International Journal of Advanced Research in IT and Engineering

ISSN: 2278-6244

THE PROBLEM OF HIGH DIMENSIONALITY WITH REPEATED PATTERNS IN CLUSTERING

Dr. T. Sudha*

Swapna Sree Reddy Obili**

Abstract: *In this paper, new algorithms, called locally supported (LS) algorithms, are developed to estimate the class label of the kind of dataset without using a dimension reduction technique. As many dimensions of each group share a similar distribution, the distance measure is confused and every data point has a very similar contribution to each estimated cluster center. Our strategy is to confine the support (the non-zero region of a function) of the necessary condition of the cluster center so that the data points, which are far away from the cluster centers, make a small or even no contribution. This is different from the role of a membership function in the FCM algorithm. For a membership function, a non-zero weight is assigned to every point. In our case, a small weight is assigned to some data points, even if they belong to the corresponding cluster. Thus, a subgroup of the data points is adopted to estimate the cluster centers and the effect of confusion is reduced accordingly. This concept is applied to both the HCM and FCM algorithms. Experiment results show that the LS algorithms are able to solve the problem without using a dimension reduction technique and able to yield very accurate results. We also compare the performance of the LS algorithms with other methods in real world high dimensional datasets. Experiments show that LS algorithms show superiority over other methods including the algorithm, which uses dimension reduction.*

Keywords: *High Dimensionality, Trunk's theorem, Repeated patterns, Clustering, LS Algorithm*

*Professor & Research Supervisor, Dept. of Computer Science, Sri Padmavathi Women's University, Tirupati, A.P., India.

**Ph.D Research Scholar, Dept. of Computer Science, Sri Padmavathi Women's University, Tirupati, A.P., India.

1 INTRODUCTION

In many real-world applications, there are number of dimensions where repeated patterns occur in a dataset. Most existing clustering methods have difficulty dealing directly with these high dimensional datasets. The reason is that these features corrupt the dataset and weaken the reliability of the distance measure between two data points. In some cases, the classification accuracy is only 50% for this type of high dimensional data.

There are many methods to resolve this type of high dimensionality problem. The common way is to prune the corrupted features, which are treated as irrelevant in cluster analysis. Fisher developed a COBWEB system for a symbolic dataset [Fisher 1987]. This system measures the dependence of each feature to partition. If the probability density function of a feature with partition is almost the same as without partition, the feature is treated as irrelevant. Talavera modifies the system by re- defining the irrelevance as the low dependencies of a feature with the rest of the features [Talavera 2000]. Devaney and Ram extend the COBWEB system to numeric datasets [Devaney and Ram 1997]. Other than the COBWEB system, Law *et al*. establish an EM based algorithm to resolve this high dimensionality problem [Law *et al.* 2004]. They assign a feature saliency weight to separate useful and irrelevant features. Another similar topic for unsupervised learning in high dimensional data analysis is the unsupervised feature selection problem [Basu *et al*. 2000; Roth and Lange 2004; Mirkin 1999; Mitra *et al*. 2002]. They also adopt a similar definition for pruning the irrelevant features. These include correlation coefficient, least square regression error and entropy measures [Dash *et al*. 2002; Jennifer and Brodley 2000; Jennifer *et al*. 2003; Pena *et al*. 2001; Rao 1973]. These methods measure the similarity between two pairs. The pairs of features having high similarity are pruned.

Although these methods can reduce the effect of feature corruption, they also lower the accuracy of the classification result. This is because the pruned dimensions may carry useful information for the label. This leads to a dilemma. If we prune the dimensions, the accuracy may be lower. If we do not prune the dimensions, the distance measure will be confused. Under this challenge, the problem is still solvable, which is inspired by Trunk's theorem. Trunk has investigated the following problem [Trunk 1979]. There are two groups in the dataset. Each dimension of the group is generated by the same Gaussian distribution with the same standard derivation but using different means. The means of the two groups are

closer to each other if the dimension is higher. Thus, the features in these two groups are generated by almost the same distribution. Due to the presence of these highly repeated patterns, the separation between the two groups is vague and even the whole dataset only contains a single cluster, which is like a cloud. In this case, Trunk shows that if the means of the two groups are known, the classification error rates will be small [Trunk 1979]. Thus, if we are able to estimate the means of the two groups accurately, the error rate will be small as well.

2 HIGH DATA-DIMENSIONALITY PROBLEM

In this section, we first state the Trunk's theorem. Then, the basic concept of the LS algorithms is presented.

