entropy

MDPI

Article

Mixture Complexity and Its Application to Gradual Clustering Change Detection

Shunki Kyoya * and Kenji Yamanishi

Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

* Correspondence: kyoya.shunki@plus-zero.co.jp

Abstract: We consider measuring the number of clusters (cluster size) in the finite mixture models for interpreting their structures. Many existing information criteria have been applied for this issue by regarding it as the same as the number of mixture components (mixture size); however, this may not be valid in the presence of overlaps or weight biases. In this study, we argue that the cluster size should be measured as a continuous value and propose a new criterion called mixture complexity (MC) to formulate it. It is formally defined from the viewpoint of information theory and can be seen as a natural extension of the cluster size considering overlap and weight bias. Subsequently, we apply MC to the issue of gradual clustering change detection. Conventionally, clustering changes have been regarded as abrupt, induced by the changes in the mixture size or cluster size. Meanwhile, we consider the clustering changes to be gradual in terms of MC; it has the benefits of finding the changes earlier and discerning the significant and insignificant changes. We further demonstrate that the MC can be decomposed according to the hierarchical structures of the mixture models; it helps us to analyze the detail of substructures.

Keywords: finite mixture model; clustering; change detection; gradual change; information theory

Citation: Kyoya, S.; Yamanishi, K. Mixture Complexity and Its Application to Gradual Clustering Change Detection. *Entropy* **2022**, *24*, 1407. https://doi.org/10.3390/e24101407

Academic Editor: Ciprian Doru Giurcaneanu

Received: 18 August 2022
Accepted: 28 September 2022
Published: 1 October 2022

1. Introduction

1.1. Motivation

Finite mixture models are widely used for model-based clustering (for overviews and references see McLachlan and Peel [1] and Fraley and Raftery [2]). In this field, determining the number of components is a typical issue. It refers to the following two aspects: the number of elements used to represent the density distribution and the number of clusters used to group the data (referred to as *mixture size* and *cluster size*, respectively). In this study, we consider the problem of interpreting the cluster size when the mixture size is given. Many existing information criteria have been applied for this issue by regarding it as the same as mixture size; however, it may not be valid when the components have overlaps or weight biases. Therefore, we need to reconsider the definitions and meanings of the cluster size.

For instance, let us observe three cases of the Gaussian mixture model, as shown in Figure 1. Although the mixture size is two in any case, the situations are different. In case (a), the two components are distinct from each other and their weights are not biased; therefore, it is sound to believe that the cluster size is two as well. Meanwhile, in case (b), although their weights are not biased, the two components are very close to each other; then, as proposed in the work of Hennig [3], we may need to regard them as one cluster by merging them. In case (c), although the two components are distinct from each other, their weights are biased; as proposed in Jiang et al. [4] and He et al. [5], we may need to regard the small components as outliers rather than a cluster. Overall, in cases (b) and (c), it may be more difficult to say that the cluster size is exactly two than in case (a). This observation gives rise to the problem of formally defining the complexity of clustering structures that reflects the overlaps and weight biases.

This paper introduces a novel concept of *mixture complexity* (MC) to resolve this problem. It is related to the logarithm of the cluster size. For example, the exponentials of the MC are 2.00, 1.39, and 1.21 for cases (a), (b), and (c), respectively. In other words, given the mixture size, MC estimates the cluster size continuously rather than discretely.

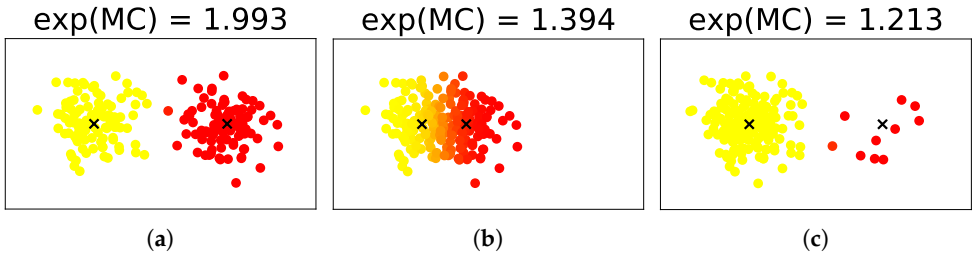


Figure 1. Examples of MC with Gaussian mixture models with a mixture size of two.

There are two reasons for the need of MC. First, it theoretically evaluates the cluster size in the finite mixture model considering the overlap and imbalance between the components. Although their impacts on the cluster size have been discussed independently, we present a unified framework to interpret the cluster size with a continuous index. It presents a new perspective on model-based clustering and can be practically applied to cluster merging or clustering-based outlier detection. The second is the application of MC to the issue of gradual clustering change detection. Conventionally, clustering changes have been considered to be abrupt, induced by changes in the mixture size or cluster size. In reality, however, there are cases where mechanisms for generating data change gradually (or incrementally in the context of concept drifts [6]). We thereby present a new methodology for tracking such changes by observing MC's changes.

We further show that MC can be used to quantify the cluster size in hierarchical mixture models. We demonstrate that the MC of a hierarchical mixture model can be decomposed into the sum of MCs for local mixture models. It enables us to evaluate the complexity of the substructures as well as the entire structure.

The concept of MC has been applied to the clustering merging problem in [7]. This study further investigates the theoretical properties of MC and proposes a new application for the issue of gradual clustering change detection.

1.2. Significance and Novelty

The significance and novelty of this paper are summarized below.

1.2.1. Mixture Complexity for Finite Mixture Models

We introduce a novel concept of MC to continuously measure the cluster size in a mixture model. It is formally defined from the viewpoint of information theory and can be interpreted as a natural extension of the cluster size considering the overlaps and weight biases among the components. We further demonstrate that MC can be decomposed into a sum of MCs according to the mixture hierarchies; it helps us in analyzing MC in a decomposed manner.

1.2.2. Applications of MC to Gradual Clustering Change Detection

We apply MC to the issue of monitoring gradual changes in clustering structures. We propose methods to monitor changes in MC instead of the mixture size or cluster size. Because MC takes a real value, it is more suitable for observing gradual changes. We empirically demonstrate that MC elucidates the clustering structures and their changes more effectively than the mixture size or cluster size.

The remainder of this paper is organized as follows. Section 2 discusses related work. In Section 3, we introduce the concept of MC and present some examples. Theoretical properties of MC are shown in Section 4. Section 5 discusses the application of

MC to clustering change detection problems and Section 6 describes the experimental results. Finally, Section 7 concludes this paper. Proofs of the propositions and theorems are described in Appendices. Programs for the experiments are available at https://github.com/ShunkiKyoya/MixtureComplexity, accessed on 17 August 2022.

2. Related Work

The issue of determining the best mixture size or cluster size (often referred to as model selection) has extensively been studied. For example, AIC [8], BIC [9], and MDL [10] have been used to select the mixture size; ICL [11] and MDL-based clustering criteria [12,13] have been invented to select the cluster size. These methods have conventionally considered the cluster size as the same as the mixture size by regarding one mixture component as one independent cluster. See also a recent review by McLachlan and Rathnayake [14] focusing on the number of components in a Gaussian mixture model.

Differences between the mixture size and cluster size have also been widely discussed. For example, McLachlan and Peel [1] pointed out that there were cases that Gaussian mixture models with more than one mixture sizes were needed to describe one skewed cluster; Biernacki et al. [11] argued that in many situations, the mixture size estimated by BIC was too large to regard it as the cluster size. The problem of estimating the cluster size under a given mixture size has also been investigated by Hennig [3]; he proposed methods to identify the cluster structure by merging heavily overlapped mixture components. MC differs from his approach in that it interprets the clustering structure by only measuring the overlap rate rather than deciding whether to merge based on a certain threshold.

The degree of overlap or closeness between components was evaluated using various measures, such as the classification error rate or the Bhattacharyya distance [15]. Wang and Sun [16] and Sun and Wang [17] formulated the overlap rate of Gaussian distributions from the geometric nature of them. All of the works above have been limited to the case of two components. On the other hand, MC considers the overlap between any number of components.

Deciding whether a small component is a cluster or a set of outliers is also a significant matter. For example, clustering algorithms such as DBSCAN [18] and constrained k-means [19] avoided generating small components to obtain a better clustering structure. Jiang et al. [4] and He et al. [5] associated the small components with outlier detection problems. MC evaluates the small components by continuously measuring the impacts on the cluster size.

Some other notions have been proposed to quantify the clustering structure. Fuzzy clustering [20] is also a method used to estimate the clustering structures with cluster overlap; however, MC is more suitable for consistent estimation in that it assumes the background mixture distributions. Rusch et al. [21] evaluated the crowdedness of the data under the concept of "clusteredness". However, its relations to the cluster size are indirect. Recently, descriptive dimensionality (Ddim) [22] was proposed to define the model dimensionality continuously. It can be implemented to estimate the clustering structure under the assumption of model fusion, that is, models with a different number of components are probabilistically mixed. MC differs from Ddim because it evaluates the overlap and weight bias in the single model without model fusion.

Clustering under the data stream has been discussed with various objectives [23–25]. We consider the problem of detecting changes in the cluster structure; Dynamic model selection (DMS) [26–28] addressed this problem by observing the changes in the models (corresponding to mixture size or cluster size in this paper). Because the models are valued discretely, the detected changes have been considered to be abrupt. Refer also to the notions of tracking best experts [29], evolution graph [30], and switching distributions [31], which are similar to DMS.

Furthermore, the issues of gradual changes have been discussed to investigate the transition periods for absolute changes. The MDL change statistics [32] and differential MDL change statistics [33] were proposed to measure the degree of gradual changes. The

notions of structural entropy [34] and graph entropy [35] were proposed to measure the degree of model uncertainty in the changes. This study quantifies the degree of gradual changes using the fluctuations in MC and presents a new methodology to detect them.

MC is based on the mutual information between the observed and latent variables, which has been considered in the clustering fields. For example, Still et al. [36] regarded clustering as data compression and applied mutual information to measure its degree. In this paper, we present a novel interpretation of mutual information as a continuous number of clusters. Furthermore, we also present its novel applications for interpreting clusterings and clustering change detection.

3. Mixture Complexity

In this section, we formally introduce the mixture complexity and describe its properties using some examples and theories.

3.1. Definitions

Given the data $\{x_n\}_{n=1}^N$ and the finite mixture model f that have generated them, we consider interpreting the cluster size of f. The distribution f is written as

$$f(x) := \sum_{k=1}^{K} \rho_k g_k(x),$$

where K denotes the mixture size, $\{\rho_k\}_{k=1}^K$ denote the proportions of each component summing up to one, and $\{g_k\}_{k=1}^K$ denotes the probability distributions. The random variable X following the distribution f is called an *observed variable* because it can be observed as a datum. We also define the *latent variable* $Z \in \{1, \ldots, K\}$ as the index of the component from which the observed variable X originated. The pair (X, Z) is called a *complete variable*. The distribution of the latent variable $P(Z)$ and the conditional distribution of the observed variable $P(X|Z)$ can be given by

$$\begin{aligned} P(Z = k) &= \rho_k, \\ P(X|Z = k) &= g_k(X). \end{aligned}$$

To investigate the clustering structures in f, we consider the following quantity:

$$I(Z; X) := H(Z) - H(Z|X),$$

where $H(Z)$ and $H(Z|X)$ denote the entropy and conditional entropy, respectively, of the latent variable Z defined as

$$\begin{aligned} H(Z) &:= -\sum_{k=1}^{K} P(Z = k) \log P(Z = k) = -\sum_{k=1}^{K} \rho_k \log \rho_k, \\ H(Z|X) &:= -E_X\left[\sum_{k=1}^{K} P(Z = k|X) \log P(Z = k|X)\right] = -E_X\left[\sum_{k=1}^{K} \gamma_k(X) \log \gamma_k(X)\right]. \end{aligned}$$

where

$$\gamma_k(X) := P(Z = k|X).$$

The quantity $I(Z; X)$ is well-known as the mutual information between the observed and latent variables; it is also known as the (generalized) Jensen–Shannon Divergence [37]. We can interpret $I(Z; X)$ as the volume of cluster structures as follows. Because $I(Z; X)$ is a subtraction of the latent variable's entropy with and without the knowledge of the observed variable, it represents the amount of information about the latent variable possessed by the observed data. Thus, its exponent $\exp(I(X; Z))$ denotes the number of the latent variables distinguished by the observed variable; it can be interpreted as a continuous extension of

the cluster size. For more information about entropy and mutual information, see the book written by Cover and Thomas [38].

However, $I(Z;X)$ cannot be calculated analytically even if f is known. Thus, noting that $\rho_k = E_X[\gamma_k(X)]$, we approximate $I(Z;X)$ using the data $\{x_n\}_{n=1}^N$ as follows:

$$I(Z;X) \approx \widetilde{H}(Z) - \widetilde{H}(Z|X),$$

where

$$\widetilde{H}(Z) := -\sum_{k=1}^{K} \widetilde{\rho}_k \log \widetilde{\rho}_k,$$
$$\widetilde{H}(Z|X) := -\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K} \gamma_k(x_n) \log \gamma_k(x_n),$$
$$\widetilde{\rho}_k := \frac{1}{N}\sum_{n=1}^{N} \gamma_k(x_n).$$

We call this the MC of the mixture model f.

Definition 1. *Given the posterior probabilities $\{\gamma_k(x_n)\}_{k,n}$, we define the mixture complexity (MC) as*

$$\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}\Big) := -\sum_{k=1}^{K} \widetilde{\rho}_k \log \widetilde{\rho}_k + \frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K} \gamma_k(x_n) \log \gamma_k(x_n),$$

where

$$\widetilde{\rho}_k := \frac{1}{N}\sum_{n=1}^{N} \gamma_k(x_n).$$

If the data have weights $\{w_n\}_n$, we define the MC as

$$\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}; \{w_n\}_n\Big) := -\sum_{k=1}^{K} \widetilde{\rho}_k \log \widetilde{\rho}_k + \frac{1}{\sum_{n'} w_{n'}}\sum_{n=1}^{N} w_n \sum_{k=1}^{K} \gamma_k(x_n) \log \gamma_k(x_n),$$

where

$$\widetilde{\rho}_k := \frac{1}{\sum_{n'} w_{n'}}\sum_{n=1}^{N} w_n \gamma_k(x_n).$$

The weighted version of MC is defined for later use.

Note that there are other ways to approximate $I(Z;X)$; we adopt the form of Definition 1 because it has the decomposition property shown in Section 4.2. See also the methods used to approximate the entropy of the mixture model [39,40] that can also be applied to approximate $I(Z;X)$.

In practice, only the data $\{x_n\}_{n=1}^N$ can be obtained without the underlying distribution f. Then, we estimate the posterior probabilities $\{\widehat{\gamma}_k(x_n)\}_{k,n}$ from the data $\{x_n\}_{n=1}^N$ and further estimate the MC as

$$\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}\Big) \approx \mathrm{MC}\Big(\{\widehat{\gamma}_k(x_n)\}_{k,n}\Big).$$

It can be calculated even if the model f cannot be estimated.

3.2. Examples

In this subsection, we discuss some examples of MC to understand its notions.

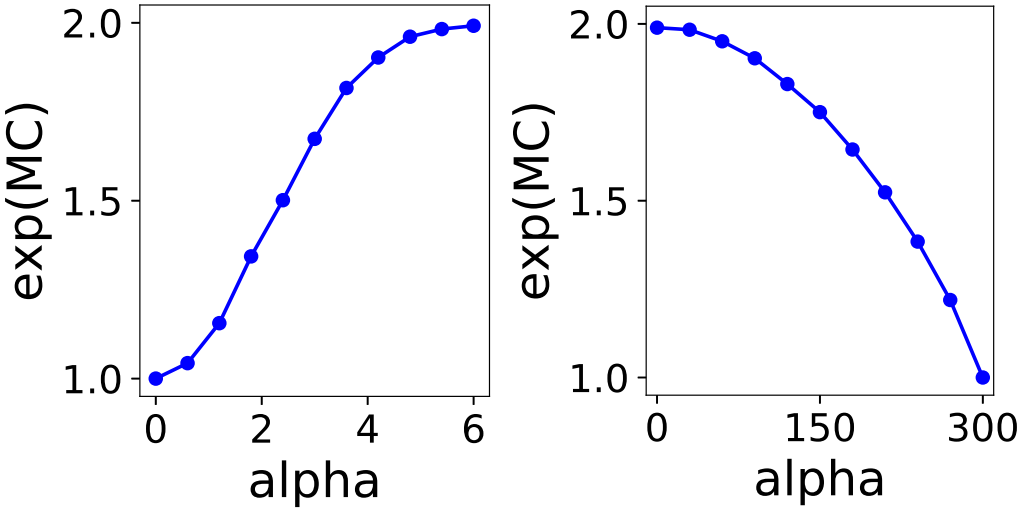
3.2.1. MC with Different Overlaps

First, we set $N = 600$ and generated the data $x_1, \ldots, x_{600} \in \mathbb{R}^2$ as follows.

$$x_n \sim \begin{cases} \mathcal{N}(x_n | \mu = [0,0]^\top, \Sigma = I_2) & (1 \leq n \leq 300), \\ \mathcal{N}(x_n | \mu = [\alpha,0]^\top, \Sigma = I_2) & (301 \leq n \leq 600), \end{cases}$$

where $\mathcal{N}(x|\mu, \Sigma)$ denotes a multivariate normal distribution with mean μ and covariance Σ, I_d denotes a d-dimensional identity matrix, and $\alpha \in \mathbb{R}$ is the parameter that determines the degree of overlap between two components.

By varying the value of α among $0, 0.6, \ldots, 6.0$, we generated the data and measured the MC by setting $\rho_1, \rho_2 = 1/2$ and g_1, g_2 as the actual distributions. The exponential of the MC for each α is plotted in Figure 2a. It is evident from the figure that the MC smoothly increases from 1.0 to 2.0 as the two components become isolated.



(**a**) MC with different overlaps (**b**) MC with different mixture biases

Figure 2. Relation between the parameter α and the exponential of the MC.

