博士論文

Detecting Model Changes and their Early Warning Signals with the Minimum Description Length Principle

（記述長最小原理に基づく
モデル変化と早期警戒信号の検知）

平井　聡

Abstract

The growing importance of data utilization has resulted in its promotion; thus, acquiring potential knowledge from data based on mathematical models is gaining attention. In this study, we focus on non-stationary data for data utilization and aim to develop a technology to detect changes in the structure of a dataset based on a uniform standard. We adopt the minimum description length (MDL) principle for the concept of uniform standard and propose algorithms based on the normalized maximum likelihood (NML) code length. This code length achieves the minimum Shtarkov's minimax risk and minimax estimation optimality. First, we propose new indices for measuring the complexity of a dataset using parametric and nonparametric models. In the parametric model, we propose structural entropy (SE), which indexes the uncertainty of the results when selecting a model. In the nonparametric model, we propose kernel complexity (KC), which indexes the concentration of data chunks. Next, we propose new methods for detecting change points and their early warning signals using these indices. In the parametric model, we propose an algorithm using SE as an index and another algorithm using sequential MDL change statistics (SMCS) to express the degree of change. In the nonparametric model, we propose an algorithm using KC as an index. Last, we analyze the efficiency of the proposed indices (SE, SMCS, and KC) in detecting changes using synthetic and practical datasets.

Contents

Chapter 1

Introduction

1.1 Background

In recent years, data generated from human activities and data that can be acquired from devices connected to the Internet have increased exponentially. Moreover, it has become crucial for private and public sectors to effectively use collected data to acquire knowledge, especially for the purposes of marketing, effect measurements, operational efficiency, and so on. The widespread use of data can be attributed to the fact that many types of data can be collected in real time from the web as well as from devices using the Internet of Things (IoT) in various fields. Thus, the scope of data utilization has broadened and various parties have analyzed data from different perspectives. However, no unified methodology exists for acquiring knowledge hidden within data; in fact, data analysis largely depends on the knowledge and experience of each party. Therefore, it is very important to promote the utilization of all collected data based on appropriate mathematical methodologies. Here, we focus on changes over time; the situation for which data is generated changes day by day. Thus, it is crucial to automatically identify changes in a situation and perceive early warning signals from the data for appropriate data utilization.

1.2 Motivation

In this thesis, we aim to detect structural changes of data and their early warning signals in a time series. We assume that multidimensional data points exist at each point in time. Thus, we assume that various types of information on the data points can be acquired at each point in time. These situations are highly relevant in the case of actual data (for actual applications such as marketing and sensor analyses). Fig. 1.1 shows an image of data points at each time.

For example, consider the case where each data item indicates multidimensional consumption data for a customer; each dimension of the data shows the consumption volume

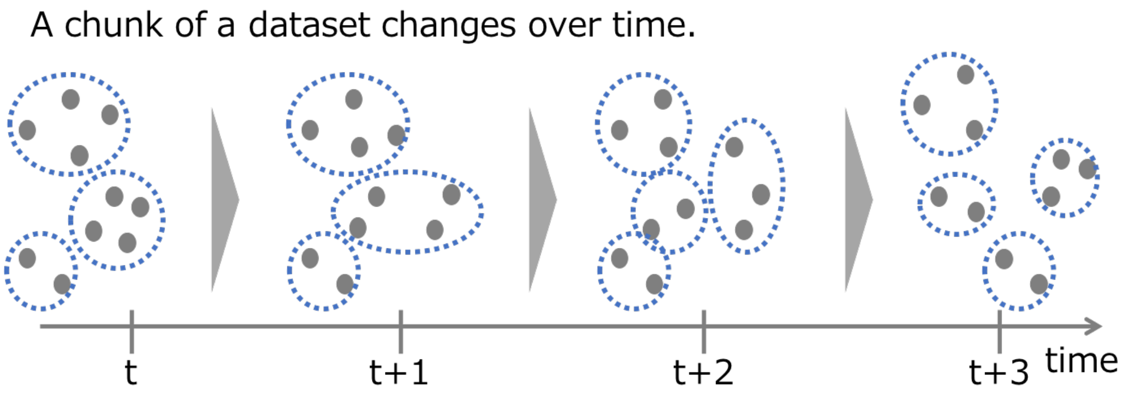


Figure 1.1: Image of distribution change in time-series.

for a specific commodity. Data from several customers are generated every time. We employ a mixture model to allow clustering of customers so that customers with similar consumption patterns are grouped together. It is important to detect changes in the clustering structures to understand customer behavior patterns in the market. Furthermore, it is important detect early warning signals of structural changes (i.e., before the changes actually appear). Accordingly, we will be able to predict changes in future market trends.

Assuming each data point is a sensor data point that can be acquired from a device, each dimension can be regarded as a specific sensor result of the corresponding device. By observing sensor data of these devices as a whole, it is possible to understand the overall operation tendency of the device by grouping devices with similar operation patterns. By detecting changes using early warning signals, information can be applied to business data utilization such as performing maintenance before a device breaks down.

1.3 Research Concepts

In this paper, we consider a situation in which dataset X_t is observed at each time t, and the distribution of the dataset gradually changes over time. Dataset X_t can be expressed as $X_t = \mathbf{x}^n = (\mathbf{x}_1, \cdots, \mathbf{x}_n)^\top \in \mathbb{R}^{n \times m}$, which consists of n data points of dimension m. We consider a situation where the distribution of this dataset gradually changes over time and aim to detect the changes. We propose three methods for detecting changes and their early warning signals.

We consider data chunks formed by the dataset at each time as a feature in this research. By observing the way in which data chunks change, we can detect changes from a macro viewpoint. In particular, the number of clusters can be considered to indicate the characteristics of the clustering structure (structure of data chunks). In this thesis, we track changes in data chunks through the changes in the number of clusters (number of data chunks).

The problem of determining the number of clusters (model selection problem in clustering) is difficult. Generally, the number of clusters is estimated using criteria such as Akaike's information criterion (AIC) [1] and Bayesian information criterion (BIC) [35]. In this research, the minimum description length (MDL) principle is introduced as a model selection criterion. This is a criterion that optimizes the fit of the model to the data and the complexity of the model in a unified framework based on information theory. To deal with non-stationary states of irregular models such as mixture models, we introduce the MDL principle, which can handle them uniformly in the form of code length, as a model selection criterion. Furthermore, by using the MDL principle with a strong theoretical background in model selection, we believe that it can be possible to extract difficult model selection tasks (data). We use normalized maximum likelihood (NML) code length, which is known as a valid index in terms of properties for statistics and information theory. These properties are described in detail in Chapter 2.

The positioning of each chapter in this thesis can be summarized as the figure 1.2. We propose two algorithms: for measuring complexity of static data and for detecting changes in dynamic data. A model selection criterion for the mixture model was proposed by Hirai and Yamanishi for the static dataset [10, 12]. [9] has been proposed as a method that applies this criterion to change detection. However, these methods are intended for discrete model selection and change detection; there are issues that cannot be handled when model selection is difficult and those that are not targeted when changes are gradual. Therefore, in this thesis, the main purpose is deriving indices that can obtain continuous results by indicating the state of data. To obtain continuous results, the aim is to propose an index that shows the model selection uncertainty.