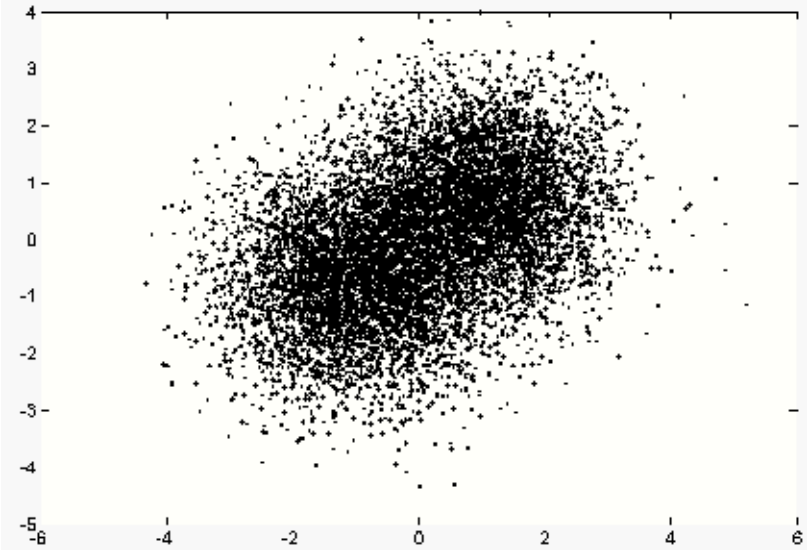
2.1 Trunk's Theorem for High Dimensional Datasets

Trunk's Theorem [Trunk 1979]: A dataset X consists of two classes with *d* dimensions and *n* samples. They are constructed by the Gaussian distributions with means and variances (μ, I_d) and $(-\mu, I_d)$, where I_d is the n×n identity matrix and μ is defined as

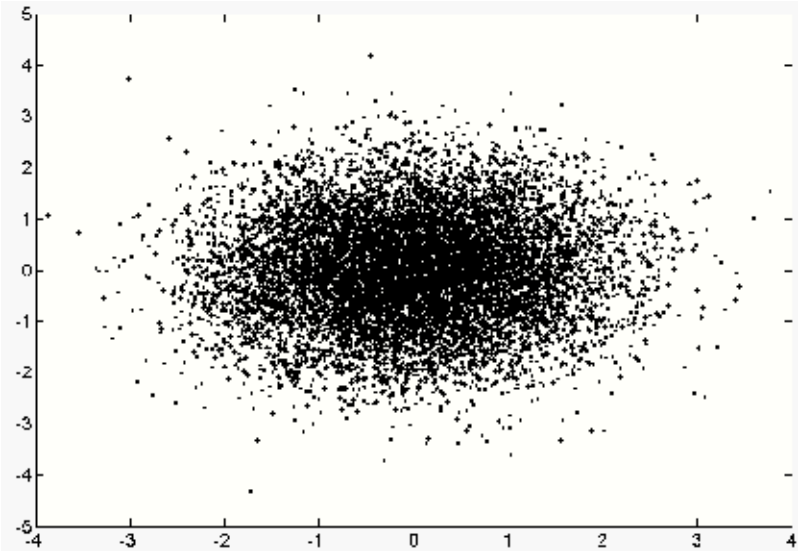
$$\mu = \left[\frac{1}{\sqrt{1}}\frac{1}{\sqrt{2}}\frac{1}{\sqrt{3}}.....\frac{1}{\sqrt{d}}\right]^T . \qquad (2.1)$$

If the mean vector μ is known, then the probability of error $p_e(d)$ monotonically decreases as the number of dimensions increases.

Figure 2.1 shows the first two and last two dimensions of the Trunk's dataset. Apparently, the separation between the two classes is smaller when the dimension is increased. In the figure, the two classes even merge together when *d* = 20. So, the two classes of this dataset may only contain a single cluster and it is a form of a cloud. Trunk shows that if the means of the two groups are known, the classification error rate will be small. Thus, if we are able to estimate the means of the two groups accurately, the error rate will be small as well.



(a) The first two dimensions of the dataset.



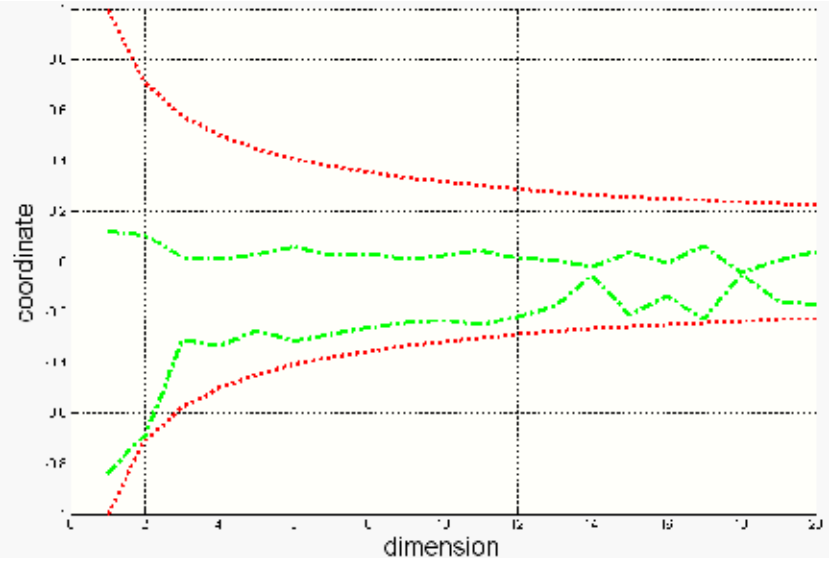
(b) The last two dimensions of the dataset.

Figure 2.1. Illustration of the Trunk's dataset.

