Clustering Stream Data by Regression Analysis

Masahiro MOTOYOSHI† Takao MIURA† Isamu SHIOYA‡

† Dept.of Elect.& Elect. Engr.
HOSEI University,
3-7-2 KajinoCho, Koganei, Tokyo, 184–8584 Japan
Email: {i02r3243, miurat}@k.hosei.ac.jp
‡ Dept.of Management and Informatics
SANNO University
1573 Kamikasuya, Isehara, Kanagawa 259–1197 Japan
Email: shioya@mi.sanno.ac.jp

Abstract

In data clustering, many approaches have been proposed such as K-means method and hierarchical method. One of the problems is that the results depend heavily on initial values and criterion to combine clusters. In this investigation, we propose a new method to cluster stream data while avoiding this deficiency. Here we assume there exists aspects of local regression in data. Then we develop our theory to combine clusters using $\mathcal{F}$ values by regression analysis as criterion and to adapt to stream data. We examine experiments and show how well the theory works.

Keywords: Data Mining, Data Stream, Clustering for Stream, Regression Analysis

1 Introduction

Cluster analysis comes from multivaliable analysis in statistics. Putting our stress on computation aspects, it is a general term of algorithms to collect similar objects into groups (clusters) where each object in one cluster shares heterogeneous feature. We can say that, in every research activity, a researcher is always faced to a problem how observed data should be systematically organized. Cluster analysis has been applied to a vast range of application domains. For instance, pattern recognition (generation of land use map in map processing), spacial data analysis, image processing, business analysis (new kinds of insurance generated from patterns of automobile accidents) and WWW (Web clustering and classification, extraction of usage pattern from weblog).

Among others, we see remarkable development of new application areas through high speed network and internet technology nowadays. Let us note that the more and more information appear along with time axis and that such information (called *data stream*) is inherently different from *temporal data*: patterns of changes with time in the latter are rather stable and homogeneous while the ones in the former are not. In the former case, the changes may have different properties or trends but with *locality*. That's why the cluster analysis technique can't be applied so easily. There should become more and more important to investigate clustering techniques against data stream.

Generally the higher similarity of objects in a cluster and the lower similarity between clusters we see, the better clustering we have. This means quality of clustering depends on definition of similarity and the calculation complexity. There is no guarantee to see whether we can interpret similarity easily or not. So is true for similarity from the view point of analysts. It is an analyst's responsibility to apply methods accurately to specific applications. The point is how to find out hidden patterns(Han & Kamber 2000).

The authors have already proposed a new method to clusters based on regression analysis by using variances and $\mathcal{F}$ statistic values of the clusters under the assumption that there exists aspects of local regression in data, i.e., observed data structure of local sub-linear space(Motoyoshi, Miura & Shioya 2003). In this investigation we assume a collection of data stream with local regression aspects and make clustering by considering adjusting weights to objects in streams.

Recently an interesting approach(Chakrabarti & Mehrotra 2000) has been proposed, called "Local Dimensionality Reduction". In this approach, data are assumed to have correlation locally same as our case. But the clustering technique is based on Principal Component Analysis (PCA) and they propose completely different algorithm from ours.

In the next section we discuss reasons why conventional approaches are not suitable to our situation. In section 3 we give some definitions and discuss about preliminary processing of data. Section 4 contains a method to combine clusters and the criterion based on local regression analysis. . Section 5 contains how to apply our approach to data stream. In section 6, we examine experimental results, and we conclude our work in section 7.

2 Clustering Data with Local Trends

In this investigation, we assume a collection of data where we see several trends within. Such kind of data could be regressed *locally* by using partial linear functions and the result forms an elliptic cluster in multidimensional space. Generally these clusters may cross each other.

The naive solution is to put clusters together by using nearest neighbor method found in hierarchical approach(Jain, Murty & Flynn 1999). However, when clusters cross, the result may not be the one that we expect, in fact, they will be divided at crossing. If clusters have different trends but they are close to each other, they could be combined. Generally any approach based on general Minkowski distance have the similar problem.

In *k-means* method, a collection of objects is represented by its center of gravity. Unfortunately there are several deficiencies in this method. As every textbook says, it is not suitable for non-convex clusters. The notion of center comes from a notion of variance, but if we look for points to improve linearity of the two clusters by reassigning objects, we can't always obtain suitable points. More serious is that we should decide the number of clusters in advance.