As a first index, we propose structural entropy (SE), which is an index of the ambiguity in model selection, as a value that represents the model selection difficulty. This index quantitatively expresses the ambiguity (uncertainty) of model selection when it is unclear which model should be selected. The concept of this proposed method is described later in 1.3.1. Next, we propose sequential MDL change statistics (SMCS) based on the sequential dynamic model selection (SDMS) change detection algorithm. Although SDMS targets abrupt model changes, SMCS is an extension of MDL change statistics [43] based on the concept of SDMS; it is an index that continuously defines the degree of model change. The concept of this proposed method is described later in 1.3.2. Last, we propose kernel complexity (KC), which defines non-parametric static dataset information without assuming a parametric mixture model. This makes it possible to define complexity in the sense of data chunks even for data that is difficult to represent as a parametric model. Using this index, we propose a change point detection algorithm. The concept of this proposed method is described later in 1.3.3.

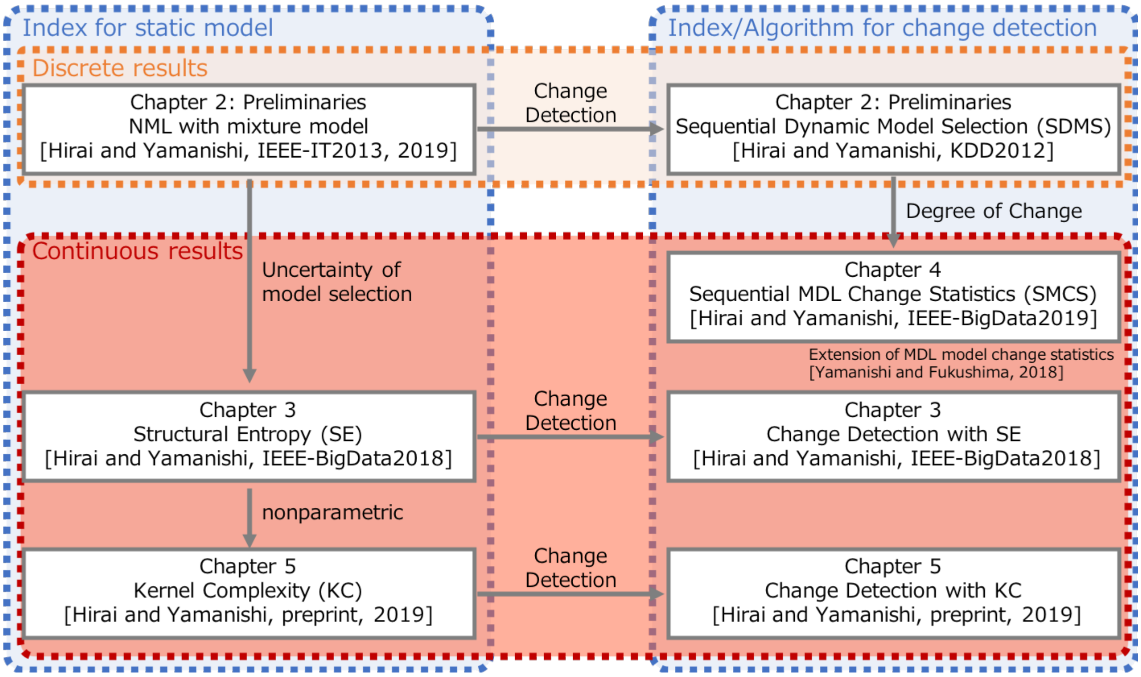

Index for static model
Discrete results
Chapter 2: Preliminaries
NML with mixture model
[Hirai and Yamanishi, IEEE-IT2013, 2019]
Change Detection
Index/Algorithm for change detection
Chapter 2: Preliminaries
Sequential Dynamic Model Selection (SDMS)
[Hirai and Yamanishi, KDD2012]
Degree of Change
Continuous results
Uncertainty of model selection
Chapter 4
Sequential MDL Change Statistics (SMCS)
[Hirai and Yamanishi, IEEE-BigData2019]
Extension of MDL model change statistics
[Yamanishi and Fukushima, 2018]
Chapter 3
Structural Entropy (SE)
[Hirai and Yamanishi, IEEE-BigData2018]
Change Detection
Chapter 3
Change Detection with SE
[Hirai and Yamanishi, IEEE-BigData2018]
nonparametric
Chapter 5
Kernel Complexity (KC)
[Hirai and Yamanishi, preprint, 2019]
Change Detection
Chapter 5
Change Detection with KC
[Hirai and Yamanishi, preprint, 2019]

Figure 1.2: Positioning of each method in this thesis.

1.3.1 Structural Entropy (SE)

Here, we aim to measure the uncertainty of a latent structure. The term “latent” refers to the underlying model structure of the data, such as the number of components in a finite mixture model or the order of an autoregression model. When the data distribution is not clearly separated, the latent structure of the data cannot be clearly determined; thus, focusing on the uncertainty of latent structure becomes an important subject. For tracking changes, high uncertainty can be considered as change points or early warning signals of change. Thus, it is important to track the uncertainty to detect changes as early as possible. We have a hypothesis that latent structure uncertainty will increase before a clear change. In this situation, it is very important to measure the uncertainty of the latent structure and determine the changing period.

The main purpose of this study is to propose a new index that can measure the uncertainty of a latent structure and an algorithm that can detect the change points in a sequential setting. For this, we specifically deal with the structural changes that occur when the number of clusters change. We employ a Gaussian mixture model (GMM) and a Poisson mixture model (PMM) as examples of clustering models, and an autoregression (AR) model as an example of a time-series model. In the experimental results, we show the usefulness of the proposed method using two types of data: artificial datasets and real marketing datasets.

This proposed method is based on [11] and discussed in Chapter 3.

1.3.2 Sequential MDL Change Statistics (SMCS)

We consider the issue of detecting changes of structure in Gaussian mixture models (GMM). The structure here represents the number of clusters, and consider the situation where it changes over time. A number of technologies have been proposed to solve this problem, such as dynamic model selection (e.g., [45, 44, 38]). Although these studies consider discrete changes, the nature of data may change gradually, and it is also important to capture continuous changes. It is natural to assume that there are continuous changes behind the discrete changes. If we can quantify continuous changes, we can detect early warning signals of changes by evaluating this index.

From this standpoint, this study proposes sequential MDL change statistics (SMCS) to develop an index that can handle both discrete and continuous changes simultaneously. We define SMCS as a continuous index that measures the degree of changes from an information-theoretic viewpoint. Also, to perform stable change detection, we propose a suitable parameter setting method by evaluating error probabilities. In the experiment, we evaluate the usefulness of the proposed index using artificial and practical datasets.

To the best of our knowledge, no studies have been conducted on a unifying methodology for detecting structural changes and their early warning signals in a time-varying cluster setting.

This proposed method is based on [13] and discussed in Chapter 4.

1.3.3 Kernel Complexity (KC)

We define new structural information related to a nonparametric distribution and propose a method to detect the changes in this distribution over time. The number of clusters in a clustering model expresses structural information as a group of aggregated data points. In contrast, because it is not possible to define a cluster in a nonparametric model, statistics have previously been used to capture structures using the method of moments. However, even in nonparametric distributions, it is important to aggregate structural information in the form of data-like clusters to allow a global understanding of data. In this study, we propose a new index for structural information, i.e., kernel complexity (KC), which defines the structural information for a nonparametric model. The index is defined by using the Gini index to measure the density of data in terms of information bias [18]. We measure the amount of information provided by the data using the MDL principle. Furthermore, we propose an algorithm to detect changes in the KC when data are in the form of a time series. This algorithm provides a framework for the detection of changes based on the KC.

