

# Initialization of Cluster Refinement Algorithms: A Review and Comparative Study

Ji He<sup>†</sup>, Man Lan<sup>‡</sup>, Chew-Lim Tan<sup>‡</sup>, Sam-Yuan Sung<sup>‡</sup>, and Hwee-Boon Low<sup>#</sup>

<sup>†‡</sup>School of Computing, National University of Singapore,  
3 Science Drive 2, Singapore 117543

<sup>†</sup>mail@jihe.net; <sup>‡</sup>{lanman, tancl, ssung}@comp.nus.edu.sg

<sup>#</sup>Institute for Infocomm Research,  
21 Heng Mui Keng Terrace, Singapore 119613  
hweeboon@i2r.a-star.edu.sg

**Abstract**— Various iterative refinement clustering methods are dependent on the initial state of the model and are capable of obtaining one of their local optima only. Since the task of identifying the global optimization is NP-hard, the study of the initialization method towards a sub-optimization is of great value. This paper reviews the various cluster initialization methods in the literature by categorizing them into three major families, namely random sampling methods, distance optimization methods, and density estimation methods. In addition, using a set of quantitative measures, we assess their performance on a number of synthetic and real-life data sets. Our controlled benchmark identifies two distance optimization methods, namely SCS and KKZ, as complements of the K-Means learning characteristics towards a better cluster separation in the output solution.

## I. INTRODUCTION

Clustering refers to the task of partitioning unlabelled data into meaningful groups (clusters), in response to a pre-defined pattern approximation measure. It is a useful approach in data mining processes for identifying hidden patterns and revealing underlying knowledge from large data collections. The application areas of clustering, to name a few, include image segmentation, information retrieval, document classification, associate rule mining, web usage tracking, and transaction analysis.

In essence, clustering involves an unsupervised learning for the regression of a mixture probability model  $M(K, w, C, Y)$ , of which the data points are approximated as  $K$  sub-groupings (patterns)  $C_i, i = 1, \dots, K$ . The probability of a data point  $x$  generated by the model is

$$P(x) = \sum_{i=1}^K w_i \cdot P(x|C_i, Y_i(x)), \quad (1)$$

where  $w_i$  is the *mixture weight* and  $Y_i(x) \equiv x \rightarrow C_i$  is a mapping from the data point  $x$  to the sub-grouping  $C_i$ . In the literature, *iterative refinement* is widely adopted by various unsupervised learning algorithms. A general iterative refinement process can be summarized as follows [3].

□

**Initialization:** Initialize the parameters of the current model.

**Refinement:** Repeat

- Generate the cluster membership assignments for all/each training instances, based on the current model;

- Refine the model parameters based on the current cluster membership assignments;  
until the model converges, in the sense that in a refining iteration the changes made on the parameters are below a threshold.

■

An important underlying assumption in this process is that, the model to be used *before* each refining iteration is correct. This makes the initialization of the model deterministic to the clustering solution [3]. That is, the process usually can obtain only one of its local optima, which is sensitive to the model's initial state. Since the problem of obtaining a globally optimal initial state has been shown to be NP-hard [9], the study on the initialization methods towards a sub-optimal clustering solution hence is more practical, and is of great value for the clustering research.

While there exist various initialization methods for different clustering methods, there are relatively few comparative studies of this important issue in the literature, due to the variety in the clustering methodology. This paper aims to fill this gap by reviewing various approaches for initializing the cluster distribution  $C$ , as it is most commonly required by various clustering methods.

For this purpose, our study focuses on the initialization of the *cluster refinement* methods only. We consider cluster refinement a sub-task of iterative refinement clustering, in the sense that the refining iterations only fine-tune the cluster distribution  $C$ , while the remaining parameters, i.e.  $K, w$ , and  $Y$  remain unchanged. Specifically, our benchmark adopted the batch K-Means clustering algorithm [21]. K-Means can be understood as a particular EM model [5] with the simplified settings  $w_i = 1$  for  $i = 1, \dots, K$ . Its cluster assignment  $Y$  is the naive nearest neighbor search, which is a disjoint many-to-one mapping. With such, the initial cluster distribution  $C$  can be simply represented with a set of so-called *seed clusters*  $c_i, i = 1, \dots, K$ . The reasons we choose K-Means in our experiments are: (1) while simple, K-Means has shown satisfactory clustering results in various studies in the literature, especially when the problem domain is approximately a mixture Gaussian distribution, i.e. clusters are convex in shape and there is a higher data density near the cluster centroids; and (2) the initialization methods for K-Means can be applied

to other iterative refinement clustering algorithms as well. ■

The rest of the paper is organized as follows. Section II briefly reviews the various cluster initialization methods in the literature. Section III lists related work and clarifies the scope of our study. Section IV reports our comparative experiments on five initialization methods, using various synthetic and real-life data sets. The last section summarizes our concluding remarks and suggests the future work.