In these approaches, we face to a common problem, how to define similarity between clusters. In our case, we want to capture local aspects of sub linearity, thus new techniques should provide us with (1) similarity to classify sub linear space, and (2) convergence on suitable level (i.e., the number of clusters) which can be interpreted easily.

Regression analysis is one of techniques of multivariable analysis by which we can predict future phenomenon in form of mathematical functions under the assumption that a collection of objects have local linearity in advance. We introduce $\mathcal{F}$ value as a criterion of the similarity to combine clusters. We consider a cluster as a line, that is, our approach is *clustering by line* while K-means method is *clustering by point.* In this investigation, by examining $\mathcal{F}$ value (as similarity measure), we combine linear clusters in one by one manner, in fact, we take an approach of *restoring* to target clusters. For more detail, see (Motoyoshi et al. 2003).

3 Initializing Clusters

In this section, we describe our approach and examine the difference with hierarchical clustering such as agglomerative nesting.

Data is a set of objects consisting of several *variables.* We assume that all variables are inputs in a form of numeric given from surroundings and that there is no other external criteria for a classification. A *criterion variable* is an attribute which plays a role of criterion of regression analysis given by analysts. Others are called *explanatory variables.* As for categorical data, readers could think about quantification theory or dummy variables.

Data is described by a *matrix* $(X|Y)$ where each object appears as a row of the matrix while criterion/explanatory variables as columns. We denote explanatory variables and a criterion variable by $x_1, x_2, \ldots, x_m$ and y respectively, and the number of objects by n:

$$(X|Y) = \left(\begin{array}{ccc|c} x_{11} & \cdots & x_{1m} & y_1 \\ \vdots & \ddots & \vdots & \vdots \\ x_{k1} & \cdots & x_{km} & y_k \\ \vdots & \ddots & \vdots & \vdots \\ x_{n1} & \cdots & x_{nm} & y_n \end{array}\right) \qquad (1)$$

Note that X denotes explanatory variables, and Y criterion variable. Each variable is assumed to be normalized (called Z score) as follows:

$$\mu_{xi} = \sum_{k=1}^{n} x_{ik} = 0 \quad ; \quad i = 1, \ldots, m \qquad (2)$$

$$\mu_y = \sum_{k=1}^{n} y_k = 0 \qquad (3)$$

$$\sqrt{\frac{1}{n}\sum (x_{ki} - \mu_{xi})^2} = 1 \quad ; \quad i = 1, \ldots, m \qquad (4)$$

$$\sqrt{\frac{1}{n}\sum (y_k - \mu_y)^2} = 1 \qquad (5)$$

An *initial cluster* is given as a set of objects. Each object is exclusively contained in one of the initial clusters.

In an agglomerative nesting, each object constitute one cluster and similarity is defined as *distance* between objects. In our approach, on the other hand, we assume every (initial) cluster should have non-trivial variances because we deal with "data as a line". To obtain these initial clusters, we divide the objects into small groups. We make initial clusters dynamically by inner product (cosine) calculation which measures the difference of angle between two vectors as the following algorithm shows:

0. Let a input vector be $\vec{s_1}, \vec{s_2}, , , \vec{s_n}$.

1. Let the first input vector $\vec{s_1}$ be center of cluster C_1 and $\vec{s_1}$ be a member of C_1.

2. Calculate similarity between $\vec{s_k}$ and existing cluster $C_1 \ldots C_i$ by (6). If every similarities is below given threshold Θ, we generate a new cluster and let it be the center of the cluster. Otherwise, let it be a member of cluster which has the highest similarity. By using (7) calculate again a center of cluster to which members are added.

3. Repeat until all the assignment is completed.

4. Remove clusters which has no $\mathcal{F}$ value and less than $m + 2$ members.

where

$$cos(k, j) = \frac{\vec{s_k} \cdot \vec{c_j}}{|\vec{s_k}||\vec{c_j}|} \qquad (6)$$

$$\vec{c_j} = \frac{\sum S_k \in C_j^{\vec{s_k}}}{M_j} \qquad (7)$$

Note M_j means the number of members in C_j and m means the number of explanatory variables.

