A Deterministic Method for Initializing K-means Clustering

Ting Su
Northeastern University
Boston, MA 02115
tsu@ece.neu.edu

Jennifer Dy
Northeastern University
Boston, MA 02115
jdy@ece.neu.edu

Abstract

The performance of K-means clustering depends on the initial guess of partition. In this paper, we motivate theoretically and experimentally the use of a deterministic divisive hierarchical method, which we refer to as PCA-Part (Principal Components Analysis Partitioning) for initialization.

The criterion that K-means clustering minimizes is the SSE (sum-squared-error) criterion. The first principal direction (the eigenvector corresponding to the largest eigenvalue of the covariance matrix) is the direction which contributes the largest SSE. Hence, a good candidate direction to project a cluster for splitting is, then, the first principal direction. This is the basis for PCA-Part initialization method.

Our experiments reveal that generally PCA-Part leads K-means to generate clusters with SSE values close to the minimum SSE values obtained by one hundred random start runs. In addition, this deterministic initialization method often leads K-means to faster convergence (less iterations) compared to random methods. Furthermore, we also theoretically show and confirm experimentally on synthetic data when PCA-Part may fail.

1. Introduction

Cluster analysis is the unsupervised classification of patterns into similar groupings. It is useful in various applications. One of the most popular clustering algorithms is the K-means algorithm. We denote our data set as a matrix $X = [x_1, \ldots, x_n]^t \in \Re^{n \times d}$, Each row of X, x_i, represents a d-dimensional instance. The goal of the K-means clustering is to partition X into K exclusive clusters $\{C_1, \ldots, C_K\}$. The most widely used criterion for the K-means algorithm is the SSE [5]: $SSE = \sum_{j=1}^{K} \sum_{x_i \in C_j} \|x_i - \mu_j\|^2$, where $\mu_j = \frac{1}{n_j} \sum_{x_i \in C_j} x_i$ denotes the mean of cluster C_j and n_j denotes the number of instances in C_j.

K-means starts with initial K centroids (means), then it assigns each data point to the nearest centroid, updates the cluster centroids, and repeats the process until the K centroids do not change. The K-means algorithm is a greedy algorithm for minimizing SSE, hence, it may not converge to the global optimum. The performance of K-means strongly depends on the initial guess of partition.

Several random initialization methods for K-means have been developed. Two classical methods are random seed and random partition. Random seed randomly selects K instances (seed points), and assigns each of the other instances to the cluster with the nearest seed point. Random partition assigns each data instance into one of the K clusters randomly. To escape from getting stuck at a local minimum, one can apply r random starts. Specifically, one can perform one of the above methods to initialize K-means, repeat the process r times, and select the final clustering with the minimum SSE from the r runs. [3] introduced a sub-sampling version of random restart to cope with large data sets. The main problem with random methods is that do not guarantee obtaining the optimal solution unless we allow r to be very large (thus, increasing the time complexity). A detailed survey of initialization methods is available in [8].

In this paper, we motivate a deterministic initialization method for K-means: PCA (principal component analysis) based divisive hierarchical approach, we refer to as PCA-Part for short. We show why PCA-Part is a good method, and also show when it may fail. In Section 2, we describe the motivation for PCA-Part. We, then, report our experimental results in Section 3. Finally, in Section 4 we draw conclusions and suggest avenues for future research.

2. PCA-Part Initialization Method

Good initial centroids are seeds that are evenly distributed [1]. [1] proposed sorting data instances on a single variable then performing the initial partition. This partitions data only in one dimension. An alternative method is to partition the sample space hierarchically. Starting with one cluster, cut it into two. Pick the next cluster to partition, and so on. PCA-Part uses the latter approach.

Which direction should we split the chosen clus-

ter? Let μ be the mean for a given cluster. The SSE of the data within this cluster C is: $SSE_{old} = \sum_{x_i \in C} \|x_i - \mu\|^2$. After dividing this cluster into two clusters, C_1 with mean μ_1 and C_2 with mean μ_2, the new SSE is: $SSE_{new} = \sum_{x_i \in C_1} \|x_i - \mu_1\|^2 + \sum_{x_i \in C_2} \|x_i - \mu_2\|^2$. Each d-dimensional vector x_i can be represented by a weighted sum of d linearly independent orthonormal basis vectors, $\Phi = [\phi_1, \ldots, \phi_d]$: $x_i = \sum_{s=1}^{d} y_{is}\phi_s$. Similarly, the mean μ_j can be represented as: $\mu_j = \sum_{s=1}^{d} \alpha_{js}\phi_s$. We restate our question as, which direction ϕ_p should we project our data for splitting? Assuming that the old mean μ and the new means, μ_1 and μ_2 lie on the axis chosen for projecting, the ϕ_p which minimizes SSE_{new} is the ϕ_p that maximizes

