Volume 2, No. 7, July 2011

Journal of Global Research in Computer Science

ISSN-2229-371X

RESEARCH PAPER

Available Online at www.jgrcs.info

AN EFFICIENT APPROACH TO DETECT FOREST FIRE USING K-MEDIODS ALGORITHM

A Gnana Baskaran*[1] and Dr.K.Duraiswamy[2]

*[1]Department of Computer Science and Engineering, Faculty of Computer Science and Engineering, KSRangasamy College of Technology, Tiruchengode, Namakkal –Dist, Tamil Nadu -637215, India.
gnanabas_2000@yahoo.com

[2]Department of Computer Science and Engineering, KSRangasamy College of Technology, Tiruchengode, Namakkal –Dist, Tamil Nadu -637215, India.

Abstract: Problem Statement: : Clustering high dimensional spatial data for Forest fire risk analysis has been major issue due to sparsity of data points**..** Most of the clustering algorithm becomes inefficient if the required distance similarity measure is computed for low dimensional spatial space of high dimensional data with sparsity of data point along different dimensions and also considering the obstacles**..** Objective of this study were to contribute the complexity of projecting clusters for traffic risk analysis, (i) lack of support for reducing the number of dimensions on spatial space to reduce the searching time (ii) the lack of support for obstacles in the spatial data space. (iii) Compare computation time of HARP, Proclus, Doc, FastDoc, SSPC algorithms. **Approach:** During the first phase the satellite captured still images for different dimensions such as time and location of the forest fire network are enhanced and this images are given as input to red color image separation, During this phase the input images groped based on red color using K-Means algorithm and during the second phase the red color images are converted to gray scale images . The third phase mainly focuses on spatial attribute relevance analysis for detecting dense and sparse forest fire regions after detecting dense and sparse fire regions the algorithm employees pruning technique to reduce the search space by taking only dense fire regions and eliminating sparse fire regions and during fourth phase K-mediods algorithm is employed to project the clusters on different spatial dimensions and also it solves the problem of obstacles **Results:** First we showed that various projecting clustering algorithm on spatial space becomes inefficient if the number of dimensions increases .The new scheme proposed reduces the spatial dimension space so that it reduces the computation time and also it solves the problem of obstacles using K-mediods algorithim and finally the result is compared with HARP ,Proclus,Doc,FastDoc,SSPC The algorithms produces acceptable results when the average cluster dimensionality is greater than 10%. **Conclusion:** Hence the findings suggested the overhead reasonably minimized and using simulations, we investigated the efficiency of our schemes in supporting high dimensional spatial clustering for forest fire risk analysis.

***Keywords*:** Data mining, clustering, high dimensions projected clustering, pruning

INTRODUCTION

Data mining is the process of extracting the knowledge or useful information from the database [1]. Clustering is the familiar data mining technique to group the related data based on similarity distance measure [2]. Clustering usually employs distance measure such as Euclidean, Manhattan or Minkowski etc. However Euclidean measure is the common approach to group the related data into different groups or partition in different dimensions. However this similarity measure for high dimensional data is crucial factor. The Recent projected clustering algorithm fails to address this problem when the dimension increases. The preprocessing of dimensionality reduction can improve the clustering efficiency but fail to prevent the data loss. This motivates our effect to propose a new clustering algorithm called an efficient approach to detect forest fire using K-Mediods algorithm. The algorithm contains four phases. In the first phase, it uses K-means algorithm to group red color forest fire images. In the second phase it converts the red color images to gray scale images. The third phase uses the attribute relevance analysis to project dense and sparse forest fire images. The fourth phase uses K-medoids algorithm to project the clusters where the forest fire images belonging to dense fire regions. The algorithm capable of detecting the clusters automatically and the clustering process is restricted to the subset dimension that is the dense fire region, which avoids computation on full dimensional space.

The numbers of projected clustering algorithms have been proposed in recent years but they fail to address to address the low dimensional clusters on high dimensional space. Feature selection technique can speed up the clustering [4] process but however there is substantial information loss [5].YIP et al [7] observed that current projected clustering algorithm results only when the dimensionality of the clusters are not much lower than that of dataset. However some partitional projected clustering algorithms such as PROCLUS [5] and ORCLUS [8] make use of similarity function that involves all dimensions to find initial approximation of the clusters.

