IOSR IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 2, Ver. II (Mar-Apr. 2014), PP 64-69
www.iosrjournals.org

Fuzzy C Means Clustering Algorithm for High Dimensional Data Using Feature Subset Selection Technique

N. Manjula[1], S. Pandiarajan[2] and J.Jagadeesan[3]
[1]*M.Tech Student, Department of Computer Science and Engineering, S.R.M. University, Chennai*
[2]*Asst.Professor, Department of Computer Science and Engineering, S.R.M. University, Chennai*
[3]*HOD, Department of Computer Science and Engineering, S.R.M. University, Chennai*

***Abstract**: Feature choice involves characteristic a set of the foremost helpful options that produces compatible results because the original entire set of options. A feature choice rule is also evaluated from each the potency and effectiveness points of read. Whereas the potency considerations the time needed to search out a set of options, the effectiveness is expounded to the standard of the set of options. Supported these criteria, an economical Fuzzy C Means (FCM) is projected and by experimentation evaluated in this paper. The quick rule works in 2 steps. Within the commencement, options area unit divided into clusters by exploitation graph-theoretic cluster ways. Within the second step, the foremost representative feature that's powerfully associated with target categories is chosen from every cluster to create a set of options. Options in numerous clusters area unit comparatively freelance; the clustering-based strategy of quick incorporates a high chance of manufacturing a set of helpful and independent options. To make sure the potency of quick, we have a tendency to adopt the economical Fuzzy C Means (FCM) cluster technique. The potency associated effectiveness of the quick rule area unit evaluated through an empirical study. in depth experiments area unit dole out to match quick and a number of other representative feature choice algorithms, namely, FCBF, ReliefF, CFS, Consist, and FOCUS-SF, with relevancy four kinds of well-known classifiers, namely, the chance primarily based Naive Thomas Bayes, the tree-based C4.5, the instance-based IB1,and also the rule-based liquidator before and once feature choice. The results, on thirty five in public on the market real-world high-dimensional image, microarray, and text knowledge, demonstrate that the quick not solely produces smaller subsets of options however additionally improves the performances of the four kinds of classifiers.*

***Index Terms**: feature subset selection, relevance, redundancy and high dimensionality.*

I. Introduction

With the aim of selecting a set of excellent options with relation to the target ideas, feature set selections an efficient method for reducing spatiality, removing impertinent information, increasing learning accuracy, and up result understandability. Several feature set choice ways are planned and studied for machine learning applications. They will be divided into four broad categories: the Embedded, Wrapper, Filter, and Hybrid approaches. The embedded ways incorporate feature choice as an {area district region locality a vicinity section} of the coaching method and are typically specific to given learning algorithms, and so is also additional economical than the opposite 3 classes. Ancient machine learning algorithms like call trees or artificial neural networks are samples of embedded approaches. The wrapper ways use the prophetic accuracy of a preset learning formula to see the goodness of the chosen subsets, the accuracy of the educational algorithms is sometimes high. However, the generality of the chosen options is restricted and therefore the process complexness is massive. The filter ways are freelance of learning algorithms, with sensible generality. Their process complexness is low, however the accuracy of the educational algorithms isn't secured. The hybrid ways are a mixture of filter and wrapper ways by employing a filter methodology to cut back search house that may be thought of by the following wrapper. The main target of combining filter and wrapper ways to attain the most effective attainable performance with a selected learning formula with similar time complexness of the filter ways. The wrapper ways are computationally dearly-won and have a tendency to over fit on tiny coaching sets. The filter ways, additionally to their generality, are typically a decent selection once the amount of options is incredibly massive. Thus, we are going to target the filter methodology during this paper. With relation to the filter feature choice ways, the applying of cluster analysis has been incontestable to be simpler than ancient feature choice algorithms. In cluster analysis, graph-theoretic ways are well studied and employed in several applications. Their results have, sometimes, the most effective agreement with human performance .The overall graph-theoretic bunch is simple: calculate a region graph of instances, and so delete any approach the graph that's a lot of longer/shorter (according to some criterion) than its neighbors. The result's a forest and every tree within the forest represents a cluster. We have a tendency to propose a quick bunch primarily based feature choice formula (FCM). The quick formula works in 2 steps. Within the initiative, options are divided into

clusters by mistreatment graph-theoretic bunch ways. Within the second step, the foremost representative feature that's powerfully associated with target categories is chosen from every cluster to make the ultimate set of options. Options in numerous clusters are comparatively freelance, the clustering based strategy of quick features a high chance of manufacturing a set of helpful and freelance options.