## II. A REVIEW OF SEVERAL CLUSTER INITIALIZATION METHODS

We categorize the cluster initialization methods into three major families, namely random sampling methods, distance optimization methods, and density estimation methods.

### A. Random Sampling Methods

Probably being most widely adopted in the literature, random sampling methods follow a naive way to initialize the seed clusters, either using randomly selected input samples, or random parameters non-heuristically generated from the inputs.

Being one of the earliest references in the literature, Forgy in 1965 [6] adopted uniformly random input samples as the seed clusters. The method, named R-SEL in our study, is formalized below.

□**R-SEL:**

For  $i = 1, \dots, K$ , set  $c_i = x_r$  such that  $r = \text{uniRand}(1, N)$ , where  $N$  is the total number of input samples and  $\text{uniRand}(\min, \max)$  is a uniform random generator producing  $r \in [\min, \max]$ . ■

Given the inputs presented in a random order, the method used by MacQueen [17], which simply initializes the seed clusters with the first  $K$  input instances, is equivalent to the R-SEL method. Based on the statistical assumption that a randomly re-sampled subset would reflect the distribution of the original set, these two methods have shown to produce satisfactory results in prior studies, especially when  $K$  is relatively large in response to the total number of input samples  $N$ , and when the designated  $K$  approximates the “natural”/“optimal” number of clusters  $K_0$  on the input set. However, the clustering solution may be drastically affected by these methods if  $K$  is too small, in which case the assumption above is suppressed. On the other hand, when  $K$  is large, the method may produce a relatively large number of dead clusters (clusters that do not attract any input samples upon convergence). An alternative method is to initialize the seed clusters by slightly perturbing the mean of the inputs [20], [10]. Through this way, each seed cluster is given nearly even probability of being selected at the beginning of the K-Means learning. Hence the chance of dead cluster generation is lowered. The method is given name R-MEAN in our studies and formalized below.

□**R-MEAN:**

For  $i = 1, \dots, K$ , set  $c_i = \text{gaussRand}(\bar{x}, \varepsilon)$ , where  $\text{gaussRand}(m, v)$  is a Gaussian random generator with mean  $m$  and variance  $v$ ,  $\bar{x} = \frac{1}{N} \cdot \sum_{j=1}^N x_j$  is the mean of the inputs, and  $\varepsilon$  is a small constant. ■

Note the above random generator is not necessarily to be Gaussian.

### B. Distance Optimization Methods

Recognizing the characteristics of many clustering methods is to locally minimize the intra-cluster variances without optimizing the inter-cluster separation, it is a natural consideration to optimize the distances among the seed clusters beforehand towards a satisfactory inter-cluster separation in the output.

Among some early practises, the Simple Cluster Seeking (SCS) initialization method [21] is adopted in the FASTCLUS procedure, which is a K-Means variance implemented in SAS [19]. The SCS method is summarized below.

□**SCS:**

- 1) Initialize the first cluster centroid with the first input, i.e.  $c_1 = x_1$ .
- 2) For  $j = 2, \dots, N$ , if  $\|x_j - c_k\| > \rho$  for all existing seed clusters  $c_k$ , where  $\rho$  is a threshold, then add  $x_j$  as a new seed cluster. Stop when  $K$  seed clusters are initialized.
- 3) If after scanning all input samples, there are less than  $K$  seed clusters generated, then decrease  $\rho$  and repeat 1 - 2. ■

It is interesting that SCS was originally proposed as a clustering method without any refinement process. It is used here due to its simplicity and its capability of identifying distinct inputs online. SCS however is sensitive to the initial  $\rho$  value and the presentation order of the inputs.

Katsavounidis et. al [13] proposed a method that utilizes the sorted pairwise distances for initialization. The method, termed KKZ in [1], is stated as follows.