4 Combining Clusters

Now let us define similarities between clusters, and describe how the similarity criterion relates to combining. We define the similarity between clusters from two aspects. One aspect comes from a distance between clusters. The authors have already proposed a method of using Euclidean distance. Because this method calculates distance from only center of gravity, it disregards bias of cluster caused by correlation between variables.

Our basic idea is that every vector in a cluster i must play its role with other vectors in the cluster, in other words, with the effect of variances and covariances of the cluster. As distance measure we take *Mahalanobis* distance in which not only center but also variance is considered. We calculate the Maharanobis distance between cluster and the other cluster center of gravity respectively, and define the average of two distances as the distance between pair of cluster.

$$d^2(i, j) = \frac{(\mu_i - \mu_j)^T C_j^{-1} (\mu_i - \mu_j) + (\mu_j - \mu_i)^T C_i^{-1} (\mu_j - \mu_i)}{2} \qquad (8)$$

Note C is "variance and co-variance" matrix of each clusters, and C^{-1} is the inverse matrix of C. Then we define *non-similarity matrix* $d^2(i,j) (\in R^{n \times n})$. Clearly one of the candidate clusters to combine should have the smallest distance and we examine whether it is suitable or not in our case. The second aspect comes from $\mathcal{F}$ test. This test is targeted towards examining whether regression analysis is really useful or not. And we define our new similarity by means of $\mathcal{F}$ values of the regression to keep effectiveness.

Let us review very quickly $\mathcal{F}$ test and presumption by regression based on least square method in multiple regression analysis. Given clusters represented by data matrix, we define a model of multiple regression analysis which is corresponded to the clusters as follows:

$$y = b_1x_1 + b_2x_2 + \ldots + b_mx_m + e_i \tag{9}$$

An estimator of the least squares $\tilde{b}_i$ of b_i is given by

$$B = (\tilde{b}_1, \tilde{b}_2, \ldots, \tilde{b}_m) = (X^TX)^{-1}X^TY \tag{10}$$

This is called *regression coefficient*. Actually it is a standardised partial regression coefficient, because it is based on Z-score.

Let y be an observed value and Y be a predicted value based on the regression coefficient B. Then, for variation factor by regression, *sum of squares* S_R and *mean square* V_R are defined as

$$S_R = \sum_{k=1}^{n}(Y_k - \bar{Y})^2 \quad ; \quad V_R = \frac{S_R}{m} \tag{11}$$

For variation factor by residual, *sum of squares* S_E and *mean square* V_E are given as

$$S_E = \sum_{k=1}^{n}(y_k - Y_k)^2 \quad ; \quad V_E = \frac{S_E}{n-m-1} \tag{12}$$

Then we define $\mathcal{F}$ value F_0 by:

$$F_0 = \frac{V_R}{V_E} \tag{13}$$

It is well known that F_0 obeys $\mathcal{F}$ distribution where the first and second degrees of freedom are m and $n-m-1$ respectively.

Given clusters A and B where $|A| = a, |B| = b$, a data matrix of the combined cluster $A \cup B$ is described as follows.

$$(X|Y) = \left(\begin{array}{ccc|c} x_{A11} & \cdots & x_{A1m} & y_{A1} \\ \vdots & \ddots & \vdots & \vdots \\ x_{Aa1} & \cdots & x_{Aam} & y_{Aa} \\ x_{B11} & \cdots & x_{B1m} & y_{B1} \\ \vdots & \ddots & \vdots & \vdots \\ x_{Bb1} & \cdots & x_{Bbm} & y_{Bb} \end{array}\right) \tag{14}$$

$$(\in R^{n\times(m+1)})$$

where $n = a + b$. As previously mentioned, we can calculate regression and $\mathcal{F}$ by (10) and (13) respectively.

Let A, B be two clusters, F_A, F_B the two $\mathcal{F}$ values and F the $\mathcal{F}$ value after combining A and B. Then we have some interesting properties.