This study detects changes that the aforementioned data undergo without assuming a specific distribution. Because we do not assume a specific parametric model, we consider an approach that captures the changes in the distribution rather than observing changes with respect to the number of clusters. For this purpose, we use kernel density estimation to represent data distribution. This approach enables us to even handle complex data chunks that cannot be represented by parametric models. We calculate the index, KC, by making some assumptions (such as restricting the parameter on the NML calculation and restricting the data structure). When capturing the changes in the distribution itself, we apply the concept of clustering to a parametric model and propose a method to express the complexity of the structure of the dataset in terms of the density of the dataset. In this case, the amount of information used as a reference when capturing the overall image of the distribution would be expected to differ between the high- and low-density sections of the dataset. Indexing the degree of deviation of this information would then be useful to determine whether it is a complex distribution (a state in which the dataset is sparsely distributed) or a plain distribution (a state in which the dataset is densely distributed). It is possible to detect changes in the density distribution of data by observing an index in a time series. For the purposes of this work, we formulate the following hypotheses for considering changes in the data distributions.

- At a time when there is no change, the distribution maintains a certain form and the index does not change.

- When the distribution changes, the change is considered to occur as a gradual transformation from the original shape until it finally stabilizes into a changed shape. The period during which the change is occurring, the index will also gradually change.

From these hypotheses, we propose an index that quantifies the structural information. In addition, we propose a method that uses this index to detect the change from the perspective of the distribution density.

This study has three aims. The first aim is to propose an index, KC, which defines the structural information of a dataset. This index is defined as an indicator of the distribution of peaks in the dataset similar to a cluster in clustering. The second aim is to develop an algorithm for detecting changes in a structure in a sequential setting. The third is to demonstrate usefulness of the proposed method by using two types of datasets, i.e., artificial and real practical datasets.

This proposed method is based on [14] and discussed in Chapter 5.

1.4 Related Work

We consider related works from two viewpoints: structural information and change detection.

1.4.1 Structural Information

From this viewpoint, we consider related works on parametric and nonparametric models.

First, for parametric models, the issue of selecting the best structure belongs to a wider topic on *model selection*. Conventionally, AIC [1], BIC [35], and MDL [29] have been employed as model selection criteria to address this issue. Specifically, the modern MDL theory uses the NML criterion [31]. Note that the above criteria cannot be applied in a straightforward manner to select latent variable models, including finite mixture models, due to the non-identifiability problem (see [47]). To solve this problem, a combination of the MDL criterion and latent variable completion techniques has been applied to the Naive Bayes model [21], GMM [10, 12], and a wide class of latent variable models [47]. Rissanen et al. proposed the sequentially the NML (SNML) code length for the AR model [32], Takahashi et al. proposed the sequentially discounting the NML (SDNML) code length which is an indicator considering forgetting in SNML [39]. Ito et al. proposed NML for nonnegative matrix factorization [16]. These models are methods aimed at clearly selecting models. In Chapter 3, we aim to express ambiguity as static structural information by quantifying the uncertainty of model selection.

Second, clustering methods (aggregating data points for subgroups) are well-known approaches used in parametric models (e.g., [25],[26]) to define structural information in terms of density. The number of clusters is optimally determined in these clustering methods [21, 10]. The moment method is typically used to define the structure and stochastic features in a nonparametric distribution [6]. Kolmogorov complexity [20] has conventionally been used as an index of the complexity of data sequences. As for nonparametric approaches for model selection, Zhang and Ksecká proposed a method to determine the number of regression lines [48]. Kyrgyzov et al. [22] devised an approach to determine the number of clusters using kernel k-means. This approach uses information related to a latent variable called the cluster index. In Chapter 5, we aim to quantify the global features of a distribution without explicitly using the concept of clustering.

1.4.2 Detection of Changes and Early Warning Signals

First, we introduce methods using parametric models. For model change detection in a dynamic setting, Yamanishi and Maruyama extended the MDL criterion to the dynamic setting to propose dynamic model selection (DMS). They designed the DMS algorithm for detecting changes in statistical models [45, 44]. In the sequential setting, Hirai and Yamanishi proposed the SDMS algorithm, which is an increment variant of DMS and can be applied to latent variable models [9]. Herbster and Warmuth devised a method for tracking the best experts, which sequentially update the weights of the model candidates [8]. Erven et al. suggested the concept of switching distribution [41]. Song and Wang proposed a statistical test-based method for dynamic clustering [38]. Xuan and Murphy created an extension of Bayesian change point detection for a multivariate setting [42].

Davis et al. proposed a method to find the best combination of the number of segments for an AR model [3]. Yamanishi and Fukushima proposed the MDL model change statistics to detect changes in a model [43].

Next, we introduce methods that use nonparametric models. In the supervised scenario, various methods have been developed in the context of concept drift [5]. Liu et al. devised a method based on estimation of the direct density ratio [24]; Jeske proposed an approach referred to as the CUSUM algorithm, which detects changes using a cumulative sum [17]. Tan et al. proposed Bayesian change point detection [40], in which they detected parameter changes in terms of the joint likelihood of data sequence. Harchaoui et al. proposed a kernel Fisher discriminant ratio algorithm [7]; Saatçi et al. devised a technique with a Gaussian process [34]. Kawahara and Sugiyama introduced an algorithm using direct density ratio estimation [19], These methods detect changes based on the differences between data distributions from one time point to another. In Chapter 5, we aim to detect changes in terms of data aggregation by proposing changes in values that quantify structural information in terms of density.

Last, several methods have focused on detection of early warning signals. Ohsawa proposed an approach for detecting explanatory signs of changes and derived a criterion called graph-based entropy [27]. Yamanishi and Miyaguchi introduced a technique for detecting gradual changes by focusing on the code length and derived the MDL change statistics [46]. These criteria enable early detection of signs of changes. As for the rate of detected changes, Huang et al. suggested a method called volatility shift to detect the changes in the distance intervals between the detected changes and privious detected changes [15]. In Chapter 3 and Chapter 4, we deal with the structure of the latent variable and aim to capture changes in the model sequentially and continuously.

1.5 Contributions of Our Research

The contribution of this thesis can be divided into three major categories. First, we define new indices of structural information. Here, the information expresses the difficulty in determining the number of clusters for a parametric model; it expresses the degree of congestion as a chunk in the case of a nonparametric model. As a result, the information of the data structure can be expressed with a unified index. Next, change detection methods using indices that indicate the information of the structure are proposed. Using an index expressed as a continuous value, not only points of abrupt changes but also their early warning signals can be observed. Finally, by experimenting on these behaviors using artificial and real datasets, we show that the techniques are practically effective. Each of the contributions are described in detail in the following sub-sections.

1.5.1 Indices for Structural Information

For parametric and nonparametric structures, we define indices that indicate the information of the structure.

For a parametric model, we consider the problem of determining the number of clusters; for this, we propose an index, called SE, for clustering structure estimation uncertainty. We can continuously grasp the uncertainty of model selection, which cannot be understood by simply determining the number of clusters. We aim to calculate the uncertainty of the MDL-based model selection using SE and determine the uncertainty point for a dataset. In addition to MDL, it is possible to measure the uncertainty of general model selection. To define the criterion, we introduce the concept of entropy. Physically, it is known that fluctuations occur in phase transitions [23, 28]. Analogically, it is assumed that fluctuations occur when model changes occur. Therefore, we quantify fluctuations in terms of SE. We use entropy as an index of the collapse of the structural model selection. The method used for stable uncertainty measurement is important. Thus, we propose a method for selecting a suitable parameter for SE. First, when we measure the uncertainty of model selection using the obtained criterion, we focus on the value itself and establish that uncertainty increases when SE exceeds a certain threshold. We then derive the lower bound for this probability. This enables us to detect the uncertainty of model selection with a certain minimum probability using the SE criterion. Second, this lower bound can be used to determine the SE parameter. To use SE in a stable manner, a suitable parameter should be proposed so that the lower bound for the probability is minimized.

