Review of a creative clustering algorithm about mixed data set

1. Abstract

This review is based on the research paper “Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number” which has revealed an efficient clustering method applied on the mixed data set. The mixed data is composed of numerical and categorical attributes. Otherwise. the traditional clustering algorithm is always applicable to the data set which has purely numerical or categorical features. Hence, this paper develops a general clustering framework and a unified similarity metric adapted to complicated attributes. The paper is composed of three significant parts briefly: **1) A unified similarity metric** can be simply applied to the categorical, numerical, and mixed data. **2) An iterative clustering algorithm (OCIL)** and its experimental performance. **3) A penalized competitive algorithm** aimed to reject the redundant clusters. Hence, we can get a more appropriate k number.

2. Unified similarity metric

The most commonly used clustering algorithms may be k-means and k-modes which have the similar processes but different definitions of cluster’s center $C_{j.}$ The k-means uses the average distance to find the certain center of a cluster that is more appropriate for numerical data. On the contrast the k-modes determines the center by the frequency of the attributes which is adapted to categorical data. And the k-prototype which this paper used is the combine of the two algorithms above.

The most significant part of the algorithm is the calculation of the similarity or distance between the data sample and the cluster. This paper used a unified method to define the similarity for the mixed data, it calculated the similarity of the categorical and numerical attributes respectively. Hereafter, we transform the different similarity into a unified metric just like those hereinafter.

Suppose the mixed data $\mathbf{x}_\mathbf{i}$ with **d** different attributes consists of $\mathbf{d}_\mathbf{c}$ categorical attributes and $\mathbf{d}_\mathbf{u}$ numerical attributes, and the different clusters is denoted as $C_1, C_2, ..., C_K$. Because the numerical attributes are always treated as a vector and handled together, the similarity between the sample $\mathbf{x}_\mathbf{i}$ and the clusters is displayed as the equation below:

$$s(\mathbf{x}_\mathbf{i}, C_j) = \frac{d_c}{d_f}\sum_{r=1}^{d_c}\frac{1}{d_c}s\left(x_{ir}^c, C_j\right) + \frac{1}{d_f}s\left(\mathbf{x}_\mathbf{i}^\mathbf{u}, C_j\right) = \frac{d_c}{d_f}s\left(\mathbf{x}_\mathbf{i}^\mathbf{c}, C_j\right) + \frac{1}{d_f}s\left(\mathbf{x}_\mathbf{i}^\mathbf{u}, C_j\right)$$

The object-cluster similarity on categorical part can be given by

$$s\left(\mathbf{x}_\mathbf{i}^\mathbf{c}, C_j\right) = \sum_{r=1}^{d_c}\left(\frac{H_{A_r}}{\sum_{t=1}^{d_c} H_{A_t}} \cdot \frac{\sigma_{A_r} = x_{ir}^c\left(C_j\right)}{\sigma_{A_r} \neq NUL_L\left(C_j\right)}\right)$$

And the H means the entropy metric which presents the weight of each categorical attribute. **The introduction of entropy is an important feature of the OCIL algorithm**. It is modified with:

$$H_{A_r} = -\frac{1}{m_r}\sum_{t=1}^{m_r} p(a_{rt}) \log p\,(a_{rt})$$

When we utilize the Euclidean distance, the similarity metric of numerical attributes will become

$$s\left(\mathbf{x}_\mathbf{i}^\mathbf{u}, C_j\right) = \frac{\exp\left(-0.5|\mathbf{x}_\mathbf{i}^\mathbf{u} - \mathbf{c}_\mathbf{j}|^2\right)}{\sum_{t=1}^{k} \quad \exp(-0.5|\mathbf{x}_\mathbf{i}^\mathbf{u} - \mathbf{c}_\mathbf{t}|^2)}$$

This method transforms the distance of numerical attributes into the similarity. And it can be unified with the similarity of the categorical features.

