REDIC K –Prototype Clustering Algorithm for Mixed Data (Numerical and Categorical Data)

Khyati R. Nirmal, K.V.V.Satyanarayana

Abstract*: In the unsupervised learning Clustering is the task to find hidden structure without any prior knowledge of data and derive the interesting patterns from the given data objects. Furthermost the real word dataset is the combination of numerical and categorical data attributes. The K-prototype Clustering algorithm is widely used to group the mixed data because of ease of implementation. The efficiency of the algorithm depends on the selection strategy of initial centroids, and here the initial centroids are randomly selected. Other constraint of this algorithm is to provide number of clusters as input, which requires the domain specific knowledge. Inappropriate choice for number of clusters will affect the complexity of algorithm. In this paper the REDIC (Removal Dependency on K and Initial Centroid Selection) K-prototype clustering algorithm is proposed which will eliminate the dependency on input parameter and creates the cluster using incremental approach. Here as a replacement for the bit by bit comparison of categorical attributes, the frequency-based method is used to calculate the dissimilarity measurement between two categorical instances. Experiments are conducted with standard datasets and the results are compared with traditional K-prototype algorithm. The better results of REDIC K -prototypes clustering algorithm proves the efficiency of algorithm and removes the dependency on initial parameter selection.

Index Terms: Cluster Analysis; K- Prototype Clustering; Initial Centroid; Number of Cluster; Frequency based Similarity Measurement.

I. INTRODUCTION

As the widespread approach in knowledge discovery the clustering targets to grouping of similar data by measuring the distance between instances. Clustering is belonged to unsupervised learning means the clusters are shaped without any prior knowledge of data, only distance measurements contribute while grouping of data. The smaller the distance between the instances are considered as similar instances which are grouped into same cluster and the growing in the distance between the instances will keep the two instances into separate clusters. In a simple way clustering minimizes the intra cluster distance and maximizes inter cluster distance. Clustering algorithms are having several categories like: partitional clustering, hierarchical clustering, density-based clustering, grid-based clustering and model-based clustering.

In partitional based Clustering the K means is the widely accepted algorithm due to its flexibility, proficiency, realistic attainment, ease of implementation. In K means clustering the Euclidean distance measurement method is used to compute the distance between two instances, which compacts with the numerical data only [1]. To handle the categorical data using k means clustering algorithm the preprocessing of data like discretization or coding is introduced which performs the data transformation steps and convert categorical data into numeric data. It increases the complexity of algorithm [2] and also introduces the probabilities of information loss from the given instances [3].

To work with categorical data the Hung has proposed K-Modes Clustering algorithm. In K-Modes Clustering algorithm major three modifications have been introduced. Firstly the Euclidean measurement method is replaced with the simple matching dissimilarity function. Secondly to represent the centroid of cluster the Modes of the cluster are calculated instead of the Mean values of clusters. Finally to update the centroid for each iteration the frequency based methods are used. The K- Mode clustering produces locally optimal solution which depends on initial centroid selection. Many researches have been carried out and many are in underway to improve the accuracy of algorithm by proposing the different method of initial centroid selection for K-Mode Clustering. The further limitation is to select the number of cluster at the initial stage of algorithm which requires the in-depth domain knowledge and the perfect clue to expect the better outcome. K-Mode Clustering is designed for the categorical data only, it is not recommended for mixed data [4].

The paper is organized as follows: The K Prototype Clustering Algorithm and related work while choosing K-prototypes clustering algorithm is specified in the second section, in third section the REDIC Algorithm is proposed and in the last section the experimental results are demonstrated along with the description of dataset and accuracy measurement used.

II. THE K PROTOTYPE CLUSTERING ALGORITHM

Consider the given Dataset D is of n instances, that is $(D_1, D_2, \ldots, D_n)$. Each Instances Di is combination the Attributes, $A_1, A_2, A_3 \ldots A_p$, $A_{p+1}, A_{p+2} \ldots A_{p+q}$, where $A_1, A_2, A_3 \ldots A_p$ are Numeric Attributes and A_p, $A_{p+1}, A_{p+}2 \ldots A_{p+q}$ are categorical attributes. The definition of dissimilarity measurement between two instances Di and Dj is given as:

$$\text{Dist}(D_i, D_j) = \text{Ndist}(D_i, D_j) + \gamma \ \text{Cdist}(D_i, D_j) \quad (1)$$

Ndist is the numerical distance between two instances Di and Dj, which is calculated by using equation 2.