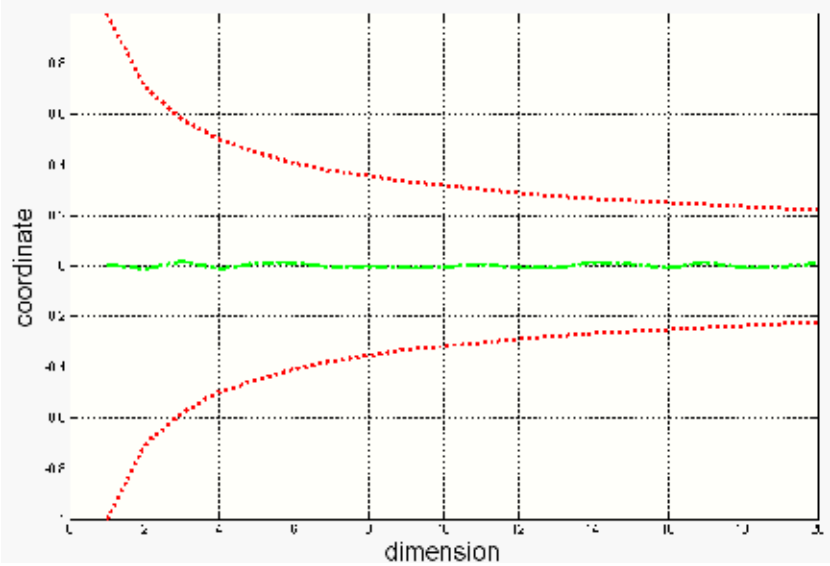
2.2 Illustration of the Problem

We apply the HCM and FCM algorithms to the Trunk's dataset with n = 5000 and d = 20 and see the goodness of these algorithms on approximating the means of the two groups. Figure 2.2 shows the two cluster centers in all 20 dimensions yielded by the HCM algorithm, the FCM algorithm and the ground truth. The estimated cluster centers are represented by the line '-.-'. We can see that the solutions yielded by the HCM and FCM algorithms are far away from the ground truth. Moreover, the two cluster centers in the FCM algorithm converge to

the same place and only one line is plotted. This shows that these algorithms are confused by the repeated patterns and the two cluster centers terminate at the nearly same place.



(a) '-.-' and '-' represent the two cluster centers yielded by the HCM algorithm and the ground truth respectively.



(b) '-.-' and '- ' represent the two cluster centers yielded by the FCM algorithm and the ground truth respectively.

Figure 2.2. Estimated cluster centers using the HCM and FCM algorithms to Trunk's dataset with *n* = 5000 and *d* = 20. The x-axes of the figures represent the dimension of the two cluster centers while the y-axes represent the corresponding values of the cluster center in this dimension.

One of the reasons why most clustering algorithms do not perform very well for this type of high dimensional dataset is that most of the dimensions share a similar distribution. Because of this, the distance measures cause every data point to make a very similar contribution to each estimated cluster center. For example, the data points in one group can influence the estimated cluster center in another group. This is the reason why the two estimated cluster centers using the FCM algorithm converge to the same place. The confusion also influences the performance of the HCM algorithm in a negative way. In contrast to the FCM algorithm, Figure 2.2(a) shows that the two cluster centers do not converge to the same place but clearly separate to two different places. This is because the membership function $I_k(x_i)$ for the HCM algorithm is a binary variable not probabilistic. There can be no relationship between two data points in the HCM algorithm (i.e. $I_k(x_i) = 1$ and $I_k(x_j) = 0$ for $i \neq j$) and the confusion from this high dimensional data is less serious. One widely used method to handle this high dimensional problem is dimension reduction. However, dimension reduction can prune some useful features and reduce the classification error rate.

2.3 Basic Concept of the LS Algorithms

To handle the high dimensionality with the repeated pattern problems, we consider the general necessary equation for the HCM and FCM clustering algorithms shown below:

$$\sum_{i=1}^{n} \omega_{ik} \left(v_k - x_i \right) f\left(x_i - v_k \right) = 0 \ , \qquad (2.2)$$

where ω_{ik} is the weighing function. If $\omega_{ik} = I_k(x_i)$, it is the necessary equation of the HCM algorithm. If $\omega_{ik} = \mu_{ik}$, it is the necessary equation of the FCM algorithm. The support of this necessary equation is constructed by the product of ω_{ik} and the function $f(x_i-v_k)$. The mathematical meaning of the term 'support' is the region where $f(x_i-v_k) \neq 0$. Here, we relax the use of this term and define it as the complement of the region where $f(x_i-v_k) \approx 0$. In the high dimensionality problem, the distance measurements among data points are very similar. In this equation, if the support of ω_{ik} $f(x_i-v_k)$ is too large, the estimated cluster center v_k will acquire the information of many data points and some of them can even be from different groups. Based on this observation, we can resolve the problem by confining the support of the necessary condition. Our objective is to modify the function of $f(x_i-v_k)$ so that

the points, which are far away from a cluster center, make small or even no contribution to the estimated cluster centers. This can greatly reduce the problem from high dimensionality. In the next section, we will use the calculus of variations to find a equation so that the support of the ω_{ik} $f(x_i-v_k)$ is small enough to achieve our goal.

