# Entropy-Based Criterion in Categorical Clustering

**Tao Li** TAOLI@CS.ROCHESTER.EDU
Department of Computer Science, University of Rochester, Rochester, NY 14627-0226 USA
**Sheng Ma** SHENGMA@US.IBM.COM
IBM T. J. Watson Research Center, Hawthorne, NY 10532 USA
**Mitsunori Ogihara** OGIHARA@CS.ROCHESTER.EDU
Department of Computer Science, University of Rochester, Rochester, NY 14627-0226 USA

## Abstract

Entropy-type measures for the heterogeneity of clusters have been used for a long time. This paper studies the entropy-based criterion in clustering categorical data. It first shows that the entropy-based criterion can be derived in the formal framework of probabilistic clustering models and establishes the connection between the criterion and the approach based on dissimilarity coefficients. An iterative Monte-Carlo procedure is then presented to search for the partitions minimizing the criterion. Experiments are conducted to show the effectiveness of the proposed procedure.

## 1. Introduction

Clustering is the problem of partitioning a finite set of points in a multi-dimensional space into classes (called clusters) so that (i) the points belonging to the same class are *similar* and (ii) the points belonging to different classes are *dissimilar*. Clustering has been extensively studied in machine learning, databases, and statistics from various perspectives. Many applications of clustering have been discussed and many clustering techniques have been developed.

An important step in designing a clustering technique is defining a way to measure the quality of partitioning in terms of the above two objectives. For clustering numerical data, it is natural to think of designing such a measure based on a geometrical distance. Given such a measure, an appropriate partition can be computed by optimizing some quantity (e.g., the sum of the distances of the points to their cluster centroids). However, if the data vectors contain categorical variables, geometric approaches are inappropriate and other strategies must be developed (Bock, 1989).

Appearing in *Proceedings of the $21^{st}$ International Conference on Machine Learning*, Banff, Canada, 2004. 

The problem of clustering becomes more challenging when the data is categorical, that is, when there is no inherent distance measures between data values. This is often the case in many applications where data is described by a set of descriptive or binary attributes, many of which are not numerical. Examples of such include the country of origin and the color of eyes in demographic data.

Many algorithms have been developed for clustering categorical data, e.g., (Barbara et al., 2002; Gibson et al., 1998; Huang, 1998; Ganti et al., 1999; Guha et al., 2000; Gyllenberg et al., 1997). Entropy-type measures for similarity among objects have been used from early on. In this paper, we show that the entropy-based clustering criterion can be formally derived in the framework of probabilistic clustering models. We also establish the connections between the entropy-based criterion with the approach based on dissimilarity coefficients. We then develop an efficient Monte-Carlo procedure to find the optimal partition for minimizing the entropy-based criterion. Experiments demonstrate the efficacy and effectiveness of our approach.

The rest of the paper is organized as follows: Section 2 formulates the problem of categorical clustering and sets down notations, Section 3 introduces the traditional entropy-based clustering criterion, Section 4 shows the equivalence between the entropy-based criterion with the classification likelihood, Section 5 establishes the relations between entropy-based criterion with dissimilarity coefficients, Section 6 presents the iterative Monte-Carlo based procedure for minimizing the entropy criterion, Section 7 gives our experimental results, and finally Section 8 concludes.

## 2. Definitions and Notations

Let $D$ be a dataset of $n$ points $d_1, d_2, \ldots, d_n$, where for each $i$, $1 \leq i \leq n$, $d_i$ is a vector of $p$ categorical attributes. For each $i$, $1 \leq i \leq n$, and for each $j$, $1 \leq j \leq p$, let $d_{ij}$ be the $j$-th component of $d_i$. We want to find a partition of $D$ into classes $C_1, C_2, \ldots, C_K$ such that the points within each class are *similar* to each other.

Each categorical variable can be decomposed into a collection of indicator variables. Suppose that every categorical variable in $D$ has at most $m$ possible values. For each variable $v$, let the $m$ values naturally correspond to the numbers from 1 to $m$ and let $v^{(1)}, \cdots, v^{(m)}$ be the binary variables such that for each $k$, $1 \leq k \leq m$, $v^{(k)} = 1$ if and only if the $v$ takes the $k$-th value. Then the data set can be expressed as a collection of $m$ $n \times r$ matrices $(d_{ij}^k), 1 \leq i \leq n, 1 \leq j \leq r, 1 \leq k \leq m$, where $d_{ij}^k = 1$ if the $j$-th attribute of the $i$-th data point is in the $k$-th category. Hence the following discussion is based on binary variables.

We set down some notations. Suppose that a set of $n$ $r$-dimensional binary data vectors, $X$, represented as an $n \times r$ matrix, $(x_{ij})$, is partitioned into $K$ classes $C = (C_1, \ldots, C_K)$ and we want the points within each class are *similar* to each other. We view $C$ as a partition of the indices $\{1, \ldots, n\}$. So, for all $i$, $1 \leq i \leq n$, and $k$, $1 \leq k \leq K$, we write $i \in C_k$ to mean that the $i$-th vector belongs to the $k$-th class. Let $N = nr$. For each $k$, $1 \leq k \leq K$, let $n_k = \|C_k\|$, $N_k = n_k r$, and for each $j$, $1 \leq j \leq r$, let $N_{j,k,1} = \sum_{i \in C_k} x_{ij}$ and $N_{j,k,0} = n_k - N_{j,k,1}$. Also, for each $j$, $1 \leq j \leq r$, let $N_{j,1} = \sum_{i=1}^{n} x_{ij}$ and $N_{j,0} = n - N_{j,1}$.