Manuscript Received on March 20, 2019.

Khyati R. NIrmal, Research Scholar, Department of CSE, KoneruLakshmaiah Education Foundation, Green Fields, Vaddeswaram, (A.P), India.

K.V.V.Satyanarayana, Department of CSE, Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram (A.P), India.

$$\text{Ndist}(D_i, D_j) = \sqrt{\sum_{t=1}^{p} (A_{it} - A_{jt})^2} \quad (2)$$

Cdist is the categorical distance between two instances D_i and D_j, which is calculated by using equation 3 and 4

$$\text{Cdist}(D_i, D_j) = \sum_{t=P+1}^{q} f(A_{it}, A_{jt}) \quad (3)$$

$$f(A_{it}, A_{jt}) = \begin{cases} 1, A_{it} = A_{jt} \\ 0, A_{it} \neq A_{jt} \end{cases} \quad (4)$$

Algorithm 1 K Prototype Clustering Algorithm

Input: The number of cluster k, value of γ
Output: A set of k Clusters with allotted instances

Step 1. Randomly k centroids (C_1, C_2, C_n) are initialized.
Step 2. Distance between each instance of $(D_1, D_2, \ldots, D_n)$ and each centroids (C_1, C_2, C_n) is calculated using dissimilarity measurement equation (1). Each Instance D_i is allocated to cluster having minimum distance.
Step 3. For each cluster, the centroids are recalculated. The average of each numerical attributes is calculated and the categorical attributes with the highest frequency of occurrence is selected which is considered as updated centroid value
Step 4. The step 1 to 3 are repeated until there will be no change in allotment of instances to cluster.
Step 5. If the algorithm is reached to the maximum number of iterations the algorithm will be stopped else it will jump to Step 2.

Here the additional parameter γ is introduced, which avoids the preferring either type of attribute. To dominate the numeric attributes, value of γ should be increase and to dominate the categorical attributes the value of γ should be decrease. This cannot be a generalized rule; user knowledge requires for deciding the value of γ. This parameter is measured as obstacle while choosing K-prototypes Clustering algorithm [5]. Along with the eliminating of this obstacle, work is in process to remove the dependency of selecting the initial value of centroid and number of clusters at first.

This section outlines the recent progresses over the K-prototypes method that is proposed to address the initial centroid selection and number of cluster decision for mixed data set.

- The fuzzy K-prototype clustering is proposed by JinchaoJi and et.al which has offered co-occurrence based dissimilarity measurements. This measurement implies for any type of data but dependency is on fuzzy parameter [6].
- In 2013 the improved K-prototypes clustering has been proposed which has defined strategies to define fuzzy rameter for mixed data. This fuzzy parameter should be decided well in advance and in the real word data set the fuzzy parameter has diverse properties which are tough to choose the significant fuzzy parameter. [7]
- A Novel Mixed Values K-prototypes Algorithm is proposed by Ahmed et.al in 2014. Clustering of multi valued categorical variable is done by bag-of-words based on the Apriori method. This algorithm focuses on the dissimilarity measurement note on the number of cluster and initial centroid selection. [8]
- In 2015 An Improved Clustering Algorithm for mixed attributes data based on K-prototypess algorithm is proposed by Xuan Chen. The initial points are calculated using grouping and averaging method. The algorithm has good stability and validity but in order to improve the flexibility and stability of the algorithm the number of cluster has to be defining priory which is not proposed in algorithm. [9]
- An equi-biased K-prototypes algorithm for clustering mixed-type data is proposed in 2018 by RAVI Sankarsangam and HariOm. Here a new dissimilarity measure the Minkowski distance metric, has been adopted for numerical attributes and for categorical attributes the weightage hamming distance has been adopted. The issues of number of clusters and initial centroid selection are not considered [10]

In this paper the Removal Dependency on K and Initial Centroid (REDIC) K-prototypes clustering algorithm is proposed, which attempts to take away the stated obstacles. Along with the removing the dependency of initial parameters, the definition of calculating the dissimilarity between the categorical attributes is slightly reformed.

III. PROPOSED ALGORITHM: REDIC K PROTOTYPE CLUSTERING ALGORITHM

The REDIC K Prototype Clustering is proposed with three major variations: 1. Select the initial centroids by calculating most significant attributes. 2. Incremental approach for deciding number of clusters. 3. Frequency based dissimilarity measurement between the categorical attributes.