The partitional algorithm PROCLUS, which is a variant of the K-medoid method, iteratively computes a good medoid for each cluster. With the set of medoids, PROCLUS finds the subspace dimensions for each cluster by examining the neighboring locality of the space near it. After the subspace has been determined, each data point is assigned to the cluster of the nearest medoid.The algorithm is run until the sum of intracluster distances ceases to change. These algorithms are failing because of irrelevant dimension detection, and also the

algorithm requires the user to provide average dimensionality of the subspace, which is not suitable for real life time.

ORCLUS is an extended version of PROCLUS that looks for non-axis-parallel clusters, by using Singular Value Decomposition (SVD) to transform the data to a new coordinate system and select principal components. PROCLUS and ORCLUS were the first to successfully introduce a methodology for discovering projected clusters in high-dimensional spaces, and they continue to inspire novel approaches.

Another algorithm HARP [9] hierarchical projected clustering also based on the fact that two data points are likely to be in the same cluster if they are very similar to each other along many dimensions. However, when the number of relevant dimension per cluster is much lower than the data set dimension, such an assumption is invalid.

The observation motivates our effort to propose a new algorithm called An Efficient approach to Detect Forest Fire using KMediods algorithm

MATERIALS AND METHODS

A new robust clustering algorithm called **"**An Efficient approach to Detect Forest Fire using K-Mediods algorithm". The algorithm contains four phases. The algorithm is efficient because it reduces and performs computations on the subspace and it avoids computation on full dimensional space.

Color Separation

The input forest fire images are given as an input to color separation system the output of these system is the grouping of images using K-means algorithm based on red color.

Fig, 1(a)

Fig.1 (b)

Fig.1(c)

Fig.1 (d)

Figure.1: sample input images
Figure.1 shows sample input images before color separation.

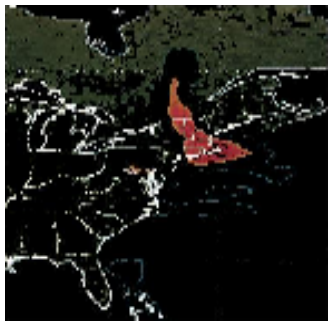

Fig.2 (a)

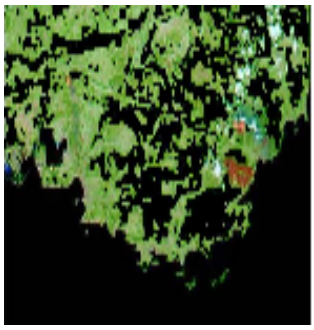

Fig.2(b)

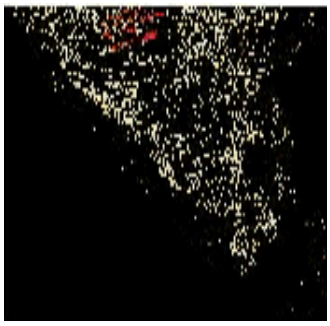

Fig.2(c)

Fig.2 (d)

Figure.2: Red Color Images

Figure.2 is the output of the first phase which contains cluster of red color images and the clustering process is done using K-means algorithim.

Gray Scale Conversions

During this second phase the red color grouped images as shown in Fig.2 are converted to gray scale images as shown below.

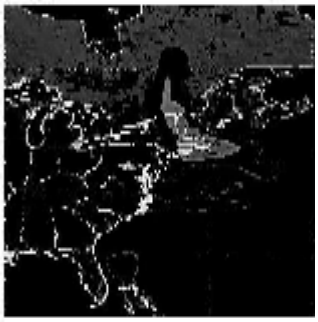

Fig.3 (a)

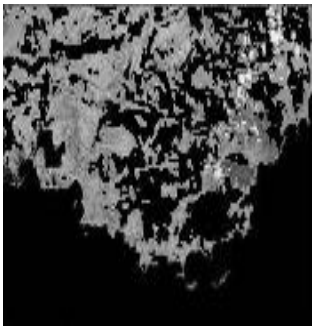

Fig.3 (b)

Fig.3(c)

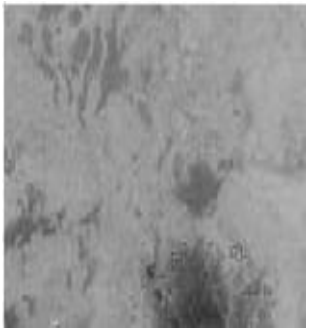

Fig.3(c)

Figure.3 Gray Scale Images
The images in Fig.3 are given a input to third phase attribute relevance analysis

Attribute Relevance Analysis

With help of attribute relevance analysis, the sparseness degree y_{ij} are computed for different forest fire gray scale images based on threshold values pertaining to fire region. The sparseness degree y_{ij} are given by the formula.