II. Feature Subset Selection Algorithm

Irrelevant options, alongside redundant options, severely affect the accuracy of the training machines. Thus feature set choice ought to be able to determine and remove the maximum amount of the orthogonal and redundant data as attainable. Keeping these in mind, I have a tendency to develop a FCM algorithm which can expeditiously and effectively subsume each irrelevant features don't contribute to recuperating deciphering ability to the target conception.

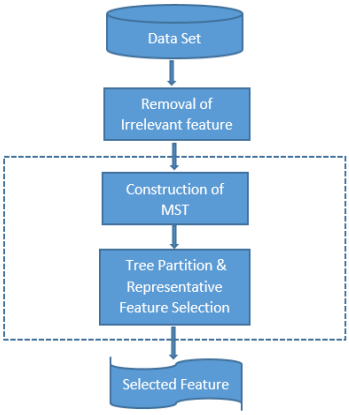


Fig.1. Framework of the feature subset selection algorithm.

We succeed this through a brand new feature choice framework (shown in Fig.1) that composed of the 2 connected components of impertinent feature removal and redundant feature elimination. The previous obtains options relevant to the target concept by eliminating impertinent ones, and therefore the latter removes redundant options from relevant ones via selecting representatives from completely different feature clusters, and thus produces the ultimate set. The impertinent feature removal is easy once the right connection lives is outlined or selected, while the redundant feature elimination may be a little bit of subtle. In our proposed quick rule, it involves 1) the development of the minimum spanning tree from a weighted complete graph; 2) the partitioning of the Minimum Spanning Time into a forest with every tree representing a cluster; and 3) the choice of representative options from the clusters.
Feature selection is the problem of choosing a small subset of features that ideally is necessary and sufficient to describe the target concept".

From the terms "necessary" and "sufficient" included in the given definition, it can be stated that feature

1. the classification accuracy do not significantly decrease; and
2. the resulting class distribution, given only the values for the selected features, is as close as possible to the original class distribution, given all features.

selection attempts to select the minimally sized subset of features according to the above criteria:

Ideally, feature selection methods search through the subsets of features, and try to find the best one among 2^N candidate subsets according to some evaluation function. However this procedure is exhaustive as it tries to find only the best one. It may be too costly and practically prohibitive even for a medium sized feature set. Other

methods based on heuristic or random search methods attempt to reduce computational complexity by compromising performance. These methods need a stopping criterion to prevent an exhaustive search of subsets. There are four basic steps in a typical feature selection method:

1. ***Starting point:*** Selecting a point in the feature subset space from which to begin the search can affect the direction of the search. One option is to begin with no features and successively add attributes. In this case, the search is said to proceed forward through the search space. Conversely, the search can begin with all features and successively remove them. In this case, the search proceeds backward through the search space. Another alternative is to begin somewhere in the middle and move outwards from this point.
2. ***Search organization:*** An exhaustive search of the feature subspace is prohibitive for all but a small initial number of features. With *N* initial features there exist 2^N possible subsets. Heuristic search strategies are more feasible than exhaustive ones and can give good results, although they do not guarantee finding the optimal subset.
3. ***Evaluation strategy:*** How feature subsets are evaluated is the single biggest differentiating factor among feature selection algorithms for machine learning. One paradigm, dubbed the *filter* [Koh95, Koh96], operates independent of any learning algorithm—undesirable features are filtered out of the data before learning begins. These algorithms use heuristics based on general characteristics of the data to evaluate the merit of feature subsets. Another school of thought argues that the bias of a particular induction algorithm should be taken into account when selecting features. This method, called the *wrapper* [Koh95, Koh96], uses an induction algorithm along with a statistical re-sampling technique such as cross-validation to estimate the final accuracy of feature subsets.
4. ***Stopping criterion:*** A feature selector must decide when to stop searching through the space of feature subsets. Depending on the evaluation strategy, a feature selector might stop adding or removing features when none of the alternatives improves upon the merit of a current feature subset. Alternatively, the algorithm might continue to revise the feature subset as long as the merit does not degrade. A further option could be to continue generating feature subsets until reaching the opposite end of the search space and then select the best.