□**KKZ:**

- 1) Initialize the first seed cluster using the input with the maximal norm, i.e.  $c_1 = x_{j_1} \equiv \text{argmax}\{\|x_j\|\}$ .
- 2) For  $i = 2, \dots, K$ , each  $c_i$  is initialized in the following way: for each input sample  $x_j$ , calculate its distance to the closest seed cluster  $d_j = \min\{\|x_j - c_k\| : \text{for all existing } c_k\}$ , and set  $c_i = x_{j_i} \equiv \text{argmax}\{d_j\}$ . ■

Compared with SCS, KKZ is more user-friendly in the sense that it does not require a threshold during seed vector search. Using a naive implementation, KKZ could be computationally intensive due to the large number of distance calculations. However, Katsavounidis et. al pointed out that, since generation of new seed clusters does not affect the existing ones, the computational complexity of KKZ can be comparable to one K-Means iteration using a buffering mechanism that stores the distances between each input sample and the existing seed clusters.

### C. Density Estimation Methods

This category of initialization methods is based on the assumption that the input data follow a Gaussian mixture distribution. Hence by identifying the dense areas of the input domain, the initialized seed clusters helps the clustering method in creating compact clusters. It is interesting that the R-SEL method could be understood as the most naive density

estimation approach due to the assumption discussed above. In contrast to R-SEL, methods reviewed here are rather heuristic.

Kaufman and Rousseeuw [14] introduced a method that estimates the density through pairwise distance comparison, and initializes the seed clusters using the input samples from areas with high local density. The method, termed KR in our study, contains the steps below.

□**KR**:

- 1) Initialize  $c_1$  with the most centrally located input sample.
- 2) For  $i = 2, \dots, K$ , each  $c_i$  is initialized in the following way: for each non-selected input sample  $x_j$ , calculate its summed distance to the other non-selected input samples  $x_l$ , who are closer to  $x_j$  than to their respective nearest seed clusters, formalized as  $s_j = \sum_{l \neq j} \max(\min\{\|x_l - c_k\| : \text{for all existing } c_k\} - \|x_l - x_j\|, 0)$ , then set  $c_i = x_{J_i} \equiv \text{argmax}\{s_j\}$ .

A notable drawback of the KR method lies in its computational complexity. Given  $N$  input samples, at least  $N \cdot (N-1)$  distance calculation are required even with a buffering mechanism. This could be much more time consuming than K-Means itself when  $N$  is large. Hence on large-scale data set, it is more practical to use only a small portion of the input set for KR processing.

More recently, Al-Daoud and Roberts [1] proposed a method which combines local density approximation and random initialization. This method is summarized as follows: the whole input domain is evenly divided into  $M$  sub-spaces  $S_j, j = 1, \dots, M$ , the statistics of the input samples in each sub-space, in a simple terms of the number of samples  $N_j$ , are then collected. The number of seed clusters in each sub-space is allocated by a rounded integer  $K \cdot N_j/N$ . Consequently the seed clusters are locally initialized in each corresponding sub-space in a random manner. While this method worked fine with a relatively large  $K$  in the prior study [1], it suffers from the dependence on a carefully chosen  $M$  value. In case that  $K$  is small,  $M$  must be correspondingly small enough so that some sub-spaces are allocated with at least one seed clusters. This in turn affects the estimation of the local density. As an improvement, Al-Daoud and Roberts in the same paper [1] proposed a method that iteratively subdivides the source domain into two disjoint volumes. The summed square error between the data points to their nearest cluster centroids in each volume are estimated. Consequently the number of clusters to be assigned in each volume which minimizes the error is given by the numeric solution for a polynomial equation, which corresponding to a quadratic equation in the two-dimensional case. However, the proof on the validity of the method (specifically on the estimation of summed square error) is not given.

### III. RELATED WORK

All the initialization methods reviewed above involve the approximation of the input distribution, through either naive sampling or heuristic search. Such an approach actually harmonizes with the original objective of clustering. Naturally this paper reaches the stage to discuss an interesting consideration:

*Why not simply initialize the seed clusters of one clustering method using the output of another complementary clustering method?*

There do exist various practices based on this consideration, also referred to as “initialization methods” in some studies. Among them, Binary Splitting (BS) [16] was reviewed together with several other initialization techniques in [1] and compared with KKZ in [13]. BS sets the first cluster centroid as the mean of the inputs. Each BS iteration contains a splitting of every existing cluster into two by randomly perturbing its cluster centroid twice, followed by the recalculation of the cluster centroids based on the new cluster assignment of the inputs. The method stops when the designated number of clusters are obtained. Similarly to BS, DSBS (for Direct Search Binary Splitting), which combined binary splitting and Principal Component Analysis (PCA) [12], was referred to as an initialization method in [13] as well. An online version of K-Means (named MA) was compared with three other batch K-Means initialization methods in [18]. More recently, ART-2 neural network [4] was used to preprocess the inputs before passing them to K-Means [15].