$$\sum_{x_i \in C} (y_{ip}\phi_p - \alpha_p\phi_p)^2 \quad - \quad \sum_{x_i \in C_1} (y_{ip}\phi_p - \alpha_{1p}\phi_p)^2 \quad - \quad \sum_{x_i \in C_2} (y_{ip}\phi_p - \alpha_{2p}\phi_p)^2 \quad (1)$$

where y_{ip}, α_p, α_{1p}, and α_{2p} are the projected values of x_i, μ, μ_1, and μ_2 on ϕ_p, respectively. Refer to [8] for the proof.

Equation 1 is SSE_{old} due to the direction ϕ_p minus SSE_{new} due to the direction ϕ_p. To find this optimal direction, we need to know the means, μ_1 and μ_2. This leads us back to a $K = 2$ clustering problem. To avoid solving a clustering problem, PCA-Part resorts to a suboptimal direction which assumes that the SSE_{new} due to the candidate direction, $\sum_{x_i \in C_1}(y_{ip}\phi_p - \alpha_{1p}\phi_p)^2 + \sum_{x_i \in C_2}(y_{ip}\phi_p - \alpha_{2p}\phi_p)^2$, is proportional to the SSE_{old} due to this direction, $\sum_{x_i \in C}(y_{ip}\phi_p - \alpha_p\phi_p)^2$, and this proportionality constant, a, is the same for all directions and $0 \leq a \leq 1$. The optimization problem is now simplified to finding the direction, ϕ_p, that maximizes $\sum_{y_i \in C}(y_{ip}\phi_p - \alpha_p\phi_p)^2$. Thus, PCA-Part chooses ϕ_p to be the direction which contributes to the largest SSE. The first principal direction is the one which contributes to the largest SSE.

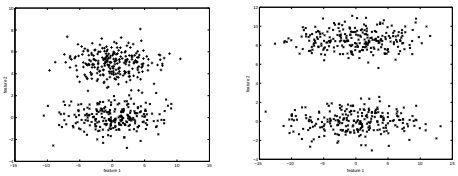
How do we partition the cluster in this principal direction? We choose to partition the data at the mean, so that the center of gravity between the two halves will be balanced at the mean. **Which cluster should we split?** Since SSE is the criterion K-means tries to minimize, we decide to split the cluster with the largest with-in cluster $SSE_j = \sum_{x_i \in C_j} \|x_i - \mu_j\|^2$.

We now give a summary for PCA-Part. Starting from a single cluster, divide it into two sub-clusters, choose the sub-cluster with the largest within-cluster SSE_j as the next cluster to partition, repeat the process until K clusters are produced. At each split stage, for the selected cluster C_j, we first project $x_i \in C_j$ to the first principal direction of $x_i \in C_j$, we then divide C_j into two sub-clusters C_{j1} and C_{j2} according to the rule: For any x_i, if y_i (the projected value of x_i)$\leq$ α_j (the projected mean), assign x_i to C_{j1}, otherwise, assign x_i to C_{j2}.

PCA-Part is similar to the "PDDP" algorithm [2], while "PDDP" is a complete hierarchical clustering. They also differ in the way they select which cluster to split next. "PDDP" selects the sub-cluster with the largest Frobenius norm of the covariance matrix to partition.

3. Experiments

In this section, we compare the performance of PCA-Part with the classical initialization methods (random seed and random partition) based on the following criteria: 1. Quality: We quantify the quality of the clustering using SSE. 2. Stability : We measure the stability of the random initialization methods using the standard deviation of SSE, σ_{SSE} for r runs (in our experiments, r=100). Obviously $\sigma_{SSE} = 0$ for PCA-Part. 3. Speed: We evaluate the speed of convergence through the number of iterations needed for K-means to converge.



(a) (b)

Figure 1. (a) Synthetic data 1. (b) Synthetic data 2.