For a nonparametric model, we propose an index to ascertain the structural information of aggregated data. We call this index KC, which is defined by measuring the density of a dataset in terms of information bias with the Gini index; a larger KC indicates that the distribution of the dataset is wider and the structure is more complex. Unlike the number of clusters in a parametric model, KC is not a discrete value, but an index that takes continuous values. We can use KC as a new quantitative index to ascertain nonparametric global information. When calculating KC, we especially use the NML code length based on the MDL principle as a criterion to express information. Even though we adopt kernel density estimation as a nonparametric density estimation method, its NML code length cannot be directly calculated because the maximum likelihood estimator is difficult to calculate. We consider two main points when calculating the NML code length. First, we introduce a subprobability distribution with kernel density estimation. Second, we propose a method of calculating the NML for the subprobability distribution of kernel density estimation by introducing the concept proposed in [31].

1.5.2 Change Detection Methods Using Indices Indicating Structural Information

We propose three methods for detecting change points and their early warning signals.

The first method is the algorithm using SE in a parametric model. The existing change detection methods (e.g., [8]) can detect changes in models, but the detected points are change points that clearly occurred. We propose a method for detecting early warning signals in terms of the uncertainty of model selection using SE.

The second method is the algorithm using SMCS in a parametric model. Most existing algorithms for model change detection listed in Section 1.4 were designed to detect discrete changes only. They could not be applied to detect early warning signals of model changes. The existing algorithms for early warning signals detection listed in Section 1.4 were designed to detect the uncertainty of the models only. They could not be applied to model change detection. We propose a unifying framework for detecting model changes and their early warning signals. The key idea is to employ SMCS. SMCS is a real-valued index that measures the degree of a model change. It is defined as the difference between the code lengths associated with the unchanged and changed models. Hence, its design is based on the MDL principle [29]. By testing the hypothesis based on SMCS, we can detect model changes. Furthermore, we can also detect early warning signals of model changes by tracking the changes of SMCS because it is a real-valued index. The original idea of MDL change statistics was proposed in [46, 43]. However, it was not applied to the detection of early warning signals of model changes. SMCS can be considered as a variant of the original MDL change statistics for the sequential setting, where model selection should be performed every time, and for settings in which latent variable models may be used. We focus on GMMs but the methodology can be extended in a straightforward manner to general latent variable model classes.

Last, we explain the algorithm using KC in a nonparametric model. We propose an algorithm to detect changes in KC when the data are in the form of a time series. This makes it possible to detect changes in the global structure of nonparametric data. Many change-point detection algorithms exist for nonparametric distributions (e.g., [7, 34]). Unlike previous research, our proposed algorithm only detects changes in the global information measured by KC and provides a new view of nonparametric change detection. In addition, KC does not always detect abrupt changes but also gradual ones. Because KC is a continuous value, it is effective in detecting gradual changes.

1.5.3 Empirical Validation of the Effectiveness of our Methods

We employ synthetic datasets to empirically demonstrate that we can raise reliable alarms for model changes and their early warning signals using SE, SMCS, and KC. Specifically, early warning signals can be detected significantly earlier than the alarms provided by existing methods. We also employ two real datasets to validate SE, SMCS, and KC: marketing dataset and household electric consumption dataset. For both datasets, we can detect meaningful change points corresponding to clear behavior changes. Regarding early warning signals, although these signals are not explicitly captured, it is possible to perceive them by changing the SE, SMCS, and KC values.

Chapter 2

Preliminaries

In this thesis, to deal with non-stationary states of irregular models such as mixture models, we derive the MDL principle, which can handle them uniformly in the form of code length, as a model selection criterion. Furthermore, by using the MDL principle with a strong theoretical background in model selection, it is possible to extract difficult model selection tasks (data). If there is an uncertainty in the model selection itself, it will be difficult to determine whether the continuous index is a result of the nature of the data or an error in model selection. Thus, we propose a reliable index based on the MDL principle. We derive the NML code length for model selection and choose a model that achieves the minimum value of the NML code length as the optimal model. The reason we employ NML code length is its following useful properties:

1. it achieves the minimum of Shtarkov's minimax risk [37].

2. it achieves the minimax estimation optimality [31].

In this chapter, we discuss existing methods required as prior knowledge, focusing on NML. Section 2.1 shows the NML code length and its features. Next, we derive sequential dynamic model selection (SDMS) through an existing method for detecting changes of models in Section 2.2.

2.1 Normalized Maximum Likelihood (NML)

Here, based on the MDL, we focus on the NML code length. The NML code length is the optimal code length in terms of Shtarkov's minimax regret [37]. We define the optimal model that satisfies the minimum of this NML code length.

Let an observed data sequence be $\mathbf{x}^n = (\mathbf{x}_1, \cdots, \mathbf{x}_n) \in \mathcal{X}^n$, where $\mathbf{x}_i = (x_{i1}, \cdots, x_{im})^\top$ $(i = 1, \cdots, n)$. We use a model class $\mathcal{P}_{\mathcal{M}(K)} = \{p(X^n; \theta, K) : \theta \in \Theta_K\}$. Here, p is a probability distribution, which has parameter θ, and Θ_K is the parameter space. n is the data size and K is the parameter representing the model (e.g., the number of clusters in clustering). The NML code length is the optimal code length in terms of Shtarkov's minimax regret [37], which is as follows:

$$\min_{q} \max_{\mathbf{x}^n \in \mathcal{X}^n} \Big\{ -\log q(\mathbf{x}^n) - \min_{\theta}(-\log p(\mathbf{x}^n|\theta, K)) \Big\}. \tag{2.1}$$

Distribution q, which achieves the minimum regret, is the NML distribution $p_{\text{NML}}(\mathbf{x}^n; K)$. The NML code length $L_{\text{NML}}(\mathbf{x}^n; K)$ is defined as the description length of probability distribution $p_{\text{NML}}(\mathbf{x}^n; K)$ as $-\log p_{\text{NML}}(\mathbf{x}^n; K)$. In addition, recently, Rissanen showed the minimax estimation optimality [31] and derived the following theorem:

Theorem 1. *As shown below, the maximum likelihood estimator $\hat{\theta}$ and the optimal model $\hat{K}$ derived using the NML code length represent the estimated values that minimize the worst value of Kullback-Leibler divergence from the true distribution $p_{\theta,K}$:*

$$\begin{aligned} \hat{\theta}, \hat{K} &= \operatorname*{argmin}_{\bar{\theta},\bar{K}} \max_{\theta,M} D(p_{\theta,K}||\bar{p}(\mathbf{x}^n)), \\ \bar{p}(\mathbf{x}^n) &= \operatorname*{argmin}_{q} \max_{\mathbf{x}^n \in \mathcal{X}^n} \Big\{ -\log q(\mathbf{x}^n) - (-\log p(\mathbf{x}^n|\bar{\theta}, \bar{K})) \Big\}, \end{aligned} \tag{2.2}$$

where $D(p_1||p_2)$ denotes the Kullback-Leibler divergence $E_{p_1}[\log(p_1(x)/p_2(x))]$, and $\bar{\theta}, \bar{K}$ are arbitrary estimators for parameter θ and model K, respectively.

This theorem shows that the maximum likelihood estimator $\hat{\theta}$ and the optimal model $\hat{K}$ derived using the NML code length have estimation optimality in this criterion. Therefore, we derive the NML code length as a criterion for model selection.