Consider a discrete random vector $Y = (y_1, y_2, \cdots, y_r)$ with $r$ independent components, where for each $i$, $1 \leq i \leq r$, $y_i$ takes a value from a finite set $V_i$.

$$\begin{aligned} H(Y) &= -\sum p(Y) \log p(Y) = \sum_{i=1}^{r} H(y_i) \\ &= -\sum_{i=1}^{r} \sum_{t \in V_i} p(y_i = t) \log p(y_i = t) \end{aligned}$$

We will use $\hat{H}$ for the estimated entropy of the partition.

## 3. Classical Entropy Criterion

### 3.1. Entropy Criterion

The classical clustering criterion (Bock, 1989; Celeux & Govaert, 1991) searches for a partition $C$ that maximizes the following quantity $O(C)$:

$$\begin{aligned} O(C) &= \sum_{k=1}^{K} \sum_{j=1}^{r} \sum_{t=0}^{1} \frac{N_{j,k,t}}{N} \log \frac{N N_{j,k,t}}{N_k N_{j,t}} \\ &= \sum_{k=1}^{K} \sum_{j=1}^{r} \sum_{t=0}^{1} \frac{N_{j,k,t}}{N} \left( \log \frac{N_{j,k,t}}{n_k} - \log \frac{N_{j,t}}{n} \right) \\ &= \frac{1}{N} \sum_{k=1}^{K} \sum_{j=1}^{r} \sum_{t=0}^{1} n_k \frac{N_{j,k,t}}{n_k} \log \frac{N_{j,k,t}}{n_k} \\ &\quad - \sum_{j=1}^{r} \sum_{t=0}^{1} \frac{N_{j,t}}{N} \log \frac{N_{j,t}}{n} \\ &= \frac{1}{r} \left( \hat{H}(X) - \frac{1}{n} \sum_{k=1}^{K} n_k \hat{H}(C_k) \right). \end{aligned} \quad (1)$$

Observe that $\frac{1}{n} \sum_{k=1}^{K} n_k \hat{H}(C_k)$ is the entropy measure of the partition, i.e., the weighted sum of each cluster's entropy. This leads to the following criterion: Given a dataset, fix $\hat{H}(X)$, then maximizing $O(C)$ is equivalent to minimizing the expected entropy of the partition:

$$\frac{1}{n} \sum_{k=1}^{K} n_k \hat{H}(C_k) \quad (2)$$

### 3.2. Kullback–Leibler Measure

The above criterion can be interpreted as the Kullback–Leibler measure (K–L measure) as follows: Suppose the observed dataset is generated by a number of classes. We first model the unconditional probability density function and then seek a number of partitions whose combination yields the density function (Roberts et al., 2000). The K–L measure then tries to measure the difference between the unconditional density and the density under partition.

Let $p(y)$ and $q(y)$ be two distributions. Then

$$KL(p(y) \parallel q(y)) = \int p(y) \log \left( \frac{p(y)}{q(y)} \right) dy.$$

For each $j$, $1 \leq j \leq r$, and for each $t \in \{0, 1\}$, let $p(y_j = t) = \frac{N_{j,t}}{n}$ and $q(y_j = t) = p_k(y_j = t) = \frac{N_{j,k,t}}{n_k}$. Then $KL(p(y) \parallel q(y)) \approx \sum p(y) \log(p(y)) - \sum p(y) \log(q(y))$. The latter quantity is equal to $\sum_{j=1}^{r} \sum_{t \in \{0,1\}} p(y_j = t) \log(p(y_j = t)) - \sum_{j=1}^{r} \sum_{t \in \{0,1\}} p(y_j = t) \log(q(y_j = t))$, and this is equal to $-\hat{H}(X) - \sum_{j=1}^{r} \sum_{t \in \{0,1\}} \frac{N_{j,t}}{n} \log(\frac{N_{j,k,t}}{n_k}) = -\hat{H}(X) + \frac{n_k}{n} \hat{H}(C_k)$. Thus, $O(C)$ is equal to

$$\frac{1}{r} \left( \hat{H}(X) - K\hat{H}(X) + \sum_{k=1}^{K} KL(p(y) \parallel p_k(y)) \right). \quad (3)$$

So, minimizing the K–L measure is equivalent to minimizing the expected entropy of partition over the observed data.

## 4. Entropy and Mixture Models

In this section, we show that the entropy-based clustering criterion can be formally derived using a likelihood principle based on Bernoulli mixture models. In mixture models, the observed data are thought of as coming from a number

of different latent classes. In our case, the observed data, $\mathcal{X} = \{x_i\}_{i=1}^n$, are $r$-dimensional $0/1$-vectors. So, we can assume that the data are samples from $\{0,1\}^r$ and are subject to a mixture of multivariate Bernoulli distributions:

$$\begin{aligned} p(x_i) &= \sum_{k=1}^{K} \pi_k p(x_t|k) \\ &= \sum_{k=1}^{K} \pi_k \prod_{j=1}^{r} \left(a_k^{(j)}\right)^{x_{i,j}} \left(1-a_k^{(j)}\right)^{(1-x_{i,j})}. \end{aligned}$$