PROPERTY 1 If $F_A > F$, $F_B > F$ holds:

When F decreases, the *gradient* is significantly different. Thus we can say that the similarity between A and B is low and linearity of the cluster decreases. In the case of $F_A = F_B, F = 0$, both A and B have same number objects and coordinates and the regressions are orthogonal at center of gravity. □

PROPERTY 2 If $F_A \leq F$, $F_B \leq F$ holds:

When F increases, the gradient isn't significantly different and the similarity between A and B is high. Linearity of the cluster increases. When $F_A = F_B, F = 2 \times F_A$, we see A and B have same number of the objects and coordinates. □

PROPERTY 3 If $F_A \leq F$,$F_B > F$ holds, or if $F_B \leq F$,$F_A > F$ holds:

One of F_A, F_B increases while another decreases, when there exists big difference between the variances of A and B, or between F_A and F_B. We can't say anything about combining. □

By above considerations, it seems better to combine clusters if F is bigger than both F_A and F_B.

Non-similarity using Mahalanobis distance is one of the ways to prohibit from combining clusters that have the distance bigger than local ones. Since our algorithm proceeds based on a criterion using $\mathcal{F}$ values, the process continues to look for candidate clusters by decreasing distance criterion until the process satisfies our $\mathcal{F}$ value criterion. This means that we may have difficulties in a case of defective initial clusters, or in a case of no cluster to regress locally: the process might combine clusters that should not be combined.

To overcome such problem, we assume a threshold Δ to a distance. When A and B satisfy both criterion of $\mathcal{F}$ value and Δ, we combine the two clusters. By Δ we manage the internal variances of clusters to avoid combine *far* clusters.

Here is our algorithm `CFR` (Clustering using F-value by Regression analysis) as follows.

1. Standardize data.
2. Calculate initial clusters that satisfy Θ. Remove clusters which the number of members don't reach the number of explanatory variables.
3. Calculate center of gravity, variance, regression coefficient, $\mathcal{F}$ value to each cluster the distance between them.
4. Choose close clusters as candidates for combining. Standardize the pair. Calculate regression coefficient and $\mathcal{F}$ value again.
5. Combine the pair if $\mathcal{F}$ value of a combined cluster is bigger than $\mathcal{F}$ value of each cluster and if it satisfy Δ. Otherwise, go to step 4 to choose other candidates. If there is not candidate any more, then stop.
6. Calculate center of gravity to each cluster and distance between them again and go to step 4.

5 Clustering Stream Data

Now we are ready to develop our theory to process data stream into clusters.

Generally data stream is organized along with time axis and processed in a sequential manner. However, there are continuous changes of their trends and it is the point to extract useful patterns in data mining research(Han et al. 2000). Here we give several assumption (principle) and show how well our clustering technique can be applied to data stream under the assumption.

As described in section 1, data stream contains several properties with local trends which cause difficulties of clustering tasks. It is well-known that *recent* events are more affected to give decision compared to past events, and it doesn't seem reasonable to keep all the events. In our approach, we summarize past information continuously while keeping recent information for a while.

This is not really new approach and we assume an interval $[t_1, t_2]$ over time axis called *block* with the constant size $t_2 - t_1$. We obtain objects along each block u(called *process unit*) in a form of stream and we give a weight to each unit. As time goes, we reduce the weights exponentially for clustering. That is, once a weight w is given where $0.0 < w < 1.0$, every object in an i-th block has been considered as a weight w^i. Given a threshold δ, there corresponds to a number h of blocks such that $w^h \leq \delta$.

Assume that there are h process units $u_0, u_1, \cdots, u_{h-1}$ and that we like to add a new unit u_0. First of all, we remove all the objects in the oldest u_{h-1}. Then we adjust the weights of all other objects in $u_0, ..., u_{h-2}$ by multiplying by w. Finally we make clustering by the new u_0 and $u_1, ..., u_{h-1}$.

Since weights decrease exponentially, *one* object with a weight w^i can be considered as the one with a count w^i. Objects $e_0, .., e_n$ of the weights $w^0, ..., w^n$ have attribute values $a_0, .., a_n$ over an attribute A respectively ($w^i > 0.0, i = 0, .., n$). A weighted *expect value* $E[A]$ is defined as $p_0 \times a_0 + \cdots + p_n \times a_n$ where $p_i = w^i/W$ and $W = w^0 + .. + w^n$. More generally, given a function $f(X)$, let us define $E[f(A)]$ as $p_0 \times f(a_0) + ... + p_n \times f(a_n)$. Then we define a weighted variance $V[A]$ as $E[(A - E[A])^2]$ which is equal to $E[A^2] - E[A]^2$ as usual. Similarly let us define weighted co-variance $C[AB]$ of two variables A, B as $E[AB] - E[A]E[B]$. All the statistical values of non-similarity based on Mahalanobis $d^2(i, j), S_R, S_E, V_R, V_E$ and F can be extended to the weighted ones.