3 METHOD BASED ON CALCULUS OF VARIATIONS

In this section, we impose the above concept to the general objective function. Then, we will show the estimation procedure for class label of this new technique.

3.1 Locally Supported Clustering Algorithms

In this section, we apply the calculus of variations [Evans 1998; Richard 1998] to constrain the support of the necessary equation. Now, we consider the general clustering model as follows.

$$J(v_k, \omega_{ik}) = \sum_{i=1}^{n}\sum_{k=1}^{c} \omega_{ik} \|x_i - v_k\|_\varepsilon \qquad (3.1)$$

where $\|u\|_\varepsilon = \sqrt{\varepsilon^2 + \|u\|_2^2}$.

Our objective is to find a necessary equation for v_k so that its support is confined. We formulate the clustering algorithm in a continuous manner and we only focus on the unknown function v(s,t).

$$J(v(s,t)) = \int_x \int_\Gamma \omega(s,t) \|x(t) - v(s,t)\|_\varepsilon H_\Gamma(s) H_x(t) ds dt \qquad (3.2)$$

with constraint v(s,t) = v(s) $\forall$ t. Here X_{space} is the data space with $X \subset X_{space}$. $H_X(t)$ is the indicator function for the dataset X. For Γ = [0,1], $H_\Gamma(s)$ is a delta function, where $H_\Gamma(s) \to \infty$ if $s \in \{s_k: 0 \le s_k \le 1, k=1,2, \ldots c\}$, otherwise, $H_\Gamma(s) = 0$. We apply the calculus of variations to this objective function with a function φ (s,t), which means the perturbation is dependent on data X. Taking the derivative with respect to γ at γ=0, we have

$$\frac{d}{d\gamma}\Big|_{\gamma=0} J(v(s,t) - \gamma\varphi(s,t)) = \frac{d}{d\gamma}\Big|_{\gamma=0} \int_{x_{space}} \int_\Gamma \omega(s,t) \|x(t) - v(s,t) - \gamma\varphi(s,t)\|_\varepsilon H_\Gamma(s) H_x(t) ds\, dt$$

$$= \int_{x_{space}} \int_\Gamma \omega(s,t) \frac{v(s,t) - x(t)}{\|v(s,t) - x(t)\|_\varepsilon} \varphi(s,t) H_\Gamma(s) H_x(t) ds\, dt \qquad (3.3)$$

We confine the support by the following setting. φ(s,t) is taken to be the composition of two functions. i.e. φ (s,t) = ϕ(s,t) Ψ(s). Ψ(s) is an arbitrary vector function while ϕ(s,t) is a

function satisfying the following three conditions. (1) $\phi(s,t)\rightarrow\infty$ if $||v(s,t)-x(t)||_\varepsilon \rightarrow 0$; (2) $\phi(s,t) \rightarrow 0$ if $||v(s,t)-x(t)||_\varepsilon \rightarrow \infty$ and (3) For each s_k, the function $h_k(t) = \phi sk,t)/ ||v(sk,t)-x(t)||_\varepsilon$ contains only one minimum. The first two conditions are adopted to constrain the support of the function so that a small weight is assigned outside the neighborhood of v(s,t). The third condition is to restrict the function so that the necessary equation is a locally convex function. There are many functions, which satisfy the above conditions. In our experiments, the function $\phi(s,t) = 1/(||v(s,t)-x(t)||^q_\varepsilon)$ performs well with q = 1. We check the conditions of $\phi(s,t)$ as follows. Obviously, the first two conditions are satisfied. For the third condition, the function $h_k(t) = \phi(s_k,t)/||v(s_k ,t)-x(t)||_\varepsilon = 1/(||v(s_k,t)-x(t)||^2_\varepsilon)$ contains only one minimum for $||v(s_k,t)-x(t)||_\varepsilon>0$. Putting these together, Equation (3.3) is now rewritten as

$$\int_{x_{space}}\int_{\Gamma}\omega(s,t)\frac{v(s,t)-x(t)}{\|v(s,t)-x(t)\|_\varepsilon}\varphi(s,t)H_\Gamma(s)H_x(t)\,ds\,dt \qquad (3.4)$$

$$= \int_{x_{space}}\int_{\Gamma}\omega(s,t)\frac{v(s,t)-x(t)}{\|v(s,t)-x(t)\|_\varepsilon}\frac{1}{\|v(s,t)-x(t)\|_\varepsilon}\Psi(s)H_\Gamma(s)H_x(t)\,ds\,dt$$

$$= \int_{x_{space}}\int_{\Gamma}\omega(s,t)\frac{v(s,t)-x(t)}{\|v(s,t)-x(t)\|^2_\varepsilon}\Psi(s)H_\Gamma(s)H_x(t)\,ds\,dt$$

$\Psi(s)$ is an arbitrary vector function and $H_\Gamma(s)$ is a delta function, which is not equal to zero only at a finite number of points. By applying the constraint v(s,t) = v(s) for $\forall$ t and setting Equation (3.4) equal to zero, we have

$$\sum_{i=1}^{n}\omega_{ik}\frac{v_k - x_i}{\|v_k - x_i\|^2_\varepsilon} = 0\ , \qquad (3.5)$$

where $v_k=v(s_k)$. We use an iterative scheme to find the solution of this equation by assuming that v_k on the numerator is the variable at the p^{th} iteration and that v_k on the denominator is known at the $(p-1)^{th}$ iteration. Thus, the new update step equation becomes