The NML code length is defined as follows:

$$\begin{aligned} L_{\text{NML}}(\mathbf{x}^n; K) &\overset{\text{def}}{=} -\log p_{\text{NML}}(\mathbf{x}^n; K) \\ &= -\log p(\mathbf{x}^n; \hat{\theta}(\mathbf{x}^n), K) + \log \mathcal{C}(\mathcal{M}(K)), \end{aligned} \tag{2.3}$$

where $\mathcal{M}(K)$ is a model defined by K (e.g., $\mathcal{M}(K)$ is a mixture model and K is the number of mixture components). $\mathcal{C}(\mathcal{M}(K))$ is a normalization term and $\hat{\theta}(\mathbf{x}^n)$ is a maximum likelihood estimator calculated as follows:

$$\begin{aligned} \mathcal{C}(\mathcal{M}(K)) &\overset{\text{def}}{=} \int p(\mathbf{y}^n; \hat{\theta}(\mathbf{y}^n), K)\, \mathrm{d}\mathbf{y}^n, \\ \hat{\theta}(\mathbf{x}^n) &\overset{\text{def}}{=} \operatorname*{argmax}_{\theta}\ p(\mathbf{x}^n; \theta, K). \end{aligned}$$

Here, model K can be determined by minimizing this criterion.

2.1.1 Calculating the Normalization Term for the NML Code Length

We describe the method for calculating the normalization term of the NML code length. Rissanen proposed a method for calculating the normalization term because it is difficult to calculate it directly [30]. In this method, when the maximum likelihood estimator of the distribution parameter θ is a sufficient statistic, it can be calculated analytically by the following method.

The distribution p can be decomposed as follows:

$$p(\mathbf{y}^n;\theta)\,\mathrm{d}y^n = f(z|\hat{\theta})\cdot g(\hat{\theta};\theta)\,\mathrm{d}z\,\mathrm{d}\hat{\theta}$$

where we define the model class $\mathcal{P}_{\mathcal{M}} = \{p(X^n;\theta) : \theta \in \Theta\}$. We assume that mapping a exists and we can write $(\hat{\theta}, z) = a(y^n)$; f is a conditional probability density function and g is a probability density function of $\hat{\theta}$, where θ is a parameter. This equality can be used to calculate the normalization term as follows:

$$\begin{aligned}\mathcal{C}(\mathcal{M}) &\overset{\text{def}}{=} \int p(\mathbf{y}^n;\hat{\theta}(\mathbf{y}^n))\,\mathrm{d}\mathbf{y}^n \\ &= \int\int f(z|\hat{\theta})\cdot g(\hat{\theta};\hat{\theta})\,\mathrm{d}z\,\mathrm{d}\hat{\theta} \\ &= \int g(\hat{\theta};\hat{\theta})\,\mathrm{d}\hat{\theta}.\end{aligned}$$

We use this method to calculate the NML code length associated with kernel density estimation. For a detailed discussion, refer to the book published by Rissanen [30].

2.1.2 NML Code Lengths for Several Models

In this section, we show the NML code lengths for various models: Gaussian mixture model (GMM), Poisson mixture model (PMM), and autoregression (AR) model.

NML with GMM

Let $\mathbf{x}^n = (\mathbf{x}_1, \cdots, \mathbf{x}_n)$ be the dataset. We denote the probability density function of the Gaussian mixture distribution with latent variable z^n as follows:

$$\begin{aligned}p(\mathbf{x}^n, z^n; \mu, \Sigma) &= \prod_{k=1}^{K} \pi_k^{h_k} \times \prod_{x_i \in z_k} \frac{1}{(2\pi)^{\frac{mh_k}{2}} \cdot |\Sigma_k|^{\frac{h_k}{2}}} \\ &\qquad \times \exp\left\{-\frac{1}{2}(\mathbf{x}_i - \mu_k)^\top \Sigma_k^{-1}(\mathbf{x}_i - \mu_k)\right\},\end{aligned}$$

where π_k is a mixture weight of cluster k, h_k is the number of data points belonging to cluster k, μ_k is the center of cluster k, and Σ_k is a variance-covariance matrix.

An upper bound on the NML code length of the GMM was derived in [10, 12] as follows:

$$\begin{aligned} L_{\mathrm{NML}}(\mathbf{x}^n, z^n; \mathcal{M}(K)) &\leq -\log p(\mathbf{x}^n, z^n; \mathcal{M}(K), \hat{\theta}(\mathbf{x}^n, z^n)) \\ &\quad + \log \mathcal{C}_{\mathrm{u}}(\mathcal{M}(K), n) \\ &=: L_{\mathrm{uNML}}(\mathbf{x}^n, z^n; Y, \mathcal{M}(K)) \\ \mathcal{C}_{\mathrm{u}}(\mathcal{M}(K), n) &= \sum_{h_1, \cdots, h_K} \frac{N!}{h_1! \cdot \cdots \cdot h_K!} \prod_{k=1}^{K} \left(\frac{h_k}{N}\right)^{h_k} \\ &\quad \times B(m, R, \epsilon) \cdot \left(\frac{h_k}{2\mathrm{e}}\right)^{\frac{mh_k}{2}} \frac{1}{\Gamma_m\left(\frac{h_k - 1}{2}\right)}, \\ B(m, R, \epsilon) &\overset{\mathrm{def}}{=} \frac{2^{m+1} R^{\frac{m}{2}} \prod_{j=1}^{m} {\epsilon_{1j}}^{-\frac{m}{2}}}{m^{m+1} \cdot \Gamma\left(\frac{m}{2}\right)}, \end{aligned}$$

where R and ϵ are parameters.

NML with PMM

Let an observed data sequence be $\mathbf{x}^n = (\mathbf{x}_1, \cdots, \mathbf{x}_n)$, where $\mathbf{x}_i = (x_{i1}, \cdots, x_{iW})^\top$ $(t = 1, \cdots, n)$. We assume the case in which each data point x_{iw} $(w = 1, \cdots, W)$ is generated from Poisson distribution $Poi(z_i)$, where z_i is a cluster index to which data point x_{iw} $(w = 1, \cdots, W)$ belongs. We denote the probability density function of Poisson mixture distribution with latent variable z^n as follows:

$$p(\mathbf{x}^n, z^n; \pi, \lambda) = \prod_{k=1}^{K} \prod_{i=1}^{n} \pi_k^{\delta(z_i = k)} \prod_{w=1}^{W} \left\{ \frac{(\lambda_k)^{x_{iw}}}{x_{iw}!} \mathrm{e}^{-\lambda_k} \right\}^{\delta(z_i = k)}, \tag{2.4}$$

where $\lambda = (\lambda_1, \cdots, \lambda_K)$ are the parameters of the Poisson distribution, π_k represents the probability that data x belongs to the cluster k, and K represents the number of clusters. For this distribution, we provide the following theorem:

Theorem 2. *The NML code length for PMM is approximated as follows:*

$$\begin{aligned} L_{\mathrm{NML}}(\mathbf{x}^n, z^n; \mathcal{M}(K)) &\approx -\log p(\mathbf{x}^n, z^n; \hat{\pi}(z^n), \hat{\lambda}(\mathbf{x}^n, z^n)) + \log \mathcal{C}(\mathcal{M}(K)), \\ \mathcal{C}(\mathcal{M}(K)) &= \sum_{h_1, \cdots, h_K \geq 0, \sum_k h_k = n} \frac{N!}{h_1! \cdot \cdots \cdot h_K!} \prod_{k=1}^{K} \left(\frac{h_k}{N}\right)^{h_k} \sqrt{\frac{2h_k W \alpha}{\pi}}. \end{aligned}$$

This theorem can be derived as follows:

Proof. The probability density distribution for PMM is defined as Equation (2.4). For this distribution, we can calculate the maximum likelihood estimator as $\hat{\lambda}_k = \frac{1}{h_k W} \sum_{i=1}^{n} \sum_{w=1}^{W} \delta(z_i = k) x_{i,w}$, $\hat{\pi}_k = \frac{h_k}{n}$, and $h_k = \sum_{i=1}^{n} \delta(z_i = k)$.