- Let I is the set of given instances, with m attributes and n instances, where, m = p + q

  1, 2, . . . , p =number of numerical attributes

  p + 1, p + 2 . . . , p + q(m) = number of categorical attributes

- Set of Instances I = {DI ∪−T I}
- DI is set of numerical instances having p attributes and n instances
- TI is set of numerical instances having q attributes and n instances

The algorithm 2 is proposed to select initial centroid by calculating row factor for each row. Row Factor RFi for particular value is addition of Numerical Row Factor NRF (i) plus the Categorical Row Factor CRF (i). To calculate the Categorical Row Factor the term Frequency of Occurrence (OCF) is introduced.

For any particular categorical value ti , OCF (t) is the number of times that categorical value is repeated for particular attribute. After calculating CRF (i) for each instance i , the instances having minimum and maximum value is selected as initial centroid.

Algorithm 2 Initial Centroid Selection by calculating Row Factor for each instance

Input: The set of given instances, with m attributes and n instances
Output: Initial k instances of Centroids

Step 1. **For each row** i **in** I do

$$NRF(i)=d_{i1}+d_{i2}+\cdots+d_{ip}.$$

$$OCF(i)=\frac{OCF(t_{i,p+1})}{q}+\frac{OCF(t_{i,p+2})}{q}+\frac{OCF(t_{i,p+q})}{q}$$

$$RF_i = NRF(i) + CRF(i)$$

End For
Step 2. k= total number of min RFi and max RF i
Step 3. **return** set of instances having min RF and max RF

The algorithm 3 is used to calculate the distance between two instances Ix and Iy . The Numerical Distance ND(Ix , Iy) is calculated using Euclidean Distance Formula. Here for Categorical Distance CD(Ix , Iy) , instead of Equation 3 and 4 ,the OCF based formula is proposed.

Algorithm 3 Distance Calculation between two instances

Input: Two instances$\{d_{x1}, d_{x2}, \ldots, d_{xp}, t_{xp+1}, t_{xp+2}, \ldots, t_{xp+q}\ (t_{xm})\}$ and $\{d_{x1}, d_{x2}, \ldots, d_{xp}, t_{xp+1}, t_{xp+2}, \ldots, t_{xp+q}\ (t_{xm})\}$
Output: Distance between two instances dist (Ix ,Iy)
Begin
Step 1. $ND(I_x, I_y)\sqrt{(d_{x1}-d_{y1})^2+(d_{x1}-d_{y1})^2+(d_{xp}-d_{yp})^2}$
Step 2. $CD(I_x, I_y) = CRF(I_x) - CRF(I_y)$
Step 3. dist $(I_x, I_y) = ND(I_x, I_y) + CD(I_x, I_y)$

returndist (I_x, I_y)

After selecting the initial centroids using algorithm 1, the initial k clusters are formed. The distance between every instance to every centroid c1 , c2 , . . . , ck is calculated and the instance is allocate to cluster having minimum distance. For refinement of cluster the DV value is calculated for each cluster using the equation 5

$$DV_i = |I_{i1}-C_{k1}| + |I_{i2}-C_{k2}| + \cdots + |I_{ip}-C_{kp}| + |OCF(I_{ip+1}) - OCF(C_{kp+1})| + |OCF(I_{ip+2}) - OCF(C_{kp+2})| + \ldots + |OCF(I_{im}) - OCF(C_{km})|$$

(5)

Consider that in particular cluster Ck , there are l instances. The DVi is calculated for all the l instances, and the minimum value among all will be considered as δk . Algorithm 4 is proposed to calculate the δk value for each cluster.