ISSN: 2278-6244

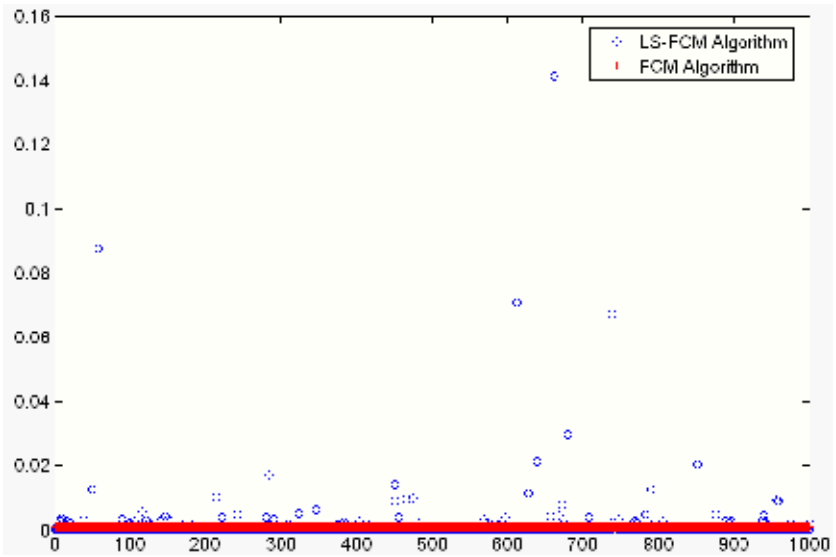
$$v_k^{(p)} = \frac{\sum_{i=1}^{n} \omega_{ik} \left(\frac{x_i}{\left\| v_k^{(p-1)} - x_i \right\|_\varepsilon^2} \right)}{\sum_{i=1}^{n} \omega_{ik} \left(\frac{1}{\left\| v_k^{(p-1)} - x_i \right\|_\varepsilon^2} \right)} . \qquad (3.6)$$

If $\omega_{ik} = \mu^m_{ik}$ it is cluster estimation using fuzzy partition function and it is named the locally supported fuzzy c-means (LS-FCM) algorithm. If $\omega_{ik} = I_k(x_i)$, it is cluster estimation using the crisp partition function an it is named the locally supported HCM (LS-HCM) algorithm.

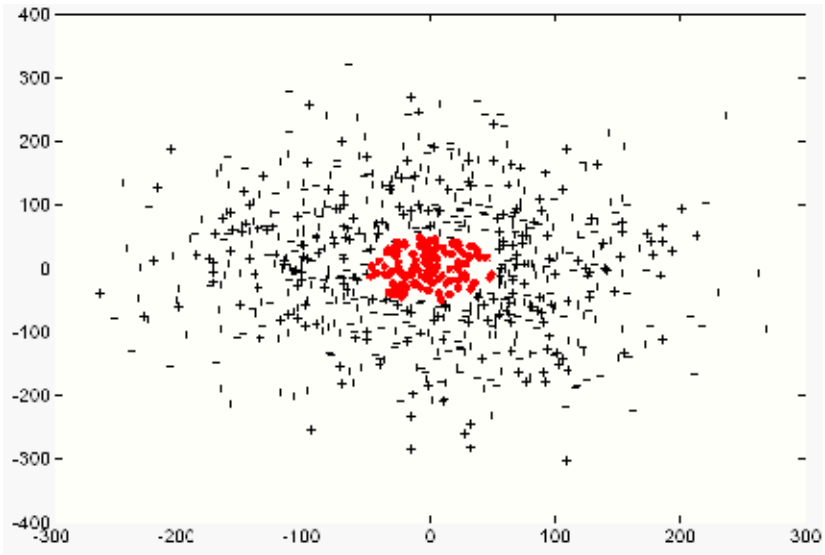
We now further illustrate the property of Equation (3.5) by the following elliptical cluster. The dataset is generated by the Gaussian distribution with means (0,0) and covariance matrix $1000I_2$. I_2 is the identity matrix with size 2×2. There are 1000 points generated in total. Figure 3.1(a) shows the samples and "+" represents each of them. We now compute the weights of both the new equation and FCM algorithms at the point (0,0). That is, we only consider one cluster case and put v = (0,0) into Equation $v_k^{(p)} = \frac{\sum_{i=1}^{n} \mu_{ik}^m x_i}{\sum_{i=1}^{n} \mu_{ik}^m}$ for the FCM algorithm and into Equation (3.5) for the LS-FCM algorithm. The weights (the term before x_i in the update equations) for each of the samples are given in Equations (3.7) and (3.8). These values are plotted in Figure 3.1(b). We can see that the weights for the FCM algorithm for every sample are almost the same. In the LS-FCM algorithm, some samples have larger weights and most of them are nearly zero. The red circles shown in Figure 3.1(a) are the weights for the LS-FCM algorithm having weights larger than 0.001. We can see that the large weights are given to the samples, which are close to the point (0,0).

$$v_{FCM} = \sum_{i=1}^{n} \left(\frac{\omega_i}{\sum_{i=1}^{n} \omega_i} \right) x_i . \qquad (3.7)$$

$$v_{LS-FCM} = \sum_{i=1}^{n} \left(\frac{\omega_i \left(\frac{1}{\| x_i \|_\varepsilon^2} \right)}{\sum_{i=1}^{n} \omega_i \left(\frac{1}{\| x_i \|_\varepsilon^2} \right)} \right) x_i \ . \qquad (3.8)$$



(a) The dataset and the region with circles are the data points having weights larger than 0.001 in the LS-FCM algorithm.



