Quantitative Analyses and Development of a $q$-Incrementation Algorithm for FCM with Tsallis Entropy Maximization

Makoto Yasuda

Department of Electrical and Computer Engineering, Gifu National College of Technology, 2236-2 Kamimakuwa, Motosu-shi, Gifu 501-0495, Japan

Correspondence should be addressed to Makoto Yasuda; yasuda@gifu-nct.ac.jp

Received 4 March 2015; Revised 16 July 2015; Accepted 2 August 2015

Academic Editor: Ning Xiong

Tsallis entropy is a $q$-parameter extension of Shannon entropy. By extremizing the Tsallis entropy within the framework of fuzzy $c$-means clustering (FCM), a membership function similar to the statistical mechanical distribution function is obtained. The Tsallis entropy-based DA-FCM algorithm was developed by combining it with the deterministic annealing (DA) method. One of the challenges of this method is to determine an appropriate initial annealing temperature and a $q$ value, according to the data distribution. This is complex, because the membership function changes its shape by decreasing the temperature or by increasing $q$. Quantitative relationships between the temperature and $q$ are examined, and the results show that, in order to change $u^q_{ik}$ equally, inverse changes must be made to the temperature and $q$. Accordingly, in this paper, we propose and investigate two kinds of combinatorial methods for $q$-incrementation and the reduction of temperature for use in the Tsallis entropy-based FCM. In the proposed methods, $q$ is defined as a function of the temperature. Experiments are performed using Fisher’s iris dataset, and the proposed methods are confirmed to determine an appropriate $q$ value in many cases.

1. Introduction

Statistical mechanics investigates the macroscopic properties of a physical system consisting of multiple elements. In recent years, a popular area of research has been the application of statistical mechanical models or tools to information science.

There exists a strong relationship between the membership functions of fuzzy $c$-means clustering (FCM) [1] and the maximum entropy or entropy regularization methods [2, 3] and the statistical mechanical distribution functions. In other words, FCM, when regularized or maximized with a Shannon-like entropy, yields a membership function that is similar to the Boltzmann (or Gibbs) distribution function [2, 4], and when regularized or maximized with fuzzy-like entropy [5], FCM yields a membership function similar to the Fermi-Dirac distribution function [6]. These membership functions are suitable for annealing methods, because they contain a parameter corresponding to a system temperature. The advantage of using entropy maximization methods is that fuzzy clustering can be interpreted and analyzed from both statistical physical and information-processing points of view.

Tsallis [7] achieved a nonextensive extension of Boltzmann-Gibbs statistics by postulating a generalized form of entropy with a generalization parameter $q$, which, in the limit as $q$ goes to 1, approaches the Shannon entropy. Tsallis entropy is applicable to numerous fields, including physics, chemistry, bioscience, networks, and computer science, and it has proved to be useful [8–10]. For example, Tsallis entropy can be applicable for attribute selection in network intrusion detection [11]. It also can be utilized as an optimization function of thresholding image segmentation [12]. In [13, 14], Menard et al. discussed fuzzy clustering in the framework of nonextensive thermostatistics. By taking the possibilistic constraint into account, the possibilistic membership function was derived, and its properties were considered from various viewpoints.

On the other hand, based on the Tsallis entropy, another form of entropy (or a measure of fuzziness) for a membership function can be defined. A form of the membership function
can then be derived by extremizing (maximizing) this entropy within the framework of FCM [15]. In comparison with the conventional entropy maximization methods [2, 3], this method yields superior results [15].

Deterministic annealing (DA) [4] is a deterministic variant of simulated annealing, and it can be applied to clustering [16]. By applying DA to FCM using Tsallis entropy, a DA-FCM algorithm using Tsallis entropy has been developed [15]. As for another application example of DA, in [17], the $q$-parameterized DA expectation maximization algorithm is introduced.

The analyses show that the temperature and $q$ affect $u_k^q$ almost inversely. Based on these results, we developed two kinds of $q$-incrementation algorithms for Tsallis entropy-based FCM, in which $q$ is defined as a function of the temperature. These algorithms are compared with the conventional Tsallis entropy-based DA-FCM method.

In the first algorithm, $q$ is increased so as to maintain similar shapes of $u_k^q$ with the conventional $T$-reduction method. In the second algorithm, $q$ is defined as an inverse of a decreasing pseudo-temperature.

Experiments are performed using Fisher's iris dataset [18], and it was confirmed that, in many cases, appropriate $q$ value is determined automatically from the temperature. Furthermore, the proposed methods improve the accuracy of classification and are superior to the conventional method.

However, it was also found that the number of computation iterations depends on $T_{\text{high}}$, and sometimes it becomes greater than that of the conventional method; this suggests that $T_{\text{high}}$ should be optimized to some extent.

### 2. Entropy Maximization Method

Let $X = \{x_1, \ldots, x_n\}$ ($x_k = (x_{k1}, \ldots, x_{kp}) \in \mathbb{R}^p$) be a data set in $p$-dimensional real space, which is to be divided into $c$ clusters. In addition, let $V = \{v_1, \ldots, v_c\}$ ($v_i = (v_{i1}, \ldots, v_{ip})$) be the centers of the clusters, and let $u_{kq} \in [0, 1]$ ($i = 1, \ldots, c; k = 1, \ldots, n$) be the membership functions. Furthermore, let

$$J = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{k}^{q} d_{ik} \quad (d_{ik} = \|x_k - v_i\|^2; \ m \in \mathbb{R}; \ m > 1) \quad (1)$$

be the FCM objective function that is to be minimized.

#### 2.1. Entropy Maximization for FCM.

The Tsallis entropy is defined as

$$S_q = -\frac{1}{q-1} \left( \sum_{i=1}^{W} p_i^q - 1 \right) \left( \sum_{i=1}^{W} p_i = 1; \ q \in \mathbb{R} \right), \quad (2)$$

where $W$ is the total number of microscopic possibilities of the system.

Based on (2), the entropy (or a measure of the fuzziness) of a membership function is defined as follows:

$$S = -\frac{1}{q-1} \left( \sum_{k=1}^{n} \sum_{i=1}^{c} u_{k}^{q} d_{ik} \right). \quad (3)$$

The objective function can be written as

$$J = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{k}^{q} d_{ik}. \quad (4)$$

Under the normalization constraint

$$\sum_{i=1}^{n} u_{ik} = 1 \quad (\forall k), \quad (5)$$

the Tsallis entropy functional is given by

$$\delta S - \sum_{k=1}^{n} \alpha_k \delta \left( \sum_{i=1}^{c} u_{ik} - 1 \right) - \beta \sum_{k=1}^{n} \sum_{i=1}^{c} \delta (u_{ik} d_{ik}). \quad (6)$$

where $\alpha_k$ and $\beta$ are the Lagrange multipliers and $\alpha_k$ must be determined so as to satisfy (5).

By extremizing (6) with respect to $u_{ik}$, the stationary condition yields the following membership function:

$$u_{ik} = \frac{1 - \beta (1 - q) d_{ik}^{1/(1-q)}}{Z}. \quad (7)$$

where

$$Z = \sum_{j=1}^{c} \left[ 1 - \beta (1 - q) d_{jk} \right]^{1/(1-q)}. \quad (8)$$

In the same way, the center of the cluster is given by

$$v_i = \frac{\sum_{k=1}^{n} u_{ik} x_k}{\sum_{k=1}^{n} u_{ik}}. \quad (9)$$

$J$ and $S$ satisfy

$$S - \beta J = \sum_{k=1}^{n} Z_{k}^{1-q} - 1 \quad (1-q), \quad (10)$$

which leads to

$$\frac{\partial S}{\partial J} = \beta. \quad (11)$$

By analogy with statistical mechanics, this relationship makes it possible to regard $J$ as the internal energy and $\beta^{-1}$ as an artificial system temperature $T$ [19].
3. Dependencies of $u_{ik}^q$ on Temperature and $q$

In (9), $u_{ik}^q$ works as a weight value to each $x_k$, and it determines $v_j$. In this paper, for simplicity, $v_j$ is set to be 0. This makes the denominator of (7) become the sum of the same forms of its numerator. In Figures 1(a) and 1(b), the numerator of $u_{ik}^q$ is plotted as a function of $x_k$, parameterized by $T$ and $q$, respectively. In these figures, in order to examine the shape of $u_{ik}^q$ (the subscript $ik$ is omitted in this formula) as a function of the distance between the center of the cluster and various data points, $x_k$ is considered to be a continuous variable $x$.

The extent of $u_{ik}^q$ becomes narrower with increasing $q$ and as the temperature decreases, the distribution becomes narrower. This leads to $q$-incrementation clustering instead of annealing or $T$-reduction.

4. Quantitative Relationship between Temperature and $q$

As stated in the previous section, $T$ and $q$ inversely affect the extent of $u_{ik}^q$, which changes in a similar way with increasing $q$ or decreasing $T$. Accordingly, in order to examine the quantitative relationship between $T$ and $q$, we change them independently, as follows.

First, we define

$$u(x, T, q) = \left\{ 1 - \frac{(1 - q) x^2}{T} \right\}^{q/(1-q)}. \quad (12)$$

Then, $u(x, T, q)$ is calculated by fixing $T$ and $q$ to some constants $T_0$ and $q_0$. Next, by decreasing $T$, we determine the $q$ values that minimize the sum of squares of the residuals of these two functions:

$$\sum_{k=0}^{k_{max}} \left| u_{ik}^q(\Delta x_k, T_0, q_0) - u_{ik}^q(\Delta x_k, T, q) \right|^2. \quad (13)$$

In these calculations, the parameters are set as follows: $T_{\text{high}} = T_0$ is set to 2.0; the domain of $x$ is set to $0 \leq x \leq 100$; the number of sampling points of the sum of residuals is 10001 ($k_{\text{max}}$ and $\Delta x$ in (13) are set to 10000 and 0.01, resp.). For $q = q_0$ values of 1.01, 2.0, 6.0, and 10.0 and for $T$ decreasing from $T_{\text{high}}$, the $q$ value that minimizes the sum of squares of the residuals (expressed by $q_{\text{min}}$) is shown in Figure 2(a).

Figure 2(b), on the other hand, shows the results of cases in which $q$ is set to 2.0 and $T$ is lowered from $T_{\text{high}} = 2.0, 20.0, 100.0, 200.0$.

Approximate curves in Figures 2(a) and 2(b) are obtained by fitting the data to the following formula:

$$q_{\text{min}} = aT^{-b}, \quad (14)$$

where $a$ and $b$ are the fitting parameters. Optimal values for these parameters are summarized in Tables 1 and 2. It was found that $b$ is nearly equal to 1.0, suggesting that $q$ is inversely proportional to $T$. In addition, it can be seen that though $b$ does not change its value, $a$ increases with increasing $T$. Accordingly, by using the relationship of $T$ and $q_{\text{min}}$ as shown in Tables 1 and 2, $q$-incrementation clustering is possible.

5. $q$-Incrementation FCM Algorithm

In this section, we develop $q$-incrementation FCM that uses Tsallis entropy instead of annealing. We begin by considering parameters $a$ and $b$ in (14) for $q = 2.0$ and $T_{\text{high}} = 2.0$. In this case, $q$ is derived from $T$ by the following equation:

$$q = 4.67T^{-1.066}. \quad (15)$$

The temperature is held at $T_{\text{high}}$ during the clustering.
The $q$-incrementation FCM algorithm using Tsallis entropy maximization is presented as follows:

(1) Set the number of clusters $c$, the highest temperature $T_{\text{high}}$, the temperature reduction parameter, the thresholds of the convergence test $\delta_1$ and $\delta_2$, and the $q$-incrementation parameter.

(2) Generate initial clusters at random positions and assign each data point to the nearest cluster. Set the current temperature to $T_{\text{high}}$.

(3) Calculate the membership function $u_{ik}$ using (7).

(4) Calculate the cluster centers $v_i$ using (9).

(5) Compare the differences between the current centers and the centers of the previous iteration $v_i'$. If the convergence condition $\max_{1 \leq i \leq c} \|v_i - v_i'\| < \delta_1$ is satisfied, then go to Step (6). Otherwise, return to Step (3).

(6) Compare the difference between the current centers and the centers obtained in the previous iteration $v_i''$. If the convergence condition $\max_{1 \leq i \leq c} \|v_i - v_i''\| < \delta_2$ is satisfied, then stop. Otherwise, update $q$ using (15) and return to Step (3).

### 6. Experiment 1

In Experiment 1, classification results of the conventional $T$-reduction method and the $q$-incrementation methods in the previous section are compared to examine if they give similar results.

In the experiment, we used Fisher’s iris dataset [18], consisting of 150 four-dimensional vectors of iris flowers. The dataset contains three clusters of flowers: versicolor, virginica, and setosa. Each cluster consists of 50 vectors. The parameters were set as follows: $c = 3$, $T_{\text{high}} = 2.0$, $\delta_1 = \delta_2 = 0.01$, the initial value of $q$ was set to 2.0, and $q = 2.0$ was used in the conventional annealing method. For the cooling schedule of the annealing method, we used very fast annealing (VFA) [15, 20] with $c_d = 2.0$.

#### 6.1. Classification Results of the $T$-Reduction and $q$-Incrementation Methods

The maximum, minimum, and average numbers of misclassified data points are summarized in Table 3.

<table>
<thead>
<tr>
<th>Method</th>
<th>Maximum</th>
<th>Minimum</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$-reduction</td>
<td>17</td>
<td>16</td>
<td>16.001</td>
</tr>
<tr>
<td>$q$-incrementation</td>
<td>12</td>
<td>10</td>
<td>11.202</td>
</tr>
</tbody>
</table>

#### Table 1: Parameters of approximate curves ($T_{\text{high}} = 2.0$).

<table>
<thead>
<tr>
<th>$q$</th>
<th>$a$</th>
<th>$b$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.01</td>
<td>2.71</td>
<td>1.126</td>
</tr>
<tr>
<td>2.0</td>
<td>4.67</td>
<td>1.066</td>
</tr>
<tr>
<td>6.0</td>
<td>12.65</td>
<td>1.023</td>
</tr>
<tr>
<td>10.0</td>
<td>20.64</td>
<td>1.014</td>
</tr>
</tbody>
</table>

#### Table 2: Parameters of approximate curves ($q = 2.0$).

<table>
<thead>
<tr>
<th>$T_{\text{high}}$</th>
<th>$a$</th>
<th>$b$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0</td>
<td>4.67</td>
<td>1.066</td>
</tr>
<tr>
<td>20.0</td>
<td>54.33</td>
<td>1.066</td>
</tr>
<tr>
<td>100.0</td>
<td>302.06</td>
<td>1.066</td>
</tr>
<tr>
<td>200.0</td>
<td>632.35</td>
<td>1.066</td>
</tr>
</tbody>
</table>
Table 4: Maximum, minimum, and average numbers of computation iterations required.

<table>
<thead>
<tr>
<th>Method</th>
<th>Maximum</th>
<th>Minimum</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>T-reduction</td>
<td>40</td>
<td>9</td>
<td>21.598</td>
</tr>
<tr>
<td>q-incrementation</td>
<td>34</td>
<td>9</td>
<td>18.954</td>
</tr>
</tbody>
</table>

Figure 3: Plot of $u^q(x, T, q)$, parameterized by $T$ and $q$: $q$ is determined from $T$ by (15).

The maximum, minimum, and average numbers of computation iterations required for the conventional $T$-reduction and the $q$-incrementation methods for 1000 trials are summarized in Table 4. Tables 3 and 4 show that the $q$-incrementation method reduces misclassifications and requires fewer computation iterations than does the $T$-reduction method; this is true even though $q$ is increased in order to minimize the sum of squares of the residuals of $u^q(x, T = \text{const.}, q)$ and $u^q(x, T, q = \text{const.})$. This suggests that there exists a significant difference between the shapes of $u^q_{ik}$ in $T$-reduction and those in $q$-incrementation.

In Figure 3, changes in $u^q(x, T, q)$ as a function of $T$ are plotted. As shown in this figure, $q = 2.23$ and 54.36 were obtained from $T = 2.0$ and 0.1, respectively, by using (15).

By comparing the plots of $u^q_{ik}(x, 2.0, 2.0)$ and $u^q_{ik}(x, 2.0, 2.23)$ and those of $u^q_{ik}(x, 0.1, 2.0)$ and $u^q_{ik}(x, 0.1, 54.36)$, it can be seen that when $T = 2.0$, both $u^q_{ik}$s have a similar shape. However, when $T = 0.1$ and $x$ is large, $u^q_{ik}(x, 0.1, 54.36)$ has a steeper slope than does $u^q_{ik}(x, 0.1, 2.0)$; this results in a lack of agreement of the clustering results.

7. Modified $q$-Incrementation FCM Algorithm

In the previous section, it was confirmed that $q$-incrementation clustering is available for FCM using Tsallis entropy maximization.

In this section, we consider a very simple and general algorithm, in which $T$ is fixed to $T_{\text{high}}$ and $q$ is defined as the inverse of $T'$, where $T'$ is a pseudo-temperature that is decreased using the DA method. That is, $q$ is given as

$$q = \frac{T_{\text{high}} + e}{T'},$$

where $e$ is any small constant (the small constant $e$ is added in order to prevent $q$ from reaching 1.0 when $T' = T_{\text{high}}$; this needs to be avoided because, in the limit of $q = 1.0$, the Tsallis entropy equals the Shannon entropy). Steps (1), (2), and (6) in the algorithm presented in Section 5 should be changed as follows:

(1) Set the number of clusters $c$, the highest temperature $T_{\text{high}}$, the temperature reduction parameter, the thresholds of the convergence test $\delta_1$, and $\delta_2$, and the initial $q$ value.

(2) Generate initial clusters at random positions and assign each data point to the nearest cluster. Set the current temperature $T'$ to $T_{\text{high}}$.

(6) Compare the difference between the current centers and the centers obtained in the previous iteration $\psi_i'$. If the convergence condition $\max_{2\leq i\leq c}\|\psi_i - \psi_i^{'}\| < \delta_2$ is satisfied, then stop. Otherwise, decrease $T'$, update $q$ using (16), and return to Step (3).

8. Experiment 2

In Experiment 2, we compare the classification results of the $T$-reduction method and those of the modified $q$-incrementation method (hereafter the proposed method) that was presented in Section 7.

Fisher’s iris dataset was clustered using the same parameters as were used in Experiment 1, with the exception that $q$ was changed from 1.1 to 10.0 for the conventional method with $T_{\text{high}} = 2.0$, and $T_{\text{high}}$ was changed from 0.01 to 10.0 for the proposed method. VFA was used as the cooling schedule for both methods, and $e$ in (16) was set to 0.01.

8.1. Classification Results of the $T$-Reduction and Modified $q$-Incrementation Methods. The maximum, minimum, and average numbers of misclassified data points and required number of computation iterations for 1000 trials each of the conventional $T$-reduction and the proposed method are summarized in Tables 5 and 6, respectively. In both cases, $T_{\text{high}}$ was set to 2.0.

Tables 5 and 6 show that the conventional method misclassifies fewer data points as $q$ increases. The number of computation iterations required increases with increasing $q$, reaching a maximum at $q = 5.0$.

The numbers of misclassified data points and the number of computation iterations required for the proposed method with $T_{\text{high}} = 2.0$ are close to the minimums of those numbers for the conventional method with $q = 10.0$ and $q = 1.1$, respectively.

In the conventional method, as $q$ increases, the number of misclassified data points decreases; this occurs because $u^q_{ik}$ is narrowly distributed when $q$ is large, and thus clustering is
Table 5: Maximum, minimum, and average numbers of misclassified data points of the $T$-reduction method parameterized by $q$ ($T_{\text{high}} = 2.0$).

<table>
<thead>
<tr>
<th>Method</th>
<th>$q$</th>
<th>Maximum</th>
<th>Minimum</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$-reduction</td>
<td>1.1</td>
<td>69</td>
<td>17</td>
<td>25.116</td>
</tr>
<tr>
<td></td>
<td>2.0</td>
<td>17</td>
<td>16</td>
<td>16.002</td>
</tr>
<tr>
<td></td>
<td>3.0</td>
<td>17</td>
<td>14</td>
<td>14.949</td>
</tr>
<tr>
<td></td>
<td>5.0</td>
<td>16</td>
<td>13</td>
<td>13.021</td>
</tr>
<tr>
<td></td>
<td>8.0</td>
<td>14</td>
<td>11</td>
<td>12.017</td>
</tr>
<tr>
<td></td>
<td>10.0</td>
<td>14</td>
<td>11</td>
<td>11.063</td>
</tr>
<tr>
<td>Proposed</td>
<td>2.0</td>
<td>13</td>
<td>11</td>
<td>11.576</td>
</tr>
</tbody>
</table>

Table 6: Maximum, minimum, and average numbers of computation iterations required for the $T$-reduction method parameterized by $q$ ($T_{\text{high}} = 2.0$).

<table>
<thead>
<tr>
<th>Method</th>
<th>$q$</th>
<th>Maximum</th>
<th>Minimum</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$-reduction</td>
<td>1.1</td>
<td>41</td>
<td>10</td>
<td>18.510</td>
</tr>
<tr>
<td></td>
<td>2.0</td>
<td>38</td>
<td>9</td>
<td>21.268</td>
</tr>
<tr>
<td></td>
<td>3.0</td>
<td>41</td>
<td>9</td>
<td>23.930</td>
</tr>
<tr>
<td></td>
<td>5.0</td>
<td>40</td>
<td>11</td>
<td>25.547</td>
</tr>
<tr>
<td></td>
<td>8.0</td>
<td>39</td>
<td>10</td>
<td>24.801</td>
</tr>
<tr>
<td></td>
<td>10.0</td>
<td>36</td>
<td>10</td>
<td>24.678</td>
</tr>
<tr>
<td>Proposed</td>
<td>2.0</td>
<td>48</td>
<td>8</td>
<td>18.865</td>
</tr>
</tbody>
</table>

Table 7: Maximum, minimum, and average numbers of misclassified data points with the modified $q$-incrementation method parameterized by $T_{\text{high}}$.

<table>
<thead>
<tr>
<th>$T_{\text{high}}$</th>
<th>Maximum</th>
<th>Minimum</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>49</td>
<td>12</td>
<td>12.037</td>
</tr>
<tr>
<td>8.0</td>
<td>12</td>
<td>12</td>
<td>12.000</td>
</tr>
<tr>
<td>5.0</td>
<td>13</td>
<td>12</td>
<td>12.995</td>
</tr>
<tr>
<td>2.0</td>
<td>13</td>
<td>11</td>
<td>11.576</td>
</tr>
<tr>
<td>1.0</td>
<td>12</td>
<td>11</td>
<td>11.000</td>
</tr>
<tr>
<td>0.1</td>
<td>12</td>
<td>12</td>
<td>12.000</td>
</tr>
<tr>
<td>0.01</td>
<td>12</td>
<td>12</td>
<td>12.000</td>
</tr>
</tbody>
</table>

Table 8: Maximum, minimum, and average numbers of computation iterations required by the modified $q$-incrementation method parameterized by $T_{\text{high}}$.

<table>
<thead>
<tr>
<th>$T_{\text{high}}$</th>
<th>Maximum</th>
<th>Minimum</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>70</td>
<td>4</td>
<td>50.494</td>
</tr>
<tr>
<td>8.0</td>
<td>80</td>
<td>20</td>
<td>39.937</td>
</tr>
<tr>
<td>5.0</td>
<td>42</td>
<td>13</td>
<td>21.638</td>
</tr>
<tr>
<td>2.0</td>
<td>48</td>
<td>8</td>
<td>18.865</td>
</tr>
<tr>
<td>1.0</td>
<td>39</td>
<td>17</td>
<td>22.262</td>
</tr>
<tr>
<td>0.1</td>
<td>41</td>
<td>19</td>
<td>27.170</td>
</tr>
<tr>
<td>0.01</td>
<td>39</td>
<td>21</td>
<td>27.906</td>
</tr>
</tbody>
</table>

done locally and optimally. On the other hand, the number of required computation iterations tends to decrease with decreasing $q$, because when $q$ is small, clustering has been done widely and efficiently.

