYC

12 min read · May 23, 2022

17 1

17 min read · Mar 17, 2022

7

Rohit Batra

Linear Regression algorithm using statsmodels and scikit-learn!

Statistics behind Linear Regression model:

10 min read · Jan 22, 2022

4

Rohit Batra

Fundamentals of Logistic Regression!

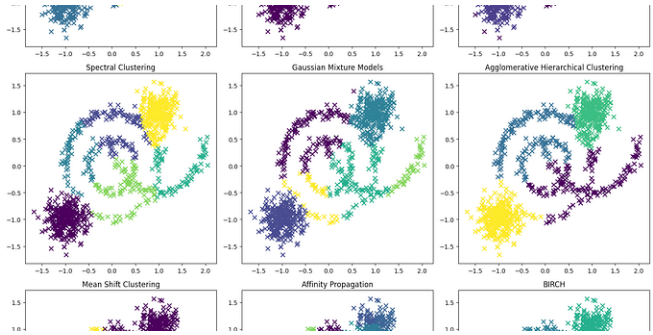
Generally, machine learning models can be classified into two major learning methods,...

9 min read · Mar 13, 2022

4

See all from Rohit Batra

Recommended from Medium



Nirmal Sankalana

K-means Clustering: Choosing Optimal K, Process, and Evaluatio...

In today's data-driven world, businesses and researchers encounter a huge amount of...

16 min read · Sep 19, 2023

2

Sina Nazeri

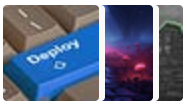
Comparing The-State-of-The-Art Clustering Algorithms

Let's generate complex data and try different clustering algorithms

9 min read · Jul 19, 2023

106

Lists



Predictive Modeling w/ Python

20 stories · 783 saves

Practical Guides to Machine Learning

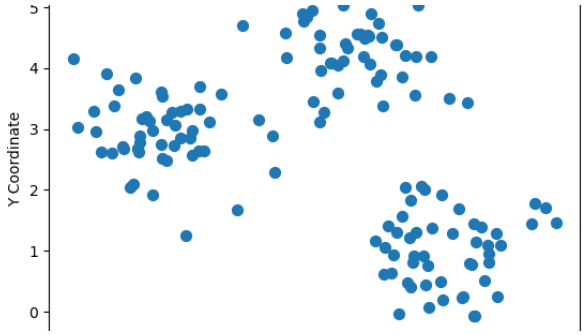
10 stories · 907 saves

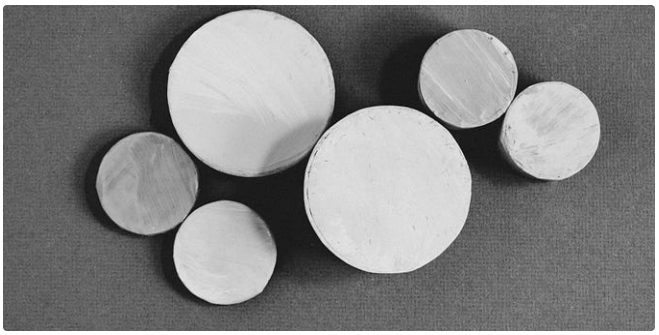
Natural Language Processing

1094 stories · 560 saves

data science and AI

39 stories · 41 saves





Megha Natarajan

Deciphering Optimal Clusters: Elbow Method vs. Silhouette...

When stepping into the realm of unsupervised learning, k-means clustering...

6 min read · Oct 16, 2023

3

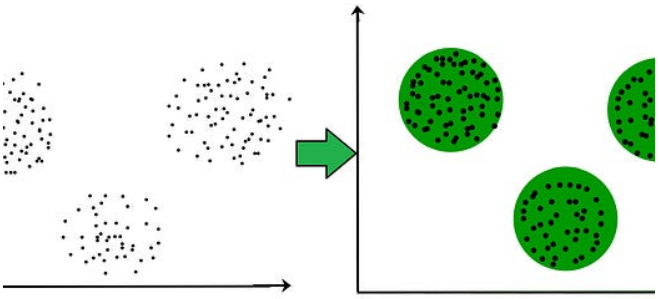
Kay Jan Wong in Towards Data Science

6 Types of Clustering Methods—An Overview

Types of clustering methods and algorithms and when to use them

· 8 min read · Mar 24, 2023

447 1



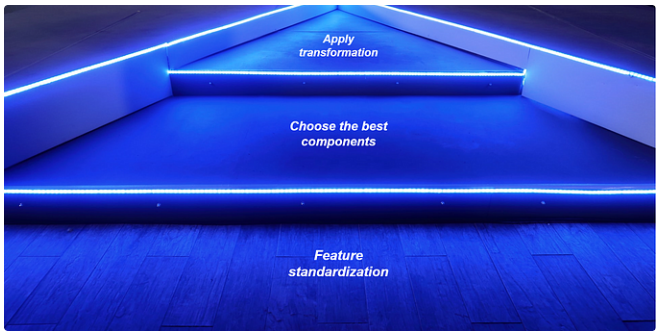
Kasun Dissanayake in Towards Dev

Machine Learning Algorithms(14) —K-Means Clustering and...

In this article, we are learning about K-means Clustering and Hierarchical Clustering. In...

19 min read · Jan 6

462 1



Rukshan Pramoditha in Data Science 365

3 Easy Steps to Perform Dimensionality Reduction Using...

Running the PCA algorithm twice is the most effective way of performing PCA

· 10 min read · Jan 3, 2023

152 2

See more recommendations