(b) The weights of the LS-FCM and FCM algorithms.

Figure 3.1 Illustration of the LS-FCM algorithms.

3.2 Cluster Estimation Procedure

Equation (3.5) is the equation for v_k, in which the support of $f(x_i - v_k)$ is smaller. Good clusters, which are able to handle the problem of highly repeated patterns, satisfy this equation together with the necessary condition of the membership function. However, the denominator of Equation (3.5) is a square of the norm $||v_k - x_i||_\varepsilon$, which causes the equation to be highly non-linear. This may produce lots of local optima. We adopt the following cluster estimation strategy to filter the local optima in both LS-FCM and LS-HCM algorithms. For the LS-FCM algorithm, the initial guess is yielded by the RL-FCM algorithm, which is described in Chapter 3.

The estimation procedure for the LS-FCM algorithm is given in Table 3.1.

Stage 1: Initial guess using the RL-FCM algorithm
Step 1: Initial V and *a* values are given by the user.

Step 2: The values U and V are updated using Equations $\mu_{ik} = \dfrac{\left|X_i - v_k^{(p-1)}\right|_\in^{-1/(m-1)}}{\sum_{k=1}^{c}\left|X_i - v_k^{(p-1)}\right|_\in^{-1/(m-1)}}$ and

$$v_k^{(p)} = \frac{\sum_{i=1}^{n}\mu_{ik}^m\left(\dfrac{x_i}{a + \left\|v_k^{(p-1)} - x_i\right\|_\varepsilon}\right)}{\sum_{i=1}^{n}\mu_{ik}^m\left(\dfrac{1}{a + \left\|v_k^{(p-1)} - x_i\right\|_\varepsilon}\right)}.$$

Step 3: Decrease *a* by a reduction rate. If *a* is not close to zero, go to Step 2.
Otherwise, go to Stage 2.
Stage 2: Detecting Cluster Centers using the LS-FCM Algorithm
Step 1: Examine if there are two cluster centers converging to the same place. If yes, remove the redundant one and randomly add one point in the dataset to V.

Step 2: The values U and V are updated by Equations $\mu_{ik} = \dfrac{\left|X_i - v_k^{(p-1)}\right|_\in^{-1/(m-1)}}{\sum_{k=1}^{c}\left|X_i - v_k^{(p-1)}\right|_\in^{-1/(m-1)}}$ and (3.6)

until termination.

Table 3.1: Estimation Procedure for the LS-FCM algorithm.

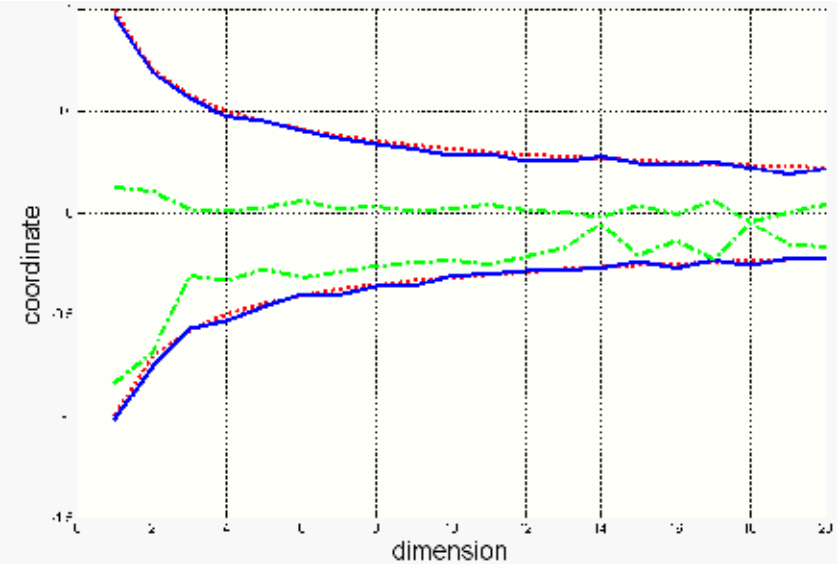
The LS-FCM algorithm converges to a limit point, which may not be an extreme of the general objective function in Equation (3.1). The convergence of the cluster centers means that the cluster centers satisfy the Equation (3.5). This means that the estimated cluster centers are the means of its local neighborhood samples. In all the experiments, *a* starts with the value 500 while the reduction rate is 0.9. The fuzzy parameter *m* in Stage 1 is set as *m* = 1.5. In Stage 2, the fuzzy parameter *m* = 1.1. The reason to set *m* in Stage 2 to be smaller is to further confine the support of the necessary equation. The estimation procedure for the LS- HCM algorithm is given as follows. The HCM algorithm is applied to the dataset. Then, the clustering result is used as an initial guess in the LS-HCM algorithm.