3. Iterative clustering learning based on object-cluster similarity metric

By using the unified similarity metric above, we can determine the similarity between the x_i and each clusters C_j and the Q matrix. And we will use the OCIL algorithm to do the iterative clustering. This algorithm has the following five steps:

Algorithm 1:

1) Do the initialization and calculate the importance of each categorical attribute by the entropy, if applicable.
2) Set the cluster label of each data samples $\mathbf{x}_i$ to zero, random select the k initial objects, one for each cluster
3) Find the new cluster label of each data sample $\mathbf{x}_i$ by calculating the unified similarity.

$$y_{i(new)} = \arg \max_{1 \leq j \leq k} \left[s\left(\mathbf{x_i}, C_j\right) \right]$$

4) Update the new position of each clusters center of numerical attributes and frequency of categorical features.
5) Do 3) and 4) until the cluster labels are not changed again, output the clusters information.

4. Automatic selection of cluster number

Similar to the k-modes and k-prototype algorithm, the OCIL algorithm also suffers from a selection problem of cluster number. The cluster number which is preassigned should be equal to the true one, otherwise it will lead to an incorrect result and be overfitting. So, we enable a method that uses the competition and penalization mechanisms to eliminate the redundant clusters automatically.

4.1 competition mechanism

The competition mechanism is aimed to solve the dead-unit problem encountered by competitive learning. This mechanism is to avoid the cluster number k being less than the right value due to the **excessive penalization mechanism.** This algorithm introduces a new parameter gammar(j), the parameter describes the winning frequency of a cluster in its competition which is occurred in the algorithm1 step 3 :

$$\gamma_j = \frac{n_j}{\sum_{t=1}^{k} n_t}$$

And the cluster label $\mathbf{y}_i$ each data sample $\mathbf{x}_i$ will be updated like below:

$$y_i = v = \arg \min_{1 \leq j \leq k} \left[\boldsymbol{\gamma}_\mathbf{j} \left(1 - s\left(\mathbf{x_i}, C_j\right)\right)\right]$$

This parameter is the key to realizing the competition mechanism

4.2 penalization mechanism

Based on 4.1, we assign a weight which is utilized to measure the importance of each cluster to the whole cluster structure. In each competition, we rise the winning cluster's weight and reduce the weight of runner-up cluster. Subsequently, the weight of redundant clusters will be zero and we will fade out those clusters finally. Hence, we define a small learning rate **η** and we find the weight below:

$$\lambda_v^{(new)} = \lambda_v^{(old)} + \eta$$

$$\lambda_r^{(new)} = \max\left(0, \lambda_r^{(old)} - \eta s(\mathbf{x_i}, C_r)\right)$$

The v is the label of winner cluster and the r is the label of runner-up by contrast. Correspondingly, the iterative equation of cluster label should be updated as:

$$v = \arg\min_{1\le j\le k}\left[\gamma_j\left(1 - \lambda_j s(\mathbf{x_i}, C_j)\right)\right]$$

$$r = \arg\min_{j\neq v}\left[\gamma_j\left(1 - \lambda_j s(\mathbf{x_i}, C_j)\right)\right]$$

By using the competitive and penalization mechanism, we can eliminate the redundant clusters and avoid the overfitting.

5. Conclusion and interest in further research

The experiments on different benchmark data sets have shown the effectiveness and efficiency of the proposed approach above. Also, this creative paper has stimulated my interest in further research. Correspondingly, I put forward a few potential ideas which I want to explore in further research.

1) The weight of numerical vector is fixed in this experiment, so could we define a parameter to describe the difference between the weight of categorical and numerical attributes?
2) The algorithm in this paper used the Euclidean distance to calculate the similarity between the numerical vector, so could we try other distance calculation methods, such as Gower distance? And we can find out the different performance of each methods.
3) Could we also introduce a concept similar to entropy in numerical attributes, so we can reduce the impact of those attributes that change less. In addition, we can try to standardize the numerical attributes by z-score.