First, we define the NML code length for a Poisson model. Generally, we can obtain the NML code length as shown in Equation (2.3); here, we calculate the NML for a Poisson model using Rissanen's approximation formula [33]. According to [33], the approximation of the NML code length can be calculated as follows:

$$L_{\text{NML}}(x^n) = -\log p(x^n; \hat{\theta}(x^n)) + \frac{k}{2}\log\frac{n}{2\pi} + \log\int\sqrt{|I(\theta)|}\mathrm{d}\theta + o(1),$$

where θ is a parameter.

Using this formula, we can calculate the NML code length for the Poisson model as follows:

$$\begin{aligned} L_{\text{NML}}(x^n) &= -\log p(\mathbf{x}^n|\hat{\lambda}(\mathbf{x}^n)) + \frac{1}{2}\log\frac{NW}{2\pi} + \log 2\sqrt{\alpha} + o(1) \\ &\approx -\log p(\mathbf{x}^n; \hat{\lambda}(\mathbf{x}^n)) + \frac{1}{2}\log\frac{NW}{2\pi} + \log 2\sqrt{\alpha} \end{aligned} \tag{2.5}$$

Next, we calculate the NML code length for PMM. Generally, the NML code length for mixture models is calculated as follows:

$$L_{\text{NML}}(x^n, z^n; \mathcal{M}(K)) = -\log p(x^n, z^n; \hat{\theta}(x^n, z^n)) + \log\mathcal{C}(\mathcal{M}(K)),$$

where $\mathcal{M}(K)$ represents a mixture model where K is the number of mixture components, z^n are cluster indices, and θ is a parameter set. Then, using Equation (2.5), we can calculate the NML code length as follows:

$$\begin{aligned} L_{\text{NML}}(\mathbf{x}^n, z^n; \mathcal{M}(K)) &\approx -\log p(\mathbf{x}^n, z^n; \hat{\pi}(z^n), \hat{\lambda}(\mathbf{x}^n, z^n)) + \log\mathcal{C}(\mathcal{M}(K)), \\ \mathcal{C}(\mathcal{M}(K)) &= \sum_{h_1,\cdots,h_K\geq 0, \sum_k h_k = n} \frac{n!}{h_1!\cdot\cdots\cdot h_K!}\prod_{k=1}^{K}\left(\frac{h_k}{N}\right)^{h_k}\sqrt{\frac{2h_kW\alpha}{\pi}}. \end{aligned} \tag{2.6}$$

□

When we calculate the normalization term in Equation (2.6), we use the recurrence formula proposed by [10] as follows:

$$\begin{aligned} \mathcal{C}(n, \mathcal{M}(K+1)) &= \sum_{r_1, r_2\geq 0, r_1+r_2=n} \frac{n!}{r_1!r_2!}\left(\frac{r_1}{N}\right)^{r_1}\left(\frac{r_2}{N}\right)^{r_2} \times \mathcal{C}(r_1, \mathcal{M}(K))I(r_1, \alpha), \\ I(n, \alpha) &\overset{\text{def}}{=} \sqrt{\frac{2NW\alpha}{\pi}}. \end{aligned}$$

This recurrence formula can be used to calculate the normalization term in $O(n^2 \cdot K)$.

NML with AR Model

We consider the calculation of the NML code length for the AR model. The AR model is used for time-series datasets in the case when data point x_t depends on past data points x_{t-K}^{t-1}; the AR model can be formulated as follows:

$$x_t = \beta^\top \bar{x}_t + \epsilon_t,$$

where $\beta = (\beta_1, \cdots, \beta_K)^\top$ is a coefficient parameter, $\epsilon_t = (\epsilon_{t1}, \cdots, \epsilon_{tK})$ is generated by a normal distribution $\mathcal{N}(0, \sigma^2 I)$, and $\bar{x}_t = (x_{t-1}, \cdots, x_{t-K})^\top$ represent past data points.

We consider a simple case where variance σ^2 is fixed. The probability density function is

$$p(x_t|x^{t-1}; \sigma^2, \beta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_t - \beta^\top \bar{x}_t)^2}{2\sigma^2}\right).$$

Let us consider the case for finding a model K at time t from data x_{t-w+1}^t. Parameter w is the window size used to calculate the optimal model K. Here, we denote the dataset as $y^w = (x_{t-w+1}, \cdots, x_t)$, and the discounting NML code length is defined as follows:

$$\begin{aligned} L_{\mathrm{NML}}(y^w) &\overset{\mathrm{def}}{=} \sum_{l=m+1}^{w} -\log p_{\mathrm{SDNML}}(y_l|y^{l-1}), \\ p_{\mathrm{SDNML}}(y_l|y^{l-1}) &= \frac{p(y^l; \hat{\theta}(y^l))}{K_l(y^l)}, \\ K_l(y^l) &= \int P(y \cdot y^{l-1}; \hat{\theta}(y \cdot y^{l-1}))\mathrm{d}y, \end{aligned} \tag{2.7}$$

where p_{SDNML} is the sequential discounting NML (SDNML) [39], which is a discounting variant of sequential NML (SNML) [32]. In this work, we use L_{NML} as the code length at each time t.

2.2 Sequential Dynamic Model Selection (SDMS)

In this thesis, to detect the change points of a latent structure, we use the sequential dynamic model selection (SDMS) code length proposed by Hirai and Yamanishi [9]. DMS is a method for identifying model sequence in a batch as proposed by Yamanishi and Maruyama [45, 44]. SDMS is a technique that applies it to changes in the number of clusters of a latent variable model and selects the optimal model sequentially. Using this criterion, we can detect the change points for abrupt changes. Here, we denote SDMS in a simple form as below.

Let an observed data sequence be $X^T = (X_1, \cdots, X_T)$, $X_t = \mathbf{x}^n = (\mathbf{x}_{t1}, \cdots, \mathbf{x}_{tn})$ where $\mathbf{x}_{ti} = (x_{ti1}, \cdots, x_{tim})^\top$ $(i = 1, \cdots, n)$, and a latent variable be $Z^T =$

$(Z_1, \cdots, Z_T)$, which represents the cluster index to which each data point belongs at each time. The SDMS code length is calculated at each time t as follows:

$$\begin{aligned} L_{\text{SDMS}}(X_t, Z_t; \mathcal{M}(\hat{K}_{t-1}), \mathcal{M}(K_t)) &= L_{\text{NML}}(X_t, Z_t; \mathcal{M}(K_t)) - \log p(K_t|\hat{K}_{t-1}; \alpha), \\ p(K_t|\hat{K}_{t-1}; \alpha) &= \begin{cases} 1-\alpha & \text{if } K_t = \hat{K}_{t-1} \text{ and } \hat{K}_{t-1} \neq 1, K_{\max}, \\ 1-\alpha/2 & \text{if } K_t = \hat{K}_{t-1} \text{ and } \hat{K}_{t-1} = 1, K_{\max}, \\ \alpha/2 & \text{if } K_t = \hat{K}_{t-1} \pm 1, \end{cases} \end{aligned} \tag{2.8}$$

where L_{NML} is the code length of clustering with X_t and Z_t, $\mathcal{M}(K)$ is the mixture model with K clusters, and $p(K_t|\hat{K}_{t-1}; \alpha)$ represents the probability that the model changes. We use the maximum a posteriori (MAP) estimator with a beta distribution as the prior distribution to estimate parameter α ($0 \leq \alpha \leq 1$). $K_{\max}$ represents the maximum possible value of K, K_t is the number of clusters at each time t, and $\hat{K}_{t-1}$ is the estimated model K_{t-1} at a previous time. The SDMS algorithm outputs K_t, which minimizes the L_{SDMS} criterion at each time t. Here, we can detect the change points where K_t changes from $\hat{K}_{t-1}$.