When every new clusters in u_0 can be combined with some cluster over $u_1, ..., u_{h-1}$, we adjust center of gravity, variance, regression coefficient and $\mathcal{F}$, but there is no need to change clustering. Otherwise, we see some change happens during these time intervals and we make clustering from scratch to $u_1, .., u_{h-1}$.

After we adjust clustering and their statistical values such as center of gravity, we simply remove all the objects in u_{h-1}. This is reasonable because we try to reflect objects as many as possible while keeping *quality* of clustering current (i.e., clustering result reflects recent trends). The weight to u_{h-1} is the smallest thus the objects have least effect to make clustering.

Here is our enhanced algorithm `ICFR` (Incremental Clustering using F-value by Regression analysis) as follows. The readers should regard any statistical values like distance and variance as the revised ones. The complexity depends on the one of matrix calculation and the number of repetition of re-clustering at worst but generally on the number of regression.

1. Calculate initial clusters to all the $u_1, .., u_{h-1}$.

2. Apply CFR to the initial clusters.

3. Collect new objects into a process unit u_0. Calculate initial clusters to new u_0.

4. Examine whether the result of 3 can be combined to the result of 2 or not. If not, we make clustering from scratch to all the initial clusters in $u_0, u_1, .., u_{h-1}$ using weighted CFR.

5. Remove u_{h-1}. Renumber $u_0, .., u_{h-2}$ as $u_1, .., u_{h-1}$.

6. Calculate new center of gravity to each cluster and distance between them and go to step 3.

6 Experiments

In this section, let us show some experiments to demonstrate the feasibility of our theory. We have `Weather Data` in Japan(Japan Weather Association 1998): on January, 1997 two meteorological observatory data of Wakkanai in Hokkaido (northern part of Japan) and Niigata in Honshu (middle part of Japan) measured in January of 1997. Each meteorological observatory contains 8736 records (and 17472 records in total), 720K bytes(Japan Weather Association 1998). To apply our method under the assumption that there are clusters to regress locally. We simply joined them.

Each data instance contains 22 attributes observed every hour. We utilize 8 items consisting of "day"(day), "hour"(hour), "pressure"(hPa), "sea-level pressure"(hPa), "air temperature"(C), "dew point"(C), "steam pressure"(hPa) and "relative humidity"(%) as candidates of variables among the 22 attributes. All of them are numerical without any missing value. We use "observation point number" additionally only for the purpose of evaluation. A table1 contains examples of the data.

We have standardized all variables in advance to analyze by our algorithm. We take "air temperature" as a criterion variable and other values as explanatory variables. We give $\Theta = 0.6$ to initialize clusters and give $\Delta = 1.0E+6$ to define distance between clusters. Also we give a weight $w = 0.8$, the threshold of weight $\delta = 0.25$, each block has one week duration and the number h of processing units is 7.

A table 2 shows the result at September 10 since January 1. At this time, re-clustering has happened 12 times among 47 repetitions.

Among 7 clusters, most objects are included in Cluster 2 and Cluster 6. Cluster 2 and 6 contain 14 and 7 "initial" clusters respectively. These reflect features of the observed points. In fact, Cluster 2 contains 653 Niigata objects among total 1176 Niigata objects (55.6%) while Cluster 6 contains 537 objects among 1176 Wakkanai objects (45.7%). That is, Cluster 2 reflects the features of Niigata and Cluster 6 Wakkanai.

For example, in a table3 about Centers of Gravity, we see "Dew Point", "Steam Pressure" and "Air Temperature" are higher and "Relative Humidity" is lower in Cluster 2 compared to other clusters. Thus this cluster contains observed objects in a region of south and on the Sea of Japan side where summer precipitation is little. Also "Pressure" is higher and "Dew Point" and "Steam Pressure" are lower in Cluster 6. Because "Month" is low, the objects were observed in the middle of summer. However "Air Temperature" is lower than we expected. We see, the objects in this cluster were observed in region of north and low altitude.

In case of Cluster 1, all of "Day" and "Month" values were characteristic. Since the center of "Month" is higher and "Day" is lower, the cluster talks about weather observed in first half in September that are common to both points(e.g., the change of the seasons) but doesn't correspond to location aspects. In fact, the cluster 1 has low "Air Temperature" and contains almost same number of objects of Niigata and Wakkanai points.

To compare our approach ICFR to weighted CFR, we examine a *naive* experiment. That is to say, whenever we obtain new objects, we have make all the objects of 7 blocks clustered without any incremental calculation. We calculate *recall* and *precision* factors of objects in each point for each cluster. For example, recall of Niigata point for A cluster is a ratio of Niigata objects included in A in all Niigata objects, and precision is a ratio of Niigata objects included in

Table 1: Weather Data

Point	Month	Day	Hour	Pressure	Sea-level Pressure	Dew Point	Air Temperature	...
604	1	1	1	1019.2	1020	2.4	5	...
604	1	1	2	1018.6	1019.4	2	5.2	...
604	1	1	3	1018.3	1019.1	1.8	5.4	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
401	12	31	24	1005.1	1006.5	-10.6	-1	...

Table 2: Final Clusters

	Variance	$\mathcal{F}$ value	Contained Clusters	Niigata	Wakkanai
Cluster1	6.71152	76433.5	5	102	152
Cluster2	4.50243	36137.4	14	653	315
Cluster3	1.67056	12109.1	3	119	47
Cluster4	2.96373	4404.97	1	48	48
Cluster5	0.12095	5528.73	1	29	0
Cluster6	0.44520	826016	7	0	537
Cluster7	0.09501	28841.3	3	216	77

A in all objects included in A. If the highest recall is a Niigata point of A cluster, we regard A as a Niigata cluster. Then we regard cluster whose recall of Wakkanai point is the highest of other clusters except A cluster a Wakkanai cluster. In figures 1 and 2, we show recall and precision factors at the two points.

The result shows ICFR's recall go down about 20% compared with CFR. As for precision, there is no meaningful difference between the two approaches. Thus, we can keep similar precision as compared with to make clustering from scratch by using our incremental approach. In CFR result is independent at each processing. But in ICFR we can guess that there is no big change in the cluster as long as re-clustering doesn't happen. In case of our experiments, re-clustering happened in the fourth processing on the average. Thus we can see, a big weather change happens every month. Figure 3 shows the total times for processing and we see our approach has been reduced to about 20% and get good results.

Let us summarize our experiment. We got 7 clusters. Especially, we have extracted regional features from cluster 2 and 6. It is evident by information on observation point in table 2 to see clustering suitably has classified objects very well. This fact means that the results in our experiment satisfy the initial condition. Processing time will improve further by efficient I/O processing.

Table 3: Center of Gravity for Clusters

	Clust1	Clust2	Clust6
Month	1.18569	0.47771	-0.2102
Day	-0.4501	-0.0566	-0.2631
Hour	0.08692	-0.0682	-0.0298
Pressure	-0.0020	0.00151	0.40699
Sea-LevelPressure	0.00152	-0.0075	0.46129
Dew Point	0.01840	0.17168	-1.1524
Steam Pressure	0.01640	0.13634	-1.1048
RelativeHumidity	0.55982	-0.2062	0.57358
Air Temperature	-0.2807	0.23217	-1.2369

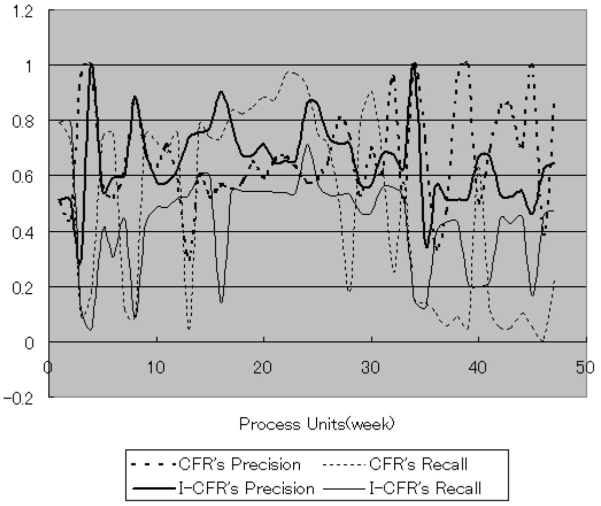

Figure 1: Recall/Precision Factors at Niigata

7 Conclusion

In this investigation, we have discussed a new approach of clustering for data stream which have several local trends among objects based on (Motoyoshi et al. 2003). We have proposed how to extract trends of clusters by using regression analysis and similarity of the cluster by $\mathcal{F}$ value of regression. We have introduced threshold of distance between clusters to keep precision of the cluster. By examining experimental data stream, we have shown that we can extract clusters of a moderate number to interpret and the features by center of gravity and regression coefficient. Then we have shown the feasibility of our approach.

We had already discussed how to mine Temporal Class Schemes to model a collection of time series data(Motoyoshi, Miura, Watanabe & Shioya 2002), and we are now developing further integrated methodologies to time series data and stream data.

Acknowledgements

We would like to acknowledge the financial support by Grant-in-Aid for Scientific Research (C)(2) (No.14580392).

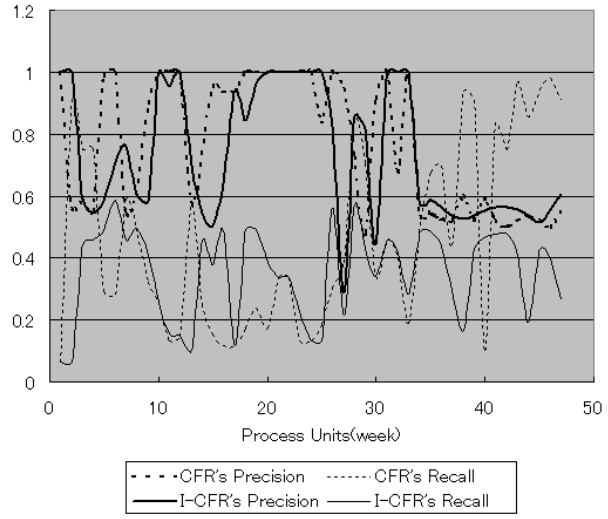

Figure 2: Recall/Precision Factors at Wakkanai

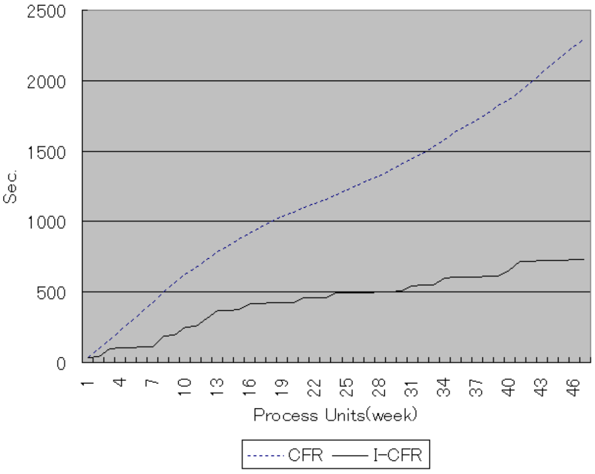

Figure 3: Total Processing Time

References

Chakrabarti, K., & Mehrotra, S. (2000), "Local Dimensionality Reduction: A New Approach to Indexing High Dimensional Spaces", proc. *VLDB*.

Han, J. & Kamber, M. (2000), "Data Mining - Concepts and Techniques", Morgan Kaufmann.

Jain, A. K., Murty, M. N. & Flynn, P. J. (1999), "Data Clutering – A Review", ACM Computing Surveys, Vol. 31–3, pp. 264–323.

Japan Weather Association (1998), "Weather Data HIMAWARI", Maruzen.

Motoyoshi,M., Miura,T., Watanabe,K. & Shioya,I. (2002), "Mining Temporal Classes from Time Series Data", proc.ACM *Conf. on Information and Knowledge Management* (CIKM), pp. 493–498.

Motoyoshi,M., Miura,T. & Shioya,I. (2003), "Clustering by Regression Analysis", proc. *Conf. on Data Warehousing and Knowledge Discovery* (DaWaK), pp. 202–211.