Here for each $i$, $1 \le i \le n$, $x_i = (x_{i,1}, x_{i,2}, \cdots, x_{i,r})$ and for each $k$, $1 \le k \le K$, $\pi_k$ is the probability that the $k$-th latent class is selected (so $\sum_{k=1}^{K} \pi_k = 1$). Also, for each $j$, $1 \le j \le r$, and for each $k$, $1 \le k \le K$, $a_k^{(j)}$ is the probability that the $j$-th attribute is exhibited in the $k$-th latent class. For each $k$, $1 \le k \le K$, let $\mathbf{a_k} = (a_k^{(1)}, \ldots, a_k^{(r)})$. Let $\mathbf{a} = \{\mathbf{a_k}\}_{1 \le k \le K}$. We use $p(x_i|\mathbf{a})$ for the above $p(x_i)$ to signify that it is under the latent variables $\mathbf{a}$.

### 4.1. Maximum Likelihood and Classification Likelihood

Recall that the Maximum Likelihood Principle states that the best model is the one that has the highest likelihood of generating the observed data. In the mixture model approach, since the data points are independent and identically distributed, the maximum likelihood of obtaining the entire sample $\mathcal{X}$ can be expressed as:

$$\begin{aligned} L(\mathbf{a}) &= \log p(\mathcal{X}|\mathbf{a}) = \log \prod_{i=1}^{n} p(x_t|\mathbf{a}) \\ &= \sum_{i=1}^{n} \log \left( \sum_{k=1}^{K} \pi_k \prod_{j=1}^{r} \left(a_k^{(j)}\right)^{x_{i,j}} \left(1-a_k^{(j)}\right)^{(1-x_{tj})} \right). \end{aligned}$$

We introduce auxiliary vectors, $u_i = (u_{i,k})$, $1 \le i \le n$, $1 \le k \le K$, where $u_{i,k} = 1$ if and only if $x_i$ comes from the cluster $C_k$. These vectors are additional unknown parameters. The classification likelihood (Symons, 1981), denoted by $CL(\mathbf{a}, \mathbf{u})$, is equal to:

$$\begin{aligned} &\sum_{i=1}^{n} \sum_{k=1}^{K} u_{i,k} \log p(x_i|\mathbf{a_k}) \\ &= \sum_{i=1}^{n} \sum_{k=1}^{K} u_{i,k} \log \prod_{j=1}^{r} \left(a_k^{(j)}\right)^{x_{i,j}} \left(1-a_k^{(j)}\right)^{(1-x_{i,j})} \end{aligned} \quad (4)$$

It is easy to see that

$$CL(\mathbf{a}, \mathbf{u}) = L(\mathbf{a}) - LP(\mathbf{a}, \mathbf{u}),$$

where

$$LP(\mathbf{a}, \mathbf{u}) = -\sum_{i=1}^{n} \sum_{k=1}^{K} u_{i,k} \log \left( \frac{\pi_k p(x_i|\mathbf{a_k})}{\sum_{\ell=1}^{K} \pi_\ell p(x_i|\mathbf{a}_\ell)} \right).$$

Note that $LP(\mathbf{a}, \mathbf{u}) \ge 0$ and the quantity can be thought of as corresponding to the logarithm of the probability of the partition induced by $\mathbf{u}$. Hence, the classification likelihood is the standard maximum likelihood penalized by a term measuring the quality of the partition.

### 4.2. Maximizing the Likelihood

From Equation 4, $CL(\mathbf{a}, \mathbf{u})$ is equal to:

$$\begin{aligned} &\sum_{k=1}^{K} \log \prod_{i \in C_k} \prod_{j=1}^{r} \left(a_k^{(j)}\right)^{x_{i,j}} \left(1-a_k^{(j)}\right)^{(1-x_{i,j})} \\ &= \sum_{k=1}^{K} \sum_{j=1}^{r} \left( N_{j,k,1} \log a_k^{(j)} + N_{j,k,0} \log(1-a_k^{(j)}) \right). \end{aligned}$$

If $\mathbf{u}$ is fixed maximization of $CL(\mathbf{a}, \mathbf{u})$ over $\mathbf{a}$ reduces to simultaneous maximization of $CL_{k,j}(a_k^{(j)})$ for all $k$, $1 \le k \le K$, and $j$, $1 \le j \le r$, where $CL_{k,j}(a_k^{(j)}) = N_{j,k,1} \log a_k^{(j)} + N_{j,k,0} \log(1-a_k^{(j)})$. For all $k$, $1 \le k \le K$, and $j$, $1 \le j \le r$, $0 < a_k^{(j)} < 1$ and $N_{j,k,0} + N_{j,k,1} = n_k$. So, $\frac{\partial CL_{k,j}}{\partial a_k^{(j)}} = 0 \iff \frac{N_{j,k,1}}{a_k^{(j)}} - \frac{N_{j,k,0}}{1-a_k^{(j)}} = 0 \iff (N_{j,k,1} + N_{j,k,0}) a_k^{(j)} = N_{j,k,1} \iff a_k^{(j)} = \frac{N_{j,k,1}}{n_k}$. By replacing $a_k^{(j)}$ by $\frac{N_{j,k,1}}{n_k}$ for all $k$, $1 \le k \le K$ and $j$, $1 \le j \le r$, we have

$$CL(\mathbf{a}, \mathbf{u}) = -\sum_{k=1}^{K} n_k \hat{H}(C_k). \quad (5)$$

Once a dataset is given, the quantities $n$, $p$, and $\hat{H}(X)$ are fixed. So, the criterion $CL(\mathbf{a}, \mathbf{u})$ is equivalent to $O(C)$ in Equation 1, since both aim at minimizing the expected entropy over the partition. Note that $a_k$ can be viewed as the "center" for the cluster $C_k$. The equivalence between the information theoretical criterion and the maximum likelihood criterion suggests a way to assess the number of clusters when using the entropy criterion: to look at the likelihood ratio based on latent classes. In addition, each cluster $C_k$ is characterized by the "center" $\mathbf{a_k}$.

