Zhang Hengrui, German Yuliya Olegovna

K-MEANS CLUSTERING ALGORITHM AND IMPROVEMENT METHODS

This article introduces the K-means clustering algorithm as well as several improved K-means methods, including: K-means++, Incremental K-means and Kernel K-means, and describes application scenarios for the K- means algorithm.

INTRODUCTION

K-means Clustering is an Unsupervised Machine Learning algorithm, which groups the unlabeled dataset into different clusters.The main idea at the heart of the K-means algorithm is to divide the data into K separate clusters so that the distance of the data points within each cluster is as small as possible and the distance between the clusters is as large as possible.

Although the K-means algorithm is a simple and easy to understand, computationally efficient, and scalable, but it also has some weaknesses, such as: the need to pre-define the value of k, the results of the algorithm could be affected by the choice of the initial center of mass, and it is susceptible to the influence of noise and outlier points, and so on.This paper describes and compares the K- means algorithm and its improvement methods.

I. K-MEANS CLUSTERING ALGORITHM

In this section the fundamentals of the K-means algorithm will be described. Given a dataset and a constant k, the clustering problem is to divide the data into k subsets such that each subset performs "well"under some measure.[1]This is achieved in the following ways:

1. Assume that the input sample set $D = \{x_1, x_2, ...x_m\}$, and the sample set is divided into k subsets with a maximum number of iterations N. The output clusters are $C = \{C_1.C_2, ...C_k\}$.

2. A random selection of k samples from the dataset D is used as the initial k center of mass vectors: $\{\mu_1, \mu_2, ...\mu_k\}$.

3. In each iteration, for n = 1, 2, ...N. initialize C to $C_t = \emptyset$. For $i = 1, 2, ...m$,the distance between the sample x_i and each center of mass vector is calculated as d_{ij}, and x_i are added to the cluster where the center of mass vector with the smallest distance from x_i is located. Then for $j = 1, 2, ...k$, the center-of-mass is recalculated for all sample points in C_j. If the mass centers in all the clusters are unchanged (none of the k mass centers are changed), then output the partitioned clusters $C = \{C_1.C_2, ...C_k\}$, Otherwise, continue iterating until the iteration limit is reached.

II. K-MEANS++ ALGORITHM

It is mentioned in the k-means clustering algorithm that the choice of the location of the k initialized centers of mass has a great impact on both the final clustering result and the running time. If, by chance, one (or most) of the cluster centers are initialized to the same cluster, the clustering algorithm will largely fail to converge to the global minimum, which means that when the cluster centers are initialized in the wrong places, the clustering results will be seriously erroneous, and therefore the k centers need to be chosen appropriately. The K-means++ algorithm is an optimization of the K-means method of randomly initializing the centers of mass.

In simple terms, in the K-means++ algorithm, the cluster centers will be selected one by one, and the further away from the other cluster centers, the more possible the sample points will be selected as the next cluster center. This is achieved in the following ways:

1. From the dataset $D = \{x_1, x_2, ...x_m\}$ a sample point is randomly selected as the first initial clustering center C_i;

2. The shortest distance between each sample and the currently existing cluster center is calculated, denoted by $d(x)$;

3. Choose the next center C_i, and selecting $C_i = x' \in D$ with probability $P(x) = d(x')^2 / \sum_{x \in D} d(x)^2$;

4. Repeat step 2 and step 3 and until we have chosen a total of k centers.[2]

III. INCREMENTAL K-MEANS METHOD

Incremental K-means is an improved algorithm for large-scale data sets. Unlike the traditional K-means algorithm, Incremental K-means deals with one data point at a time, constantly updating the center of mass, instead of dealing with the entire dataset at once. This method is suitable for distributed computing and large-scale datasets, and can greatly improve computational efficiency.

