A Clustering Algorithm for Predicting CardioVascular Risk

Thuy. T.T. Nguyen, Darryl. N. Davis.

Abstract—**Cluster analysis is one area of machine learning of particular interest to data mining. It provides for the organization of a collection of patterns, represented as a vector in a multidimensional space, into clusters based on the similarity of these patterns. Medical decision support is also of increasing research interest. Ongoing collaborations between cardiovascular clinicians and computer science are looking at the application of neural networks, and in particular clustering, to the area of individual patient diagnosis, based on clinical records. The cardiovascular domain is characterized as a mixture of continuous and discrete data. This limits the use of the K-means algorithm, which is widely used for partitioning clusters in data mining. This paper presents an improvement on the K-means algorithm (KMIX) and allows its application to the mixture of attribute types found in the cardiovascular domain.**

Index Terms— **Clustering; KMIX; K-means; dissimilarity; patient diagnosis.**

I. INTRODUCTION

Partitioning is a fundamental operation in data mining for dividing a set of objects into homogeneous clusters. Clustering is a popular partitioning approach. A set of objects are placed into clusters such that objects in the same cluster are more similar to each other than objects in different clusters according to some defined criteria. The K-means algorithm [7] is well used for implementing this operation because of its efficiency in clustering large data sets. However, K-means only works on continuous values. This limits its use in medical domains where data sets often contain Boolean, categorical, and continuous data. The traditional approach to convert categorical data into numeric values does not necessarily produce meaningful results where categorical attributes are not ordered. In this paper, we propose an algorithm, KMIX, which is improved from K-means in order to cluster mixed numerical and categorical data values. In the KMIX algorithm, a dissimilarity measure is defined that takes into account both numeric and

categorical attributes via the Euclidean distance for numerical features and the number of mismatches of categorical values for discrete feature. For example, assume that $d^N(X,Y)$ is the squared Euclidean distance between two objects X and Y over continuous features; and $d^C(X,Y)$ is the dissimilarity measure on categorical features in X, Y. The dissimilarity between two objects X, Y is given by the distance

$$d(X,Y) = d^N(X,Y) + d^C(X,Y).$$

The clustering process of the KMIX algorithm is similar to the K-means algorithm except that a new method is used to update the categorical attribute values of cluster. The motivation for proposing KMIX based on K-means is that KMIX can be used for large data sets, where hierarchical clustering methods are not efficient.

II. NOTATIONS AND DATA DOMAIN

Cluster analysis provides the means for the organization of a collection of patterns into clusters based on the similarity of these patterns, where each pattern is represented as a vector in a multidimensional space. Assume that X is a pattern (an observation or sample from a data set). X typically consists of m components, represented in multidimensional space as:

$$X = (x_1, x_2, \ldots, x_m) = (x_j)_{j=1,\ldots m}.$$

Each component in multidimensional space is called a feature (attribute). A data set includes n patterns X_i where $i \in [1, n]$ and $X_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,m})$. Hence, we have a $n \otimes m$ pattern matrix.

Note that the m features here may include continuous and discrete valued features. According to [1] - [2], because of the variety of feature types and scales, the distance measure (or measures) must be chosen carefully. It is most common to calculate the dissimilarity between two patterns using a distance measure defined on the feature space.

A. Similarity measurement of patterns

A similarity measurement is the strength of the relationship between two patterns given in the same multidimensional space. It can be represented as some function of their observed values such as

$$sim_{ij} = sim(x_i, x_j), \; i,j \in [1,n].$$

Similarity is regarded as a symmetric relationship requiring $sim(x_i, x_j) = sim(x_j, x_i)$ [8]. However, the dissimilarity measures

of patterns has been introduced as the complement of similarity measures, and so called distance measure. A list of dissimilarity measures can be seen in [9].

For continuous features, the most common used measure is the Euclidean distance between two patterns. This is very dependent upon the particular scales chosen for the variables [3].

The dissimilarity measure of two "continuous" patterns using Euclidean distance is given as:

$$dissim(x_i, x_j) = [D(x_i, x_j)]^2$$
$$= \sum_{k=1}^{m}(x_{ik} - x_{jk})^2, i, j \in [1, n_1], n_1 \le n \quad (1)$$

where D is Euclidean distance.

This means the dissimilarity of two patterns x_i and x_j is the square of the Euclidean distance between them.

For discrete features, the similarity measure between two patterns depends on the number of similarity values in each categorical feature [10]. This means the dissimilarity will be the number of different values of two considering pattern in each categorical feature. We can represent this dissimilarity in the following formula:

$$dissim(x_i, x_j) = d(x_i, x_j)$$
$$= \sum_{k=1}^{m}\theta(x_{ik}, x_{jk}) \quad i, j \in [1, n_2], n_2 \le n \quad (2)$$

where $\theta(x_{ik}, x_{jk}) = \begin{cases} 0 & if \ x_{ik} = x_{jk} \\ 1 & if \ x_{ik} \ne x_{jk} \end{cases}, \quad k = 1, 2, ..m; i, j \in [1, n_2]$

For binary features, the dissimilarity measures are calculated as either discrete features or continuous ones dependent on the interpretation of the provided binary data set.