## 5. Entropy and Dissimilarity Coefficients

In this section, we show the relations between the entropy criterion and the dissimilarity coefficients. A popular partition-based criterion (within-cluster) for clustering is to minimize the summation of dissimilarities inside the cluster. The within-cluster criterion can be described as minimizing

$$D(C) = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,i' \in C_k} \delta(x_i, x_{i'}), \quad (6)$$

where $\delta(x_i, x_{i'})$ is the distance measure between $x_i$ and $x_{i'}$. In general, the distance function can be defined using $L_p$ norm for some integer $p > 0$. For binary clustering, however, the dissimilarity coefficients are popular measures of the distances.

### 5.1. Dissimilarity Coefficients

Given two data points, $w$ and $w'$, there are four fundamental quantities that can be used to define similarity between the two (Baulieu, 1997): $a = \|\{j \mid w_j = w'_j = 1\}\|$, $b = \|\{j \mid w_j = 1 \wedge w'_j = 0\}\|$, $c = \|\{j \mid w_j = 0 \wedge w'_j = 1\}\|$, and $d = \|\{j \mid w_j = w'_j = 0\}\|$, where $1 \leq j \leq r$. It has been shown in (Baulieu, 1997) that the presence/absence based dissimilarity measure can be generally [1] written as $D(a, b, c, d) = \frac{b+c}{\alpha a+b+c+\beta d}$, where $\alpha > 0$ and $\beta \geq 0$. Dissimilarity measures can be transformed into a similarity function by simple transformations such as adding 1 and inverting, dividing by 2 and subtracting from 1, etc. (Jardine & Sibson, 1971). If the joint absence of the attribute is ignored, i.e., $\beta$ is set to 0, then the binary dissimilarity measure can be generally written as $D(a, b, c, d) = \frac{b+c}{\alpha a+b+c}$, where $\alpha > 0$. Table 1 shows several common dissimilarity coefficients and the corresponding similarity coefficients.

| Name | Similarity | Dissimilarity | Metric |
|---|---|---|---|
| Simple Matching Coeff. | $\frac{a+d}{a+b+c+d}$ | $\frac{b+c}{a+b+c+d}$ | Y |
| Jaccard's Coeff. | $\frac{a}{a+b+c}$ | $\frac{b+c}{a+b+c}$ | Y |
| Dice's Coeff. | $\frac{2a}{2a+b+c}$ | $\frac{b+c}{2a+b+c}$ | N |
| Russel&Rao's Coeff. | $\frac{a}{a+b+c+d}$ | $\frac{b+c+d}{a+b+c+d}$ | Y |
| Rogers&Tanimoto's Coeff. | $\frac{\frac{1}{2}(a+d)}{\frac{1}{2}(a+d)+b+c}$ | $\frac{b+c}{\frac{1}{2}(a+d)+b+c}$ | Y |
| Sokal&Sneath's Coeff. I | $\frac{\frac{1}{2}a}{\frac{1}{2}a+b+c}$ | $\frac{b+c}{\frac{1}{2}a+b+c}$ | Y |
| Sokal&Sneath's Coeff. II | $\frac{2(a+d)}{2(a+d)+b+c}$ | $\frac{b+c}{2(a+d)+b+c}$ | N |

*Table 1.* Binary dissimilarity and similarity coefficients. The "Metric" column indicates whether the given dissimilarity coefficient is metric or not. A 'Y' stands for 'YES' while an 'N' stands for 'No'.

### 5.2. Global Equivalence on Coefficients

In cluster applications, the rankings based on a dissimilarity coefficient is often of more interest than the actual value of the dissimilarity coefficient. The following results from (Baulieu, 1997) establish the equivalence among dissimilarity coefficients.

**Definition 1** *Two dissimilarity coefficients, $D$ and $D'$, are said to be* globally order equivalent *if for all $(a_1, b_1, c_1, d_1), (a_2, b_2, c_2, d_2) \in (Z^+)^4$, it holds that $D(a_2, b_2, c_2, d_2) < D(a_1, b_1, c_1, d_1) \iff D'(a_2, b_2, c_2, d_2) < D'(a_1, b_1, c_1, d_1)$.*

**Proposition 1** *Let $D = \frac{b+c}{\alpha a+b+c+\beta d}$ and $D' = \frac{b+c}{\alpha' a+b+c+\beta' d}$ such that $\alpha\beta' = \alpha'\beta$. Then, $D$ and $D'$ are globally order equivalent.*

**Corollary 1** *For all $\alpha, \alpha' > 0$, $D = \frac{b+c}{\alpha a+b+c}$ and $D' = \frac{b+c}{\alpha' a+b+c}$ are globally order equivalent.*

In other words, if the paired absences are to be ignored in the calculation of dissimilarity values, then there is only one single dissimilarity coefficient modulo the global order equivalence: $\frac{b+c}{a+b+c}$. With the equivalence results, our following discussion is then based on the single dissimilarity coefficient.