$$y_{ij} = \sum \frac{(r - c_i^j)^2}{k}$$

$$r \in p_i^j (x_{ij})$$

The minimum value of y_{ij} represents dense region and maximum value represents sparse region. Similarly different y_{ij} values are computed for different forest fire images across different dimensions and the simulation results are tabulated and the histogram is also generated as shown below

Table 1: Relevance Attribute analysis table

Images	Sparseness Values(yij)
Fig.3(a)	0.0310
Fig.3(b)	0.0474
Fig.3(c)	0.0350
Fig.3(d)	0.0210

With the help of above value of yij for each image we can easily detect the dense regions. The images with larger values of yij denote the sparse regions, for example in the table values of Fig.3(b) and Fig.3(c) are higher that represents the sparse region. The images with small values of yij denotes the dense regions, for example in the table the values of images Fig.3(d) and Fig.3(a) represent dense regions.

For the above sparseness values histograms are generated for identifying sparse and dense forest fire regions as shown below.

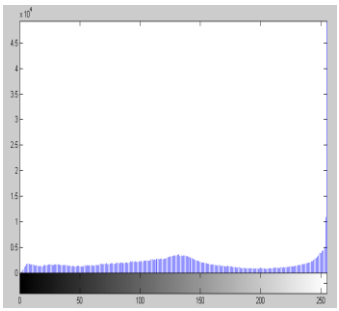

Fig.4(a)

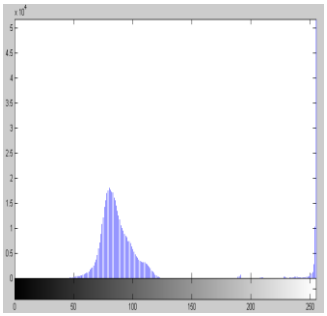

Fig.4(b)

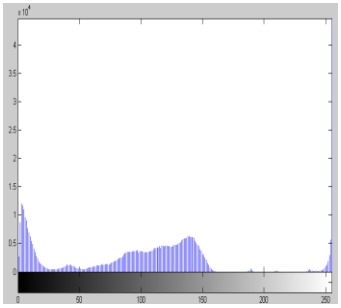

Fig.4(c)

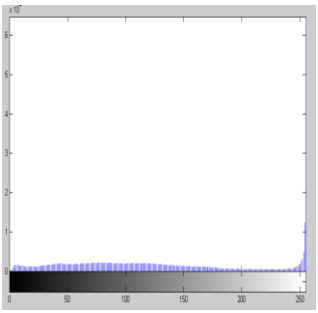

Fig.4(d)

Fig.4: Histogram corresponding to the satellite images

The histograms with high peaks corresponds to the sparse fire regions (Fig.4 (b) and fig.4(c)) and low peaks corresponds to the dense fire regions (Fig.4(d) and Fig.4(a)).

The above dense and sparseness degree can be represented in the binary form using binary matrix as shown below.

		d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}
Cluster1	y_{1i}	1	0	0	1	0	0	0	1	0	1
		:	:	:	:	:	:	:	:	:	:
		1	0	0	1	0	0	0	1	0	1
Cluster2	y_{2i}	0	1	0	1	0	1	0	1	1	1
		:	:	:	:	:	:	:	:	:	:
		0	1	0	1	0	1	0	1	1	1
Cluster3	y_{3i}	0	1	0	1	0	0	0	0	0	1
		:	:	:	:	:	:	:	:	:	:
		0	1	0	1	0	0	0	0	0	1
Cluster4	y_{4i}	0	1	0	1	0	1	0	0	1	0
		:	:	:	:	:	:	:	:	:	:
		0	1	0	1	0	1	0	0	1	0

Figure.5 The binary matrix

In the above binary matrix the data point falls on the dense fire region it is represented as 1 otherwise it is represented as 0.

Outlier Detection

Outliers are noisy, Inconstant, dissimilar data, these noisy data are removed and similar data's are taken into account. The algorithm uses binary matrix as input and checks whether two binary values Z_1 and Z_2 are similar using Jacord coefficient, the Jacord coefficient are given by the formula.

$$J_C(Z_1, Z_2) = \frac{a}{a + b + c}$$

$$where \quad a = |Z_{1j} = Z_{2j} = 1|$$

$$b = |Z_{1j} = 0 \wedge Z_{2j} = 1|$$

$$c = |Z_{1j} = 1 \wedge Z_{2j} = 1|$$

$$d = |Z_{1j} = Z_{2j} = 0|$$

the values of Jacord coefficient is between 0 (than both, z_1 and z_2 are dissimilar) and 1(both are similar).The Jacord coefficient should not be less than λ. In our example λ is set as = 0.70, normally the Jacord coefficient searches for the matching 1's in binary matrix.