We argue that there exists much confusion at the ill-defined term “initialization” when these techniques are studied and compared as “clustering initialization methods”. Our statement is based on two facts. Firstly, most of the above clustering methods being combined with K-Means also require an initialization process. For example, each iteration of BS essentially involves the initialization of two new seed clusters using R-MEAN out of every existing cluster followed with an iteration of batch cluster refinement, while the MA technique is essentially initialized with R-SEL and the clusters are refined online. Secondly, the combination of a decent clustering method with K-Means practically produces a new hybrid model. In other words, a comparison between such a combination and K-Means initialized with some method reviewed in Section II is rather the comparison between *two clustering systems*, than *two initialization methods for a clustering system* in a strict sense. As to the SCS method reviewed in Section II-B, while it is originally proposed as a clustering method, we still consider it a cluster initialization technique due to the fact that, in contrast to the various methods discussed in this section, SCS does not contain a cluster refinement process and hence the output solution is still primarily decided by K-Means.

There is another category of studies that deal with the refinement of the initial state of clustering methods. These studies, to name a few, include Genetic Algorithm [2], a refinement method proposed by Bradly and Fayyad [3], Tabu Search [8], and Randomized Local Search [7]. Most of these approaches involve a heuristic search for the sub-optimal *parameters* of a specific initialization method, guided with some goodness assessment function. While they are not the focus of this paper, readers shall note that the use of any initialization method reviewed in Section II can be further optimized with these approaches.

## IV. COMPARATIVE EXPERIMENTS

### A. Benchmark Methodology

Our benchmark evaluates five initialization methods, namely R-MEAN, R-SEL, SCS, KKZ, and KR on various synthetic and real-life data sets. While it is possible to obtain a sub-optimal solution for each method through a refinement approach mentioned above, our work intends to observe the general performance of each method without any other optimization process. Hence, we carried out multiple experiments on each data set, each experiment reshuffling the representation order of the inputs and using a common set of parameters for each method. The initial threshold of SCS is given through manual observation on the range of each data set. Considering the high computational complexity of KR, the seed clusters generated by KR are based on a subset of maximal 1,500 data points uniformly re-sampled from the inputs.

The K-Means clustering method, utilizing Euclidean distance measure and working under batch learning mode, is said to reach convergence when the mismatch on the cluster assignment in a learning iteration, with reference to the previous iteration, is less than a threshold 0.005. The number of iterations that K-Means used to reach convergence is reported.

Since the objectives of clustering is to reach maximal intra-cluster homogeneity and maximal inter-cluster separation, we adopted the *Cluster Compactness* ( $Cmp$ ) and the *Cluster Separation* ( $Sep$ ) indices [11] to assess the goodness of the clustering solution. The cluster compactness measure is based on the generalized definition of the *deviation* of a data set given by

$$dev(X) = \sqrt{\frac{1}{N} \sum_{i=1}^N \|x_i - \bar{x}\|^2}, \quad (2)$$

where  $N$  is the number of members in  $X$ , and  $\bar{x} = \frac{1}{N} \sum_i x_i$  is the mean of  $X$ . The cluster compactness for the output clusters  $C_1, C_2, \dots, C_K$  generated by a system is then defined as

$$Cmp = \frac{1}{K} \sum_i \frac{dev(C_i)}{dev(X)}, \quad (3)$$

where  $K$  is the number of clusters generated on the data set  $X$ ,  $dev(C_i)$  is the deviation of the cluster  $C_i$ , and  $dev(X)$  is the deviation of the data set  $X$ . The cluster separation of a clustering system's output is defined by

$$Sep = \frac{1}{K(K-1)} \sum_{i=1}^K \sum_{j=1, j \neq i}^K \exp\left(-\frac{\|c_i - c_j\|^2}{2\sigma^2}\right), \quad (4)$$

where  $c_i$  is the centroid of the cluster  $C_i$  and  $\sigma$  is a Gaussian constant. It is understandable that for both measures, a smaller value indicates a better output quality.