### 5.3. Entropy and Dissimilarity Coefficients

In the coefficient $\frac{b+c}{a+b+c}$, $b+c$ is the number of mismatches between binary vectors. If we assume that there is no joint absence in the dataset, i.e., for every single pairwise comparison $d = 0$, then $a + b + c$ is constant because the vectors are drawn from the same set and they all have the same number of attributes.

It is easy to see that the following property holds:

**Remark 1** *Let $u$, $v$, and $w$ be binary vectors. If $\delta(u, v) < \delta(u, w)$, then $H(\{u, v\}) < H(\{u, w\})$.*

Now examine the within-cluster criterion in Equation 6. We have:

$$\begin{aligned} D(C) &= \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,i' \in C_k} \delta(x_i, x_{i'}) \\ &= \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,i' \in C_k} \frac{1}{r} \sum_{j=1}^{r} |x_{i,j} - x_{i',j}| \\ &= \frac{1}{r} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{j=1}^{r} n_k \rho_k^{(j)} n_k (1 - \rho_k^{(j)}) \\ &= \frac{1}{r} \sum_{k=1}^{K} \sum_{j=1}^{r} n_k \rho_k^{(j)} (1 - \rho_k^{(j)}). \end{aligned}$$

Here for each $k$, $1 \leq k \leq K$, and for each $j$, $1 \leq j \leq r$, $\rho_k^{(j)}$ is the probability that the $j$-th attribute is 1 in $C_k$.

Havrda and Charvat (Havrda & Charvat, 1967) proposed a generalized entropy of degree $s$, $s > 0$ and $s \neq 1$, for a discrete probability distribution $Q = (q_1, q_2, \ldots, q_n)$:

$$H^s(Q) = (2^{(1-s)} - 1)^{-1} \left( \sum_{i=1}^{n} q_i^s - 1 \right).$$

It holds that

[1] Basically, the presence/absence based dissimilarity measure satisfies a set of axioms such as non-negative, range in $[0, 1]$, rationality whose numerator and denominator are linear and symmetric, etc. (Baulieu, 1997).

$\lim_{s\to 1} H^s(Q) = -\sum_{i=1}^n q_i \log q_i$ and
$H^2(Q) = -2\left(\sum_{i=1}^n q_i^2 - 1\right)$.

If we use the entropy with $s = 2$, then

$$\begin{aligned}
&\frac{1}{n}\sum_{k=1}^{K} n_k \hat{H}(C_k) \\
&= -\frac{1}{2n}\sum_{k=1}^{K}\sum_{j=1}^{r} n_k \left((\rho_k^{(j)})^2 + (1-\rho_k^{(j)})^2 - 1\right) \\
&= \frac{1}{n}\sum_{k=1}^{K}\sum_{j=1}^{r} n_k \rho_k^{(j)}(1-\rho_k^{(j)}) \\
&= \frac{r}{n} D(C).
\end{aligned}$$

Thus, we have established the connections between the entropy-criterion and the dissimilarity coefficients. Figure 1 shows the relations between the entropy-based criterion and other criteria. The relations of the entropy-based criterion to Minimum Description Length (MDL) / Minimum Message Length (MML) and to rate distortion theory can be found in (Cover & Thomas, 1991; Baxter & Oliver, 1994).

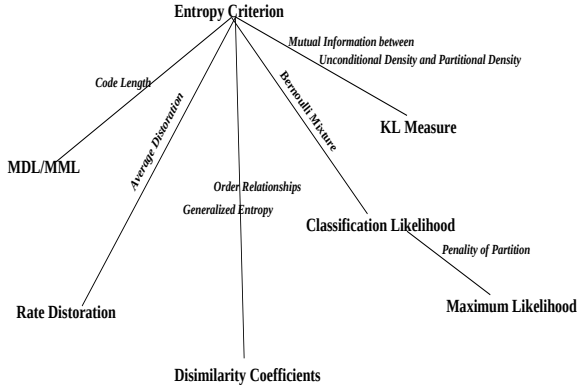


*Figure 1.* A summary of relations among various clustering criteria.

We note here that Wallace (Wallace, 1989) proposed a two-step procedure for numerical hierarchical cluster analysis by minimizing *Gaussian entropy*, defined based on the logarithm of the covariance matrix determinant. The relationships between minimization of Gaussian entropy and other objective functions such as minimum variance, maximum likelihood and information radius were also discussed in (Wallace, 1989).

## 6. Entropy-based Clustering

In the previous sections, we showed the connections between the entropy-based criterion and other criteria. The relations seem to indicate that many clustering problems can be reduced to the problem of minimizing the entropy criterion. In this section, we develop an efficient procedure to find the optimal partition for minimizing the entropy-based criterion.