Discovery of Clusters

Once the dense fire region has been identified using jacord coefficient. Then the next step is used to project clusters on dense fire region with the help of binary matrix.

For calculating minimum distance using Euclidian distance measure, the no of 1's in binary matrix are taken, which represents the dense region and the distance is calculated only for dense region and it is given by the formula.

$$dist\left(x_{ij}, us_j\right) = \sqrt{\sum_{i=j}^{d} t_{ij} * \left(x_{ij} - us_j\right)^2}$$

x_{ij}- data point

us_j - representation of object

Here t_{ij} represent the i^{th} row and j^{th} column in binary matrix and it will calculate the distance only for the entry one that t_{ij} =1.

EMPIRICAL EVALUATION

In this section we compare the performance of algorithm with SSPC [7], HARP [0], PROCLUS [5] and FASTDOC [10] the evaluation is performed on a number of Sequential data set with different characteristics. Clustering error (CE) is the one of familiar technique to measure accuracy of projected clustering as shown in Fig.5. The first step of the algorithm is to analysis the impact of cluster dimensionality on the Quality of clustering for this purpose, we selected twenty different dataset with data point N=4000, number of dimension d=100.The average cluster dimensionalities are varied from 2% to 70% of the data dimensionality d. Our goal is that no outliers are generated. The CE distance between the utputs of each of fine algorithm Projected Clustering using K - Mediods, SSPC, PROCLUS and FASTDOC.

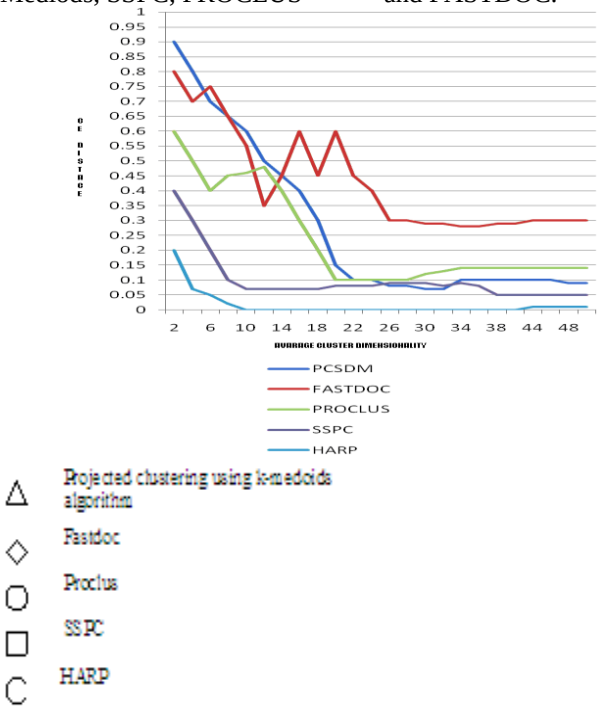

Figure.6 CE distance measure

The performance of CPU time is measured with respect to number of images and dimensionality increase as shown below

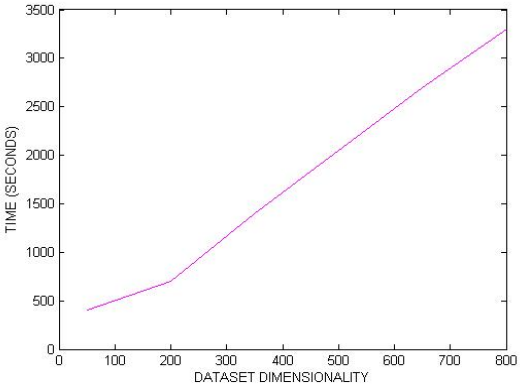

Figure.7: CPU Time execution

Table 2: CPU Time table

S. No.	Cluster Size (number of Images)	Time taken (Seconds)
01	20	150
02	30	200
03	40	250
04	50	350
05	60	360
06	70	400
07	80	460
08	90	550
09	100	630

In the above Fig.7 and Table 2, CPU time linearly increases with the number of images in a cluster.