B. Centre vectors

As the data feature set includes both continuous and discrete features (binary features are treated as continuous or discrete ones dependent upon domain knowledge), the centre vector will include 2 group components of continuous and discrete. Assume that the data feature set includes m features, where the p first features are continuous features and the $m-p$ remaining features are discrete. This means each pattern X in the space can be seen as

$$X=(x_{i1}, x_{i2}, ...x_{ip}, x_{ip+1}, x_{ip+2}, ... x_{im}).$$

Assume that Q is a centre vector for the data set C (C is a sub set of whole data set). So Q can be represented as

$$Q = (q_{j1}, q_{j2}, ... q_{jp}, q_{jp+1}, q_{jp+2}, ..., q_{jm}).$$

The task is to find p "continuous" component values, and $m-p$ "discrete" component values for vector Q_j.

For "continuous" component values, $\{q_{jk}\}_{k=1,..p}$ are the means of the kth feature in C [5].

For "discrete" component values, $\{q_{jk}\}_{k=p+1,..,m}$ are the set of $mode_k$, where $mode_k$ is the mode of the kth feature.

Definition 1: A vector Q is a mode vector of a data set

$$C = (X_1, X_2, ... X_c, c \le n)$$

if the distance from each vector X_i ($i \in [1, c]$) is minimized. This means

$$d(C, Q) = \sum_{i=1}^{c} d(X_i, Q)$$

is minimized. Huan [16] proved that this distance will be minimized only if frequency of value $q_j \in Q$ is maximized. This means the frequency of each value q_j in data set C, considered in terms of feature j needs to be greater or equal to the frequency of all different $x_{i,j}$ such that $x_{i,j} \ne q_j$ for the same feature ($j \in [1, m]$).

Hence we can choose the mode vector for the $m-p$ categorical components where each component value is the mode of that feature or the value which has biggest frequency value in that feature.

$$\{q_{jk}\}_{k=p+1,..,m} = mode_{k=} \{\text{"max freq"} Val_{Ck}\}.$$

C. Data domain

Data used in this paper is derived from a clinical site in Britain, and includes 341 patient's records in the cardiovascular domain. The data domain itself is inconsistent, and not immediately useable. The redundant attributes which are not relevant to the data-mining task or derived from other attributes, are eliminated. The continuous valued numerical attributes are transformed into the range *[0,1]* using the linear transformation method, *new=(original-minimum)/range*. Boolean data is transformed into discrete number text form. For example, attribute "*STATUS*" has "*yes*"/"*no*" values so the transformed values will be "*1*"/"*0*". This means the values are considered as text values in algorithm processing and will use the mode for centre vector instead the Euclidean distant for centre vector. Categorical attributes are divided into either Boolean; Discrete Categorical or Discrete Sub-Attribute Categorical (with Boolean values). For example, the "*CAROTID_DISEASE*" attribute takes values of "*Asx*", "*A-F*", "*CVA*" and "*TIA*", transformed a four-place Boolean string "*1000*"; "*0100*"; "*0010*"; and "*0001*" respectively. The finalised data set includes 341 records with 19 attributes' with 3 numerical attributes, and the rest of "categorical Boolean" ones.

The research project requires that a comparative audit of the data for different outcomes to be investigated. Patient parameters such as "*Patient Status*", and the combination of other risk outcomes, such as "*Heart Disease*" (HD), "*Diabetes*" (D), and "*Stroke*" (St) may all be used as outcome indicators for individual patients. Subsequently a new summary output attribute (*Risk*) is built based on the value for the combination of the main disease symptoms. For alternative outcomes the appropriate models are built based on different heuristic rules:

- Model 1 (CM32): Two outcome levels are defined as:

$\Sigma(Status, Combine) = 0 \rightarrow Risk = Low$

$\Sigma(Status, Combine) > 0 \rightarrow Risk = High$

- Model 2 (CM33): Similar to model 1 but divided into three levels of risk:

$\Sigma(Status, Combine) = 0 \rightarrow Risk = Low$

$\Sigma(Status, Combine) = 1 \rightarrow Risk = Medium$

$\Sigma(Status, Combine) > 1 \rightarrow Risk = High$

III. K-MIX ALGORITHM

The K-MIX algorithm is a four stage process:

Step 1: Initialise K clusters according to K partitions of data matrix.