### 6.1. The Entropy-Based Criterion

The entropy-based criterion (eq. 2) can be written as

$$\begin{aligned}
H(C) &= \frac{1}{n}\sum_{k=1}^{K} n_k \hat{H}(C_k) \\
&= -\frac{1}{n}\sum_{k=1}^{K}\sum_{j=1}^{r}\sum_{t=0}^{1} n_k \frac{N_{j,k,t}}{n_k}\log\frac{N_{j,k,t}}{n_k}.
\end{aligned}$$

**Proposition 2** $\hat{H}(X) \geq H(C) = \frac{1}{n}\sum_{k=1}^{K} n_k \hat{H}(C_k)$

**Proof** We have

$$\begin{aligned}
&\sum_{k=1}^{K}\sum_{j=1}^{r}\sum_{t=0}^{1} N_{j,k,t}\log\frac{N_{j,k,t}}{n_k} \\
&= \sum_{j=1}^{r}\sum_{t=0}^{1}\sum_{k=1}^{K} N_{j,k,t}\log\frac{N_{j,k,t}}{n_k} \\
&\geq \sum_{j=1}^{r}\sum_{t=0}^{1}\left(\sum_{k=1}^{K} N_{j,k,t}\right)\log\frac{\sum_{k=1}^{K} N_{j,k,t}}{\sum_{k=1}^{K} n_k} \\
&= \sum_{j=1}^{r}\sum_{t=0}^{1} N_{j,t}\log\frac{N_{j,t}}{n}.
\end{aligned}$$

The inequality follows from the *log sum inequality* in (Cover & Thomas, 1991). Note that

$$\begin{aligned}
\hat{H}(X) &= -\sum_{j=1}^{r}\sum_{t=0}^{1}\frac{N_{j,t}}{n}\log\frac{N_{j,t}}{n} \quad \text{and} \\
H(C) &= -\frac{1}{n}\sum_{k=1}^{K}\sum_{j=1}^{r}\sum_{t=0}^{1} N_{j,k,t}\log\frac{N_{j,k,t}}{n_k}.
\end{aligned}$$

Hence, we have $\hat{H}(X) \geq H(C)$. ∎

Proposition 2 shows that any clustering process decreases the entropy. The goal of clustering is to find a partition $C$ such that the increase in the entropy, i.e., $\hat{H}(X) - H(C)$, is maximized (as in eq. 1). In other words, we should minimize $H(C)$.

**Proposition 3** *$H(C)$ is maximized when all data points are in the same cluster.*

**Proof** If all the data points are put in one cluster, then $H(C) = \hat{H}(X)$. Then, by Proposition 2, $H(C)$ is maximized. ∎

For each $j$, $1 \le j \le r$, and each $k$, $1 \le k \le K$, let $\theta(j|k)$ be the probability that the $j$-th component of a vector is 1 given that it belongs to $C_k$. Then $H(C)$ is equal to:

$$\begin{aligned} -\frac{1}{n}\sum_{k=1}^{K}\sum_{j=1}^{r} & n_k(\theta(j|k)\log\theta(j|k) \\ & + (1-\theta(j|k))\log(1-\theta(j|k)) \\ = & -\sum_{k=1}^{K}\sum_{j=1}^{r}\theta(j,k)\log\theta(j|k) \\ & + (\theta(k)-\theta(j,k))\log(1-\theta(j|k)) \\ = & -\sum_{k=1}^{K}\sum_{j=1}^{r}\sum_{i\in C_k}\theta(j,x_i)\log\theta(j|k) \\ & + \left(\frac{1}{n}-\theta(j,x_i)\right)\log(1-\theta(j|k)), \end{aligned}$$

where $\theta(k) = \frac{n_k}{n}$, $\theta(j,k) = \theta(k)\theta(j|k)$, $\theta(j,k) = \sum_{i\in C_k} p(j,x_i)$. Generally $p(j,x_i) = \frac{N_{j,1}}{n^2}$. The above shows that $H(C)$ is convex when varying the clustering since $p(j,x_i)$ is an invariant in the process, so $H(C)$ evolves like a negative logarithm which is a convex function. The convexity of $H(C)$ allows the optimization procedures to reach global minimum.

### 6.2. Optimization Procedure

We use a Monte-Carlo method to perform the optimization. Initially, all the points are placed in the same cluster. By Proposition 3, this initialization attains the maximal criterion. We then perform an iterative Monte-Carlo process to find the optimal partition. The clustering procedure is the following Algorithm 1.

Algorithm 1 uses a Monte-Carlo method to perform optimization (Rubinstein, 1981). Randomly picking a data point $x$ and putting it into another cluster is a trial step of modifying the parameters $\theta(j|k)$. We then check whether the entropy criterion is decreased, and if so, we accept the update and continue; otherwise, no modification will be made. This is repeated until there are no changes in the cluster assignment. The convergence property of the Monte-Carlo optimization is shown in (Rubinstein, 1981).