The central concept of the proposed method is, in each iteration, it increments the seed values by one until it reaches k number of seeds. Initially, the first seed is selected as a mean data object of a given dataset D. In each iteration, the i^{th} center

i.e. ($i \in 2, 3, ..., k$) is selected from maximum sum of squared error($SSE_{partial}$) of a cluster c_i.The i^{th} center is a maximum distance from the data object and the mass of maximum $SSE_{partial}$ cluster. Similarly, repeat the iteration until it reaches k number of seeds. Finally, we formed k desired number of clusters and also minimized the SSE_{total} error.[3]This is achieved in the following ways:

1. From the dataset $D = \{x_1, x_2, ...x_m\}$ find the mean data object as $c_i(c_1)$ and add it to mass list C.

2. Data objects in D are assigned Cluster id starting from 1 to i based on the mass in C. At any point of time C consists of i number of mass.

3.In each iteration, for $i = 1, 2, ...k$, The value c_p with the largest $SSE_{partial}$ is computed as the center of mass of the cluster.Then find the sample with the largest distance from c_p as c_i and add it to mass list C, repeat step 2, mass are recomputed based on the cluster ids assigned to objects.

4.Finally, obtain k -number of clusters and also minimize the error in terms of SSE_{total}.

IV. KERNEL K-MEANS METHOD

K-means clustering algorithm solves problems that are linearly divisible, if we use K-means algorithm on non-linearly divisible data, the final result may be very different from what we want. Therefore, the Kernel K-means algorithm was born. Kernel k-means clustering is a powerful tool for unsupervised learning on non-linearly differentiable data.[4]

Kernel K-means is a K-means algorithm based on the kernel method. The kernel method makes data that would otherwise be indivisible in a low-dimensional space linearly divisible in a high-dimensional space by mapping the data to a high-dimensional feature space. This is achieved in the following ways:

1. choose the appropriate kernel function and parameters.

2. Map the dataset to a high-dimensional feature space.

3. Execute the K-means algorithm in the high-dimensional feature space.

4. Project the clustering results back to the original data space.

The kernel functions include, but are not limited to, Gaussian Kernel, Polynomial Kernel, Linear Kernel, Exponential Kernel, Cauchy Kernel, and so on.[5]In practice, we can also design different kernel functions according to our own needs.

SUMMARY

K-means algorithm, as one of the most classical clustering algorithms, was proposed earlier and is widely used in various fields today. Such as:

1.Image segmentation: clustering pixels in an image into K clusters enables image segmentation and simplification.

2. Document clustering: Clustering documents according to content similarity helps in document categorization, information retrieval and recommender systems.

3. Customer Segmentation: Clustering customers according to purchasing behavior, interests and hobbies and other characteristics helps enterprises to develop personalized marketing strategies for different groups.

4. Anomaly Detection: Through clustering, outliers or anomalies in the data can be found, and then anomaly detection or data cleaning. [6]

Although K-means has many advantages and is widely used, it also has many disadvantages. Therefore the development has extended many variants to solve the various problems encountered in the practical application of K- means. In different data samples, choosing the appropriate variant of the K-means method can achieve twice the result with half the effort.

References

1. Pelleg, D., and Moore, A. (1999). Accelerating exact k-means algorithms with geometric reasoning. In Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 277-281). Association for Computing Machinery, New York, NY, USA.
2. Arthur, D., and Vassilvitskii, S. (2007). k-means++: The advantages of careful seeding. Soda. Vol. 7.
3. Prasad, R.K., Sarmah, R., Chakraborty, S. (2019). Incremental k-Means Method. In: Deka, B., Maji, P., Mitra, S., Bhattacharyya, D., Bora, P., Pal, S. (eds) Pattern Recognition and Machine Intelligence. PReMI 2019. Lecture Notes in Computer Science(), vol 11941. Springer, Cham.
4. Paul, D., Chakraborty, S., Das, S., Xu, J. (2020). Kernel k-Means, By All Means: Algorithms and Strong Consistency. Retrieved from arXiv:2011.06461 [stat.ML].
5. Mateus Maia, and Anderson Ara. (2018). The Kernel Trick and clustering: the kernel k-means method.
6. Wang, B., Li, H. F., and Liang, Q. Q. (2021). (2021). K-means algorithm application status and research development trend. Computer Programming Skills and Maintenance (12), 2.

Zhang Hengrui, master's student of the Faculty of Information Technology and Control of BSUIR,15058556211@163.com

German Yuliya Olegovna, PhD, Associate Professor of Information Technologies in Automated Systems Department, Faculty of Information Technology and Control of BSUIR, jgerman@bsuir.by