Algorithm 4 Delta value calculation for cluster Ck

Input: For particular Cluster k , all instances in that cluster $I_1, I_2 \ldots I_k$

Output: Distance between two instances dist (I_x ,I_y)
Step 1 $DV_i = \phi$
Step 2 **for each instance** i **in**C_k**do**
calculate the DV_i
$DV_i = DV_i \cup DV_i$
End for
Return $\delta_k = \min\{DV_1, DV_2, \ldots DV_k\}$

Algorithm 5 REDIC K Prototype Clustering Algorithm

Input : Set of Given instances I with m attributes and n instances
Output: a set of k Clusters with allotted instances

Begin:

Step 1. Calculate Row Factor RFi for each Row.
Step 2. Select and count minimum and maximum value among all row
Step 3. Initialize the count as number of cluster k and create initial cluster C = {C_1 , C2,….,C_k}
Step 4. Calculate centroid for each Cluster
Step 5. Calculate the distance between each instance to each cluster dist (I_i ,C_k) and allocate the instance to cluster Ck having minimum distance
Step 6. Calculate delta factorδ_k for each cluster Ck for refinement of cluster
Step 7. Refine the cluster by comparing dist (I_i ,C_k) with δ_k. If dist $(I_i, C_k) > \delta_k$ then remove the non-promising instance Ii from cluster Ck.
Step 8. Increment the k by 1 and create new Cluster
Step 9. Allocate the non-promising instance identified in step 7 to the newly created cluster.
Step 10. Repeat Step 4 to 9 until all instances are allotted to appropriated cluster or no change in clusters.
Step 11. Return value of k (number of cluster) generated by algorithm

IV. PERFORMANCE ANALYSIS

A. In this section, the initial dataset, Performance Measurement Indices and Results are discussed.

A. Dataset

For evaluating the performance of REDIC-K prototype clustering algorithm post-operative patient dataset, Australian credit dataset, German credit dataset and Statlog (Heart) dataset are used. This dataset are openly available on UCI repository. The basic description of these four datasets are specified in below table.

TABLE 1
DESCRIPTION OF STANDARD DATASET

Dataset	No of Instances	Total Attributes	Numerical Attributes	Categorical Attributes
Post-operative patient	90	8	1	7
Australian credit Data Set	690	15	6	9
German credit Data Set	1000	20	7	13
Statlog (Heart) Data Set	270	13	9	4

In Post-Operative Dataset the decision is separated into 3 classes that is ADM-DECS (discharge decision): I class means for the patient sent to Intensive Care Unit), S Class stands for patient prepared to go home and A Class of patient sent to general hospital floor. In Australian credit Data Set the decision class is formerly separated into two classes + (positive) and - (negative). In Germen Credit Dataset customer is classified as good or bad as per the attributes value. In Statlog (Heart) Data Set the variable to be predicted is Absence or presence of heart disease.
Along with this the two more synthesis datasets are used, the description of this are given in below table.

TABLE II
DESCRIPTION OF SYNTHESIS DATASET

Dataset	No of Instances	Total Attributes	Numerical Attributes	Categorical Attributes
Synthesis Data Set 1	7	3	1	2
Synthesis Data Set 2	8	3	2	1

Synthesis Data Set 1, the predictor variable is categorized in 3 divisions: F,P and N .For Synthesis Data Set 2, the decision class is classified in 3 classes: M,H and L

B. Performance MEasurment Indices

The performance measurement indices are broadly divided into two categories: External Validation Indices and Internal Validation Indices. External Indices compare the clustering result with ground truth that is reference model. In case of Internal Indices, the ground truth or the prior knowledge is not available so, it directly examines the clustering result by comparing the structure of clusters. External Indices compare that a clustering is similar to a partition or not. Here the labeled dataset work as ground truth. The key advantage with this type of indices is that it is independent of the examples or cluster description. For that reason, it can be applied to measure the performance of any clustering algorithm. In this paper the three external Performance Measurement Indices are considered: Rand Index, Jaccard Index, Rand Coefficient, Folkes and Mallow index.

Consider the Clustering result set C {c1, c2, c3, …,cn} and the set of actual distribution of instances in ground truth data G {g1,g2,g3…,gn}. Here the computation is based on four factors. Here α and βrepresent the vectors of set C and G respectively.

1) $a = |SS|, SS = \{(Ri , Rj)|\alpha i = \alpha j,\ \beta i = \beta j,\ i<j\}$,
   It indicates that the two instances in the same cluster in clustering result and same classes of labeled data
2) $b = |SD|, SD = \{(Ri , Rj)|\alpha i = \alpha j,\ \beta i \neq \beta j,\ i<j\}$
   It indicates that the two instances in the same cluster in clustering result, but different classes of labeled data (
3) $c = |DS| , DS = \{(Ri , Rj)|\alpha i \neq \alpha j,\ \beta i = \beta j,\ i<j\}$
   It indicates that the two instances in the same classes of labeled data but different clusters in clustering result
4) $d = |DS|, DS = \{(Ri , Rj)|\alpha i \neq \alpha j,\ \beta i \neq \beta j,\ i<j\}$

It indicates that the two instances in the same classes of labeled data but different clusters in clustering result.