Based on the measures out of multiple runs, we are capable of carrying out pair-wise comparison of these initialization methods based on  $t$ -test. We consider the comparison result to be statistically significant if the  $t$ -test significance level reaches 0.1.

TABLE I

THE WORK-FLOW OF THE TWO-DIMENSIONAL APPROXIMATE MIXTURE GAUSSIAN RANDOM GENERATOR.

**Parameters:** The number of data points  $N$ , the number of clusters  $K$ , the noise level  $r \in [0, 1]$ , and the intra-cluster variance level  $v$ .

**Process:**

- 1) Preset  $K$  random cluster centroids  $c_i = (x_i, y_i)$  such that  $x_i = uniRand(0, 1)$  and  $y_i = uniRand(0, 1)$ .
- 2) Repeat
  - a) Select a random cluster id  $J = uniRand(1, K)$  from above;
  - b) Generate a random data point  $p = gaussRand(c_J, v)$ ; until  $N \cdot (1 - r)$  valid data points are obtained. A data point  $p = (x, y)$  is considered valid only  $x \in [0, 1]$  and  $y \in [0, 1]$ .
- 3) Generate  $N \cdot r$  white noises  $p_i = (x_i, y_i)$  such that  $x_i = uniRand(0, 1)$  and  $y_i = uniRand(0, 1)$ .

**Output:**  $N$  data points loosely in a mixture Gaussian distribution, with noises in a certain degree.

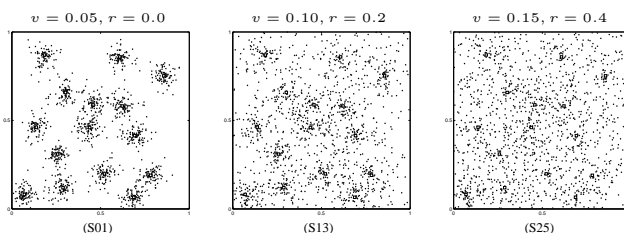


Fig. 1. The three representative synthetic data sets generated with varying variance ( $v$ ) and noise levels ( $r$ ), based a common set of 15 designated cluster centroids (large squares in each sub-figure). S01 and S25 are the cleanest and most noisy data sets respectively, while S13 is of moderate noisiness.

### B. Synthetic Data Sets

We used a random generator (Table I)<sup>1</sup> to produce various two-dimensional synthetic data sets approximately in the mixture Gaussian distribution. They are believed to be the best test bed for K-Means clustering. Based on a common set of 15 designated cluster centroids, using varying variances  $v$  and noise rates  $r$ , we obtained 25 data sets, each containing 150,000 data points. Due to page constraint, the results on three representative data sets (Figure 1) are reported. On each data set, K-Means is trained to reach convergence in response to a varying number of designated output clusters, i.e.  $K = 5, 15$  and  $60$ . Table II reports the comparison results with  $K = 15$  only due to page constraint.

While not reflected in Table II, it is worth to mention that for all initialization methods, as the number of output clusters  $K$  increases, K-Means generally used a larger number of iterations  $I$  to reach convergence, the  $Cmp$  index decreases (which indicates more compact clusters), and the  $Sep$  index increases (which indicates worse separation among clusters). As the noises in the inputs increases, K-Means generally produces worse  $Cmp$  indices. In general, the differences among the outputs of these five initialization methods are more

<sup>1</sup>The random generator, the synthetic data sets, and the complete experimental results are available via <http://www.jihe.net/research/ijcnn04>.

TABLE II

THE EXPERIMENTAL RESULTS OF THE FIVE INITIALIZATION METHODS ON THREE SYNTHETIC DATA SETS, IN TERMS OF THE NUMBER OF ITERATIONS K-MEANS USED TO REACH CONVERGENCE ( $I$ ), *Cluster Compactness* ( $Cmp$ ), AND *Cluster Separation* ( $Sep$ ).  $\sigma$  IS THE GAUSSIAN CONSTANT USED IN EQUATION 4.