4 EXPERIMENT RESULTS

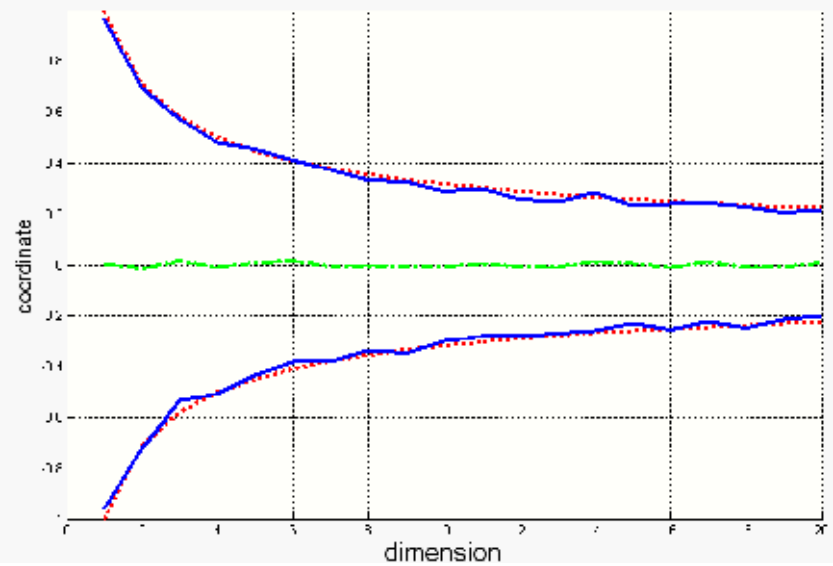
In this section, we perform experiments to show the performance of the LS algorithms by the Trunk's dataset.

4.1 Trunk's Dataset for the High Dimensionality Problem

We apply the locally supported algorithms to Trunk's dataset with *n* = 5000 and *d* = 20. The clustering results are shown in Figure 4.1 and the lines are represented by "-". We can see that the estimated cluster centers are almost the same as the ground truth. Moreover, the LS algorithms yield the same results using 10 different initial guesses.



(a) '-.-', '-' and '…' represent the two cluster centers yielded by the HCM algorithm, the LS-HCM algorithm and the ground truth respectively.



(b) '-.-', '-' and '…' represent the two cluster centers yielded by the FCM algorithm, the LS-FCM algorithm and the ground truth respectively.

Figure 4.1. Estimated cluster centers using the HCM and FCM algorithms on Trunk's dataset with *n* = 5000 and *d* = 20. The x-axes of the figures represent the dimension of the two cluster centers while the y-axes represent the corresponding value of the cluster center in this dimension.

Table 4.1 shows the MSEs between the estimated cluster centers and the ground truth. In this table, we can see that the locally supported algorithms yield the smallest distance to the ground truth. The other methods have MSEs larger than two.

Method	MSE
HCM	2.5750
LS- HCM	0.2313
Orig- HCM	3.7891
1_{2m}- HCM	3.7914
RL- HCM	3.7914
LS- HCM	0.1739

Table 4.1. MSEs between the estimated cluster centers and the ground truth.

5 CONCLUSION

In this paper, we study the high dimensionality problem. If many dimensions of the two groups have a similar distribution, the classification error rates can be as high as 50%. Although dimension reduction can resolve this problem, it may result in an increase in the classification error rate. A new clustering technique is developed to resolve this problem

without using dimension reduction. The key is to confine the support of the optimization procedure so that the points, which are far away from a cluster center, make small or even no contributions to the estimated cluster centers. This greatly reduces the confusion yielded by the high dimensional data. This idea is applied to both the HCM and FCM algorithms. The LS algorithms solve the problem of highly repeated patterns and are able to yield accurate results. The experiments on real world datasets also show that the new methods yield better results than other existing methods including the one using dimension reduction.

REFERENCES

1. [Aggarwal *et al*. 1999] - C.C. Aggarwal, C. Procopiuc, J.L. Wolf, P.S. Yu, and J.S. Park, "Fast Algorithms for Projected Clustering," *Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 61-72, 1999.*
2. [Agrawal *et al*. 1998] - R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications," *In Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 94-105. ACM Press, 1998.
3. [Akaike 1974] - H. Ak aike, "A New Look at the Statistical Model Identification," *IEEE Transaction on Automatic Control,* Vol. 19, pp. 716-723, 1974.
4. [Ball and Hall 1965] - G. H. Ball and D. J. Hall, "A Clustering Technique for Summarizing Multivariate Data_, *Behav. Sci.*, Vol. 12, pp. 153-155, 1967.
5. [Banfield and Raftery 1993] - J. D. Banfield and A. E. Raftery, "Model-based Gaussian and Non-Gaussian Clustering," *Biometrics*, Vol. 49, pp. 803-821, 1993.
6. [Barbara *et al*. 2002] - D. Barbara, Y. Li, and J. Couto. Coolcat, "An Entropy Based Algorithm for Categorical Clustering," *In Proceedings of the Eleventh International Conference on In Formation and Knowledge Management,* ACM Press, pp. 582-589, 2002.
7. [Burden and Faires 1997] - R. Burden and J. Faires, *Numerical Analysis*, 6th edition, Pacific Grove, Calif.: Brooks/Cole Pub, 1997.
8. [Castleman 1996] - K. Castleman, *Digital Image Processing*, Englewood Cliffs, N. J. : Prentice Hall, 1996.
9. [Chan and Vese 2001] - T. Chan and L Vese, "Active Contours Without Edges," *IEEE Transactions on Image Processing*, Vol. 10, pp. 266-277, 201.