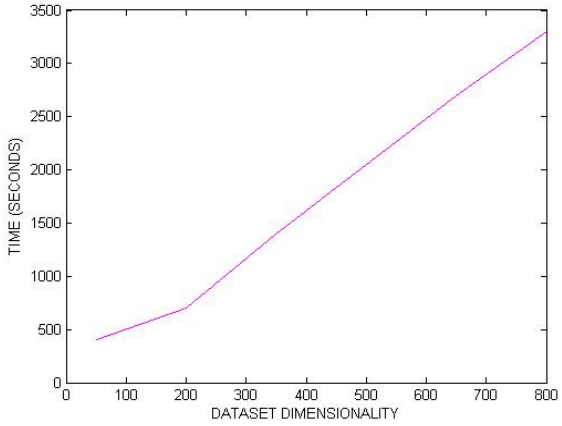

Figure.8 : Dimensionality performance

Table.3: Dimensionality table

SNO.	Data Base Dimensionality	Time(Seconds)
01	100	500
02	200	750
03	300	1000
04	400	1500
05	500	2000
06	600	2500
07	700	3000
08	800	3300

From the above Fig.8 and table 3 time increases linearly along with the dimension increase.

In the above graph Fig.5, at 2% of average cluster dimensionality. The algorithm shows acceptance results. By it performs well when for example in the above graph for projected clustering using K-medoids algorithm. If the cluster dimensionality is greater than 10, then it shows acceptable results for deleting dimension when compared with other projected clustering algorithm. In the above graph, when the average cluster dimensionality is less than 2% the SSPC produces the acceptable result similar to our projected clustering algorithm and SPPC works fine for higher average dimensionality. When the average cluster dimensionality is greater than 30 the HAPP works find and finds the cluster dimension without outliers.When the dimension is less than 20 to 2% the dimension detection produce acceptance result. The Proclus are less accurate when compared to Projected clustering using k-medoids

algorithm and SSPC when the average dimensions is less than 10% Fastdoc works fine when the average cluster dimensions greater than 70.

However outlier detections are removed. Projected clustering using K-medoids algorithm performs well when compared with SSPC, HARP, Proclus, and FASTDOC. Similarly the algorithm out forms SSPC, HARP and Proclus when the dimensions and size of data set increases.

CONCLUSION

We have proposed a robust distance-based projected clustering algorithm for the challenging problem of high-dimensional clustering, and illustrated the suitability of our algorithm in tests and comparisons with previous work. Experiments show that projected clustering in K-medoids algorithms provides meaningful results and significantly improves the quality of clustering when the dimensionalities of the clusters are much lower than that of the dataset. Moreover, our algorithm yields accurate results when handling data with outliers. The performance of projected clustering in K-medoids algorithms on real data sets suggests that our approach could be an interesting tool in practice. The accuracy achieved by projected clustering in K-medoids algorithms results from its restriction of the distance computation to subsets of attributes, and its procedure for the initial selection of these subsets. Using this approach, we believe that many distance-based clustering algorithms could be adapted to cluster high dimensional data sets.

REFERENCES

[1] R.Agrawal, J.Gehrke, D.Gunopulos and P.Raghavan, "Automatic Subspace Clustering of High Dimensional Data," Data Mining and Knowledge Discovery, vol.11, no.1, pp.5-33, 2005. doi:10.1109 /TKDE.2008.224.

[2] K. Jain, M. N. Mutry, and P.J. Flynn, "Data Clustering: A Review,"ACM Computing Surveys, vol. 31, no. 3, pp. 264–323, 1999. doi :10.1186/1471-2105-7S4-S10 3.

[3] K. Beyer, J. Goldstein, R.Ramakrishan, and U. Shaft, "When Is Nearest Neighbor Meaningful?," Proc. of the 7th International Conference on Database Theory, pp.217–235, 1999. doi:10.1145/1835449.1835482.

[4] H. Liu and L. Yu, "Toward Integrating Feature Selection Algorithms for Classification and Clustering," IEEE Trans.Knowledge and Data Eng., vol. 17, no. 3,pp. 1–12, 2005.doi: 10.1109/TKDE.2005.66.

[5] C.C. Aggarwal, C. Procopiuc, J. L. Wolf, P.S. Yu, and J.S. Park, "Fast Algorithm for Projected Clustering," Proc. ACM SIGMOD Conf.,pp.61–72,1999. doi:10.1109/ICDE.2009 .188. 1152.

[6] K.Y.L. Yip, D. W. Cheung, M. K.Ng and K. Cheung, "Identifying Projected Clusters from Gene Expression Profiles," Journal of Biomedical Informatics, vol. 37, no. 5, pp. 345–357, 2004. doi:10.1109/TKDE. 2008.162.