	$I$	$Cmp$	$Sep$
<b>Data Set: S01, <math>\sigma = 0.5, K = 15</math></b>			
R-MEAN	11.20±1.55	0.1574±0.0155	0.6891±0.0375
R-SEL	7.70±2.26	0.1504±0.0115	0.6316±0.0189
SCS	14.70±4.62	0.1461±0.0058	0.6004±0.0172
KKZ	12.10±2.51	0.1482±0.0075	0.6071±0.0131
KR	11.70±6.06	0.1572±0.0103	0.6311±0.0295
<b>Data Set: S13, <math>\sigma = 0.5, K = 15</math></b>			
R-MEAN	24.70±8.10	0.2372±0.0047	0.6217±0.0037
R-SEL	22.70±9.91	0.2373±0.0057	0.6159±0.0110
SCS	20.90±7.50	0.2388±0.0042	0.6078±0.0104
KKZ	20.60±6.47	0.2388±0.0039	0.6077±0.0062
KR	20.50±13.13	0.2377±0.0043	0.6191±0.0070
<b>Data Set: S25, <math>\sigma = 0.5, K = 15</math></b>			
R-MEAN	23.70±8.63	0.2593±0.0029	0.6050±0.0032
R-SEL	30.10±8.95	0.2586±0.0017	0.6025±0.0034
SCS	26.40±5.56	0.2593±0.0023	0.5990±0.0063
KKZ	29.90±6.89	0.2592±0.0019	0.5980±0.0039
KR	31.30±9.53	0.2587±0.0020	0.6006±0.0034

significant on clean data sets (such as S01) than on noisy data sets (such as S25).

While the  $Cmp$  scores of R-MEAN and R-SEL are notably close in all experiments, the  $Sep$  scores of R-MEAN are almost consistently worse than those of R-SEL. The difference is particularly significant when  $K$  is large and the data set is clean. This is probably due to the initial state of R-MEAN which locates the seed clusters very close to each other. In addition, while not reflected in Table II, R-MEAN tends to give slightly faster convergence speed than R-SEL when  $K$  is small ( $K = 5$ ), while significantly slower convergence when  $K$  is large ( $K = 60$ ).

The performance of the two distance optimization methods, namely SCS and KKZ, are rather comparable, in terms of  $Cmp$ ,  $Sep$ , and  $I$ . Compared with the two random methods R-MEAN and R-SEL, both SCS and KKZ showed comparable  $Cmp$  scores. However, the  $Sep$  scores generated by SCS and KKZ are drastically better than those of R-MEAN, R-SEL, as well as KR. This suggests that these initialization methods do help to optimize the cluster separation of K-Means output.

Despite of its high computational complexity in pre-estimating the density of the inputs, KR does not show competitive performance over the other four methods in our experiments, both in terms of  $Cmp$  and  $Sep$ . The only advantage of using KR seems to be on the marginally faster convergence speed in some experiments when  $K$  is small. This observation however is not statistically significant in most other experiments.

### C. Real-life Data Sets

In addition to the synthetic data sets, our experiments adopted a variety of publicly available real-life data sets.

TABLE III

THE STATISTICS OF THE FOUR REAL-LIFE DATA SETS USED IN OUR EXPERIMENTS.

	Iris	ImgSeg	LtrRec	Reuters
Num. of instances	150	2,310	20,000	9,530
Num. of features	4	19	16	365
Num. of clusters	4	7	26	10

TABLE IV

THE EXPERIMENTAL RESULTS OF THE FIVE INITIALIZATION METHODS ON FOUR REAL-LIFE DATA SETS. NOTATIONS USED HERE ARE SAME AS THOSE IN TABLE II.