## 7. Experiments

### 7.1. Performance Measures

There are many ways to measure how clustering algorithms perform. One is the *confusion matrix*. Entry $(o,i)$ of the confusion matrix is the number of data points assigned to output class $o$ and generated from input class $i$. The input map $I$ is the map of the data points to the input classes. So, the information of the input map can be measured by the entropy $H(I)$. The goal of clustering is to find an output map $O$ that recovers the information. Thus, the conditional entropy $H(I|O)$ is interpreted as the information of the input map given the output map $O$, i.e., the proportion of information not recovered by the clustering algorithm. Therefore, the *recovery rate* of a clustering algorithm, defined as $1 - H(I|O)/H(I) = MI(I,O)/H(I)$[2], can also be used as a performance measure for clustering. The *purity* (Zhao & Karypis, 2001), which measures the extent to which each cluster contains data points primarily from a single class, is also a good measure. The purity of a clustering solution is obtained as a weighted sum of the purity of individual clusters, given by

$$Purity = \sum_{k=1}^{K}\frac{n_k}{n}P(C_k),$$

**Algorithm 1 clustering procedure**

Input: (data points: $X$, # of classes: $k$)
Output: cluster assignment;
**begin**
1. **Initialization:**
1.1 Put all data points into one cluster
1.2 Compute Initial Criterion $H_0$
2. **Iteration:**
**Repeat until no more changes in cluster assignment**
2.1 Randomly pick a point $x$ from a cluster $A$
2.2 Randomly pick another cluster $B$
2.3 Put $x$ into $B$
2.4 Compute the new entropy $H$
2.5 if $H \ge H_0$
2.5.1 Put $x$ back into $A$
2.5.2 $H = H_0$
2.6 end
2.7 $H_0 = H$
2.8 Goto Step 2.1
**end**
3. **Return** the cluster assignment
**end**

| Output \ Input | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| A | 9 | 0 | 1 | 0 | 0 | 0 | 0 |
| B | 0 | 20 | 0 | 0 | 0 | 0 | 0 |
| C | 32 | 0 | 0 | 0 | 1 | 2 | 2 |
| D | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| E | 0 | 0 | 4 | 0 | 2 | 0 | 8 |
| F | 0 | 0 | 0 | 13 | 0 | 0 | 0 |
| G | 0 | 0 | 0 | 0 | 0 | 6 | 0 |

*Table 2.* Confusion matrix of the zoo data.

[2] $MI(I,O)$ is the mutual information between $I$ and $O$.

where $P(C_k) = \frac{1}{n_k} max_j(n_k^{(j)})$, $n_k^j$ is the number of points of the $i$-th input class that were assigned to the $j$-th cluster[3]. A high purity value implies that the clusters are "pure" subsets of the input classes. In general, the larger the values of purity, the better the clustering solution is.

### 7.2. Zoo Dataset

We evaluate the performance of the algorithm on the zoo database available at the UC Irvine Machine Learning Repository. The database contains 100 animals, each of which has fifteen boolean attributes and one categorical attribute[4]. We translate the numeric attribute, "legs," into six features, which correspond to 0, 2, 4, 5, 6, and 8 legs, respectively. Table 2 shows the confusion matrix of this experiment and Table 3 shows comparison against K-means.

| | Entropy-based | K-means |
|---|---|---|
| Recovery Rate | 0.8001 | 0.7374 |
| purity | 0.9000 | 0.8400 |

*Table 3.* Results Comparison on Zoo dataset.

At each step of the iteration process, one data point experiments a new cluster. A Monte-Carlo method accepts any better solutions instead of performing systematic searches. Figure 2 shows the entropy descent of our Monte-Carlo method.

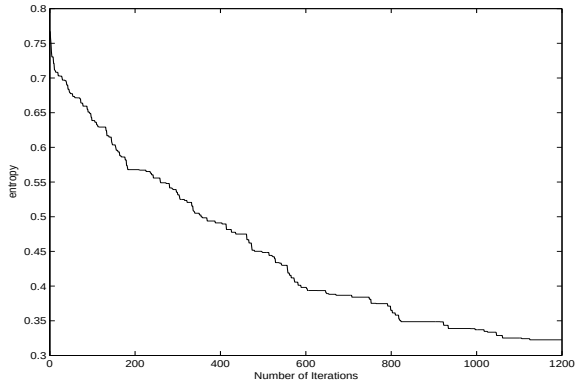


*Figure 2.* Entropy Descent of the Monte-Carlo Method.

### 7.3. Clustering Document Datasets

We also evaluate our entropy-based clustering algorithm on document datasets. In our experiments, documents are represented using binary vector-space model where each document is a binary vector in the term space and each element of the vector indicates the presence of the corresponding term. Our main goal is to find out how well this method would perform on document corpus. Therefore, we evaluate the method on standard labeled corpus widely used in information retrieval literature to evaluate supervised text categorization algorithms. In this way, we view the labels of the dataset as the objective knowledge on the structure of the datasets and then use the *purity* as the performance measure.

#### 7.3.1. Document Datasets

For our experiments we use a variety of datasets, most of which are frequently used in the information retrieval research. The range of the number of classes is from four to ten, and the range of the number of documents is from 476 to 8280, which seem varied enough to obtain good insights on the algorithm. Table 4 summarizes the characteristics of the datasets. **CSTR:** This is the dataset of the abstracts of technical reports published in the Department of Computer Science at the University of Rochester between 1991 and 2002. The TRs are available at *http://www.cs.rochester.edu/trs*. It has been used in (Li et al., 2003) for text categorization. The dataset contained 476 abstracts, which were divided into four research areas: Natural Language Processing(NLP), Robotics/Vision, Systems, and Theory. **WebKB:** The WebKB dataset contains webpages gathered from university computer science departments. There are about 8280 documents and they are divided into seven categories: student, faculty, staff, course, project, department and other. The raw text is about 27MB. Among these seven categories, student, faculty, course and project are four most populous entity-representing categories. The associated subset is typically called **WebKB4**. In this paper, we perform experiments on both seven-category and four-category datasets. **Reuters:** The Reuters-21578 Text Categorization collection contains documents collected from the Reuters newswire in 1987. It is a standard text categorization benchmark and contains 135 categories. In our experiments, we use a subset of the data collection which include the ten most frequent categories.