10. [Chintalapudi and Kam 1998b] - K. K. Chintalapudi and M. Kam, _ The Credibilistic Fuzzy C-means Clustering Algorithm, " *In Proc. IEEE Int. Conf. Systems Man Cybernetics*, pp. 2034–2040, 1998.
11. [Dash *et al*. 2002] - M. Dash, K. Choi, P. Scheuermann and L. Huan, "Feature Selection for Clustering - A Filter Solution," *IEEE International Conference on Data Mining*, pp. 115-122, 2002.
12. [Dubes and Jain 1979] - R. Dubes and A. Jain, "Validity Studies in Clustering Methodologies," *Pattern Recognition*, Vol. 11, pp. 235-254, 1979.
13. [Evans 1998] - L. C. Evans, *Partial Differential Equations*, Providence, R.I.: American Mathematical Society, 1998.
14. [Fisher 1987] - D. H. Fisher, "Knowledge Acquisition via Incremental Conceptual Clustering," *Machine Learning*, Vol. 2, pp. 139-172, 1987.
15. [Friedman and Meulman 2004] - J. H. Friedman and J. J. Meulman, "Clustering Objects on Subsets of Attributes," *Journal of the Royal Statistical Society: Series B (Statistical Methodology),* Vol. 66, pp. 815-849, 2004.
16. [Jain *et al*. 1999] - A. K. Jain, M. N. Murty and P. J. Flynn, "Data Clustering: a Review," *ACM Computing Surveys*, Vol. 31, No. 3, pp.264-323, 1999.
17. [Kaplan 1991] - W. Kaplan, *Advanced Calculus*, 4th Edition, Reading, Mass.: Addison-Wesley, 1991.
18. [Kaufman and Rousseeuw 1990] - L. Kaufman and P. J. Rousseeuw*, Finding Groups In Data: An Introduction To Cluster analysis,* New York: Wiley, 1990.
19. [Krishnapuram and Keller 1993] - R. Krishnapuram and J. M. Keller, "A Possibilistic Approach to Clustering," *IEEE Transactions on Fuzzy Systems*, Vol. 1, pp. 98-110, 1993.
20. [Law *et al*. 2004] - M. Law, M. Figueiredo, A. Jain, "Simultaneous Feature Selection and Clustering Using Mixture Models," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 26, pp. 1154-1166, 2004.
21. [Maulik and Bandyopadhyay 2002] - U. Maulik and S. Bandyopadhyay, "Performance Evaluation of Some Clustering Algorithms and Validity Indices," *IEEE Trans. Pattern Analysis Machine Intelligence*, Vol. 24, pp. 1650-1654, 2002.

22. [Melek *et al*. 1999] - W. W. Melek, M. R. Emami, A. A. Goldenberg, "An Improved Robust Fuzzy Clustering Algorithm," *IEEE International Fuzzy Systems Conference Proceedings*, Vol. 3, pp. 1261 - 1265, 1999.
23. [Milligan and Cooper 1985] - G. Milligan and C. Cooper, "An Examination of Procedures for Determining the Number of Clusters in a Data Set," *Psychometrika*, Vol. 50, pp. 159-179, 1985.
24. [Miyamoto and Agusta 1998] - S. Miyamoto and Y. Agusta, "Algorithms for L1 and Lp Fuzzy C-means and Their Convergence," *in Studies in Classification, Data Analysis, and Knowledge Organization: Data Science, Classification, and Related Methods*, Japan: Springer-Verlag, pp. 295-302, 1998.
25. [Modha and Spangler 2003] - D. Modha and S. Spangler, "Feature Weighting in HCM Clustering," *Machine Learning*, Vol. 52, pp. 217-237, 2003.
26. [Roth *et al*. 2004] - T. Lange, V. Roth, M. Braun and J. Buhmann, "Stability-Based Validation of Clustering Solutions," *Neural Computation*, 16, pp. 1299-1323, 2004.
27. [Rousseeuw 1987] - P. Rousseeuw, "Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis," *Journal of Computational and Applied Mathematics*, Vol. 20, pp. 53-65, 1987.
28. [Rudin 1976] - W. Rudin, *Principles of Mathematical Analysis*, 3rd Edition, New York : McGraw-Hill, 1976.
29. [Zhang and Leung 2004] - J. S. Zhang and Y. W. Leung, "Improved Possibilistic C-Means Clustering Algorithms," *IEEE Transactions on Fuzzy Systems*, Vol. 12, pp. 209-217, 2004.