Many learning algorithms can be viewed as making a (biased) estimate of the probability of the class label given a set of features. This is a complex, high dimensional distribution. Unfortunately, induction is often performed on limited data. This makes estimating the many probabilistic parameters difficult. In order to avoid over fitting the training data, many algorithms employ the Occam's Razor [Gam97] bias to build a simple model that still achieves some acceptable level of performance on the training data. This bias often leads an algorithm to prefer a small number of predictive attributes over a large number of features that, if used in the proper combination, are fully predictive of the class label. If there is too much irrelevant and redundant information present or the data is noisy and unreliable, then learning during the training phase is more difficult.

Feature subset selection is the process of identifying and removing as much irrelevant and redundant information as possible. This reduces the dimensionality of the data and may allow learning algorithms to operate faster and more effectively. In some cases, accuracy on future classification can be improved; in others, the result is a more compact, easily interpreted representation of the target concept.

III. Proposed System

Feature set choice may be viewed because the method of distinctive and removing as several inapplicable and redundant options as potential. this can be as a result of inapplicable options don't contribute to the prophetic accuracy and redundant options don't redound to obtaining a far better predictor for that they supply principally data that is already gift in alternative feature(s). Of the numerous feature set choice algorithms, some will effectively eliminate inapplicable options however fail to handle redundant options however a number of others will eliminate the inapplicable whereas taking care of the redundant options. Our projected FCM formula falls into the second cluster. Historically, feature set choice analysis has centered on checking out relevant options. A well-known example is Relief that weighs every feature in line with its ability to discriminate instances below totally different targets supported distance-based criteria operate. However, Relief is ineffective at removing redundant options as 2 prophetic however extremely correlative options area unit possible each to be extremely weighted. Relief-F extends Relief, enabling this technique to figure with howling and incomplete knowledge sets and to take care of multiclass issues, however still cannot establish redundant options.

IV. Algorithm Analysis

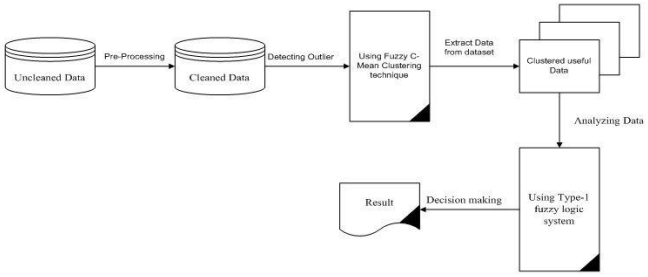
Fuzzy cluster plays a vital role in determination issues within the areas of pattern recognition and fuzzy model identification. A spread of fuzzy cluster strategies is projected and most of them as primarily based upon distance criteria [6]. One wide used rule is that the fuzzy c-means (FCM) rule. It uses reciprocal distance to cipher fuzzy weights. A lot of economical rule is that the new FCFM. It computes the cluster center exploitation Gaussian weights, uses giant initial prototypes, and adds processes of eliminating, cluster and merging. Within the following sections we have a tendency to discuss and compare the FCM rule and FCFM rule.