#### 7.3.2. Results on Document Datasets

To pre-process the datasets, we remove the stop words use a standard stop list and perform stemming using a porter stemmer, all HTML tags are skipped and all header fields except subject and organization of the posted article are ignored. In all our experiments, we first select the top 200 words by mutual information with class labels. The feature selection is done with the rainbow package (McCallum, 1996). In our experiments, we compare the performance of our entropy-based method with the popular vector space variant of the partitioning algorithms provided in the CLUTO package (Zhao & Karypis, 2001). CLUTO package is built on a sophisticated multi-level graph partitioning engine and offers many different criteria that be used to drive both partitional and agglomerative cluster-

[3] $P(C_k)$ is also called the individual cluster purity.

[4] The original data set has 101 data points but one animal, "frog," appears twice. So we eliminated one of them. We also eliminated two attributes, "animal name" and "type."

ing algorithms. Figure 3 shows the comparison and the entropy-based method is effective on all four datasets.

| Datasets | # documents | # class |
|---|---|---|
| CSTR | 476 | 4 |
| WebKB4 | 4199 | 4 |
| WebKB | 8,280 | 7 |
| Reuters | 2,900 | 10 |

*Table 4.* Document DataSets Descriptions.

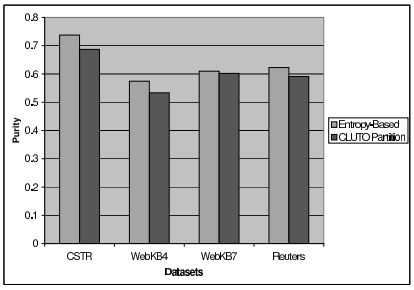


*Figure 3.* Performance Comparison on Document Datasets.

## 8. Conclusions

In this paper, we study the entropy-based criterion for categorical data clustering and illustrate its relations with other criteria. An efficient, iterative Monte-Carlo procedure for optimization that takes advantage of the convexity of the criterion is presented. The experimental results indicate the effectiveness of the proposed method.

## Acknowledgments

We would like to thank the reviewers for their valuable comments and suggestions. The first and the third authors are supported in part by NSF grants EIA-0080124 and EIA-0205061 and in part by NIH grants P30-AG18254.

## References

Barbara, D., Couto, J., & Li, Y. (2002). COOLCAT: an entropy-based algorithm for categorical clustering. *Proceedings of the Eleventh ACM CIKM Conference* (pp. 582–589).

Baulieu, F. B. (1997). Two variant axiom systems for presence/absence based dissimilarity coefficients. *Journal of Classification*, *14*, 159–170.

Baxter, R. A., & Oliver, J. J. (1994). *MDL and MML: similarities and differences* (Technical Report 207). Monash University.

Bock, H.-H. (1989). Probabilistic aspects in cluster analysis. In O. Opitz (Ed.), *Conceptual and numerical analysis of data*, 12–44. Berlin: Springer-verlag.

Celeux, G., & Govaert, G. (1991). Clustering criteria for discrete data and latent class models. *Journal of Classification*, *8*, 157–176.

Cover, T. M., & Thomas, J. A. (1991). *Elements of information theory*. John Wiley & Sons.

Ganti, V., Gehrke, J., & Ramakrishnan, R. (1999). CACTUS - clustering categorical data using summaries. *Proceedings of the Fifth ACM SIGKDD Conference* (pp. 73–83).

Gibson, D., Kleinberg, J., & Raghavan, P. (1998). Clustering categorical data: An approach based on dynamical systems. *Proceedings of the 24rd VLDB Conference* (pp. 311–322).

Guha, S., Rastogi, R., & Shim, K. (2000). ROCK: A robust clustering algorithm for categorical attributes. *Information Systems*, *25*, 345–366.

Gyllenberg, M., Koski, T., & Verlaan, M. (1997). Classification of binary vectors by stochastic complexity. *Journal of Multivariate Analysis*, 47–72.

Havrda, J., & Charvat, F. (1967). Quantification method of classification processes: Concept of structrual a-entropy. *Kybernetika*, *3*, 30–35.

Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, *2*, 283–304.

Jardine, N., & Sibson, R. (1971). *Mathematical taxonomy*. John Wiley & Sons.

Li, T., Zhu, S., & Ogihara, M. (2003). Efficient multi-way text categorization via generalized discriminant analysis. *Proceedings of Twelfth ACM CIKM Conference* (pp. 317–324).

McCallum, A. K. (1996). Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/ mccallum/bow.

Roberts, S., Everson, R., & Rezek, I. (2000). Maximum certainty data partitioning. *Pattern Recognition*, *33*, 833–839.

Rubinstein, R. Y. (1981). *Simulation and the monte carlo method*. John Wiley & Sons.

Symons, M. J. (1981). Clustering criteria and multivariate normal mixtures. *Biometrics*, *37*, 35–43.

Wallace, R. S. (1989). *Finding natural clusters through entropy minimization* (Technical Report CMU-CS-89-183). Carnegie Mellon University.

Zhao, Y., & Karypis, G. (2001). *Criterion functions for document clustering: Experiments and analysis* (Technical Report). Department of Computer Science, University of Minnesota.