K-Means: Variance Based Initialization And A Comparative Study

Masum Billal

November 12, 2019

Abstract

K-Means is a popular clustering algorithm that reduces sum of minimum distances from data points to the centers. In this paper, we introduce a new way of initializing the centers. We also show comparison of different initialization for K-Means centers in terms of inertia, convergence speed and variance.

1 Introduction

Widely regarded as the most popular clustering techniques, K-Means remains a humble interesting topic in machine learning as well as computational geometry. Roughly the problem is: given a set of points $\mathbb{X}$ in $\mathbb{R}^d$. Find a set of centers $\mathcal{C}$ such that the function *inertia*

$$\mathcal{I} = \sum_{x \in \mathbb{X}} \min_{c \in \mathcal{C}} (||c - x||^2)$$

is minimum.

Default K-Means algorithm starts with random centers and then converge based on minimum distances of the centers from the data points. New centers are calculated based on the centroid. This is known as Lloyd's algorithm[2]. This is done until no more change is possible. K-Means++ takes it one step further by choosing the initial centers carefully. Only the first center is chosen at random. Then the rest of the $k-1$ centers are chosen using D^2 *weighting* as Arthur and Vassilvitskii call it[1]. At first it seems very surprising that no one really wants to work on improving on the centers. However, that is easily explained with the following.

$$\begin{aligned} E &= \sum_{i=1}^{n} ||\mathbf{x}_i - \mathbf{x}||^2 \\ \implies E &= \sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{x}) I (\mathbf{x}_i - \mathbf{x})^T \\ \implies \frac{\partial E}{\partial x} &= -2 \sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{x}) \\ \frac{\partial E}{\partial x} &= 0 \\ \iff \mathbf{x} &= \frac{\sum_{i=1}^{n} \mathbf{x}_i}{n} \end{aligned}$$

where I is the identity matrix of the same rank as $\mathbf{x}$. This pretty much shows why taking the average of all the coordinates in a cluster minimizes the sum of squared distances.

There has been multiple surveys on KMeans in the literature. Probably the most relevant work in this regard is done by Celebi, Kingravi and Vela [5]. However, most of the algorithms used for comparison are practically not used very much, as also noted by the authors themselves that `k-means++` and its greedy version work better than most. They also mention that probabilistic algorithms work better than deterministic ones. Moreover, it seems the first of the most influential center initialization algorithm [4] was not considered for their experiments. There is no mention of Ostrovsky's algorithm in their paper whatsoever.

The other reason why we need to take a second look at the results in [1] is that, experiments suggest that some standard procedures were not performed for comparison in [1]. Moreover, even without following those procedures, our results seem to differ from theirs. We would also like to point out that to our knowledge

no surveys were done using dimension reduction algorithm prior to running the experiments. This is a very important step if we are to get a meaningful clustering out of K-Means algorithm. Specially, if we use PCA (principal component analysis) before running a clustering algorithm, this helps us redefine the variables into linearly independent variables since PCA decomposes the existing variables into orthogonal[1] ones. For our experiment, we have used PCA on every dataset before running cluster algorithm.

While experimenting on such algorithms, it is of utmost importance to run the same experiment more than once under the same parameters and conditions. For example, assume that we want to compare `k-means++` and Forgy's algorithm [6] for $k = 5$ clusters. We should run this experiment at least m times where $m > 1$ in order to eliminate bias and account for randomness. It is commendable that [5] follows the literature in using $m = 100$ as mentioned in the paper. In this paper, we use $m = 100$ as well. We will also take variance of the distances between the centers into account. The reason behind doing so is not just to make sure that the cluster centers are not too close to each other, but also to keep the clusters as balanced as possible. It is surprising that variance of these distances is not particularly a standard metric that is usually used for K-Means type algorithm.

To summarize, we will be mostly looking at these variations of initial seeding of centers: Lloyd, Ostrovsky, `k-means++` and a variance based seeding. Then we will be comparing inertia, variance and convergence speed for a comparative study among these algorithms.

2 Experiment Setup

We ensure that all algorithms are run under the same conditions. All of them share the same environment and no special optimizations were made for any particular algorithm. Only CPU was used to determine the values we are interested in and no parallelism mechanism was in place for speeding up the process. This way, we can get an idea about the raw performance metrics of the algorithms involved.

Python is used as the programming language to write necessary codes. Some common auxiliary packages such as *scipy, scikit-learn, numpy* etc are used to help with the code. All the algorithms are simply different methods of the same class, so they share the same fitting and prediction function. Only the initialization differs for different algorithm. It should be mentioned that even though some packages have native support for K-Means implementation, we did not use them to run the experiments. The reason is that not all the algorithms we want to test are available in those packages. Therefore, in order to ensure same environment and optimizations for every algorithm, we wrote every algorithm ourselves so that we could be sure they are done in the same setting.

The datasets used are some popular ones: Iris dataset, Mall customers dataset, Airline clustering dataset, Wine testing dataset. We did not duel too much on using too many datasets. The number of datasets does not really mean very much for K-Means. It is the quality of clustering we are interested in, not the quality of dataset itself. And a good clustering algorithm should be able to handle all type of datasets. For a fixed number of clusters k, we choose a dataset we want to run the experiment on. After loading and cleaning the data, we use PCA with number of components set to either 2 or 3. Then we run the experiment on the modified dataset. For every dataset, we repeat the experiment a total of 100 times as mentioned before. Finally, we use the average and minimum values of inertia, number of steps taken for convergence and variance. We would like to mention that for convergence, we checked if the sum of distance between new centers and old centers is less than a certain value ϵ where ϵ is a very small real number. For our experiment, we chose $\epsilon = 10^{-15}$ which practically says that the change in the position of the centers is negligible.

References

[1] David Arthur and Sergei Vassilvitskii. `k-means++`: The Advantages of Careful Seeding. Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms. Society for Industrial and Applied Mathematics Philadelphia, PA, USA. pp. 1027–1035.

[2] Stuart P. Lloyd. Least squares quantization in pcm. IEEE Transactions on Information Theory, 28(2) : 129–136, 1982.

[3] M. E. Dyer. A simple heuristic for the p-center problem. Operations Research Letters, Volume 3, February 1985, pp. 285 − 288.

[1] It is easy to prove that orthogonal vectors are linearly independent.

[4] Rafail Ostrovsky, Yuval Rabani, Leonard J. Schulman, Chaitanya Swamy. The Effectiveness of Lloyd-Type Methods for the k-Means Problem. Proceedings of the 47th Annual Symposium on Foundations of Computer Science. 2006.

[5] M. Emre Celebi, Hassan A. Kingravi, Patricio A. Vela. A Comparative Study of Efficient Initialization Methods for the K-Means Clustering Algorithm. Expert Systems with Applications, 40(1) : 200–210, 2013.

[6] E. Forgy. Cluster analysis of multivariate data: Efficiency versus interpretability of classification. Biometrics, $21, 768 - 780, 1965$.