[7] K.Y.L. Yip, D.W. Cheng and M.K.Ng,"On Discovery of Extremely Low-Dimensional Clusters using Semi-Supervised Projected Clustering," Proc. ICDE, pp. 329– 340, 2005.doi.ieeecomputersociety.org /10.1109/ICDE.2005.96.

[8] C.C. Aggarwal and P.S. Yu,"Redefining Clustering for High Dimensional Applications," IEEE Trans. Knowledge and Data Eng., vol. 14, no. 2, pp. 210–225, 2002. doi:10.1023/A:1009769707641.

[9] K.Y.L. Yip, D.W. Cheng and .K. Ng, "HARP: A Practical Projected Clustering Algorithm," IEEE Trans. Knowledge and Data Eng., vol. 16, No. 11, pp. 1387–1397, 2004. doi: 10.1186/1471-2148- 8-116.

[10] C.M. Procopiuc, M. Jones, P.K. Agarwal, and T.M. Murali, "Monte Carlo Algorithm for Fast Projective Clustering," Proc.ACM SIGMOD, pp. 418 – 427, 2002. doi:10.1109/TKDE.2008.224.

[11] M. Lung and N. Mamoulis, "Iterative Projected Clustering by Subspace Mining," IEEE Trans. Knowledge and Data Eng., vol. 17,no.2,pp.176-189,Feb.2005.doi:10.1109/ TKDE.2005.29.

[12] E.K.K. Ng, A.W. Fu, and R.C. Wong, "Projective Clustering by Histograms," IEEE Trans. Knowledge and Data Eng., vol. 17,no.3,pp.369-383,Mar.2005.doi:10.1109/ TKDE . 2005.47.

[13] M. Bouguessa, S. Wang, and Q. Jiang, "A K-Means-Based Algorithm for Projective Clustering," Proc. 18th IEEE Int'l Conf. Pattern Recognition (ICPR '06), pp. 888-891,2006. doi:10.1109/TKDE. 2008.162.

[14] C.H. Cheng, A.W. Fu, and Y. Zhang, "Entropy-Based Subspace Clustering for Mining Numerical Data," Proc. ACM SIGMOD'99,pp.84-93,1999.doi:10.1109 /TKDE.2008.224 ...

[15] S. Goil, H. Nagesh, and A. Choudhary, "MAFIA: Efficient and Scalable Subspace Clustering for Very Large Data Sets," Technical Report CPDC-TR-9906-010, Northwestern,univ, 1999. doi:10.1234/12345678.

[16] K. Kailing, H.-P. Kriegel, and P. Kroger, "Density-Connected Subspace Clustering for High-Dimensional Data," Proc. Fourth SIAM Int'l Conf. Data Mining (SDM '04), pp. 246-257, 2004. doi:10.1109/TKDE.2008.224.

[17] L. Parsons, E. Haque, and H. Liu, "Subspace Clustering for High Dimensional Data: A Review," ACM SIGKDD Explorations Newsletter, vol. 6, no. 1, pp. 90-105,2004.doi.ieeecomputersociety.org/ 10.1109/TKDE.2006.106.

[18] K.Y.L. Yip, "HARP: A Practical Projected Clustering Algorithm for Mining Gene Expression data," master's thesis, The Univ.ofHongKong, 2004.doi:10.1109/TKDE.2004.74.

[19] K. Bury, Statistical Distributions in Engineering. Cambridge Univ. Press, 1998 . doi:10.1002/(SICI)1521.

[20] N. Balakrishnan and V.B. Nevzorov, A Primer on Statistical Distributions. John Wiley & Sons, 2003.

[21] R.V. Hogg, J.W. McKean, and A.T. Craig, Introduction to Mathematical Statistics, sixth ed. Pearson Prentice Hall, 2005.

[22] J.F. Lawless, Statistical Models and Methods for Lifetime Data. John Wiley & Sons, 1982.

[23] M. Bouguessa, S. Wang, and H. Sun, "An Objective Approach to Cluster Validation," Pattern Recognition Letters, vol. 27, no. 13, pp. 1419-1430, 2006.

[24] J.J. Oliver, R.A. Baxter, and C.S. Wallace, "Unsupervised Learning Using MML," Proc. 13th Int'l Conf. Machine Learning (ICML '96), pp.364-3721996.doi.ieeecomputersociety.org /10. 1109 /3.