Technology Reivew: K-Means algorithm and its variants

Yuheng Zhang yuhengz2@illinois.edu

November 16, 2020

1 Introduction

The goal of clustering task is to group a set of instances such as instances in the same group are more similar compared with instances in different groups. Clusters can help reveal structure information in the dataset which is difficult to see and clustering algorithms have wide applications in data mining, information retrieval and computer vision. In this review, I will introduce one of the famous clustering algorithms k-Means[3] and its variants including k-Means++[2], hierarchical k-Means[1] and K-Medoids[4].

2 K-Means

K-Means[3] is a centroid-based clustering technique. Assume now we already know the center of each cluster, then it is very easy to tell that each instance will belong to the cluster with minimum distance. Similarly, if the cluster id each instance belongs to is given, then the cluster center is very easy to calculate, it equals to the mean value of instances in it. Let x_i denote the feature vector of instance i, c_j denote the center of cluster j, $\delta_{i,j}$ denote if instance i belongs to cluster j. The goal of K-Means is to minimize the sum of squared distances from instances to cluster centers which could be written as follows:

$$E = \sum_{i,j} \delta_{i,j}[(x_i - c_j)^T(x_i - c_j)] \qquad [2.1]$$

Notice that δ only has two values 0 and 1, and there is only one non-zero $\delta_{i,j}$ for the ith instance. To minimize E, k-Means algorithm employs an iterative optimization strategy. At the beginning of the algorithm, we arbitrarily choose k feature vectors to act as initial cluster centers. Then the following process is repeated until cluster results change very little:

1. Allocate each sample to the cluster whose center is nearest.
2. Supply empty clusters with a sample chosen at random to ensure every cluster has at least one sample.
3. Calculate the new cluster centers with the mean value of the samples in the clusters.

How to choose the value of k is a very important issue in k-Means since usually we do not know how many clusters should be in the dataset. An empirical strategy is to cluster with different values of k and look at the value of E for each. One intuitive method is to choose k

which yields the smallest value of E. However when k goes up, E always goes down, when k equals to the total number of samples, E can always achieve zero value, thus the smallest value of E is meaningless. Instead, we can plot E as a curve of k and look at the "knee" of the curve. Empirically, this method often produces good performance.

3 K-Means++

The initial cluster centers play an important role in k-Means, different values of cluster centers usually yield different performance. The original k-Means method randomly choose k feature vectors of samples as the initial centers and an unlucky choice might result in a poor performance. [2] proposes an initialization strategy with quite good theoretical properties. At first, we choose a x randomly as the first initial cluster center, and then we compute $d_i^2(x)$ as the squared distance between the ith sample and the first center. Now, we choose the other cluster centers drawing from the following probability distribution:

$$\frac{d_i^2(x)}{\sum_u d_u^2(x)} \tag{3.1}$$

4 K-Medoids

Since k-Means algorithm uses the mean value of feature vectors to update the cluster centers, it is very sensitive to outliers. When outliers are assigned to the clusters, they can change the mean value of the cluster a lot. To diminish such sensitivity to outliers, [4] minimizes the following absolute distance:

$$E = \sum_{i,j} \delta_{i,j} dist(x_i, o_j) \tag{4.1}$$

where $dist(x_i, o_j)$ is the euclidean distance between the ith sample and the representative sample for the jth cluster. In k-Medoids, we use actual samples to represent the clusters, one representative sample per cluster. When updating cluster centers, we choose the sample which minimizes distance E, and the rest of the algorithm is the same as k-Means.

5 Hierarchical K-Means

In many scenarios, we need to deal with datasets consisting of millions of instances which is very difficult in limited time. Hierarchical k-Means[1] is an effective strategy to solve that. First we sample some instances and cluster them with a small value of k. Then we allocate each instance to the closest cluster center and cluster the instances in each cluster again with k-means. Repeating the process, we could yield a hierarchical structure of clusters which seems like a multi-level tree.

6 Conclusion

In this report, I review one of the most important cluster algorithm k-Means and its variants which improve it. K-Means exploits an iterative strategy to optimize the squared distance E, however, it is very sensitive to outliers. To diminish the sensitivity, [4] employ the absolute distance and choose the most representative sample as the cluster center. Also researchers notice the performance of k-Means depends on the initial cluster centers a lot, thus a better initialization strategy [2] is proposed to improve that. To support clustering with large dataset, [1] uses a hierarchical strategy which could yield a multi-level tree of clusters. K-Means algorithm has wide applications in data mining and machine learning, we can adopt it and its variants in our own scenarios according to our need.

References

[1] Kohei Arai and Ali Ridho Barakbah. Hierarchical k-means: an algorithm for centroids initialization for k-means. *Reports of the Faculty of Science and Engineering*, 36(1):25–31, 2007.

[2] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. Technical report, Stanford, 2006.

[3] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

[4] Leonard KAUFMAN Peter J RDUSSEEUN and PJ KAUFMAN. Clustering by means of medoids. 1987.