Step 2: Update K centre vectors in the new data set (for the first time the centre vectors are calculated)

$Q_j = (q^N_{j1}, q^N_{j2}, ..., q^N_{jp}, q^C_{jp+1}, ..., q^C_{jm}), j \in [1, K].$

where $\{q^N_{jk}\}(k \in [1,p]) = \{mean^N_{jk}\}$ (mean of kth feature in cluster j);

and $\{q^C_{jk}\}_{k=p+1,...m} = \{mode^C_{ji}\}$ (max freq Value in feature k^{th} in cluster j).

Step 3: Update clusters:

Calculate the distance between X_i in i^{th} cluster to K centre vectors:

$d(X_i, Q_j) = d^N(X_i, Q_j) + d^C(X_i, Q_j); j=1,2,...k;$

where $d^N(X_i, Q_j)$ is calculated according to (1),
and $d^C(X_i, Q_j)$ is calculated according to (2).

Allocate X_i into the nearest cluster such that $d(X_i, Q_j)$ is least.
Do this for whole data set, and save them to the new data set with K new centre vectors.

Step 4: Repeat step 2 and 3 until no change in the distance between X_i and new K centre vectors is seen.

IV. EXPERIMENTS

Before running on the target data domain, many experiments were run with data derived from UCI repository of databases as used by the machine learning community for the empirical analysis of machine learning algorithms [4]. The clustering accuracy for measuring the clustering results was computed as follows. Given the final number of clusters, K, clustering accuracy r was defined as:

$$r = \frac{\sum_{i=1}^{K} a_i}{n}$$

where n is the number of samples in the dataset, a_i is the number of data samples occurring in both cluster i and its corresponding class, which had the maximal value. Consequently, the clustering error is defined as $e = 1 - r$. The lower value of e suggests the better clustering result.

The experimental data sets are Small Soybean data set [12] with 47 samples and 35 attributes, in 4 class distributions, Votes data set [13] containing 16 keys votes with all categorical data types in 435 records (included meaningful missing value records "?"), and 2 output classes labelled to 168 republicans and 267 democrats. This algorithm is also used for experiments

with Zoo small data set [4]. It has 101 records distributed in 7 categories with 18 attributes (included 15 Boolean, 2 numerical, and 1 unique attribute(s)). The fourth experiment for KMIX is with the Wiscons Breast Cancer data set [4]. It contains 683 records by removing 16 missing value records. The data set includes 9 numerical attributes divided into 2 class label of "2" or "4". The comparison results can be seen in Table I below.

Table I: Comparisons to publication results.

Data set	Publication results	KMIX results
Soy Bean	0.11^1 ~	0.07
Votes	$0.132^{2,3}$	0.141
Zoo small	0.166^2	0.151
WBreast Cancer	0.03^4; 0.132^2	0.03

From Table I, KMIX performs as well as other published results for the Soy Bean[1] [11]; Votes[2,3] [14], [6], and Wisconsin Breast Cancer[4] [17] data sets. For the latter, the KMIX result of 0.03 compares favourably compared to 0.132^2 [14]. Further more, this algorithm solves problems associated with the mixture of categorical and discrete data; a common feature of medical data domains.

The next experiment used the K-means algorithm (in the WEKA package) for comparison on the cardiovascular data. The results show sensitivity defined as the frequency of correctly classified positive Medium or (Very) High Risk, and specificity rate defined as the frequency of correctly classified negative- (very) Low Risk as defined by model CM32.

Table II: Clustering results of K-means and KMIX algorithms.

Algorithm	Risk	C1 (High)	C2 (Low)	Sen	Spec
K-means	High	36	21	0.15	0.82
	Low	168	116		
KMIX	High	35	22	0.25	0.89
	Low	107	177		

From Table II, the rates of sensitivity in two algorithms are small (0.15, 0.25 for K-means, and KMIX). However Table II clearly shows the advantage of KMIX over K-means. The CM32 and CM33 results models are validated using supervised neural network techniques such as Support Vector Machine (SVM), and Multilayer Perceptron (MLP). Tables III and Table IV below show the results in using these techniques with 10 times cross validation test set and near optimal parameters

Table III: Using NN results of CM32

	Cluster	C1	C2	Sensitivity	Specificity
MLP	C1H	121	21	0.90	0.90
	C2L	13	186		
SVM	C1H	120	22	0.85	0.89
	C2L	22	177		

Table IV: Using NN results of CM33

	Cluster	C1	C2	C3	Sen	Spec
MLP	C1M	75	4	6	0.98	0.96
	C2H	5	112	0		
	C3L	4	0	135		
SVM	C1M	71	0	14	0.98	0.91
	C2H	8	109	0		
	C3L	3	1	135		

Overall, using MLP shows better results using the measures shown in Table IV (0.98). More over, in Table IV the rate of sensitivity is higher than the rate of specificity. This means the prediction of NN in the side of high level risk (including Medium, and High risk) is appropriate.