Chapter 3

Structural Entropy

We consider the case that a dataset can be defined by parametric model. Under this condition, we aim to measure uncertainty of model selection and propose a novel index called structural entropy (SE). In addition, we propose a novel algorithm for detecting early warning signals using SE. In this chapter, Section 3.1 introduces SE. Section 3.2 discusses how to calculate optimal parameter in SE. Section 3.4 gives an algorithm for detecting early warning signals using SE. In Section 3.3, we examine how much error rate can be reduced in MDL model selection by using SE. Lastly, we show experimental results in Section 3.5. We present works of Section 3.1, Section 3.2, Section 3.4, and Section 3.5 in BigData 2018 [11]. Section 3.3 is an extended work of [11].

3.1 Structural Entropy (SE)

We propose SE as an index for measuring the uncertainty of model selection. SE is defined from the viewpoint of model selection, and we aim to use the SE to measure the uncertainty.

Let we consider uncertainty of model selection in terms of the code length. As discussed in Chapter 2, we employ the code length to select the optimal model at each time instant. However, it is anticipated that a model will not always be clearly determined. Thus we suppose that model selection will be performed in the presence of uncertainty. We propose the SE index using the code length as follows:

$$\begin{aligned} SE &\overset{\text{def}}{=} -\sum_{k\in\mathcal{K}} p(K)\log p(K), \\ p(K) &\overset{\text{def}}{=} \frac{\exp(-\beta\cdot L(K))}{\sum_{K\in\mathcal{K}}\exp(-\beta\cdot L(K))}, \end{aligned} \tag{3.1}$$

where β is a parameter, and $\mathcal{K}$ is defined as the domain of the model parameter K and $L(K)$ is the code length of data X with model $\mathcal{M}(K)$. SE is an index that expresses how much uncertainty occurs in model selection in the form of entropy. To create a simple definition for the SE, we define the domain as $\mathcal{K} = \{\hat{K}, \hat{K}'\}$, where $\hat{K}, \hat{K}'$ are the first and second best models, respectively. They are as follows:

$$\hat{K} \overset{\text{def}}{=} \underset{K \in \mathcal{K}_{\text{all}}}{\operatorname{argmin}} L(K), \tag{3.2}$$

$$\hat{K}' \overset{\text{def}}{=} \underset{K \in \mathcal{K}_{\text{all}} \setminus \{\hat{K}\}}{\operatorname{argmin}} L(K), \tag{3.3}$$

where $\mathcal{K}_{\text{all}} = \{1, \cdots, K_{\max}\}$ describes the domain of K, and $K_{\max}$ is the largest possible K.

3.2 Calculation of Suitable Parameter

In this section, we consider the nature of the SE and derive a method for calculating the suitable parameter using the theoretical property of SE. We expect the probability of raising an alarm to be in the range of values from 0 to 1. We discuss a suitable parameter setting in the following steps:

1. The probability that SE exceeds threshold ϵ:
   We discuss the nature of this probability. Here, we derive a lower bound for this probability (which we call P_{Low}).

2. The principle for calculating the suitable parameter:
   We consider the case where the P_{Low} is upper bounded by a small value. Using this property, we can choose the suitable parameter ϵ.

3.2.1 Probability that the SE Exceeds Threshold ϵ

Consider the case where SE exceeds threshold ϵ, we define an alarm condition as below:

$$a(t) \overset{\text{def}}{=} \begin{cases} 1 & \text{if } SE > \epsilon, \\ 0 & \text{otherwise.} \end{cases}$$

We introduce the function g, which is the inverse function of the entropy function h as follows:

$$h(p) \overset{\text{def}}{=} -p \log p - (1-p) \log(1-p),$$
$$g(p) \overset{\text{def}}{=} h^{-1}(p), \tag{3.4}$$

where the domain of $g(p)$ is defined so that $0 < g(p) \leq 1/2$.

To discuss the nature of SE , we make following assumptions hold:

Assumption 1. *Let*

$$r \overset{\text{def}}{=} \frac{p(X;\hat{\theta}(X,\mathcal{M}(\hat{K})))}{p(X;\hat{\theta}(X,\mathcal{M}(\hat{K}')))},$$

where $\hat{\theta}(X,\mathcal{M}(K))$ is a maximum likelihood estimator. The value of r satisfies the following inequality:

$$R^{-n} < r < R^{n},$$

where the value $R(>1)$ is constant.

For example, we briefly show that Assumption 1 can be natural for the GMM. Let the dataset be $\mathbf{x}^n = (\mathbf{x}_1, \cdots, \mathbf{x}_n) \in \mathbb{R}^{m\times n}$. We denote the probability density function of Gaussian mixture distribution with the latent variable z^n as follows:

$$\begin{aligned} p(\mathbf{x}^n, z^n; \mu, \Sigma) &= \prod_{k=1}^{K} \pi_k^{h_k} \times \prod_{x_i \in z_k} \frac{1}{(2\pi)^{\frac{mh_k}{2}} \cdot |\Sigma_k|^{\frac{h_k}{2}}} \\ &\quad \times \exp\left\{-\frac{1}{2}(\mathbf{x}_i - \mu_k)^\top \Sigma_k^{-1} (\mathbf{x}_i - \mu_k)\right\}, \end{aligned}$$

where π_k is a mixture weight of cluster k, h_k is the number of data points belonging to cluster k, μ_k is a center of cluster k, and Σ_k is a variance-covariance matrix. Then, the maximum likelihood is calculated as follows:

$$p(\mathbf{x}^n, z^n; \hat{\theta}(\mathbf{x}^n, z^n)) = \prod_{k=1}^{K} \left(\frac{h_k}{n}\right)^{h_k} (2\pi \mathrm{e})^{-\frac{mh_k}{2}} \prod_{j=1}^{m} \hat{\lambda}_{jk}^{-\frac{h_k}{2}}. \tag{3.5}$$

Using this formula, we can calculate upper and lower bounds on the likelihood for $(\mathbf{x}^n, z^n)$ as follows:

$$\left(\frac{1}{K}\right)^{n} (2\pi\mathrm{e})^{-\frac{mn}{2}} \hat{\lambda}_{\max}^{-\frac{mn}{2}} \leq \text{Equation (3.5)} \leq (2\pi\mathrm{e})^{-\frac{mn}{2}} \hat{\lambda}_{\min}^{-\frac{mn}{2}},$$

where $\hat{\lambda}_{\max} \overset{\text{def}}{=} \underset{j,k}{\operatorname{argmax}}\, \hat{\lambda}_{jk}$ and $\hat{\lambda}_{\min} \overset{\text{def}}{=} \underset{j,k}{\operatorname{argmin}}\, \hat{\lambda}_{jk}$, and $\hat{\lambda}_{jk}$ is the j-th eigenvalue of the maximum likelihood estimator $\hat{\Sigma}_k$. We use this formula to calculate the upper and lower bounds on r as follows:

$$\frac{1}{\hat{K}^n} \left(\frac{\hat{\lambda}'_{\min}}{\hat{\lambda}_{\max}}\right)^{\frac{mn}{2}} \leq r \leq \hat{K}'^n \left(\frac{\hat{\lambda}'_{\max}}{\hat{\lambda}_{\min}}\right)^{\frac{mn}{2}}. \tag{3.6}$$

