LEARNING k-MEANS CLUSTERING FROM SCRATCH - In Python

The k-means clustering algorithm is a simple and popular unsupervised machine-learning algorithm. It is used for clustering problems, with no pre-defined labels defined, meaning that we do not have a target variable or a class for supervised learning. It is often referred to as the unsupervised algorithm.

PartedClustering - k-Means in Sci-Kit Learn

Mar 18, 2020 · PartedClustering is a library that provides a performant machine-learning implementation of the k-means clustering algorithm. It provides the user with a simple and easy-to-use interface for performing k-means clustering on a dataset.

k-Means Algorithm

K-Means is a versatile clustering technique that is used in a variety of fields, from image processing to natural language processing. It is a simple and efficient algorithm that can be used to group data points into clusters, where each cluster represents a group of similar data points.

K-Means Clustering: The Complete Guide

In this article, we will cover the basics of the k-means clustering algorithm, including its motivation, the intuition behind it, and some of the key steps involved in implementing it. We will also look at some of the limitations of the algorithm and how they can be addressed.

k-Means Variance

K-Means variance is a measure of the spread of data points within each cluster. It is calculated as the sum of the squared distances between each data point and its cluster center. A higher variance indicates that the data points are more spread out, while a lower variance indicates that they are closer together.


K-Means variance is a measure of the spread of data points within each cluster. It is calculated as the sum of the squared distances between each data point and its cluster center. A higher variance indicates that the data points are more spread out, while a lower variance indicates that they are closer together.

The k-means algorithm is an iterative process that starts with an initial guess of cluster centers and then iterates until the cluster centers stop moving or a maximum number of iterations is reached. The algorithm works as follows:

1. Choose k initial cluster centers (randomly or by some other method).
2. Assign each data point to the nearest cluster center.
3. Recalculate the cluster centers as the mean of all the data points assigned to that cluster.
4. Repeat steps 2 and 3 until the cluster centers stop moving or a maximum number of iterations is reached.

The main advantage of the k-means algorithm is its simplicity and efficiency. It is easy to implement and computationally inexpensive. However, it has some disadvantages, including the sensitivity to the initial guess of cluster centers and the tendency to get stuck in local optima if the initial guess is not good.