3.2.2. MC with Different Mixture Biases

Next, we set $N = 600$ and generated the data $x_1, \ldots, x_{600} \in \mathbb{R}^2$ as follows:

$$x_n \sim \begin{cases} \mathcal{N}(x_n | \mu = [0,0]^\top, \Sigma = I_2) & (1 \leq n \leq 300 + \alpha), \\ \mathcal{N}(x_n | \mu = [6,0]^\top, \Sigma = I_2) & (301 + \alpha \leq n \leq 600), \end{cases}$$

where $\alpha \in \{0, \ldots, 300\}$ is the parameter that determines the degree of bias between the proportion of two components.

By varying α among $0, 30, \ldots, 300$, we generated the data and measured the MC by setting $\rho_1 = (300 + \alpha)/600, \rho_2 = (300 - \alpha)/600$ and g_1, g_2 as the actual distributions. The exponential of the MC for each α is plotted in Figure 2b. It is evident from the figure that the MC smoothly decreases from 2.0 to 1.0 as the balance becomes biased.

4. Theoretical Properties

In this subsection, we discuss the theoretical properties of MC.

4.1. Basic Properties

We discuss the basic properties of MC. The proofs are described in Appendix A.

First, we discuss the minimum and maximum of MC. We show that MC takes the minimum when the components entirely overlap and maximum when they are entirely separate.

Proposition 1. *If the components entirely overlap, i.e., there exists* $\gamma_1, \ldots, \gamma_K$ *such that* $\gamma_k(x_n) = \gamma_k$ *for all* k *and* n*, then,*

$$\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}; \{w_n\}_n\Big) = 0.$$

Proposition 2. *If the components are entirely separate, i.e., for all* x_n*, there is a unique index* k_n *that satisfies*

$$\gamma_l(x_n) = \begin{cases} 1 & (l = k_n) \\ 0 & (l \neq k_n) \end{cases},$$

then,

$$\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}; \{w_n\}_n\Big) = \widetilde{H}(Z).$$

In particular, if the components are entirely balanced, i.e., $\widetilde{\rho}_1 = \cdots = \widetilde{\rho}_K = 1/K$*, then,*

$$\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}; \{w_n\}_n\Big) = \log K.$$

Proposition 3. *For all* $\{\gamma_k(x_n)\}_{k,n}$*, MC satisfies*

$$0 \leq \mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}; \{w_n\}_n\Big) \leq \log K.$$

Moreover, MC takes 0 only if the components are entirely overlapping as stated in Proposition 1 and takes $\log K$ *only if the components are entirely separate as stated in Proposition 2.*

Next, we show that the value of MC is invariant with the representation of the mixture distribution. For example, consider the following three mixture distributions:

$$\begin{aligned} f_1(x) &= \frac{1}{2}g_1(x) + \frac{1}{2}g_2(x), \\ f_2(x) &= \frac{1}{2}g_1(x) + \frac{1}{4}g_2(x) + \frac{1}{4}g_2(x), \\ f_3(x) &= \frac{1}{2}g_1(x) + \frac{1}{4}g_2(x) + \frac{1}{4}g_2(x) + 0 \cdot g_3(x). \end{aligned}$$

In f_2 and f_3, we need to manually remove the redundant components and regard the mixture size as two [1]. On the other hand, the following property indicates that the MCs for f_1, f_2, and f_3 are the same; thus, we need not to care about their differences in evaluating MC.

Proposition 4. *If there exists a set* $I_1, \ldots, I_L, I_\infty$ *that partitions* $\{1, \ldots, K\}$ *and distributions* $g_1^0, \ldots, g_L^0$ *such that*

$$\begin{gathered} k \in I_l \Rightarrow g_k = g_l^0 \quad (l = 1, \ldots, L), \\ k \in I_\infty \Rightarrow \rho_k = 0, \end{gathered}$$

then

$$\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}; \{w_n\}_n\Big) = \mathrm{MC}\Big(\Big\{\gamma_l^0(x_n)\Big\}_{l,n}; \{w_n\}_n\Big),$$

where

$$\gamma_l^0(x) = \frac{\Big(\sum_{k \in I_l} \rho_k\Big) g_l^0(x)}{f^0(x)}, \quad f^0(x) = \sum_{l=1}^{L} \Bigg(\sum_{k \in I_l} \rho_k\Bigg) g_l^0(x).$$

4.2. Decomposition Property

In this section, we discuss a method to decompose MC along the hierarchies in mixture models; this can help us in analyzing the structures in more detail.

Consider that the mixture distribution f has a two-stage hierarchy, as shown in Figure 3. It has K components $\{g_k\}_{k=1}^K$ on the lower side and L components $\{h_l\}_{l=1}^L$ on the upper side, where $\{g_k\}_{k=1}^K$ denote the probability distributions and $\{h_l\}_{l=1}^L$ denote their mixture distributions, respectively. We construct the hierarchy as follows. First, we estimate the distribution $f = \sum_{k=1}^K \rho_k g_k$. Then, we obtain $\{h_l\}_{l=1}^L$ by partitioning (or clustering) the lower components into L groups. Formally, we denote $Q_k^{(l)} \in \mathbb{R}_{\geq 0}$ as the proportion of the lower component $k \in \{1, \ldots, K\}$ that belongs to the upper component l, which satisfies $\sum_{l=1}^L Q_k^{(l)} = 1$ for all k. Then, we derive $\{h_l\}_{l=1}^L$ by rewriting $f = \sum_{k=1}^K \rho_k g_k$ as

$$f(x) = \sum_{k=1}^K \rho_k g_k(x) = \sum_{k=1}^K \sum_{l=1}^L Q_k^{(l)} \rho_k g_k(x) = \sum_{l=1}^L \tau_l h_l(x),$$

where

$$\tau_l := \sum_{k=1}^K Q_k^{(l)} \rho_k, \quad h_l(x) := \sum_{k=1}^K \rho_k^{(l)} g_k(x), \quad \rho_k^{(l)} := \frac{Q_k^{(l)} \rho_k}{\sum_{k'} Q_{k'}^{(l)} \rho_{k'}}.$$

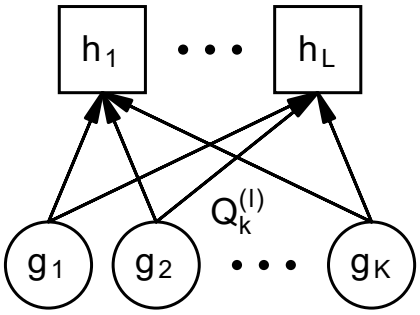


Figure 3. Hierarchy in the mixture model.

According to the hierarchy, we can decompose the MC.

Theorem 1. *We can decompose the MC as follows:*

$$\begin{aligned}
&\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}; \{w_n\}_n\Big) \\
&= \mathrm{MC}\left(\left\{\sum_{k=1}^K Q_k^{(l)} \gamma_k(x_n)\right\}_{l,n}; \{w_n\}_n\right) + \sum_{l=1}^L W_l \cdot \mathrm{MC}\left(\left\{\gamma_k^{(l)}(x_n)\right\}_{k,n}; \left\{w_n^{(l)}\right\}_n\right),
\end{aligned}$$

where

$$\begin{aligned}
W_l &= \frac{\sum_n w_n \sum_k Q_k^{(l)} \gamma_k(x_n)}{\sum_{n'} w_{n'}} = \sum_{k=1}^K Q_k^{(l)} \widetilde{\rho}_k, \\
w_n^{(l)} &= w_n \sum_{k=1}^K Q_k^{(l)} \gamma_k(x_n), \\
\gamma_k^{(l)}(x_n) &= \frac{Q_k^{(l)} \gamma_k(x_n)}{\sum_{k'} Q_{k'}^{(l)} \gamma_{k'}(x_n)}.
\end{aligned}$$

The proof is described in the Appendix B. For notational simplicity, we will use the following terms:

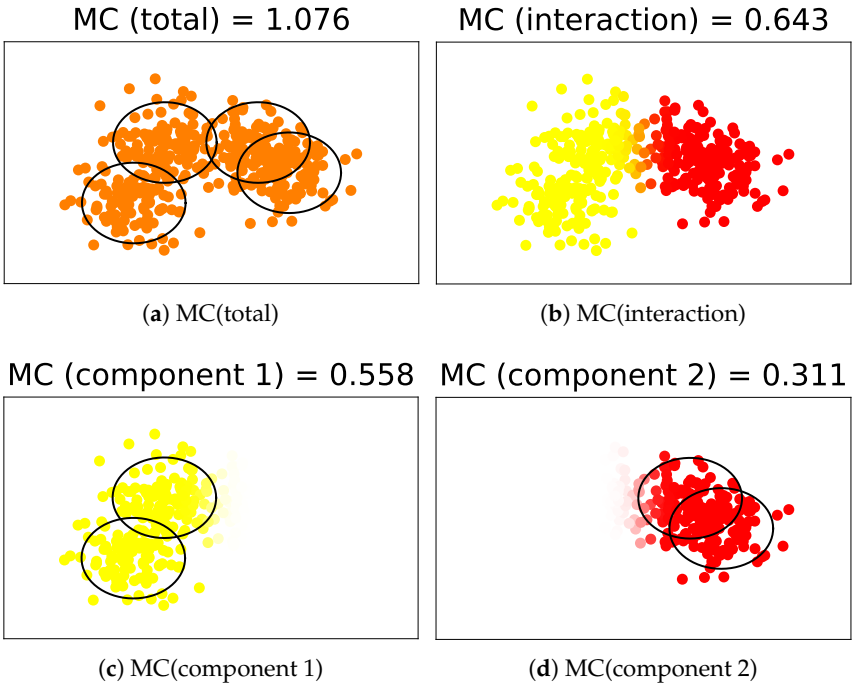
$$\begin{aligned}
\mathrm{MC}(\text{total}) &:= \mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}; \{w_n\}\Big),\\
\mathrm{MC}(\text{interaction}) &:= \mathrm{MC}\left(\left\{\sum_k Q_k^{(l)} \gamma_k(x_n)\right\}_{l,n}; \{w_n\}_n\right),\\
\mathrm{Contribution}(\text{component } l) &:= W_l \cdot \mathrm{MC}\left(\left\{\gamma_k^{(l)}(x_n)\right\}_{k,n}; \left\{w_n^{(l)}\right\}_n\right),\\
W(\text{component } l) &:= W_l,\\
\mathrm{MC}(\text{component } l) &:= \mathrm{MC}\left(\left\{\gamma_k^{(l)}(x_n)\right\}_{k,n}; \left\{w_n^{(l)}\right\}_n\right).
\end{aligned}$$

Then, we can rewrite Theorem 1 as

$$\begin{aligned}
\mathrm{MC}(\text{total}) &= \mathrm{MC}(\text{interaction}) + \sum_{l=1}^{L} \mathrm{Contribution}(\text{component } l),\\
\mathrm{Contribution}(\text{component } l) &= \mathrm{W}(\text{component } l) \cdot \mathrm{MC}(\text{component } l).
\end{aligned}$$

In Theorem 1, the MC of the entire structure (MC(total)) is decomposed into a sum of the MC among the upper components (MC(interaction)) and their respective contributions (Contribution(component l)). Contribution(component l) is further decomposed into a product of the weight (W(component l)) and complexity (MC(component l)) of the component. Because $w_n^{(l)}$ denotes the weight of x_n that belongs to component l, its sum W(component l) represents the total weights of the data contained in it. Additionally, MC(component l) denotes the clustering structures in component l considering the data weights.

An example of the decomposition is illustrated in Figure 4 and Table 1. In this example, there are $K = 4$ lower components generated from a Gaussian mixture model; additionally, there are $L = 2$ upper components on the left and right sides. By decomposing MC(total), we can evaluate the complexities in the local structures as well as those in the entire structure.



(**a**) MC(total) (**b**) MC(interaction)

(**c**) MC(component 1) (**d**) MC(component 2)

Figure 4. Example of the decomposition of MC. The data's color in (**b**) and thickness in (**c**,**d**) correspond to the data weights $w_n^{(l)}$.

Table 1. Quantities in the example of the decomposition.

	Component 1	**Component 2**
MC (total)	1.076	
MC (interaction)	0.643	
Contribution (component l)	0.277	0.157
W (component l)	0.496	0.504
MC (component l)	0.558	0.311

4.3. Consistency

In this subsection, we discuss the consistency of the MC: as the estimated distribution becomes close to the true distribution, the estimated MC also converges to the true value. Formally, we define the set of K-component mixture models as

$$\mathcal{F}_K := \left\{ f(x) = \sum_{k=1}^{K} \rho_k g(x|\theta_k) \,\middle|\, \rho_1, \ldots, \rho_K \geq 0, \quad \sum_{k=1}^{K} \rho_k = 1, \quad \theta_1, \ldots, \theta_K \in \Theta \right\}.$$

We assume that the space $\mathcal{F}_K$ is weakly identifiable, that is,

$$\sum_{k=1}^{K} \rho_k^0 g\left(\cdot|\theta_k^0\right) = \sum_{k=1}^{K} \rho_k^1 g\left(\cdot|\theta_k^1\right) \quad \Leftrightarrow \quad \sum_{k=1}^{K} \rho_k^0 \delta_\Theta(\cdot = \theta_k^0) = \sum_{k=1}^{K} \rho_k^1 \delta_\Theta(\cdot = \theta_k^1),$$

where δ_Θ is the Kronecker's delta function on Θ. This condition states that the same distributions should have the same mixtures of parameters. See Teicher [41] and Yakiwitz and Spragins [42] for sufficient conditions on this kind of identifiability; in their work, it has been shown that this is satisfied in Gaussian or gamma mixtures.

We also assume some true mixture distribution written as

$$f^\star(x) = \sum_{k=1}^{K^\star} \rho_k^\star g(x|\theta_k^\star), \quad \rho_1^\star, \ldots, \rho_{K^\star}^\star > 0, \quad \theta_1^\star, \ldots, \theta_{K^\star}^\star \in \Theta, \quad \theta_i^\star \neq \theta_j^\star \ (i \neq j)$$

generates the data x^N. We consider estimating the true mixture complexity written as $\mathrm{MC}(\{\gamma_k^\star(x_n)\}_{k,n})$ by substituting the estimated distribution $f \in \mathcal{F}_K$ into $f^\star$. We restrict our analysis to the case that $K \geq K^\star$ so that $\mathcal{F}_K$ contains distributions that are equivalent to $f^\star$. Then, we show that $\mathrm{MC}(\{\gamma_k(x_n)\}_{k,n})$ converges to $\mathrm{MC}(\{\gamma_k^\star(x_n)\}_{k,n})$ as f and $f^\star$ become closer.

To analyze the convergence, we re-parametrize the estimated parameters using the method proposed in Liu and Shao [43]. They note that if $f = f^\star$, there exist integers $0 = i_0 < \cdots < i_{K^\star} \leq K$ such that the following holds under some permutation of the components:

$$\begin{cases} \theta_l = \theta_k^\star & (l \in I_k, k = 1, \ldots, K^\star), \\ \rho_k = 0 & (k \in I_\infty), \end{cases}$$

where

$$I_k := \{i_{k-1} + 1, \ldots, i_k\} \quad (k = 1, \ldots, K^\star),$$
$$I_\infty := \{i_{K^\star} + 1, \ldots, K\}.$$

Then, they parametrize the parameters in f using two kinds of parameters defined as

$$\phi := \left(\{\theta_k\}_{k=1}^{i_{K^\star}}, \{r_l\}_{l=1}^{K^\star}, \{\rho_k\}_{k=i_{K^\star}+1}^{K} \right),$$
$$r_l := \sum_{k \in I_l} \rho_k,$$
$$\psi := \left(\{s_k\}_{k=1}^{i_{K^\star}}, \{\theta_k\}_{k=i_{K^\star}+1}^{K} \right),$$
$$s_k := \frac{\rho_k}{r_k} \ (k \in I_l)$$

and rewrite f as

$$f(x) = \sum_{k=1}^{K} \rho_k g(x|\theta_k) = \sum_{l=1}^{K^\star} r_l h_l(x) + \sum_{k=i_{K^\star}+1}^{K} \rho_k g_k(x),$$
$$h_l(x) = \sum_{k \in I_l} s_k g_k(x).$$

In this parametrization, $f = f^\star$ is equivalent to

$$\phi = \phi^\star := (\{\theta_1^\star, \ldots, \theta_1^\star, \ldots, \theta_{K^\star}^\star, \ldots, \theta_{K^\star}^\star\}, \{\rho_1^\star, \ldots, \rho_{K^\star}^\star\}, \{0, \ldots, 0\});$$

the parameter ψ has nothing to do with equivalence. This parametrization represents two types of convergence in mixture models. First, it overlaps the components to the true distributions, which is realized by

$$\{\theta_k\}_{k=1}^{i_{K^\star}} \to \{\theta_1^\star, \ldots, \theta_1^\star, \ldots, \theta_{K^\star}^\star, \ldots, \theta_{K^\star}^\star\},$$
$$\{r_l\}_{l=1}^{K^\star} \to \{\rho_1^\star, \ldots, \rho_{K^\star}^\star\}.$$

The other is shrinking the weights of the redundant components to zero, which is realized by

$$\{\rho_k\}_{k=i_{K^\star}+1}^{K} \to \{0, \ldots, 0\}.$$

We use the following conditions for our proof:

(C1) $g(\cdot|\theta)$ is differentiable once and for every $k = 1, \ldots, K^\star$ and there exists $\epsilon > 0$ such that

$$E_{\theta_k^\star}\left[\sup_{\theta : \|\theta - \theta_k^\star\| \le \epsilon} \left\| \frac{\nabla g(\cdot|\theta)}{g(\cdot|\theta)} \right\|\right] < \infty.$$

(C2) As $N \to \infty$, the estimated parameter ϕ satisfies

$$\|\phi - \phi^\star\| = o_P(1).$$

(C3) Let us define the approximations of mixture proportions as

$$\widetilde{r}_l := \frac{1}{N} \sum_{n=1}^{N} \sum_{k \in I_l} \gamma_k(x_n) \quad (l = 1, \ldots, K^\star),$$
$$\widetilde{\rho}_\infty := \frac{1}{N} \sum_{n=1}^{N} \sum_{k=i_{K^\star}+1}^{K} \gamma_k(x_n).$$

Then, as $N \to \infty$, they satisfy

$$|\widetilde{r}_l - \rho_l^\star| = o_P(1) \quad (l = 1, \ldots, K^\star),$$
$$\widetilde{\rho}_\infty = o_P(1).$$

Condition (C1) is a usual differentiability condition, and (C2) and (C3) require consistency of the parameters. It is known that consistent estimations are possible by penalized maximum likelihood estimation [44,45] or Bayesian estimation [46], for example. Then, the consistency of the MC is shown as the following theorem.