The pseudocode of our FCM algorithm follows.

```
// K is initial number of clusters, Imax is the iteration of fuzzy
    // c-means, p is for the weight
    Input initial number of clusters K, Imax, p

    ----------step 1: ----------
    //initialize weights of prototype
    for k = 0 to K-1
        for q = 0 to Q-1
                w[q,k] = random();
    ----------step 2: ----------
    //standardize the initial weight over K
    for q = 0 to Q-1
    sum = 0.0;
    for k = 0  to K-1
        sum = sum + w[q,k];
        for k = 0 to K-1
                w[q,k] = w[q,k] /sum;

    ****************************************
    // starting  fuzzy c-means loop
    I = 0

    ----------step 3: ----------
        // standardize cluster weights over Q
        for k = 0 to K-1
                min = 99999.0; max =0.0;
                for q = 0 to Q-1
                        if (w[q,k] > max)
                                max = w[q,k];
                        if (w[q,k] < min)
                                min = w[q,k];

                sum = 0.0
                for q = 0 to Q-1
                        sum = sum + (w[q,k] – min) /( max –min);

                for q = 0 to Q-1
                        w[q,k] = w[q,k]/sum;

        ----------step 4: ----------
        // compute new prototype center
        for k = 0 to K-1
                for n = 0 to N-1
                        sum = 0.0;
                        for q = 0 to Q-1
                                sum = sum + w[q,k] x[n,q];
                        z[n,k] = sum;
```

```
----------step 5: ----------
// compute new weight
for q = 0 to Q-1
        sum = 0.0
        for k = 0 to K-1
                D[q,k] =0.0;
                for n = 0 to N-1
                        D[q,k] = D[q,k] + (x[n,q] – z[n,k])^2
                sum = sum + (1/(1 + D[q,k]))^1/(p-1) ;
        for k = 0 to K-1
                W[q,k] = (1/(1 + D[q,k]))^1/(p-1) /sum;

----------step 6: ----------
I = I + 1

If I <Imax
    Goto step 3;
// end of fuzzy c-means loop
*********************************************

----------step 7: ----------
// assign feature vector according the max weight
for q = 0 to Q-1
    maxWeight = 0.0;
    for k = 0 to K-1
        ifmaxWeight< weight[q,k];
            maxWeight = weight[q,k];
            kmax = k;
    cluster[q] = k;

----------step 8: ----------

// eliminate clusters with no feature vectors
eliminate(0);   /* call the process of eliminating clusters contains less than or equal to the number passed to
it.Here we only pass 0 for this algorithm. */

----------step 9: ----------
// compute arithmetic center of clusters
// calculate sigma and Xie_Beni value
            .
for k = 1 to K do
    fuzzyweights();     /* Calculate fuzzy weight (Eqn. 4)
□2^k = variance();      /* Get variance (mean-square error) of
                                        each cluster (Eqn. 9) */
□ = Xie-Beni();     /* Compute modified XB (Eqn. 8) */
```

Fuzzy C Means Clustering Algorithm:



V. Conclusion

In this paper, we've got conferred a Fuzzy C Means suggests that clustering-based feature set choice rule for rime dimensional data. The rule involves 1) removing digressive features, 2) constructing a minimum spanning tree from relative ones, and 3) partitioning the Mountain Time and choosing representative options. Within the projected rule, a cluster consists of options. Every cluster is treated as one feature and so spatiality is drastically reduced. We have compared the performance of the projected algorithm with those of the 5 well-known feature selection algorithms FCBF, ReliefF, CFS, Consist, and FOCUS-SF on the thirty five in public out there image, microarray, and text information from the four totally different aspects of the proportion of chosen options, runtime, classification accuracy of a given classifier, and also the Win/Draw/Loss record. Generally, the planned formula obtained the most effective proportion of chosen options, the most effective runtime, and the best classification accuracy for Naive Bayes, C4.5, and RIPPER, and also the runner-up classification accuracy for IB1.The Win/Draw/Loss records confirmed the Conclusions.

We additionally found that FCM obtains the rank of one for microarray information, the rank of two for text information, and also the rank of three for image information in terms of classification accuracy of the four different types of classifiers, and CFS could be a sensible various. At identical time, FCBF could be a sensible various for image and text data. Moreover, Consist, and FOCUS-SF square measure alternatives for text information. For the long run work, we tend to commit to explore differing kinds of correlation measures, and study some formal properties of feature house.

References

[1] H. Almuallim and T.G. Dietterich (1992),Algorithms for IdentifyingRelevant Features.
[2] H. Almuallim and T.G. Dietterich (1994), Learning Boolean Concepts inthe Presence of Many Irrelevant Features.
[3] A. Arauzo-Azofra, J.M. Benitez, and J.L. Castro (2004),A Feature SetMeasure Based on Relief.
[4] L.D. Baker and A.K. McCallum (1998), Distributional Clustering ofWords for Text Classification.
[5] R. Battiti (1994),Using Mutual Information for Selecting Features inSupervised Neural Net Learning.
[6] D.A. Bell and H. Wang (2000),A Formalism for Relevance and ItsApplication in Feature Subset Selection.
[7] J. Biesiada and W. Duch (2008),Features Election for High-Dimensionaldata a Pearson Redundancy Based Filter.
[8] R. Butterworth, G. Piatetsky-Shapiro, and D.A. Simovici (2005),OnFeature Selection through Clustering.
[9] C. Cardie (1993),Using Decision Trees to Improve Case-Based Learning.
[10] P. Chanda, Y. Cho, A. Zhang, and M. Ramanathan (2009),Mining ofAttribute Interactions Using Information Theoretic Metrics.