	$I$	$Cmp$	$Sep$
<b>Data Set: Iris, <math>\sigma = 1, K = 4</math></b>			
R-MEAN	9.80±1.93	0.2921±0.0000	0.8323±0.0000
R-SEL	17.00±0.00	0.2810±0.0000	0.7906±0.0000
SCS	7.00±0.00	0.2784±0.0000	0.7866±0.0000
KKZ	7.00±0.00	0.2784±0.0000	0.7866±0.0000
KR	6.00±0.00	0.2790±0.0000	0.7881±0.0000
<b>Data Set: ImgSeg, <math>\sigma = 500, K = 7</math></b>			
R-MEAN	13.60±4.65	0.8762±0.0029	0.7346±0.0057
R-SEL	13.20±2.35	0.8835±0.0013	0.7424±0.0005
SCS	9.00±2.11	1.0183±0.0555	0.4236±0.0085
KKZ	9.80±1.93	1.0369±0.0492	0.4266±0.0076
KR	13.60±5.89	0.8829±0.0063	0.7398±0.0075
<b>Data Set: LtrRec, <math>\sigma = 5, K = 26</math></b>			
R-MEAN	29.50±10.39	0.6037±0.0044	0.1444±0.0069
R-SEL	32.90±8.75	0.6017±0.0024	0.1728±0.0040
SCS	38.10±13.37	0.6017±0.0024	0.1278±0.0040
KKZ	30.80±6.16	0.6020±0.0068	0.1258±0.0073
KR	37.00±8.86	0.6025±0.0059	0.1388±0.0063
<b>Data Set: Reuters, <math>\sigma = 0.5, K = 10</math></b>			
R-MEAN	14.90±3.28	0.8018±0.0358	0.3192±0.0498
R-SEL	18.00±7.85	0.8571±0.0270	0.3973±0.0466
SCS	18.10±6.33	0.8168±0.0391	0.3460±0.0575
KKZ	19.20±5.18	0.7959±0.0278	0.3133±0.0353
KR	13.50±4.35	0.8343±0.0263	0.3619±0.0427

These include the Reuters-21578 document collection (Reuters in short), and three databases from the UCI machine learning repository (MLDB), namely Iris, Image Segmentation (ImgSeg), and Letter Recognition (LtrRec).

The documents from the top ten categories of the Reuters-21578 document collection based on the ModApte split are used in our experiments. We adopted the bag-of-words feature representation scheme for the documents.  $CHI$  ( $\chi$ ) statistics [23] was employed as the ranking metric for feature selection. Based on a bag of 365 top-ranking keyword features, the content of each document was represented as an in-document term frequency (TF) vector, which was then processed using an inverse document frequency (IDF) based weighting method [22] and subsequently Euclidean normalized. Null vectors (i.e. vectors with all attributes equal to 0) are removed from the data set. The three databases from UCI MLDB are directly used without preprocessing. Table III summarizes the statistics of these four data sets.

Table IV reports the experimental results on these four

TABLE V  
THE NUMBER OF EXPERIMENTAL BATCHES IN WHICH EACH  
INITIALIZATION METHOD PERFORMED SIGNIFICANTLY BEST.

	R-MEAN	R-SEL	SCS	KKZ	KR
Best $I$	3	1	3	3	3
Best $Cmp$	2	0	3	2	1
Best $Sep$	0	0	3	7	0

data sets, by setting  $K$  equal to the “optimal” number of clusters in each data set. In terms of  $Cmp$  and  $Sep$ , while R-MEAN performs notably worse than other four methods on the compact and clean Iris data set, its performance is significantly better than that of R-SEL and KR and is comparable to that of SCS and KKZ on the sparse and noisy Reuters data set. The  $Sep$  scores of SCS and KKZ are consistently better than those of R-MEAN, R-SEL, and KR on all the four data sets. Although the  $Cmp$  scores of SCS and KKZ are competitive to other methods, they can be drastically worse on some data distribution like the  $ImgSeg$  data set. In general, the observations based on the real-life data benchmark are consistent with those on the synthetic data sets.

#### V. CONCLUDING REMARKS AND FUTURE WORK

Table V summarizes the statistics across the thirteen batches of experiments, corresponding to the three synthetic data sets with  $K = 5, 15$  and  $60$ , as well as the four real-life data sets, in terms of the number of batches in which each initialization method performed significantly better than its competitors.

Our work highlights that SCS and KKZ, both based on distance optimization, tend to help K-Means in producing clustering solution with significantly better cluster separation. The cluster compactness of SCS and KKZ outputs are satisfactory as well. Compared with SCS, KKZ produces slightly higher performance in our experiments. In addition, KKZ is easier to use than SCS, in sense that it is a fully automated approach, while SCS depends on a predesignated threshold parameter.

The KR method based on density estimation did not bring significant performance gain over the random sampling methods R-MEAN and R-SEL. The only advantage of KR seems to be the slightly faster convergence speed in some cases.

The comparisons between R-MEAN and R-SEL, being the least computationally complex methods among the five studies in our experiments, give mixed results. Generally, the cluster compactness indices of their outputs are comparable. The cluster separation of R-MEAN output is worse than that of R-SEL on compact and clean data sets (such as  $S01$  and  $Iris$ ), but better on sparse and noisy data sets (such as  $Reuters$ ). When the number of output clusters is small, R-MEAN will give faster convergence speed than R-SEL does. But when the number of output clusters is large, it gives slower convergence speed.