Rand Index: $RI = \frac{(a+d)}{a+b+c+d}$

Jaccard coefficient: $JC = \frac{a}{a+b+c}$

Folkes and Mallow index: $FMI = \sqrt{\frac{a}{a+b} \cdot \frac{a}{a+c}}$

The value of above performance indices ranges from 0 to 1. The 0 value of RI indicates that no pair of instances categorized in the same way under both clustering result and ground truth of class labels. The 1 value for RI indicates the data clustering are exactly same as class of labeled data hence it is identical clustering [11]. The increasing value for these indices indicates the better performance of algorithm.

C. Results

To measure the performance of REDIC K-prototype clustering algorithm, it is compared with the result of standard K-Prototype Clustering Algorithm. The REDIC K-prototype Clustering algorithm is implemented in Java language and the Performance Indices are calculated. The performance indices for the standard K-prototype clustering algorithm are derived using R Studio, by considering the following constraints:

- The ϒ value of Standard K-prototype clustering algorithm is set to default value.
- Here the centroids are randomly selected by standard K-Prototype Clustering algorithm.
- The number of clusters is again used depended; here the value of k is selected by refereeing the number of classes in ground truth dataset. So, it is tricky to specify the number of cluster if the prior information is not available or the user is not domain expert.

The comparisons for 6 datasets for 3 performance indices are shown in following tables.

According to Rand Index analysis in table 3, the performance of REDIC K- Prototype clustering yields consistent and slightly improves results for Post-Operative Patient Dataset and Australian Credit Dataset.

TABLE III

COMPARISON FOR RAND INDEX

Rand Index		
Data Set	**Standard Algorithm**	**Proposed algorithm**
Australian Credit DataSet (crds)	0.49	0.49
Germen Credit DataSet(creditg)	0.48	0.42
Post Operative Patient Dataset	0.34	0.49
Statlog Heart	0.2	0.49

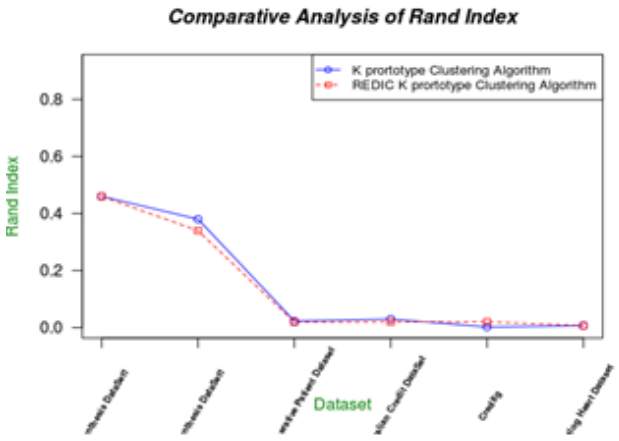

Fig.1 Comparison for Rand Index

From the Table-4 based to Jaccard Index, the performance of REDIC K- Prototype clustering yields consistent and better results for almost all data sets.

TABLE IV

COMPARISON FOR JACCARD COEFFICIENT

Jacccard Coefficient		
Data Set	**Standard Algorithm**	**Proposed algorithm**
Synthesis DataSet1	0.57	0.85
Synthesis DataSet1	0.57	0.75
Post Operative Patient Dataset	0.43	0.56
Australian Credit DataSet	0.54	0.55
Germen Credit DataSet(creditg)	0.51	0.69
Statlog Heart	0.39	0.41

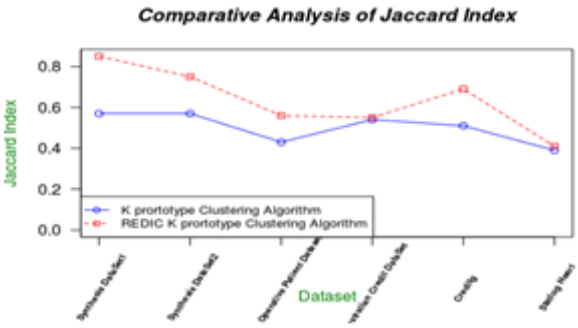

Fig.2 Comparison for Jaccard Index

In case of F-Measure, the value 1 indicates that the data clusters are exactly same and so the increase in the values of these measures proves the better performance. Based on this, the results of REDIC K- Prototype clustering is more promising than K-Prototype algorithm for all datasets represented in Table 5.