DATA SET	# OF SAMPLES	# OF FEATURES	# OF CLUSTERS
PENDIGITS	10992	16	10
SEGMENTATION	2310	19	7
LETTER	20000	16	26

Table 1. Real data set descriptions

We compare the different initialization schemes on several data sets. Due to space limitation, here we only show results for 5 data sets: two two-dimensional synthetic data sets as shown in Figures 1, and three real data sets from the UCI Machine Learning Repository [7](see table 1 for a summary). The complete experimental results are presented in [8]. Synthetic data 1 presents a case where the first principal direction maximizes Equation 1. Data 2 has a similar configuration as Data 1 while the second principal direction

INIT. METHOD		SYN. 1	SYN. 2	PENDIGITS	SEGMENTATION	LETTER
RAND.	MAX	6819.39	14922.4	5.44E+7	2.15E+7	636249
SEED	μ_{SSE}	6185.06 ± 271.82	13684.1 ± 507.29	$4.52E+7 \pm 1.88E+6$	$1.49E+7 \pm 2.14E+6$	620258 ± 4151.52
	MIN	6067.99	12971.1	4.37E+7	1.35E+7	611567
RAND.	MAX	6925.2	14921.8	5.47E+7	2.15E+7	646874
PART.	μ_{SSE}	6253.59 ± 323.61	13681.8 ± 553.95	$5.09E+7 \pm 1.23E+6$	$1.44E+7 \pm 1.49E+6$	623269 ± 6416.11
	MIN	6067.99	12971.1	4.93E+7	1.35E+7	611008
PCA-PART	μ_{SSE}	6068.07 ± 0	14029.6 ± 0	$5.00E+7 \pm 0$	$1.38E+7 \pm 0$	617846 ± 0

Table 2. The SSE values for the Data.

INIT. METHOD		SYN. 1	SYN. 2	PENDIGITS	SEGMENTATION	LETTER
RAND. SEED	μ_{ite}	10.54 ± 3.82	9.41 ± 7.24	32.25 ± 12.49	23.86 ± 11.04	85.22 ± 33.21
RAND. PART.	μ_{ite}	8.85 ± 3.80	10.30 ± 8.60	30.29 ± 10.28	24.70 ± 10.60	84.94 ± 29.84
PCA-PART	μ_{ite}	3 ± 0	2 ± 0	15 ± 0	14 ± 0	85 ± 0

Table 3. The number of iterations for the Data.

maximizes Equation 1. Data set 2 is difficult for PCA-Part because it violates the assumption of PCA-Part.

Table 2 lists μ_{SSE}, σ_{SSE}, the minimum SSE and the maximum SSE returned by K-means when each initialization method is used. We observe that PCA-Part obtains smaller SSE values than μ_{SSE} obtained from random partition and random seed for all data sets except for synthetic data 2 and pen digits data. In addition, usually PCA-Part leads to SSE values close to the minimum SSE values obtained from the random methods. Moreover, the worst case reached by the K-means algorithm when initialized with the random methods may be far from the best case, confirming the need for stable initialization methods.

Table 3 lists the average and standard deviation of the number of iterations that K-means needs to reach convergence for different initialization methods. This table shows that in all cases except the letter data, PCA-Part lets the K-means algorithm run less iterations to converge than the random methods. One can apply the power method [6] for computing the first principal direction only to save time. In addition, here we only present the average iteration numbers for one K-means run when random method are used. In practice, random methods are re-started around ten or more times to escape from local minima. Therefore, generally using PCA-Part to initialize the K-means algorithm requires less computation than the random methods.

4. Conclusions

The performance of K-means depends on the initial condition. We theoretically and experimentally analyze the motivation behind PCA-Part and show its strengths and limitations. Our results are encouraging. It presents some promise in initializing at intelligent starting points for the K-means algorithm, instead of just random start. This work suggests research directions, such as exploring other ways of partitioning the sample space (e.g., "pie"-slices). When time complexity is not crucial, one may apply different deterministic intelligent restarts (capturing different possible data configuration scenarios), or combine random and deterministic restarts for initializing K-means. This way, we are assured that at least one of the K-means runs would lead to good clustering result in terms of SSE. In addition, one may also explore the effectiveness of PCA-Part for initializing other clustering methods such as mixtures of Gaussians with the Expectation-Maximization algorithm [4] .

References

[1] M. R. Anderberg. *Cluster Analysis for Applications*. Academic Press, NewYork, NY, 1973.

[2] D. Boley. Principal direction divisive partitioning. *Data Mining and Knowledge Discovery*, 2(4):325–344, 1998.

[3] P. S. Bradley and U. M. Fayyad. Refining initial points for K-Means clustering. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 91–99, San Francisco, CA, 1998. Morgan Kaufmann.

[4] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal statistical Society, Series B*, 39(1):1–38, 1977.

[5] O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience, NewYork, NY., 2000.

[6] G. Golub and C. V. Loan. *Matrix computations*. John Hopkins University Press, 1996.

[7] C. J. Merz, P. Murphy, and D. Aha. UCI repository of machine learning databases, 1996.

[8] T. Su. Another look at non-random methods for initializing K-means clustering. Master Project, June 2003. Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA.