Hierarchical Clustering of Time Series Based on Linear Information Granules

Hailan CHEN, Xuedong GAO, Yifan GUO

Abstract: Time series clustering is one of the main tasks in time series data mining. In this paper, a new time series clustering algorithm is proposed based on linear information granules. First, we improve the identification method of fluctuation points using threshold set, which represents the main trend information of the original time series. Then using fluctuation points as segmented nodes, we segment the original time series into several information granules, and linear function is used to represent the information granules. With information granulation, a granular time series consisting of several linear information granules replaces the original time series. In order to cluster time series, we then propose a linear information granules based segmented matching distance measurement (LIG_SMD) to calculate the distance between every two granular time series. In addition, hierarchical clustering method is applied based on the new distance (LIG_SMD_HC) to get clustering results. Finally, some public and real datasets about time series are experimented to examine the effectiveness of the proposed algorithm. Specifically, Euclidean distance based hierarchical clustering (ED_HC) and Dynamic Time Warping distance based hierarchical clustering (DTW_HC) are used as the compared algorithms. Our results show that LIG_SMD_HC is better than ED_HC and DTW_HC in terms of F-Measure and Accuracy.

Keywords: distance measurement; hierarchical clustering; information granules; time series

1 INTRODUCTION

Time series data arises in many areas, such as finance [1], medicine [2], environment [3] and traffic [4]. A time series is a collection of data points made chronologically. In recent years, researchers have paid more attention to time series data mining techniques, including clustering [5, 6], association rule discovery [7], classification [8, 9] and prediction [10, 11]. Such techniques tend to be used to extract meaningful knowledge and patterns of time series data. Clustering, among these techniques, is generally considered a method for data pre-processing, so it also plays an important role for the use of techniques in time series data mining.

Clustering is a process of finding natural groups in a dataset, also known as clusters. The objective is to find the most homogeneous clusters that are as distinct as possible from other clusters [12]. Any clustering technique need to rely on two concepts [13]: clustering algorithm and distance measurement. It is obvious that distance measurement is the basis of clustering algorithm, as it has a significant impact on clustering results.

The traditional time series similarity measurements are classified into five categories [14]: Euclidean distance, Dynamic Time Warping distance, Symbolic distance, Model distance and Compression distance. The most widely used distance measurement is Euclidean distance [15], but it performs unsatisfactorily in terms of measuring time series trend features. In 1994, Berndt and Clifford [16] proposed Dynamic Time Warping (DTW) distance and applied it to discover patterns in time series. DTW can measure time series of different length, and it is also robust to time series offset and amplitude variation. However, the demerits of this measure include a high computational complexity, and its failure to satisfy the triangular inequality of distance. For the similarity measurement based on symbolic distance [17], time series are divided into intervals, and a short trend symbol sequence is determined by the trends judgments of intervals. Then the connectivity indexes of each trend symbol are calculated. Finally, the Mitani coefficients of the connectivity index for each trend symbol are calculated. However, this method entails the conversion of time series into the corresponding trend symbols, and its accuracy is not satisfying. Consequently, recent academic attention [18-21] has turned to information granules and granular computing for time series distance measurements.

The concept of information granules was firstly proposed by Zadeh [22]. It refers to decomposing a whole into small parts, and each part is considered as a granule. In other words, the information granules is a collection of elements, which are similar, indistinguishable, or functional [23]. In the field of time series data mining, there are various types of information granules, such as interval information granule [24], rough information granule [25] and fuzzy information granule [18]. In general, information granulation mainly includes two phases: information granule division and information granule representation. However, most researchers [26, 27] focus more on the methods of information granule representation, ignoring the study of information granule division. Previous literature usually uses the fixed time interval to segment the time series. Such operation does not take into consideration the trend features of time series, and hence the information granules cannot effectively represent the trend information.

In recent years, fuzzy information granule (FIG) has been widely studied, and there are many fuzzy membership functions. However, this approach ignores the concept of time. In addition, a few researchers [21] have used variable time intervals to divide information granules. For example, 11 trend filtering [28] is introduced to divide and represent time series, which can extract the underlying linear trend and give a linear fitting of time series, but the selected parameter has a great impact on the results. Therefore, in this paper we propose a new method of information granulation including information granule division and information granule representation.

The rest of this paper is organized as follows. In Section 2, we identify the fluctuation points to segment the original time series into information granules, and use linear function to describe each information granule. In Section 3, we propose a new distance measurement—segmented matching distance based on linear information granule (LIG_SMD), which is used with hierarchical
clustering algorithm. In Section 4, we apply the proposed clustering algorithm—hierarchical clustering based on LIG_SMD (LIG_SMD_HC) to some public and real datasets, and present its performance. Finally, Section 5 summarizes our method and points out problems to be improved in future study.

2 INFORMATION GRANULATION OF TIME SERIES

Information granulation for time series mainly includes two phases: information granule division and information granule representation. Information granule division is that the time series is segmented into several small subsequences, which are then used as operation windows. For the information granule representation, the method is constructed on the divided window instead of the original window information. In this section, we propose the method of information granule division and information granule representation.

2.1 Information Granule Division

In this section, we introduce the method of fluctuation points identification [29] to divide the original time series into several information granules.

2.1.1 Identification of Extreme Points

**Definition 1 (Extreme Point)** Suppose there are three consecutive points of uniform sampling sequence \( x_{i-1}, x_i, x_{i+1} \), if \( (x_i - x_{i-1})(x_{i+1} - x_i) < 0 \), then \( x_i \) is an Extreme Point. Where \( i = 2, 3, \ldots, n - 1 \).

Using the above definition to identify the extreme points of a time series, the result is shown in Fig. 1.

![Figure 1 Identification of extreme points](image1.png)

2.1.2 Determination of Fluctuation Points

As shown in Fig. 1, there are many extreme points distributed on the curve segments, some of which are not obvious. Therefore, we need to set the threshold \( \varepsilon \) to select the points that its variety is larger from the extreme point. Meanwhile, we mark the attribute of the extreme points as 1 or \(-1\).

After selecting extreme points, the remaining points that cannot represent the trend features of the original time series are called candidate fluctuation points. In order to obtain the fluctuation points, we perform the corresponding operations. The definition of fluctuation point is given as below.

**Definition 2 (Fluctuation Point)** Suppose there are two consecutive points of candidate fluctuation points \( x_{i-1}, x_i \), if \( |x_i - x_{i-1}| > \varepsilon \) and \( Attr_{x_i} 	imes Attr_{x_{i-1}} = -1 \), then \( x_i \) is an Fluctuation Point.

Where \( \varepsilon \) is the threshold, \( Attr_{x_i} \) is the attribute of \( x_i \), and \( i = 2, 3, \ldots, n \).

The detailed operations are as follows. When the attributes of two consecutive points are 1, we delete the minimum one of two points. Conversely, when the attributes of two consecutive points are \(-1\), we delete the maximum one of them. Repeating this operation until \( Attr_{x_i} 	imes Attr_{x_{i-1}} = -1 \), then the remaining points are fluctuation points.

According to the above operation, fluctuation points can be identified (Fig. 3).

![Figure 2 Fluctuation points of a time series using the single threshold](image2.png)

![Figure 3 Fluctuation points of a time series using the threshold set](image3.png)

In [30] a single threshold method is used to filter the extreme points to identify the fluctuation points. Fig. 2 shows the fluctuation points that are obtained by the single threshold method. It can be seen that the method fails to identify some fluctuation points. In order to solve this problem, a method of setting a threshold set is proposed. By filtering each threshold in the threshold set, the fluctuation points are finally obtained, which are shown in
Fig. 3. It is obvious that this method is more accurate in identifying the fluctuation points.
To sum up, the process of fluctuation points identification is presented in Algorithm 1.

Algorithm 1 Fluctuation Points Identification

**Input**: a time series

**Output**: fluctuation points of the time series

**Step 1**: Identify the extreme points of the time series, and mark the attributes of them. Where \( \text{Attr}_x = 1 \) represent \( x \) is a maximum point, and \( \text{Attr}_x = -1 \) represent \( x \) is a minimum point;

**Step 2**: Set the threshold set to filter extreme points;

**Step 3**: For each threshold of the threshold set, select the points from extreme points that are greater than the threshold. For \( \text{Attr}_x \times \text{Attr}_x = 1 \), when the attributes of two consecutive points are 1, delete the minimum one of them; when the attributes of two consecutive points are \(-1\), delete the maximum one of them; repeat this process until \( \text{Attr}_x \times \text{Attr}_x = -1 \);

**Step 4**: CycleStep 3 from the first value to the last one of the threshold set.

**Step 5**: Finally, the remaining points are fluctuation points.

By identifying fluctuation points, the information granule division of time series is finished. In other words, an information granule is formed between two adjacent fluctuation points in the time series.

2.2 Information Granule Representation

Completing information granule division, subsequently we need to describe each information granule. The traditional information granule representation method adopts fuzzy membership function, which mainly includes triangular membership function, parabolic membership function, and Gaussian membership function. Nevertheless, these membership functions ignore the concept of time. Thus, in this section we use the linear function to describe the information granules, which is obviously time-dependent.

![Figure 4](image)

**Figure 4** The information granule representation of a time series

Based on information granule division, the original series is divided into a set of subsequences. Given a subsequence \( Y = \{ Y_1, Y_2, ..., Y_l \} \), a \( LIG(k, b) \) is generated through linear function \( Y = kt + b \), where \( k, b \) can be calculated by two adjacent fluctuation points.

For the time series in the previous section, we use linear function to describe the divided information granules (Fig. 4).

In Fig. 4, there are 13 information granules, and each of them is represented in different linear functions (Tab. 1).

<table>
<thead>
<tr>
<th>Number</th>
<th>Time Interval</th>
<th>( LIG )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[1, 31]</td>
<td>( y = -0.019t - 0.499 )</td>
</tr>
<tr>
<td>2</td>
<td>[31, 72]</td>
<td>( y = 0.053t + 2.743 )</td>
</tr>
<tr>
<td>3</td>
<td>[72, 108]</td>
<td>( y = -0.025t + 2.857 )</td>
</tr>
<tr>
<td>4</td>
<td>[108, 124]</td>
<td>( y = 0.035t - 3.548 )</td>
</tr>
<tr>
<td>5</td>
<td>[124, 137]</td>
<td>( y = -0.041t + 5.773 )</td>
</tr>
<tr>
<td>6</td>
<td>[137, 152]</td>
<td>( y = 0.056t - 7.461 )</td>
</tr>
<tr>
<td>7</td>
<td>[152, 178]</td>
<td>( y = -0.054t + 9.256 )</td>
</tr>
<tr>
<td>8</td>
<td>[178, 210]</td>
<td>( y = 0.068t + 12.370 )</td>
</tr>
<tr>
<td>9</td>
<td>[210, 214]</td>
<td>( y = 0.099t + 22.503 )</td>
</tr>
<tr>
<td>10</td>
<td>[214, 221]</td>
<td>( y = 0.039t - 6.892 )</td>
</tr>
<tr>
<td>11</td>
<td>[221, 227]</td>
<td>( y = -0.129t + 30.199 )</td>
</tr>
<tr>
<td>12</td>
<td>[227, 233]</td>
<td>( y = 0.078t - 16.808 )</td>
</tr>
<tr>
<td>13</td>
<td>[233, 286]</td>
<td>( y = -0.062t + 15.922 )</td>
</tr>
</tbody>
</table>

After information granulation, several linear information granules (LIG) form a granular time series to represent the original time series.

3 Time Series Clustering Based on LIG_SMD

The next step after information granulation is to calculate the distance of each two linear information granules. It is also the most important part for clustering. Since the time points and the time intervals of two linear information granules from any two time series are different, in this section we propose a new distance measurement—linear information granules based segmented matching distance (LIG_SMD). Afterwards, we apply hierarchical clustering method based on the new distance to give the clustering result.

3.1 Segmented Matching of LIG

Since the information granular division of each time series is different, when calculating the distance of two time series, firstly we need to match the linear information granules of each two time series. Fig. 5 shows the LIG of two time series.

![Figure 5](image)

**Figure 5** LIG of time series A and B
In order to match the linear information granules of each two time series, the following requirements must be met:

1. The starting point and the end point of two LIGs from two time series should be as identical as possible;
2. One LIG of a time series can be matched with multiple LIGs of another time series. That is, the matching relationship of LIG is 1:1 or 1:n.

The matching progress of LIG for any two time series is as follows (Algorithm 2).

**Algorithm 2 Segmented Matching of LIG**

**Input:** LIG of time series A, LIG of time series B

**Output:** the segmented matching LIG between time series A and B

**Step 1:** Extract all time points of LIG from time series A and B;

**Step 2:** Delete the repeated time points to obtain a chronologically non-repeating time points set T;

**Step 3:** According to T to segment and match LIG.

![Figure 6 Segmented matching of LIG from two time series](image)

**Table 2** LIG matching of time series A and B

<table>
<thead>
<tr>
<th>Number</th>
<th>Time Interval</th>
<th>LIG of Time Series A</th>
<th>LIG of Time Series B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[1, 31]</td>
<td>y = -0.019r - 0.499</td>
<td>y = -0.032r + 0.050</td>
</tr>
<tr>
<td>2</td>
<td>[31, 33]</td>
<td>y = 0.053r - 2.743</td>
<td>y = -0.032r + 0.050</td>
</tr>
<tr>
<td>3</td>
<td>[33, 50]</td>
<td>y = 0.033r - 2.743</td>
<td>y = -0.035r + 2.160</td>
</tr>
<tr>
<td>4</td>
<td>[50, 72]</td>
<td>y = -0.053r + 2.743</td>
<td>y = -0.025r + 1.379</td>
</tr>
<tr>
<td>5</td>
<td>[72, 103]</td>
<td>y = -0.025r + 2.857</td>
<td>y = -0.025r + 1.379</td>
</tr>
<tr>
<td>6</td>
<td>[103, 108]</td>
<td>y = -0.025r + 2.857</td>
<td>y = 0.020r - 3.300</td>
</tr>
<tr>
<td>7</td>
<td>[108, 109]</td>
<td>y = 0.035r - 3.548</td>
<td>y = 0.020r - 3.300</td>
</tr>
<tr>
<td>8</td>
<td>[109, 116]</td>
<td>y = 0.035r - 3.548</td>
<td>y = -0.017r + 0.698</td>
</tr>
<tr>
<td>9</td>
<td>[116, 124]</td>
<td>y = 0.035r - 3.548</td>
<td>y = 0.035r - 5.249</td>
</tr>
<tr>
<td>10</td>
<td>[124, 137]</td>
<td>y = -0.041r + 5.773</td>
<td>y = 0.035r - 5.249</td>
</tr>
<tr>
<td>11</td>
<td>[137, 145]</td>
<td>y = 0.056r - 7.461</td>
<td>y = 0.035r - 5.249</td>
</tr>
<tr>
<td>12</td>
<td>[145, 147]</td>
<td>y = 0.056r - 7.461</td>
<td>y = -0.062r + 8.769</td>
</tr>
<tr>
<td>13</td>
<td>[147, 152]</td>
<td>y = 0.056r - 7.461</td>
<td>y = 0.025r - 4.062</td>
</tr>
<tr>
<td>14</td>
<td>[152, 178]</td>
<td>y = -0.054r + 9.256</td>
<td>y = 0.025r - 4.062</td>
</tr>
<tr>
<td>15</td>
<td>[178, 196]</td>
<td>y = 0.068r - 12.370</td>
<td>y = 0.025r - 4.062</td>
</tr>
<tr>
<td>16</td>
<td>[196, 210]</td>
<td>y = 0.068r - 12.370</td>
<td>y = 0.016r + 3.945</td>
</tr>
<tr>
<td>17</td>
<td>[210, 214]</td>
<td>y = -0.099r + 22.503</td>
<td>y = 0.016r + 3.945</td>
</tr>
<tr>
<td>18</td>
<td>[214, 221]</td>
<td>y = 0.039r - 6.892</td>
<td>y = -0.016r + 3.945</td>
</tr>
<tr>
<td>19</td>
<td>[221, 227]</td>
<td>y = -0.129r + 30.199</td>
<td>y = -0.016r + 3.945</td>
</tr>
<tr>
<td>20</td>
<td>[227, 233]</td>
<td>y = 0.078r + 16.808</td>
<td>y = -0.016r + 3.945</td>
</tr>
<tr>
<td>21</td>
<td>[233, 235]</td>
<td>y = -0.062r + 15.922</td>
<td>y = -0.016r + 3.945</td>
</tr>
<tr>
<td>22</td>
<td>[235, 239]</td>
<td>y = -0.062r + 15.922</td>
<td>y = 0.019r + 4.165</td>
</tr>
<tr>
<td>23</td>
<td>[239, 264]</td>
<td>y = -0.062r + 15.922</td>
<td>y = 0.027r + 7.801</td>
</tr>
<tr>
<td>24</td>
<td>[284, 286]</td>
<td>y = -0.062r + 15.922</td>
<td></td>
</tr>
</tbody>
</table>

According to Algorithm 2, we segment the LIG of two time series (Fig. 6), and then match them. As shown in Fig. 7, except for the first and last part, the whole middle parts are matched.

### 3.2 The New Distance Measurement Based on Segmented Matching LIG

The traditional distance measurements are mostly based on point-to-point calculation mode, such as Euclidean distance and Dynamic Time Warping (DTW) distance. It is apparent that this pattern of matching points is costly in time.

After matching LIG, the original time series is cut into subsequences. Next the essential step is to calculate the distance of each segmented matching LIG and sum them up to obtain the final distance. In this section, we use calculating integral to represent the distance between the corresponding segmented LIG of two time series, which is based on piece-to-piece calculation mode.

The segmented linear information granules are equal in length except for the first and last parts. According to the different morphological features of LIG, the distance is considered in four cases.

![Figure 7 Four cases of distance calculation](image)

**Case 1:** As shown in Fig. 7(a), the two linear information granules do not intersect in the time interval \([T_1, T_2]\), and the distance between LIG1 and LIG2 is:

\[
D(LIG_1, LIG_2) = \int_{T_1}^{T_2} [(k_1t + b_2) - (k_2t + b_2)]dt
\]

**Case 2:** As shown in Fig. 7 (b), the two linear information granules intersect at \(T^*\) in the time interval \([T_1, T_2]\), and the distance between LIG1 and LIG2 is:

\[
D(LIG_1, LIG_2) = \int_{T_1}^{T^*} [(k_1t + b_2) - (k_2t + b_2)]dt + \int_{T^*}^{T_2} [(k_1t + b_1) - (k_2t + b_2)]dt
\]

where \(T^* = (k_1 - b_2) / (k_1 - k_2)\).

**Case 3:** As shown in Fig. 7 (c), time series A has no information granule in the time interval \([0, T_1]\), and in \([T_1, T_2]\),
$T_2$ the linear information granule is $k_1t+b_1$; while for time series B, in the time interval $[0, T_2]$ the linear information granule is $k_2t+b_2$, in that way, the distance between $LIG_1(T_1, T_2)$ and $LIG_2(0, T_2)$ is:

$$D(LIG_1(T_1, T_2), LIG_2(0, T_2)) = \int_0^{T_2} [(k_1t+b_1)-(k_2t+b_2)]dt$$

(3)

Case 4: As shown in Fig. 7 (d), the linear information granule of time series A is $k_1t+b_1$ in the time interval $[T_1, T_3]$; while for time series B, the linear information granule is $k_2t+b_2$ in $[T_1, T_3]$, and there is no information granule in the time interval $[T_2, T_3]$. Then the distance between $LIG_1(T_1, T_3)$ and $LIG_2(T_2, T_3)$ is:

$$D(LIG_1(T_1, T_3), LIG_2(T_2, T_3)) = \int_{T_1}^{T_2} [(k_1t+b_1)-(k_2t+b_2)]dt$$

(4)

In summary, the distance between time series A and B is:

$$D(A, B) = \sum_{i=1}^{n} D(LIG_A^i, LIG_B^i)$$

(5)

where $n$ is the number of segmented matching LIG, $LIG_A^i$ is $i$-LIG of time series A, $LIG_B^i$ is $i$-LIG of time series B.

The algorithm process of LIG_SMD is as follows:

**Algorithm 3** Linear Information Granules based Segmented Matching Distance (LIG_SMD)

**Input:** the segmented matching LIG between time series A and B

**Output:** the distance between time series A and B

**Step 1:** Using Algorithm 2, we get the segmented matching LIG between time series A and B;

**Step 2:** Judge whether there are intersections between each pair of segmented LIG except for the first and last pairs; if they belong to (a), calculate the distance according to Eq. (1); if they belong to (b), find $T^*$ and calculate the distance according to Eq. (2);

**Step 3:** Calculate the distance between the first pair of segmented LIG according to Eq. (3), and calculate the distance between the last pair of segmented LIG according to Eq. (4);

**Step 4:** Finally, sum up the distance of each pair to get the total distance between time series A and B.

### 3.3 Hierarchical Clustering Method based on The New Distance

The commonly used clustering methods include segmentation clustering method, hierarchical clustering method, density-based clustering method and grid-based clustering method [30].

Hierarchical clustering is a prototype-based clustering algorithm that attempts to divide datasets at different levels to form a tree-like clustering structure. This method can help us interpret the clustering results in a visual way by drawing a dendrogram. Moreover, we do not need to specify the number of classes before clustering.

According to Algorithm 3, the distance between every two time series is calculated. Then we apply hierarchical clustering method based on the new distance proposed in the previous section.

In conclusion, we summarize the frame of LIG_SMD based Hierarchical Clustering (LIG_SMD_HC) method as illustrated by Fig. 8.

![Figure 8 The framework of LIG_SMD_HC](image)

### 4 EXPERIMENTAL STUDIES

In this section, we will apply the proposed clustering method (LIG_SMD_HC) on some public and real datasets to show its effectiveness. The compared methods are hierarchical clustering based on Euclidean distance (ED_HC) and hierarchical clustering based on Dynamic Time Warping distance (DTW_HC). In order to validate the clustering quality of these methods, we use F-Measure and Accuracy as evaluation metrics. Before the experiment, we first introduce the calculation method of these metrics.

#### 4.1 Evaluation Metrics

Given any two sample points, if they belong to the same class before and after clustering, they are called positive events $T$; in contrast, if they belong to the same class before clustering, but not after clustering, they are called negative events $F$.

Then True Positive ($TP$), False Negative ($FN$), False Positive ($FP$) and True Negative ($TN$) can be defined. $TP$ refers to the number of sample pairs that are in the same class before and after clustering. $FN$ refers to the number of sample pairs that are in the same class before clustering but not in the same class after clustering. $FP$ refers to the number of sample pairs that are not in the same class before
clustering but are clustered in the same cluster after clustering. \( TN \) refers to the number of sample pairs that are not in the same class before and after clustering.

Based on TP, FN, FP and TN, we could calculate Recall, Precision, Accuracy and \( F \)-Measure, and the equations of them are given as below.

\[
\text{Recall} = \frac{TP}{TP + FN}
\]

\[
\text{Precision} = \frac{TP}{TP + FP}
\]

\[
\text{Accuracy} = \frac{TN + TP}{TN + TP + FP + FN}
\]

\[
F - \text{Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]

4.2 Experiments

4.2.1 Experiment A on FaceFour Dataset

Experiment A operates on FaceFour dataset in the UCR Time Series Classification Archive [31]. 6 time series from 3 classes is selected to experiment as shown in Fig. 9. In addition, the clustering results of different methods are shown in Fig. 10.

As shown in Fig. 10, the clustering result of ED_HC is \( \{1, 2\} \) and \( \{3, 4\} \) and \( \{5, 6\} \) using LIG+SMD_HC. This result is in line with the real classes division. In Fig. 8, the trend and shape of the 6 time series across in the 3 classes are very similar, and inter-class difference is not very obvious.

The above experiment proves that the proposed clustering method in this paper is effective, particularly in distinguishing time series that have similar shapes of different classes.

4.2.2 Experiment B on 8 UCR Datasets

The data of experiment B are eight datasets in the UCR Time Series Classification Archive [31]. The basic information of these datasets is shown in Tab. 3. We also use ED_HC and DTW_HC as the compared methods, and K-Means method is added in comparison. Moreover, we use \( F \)-Measure and Accuracy to validate the clustering quality of these methods. Tab. 4 presents the calculated results of 4 clustering methods.

As shown in Tab. 4 and Tab. 5, LIG+SMD_HC achieves the highest \( F \)-Measure on 4 datasets and the highest Accuracy on 5 datasets respectively. To make it easier to observe the results, we sort the \( F \)-Measure and Accuracy values calculated by each clustering method for eight datasets, then calculate the average rank. The average ranks of \( F \)-Measure for these four methods are 1.625, 2.375, 2.625 and 3.125, with average rank of Accuracy of 1.375, 2, 1.5 and 3.875 respectively. Therefore, compared
to other three clustering methods, the result of LIG_SMD_HC performs better in clustering.

4.2.3 Experiment C on Real Stock Dataset

Experiment C is carried on a real stock dataset which consists of the closing prices from 2016.1.4 to 2017.9.22 for 37 stocks, totalling 423 trading days. The stock price data were downloaded from NetEase Finance [32]. We normalize the dataset and fill in the missing values. The stock industry in this dataset mainly involves 8 industries, including finance, security, real estate, food and beverage, energy, electronic equipment, medical biology, and information technology. The industry of stocks is used as a class label to evaluate the clustering results (Tab. 6).

Table 6 Comparison of three clustering methods on real stock dataset in terms of F-Measure and Accuracy

<table>
<thead>
<tr>
<th>Evaluation Index</th>
<th>LIG_SMD_HC</th>
<th>ED_HC</th>
<th>DTW_HC</th>
</tr>
</thead>
<tbody>
<tr>
<td>F-Measure</td>
<td>0.5659</td>
<td>0.5427</td>
<td>0.4718</td>
</tr>
<tr>
<td>Accuracy</td>
<td>0.6486</td>
<td>0.6216</td>
<td>0.5676</td>
</tr>
</tbody>
</table>

Since the trend features of stocks are not completely related to the industry, the F-Measure and Accuracy values are not very good considering the industry as the class label. Even so, the proposed method LIG_SMD_HC is still more accurate than ED_HC and DTW_HC.

5 CONCLUSION

In this study, a new time series clustering algorithm LIG_SMD_HC is proposed. We first improve the method of fluctuation points identification by using threshold set, and this method turns out to be more accurate in identifying fluctuation points. Then using fluctuation points we segment the original time series into several information granules, and linear function is applied to represent the information granules. After information granulation, several linear information granules form a granular time series on behalf of the original time series. Then, we propose a new distance measurement LIG_SMD, which consists of two steps: matching the segmented LIG and calculating the corresponding distance. Finally, the experiments operate on some public and real datasets, and the results show that the LIG_SMD_HC algorithm is superior in accuracy than ED_HC and DTW_HC with respect to F-Measure and Accuracy metrics.

Since the representation of information granules is linear function, it is more suitable for the time series with severe fluctuations. In the future, we will study the method that more accurately represents the various shapes and features of information granules for time series.

Acknowledgements

This work is supported by national science fund of China (No. 71272161).

6 REFERENCES


Contact information:

Hailan CHEN, PhD Candidate
Donlinks School of Economics and Management,
University of Science and Technology Beijing,
No. 30 Xueyuan Road, Haidian District, Beijing, China
chi_hld@163.com

Xuedong GAO, Full Professor
(Corresponding author)
Donlinks School of Economics and Management,
University of Science and Technology Beijing,
No. 30 Xueyuan Road, Haidian District, Beijing, China
gaoxuedong@manage.ustb.edu.cn

Yifan GUO, M.S. Candidate
School of Management,
China University of Mining and Technology,
Ding No. 11 Xueyuan Road, Haidian District, Beijing, China
guo_onlyIool@163.com