Theorem 2. *Under assumptions (C1), (C2), and (C3), the following holds as $N \to \infty$:*

$$\left|\mathrm{MC}\left(\{\gamma_k(x_n)\}_{k,n}\right) - \mathrm{MC}\left(\{\gamma_k^\star(x_n)\}_{k,n}\right)\right|$$
$$= O_P\left(\|\phi - \phi^\star\| + \sum_{l=1}^{K^\star} |\widetilde{r}_l - \rho_l^\star| + \widetilde{\rho}_\infty \log(-\widetilde{\rho}_\infty) + \frac{1}{\sqrt{N}}\right).$$

The proof is described in Appendix C. Theorem 2 shows the convergence rate of the estimation error of the MC. It is interesting that this even holds when $K \neq K^\star$. Therefore, it can be said that MC is a fundamental quantity to represent the cluster structures in mixture models by overcoming the differences in mixture size.

We discuss the overview of the proofs below. First, applying Theorem 1 repeatedly, we decompose the entire MC into the following four terms:

(a) Interaction between $\sum_{l=1}^{K^\star} r_l h_l$ and $\sum_{k=i_{K^\star}+1}^{K} \rho_l g_l$.
(b) Contribution from $\sum_{k=i_{K^\star}+1}^{K} \rho_l g_l$.
(c) Interaction among $h_1, \ldots, h_{K^\star}$
(d) Contributions from $h_1, \ldots, h_{K^\star}$, respectively.

The procedure of the decomposition is also illustrated in Figure 5. Then, we show that

(a) tends to 0 because $\widetilde{\rho}_\infty \to 0$;
(b) tends to 0 because $\widetilde{\rho}_\infty \to 0$;
(c) tends to $\mathrm{MC}(\{\gamma_k^\star(x_n)\}_{k,n})$ because $h_1, \ldots, h_{K^\star}$ tends to $g_1, \ldots, g_{K^\star}$;
(d) tends to 0 because for all l, all components in h_l tends to g_l.

The proofs are mainly based on the mean-value theorem. However, differentiation of $\log f$ by ρ_k $(k \in I_\infty)$ may be infinite; we need additional treatments to avoid it.

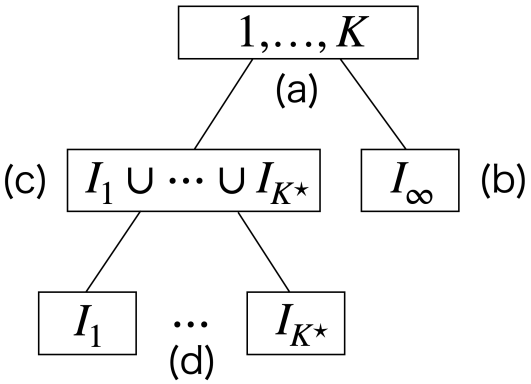


Figure 5. Decomposition of the MC to prove Theorem 2.

5. Applications

In this section, we propose methods to apply the MC to clustering change detection problems. Formally speaking, given the dataset $\mathcal{X} := \{\{x_{n,t}\}_{n=1}^N \mid t \in 1, \ldots, T\}$, where t denotes the time and $\{x_{n,t}\}_{n=1}^N$ denote the data generated at each t, we consider the problem of monitoring the changes in the clustering structures over $t = 1, \ldots, T$.

First, we briefly summarize the method named sequential dynamic model selection (SDMS) [28] that addresses this problem. Then, we introduce our ideas and discuss the differences between SDMS.

Hereafter, we assume that the data points $x_{n,t}$ are d-dimensional vectors and consider a Gaussian mixture model

$$f_t(x) = \sum_{k=1}^{K_t} \rho_{k,t} \mathcal{N}(x | \mu_{k,t}, \Sigma_{k,t})$$

for each t.

5.1. Sequential Dynamic Model Selection

SDMS is an algorithm that is used to sequentially estimate models and find changes. In clustering change detection problems, it sequentially estimates the mixture sizes $\hat{K}_t$ and parameters $\eta_{\hat{K}_t} := \{\hat{\rho}_{k,t}, \hat{\mu}_{k,t}, \hat{\Sigma}_{k,t}\}_{k=1}^{\hat{K}_t}$ and finds model changes as changes in $\hat{K}_t$.

The estimation procedures are explained below. First, depending on the estimated mixture size at the last time point $\hat{K}_{t-1}$, we set the candidate for K_t. Then, for each K_t

in the candidate, we estimate the parameters θ_{K_t} from the data $\{x_{n,t}\}_{n=1}^N$ and calculate a cost function $\mathcal{L}_{\mathrm{SDMS}}(\{x_{n,t}\}_{n=1}^N; K_t, \theta_{K_t}, \hat{K}_{t-1})$. Finally, we select K_t as the mixture size that minimizes the costs. The candidates of K_t are set as

$$\{1, \ldots, K_{\max}\}$$

at $t = 1$, and

$$\{K_{t-1} - 1, K_{t-1}, K_{t-1} + 1\} \cap \{1, \ldots, K_{\max}\}$$

at $t \geq 2$, where $K_{\max}$ is a pre-defined parameter. The cost function denotes the sum of the code length functions of the model and model changes given by

$$\mathcal{L}_{\mathrm{SDMS}}\left(\{x_{n,t}\}_{n=1}^N; K_t, \eta_{K_t}, \hat{K}_{t-1}\right) = \mathcal{L}_{\mathrm{model}}\left(\{x_{n,t}\}_{n=1}^N; K_t, \eta_{K_t}\right) + \mathcal{L}_{\mathrm{change}}\left(K_t \,\middle|\, \hat{K}_{t-1}\right).$$

Code Length of the Model

The score $\mathcal{L}_{\mathrm{model}}(\{x_n\}_{n=1}^N; K, \eta_K)$ denotes a sum of the logarithm of the likelihood functions and penalty terms corresponding to the complexity of the model. In this study, we consider two likelihood functions and four penalty terms. For the (logarithm of) likelihood functions, we consider the *observed likelihood* $L(\{x_n\}_{n=1}^N; \theta_K)$ and *complete likelihood* $L(\{x_n, z_n\}_{n=1}^N; \theta_K)$, provided by

$$L(\{x_n\}_{n=1}^N; \theta_K) := \sum_{n=1}^N \log P(X = x_n) = \sum_{n=1}^N \log\left(\sum_{k=1}^K \rho_k \mathcal{N}(x_n|\mu_k, \Sigma_k)\right),$$
$$L(\{x_n, z_n\}_{n=1}^N; \theta_K) := \sum_{n=1}^N \log P(X = x_n, Z = z_n) = \sum_{n=1}^N \log(\rho_{z_n} \mathcal{N}(x_n|\mu_{z_n}, \Sigma_{z_n})),$$

where $\{z_n\}_{n=1}^N$ are the latent variables for the data estimated by

$$z_n := \underset{z \in 1,\ldots,K}{\mathrm{argmax}} P(Z = z|X = x_n).$$

They correspond to the likelihood of the observed data and complete data, respectively; the former is used to determine the mixture size, and the latter is used to determine the cluster size under the assumption that it is equal to the mixture size. For the penalty terms, we consider AIC [8], BIC [9], NML [13], and DNML [47,48]. By combining the log-likelihood and the penalty terms, we consider the following six scores:

- AIC with observed likelihood (AIC):

$$-L\left(\{x_n\}_{n=1}^N; \eta_K\right) + D,$$

- AIC with complete likelihood (AIC+comp):

$$-L\left(\{x_n, z_n\}_{n=1}^N; \eta_K\right) + D,$$

- BIC with observed likelihood (BIC):

$$-L\left(\{x_n\}_{n=1}^N; \eta_K\right) + \frac{D}{2} \log N,$$

- BIC with complete likelihood (BIC+comp):

$$-L\left(\{x_n, z_n\}_{n=1}^N; \eta_K\right) + \frac{D}{2} \log N,$$

- NML:

$$-L\left(\{x_n, z_n\}_{n=1}^N; \eta_K\right) + \log \mathrm{PC}_{\mathrm{NML}}(N, K),$$

- DNML:

$$-L\left(\{x_n, z_n\}_{n=1}^N; \theta_K\right) + \log \mathrm{PC}_{\mathrm{DNML}}\left(N, \{z_n\}_{n=1}^N, K\right).$$

where $D := (K-1) + d(d+3)/2$ denotes the number of the free parameters required to represent a Gaussian mixture model; $\mathrm{PC}_{\mathrm{NML}}(N, K)$ and $\mathrm{PC}_{\mathrm{DNML}}(N, \{z_n\}_{n=1}^N, K)$ denote the parametric complexities. In our experiments, we estimated the parameter η_K by conducting the EM algorithm [49] implemented in the Scikit-learn package [50] ten times and selected the best parameter that minimized each score. Note that in NML and DNML, we only considered the complete likelihood functions because only the methods to calculate their parametric complexities are known.

5.2. Track MC

In SDMS, clustering changes are detected as the changes of the mixture size or cluster size K; because it is discrete, the changes have been considered to be abrupt. Then, we propose to track MC instead of K while estimating the parameters using SDMS. Because MC takes a real value, monitoring it is more suitable for observing gradual changes than monitoring K. The algorithm for tracking MC is explained in Algorithm 1.

Algorithm 1 Tracking MC

Require: A dataset $\mathcal{X} = \{\{x_{n,t}\}_{n=1}^N \mid t \in 1, \ldots, T\}$.
1: **for** $t = 1$ **to** $t = T$ **do**
2: Estimate $\hat{K}_t$ and $\{\hat{g}_{k,t}\}_{k=1}^{\hat{K}_t}$ from the data $\{x_{n,t}\}_{n=1}^N$ using SDMS.
3: Calculate $\mathrm{MC}_t := \mathrm{MC}(\{\hat{\gamma}_k(x_n)\}_{k,n})$.
4: **end for**
5: **return** $\{\mathrm{MC}_t\}_{t=1}^T$.

5.3. Track MC with Its Decomposition

In addition to monitoring the MC of the entire structure, we also propose an algorithm to track its decomposition. To accomplish this, we must estimate the upper L components and their corresponding partitions $Q_{k,t}^{(l)}$ for each t.

Here, we assume that the upper L components are common at every t and estimate the partition $Q_{k,t}^{(l)}$ after estimating the lower components at each time. Specifically, we consider $\mu_{k,t}$ as a point with weights $\rho_{k,t}$ for each k and t and cluster them. As the clustering algorithm, we modified the fuzzy c-means [20] to handle the weighted points. Formally, we estimated the centers of the upper L components $\tilde{\mu}_l$ and their corresponding partitions $Q_{k,t}^{(l)}$ by minimizing the loss function

$$\sum_{t,k} \rho_{k,t} \sum_{l=1}^{L} \left(Q_{k,t}^{(l)}\right)^m \left\|\mu_{k,t} - \tilde{\mu}_l\right\|^2,$$

where $m > 0$ is parameter that determines the fuzziness of the partition.

We estimated $\tilde{\mu}_l$ and $Q_{k,t}^{(l)}$ by minimizing one iteratively while fixing another. We can formulate the iteration as follows:

$$\tilde{\mu}_l = \frac{\sum_{k,t} \rho_{k,t} \left(Q_{k,t}^{(l)}\right)^m \mu_{k,t}}{\sum_{k',t'} \rho_{k',t'} \left(Q_{k',t'}^{(l)}\right)^m},$$

$$Q_{k,t}^{(l)} = \frac{\left\|\mu_{k,t} - \tilde{\mu}_l\right\|^{2/(m-1)}}{\sum_{l'=1}^{L} \left\|\mu_{k,t} - \tilde{\mu}_{l'}\right\|^{2/(m-1)}}.$$

Finally, we present an algorithm to track the MC and its decomposition in Algorithm 2. We can analyze the structural changes in more detail by evaluating the decomposed values.

Algorithm 2 Tracking MC with its decomposition

Require: A dataset $\mathcal{X} = \{\{x_{n,t}\}_{n=1}^{N} \mid t \in 1, \ldots, T\}$, parameters m and L.

1:
2: *# Step 1: Estimate lower components.*
3: **for** $t = 1$ **to** $t = T$ **do**
4: Estimate $\hat{K}_t$ and $\{\hat{g}_{k,t}\}_{k=1}^{\hat{K}_t}$ from the data $\{x_{n,t}\}_{n=1}^{N}$ using SDMS.
5: Calculate MC(total)$_t$:= MC($\{\hat{\gamma}_k(x_n)\}_{k,n}$).
6: **end for**
7:
8: *# Step 2: Estimate upper components and partition.*
9: Estimate the centers $\tilde{\mu}_l$ and the partition $Q_{k,t}^{(l)}$ using fuzzy c-means.
10:
11: *# Step 3: Calculate the decomposition of MC.*
12: **for** $t = 1$ **to** $t = T$ **do**
13: Calculate MC(interaction)$_t$ defined in Section 4.2.
14: **for** $l = 1$ **to** $l = L$ **do**
15: Calculate W(component l)$_t$ defined in Section 4.2.
16: Calculate MC(component l)$_t$ defined in Section 4.2.
17: **end for**
18: **end for**
19: **return** $\{\text{MC(total)}_t\}_{t=1}^{T}$, $\{\text{MC(interaction)}_t\}_{t=1}^{T}$, $\{\{\text{W(component } l)_t\}_{l=1}^{L}\}_{t=1}^{T}$, $\{\{\text{MC(component } l)_t\}_{l=1}^{L}\}_{t=1}^{T}$.

6. Experimental Results

In this section, we present the experimental results that demonstrate the MC's ability to monitor the clustering changes. We compare our methods to the monitoring of K.

6.1. Analysis of Artificial Data

To reveal the behaviors of MC, we conducted experiments with two artificial datasets called *move Gaussian dataset* and *imbalance Gaussian dataset*. Their experimental designs are discussed below. First, we generated artificial datasets $\mathcal{X} = \{\{x_{n,t}\}_{n=1}^{N} \mid t \in 1, \ldots, T\}$ by setting $T = 150$ and $N = 1000$. The datasets have one transaction period $t = 51, \ldots, 100$ in which the data change their clustering structures gradually. Then, we estimated the MC and K using the methods in Sections 5.1 and 5.2 by setting $K_{\max} = 10$. To compare them, we first created a simple algorithm to detect the changes from the sequence of MC or K. Then, we compared the abilities of this algorithm in terms of the speed and accuracy of detecting the change points. Moreover, to evaluate the abilities to find the changes in the opposite direction, we performed experiments with the same datasets in the reverse order.

Given a sequence of the MC or K written as $y_1, \ldots, y_{150}$, we constructed an algorithm to detect the change points as follows. For $t = 10, \ldots, 150$, we raised a change alert if

$$|\text{median}(y_{t-9}, \ldots, y_{t-5}) - \text{median}(y_{t-4}, \ldots, y_t)| > \varepsilon$$

in the case of MC, and

$$\text{median}(y_{t-9}, \ldots, y_{t-5}) \neq \text{median}(y_{t-4}, \ldots, y_t)$$

in the case of K, where ε is the threshold to raise an alert in MC. It should be to some extent large for avoiding too many false alerts and smaller than 1 to find the changes earlier than with monitoring K. In this section, we set ε as 0.01 so as not to raise alerts from t = 1 to 10 assuming that we know that there are no changes in this period. We calculated the medians instead of the means of the subsequences for robustness. However, to avoid redundant

alerts, we neglected them when the difference between t and the latest alert was less than 5 even if the conditions were satisfied.

To evaluate the quality of the algorithm, we calculated *Delay* and *False alarm rate* (FAR), defined as

$$\begin{aligned} \text{Delay} &:= \min(t^* - 51, 50), \\ \text{FAR} &:= \frac{\#\{t \in [10, 150] \mid t \notin \text{ACCEPT} \wedge t \in \text{ALERT}\}}{\#\{t \in [10, 150] \mid t \notin \text{ACCEPT}\}}, \end{aligned}$$

where t^* denotes the first time point in the transaction period when the algorithm generated an alert, ACCEPT denotes the set of time points when alerts can be defined as $\{t \mid \exists t - 9, \ldots, t \in [51, \ldots, 100]\} = [51, 109]$, and ALERT denotes the set of time points when the algorithm generates alerts.

6.1.1. Move Gaussian Dataset

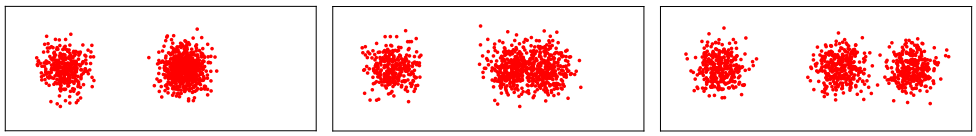
The move Gaussian dataset is a set of three-dimensional Gaussian distributions, whose means move gradually in the transaction period. Formally, for each t, we generated the data $\{x_{n,t}\}_{n=1}^{1000}$ as follows:

$$x_{n,t} \sim \begin{cases} \mathcal{N}(x|\mu = [0,0,0]^\top, \Sigma = I_3) & (1 \le n \le 333), \\ \mathcal{N}(x|\mu = [10,0,0]^\top, \Sigma = I_3) & (334 \le n \le 666), \\ \mathcal{N}(x|\mu = [10 + \alpha(t), 0, 0]^\top, \Sigma = I_3) & (667 \le n \le 1000), \end{cases}$$

where

$$\alpha(t) = \begin{cases} 0 & (1 \le t \le 50), \\ 0.12(t - 50) & (51 \le t \le 100), \\ 6 & (101 \le t \le 150). \end{cases}$$

The first and second dimensions of some data are visualized in Figure 6. In the direction $t = 1 \to 150$, the number of clusters increases from two to three as the two clusters leave; in the direction $t = 150 \to 1$, it decreases from three to two as the two clusters merge.



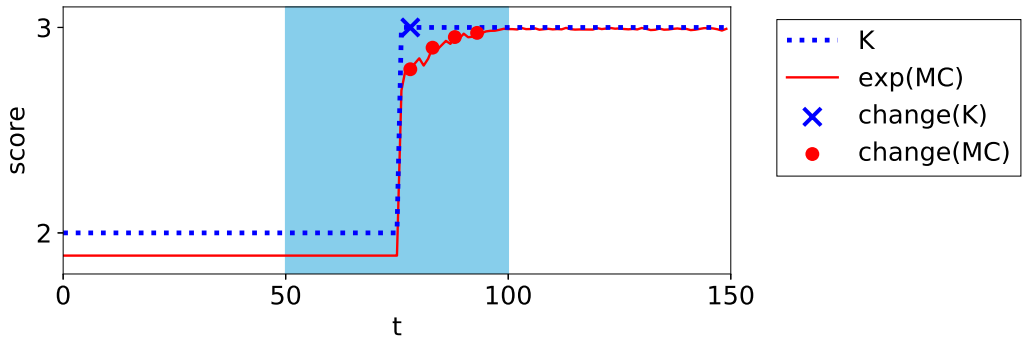
(**a**) $t = 1$ (**b**) $t = 75$ (**c**) $t = 101$

Figure 6. Scatter plots of the first and second dimensions of the data at $t = 1, 75, 101$ in the move Gaussian dataset.

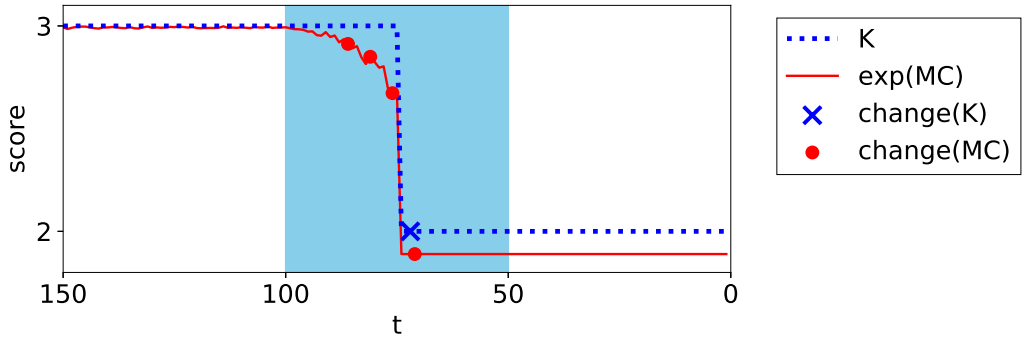
The experiments were performed ten times by randomly generating the datasets; accordingly, the average performance scores were calculated. The differences in the scores between the MC and K for each criterion are presented in Table 2; the estimated MC and K in one trial are proposed in Figure 7. This figure illustrates the result of BIC as an example.

Table 2. Difference in the average performance score between MC and K for the move Gaussian dataset.

	(Score of MC) − (Score of K)			
	$t = 1 \to 150$		**$t = 150 \to 1$**	
Criterion	**Delay**	**FAR**	**Delay**	**Far**
AIC	0.0	0.000	−20.6	0.000
AIC+comp	0.0	0.000	−10.9	0.000
BIC	0.0	0.000	−17.5	0.000
BIC+comp	0.0	0.000	−8.9	0.000
NML	0.0	0.000	−7.9	0.000
DNML	0.0	0.000	−7.7	0.000



(**a**) $t = 1 \to 150$



(**b**) $t = 150 \to 1$

Figure 7. Plots of the exponential of MC and K for the move Gaussian dataset. The filled area represents the transaction period. The markers on the plot represent the alerts in each method.

With respect to the speed to find changes, in every criterion, MC performed as well as K in the direction $t = 1 \to 150$; however, it performed significantly better than K in the direction $t = 150 \to 1$. The reason for the differing performances is discussed below. In the direction $t = 1 \to 150$, the model selection algorithms underestimated the number of components at the beginning of the transaction period. In such time points, they ignored the overlapping of the two components and considered them as one cluster. Thus, MC, based on such model selection methods, was unable to find the changes earlier than K. However, in the direction $t = 150 \to 1$, the overlap between the components was correctly estimated at some time points before K changed. In this case, MC changed smoothly according to the overlap and found changes earlier than K.

With respect to the accuracy of finding changes, MC performed as well as K in terms of FAR. Additionally, it is evident from Figure 7 that MC stably estimated the clustering structures.

6.1.2. Imbalance Gaussian Dataset

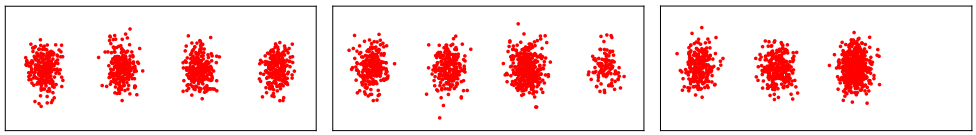
The imbalance Gaussian dataset is a set of three-dimensional Gaussian mixture distributions whose balances change gradually in the transaction period. Formally, for each t, we generated the data $\{x_{n,t}\}_{n=1}^{1000}$ as follows:

$$x_{n,t} \sim \begin{cases} \mathcal{N}(x|\mu = [0,0,0]^\top, \Sigma = I_3) & (1 \leq n \leq 250), \\ \mathcal{N}(x|\mu = [10,0,0]^\top, \Sigma = I_3) & (251 \leq n \leq 500), \\ \mathcal{N}(x|\mu = [20,0,0]^\top, \Sigma = I_3) & (501 \leq n \leq 750 + \alpha(t)), \\ \mathcal{N}(x|\mu = [30,0,0]^\top, \Sigma = I_3) & (751 + \alpha(t) \leq n \leq 1000), \end{cases}$$

where

$$\alpha(t) = \begin{cases} 0 & (1 \leq t \leq 50), \\ 5(t-51) & (51 \leq t \leq 100), \\ 250 & (101 \leq t \leq 150). \end{cases}$$

The first and second dimensions of some data are visualized in Figure 8. In the direction $t = 1 \rightarrow 150$, the number of clusters decreases from four to three as the edge cluster disappears. In the direction $t = 150 \rightarrow 1$, it increases from three to four as the edge cluster emerges.



(**a**) $t = 1$ (**b**) $t = 80$ (**c**) $t = 101$

Figure 8. Scatter plots of the first and second dimensions of the data at $t = 1, 80, 101$ in the imbalance Gaussian dataset.

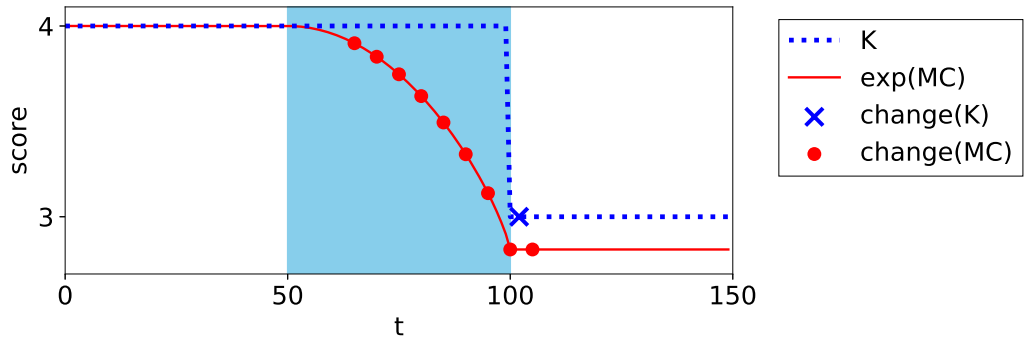
The experiments were performed ten times by randomly generating datasets; accordingly, the average performance scores were calculated. The difference in the scores between the MC and K for each criterion are listed in Table 3. The estimated MC and K in one trial are plotted in Figure 9. This figure illustrates the result of BIC as an example.

Table 3. Differences in the average performance score between MC and K for the imbalance Gaussian dataset.

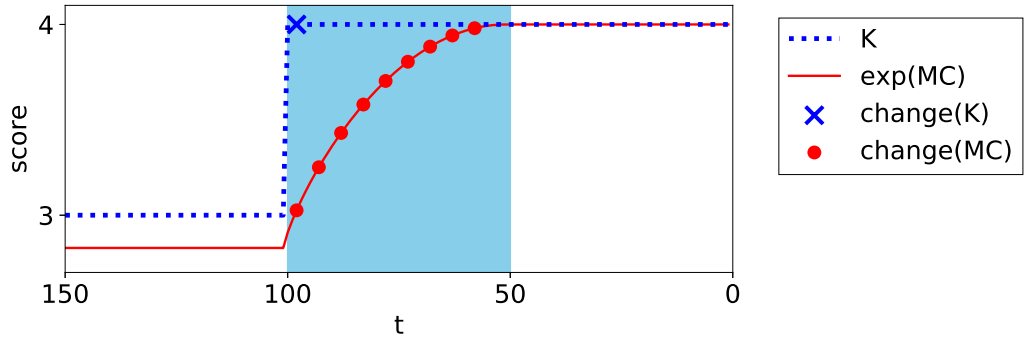
	(Score of MC) − (Score of *K*)			
	$t = 1 \rightarrow 150$		$t = 150 \rightarrow 1$	
Criterion	**Delay**	**FAR**	**Delay**	**Far**
AIC	−30.2	0.010	−4.6	0.000
AIC+comp	−34.0	0.000	0.0	0.000
BIC	−34.0	0.000	0.0	0.000
BIC+comp	−34.0	0.000	0.0	0.000
NML	−34.0	0.000	0.0	0.000
DNML	−34.0	0.000	0.0	0.000

In terms of the speed to find changes, in every model selection method, MC performed significantly better than K in the direction $t = 1 \rightarrow 150$; however, MC performed as well as K in the direction $t = 150 \rightarrow 1$. The reason for the differing performances is discussed below. In the transaction period, all model selection methods counted the minor components as independent clusters. Then, in the direction $t = 1 \rightarrow 150$, MC changed smoothly according to the imbalance and determined the changes earlier than K. In the direction $t = 150 \rightarrow 1$, K increased significantly early in the transaction period. Then, MC increased along with K and determined the changes simultaneously.

In terms of the accuracy of finding changes, MC performed as well as K in terms of FAR. Additionally, it is evident from Figure 9 that MC stably estimated the clustering structures.



(**a**) $t = 1 \rightarrow 150$



(**b**) $t = 150 \rightarrow 1$

Figure 9. Plots of the exponential of MC and K for the imbalance Gaussian dataset. The filled area represents the transaction period. The markers on the plot represent the alerts in each method.

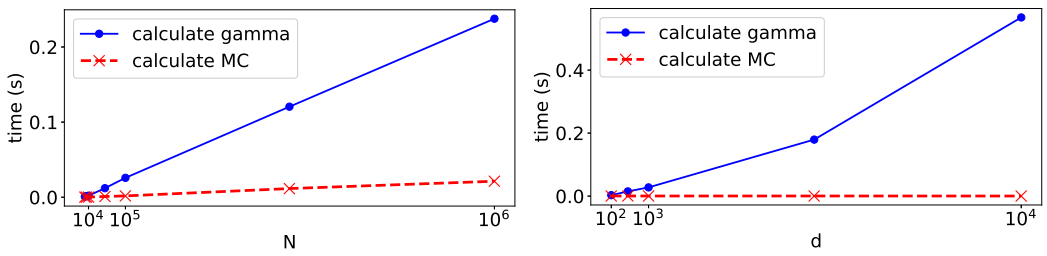
6.1.3. Scalability

To discuss the scalability for the large datasets, we explored the increase in the computation time for the data size. First, we set the mixture distribution f as

$$f(x) = 0.5 \times \mathcal{N}\left(x \middle| \mu = [0, \ldots, 0]^\top, \Sigma = I^d\right) + 0.5 \times \mathcal{N}\left(x \middle| \mu = [1, \ldots, 1]^\top, \Sigma = I^d\right)$$

and sampled N points from f. Then, we recorded the time to calculate $\{\gamma_{k,n}\}$ from f and calculated the MC from $\{\gamma_{k,n}\}$. We repeatedly measured the computation times by increasing N and d. For each N and d, we measured them ten times and took their averages.

The increase in the computation times is illustrated in Figure 10. In (a), although both computation times increased linearly as N grew, calculating MC was faster than calculating $\{\gamma_{k,n}\}$. In (b), although the time to calculate $\{\gamma_{k,n}\}$ increased as d grew, and the computation time for MC was almost constant because K and N were not changed. Overall, the cost of computing MC is much smaller than that of computing or estimating $\{\gamma_{k,n}\}$.



(**a**) (**b**)

Figure 10. Relationships between the computation time and N and d. In (**a**), we fixed $d = 10$ and varied N from 10 to 10^6. In (**b**), we fixed N = 10,000 and varied d from 10 to 10^4.

6.2. Analysis of Real Data

We analyzed two types of real data named the *beer dataset* and *house dataset*, which are summarized in Table 4. In the following subsections, we discussed the detail of the datasets and results of the experiments.

Table 4. Summary of the dataset.

Dataset	T	N_t	d	Description
beer	92	3185	16	purchase data of beer.
house	96	4326	3	electricity consumption data in a house.

6.2.1. Beer Dataset

We discuss the results of the beer dataset, obtained from Hakuhodo, Inc. and M-CUBE, Inc. This has also been analyzed in [28,34]. The dataset comprises the records of customer's beer purchases from November 1st, 2010 to January 31th, 2011. The dataset $\mathcal{X}$ is constructed as follows. The time unit is a day. For each day $t \in \{\tau, \ldots, T\}$, $x_{n,t} \in \mathbb{R}^d$ denotes the n-th customer's consumption of the beer from time $t - \tau + 1$ to t, where we set $\tau = 14$. The dimension d of the vector is 16, which correspond to the consumptions of the following drink:

- beer (A), ..., beer (F): beer with brand name A, ..., F.
- beer (other): beer with other brands.
- beerlike (A), ..., beerlike (H): beer-like drink with brand name A, ..., H.
- beerlike (other): beer-like drink with other brands.

First, we compare the plots of the estimated MC and K in Figure 11. The results of BIC and NML are illustrated as an example. Note that we omit the results of AIC because it chose $K_{\max}$ for $\widehat{K}_t$ at many t. In any method, the score was high at the end and beginning of the year, reflecting the increased activities in transactions. However, because the critical changes in the clustering structure and changes due to ineffective components were mixed, the sequence of K had a lot of change points; as a result, it was difficult to interpret their meanings. On the other hand, MC identified the clustering structure by discounting the effects of the ineffective components. As a result, the sequence of MC highlighted the significant changes at the end and beginning of the year. It is also worthwhile noting that the differences of the scores between the model selection methods were much smaller in MC than those in K; this indicates that both BIC and NML estimates similar clustering structures under the concept of MC even though the number of components differs significantly.

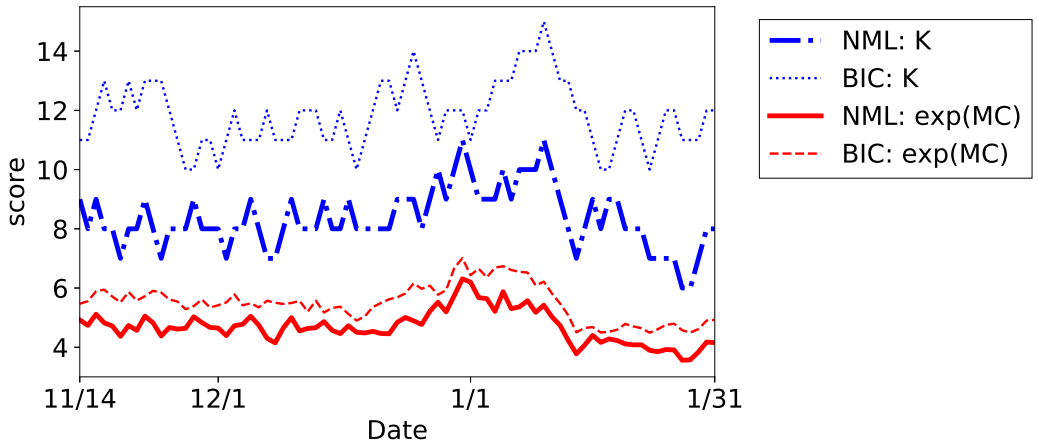


Figure 11. Plots of the sequences of the MC and K in the beer dataset.

Next, we discuss the results of the decomposition of MC. We present the results of BIC and NML with $L = 4$ and $m = 1.5$. The centers of the upper components are listed in Tables 5 and 6, respectively, and the plots of each decomposed value are illustrated

in Figures 12 and 13, respectively. The indices of the upper components are manually rearranged so that they correspond with each other; then, it can be observed that the results were also similar to each other. The structures can be extensively evaluated by analyzing the decomposed values. For instance, let us analyze the decomposed values at the end and beginning of the year. As evident from the tables, they had different characteristics. It can be observed from the figures that the contributions increased in all components, indicating that they were related to the increase in MC(total). The weight of the component decreased in cluster 1 and increased in component 2 and 3, indicating that the customers moved from component 1 to component 2 and 3. Additionally, MC(component l) increased in all components, indicating that the complexity or diversity increased within them.

Table 5. Centers of the upper components estimated by BIC in the beer dataset. For each dimension, the maximum value is denoted in boldface.

	Component 1	**Component 2**	**Component 3**	**Component 4**
beer(A)	0.09	0.44	**1.93**	0.16
beer(B)	0.07	0.23	**0.96**	0.06
beer(C)	0.07	0.20	**0.83**	0.07
beer(D)	0.05	0.20	**0.58**	0.05
beer(E)	0.03	0.06	**0.35**	0.03
beer(F)	0.03	0.06	**0.35**	0.02
beer(other)	0.04	0.12	**0.69**	0.10
beerlike(A)	0.02	**5.85**	0.23	0.07
beerlike(B)	0.09	0.57	**0.80**	0.21
beerlike(C)	0.10	0.63	**0.83**	0.22
beerlike(D)	0.07	0.40	**0.57**	0.18
beerlike(E)	0.04	0.12	**0.51**	0.06
beerlike(F)	0.04	0.20	**0.34**	0.13
beerlike(G)	0.05	0.10	**0.40**	0.06
beerlike(H)	0.03	0.09	**0.26**	0.04
beerlike(other)	0.09	1.27	1.11	**6.78**

Table 6. Centers of the upper components estimated by NML in the beer dataset. For each dimension, the maximum value is denoted in boldface.

	Component 1	**Component 2**	**Component 3**	**Component 4**
beer(A)	0.08	0.48	**1.90**	0.12
beer(B)	0.04	0.30	**1.04**	0.07
beer(C)	0.05	0.20	**0.95**	0.04
beer(D)	0.04	0.19	**0.64**	0.09
beer(E)	0.02	0.06	**0.38**	0.02
beer(F)	0.02	0.07	**0.40**	0.01
beer(other)	0.03	0.11	**0.68**	0.19
beerlike(A)	0.02	**5.79**	0.21	0.07
beerlike(B)	0.10	0.52	**0.73**	0.18
beerlike(C)	0.11	0.61	**0.70**	0.21
beerlike(D)	0.06	0.49	**0.52**	0.24
beerlike(E)	0.04	0.12	**0.47**	0.07
beerlike(F)	0.04	0.18	**0.30**	0.24
beerlike(G)	0.04	0.11	**0.44**	0.07
beerlike(H)	0.02	0.10	**0.23**	0.09
beerlike(other)	0.08	1.42	1.08	**6.54**

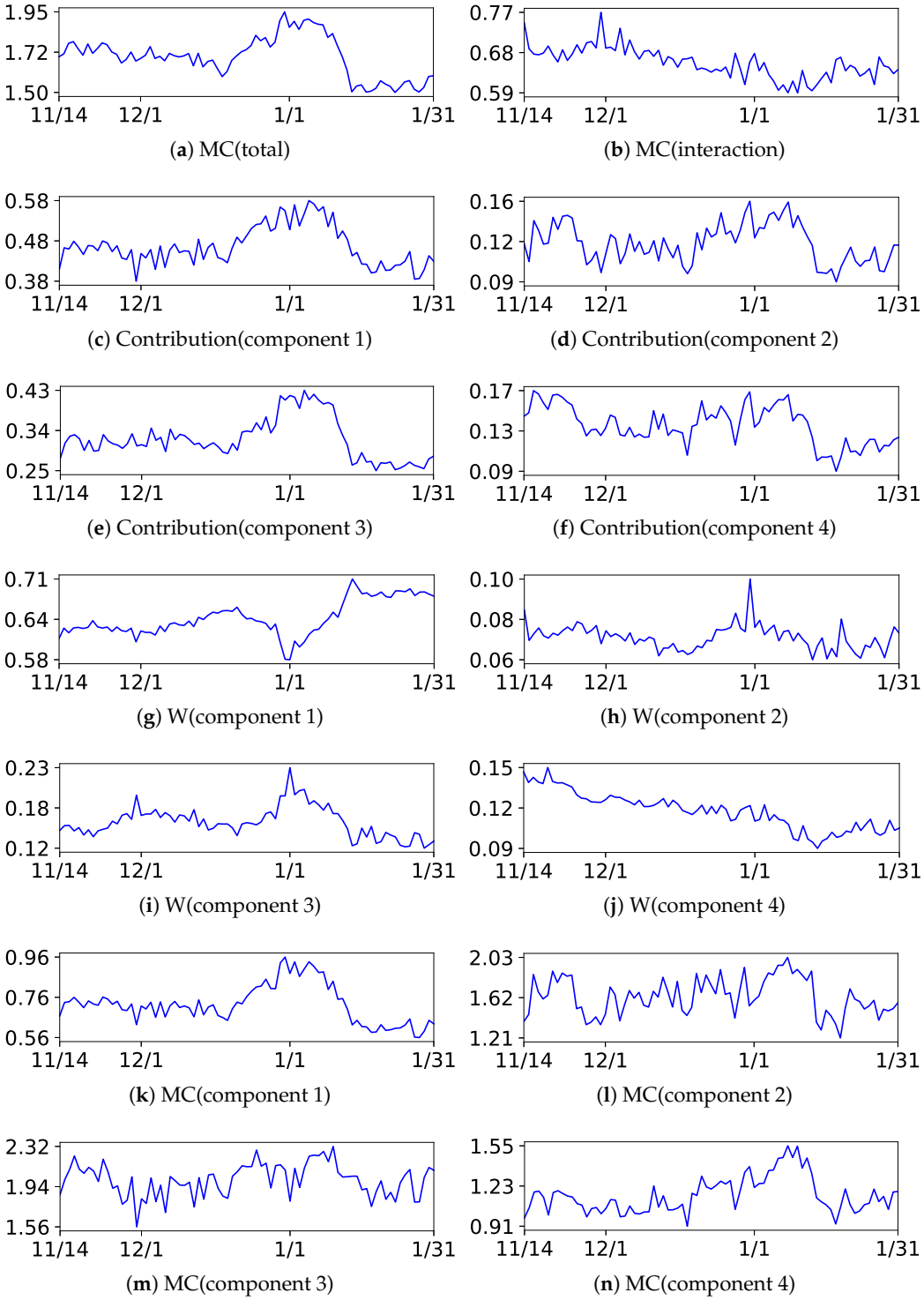


Figure 12. Plots of the decomposition of MC with BIC in the beer Dataset.

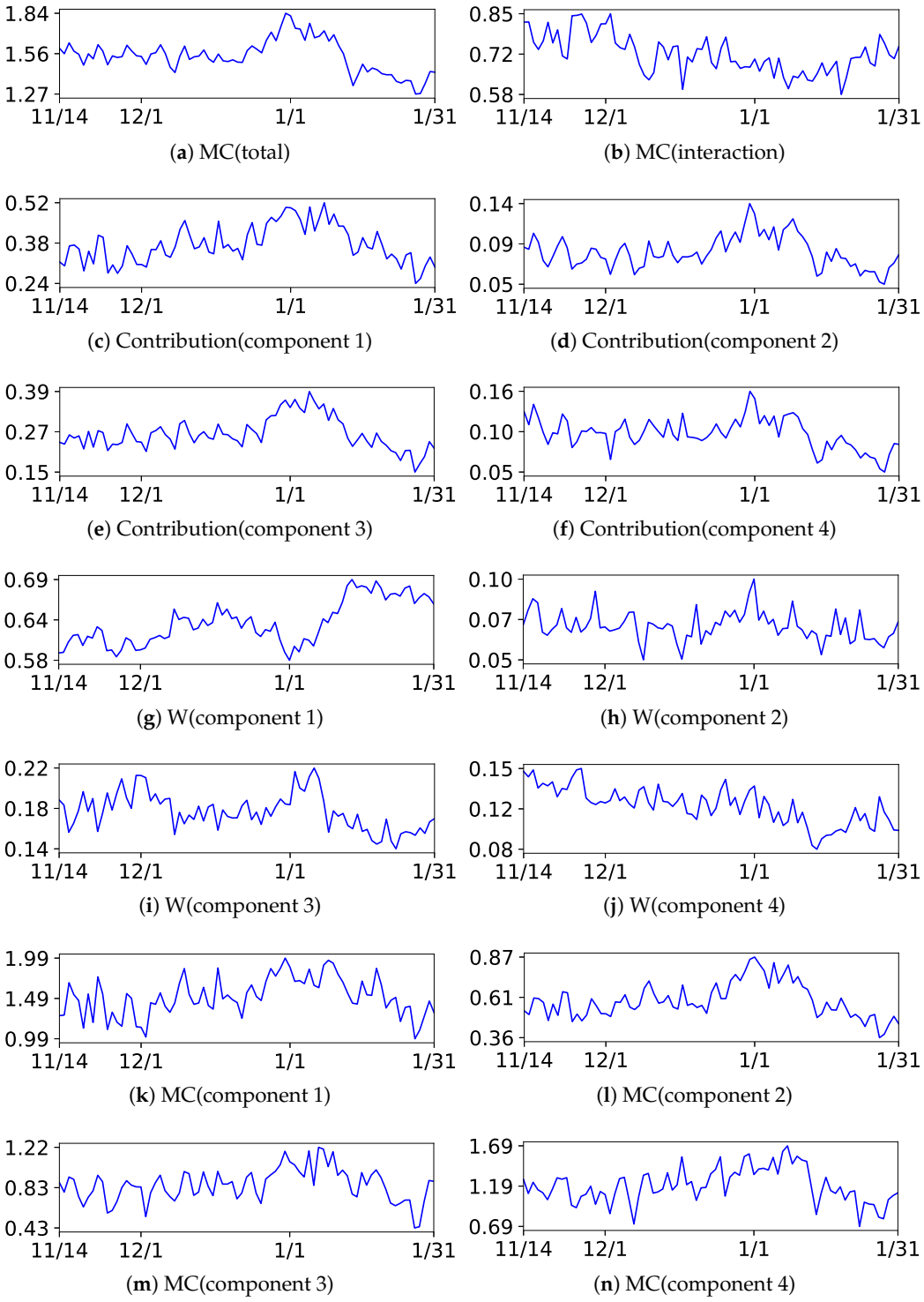


Figure 13. Plots of the decomposition of MC with NML in the beer Dataset.

6.2.2. House Dataset

We discuss the results of the house dataset, obtained from the UCI Machine Learning Repository [51]. The dataset comprises the records of electricity consumption in a house every five minutes from 16 December 2006 to 26t November 2010. The dataset $\mathcal{X}$ is constructed as follows. The time unit is 15 min from 00:00–00:15 to 23:45–24:00. For each t, the data $\{x_{n,t}\}_{n=1}^{N}$ denotes the set of the records on the various days included in the t-th time unit. The dimension d of the vector is 3, which corresponds to the metering of the following three points:

- metering(A): a kitchen.
- metering(B): a laundry room.
- metering(C): a water-heater and an air-conditioner.

First, we compare the plots of the estimated K and the corresponding MC in Figure 14. The results of BIC and NML are illustrated as an example. Note that we omit the results of AIC because it chose $K_{\max}$ for $\widehat{K}_t$ at many t. It can be observed from the figure that the MC smoothly connected the discrete changes in K; therefore, MC expressed gradual changes in the dataset more effectively than K. Additionally, the MCs in BIC and NML were more similar to each other than K as well as in the beer dataset. The values of MC started increasing from around 7:00 a.m.; after slight fluctuations, the value reached its peak around 21:00. Therefore, MC seemed to represent the amount of activities in this house.

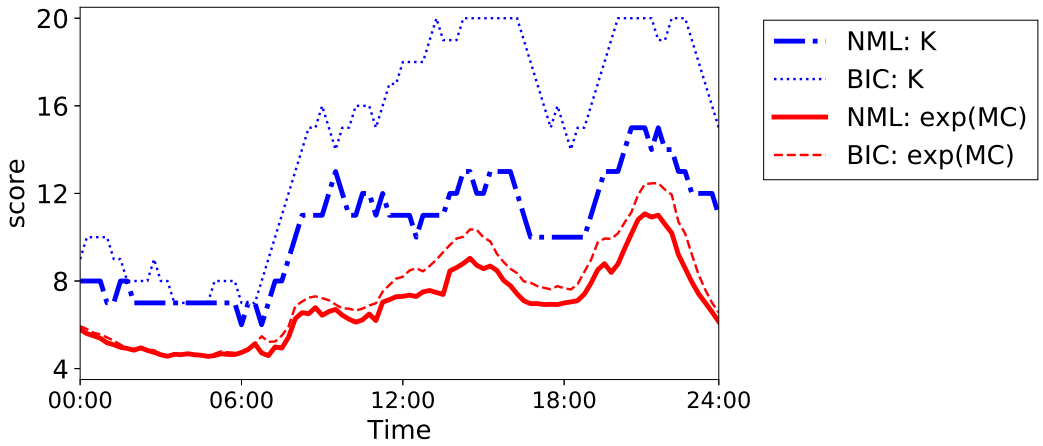


Figure 14. Plots of the sequences of the MC and K in the house dataset.

Next, we discuss the results of the decomposition of MC. We present the results of BIC and NML with $L = 4$ and $m = 1.5$. The centers of the upper components are listed in Tables 7 and 8, respectively, and the plots of each decomposed value are illustrated in Figures 15 and 16, respectively. The indices of the upper components are manually rearranged so that they correspond with each other; then, it can be observed that the results were also similar to each other. The structures can be extensively evaluated by analyzing the decomposed values. For instance, let us analyze the decomposed values in component 3. It can be observed from the tables that the value in metering(C) was specifically high in this component. Looking at the contribution (component 3), there were two peaks around 9:00 and 21:00; it represented the increased activities in this component. However, the proportions of the weight and MC were different. W(component 3) was specifically high at 9:00, indicating that the first half of the peaks was due to the increase in the weight of the component; whereas, MC(component 3) was specifically high at 21:00, indicating that the second half of the peaks was due to the increase in the complexity within the component.

Table 7. Centers of the upper components estimated by BIC in the house dataset. For each dimension, the maximum value is denoted in boldface.

	Component 1	**Component 2**	**Component 3**	**Component 4**
metering(A)	0.04	**4.47**	0.13	0.41
metering(B)	0.53	0.89	0.56	**4.40**
metering(C)	0.75	3.34	**4.37**	2.96

Table 8. Centers of the upper components estimated by NML in the house dataset. For each dimension, the maximum value is denoted in boldface.

	Component 1	Component 2	Component 3	Component 4
metering(A)	0.04	**4.24**	0.11	0.35
metering(B)	0.53	1.00	0.57	**4.48**
metering(C)	0.76	3.37	**4.38**	2.93

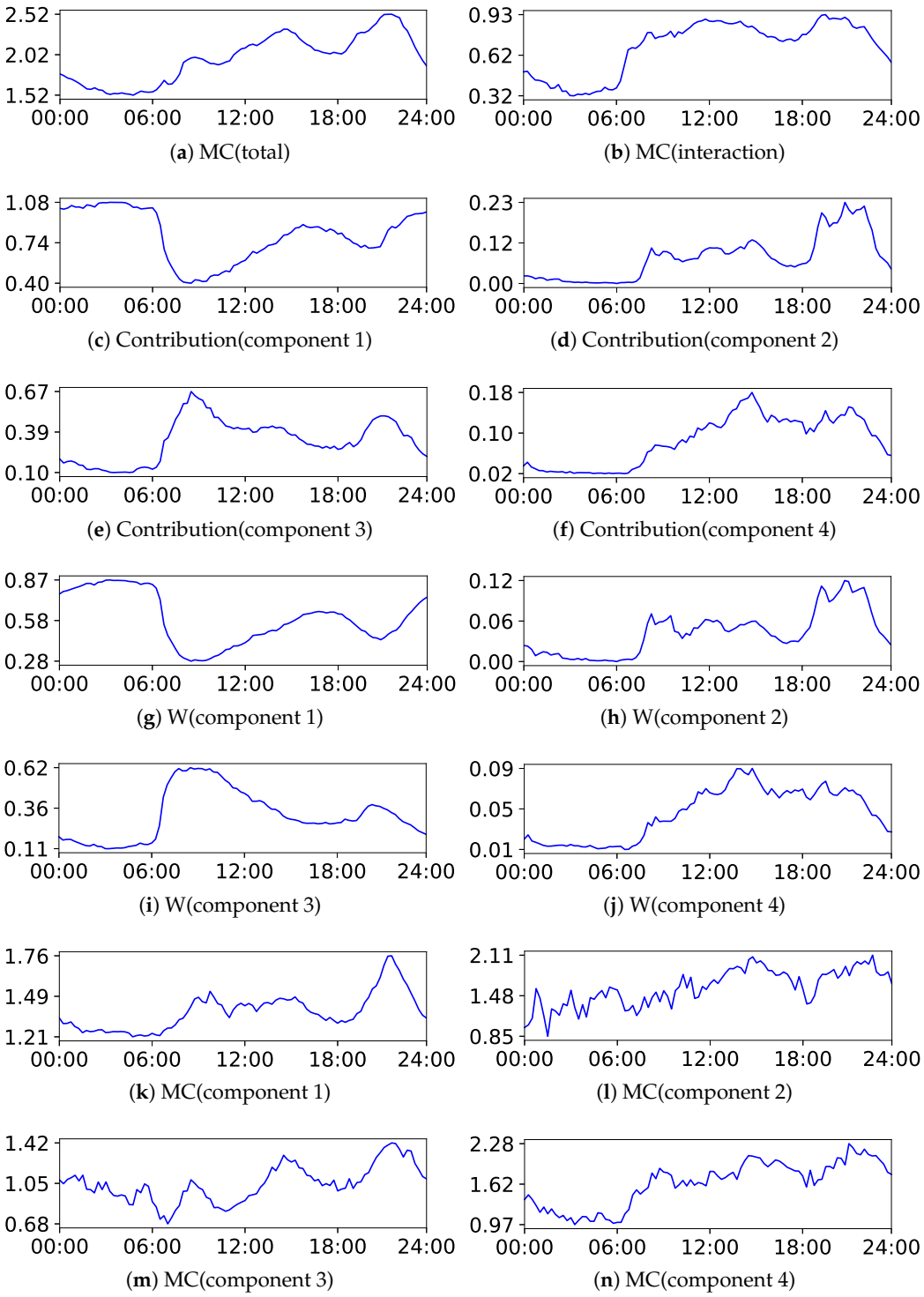


Figure 15. Plots of the decomposition of MC with BIC in the house Dataset.

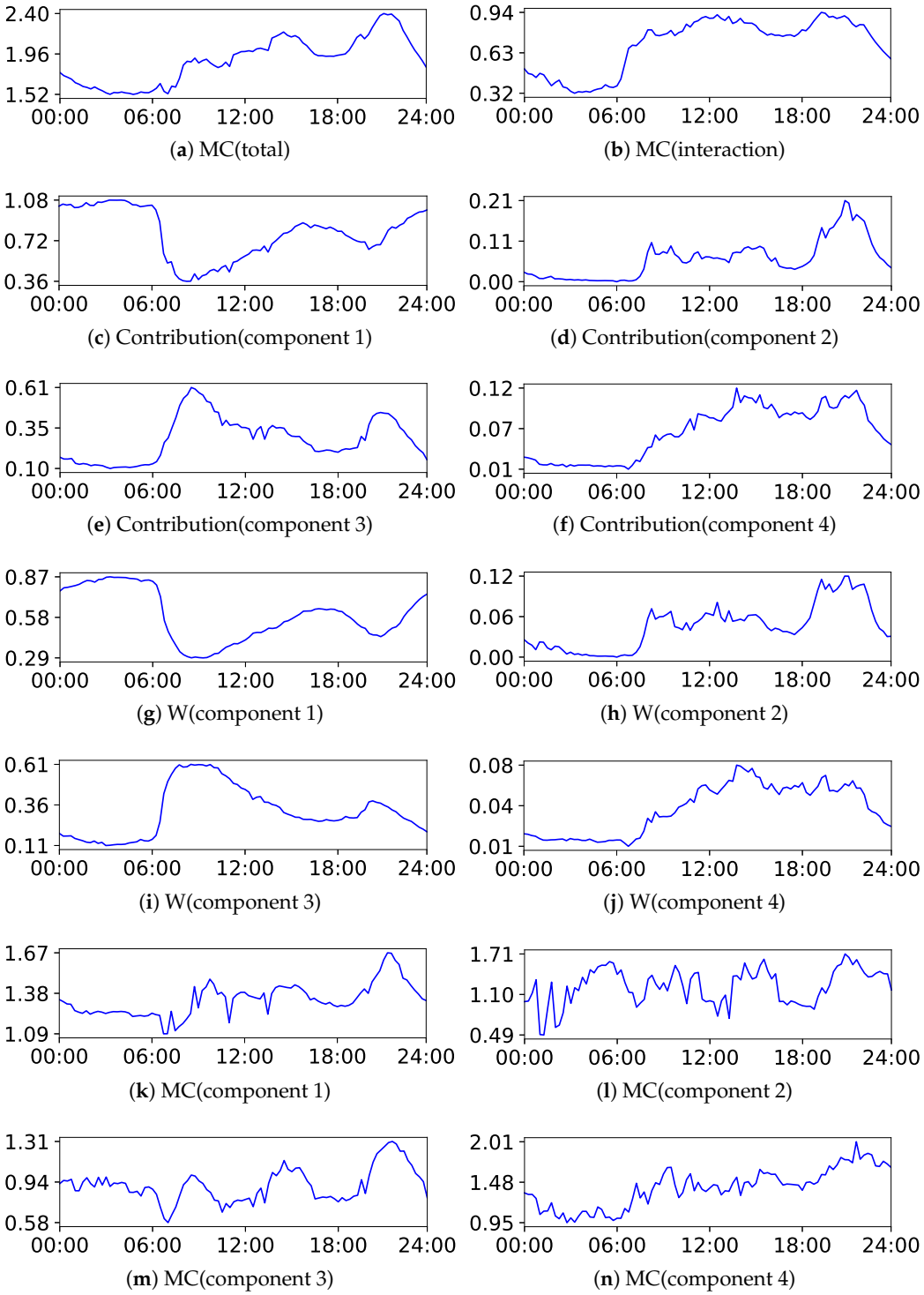


Figure 16. Plots of the decomposition of MC with NML in the house Dataset.

7. Conclusions

We proposed the concept of MC to measure the cluster size continuously in the mixture model. We first pointed out that the cluster size might not be equal to the mixture size when the mixture model had overlap or weight bias; then, we introduced MC as an extended concept of the cluster size considering the effects of them. We also presented methods to decompose the MC according to the mixture hierarchies, which helped us in extensively analyzing the substructures. Subsequently, we implemented the MC and its decomposition to the gradual clustering change detection problems. We conducted experiments to verify

that the MC effectively elucidates the clustering changes. In the artificial data experiments, MC found the clustering changes significantly earlier in the case where the overlap or weight bias was correctly estimated. In the real data experiments, MC expressed the gradual changes better than K because it discerned the significant and insignificant changes and smoothly connected the discrete changes in K. We also found that the MC took similar values for each model selection method; it indicates that the estimated clustering structures are alike under the concept of MC. Moreover, its decomposition enabled us to evaluate the contents of changes.

Issues of the MC will be tackled in future study. For example, it does not capture the clustering structure well when the number of the components is underestimated; thus, we need to explore the model selection methods that are more compatible with MC. Also, we further need to study its theoretical aspects, such as convergence and methods for approximating the mutual information. Furthermore, we need to consider extending the concept of MC into other clustering approaches, e.g., considering co-clustering by relating non-diagonal blocks in co-clustering and cluster overlaps in finite mixture models.

Author Contributions: Conceptualization, S.K. and K.Y.; methodology, S.K. and K.Y.; software, S.K.; validation, S.K. and K.Y.; formal analysis, S.K.; investigation, S.K. and K.Y; resources, S.K.; data curation, S.K.; writing—original draft preparation, S.K. and K.Y.; writing—review and editing, S.K. and K.Y.; visualization, S.K.; supervision, K.Y.; project administration, K.Y.; funding acquisition, K.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by JST KAKENHI JP19H01114.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Proof of the Basic Properties

We present a proof of Propositions 1–4.

Appendix A.1. Proof of Proposition 1

We can directly calculate as follows:

$$\begin{aligned}
&\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n};\{w_n\}_n\Big)\\
&= -\sum_{k=1}^{K}\left(\frac{\sum_n w_n\gamma_k}{\sum_{n'} w_{n'}}\right)\log\left(\frac{\sum_n w_n\gamma_k}{\sum_{n'} w_{n'}}\right)+\frac{1}{\sum_{n'} w_{n'}}\sum_{n=1}^{N} w_n\sum_{k=1}^{K}\gamma_k\log\gamma_k\\
&= -\sum_{k=1}^{K}\gamma_k\log\gamma_k+\sum_{k=1}^{K}\gamma_k\log\gamma_k\\
&= 0.
\end{aligned}$$

Appendix A.2. Proof of Proposition 2

Because $x\log x = 0$ when $x = 0$ or 1, $\widetilde{H}(Z|X)$ becomes 0 when the components are entirely separate. Thus, the proposition hols.

Appendix A.3. Proof of Proposition 3

It is immediate that

$$\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n};\{w_n\}_n\Big) = \widetilde{H}(Z)-\widetilde{H}(Z|X)\overset{(a)}{\leq}\widetilde{H}(Z)\overset{(b)}{\leq}\log K.$$

The equality of (a) holds only if the components are entirely separate and the equality of (b) holds only if the components are balanced. Thus, MC equals $\log K$ only if the components are entirely separate and balanced.

Also, by applying the Jensen's inequality to $x \mapsto -x \log x$, we obtain that

$$-\sum_{k=1}^{K}\left(\frac{\sum_n w_n\gamma_k(x_n)}{\sum_{n'} w_{n'}}\right)\log\left(\frac{\sum_n w_n\gamma_k(x_n)}{\sum_{n'} w_{n'}}\right) \geq -\sum_{k=1}^{K}\frac{1}{\sum_{n'} w_{n'}}\sum_{n=1}^{N} w_n\gamma_k(x_n)\log\gamma_k(x_n),$$

which is equivalent to that $\mathrm{MC}(\{\gamma_k(x_n)\}_{k,n}, \{w_n\}_n) \geq 0$. The equality holds only if the components entirely overlap.

Appendix A.4. Proof of Proposition 4

By applying Theorem 1 to partition $I_1 \cup \cdots \cup I_L$ into each set, we can calculate that

$$\begin{aligned}
&\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}; \{w_n\}_n\Big)\\
&= \mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k\in I_1\cup\cdots\cup I_L,n}; \{w_n\}_n\Big)\\
&= \mathrm{MC}\left(\left\{\sum_{k\in I_l}\gamma_k(x_n)\right\}_{l=1,\ldots,L,n}; \{w_n\}_n\right)\\
&\quad + \sum_{l=1}^{L}\left(\sum_{k\in I_l}\widetilde{\rho}_l\right)\mathrm{MC}\left(\left\{\frac{\gamma_k(x_n)}{\sum_{k'\in I_l}\gamma_{k'}(x_n)}\right\}_{k\in I_l,n}; \left\{w_n\sum_{k\in I_l}\gamma_k(x_n)\right\}_n\right)\\
&= \mathrm{MC}\Big(\left\{\gamma_k^0(x_n)\right\}_{k,n}; \{w_n\}_n\Big)\\
&\quad + \sum_{l=1}^{L}\left(\sum_{k\in I_l}\widetilde{\rho}_l\right)\mathrm{MC}\left(\left\{\frac{\rho_k}{\sum_{k'\in I_l}\rho_{k'}}\right\}_{k\in I_l,n}; \left\{w_n\sum_{k\in I_l}\gamma_k(x_n)\right\}_n\right)\\
&= \mathrm{MC}\Big(\left\{\gamma_k^0(x_n)\right\}_{k,n}; \{w_n\}_n\Big) \quad (\because \text{Proposition 1}).
\end{aligned}$$

Appendix B. Proof of the Decomposition Property

We present a proof of Theorem 1. Let

$$\widetilde{\rho}_k^{(l)} = \frac{\sum_n w_n^{(l)}\gamma_k^{(l)}(x_n)}{\sum_{n'} w_{n'}^{(l)}} = \frac{\sum_n w_n Q_k^{(l)}\gamma_k(x_n)}{\sum_{n'} w_{n'}\sum_{k'} Q_{k'}^{(l)}\gamma_{k'}(x_{n'})}.$$

Then, we can calculate as

$$\begin{aligned}
&\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}; \{w_n\}_n\Big) - \mathrm{MC}\left(\left\{\sum_{k=1}^{K} Q_k^{(l)}\gamma_k(x_n)\right\}_{l,n}; \{w_n\}_n\right)\\
&= \sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{l=1}^{L}\frac{w_n Q_k^{(l)}\gamma_k(x_n)}{\sum_{n'} w_{n'}}\log\left(\frac{\gamma_k(x_n)}{\frac{\sum_{n'} w_{n'}\gamma_k(x_{n'})}{\sum_{n'} w_{n'}}}\cdot\frac{\frac{\sum_{n'} w_{n'}\sum_{k'} Q_{k'}^{(l)}\gamma_{k'}(x_{n'})}{\sum_{n'} w_{n'}}}{\sum_{k'} Q_{k'}^{(l)}\gamma_{k'}(x_n)}\right)\\
&= \sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{l=1}^{L}\frac{w_n Q_k^{(l)}\gamma_k(x_n)}{\sum_{n'} w_{n'}}\log\left(\frac{Q_k^{(l)}\gamma_k(x_n)}{\sum_{k'} Q_{k'}^{(l)}\gamma_{k'}(x_n)}\cdot\frac{\sum_{n'} w_{n'}\sum_{k'} Q_{k'}^{(l)}\gamma_{k'}(x_{n'})}{\sum_{n'} w_{n'} Q_k^{(l)}\gamma_k(x_{n'})}\right)\\
&= \sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{l=1}^{L} W_l\cdot\frac{w_n Q_k^{(l)}\gamma_k(x_n)}{\sum_{n'} w_{n'}\sum_{k'} Q_{k'}^{(l)}\gamma_{k'}(x_{n'})}\log\left(\frac{\gamma_k^{(l)}(x_n)}{\widetilde{\rho}_k^{(l)}}\right)\\
&= \sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{l=1}^{L} W_l\cdot\frac{w_n^{(l)}\gamma_k^{(l)}(x_n)}{\sum_{n'} w_{n'}^{(l)}}\log\left(\frac{\gamma_k^{(l)}(x_n)}{\widetilde{\rho}_k^{(l)}}\right)\\
&= \sum_{l=1}^{L} W_l\cdot\left(\left\{\gamma_k^{(l)}(x_n)\right\}_{k,n}; \left\{w_n^{(l)}\right\}_n\right).
\end{aligned}$$

Appendix C. Proof of the Consistency

We present a proof of Theorem 2. We use the following notations for convenience:

$$[\gamma_1(x_n)|\dots|\gamma_K(x_n)] \coloneqq \{\gamma_k(x_n)\}_{k,n},$$
$$I_0 \coloneqq I_1 \cup \dots \cup I_{K^\star}.$$

First, applying Theorem 1 repeatedly, we decompose the entire MC as follows:

$$\begin{aligned}
&\mathrm{MC}\Big(\{\gamma_k(x_n)\}_{k,n}\Big)\\
&= \mathrm{MC}\left(\left[\sum_{k\in I_0}\gamma_k(x_n)\,\middle|\,\sum_{k\in I_\infty}\gamma_k(x_n)\right]\right)\\
&\quad + \widetilde{\rho}_\infty \mathrm{MC}\left(\left\{\frac{\gamma_k(x_n)}{\sum_{k'\in I_\infty}\gamma_{k'}(x_n)}\right\}_{k\in I_\infty,n}; \left\{\sum_{k\in I_\infty}\gamma_k(x_n)\right\}\right)\\
&\quad + (1-\widetilde{\rho}_\infty)\mathrm{MC}\left(\left\{\frac{\gamma_k(x_n)}{\sum_{k'\in I_0}\gamma_{k'}(x_n)}\right\}_{k\in I_0,n}; \left\{\sum_{k\in I_0}\gamma_k(x_n)\right\}\right)\\
&= \mathrm{MC}\left(\left[\sum_{k\in I_0}\gamma_k(x_n)\,\middle|\,\sum_{k\in I_\infty}\gamma_k(x_n)\right]\right)\\
&\quad + \widetilde{\rho}_\infty \mathrm{MC}\left(\left\{\frac{\gamma_k(x_n)}{\sum_{k'\in I_\infty}\gamma_{k'}(x_n)}\right\}_{k\in I_\infty,n}; \left\{\sum_{k\in I_\infty}\gamma_k(x_n)\right\}\right)\\
&\quad + (1-\widetilde{\rho}_\infty)\mathrm{MC}\left(\left\{\frac{\sum_{k\in I_l}\gamma_k(x_n)}{\sum_{k'\in I_0}\gamma_{k'}(x_n)}\right\}_{l=1,\dots,K^\star}; \left\{\sum_{k\in I_0}\gamma_k(x_n)\right\}\right)\\
&\quad + \sum_{l=1}^{K^\star}\left(\sum_{k\in I_l}\widetilde{\rho}_k\right)\mathrm{MC}\left(\left\{\frac{\gamma_k(x_n)}{\sum_{k'\in I_l}\gamma_{k'}(x_n)}\right\}_{k\in I_l}; \left\{\sum_{k\in I_l}\gamma_k(x_n)\right\}\right)\\
&= \mathrm{MC}\left(\left[\frac{\sum_{l=1}^{K^\star} r_l h_l(x_n)}{f(x_n)}\,\middle|\,\frac{\sum_{k\in I_\infty}\rho_k g_k(x_n)}{f(x_n)}\right]\right)\\
&\quad + \widetilde{\rho}_\infty \mathrm{MC}\left(\left\{\frac{\rho_k(x_n)}{\sum_{k'\in I_\infty}\rho_{k'}(x_n)}\right\}_{k\in I_\infty}; \left\{\sum_{k\in I_\infty}\gamma_k(x_n)\right\}\right)\\
&\quad + (1-\widetilde{\rho}_\infty)\mathrm{MC}\left(\left\{\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star} r_{l'} h_{l'}(x_n)}\right\}_{l=1,\dots,K^\star}; \left\{\sum_{k\in I_0}\gamma_k(x_n)\right\}\right)\\
&\quad + \sum_{l=1}^{K^\star}\left(\sum_{k\in I_l}\widetilde{\rho}_k\right)\mathrm{MC}\left(\left\{\frac{s_k g_k(x_n)}{h_l(x_n)}\right\}_{k\in I_l}; \left\{\sum_{k\in I_l}\gamma_k(x_n)\right\}\right). \qquad \text{(A1)}
\end{aligned}$$

The terms in (A1) correspond to the four quantities (a), …, (d) referred to in Section 4.3. Then, it is sufficient to show that

$$\mathrm{MC}\left(\left[\frac{\sum_{l=1}^{K^\star} r_l h_l(x_n)}{f(x_n)}\,\middle|\,\frac{\sum_{k\in I_\infty}\rho_k g_k(x_n)}{f(x_n)}\right]\right) \to 0,$$
$$\widetilde{\rho}_\infty \mathrm{MC}\left(\left\{\frac{\rho_k(x_n)}{\sum_{k'\in I_\infty}\rho_{k'}(x_n)}\right\}_{k\in I_\infty}; \left\{\sum_{k\in I_\infty}\gamma_k(x_n)\right\}\right) \to 0,$$
$$(1-\widetilde{\rho}_\infty)\mathrm{MC}\left(\left\{\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star} r_{l'} h_{l'}(x_n)}\right\}_{l=1,\dots,K^\star}; \left\{\sum_{k\in I_0}\gamma_k(x_n)\right\}\right) \to \mathrm{MC}\Big(\{\gamma_k^\star(x_n)\}_{k,n}\Big),$$
$$\sum_{l=1}^{K^\star}\left(\sum_{k\in I_l}\widetilde{\rho}_k\right)\mathrm{MC}\left(\left\{\frac{s_k g_k(x_n)}{h_l(x_n)}\right\}_{k\in I_l}; \left\{\sum_{k\in I_l}\gamma_k(x_n)\right\}\right) \to 0.$$

Step 1

First, we show that

$$\mathrm{MC}\left(\left[\frac{\sum_{l=1}^{K^\star} r_l h_l(x_n)}{f(x_n)} \,\middle|\, \frac{\sum_{k\in I_\infty} \rho_k g_k(x_n)}{f(x_n)}\right]\right) = \mathrm{O}_P(\widetilde{\rho}_\infty(-\log\widetilde{\rho}_\infty)).$$

Using Proposition 3, it is easily shown as follows:

$$\begin{aligned}\mathrm{MC}\left(\left[\frac{\sum_{l=1}^{K^\star} r_l h_l(x_n)}{f(x_n)} \,\middle|\, \frac{\sum_{k\in I_\infty} \rho_k g_k(x_n)}{f(x_n)}\right]\right) &\leq -\widetilde{\rho}_\infty \log\widetilde{\rho}_\infty - (1-\widetilde{\rho}_\infty)\log(1-\widetilde{\rho}_\infty)\\ &= \mathrm{O}_P(\widetilde{\rho}_\infty(-\log\widetilde{\rho}_\infty)).\end{aligned}$$

Step 2

Second, we show that

$$\widetilde{\rho}_\infty \mathrm{MC}\left(\left\{\frac{\rho_k(x_n)}{\sum_{k'\in I_\infty}\rho_{k'}(x_n)}\right\}_{k\in I_\infty}; \left\{\sum_{k\in I_\infty}\gamma_k(x_n)\right\}\right) = \mathrm{O}_P(\widetilde{\rho}_\infty).$$

It is also evident from Proposition 3:

$$\begin{aligned}\widetilde{\rho}_\infty \mathrm{MC}\left(\left\{\frac{\rho_k(x_n)}{\sum_{k'\in I_\infty}\rho_{k'}(x_n)}\right\}_{k\in I_\infty}; \left\{\sum_{k\in I_\infty}\gamma_k(x_n)\right\}\right) &\leq \widetilde{\rho}_\infty \log(K - i_{K^\star})\\ &= \mathrm{O}_P(\widetilde{\rho}_\infty).\end{aligned}$$

Step 3

Third, we show that

$$\begin{aligned}&(1-\widetilde{\rho}_\infty)\mathrm{MC}\left(\left\{\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star} r_{l'} h_{l'}(x_n)}\right\}_{l=1,\ldots,K^\star}; \left\{\sum_{k\in I_0}\gamma_k(x_n)\right\}\right)\\ &= \mathrm{MC}\Big(\{\gamma_k^\star(x_n)\}_{k,n}\Big) + \mathrm{O}_P\left(\|\phi-\phi^\star\| + \sum_{l=1}^{K^\star}|\widetilde{r}_l - \rho_l^\star| + \widetilde{\rho}_\infty\right).\end{aligned}$$

To this end, we further decompose the left hand as

$$\begin{aligned}&(1-\widetilde{\rho}_\infty)\mathrm{MC}\left(\left\{\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star} r_{l'} h_{l'}(x_n)}\right\}_{l=1,\ldots,K^\star}; \left\{\sum_{k\in I_0}\gamma_k(x_n)\right\}\right)\\ &= -(1-\widetilde{\rho}_\infty)\sum_{l=1}^{K^\star}\frac{\widetilde{r}_l}{1-\widetilde{\rho}_\infty}\log\left(\frac{\widetilde{r}_l}{1-\widetilde{\rho}_\infty}\right)\\ &\quad + \frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\left(\sum_{k\in I_0}\gamma_k(x_n)\right)\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star} r_{l'} h_{l'}(x_n)}\log\left(\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star} r_{l'} h_{l'}(x_n)}\right);\end{aligned}$$

they correspond to the unconditional and conditional entropy of the latent variables, respectively. On the other hand, the true MC is defined as

$$-\sum_{l=1}^{K^\star}\widetilde{\rho}_l^\star\log\widetilde{\rho}_l^\star + \frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\gamma_l^\star(x_n)\log\gamma_l^\star(x_n).$$

Then, it is sufficient to show that

$$-(1-\widetilde{\rho}_\infty)\sum_{l=1}^{K^\star}\frac{\widetilde{r}_l}{1-\widetilde{\rho}_\infty}\log\left(\frac{\widetilde{r}_l}{1-\widetilde{\rho}_\infty}\right)\to-\sum_{l=1}^{K^\star}\widetilde{\rho}_l^\star\log\widetilde{\rho}_l^\star,$$

$$\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\left(\sum_{k\in I_0}\gamma_k(x_n)\right)\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star}r_{l'}h_{l'}(x_n)}\log\left(\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star}r_{l'}h_{l'}(x_n)}\right)\to\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\gamma_l^\star(x_n)\log\gamma_l^\star(x_n).$$

First, we show that

$$\begin{aligned}&-(1-\widetilde{\rho}_\infty)\sum_{l=1}^{K^\star}\frac{\widetilde{r}_l}{1-\widetilde{\rho}_\infty}\log\left(\frac{\widetilde{r}_l}{1-\widetilde{\rho}_\infty}\right)\\&=-\sum_{l=1}^{K^\star}\widetilde{\rho}_l^\star\log\widetilde{\rho}_l^\star+\mathrm{O}_P\left(\sum_{l=1}^{K^\star}|\widetilde{r}_l-\rho_l^\star|+\widetilde{\rho}_\infty+\frac{1}{\sqrt{N}}\right).\end{aligned}$$

Indeed, by the mean-value theorem, there exist $r_1^{\mathrm{m}},\dots,r_{K^\star}^{\mathrm{m}}$ between $\widetilde{r}_1,\dots,\widetilde{r}_{K^\star}$ and $\rho_1^\star,\dots,\rho_{K^\star}^\star$ and ρ_∞^{m} between 0 and $\widetilde{\rho}_\infty$ such that

$$\sum_{l=1}^{K^\star}\widetilde{r}_l\log\widetilde{r}_l=\sum_{l=1}^{K^\star}\rho_l^\star\log\rho_l^\star+\sum_{l=1}^{K^\star}(1+\log r_l^{\mathrm{m}})(\widetilde{r}_l-\rho_l^\star)\quad(l=1,\dots,K^\star),\tag{A2}$$

$$(1-\widetilde{\rho}_\infty)\log(1-\widetilde{\rho}_\infty)=(1-\log(1-\rho_\infty^{\mathrm{m}}))\widetilde{\rho}_\infty.\tag{A3}$$

Also, from assumption (C3), if N is sufficiently large, $\log r_l^{\mathrm{m}}$ and $\log(1-\rho_\infty^{\mathrm{m}})$ are finite because r_l^{m} and ρ_∞^{m} become arbitrarily close to $\rho_l^\star(>0)$ and 0, respectively. Similarly, there exist $\rho_1^{\mathrm{m}},\dots,\rho_{K^\star}^{\mathrm{m}}$ between $\widetilde{\rho}_1^\star,\dots,\widetilde{\rho}_{K^\star}^\star$ and $\rho_1^\star,\dots,\rho_{K^\star}^\star$ such that

$$\sum_{l=1}^{K^\star}\widetilde{\rho}_l^\star\log\widetilde{\rho}_l^\star=\sum_{l=1}^{K^\star}\rho_l^\star\log\rho_l^\star+\sum_{l=1}^{K^\star}(1+\log\rho_l^{\mathrm{m}})(\widetilde{\rho}_l^\star-\rho_l^\star)\quad(l=1,\dots,K^\star).\tag{A4}$$

Also, from the central limit theorem, $\widetilde{\rho}_l^\star$ converges to $\rho_k^\star$ at the speed of $\mathrm{O}_P(1/\sqrt{N})$.

Using (A2)–(A4), we can calculate as

$$\begin{aligned}&-(1-\widetilde{\rho}_\infty)\sum_{l=1}^{K^\star}\frac{\widetilde{r}_l}{1-\widetilde{\rho}_\infty}\log\left(\frac{\widetilde{r}_l}{1-\widetilde{\rho}_\infty}\right)\\&=-\sum_{l=1}^{K^\star}\widetilde{r}_l\log\widetilde{r}_l+K^\star(1-\widetilde{\rho}_\infty)\log(1-\widetilde{\rho}_\infty)\\&=-\sum_{l=1}^{K^\star}\rho_l^\star\log\rho_l^\star+\mathrm{O}_P\left(\sum_{l=1}^{K^\star}|\widetilde{r}_l-\rho_l^\star|+\widetilde{\rho}_\infty\right)\\&=-\sum_{l=1}^{K^\star}\widetilde{\rho}_l^\star\log\widetilde{\rho}_l^\star+\mathrm{O}_P\left(\sum_{l=1}^{K^\star}|\widetilde{r}_l-\rho_l^\star|+\widetilde{\rho}_\infty+\frac{1}{\sqrt{N}}\right).\end{aligned}$$

Next, we show that

$$\begin{aligned}&\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\left(\sum_{k\in I_0}\gamma_k(x_n)\right)\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star}r_{l'}h_{l'}(x_n)}\log\left(\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star}r_{l'}h_{l'}(x_n)}\right)\\&=\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\gamma_l^\star(x_n)\log\gamma_l^\star(x_n)+\mathrm{O}_P(\|\phi-\phi^\star\|+\widetilde{\rho}_\infty).\end{aligned}$$

We first define the following functions for $l=1,\dots,K^\star$:

$$F_l(\phi,\psi,x):=\frac{r_l h_l(x)}{\sum_{l'=1}^{K^\star}r_{l'}h_{l'}(x)}\log\left(\frac{r_l h_l(x)}{\sum_{l'=1}^{K^\star}r_{l'}h_{l'}(x)}\right).$$

When $\phi=\phi^\star$, $F_l(\phi^\star,\psi,x)$ is calculated as $\gamma_l^\star(x)\log\gamma_l^\star(x)$; it is independent of ψ. In this case, we omit ψ and write $F_l(\phi^\star,\psi,x)$ as $F_l(\phi^\star,x)$. Applying the mean-value theorem to this function, there exists a function $0<\tau(x)<1$ such that

$$F_l(\phi,\psi,x)=F_l(\phi^\star,x)+\left(\frac{\partial F_l(\phi^\star+t(x)(\phi-\phi^\star),\psi,x)}{\partial\phi}\right)^\top(\phi-\phi^\star).\tag{A5}$$

Moreover, it can be shown that $\partial F_l(\phi^\star + t(x)(\phi - \phi^\star), \psi, x)/\partial\phi = \mathrm{O}_P(1)$ uniformly with $x, \tau(x)$, and ψ. Indeed, letting

$$\phi^{\mathrm{m}} := \left(\{\theta_k^{\mathrm{m}}\}_{k=1}^{i_{K^\star}}, \{r_l^{\mathrm{m}}\}_{l=1}^{K^\star}, \{\rho_k^{\mathrm{m}}\}_{k=i_{K^\star}+1}^{K}\right) := \phi^\star + t(x)(\phi - \phi^\star),$$
$$h_l^{\mathrm{m}}(x) := \sum_{k\in I_l} s_k g_k(x|\theta_k^{\mathrm{m}}),$$
$$\gamma_l^{\mathrm{m}}(x) := \frac{r_l^{\mathrm{m}} h_l^{\mathrm{m}}(x)}{\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)},$$

derivative of each parameter in ϕ is bounded as follows:

$$\begin{aligned}
&\left|\frac{\partial F_l(\phi^{\mathrm{m}}, \psi, x)}{\partial r_l}\right| \\
&= \left|\left(\frac{h_l^{\mathrm{m}}(x)}{\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)} - \frac{r_l^{\mathrm{m}}\left(h_l^{\mathrm{m}}(x)\right)^2}{\left(\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)\right)^2}\right)\left(1 + \log\left(\frac{r_l^{\mathrm{m}} h_l^{\mathrm{m}}(x)}{\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)}\right)\right)\right| \\
&\le \frac{\gamma_l^{\mathrm{m}}(x)}{r_l^{\mathrm{m}}} + \frac{\left(\gamma_l^{\mathrm{m}}(x)\right)^2}{r_l^{\mathrm{m}}} + \frac{1}{r_l^{\mathrm{m}}}|\gamma_l^{\mathrm{m}}(x)\log\gamma_l^{\mathrm{m}}(x)| + \frac{1}{r_l^{\mathrm{m}}}\left|(\gamma_l^{\mathrm{m}}(x))^2\log\gamma_l^{\mathrm{m}}(x)\right| \\
&\le \frac{4}{r_l^{\mathrm{m}}}, \\
&\left|\frac{\partial F_l(\phi^{\mathrm{m}}, \psi, x)}{\partial r_m}\right| \quad (m \neq l) \\
&= \left|\frac{r_l^{\mathrm{m}} h_l^{\mathrm{m}}(x) h_m^{\mathrm{m}}(x)}{\left(\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)\right)^2}\left(1 + \log\left(\frac{r_l^{\mathrm{m}} h_l^{\mathrm{m}}(x)}{\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)}\right)\right)\right| \\
&\le \frac{\gamma_m^{\mathrm{m}}(x)}{r_m^{\mathrm{m}}}|\gamma_l^{\mathrm{m}}(x)\log\gamma_l^{\mathrm{m}}(x)| \\
&\le \frac{1}{r_m^{\mathrm{m}}}, \\
&\left\|\frac{\partial F_l(\phi^{\mathrm{m}}, \psi, x)}{\partial \theta_k}\right\| \quad (k \in I_l) \\
&= \left\|\left(\frac{r_l^{\mathrm{m}}}{\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)}\frac{\partial h_l^{\mathrm{m}}(x)}{\partial\theta_k} - \frac{\left(r_l^{\mathrm{m}}\right)^2 h_l^{\mathrm{m}}(x)}{\left(\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)\right)^2}\frac{\partial h_l^{\mathrm{m}}(x)}{\partial\theta_k}\right)\right. \\
&\qquad \left.\times\left(1 + \log\left(\frac{r_l^{\mathrm{m}} h_l^{\mathrm{m}}(x)}{\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)}\right)\right)\right\| \\
&\le \left(\gamma_l^{\mathrm{m}}(x) + (\gamma_l^{\mathrm{m}}(x))^2 + |\gamma_l^{\mathrm{m}}(x)\log\gamma_l^{\mathrm{m}}(x)| + \left|(\gamma_l^{\mathrm{m}}(x))^2\log\gamma_l^{\mathrm{m}}(x)\right|\right)\left\|\frac{1}{h_l^{\mathrm{m}}(x)}\frac{\partial h_l^{\mathrm{m}}(x)}{\partial\theta_k}\right\| \\
&\le 4\left\|\sup_{\theta\in\Theta_\epsilon}\left(\frac{1}{g(x|\theta)}\frac{\partial g(x|\theta)}{\partial\theta_k}\right)\right\|, \\
&\left\|\frac{\partial F_l(\phi^{\mathrm{m}}, \psi, x)}{\partial \theta_k}\right\| \quad (k \in I_m, m \neq l) \\
&= \left\|\frac{r_l^{\mathrm{m}} r_m^{\mathrm{m}} h_l^{\mathrm{m}}(x)}{\left(\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)\right)^2}\frac{\partial h_m^{\mathrm{m}}(x)}{\partial\theta_k}\left(1 + \log\left(\frac{r_l^{\mathrm{m}} h_l^{\mathrm{m}}(x)}{\sum_{l'=1}^{K^\star} r_{l'}^{\mathrm{m}} h_{l'}^{\mathrm{m}}(x)}\right)\right)\right\| \\
&\le (\gamma_m^{\mathrm{m}}(x)\gamma_l^{\mathrm{m}}(x) + \gamma_m^{\mathrm{m}}(x)|\gamma_l^{\mathrm{m}}(x)\log\gamma_l^{\mathrm{m}}(x)|)\left\|\frac{\partial h_m^{\mathrm{m}}(x)}{\partial\theta_k}\right\| \\
&\le 2\left\|\sup_{\theta\in\Theta_\epsilon}\left(\frac{1}{g(x|\theta)}\frac{\partial g(x|\theta)}{\partial\theta_k}\right)\right\|, \\
&\left|\frac{\partial F_l(\phi^{\mathrm{m}}, \psi, x)}{\partial \rho_k}\right| = 0 \quad (k \in I_\infty),
\end{aligned}$$

where

$$\Theta_\epsilon := \left\{ \theta \,\middle|\, {}^\exists l \in \{1, \dots, K^\star\}, \|\theta - \theta_l^\star\| < \|\phi - \phi^\star\| \right\}.$$

They are all finite because r_l^{m} become arbitrarily close to $\rho_l^\star$ and Θ_ϵ become arbitrarily smaller as $N \to \infty$; condition (C1) is also employed.

Using (A5), we can calculate as

$$\begin{aligned}
&\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\left(\sum_{k\in I_l}\gamma_k(x_n)\right)\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star} r_{l'} h_{l'}(x_n)}\log\left(\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star} r_{l'} h_{l'}(x_n)}\right)\\
&=\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\left(1-\sum_{k\in I_\infty}\gamma_k(x_n)\right)\left(F_l(\phi^\star, x_n)+\left(\frac{\partial F_l(\phi^{\mathrm{m}}, \psi, x_n)}{\partial \phi}\right)^\top(\phi-\phi^\star)\right)\\
&=\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\left(1-\sum_{k\in I_\infty}\gamma_k(x_n)\right)\left(\gamma_l^\star(x)\log\gamma_l^\star(x)+\left(\frac{\partial F_l(\phi^{\mathrm{m}}, \psi, x_n)}{\partial \phi}\right)^\top(\phi-\phi^\star)\right).
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\left|\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\left(\sum_{k\in I_0}\gamma_k(x_n)\right)\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star} r_{l'} h_{l'}(x_n)}\log\left(\frac{r_l h_l(x_n)}{\sum_{l'=1}^{K^\star} r_{l'} h_{l'}(x_n)}\right)\right.\\
&\qquad\left.-\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\gamma_l^\star(x_n)\log\gamma_l^\star(x_n)\right|\\
&\le\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\left|\left(\frac{\partial F_l(\phi^{\mathrm{m}}, \psi, x_n)}{\partial \phi}\right)^\top(\phi-\phi^\star)\right|\\
&\qquad+\frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{K^\star}\left(\sum_{k\in I_\infty}\gamma_k(x_n)\right)|\gamma_l^\star(x_n)\log\gamma_l^\star(x_n)|\\
&\le\sum_{l=1}^{K^\star}\sup_{x_n,\psi}\left(\left\|\frac{\partial F_l(\phi^{\mathrm{m}}, \psi, x_n)}{\partial \phi}\right\|\|\phi-\phi^\star\|\right)+3K^\star\widetilde{\rho}_\infty\\
&=\mathrm{O}_P(\|\phi-\phi^\star\|+\widetilde{\rho}_\infty).
\end{aligned}$$

Step 4

Finally, we show that

$$\sum_{l=1}^{K^\star}\left(\sum_{k\in I_l}\widetilde{\rho}_k\right)\mathrm{MC}\left(\left\{\frac{s_k g_k(x_n)}{h_l(x_n)}\right\}_{k\in I_l}; \left\{\sum_{k\in I_l}\gamma_k(x_n)\right\}\right)=\mathrm{O}_P(\|\phi-\phi^\star\|)$$

for every $l = 1, \dots, K^\star$.

To this end, we write the left hand as $\mathrm{G}(\{\theta_k\}_{k\in I_l})$ and consider it as a function of $\{\theta_k\}_{k\in I_l}$. Then, for all other parameters, $\mathrm{G}(\{\theta_k^\star\}_{k\in I_l}) = 0$. Also, the derivative of G by θ_l is $\mathrm{O}_P(1)$ as $N \to \infty$. Indeed, it can be rewritten as

$$\begin{aligned}
&\mathrm{G}\Big(\{\theta_k\}_{k\in I_l}\Big)\\
&=\frac{1}{N}\sum_{n=1}^{N}\sum_{k\in I_l}\frac{r_l s_k g_k(x_n)}{f(x_n)}\log\left(\frac{s_k g_k(x_n)}{h_l(x_n)}\right)\\
&\quad-\frac{1}{N}\sum_{k\in I_l}\left(\sum_{n=1}^{N}\frac{r_l s_k g_k(x_n)}{f(x_n)}\right)\log\left(\sum_{n'=1}^{N}\frac{r_l s_k g_k(x_{n'})}{f(x_{n'})}\right)\\
&\quad+\frac{1}{N}\sum_{k\in I_l}\left(\sum_{n=1}^{N}\frac{r_l s_k g_k(x_n)}{f(x_n)}\right)\log\left(\sum_{n'=1}^{N}\frac{r_l h_l(x_{n'})}{f(x_{n'})}\right).
\end{aligned}$$

Also, we define the posterior probabilities within h_l as

$$\gamma_k^{(l)}(x) := \frac{s_k g_k(x)}{h_l(x)} \quad (k \in I_l).$$

Then, the derivatives are bounded as

$$
\begin{aligned}
&\left\| \frac{\partial \mathrm{G}\left(\{\theta_k\}_{k\in I_l}\right)}{\partial \theta_m} \right\| \quad (m \in I_l) \\
&= \Bigg\| \frac{1}{N}\sum_{n=1}^{N}\sum_{k\in I_l}\left(\frac{r_l s_k}{f(x_n)}\frac{\partial g_k(x_n)}{\partial \theta_m} - \frac{r_l^2 s_k s_m g_k(x_n)}{(f(x_n))^2}\frac{\partial g_m(x_n)}{\partial \theta_m}\right) \\
&\quad \times \left(\log\left(\frac{s_k g_k(x_n)}{h_l(x_n)}\right) - \log\left(\sum_{n'=1}^{N}\frac{r_l s_k g_k(x_{n'})}{f(x_{n'})}\right) + \log\left(\sum_{n''=1}^{N}\frac{r_l h_l(x_{n''})}{f(x_{n''})}\right)\right) \\
&\quad + \frac{1}{N}\sum_{n=1}^{N}\sum_{k\in I_l}\left(\frac{r_l s_k}{f(x_n)}\frac{\partial g_k(x_n)}{\partial \theta_m} - \frac{r_l s_k s_m g_k(x_n)}{f(x_n)h_l(x_n)}\frac{\partial g_m(x_n)}{\partial \theta_m}\right) \\
&\quad - \frac{1}{N}\sum_{k\in I_l}\sum_{n=1}^{N}\left(\frac{r_l s_k}{f(x_n)}\frac{\partial g_k(x_n)}{\partial \theta_m} - \frac{r_l^2 s_k s_m g_k(x_n)}{(f(x_n))^2}\frac{\partial g_m(x_n)}{\partial \theta_m}\right) \\
&\quad + \frac{1}{N}\sum_{k\in I_l}\frac{\sum_{n'=1}^{N}\frac{r_l s_k g_k(x_{n'})}{f(x_{n'})}}{\sum_{n''=1}^{N}\frac{h_l(x_{n''})}{f(x_{n''})}}\sum_{n=1}^{N}\left(\frac{r_l s_m}{f(x_n)}\frac{\partial g_k(x_n)}{\partial \theta_m} - \frac{r_l s_m h(x_n)}{(f(x_n))^2}\frac{\partial g_m(x_n)}{\partial \theta_m}\right)\Bigg\| \\
&\le \frac{1}{N}\sum_{n=1}^{N}\sum_{k\in I_l}\left|\gamma_k^{(l)}(x_n)\log\gamma_k^{(l)}(x_n)\right|\left\|\frac{1}{g_k(x_n)}\frac{\partial g_k(x_n)}{\partial \theta_m}\right\| \\
&\quad + \frac{1}{N}\sum_{n=1}^{N}\sum_{k\in I_l}\left|\gamma_k^{(l)}(x_n)\gamma_m^{(l)}(x_n)\log\gamma_k^{(l)}(x_n)\right|\left\|\frac{1}{g_m(x_n)}\frac{\partial g_m(x_n)}{\partial \theta_m}\right\| \\
&\quad + \sum_{k\in I_l}\left[\frac{2}{N}\sum_{n=1}^{N}\frac{r_l h_l(x_n)}{f(x_n)}\sup_{\theta\in\Theta_\epsilon}\left\|\frac{1}{g(x_n|\theta)}\frac{\partial g(x_n|\theta)}{\partial \theta}\right\|\right. \\
&\quad \left.\times\left|\frac{\sum_{n'=1}^{N}\frac{r_l s_k g_k(x_{n'})}{f(x_{n'})}}{\sum_{n'=1}^{N}\frac{r_l h_l(x_{n'})}{f(x_{n'})}}\log\left(\frac{\sum_{n'=1}^{N}\frac{r_l s_k g_k(x_{n'})}{f(x_{n'})}}{\sum_{n'=1}^{N}\frac{r_l h_l(x_{n'})}{f(x_{n'})}}\right)\right|\right] \\
&\quad + \frac{2}{N}\sum_{n=1}^{N}\sum_{k\in I_l}\left|\gamma_k^{(l)}(x_n)\right|\left\|\frac{1}{g_k(x_n)}\frac{\partial g_k(x_n)}{\partial \theta_m}\right\| \\
&\quad + \frac{2}{N}\sum_{n=1}^{N}\sum_{k\in I_l}\left|\gamma_k^{(l)}(x_n)\gamma_m^{(l)}(x_n)\right|\left\|\frac{1}{g_m(x_n)}\frac{\partial g_m(x_n)}{\partial \theta_m}\right\| \\
&\le 8K\sup_{\theta\in\Theta_\epsilon}\left\|\frac{1}{g(x_n|\theta)}\frac{\partial g(x_n|\theta)}{\partial \theta}\right\| \\
&= \mathrm{O}_P(1),
\end{aligned}
$$

where, it is assumed that $\{\theta_k\}_{k\in I_l}$ are sufficiently close to $\theta_k^\star$, which holds if N is sufficiently large because of condition (C2).

Therefore, by the mean-value theorem, there exist $\{\theta_k^{\mathrm{m}}\}_{k\in I_l}$ such that

$$
\begin{aligned}
\mathrm{G}(\{\theta_k\}_{k\in I_l}) &= \sum_{k\in I_l}\left(\frac{\partial \mathrm{G}\left(\{\theta_k^{\mathrm{m}}\}_{k\in I_l}\right)}{\partial \theta_k}\right)^{\top}(\theta_k^{\mathrm{m}} - \theta_k^\star) \\
&= \mathrm{O}_P(\|\phi - \phi^\star\|),
\end{aligned}
$$

which concludes the proof.

References

1. McLachlan, G.J.; Peel, D. *Finite Mixture Models*; Wiley Series in Probability and Statistics: New York, NY, USA, 2000.
2. Fraley, C.; Raftery, A.E. How Many Clusters? Which Clustering Method? Answers via Model-based Cluster Analysis. *Comput. J.* **1998**, *41*, 578–588. [CrossRef]
3. Hennig, C. Methods for Merging Gaussian Mixture Components. *Adv. Data Anal. Classif.* **2010**, *4*, 3–34. [CrossRef]
4. Jiang, M.F.; Tseng, S.S.; Su, C.M. Two-phase Clustering Process for Outliers Detection. *Pattern Recognit. Lett.* **2001**, *22*, 691–700. [CrossRef]

5. He, Z.; Xu, X.; Deng, S. Discovering Cluster-based Local Outliers. *Pattern Recognit. Lett.* **2003**, *24*, 1641–1650. [CrossRef]
6. Gama, J.; Žliobaitė, I.; Bifet, A.; Pechenizkiy, M.; Bouchachia, A. A Survey On Concept Drift Adaptation. *ACM Comput. Surv.* **2014**, *46*, 1–37. [CrossRef]
7. Kyoya, S.; Yamanishi, K. Summarizing Finite Mixture Model with Overlapping Quantification. *Entropy* **2021**, *23*, 1503. [CrossRef] [PubMed]
8. Akaike, H. A New Look at the Statistical Model Identification. *IEEE Trans. Autom. Control* **1974**, *19*, 716–723. [CrossRef]
9. Schwarz, G. Estimating the Dimension of a Model. *Ann. Stat.* **1978**, *6*, 461–464. [CrossRef]
10. Rissanen, J. Modeling by Shortest Data Description. *Automatica* **1978**, *14*, 465–471. [CrossRef]
11. Biernacki, C.; Celeux, G.; Govaert, G. Assessing a Mixture Model for Clustering with the Integrated Completed Likelihood. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 719–725. [CrossRef]
12. Kontkanen, P.; Myllymäki, P.; Buntine, W.; Rissanen, J.; Tirri, H., An MDL Framework for Data Clustering. In *Advances in Minimum Description Length*; MIT Press: Cambridge, MA, USA, 2005; pp. 323–353.
13. Hirai, S.; Yamanishi, K. Efficient Computation of Normalized Maximum Likelihood Codes for Gaussian Mixture Models with Its Applications to Clustering. *IEEE Trans. Inf. Theory* **2019**, *59*, 7718–7727; Erratum in *IEEE Trans. Inf. Theory* **2019**, *65*, 6827–6828. [CrossRef]
14. McLachlan, G.J.; Rathnayake, S. On the Number of Components in a Gaussian Mixture Model. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2014**, *4*, 341–355. [CrossRef]
15. Fukunaga, K. *Introduction to Statistical Pattern Recognition*, 2nd ed.; Academic Press Professional: San diego, CA, USA, 1990.
16. Wang, S.; Sun, H. Measuring Overlap-Rate for Cluster Merging in a Hierarchical Approach to Color Image Segmentation. *Int. J. Fuzzy Syst.* **2004**, *6*, 147–156.
17. Sun, H.; Wang, S. Measuring the Component Overlapping in the Gaussian Mixture Model. *Data Min. Knowl. Discov.* **2011**, *23*, 479–502. [CrossRef]
18. Ester, M.; Krigel, H.P.; Sander, J.; Xu, X. A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Oregon, Portland, 2–4 August 1996; pp. 226–231.
19. Bradley, P.S.; Bennett, K.P.; Demiriz, A. *Constrained K-Means Clustering*; Technical Report MSR-TR-2000-65; Microsoft Research: Redmond, WA, USA, 2000.
20. Bezdec, J.C.; Ehrlich, R.; Full, W. FCM: The Fuzzy c-Means Clustering Algorithm. *Comput. Geosci.* **1984**, *10*, 191–203. [CrossRef]
21. Rusch, T.; Hornik, K.; Mair, P. Asessing and Quantifying Clusteredness: The OPTICS Cordillera. *J. Comput. Graph. Stat.* **2018**, *27*, 220–233. [CrossRef]
22. Yamanishi, K. Descriptive Dimensionality and Its Characterization of MDL-based Learning and Change Detection. *arXiv* **2019**, arXiv:1910.11540.
23. Guha, S.; Mishra, N.; Motwani, R.; O'Callaghan, L. Clustering Data Streams. In Proceedings of the 41st Annual Symposium on Foundations of Computer, Redondo Beach, CA, USA, 12–14 November 2000; pp. 359–366.
24. Song, M.; Wang, H. Highly Efficient Incremental Estimation of Gaussian Mixture Models for Online Data Stream Clustering. In Proceedings of the Intelligent Computing: Theory and Applications III, Orlando, FL, USA, 28 March–1 April 2005; pp. 174–183.
25. Chakrabarti, D.; Kumar, R.; Tomins, A. Evorutionary Clustering. In Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA, 20–23 August 2006; pp. 554–560.
26. Yamanishi, K.; Maruyama, Y. Dynamic Syslog Mining for Network Failure Monitoring. In Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, Chicago, IL, USA, 21–24 August 2005; pp. 499–508.
27. Yamanishi, K.; Maruyama, Y. Dynamic Model Selection with Its Applications to Novelty Detection. *IEEE Trans. Inf. Theory* **2007**, *53*, 2180–2189. [CrossRef]
28. Hirai, S.; Yamanishi, K. Detecting Changes of Clustering Structures Using Normalized Maximum Likelihood Coding. In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Beijing, China, 12–16 August 2012; pp. 343–351.
29. Herbster, M.; Warmuth, M.K. Tracking the Best Expert. *Mach. Learn.* **1998**, *1998*, 151–178. [CrossRef]
30. Ntoutsi, E.; Spiliopoulou, M.; Theodoridis, Y. FINGERPRINT: Summarizing Cluster Evolution in Dynamic Environments. *Int. J. Data Warehous. Min.* **2012**, *8*, 27–44. [CrossRef]
31. van Erven, T.; Grünwald, P.; de Rooji, S. Catching Up Faster by Switching Sooner: A Predictive Approach to Adaptive Estimation with an Application to the AIC-BIC dilemma. *J. R. Stat. Soc. Ser. B (Stat. Methodol.)* **2012**, *74*, 367–417. [CrossRef]
32. Yamanishi, K.; Miyaguchi, K. Detecting Gradual Changes from Data Stream Using MDL-change Statistics. In Proceedings of the 2016 IEEE International Conference on Big Data (Big Data), Washington, DC, USA, 5–8 December 2016; pp. 156–163.
33. Yamanishi, K.; Xu, L.; Yuki, R.; Fukushima, S.; Lin, C.H. Change sign detection with differential MDL change statistics and its applications to COVID-19 pandemic analysis. *Sci. Rep.* **2021**, *11*, 19795. [CrossRef] [PubMed]
34. Hirai, S.; Yamanishi, K. Detecting Latent Structure Uncertainty with Structural Entropy. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018; pp. 26–35.
35. Ohsawa, Y. Graph-Based Entropy for Detecting Explanatory Signs of Changes in Market. *Rev. Socionetwork Strateg.* **2018**, *12*, 183–203. [CrossRef]

36. Still, S.; Biarlek, W.; Léon, B. Geometric Clustering Using the Information Bottleneck Method. In Proceedings of the Advances in Neural Information Processing Systems 16, Vancouver, BC, Canada, 8–13 December 2003.
37. Lin, J. Divergence Measures Based on the Shannon Entropy. *IEEE Trans. Inf. Theory* **1991**, *37*, 145–151. [CrossRef]
38. Cover, T.M.; Thomas, J.A. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*; Wiley-Interscience: New York, NY, USA, 2006.
39. Huber, M.F.; Bailey, T.; Durrant-Whyte, H.; Hanebeck, U.D. On Entropy approximation for Gaussian Mixture Random Vectors. In Proceedings of the IEEE Information Conference on Multisensor Fusion and Integration for Intelligent Systems, Seoul, Korea, 20–22 August 2008; pp. 181–188.
40. Kolchinsky, A.; Tracey, B.D. Estimating Mixture Entropy with Pairwise Distance. *Entropy* **2017**, *19*, 361. [CrossRef]
41. Teicher, H. Identifiability of Finite Mixtures. *Ann. Math. Stat.* **1963**, *34*, 1265–1269. [CrossRef]
42. Yakiwitz, S.J.; Spragins, J.D. On the Identifiability of Fimite Mixtures. *Ann. Math. Stat.* **1968**, *39*, 209–214. [CrossRef]
43. Liu, X.; Shao, Y. Asymptotics for Likelihood Ratio Tests Under Loss of Identifiability. *Ann. Stat.* **2003**, *31*, 807–832. [CrossRef]
44. Dacunha-Castelle, D.; Gassiat, E. Testing in Locally Conic Models and Application to Mixture Models. *ESAIM Probab. Stat.* **1997**, *1*, 285–317. [CrossRef]
45. Keribin, C. Consistent Estimation of the Order of Mixture Models. *Sankhyā Indian J. Statics Ser. A* **2000**, *62*, 49–66.
46. Ghosal, S.; van der Vaart, A.W. Entropies and Rates of Convergence for Maximum Likelihood and Bayes Estimation for Mixtures of Normal Densities. *Ann. Stat.* **2001**, *29*, 1233–1263. [CrossRef]
47. Wu, T.; Sugawara, S.; Yamanishi, K. Decomposed Normalized Maximum Likelihood Codelength Criterion for Selecting Hierarchical Latent Variable Models. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; pp. 1165–1174.
48. Yamanishi, K.; Wu, T.; Sugawara, S.; Okada, M. The Decomposed Normalized Maximum Likelihood Code-length Criterion for Selecting Hierarchical Latent Variable Models. *Data Min. Knowl. Discov.* **2019**, *33*, 1017–1058. [CrossRef]
49. Dempster, A.P.; Laird, N.M.; Rubin, D.B. Maximum Likelihood from Incomplete Data via the EM Algorithm. *J. R. Stat. Soc. Ser. (Methodol.)* **1977**, *39*, 1–38.
50. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
51. Dheeru, D.; Casey, G. UCI Machine Learning Repository. 2017. Available online: http://archive.ics.uci.edu/ml (accessed on 17 August 2022).