TABLE V

COMPARISON FOR FOLKES AND MALLOW INDEX

Folkes and Mallow index		
Data Set	**Standard Algorithm**	**Proposed algorithm**
Synthesis DataSet1	0.75	0.92
Synthesis DataSet1	0.75	0.86
Post Operative Patient Dataset	0.65	0.75
Australian Credit DataSet	0.73	0.74
Germen Credit DataSet(creditg)	0.77	0.83
Statlog Heart	0.62	0.64

Along with the promising results in case of Similarity indices, the number of clusters is intended by REDIC K-Prototype clustering using incremental approach. This measure removes the dependency on user for deciding number of cluster as a prerequisite of an algorithm.

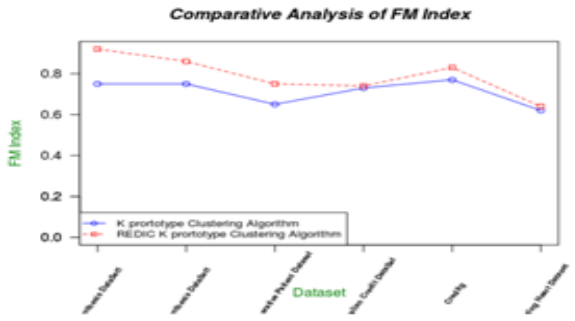

Fig. 3 Comparison for FM Index

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

V. CONCLUSION

This paper proposed REDIC K-Prototype Clustering algorithm by incorporating 3 major variation to improve the result of cluster analysis. 1. Initial Centroids selection by row factor, 2. The incremental approach to decide value for number of clusters K, 3. Frequency based dissimilarity measurement for categorical attributes. The proposed algorithm has been tested on the four benchmark data sets which include both numeric and categorical attributes. It is proved that the performance of the proposed algorithm is superior to the performance of K-Prototype clustering algorithms. The proposed algorithm has also taken away the obstacle of selecting the number of clusters as a prior requirement of an algorithm. In future for performing the cluster analysis of large dataset, the REDIC K-Prototype algorithm is intended to migrate on Map-Reduce paradigm, that will also speed-up the performance of algorithm.

REFERENCES

1. J. a. P. J. a. K. M. Han, Data mining: concepts and techniques, Elsevier, 2011.
2. Ralambondrainy, "A conceptual version of the K-means algorithm," Pattern Recognition Letters, pp. 1147--1157, 1995.
3. F. a. F. H. a. P. J. a. R. R. Wang, Empirical Comparative Analysis of 1-of-K Coding and K-Prototypes in Categorical Clustering, Dublin Institute of Technology, 2016.
4. Huang, "Extensions to the k-means algorithm for clustering large data sets with categorical values," Data mining and knowledge discovery, pp. 283-304, 1998.
5. J. Jinchao, P. Wei, C. Z. H. Xiao and W. Zhe, "A fuzzy k-prototype clustering algorithm for mixed numeric and categorical data," Knowledge-Based Systems, pp. 129-315, 2012.
6. J. Ji, i. T. Ba, C. Zhou, C. Ma and W. Zhe, "An improved k-prototypes clustering algorithm for mixed numeric and categorical data," Neurocomputing, pp. 590-596, 2013.
7. A. a. G. C. a. R. D. Najjar, "A novel mixed values k-prototypes algorithm with application to health care databases mining," Computational Intelligence in Healthcare and e-health (CICARE), 2014 IEEE Symposium , pp. 159--166, 2014.
8. C. Xuan, "An improved clustering algorithm for mixed attributes data based on k-prototypes algorithm}," 2015 10th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA), , pp. 396--399, 2015.
9. R. S. a. O. H. Sangam, "An equi-biased k-prototypes algorithm for clustering mixed-type data," Indian Academy of Sciences, p. 37, 2018.
10. S. a. W. D. Wagner, Comparing clusterings: an overview, 2007.
11. D. a. K. T. E. Dheeru, "UCI Machine Learning Repository," 2017. [Online].