We shall point out that the observations above probably are due to the nature of K-Means clustering which only optimizes the cluster compactness without optimizing the cluster separation in the output. Such an optimization approach would

suppress the density estimation efforts done by KR, and could be complemented by SCS and KKZ towards better cluster separation. It will be interesting to see if we can observe the similar results on a more general iterative refinement clustering method, such as EM [5].

#### REFERENCES

- [1] M. Al-Daoud and S. Roberts. New methods for the initialisation of clusters. Technical Report 94.34, School of Computer Studies, University of Leeds, 1994.
- [2] G. Babu and M. Murty. A near-optimal initial seed value selection in K-Means algorithm using a genetic algorithm. *Pattern Recognition letters*, 14:763–769, 1993.
- [3] Paul S. Bradley and Usama M. Fayyad. Refining initial points for K-Means clustering. In *Proceedings of 15th International Conference on Machine Learning*, pages 91–99. Morgan Kaufmann, San Francisco, CA, 1998.
- [4] G.A. Carpenter and S. Grossberg. ART 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, 26:4919–4930, 1987.
- [5] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, B(39):1–38, 1977.
- [6] E. Forgy. Cluster analysis of multivariate data: efficiency vs. interpretability of classifications. In *WNAR meetings, Univ of Calif Riverside*, number 768, 1965.
- [7] P. Franti and J. Kivijarvi. Randomised local search algorithm for the clustering problem. *Pattern Analysis & Applications*, (3):358–369, 2000.
- [8] P. Franti, J. Kivijarvi, and O. Nevalainen. Tabu search algorithm for codebook generation in vector quantization. *Pattern Recognition*, 31(8):1139–1148, 1998.
- [9] M.R. Garey, D.S. Johnson, and H.S. Witsenhausen. The complexity of the generalized Lloyd-max problem. *IEEE Transactions on Information Theory*, 28(2):255–256, 1980.
- [10] J. He, A.H. Tan, and C.L. Tan. ART-C: A neural architecture for self-organization under constraints. In *Proceedings of International Joint Conference on Neural Networks (IJCNN)*, pages 2550–2555, 2002.
- [11] J. He, A.H. Tan, C.L. Tan, and S.Y. Sung. On quantitative evaluation of clustering systems. In Weili Wu, Hui Xiong, and Shashi Shekhar, editors, *Information Retrieval and Clustering*. Kluwer Academic Publishers, 2003. ISBN 1-4020-7682-7.
- [12] I.T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986.
- [13] I. Katsavounidis, C. Kuo, and Z. Zhang. A new initialization technique for generalized Lloyd iteration. *IEEE Signal Processing Letters*, 1(10):144–146, 1994.
- [14] L. Kaufman and Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. Wiley, New York, 1990.
- [15] R.J. Kuo. Integration of adaptive resonance theory II neural network and genetic K-Means algorithm for data mining. *Journal of the Chinese Institute of Industrial Engineers*, 19(4):64–70, 2002.
- [16] Y. Linde, A. Buzo, and R.M. Gray. An algorithm for vector quantizer design. *IEEE Transactions on Communication*, 28(1):84–95, 1980.
- [17] J.B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the 5th Berkeley symposium in mathematics and probability*, pages 281–297, 1967.
- [18] J.M. Pena, J.A. Lozano, and P. Larranaga. An empirical comparison of four initialization methods for the K-Means algorithm. *Pattern Recognition Letters*, 20:1027–1040, 1999.
- [19] SAS Institute Inc. *SAS User’s Guide: Statistics*, version 5 edition, 1985.
- [20] B. Thiesson, C. Meek, D. Chickering, and D. Heckerman. Learning mixtures of bayesian networks. Technical Report MSR-TR-97-30, Microsoft Research, 1997.
- [21] J.T. Tou and R.C. Gonzalez. *Pattern Recognition Principles*. Addison-Wesley, Massachusetts, 1974.
- [22] Y. Yang and X. Liu. A re-examination of text categorization methods. In *22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 42–49, 1999.
- [23] Y. Yang and J.P. Pedersen. A comparative study on feature selection in text categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning (ICML)*, pages 412–420, 1997.