In the proposed method, $q$ is initially given as $q = (T_{\text{high}} + e)/T_{\text{high}}$, which is nearly equal to 1.0. Thus, at an early stage in annealing, clustering is automatically done widely. This is because the modified method does not require as many iterations.

In summary, the conventional method has an inconsistency, in that the $q$ value that minimizes the number of misclassified data points increases the number of computation iterations that are required. However, by setting $T_{\text{high}}$ to be the same as the value used in the conventional method, the proposed method is better than the conventional method to balance the number of misclassified data points with the number of computation iterations that are required.

8.2. Properties of the Modified $q$-Incrementation Method. In this subsection, we examine the reason why the proposed method has improved clustering.

Tables 7 and 8 summarize, for $T_{\text{high}}$ changing from 10.0 to 0.01, the maximum, minimum, and average numbers of misclassified data points and the number of computation iterations required for 1000 trials of the proposed method.

In Table 8, the number of computation iterations required by the proposed method decreases with decreasing $T_{\text{high}}$ until $T_{\text{high}} = 2.0$. After that point, it begins to increase, suggesting that there exist minima in $T_{\text{high}}$. The reasons for this property are considered to be as follows. When $T_{\text{high}}$ is as high as 10.0, the relative size of $q$ is too small to change the shape of $u_{ik}^q$, and this increases the required number of computation iterations. However, when $T_{\text{high}} = 0.01$, the width of $u_{ik}^q$ is very narrow, and a long time is required for the centers of the clusters to converge. For these reasons, we assume there is at least one minimum in $T_{\text{high}}$.

As stated in the previous subsection, the proposed method can limit the number of misclassified data points to as few as 11 to 13 points. Thus, we conclude that $T_{\text{high}}$ does not significantly affect the number of misclassifications.

In summary, our experiments confirmed that when $q$ increases as the inverse of the decreasing pseudo-temperature, the proposed method works at least as well as the conventional method. However, the number of computation iterations required by the proposed method apparently depends on the value of $T_{\text{high}}$, and it is not yet known how to determine an appropriate value of $T_{\text{high}}$ for a given dataset.

9. Conclusion

We formulated the membership function of Tsallis entropy-based FCM by maximizing the Tsallis entropy functional. In this formulation, the $q$-parameter of the Tsallis entropy strongly affects the accuracy of the clustering.

In order to determine an appropriate value of $q$ for a given data distribution, it is first necessary to examine quantitatively the effect of $q$ on the extent of $u_{ik}^q$. We determined that, in order to minimize the square of the residual of $u_{ik}^q$ for the $T$-reduction and $q$-incrementation, $q$ must be increased as the inverse of the temperature.
Based on this relationship, we proposed two kinds of $q$-incrementation methods and combined them with the DA method. In the first method, $q$ is increased according to the approximation function that minimizes the square of the residual of $u_{ik}^q$. The experimental results show that, compared to the conventional annealing method, the proposed method reduces both the number of misclassifications and the number of required computation iterations.

In the second method, $q$ is simply defined as the inverse of the decreasing pseudo-temperature. The experimental results reveal that, in most cases, this method determines an appropriate $q$ value, has improved accuracy, and is superior to the conventional method.

However, the results also confirm that the number of computation iterations depends on $T_{\text{high}}$, and, in some cases, it can become greater than that of the conventional method. This should be avoidable by using the value of $T_{\text{high}}$ that minimizes the number of iterations.

In the future, first we intend to estimate the validity of our approximation method used in Sections 3 and 4 accurately. We then intend to explore ways to increase efficiency by performing convergence and computation time tests using various formulas of $q$ and $T$. Furthermore, we intend to develop a better schedule for annealing and a method for optimizing $T_{\text{high}}$.

**Conflict of Interests**

The author declares that there is no conflict of interests regarding the publication of this paper.

**Acknowledgment**

The present study was supported by JSPS KAKENHI Grant no. 25330297.

**References**