From Table III and Table IV the rate of sensitivity and specificity are quite high (more than 85%) in particular more than 91% in Table IV. This validates that the clustering results from using KMIX are appropriate for the cardiovascular data domain.

V. Conclusion and further works

The proposed algorithm KMIX is compared to publicised results and compares favourably. Further more, because this algorithm was developed for use with a specific medical data domain, it meets the need to be adaptable to the data set which contains a mixture of numerical, categorical, Boolean data types. By using the clustering algorithm, new outcomes for CM32, CM33 risk models are generated. These models can be evaluated through the use of the neural networks techniques, such as MLP and SVM, and indicate the level of performance for the KMIX clustering algorithm. From Table III, IV we can see that the boundary of each cluster is not clear. For example, in Table III, C1H (number of high patients) fell in cluster C1, and C2. However no high risk patients were placed in the cluster most closely associated with low risk patients (C3 in Table IV). Fortunately in the cardiovascular domain, the clinician's interest is in the rate of potential risk of patients (both medium and high risk). These cases can be reported in the rate of sensitivity. From Table III and Table IV these rates are quite high. Hence it can be seen that the performance of the KMIX clustering algorithm is appropriate.

Further work on improving this algorithm will include the random ordering of samples. This will hopefully see an improvement in the sensitivity rate. Furthermore weights might be applied in order to indicate the level of importance for attributes in the feature set for the data domain. This increases the regression of the algorithm in a short time period with the hope of more accurate modes in term of having the most importantly attribute values for centre vectors. Finally alternative data domains will be investigated to determine the usefulness of using this algorithm for medical data domains in general.

References

[1] A. K. Jain, & R. C. Dubes, "*Algorithms for Clustering Data*", Prentice-Hall advanced reference series. Prentice-Hall, Inc., Upper Saddle River, NJ, 1988.

[2] A. K. Jain, "*Data clustering: A review*". ACM Computing Surveys, Vol. 31, No. 3, 1999.

[3] B. S. Everitt, "Cluster Analysis", 3rd ed. John Wiley & Son, New York, 1994.

[4] C. J. Merz, & P. Merphy, (1996). UCI Repository of Machine Learning Database. Available: http://www.ics.uci.edu/~mlearn/MLRepository.html

[5] D. J. Hand, "*Discrimination and Classification*". John Wiley & Sons, 1981.

[6] H. Zengyou, X. Xiaofei, D. Shengchun, "TCSOM: Clustering Transactions Using Self-Organizing Map". Springer, Neural Processing Letters 22, 2005, pp.249–262.

[7] J. B. MacQueen, "*Some Methods for Classification and Analysis of Multivariate Observations*", In Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, 1967, pp. 281-297.

[8] J. C. Gower, "*Classification, geometry and data analysis*", In Classification and Related Methods of Data Analysis (H.H. Bock, ed). Elsevier, North-Holland, Amsterdam, 1988.

[9] J. C. Gower, "*Measure of similarity, dissimilarity and distance*". In Encyclopedia of Statistical Sciences, Vol. 5. John Wiley & Son, New York, 1985.

[10] L. Kaufman, & P. J. Rousseeuw, "*Finding Groups in Data—An Introduction to Cluster Analysis*". Wiley, 1990.

[11] M. S. Ohn, H. Van-Nam, and N. Yoshiteru, "*An Alternative extension of the K-means algorithm for clustering categorical data*". International Journal Mathematic Computer Science, Vol. 14, No 2, 2004, pp: 241-247.

[12] R.S. Michalski, and R. L. Chilausky, "*Learning by Being Told and Learning from Examples: An Experimental Comparison of the Two Methods of Knowledge Acquisition in the Context of Developing an Expert System for Soy- bean Disease Diagnosis*". International Journal of Policy Analysis and Information Systems, 4(2), 1980, pp.125-161.

[13] S. Jeff, "Concept acquisition through representational adjustment". Doctoral dissertation, Department of Information and Computer Science, University of California, Irvine, CA, 1987.

[14] S. K. Shehroz, and K. Shri, "*Computation of Initial Modes for K-modes clustering Algorithm using Evidence Accumulation*". 20th International Joint Confference on Aftificial Intelligence (IJCAI-07), India, 2007.

[15] WEKA software (University of Waikato, New Zealand, version 3.4.5). Download free at : http://www.cs.waikato.ac.nz/~ml/weka/index.html.

[16] Z. Huan, "*Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values*", Data Mining and Knowledge Discovery Vol. 2, Issue 3. Kluwer Academic Publishers, 1998.

[17] F.Camastra, A.Verri "*A novel Kernel Method for Clustering*", *IEEE Transaction on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 27, no. 5, May 2005, pp. 801-805.