Assumption 2. *We consider the case where the expected value of* $\log r$ *satisfies the following inequality:*

$$\exists\gamma, 0 < \gamma < \left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} - 1,$$
$$\log\frac{\exp(\ell(\hat{K}))}{\exp(\ell(\hat{K}'))} < E_X[\log r] < \log\left\{(1+\gamma)\frac{\exp(\ell(\hat{K}))}{\exp(\ell(\hat{K}'))}\right\}, \tag{3.7}$$

where $\ell(\hat{K})$ *is the code length other than likelihood in total code length.*

Using the definitions of Equation (3.2) and Equation (3.3), the first inequality of Equation (3.7) can be derived as follows:

$$\begin{aligned} & \hat{K} = \operatorname*{argmin}_{K\in\mathcal{K}_{\text{all}}} L(K) \\ \Leftrightarrow\quad & \frac{p(X;\hat{\theta}(X),\mathcal{M}(\hat{K}))}{\exp(\ell(\hat{K}))} > \frac{p(X;\hat{\theta}(X),\mathcal{M}(\hat{K}'))}{\exp(\ell(\hat{K}'))} \\ \Leftrightarrow\quad & \log\frac{\exp(\ell(\hat{K}))}{\exp(\ell(\hat{K}'))} < E_X[\log r]. \end{aligned}$$

In addition, in measuring the uncertainty of model selection, we want to consider the situation where the structure selection is uncertain to some extent. For this reason, we consider making an assumption like the second inequality of Equation (3.7) with respect to the optimal parameters $\hat{K}$ and $\hat{K}'$.

Then, we get the following theorem:

Theorem 3. *Under the condition of* $\mathcal{K} = \{\hat{K}, \hat{K}'\}$*, the probability that* $SE > \epsilon$ *(*$\epsilon(> 0)$ *is a constant value) is lower bounded as follows:*

$$Prob\,[SE > \epsilon] \geq 1 - \exp\left[-\frac{\eta^2}{2(\sigma^2 + M\eta/3)}\right], \tag{3.8}$$

where

$$\eta \overset{\text{def}}{=} \ell(\hat{K}) - \ell(\hat{K}') + \frac{1}{\beta}\log\frac{1-g(\epsilon)}{g(\epsilon)} - E_X[\log r], \tag{3.9}$$
$$\sigma^2 \overset{\text{def}}{=} Var_X[\log r], \tag{3.10}$$
$$M \overset{\text{def}}{=} n\log R + |E_X[\log r]|. \tag{3.11}$$

From Equation (3.8), SE enables us to find the uncertainty of model selection with a certain probability or greater, Then, this probability ranges from the lower bound value to 1.

Proof. When the SE exceeds threshold ϵ, the probability $p(\hat{K})$ $(\geq 1/2)$ satisfies the following condition:

$$\begin{aligned} p(\hat{K}) &= \frac{1}{1+\exp\left\{-\beta\cdot(L(\hat{K}')-L(\hat{K}))\right\}} < 1-g(\epsilon) \\ &\Leftrightarrow \; r < \frac{\exp(\ell(\hat{K}))}{\exp(\ell(\hat{K}'))}\cdot\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta}, \end{aligned}$$

where $g(\epsilon)$ is defined as Equation (3.4). Here, we can transform this probability as follows:

$$\begin{aligned} & Prob\,[SE>\epsilon] \\ =\; & Prob\left[r < \frac{\exp(\ell(\hat{K}))}{\exp(\ell(\hat{K}'))}\cdot\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta}\right] \\ =\; & 1-Prob\left[\frac{\exp(\ell(\hat{K}))}{\exp(\ell(\hat{K}'))}\cdot\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} \leq r\right]. \end{aligned} \tag{3.12}$$

Under Assumptions 1, 2, we employ the Bernstein's inequality to obtain a lower bound on the probability of Equation (3.12). The Bernstein's inequality is formulated as follows:

Lemma 1. *For variable X_i, the following inequality holds:*

$$Prob\left[\sum_{i=1}^{n} X_i \geq \eta\right] \leq \exp\left(-\frac{\eta^2}{2(n\sigma^2+M\eta/3)}\right), \tag{3.13}$$

where $E[X_i]=0$, $|X_i|<M$ (with probability 1 for all i), $\sigma^2=\sum Var[X_i]/n$, and $\eta\geq 0$.

Using Lemma 1, we obtain a lower bound on the probability of Equation (3.12) as follows:

$$\begin{aligned} & Prob\,[SE>\epsilon] \\ =\; & 1-Prob\left[\log\left\{\frac{\exp(\ell(\hat{K}))}{\exp(\ell(\hat{K}'))}\cdot\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta}\right\} - E_X[\log r] \leq \log r - E_X[\log r]\right] \\ \geq\; & 1-\exp\left[-\frac{\eta^2}{2(\sigma^2+M\eta/3)}\right], \end{aligned}$$

where η, σ^2, and M is defined in Equation (3.9), Equation (3.10), and Equation (3.11), respectively. □

3.2.2 Method for Choosing Suitable Parameter β

Here, we consider selecting the suitable parameter β. As previously discussed, we expect the probability of raising an alarm to assume a wide range of values. In order to increase the precision range of this probability, we expect that a lower bound on the probability $Prob\,[SE > \epsilon]$ will satisfy the property as follows:

$$\exists\delta > 0,\ P_{\text{Low}} \leq \delta \text{ (} \delta \text{ is a small constant)}, \tag{3.14}$$

where P_{Low} is a lower bound as described in Theorem 3. If P_{Low} does not satisfy this property, the probability $Prob\,[SE > \epsilon]$ is sufficiently high to raise an alert when the latent structure is not very uncertain. This leads to an increase in the false alarm rate, which is a disadvantage to the stability of the SE. Thus, the mentioned property is required.

In order to satisfy the property of Equation (3.14), we have the following theorem that β satisfies:

Theorem 4. *β should be upper and lower bounded as follows:*

$$\frac{\Gamma}{G} \leq \frac{1}{\beta} \leq \frac{\Gamma}{G} - \frac{2M}{3G}\log(1-\delta), \tag{3.15}$$

where

$$\begin{aligned} G &\overset{\text{def}}{=} \log\frac{1-g(\epsilon)}{g(\epsilon)}, && (3.16)\\ \Gamma &\overset{\text{def}}{=} \log(1+\gamma'), && (3.17)\\ M &\overset{\text{def}}{=} n\log R + \left|\log\left\{(1+\gamma')\frac{\exp(\ell(\hat{K}))}{\exp(\ell(\hat{K}'))}\right\}\right|, && (3.18)\\ 0 &< \gamma' < \gamma. && (3.19) \end{aligned}$$

Here, if the parameter β is less than $\beta_{\text{Low}} = 1/\left(\frac{\Gamma}{G} - \frac{2M}{3G}\log(1-\delta)\right)$, we can predict the situation where the alerts for the uncertainty of the latent structure tend to be too frequent, and the false alarm rate increases. In the particular case of $\beta \to 0$, $p(\hat{K})$ is always equal to $1/2$, as shown in Equation (3.1), and it is observed that any estimation result has uncertainty. This is the reason why we derive the boundary of β as shown in Theorem 4. By using Theorem 4, we obtain the stability of SE by setting parameters ϵ, R, etc. that are easier to set instead of setting parameter β. For example, in GMM, we set R naturally, as shown in Equation (3.6).

Proof. We can derive a boundary of β from Equation (3.14) as follows:

$$P_{\text{Low}} \leq \delta$$

$$\begin{aligned}&\Leftrightarrow \quad -\eta^2 \geq 2\log(1-\delta)\cdot\big(\sigma^2 + M\eta/3\big)\\&\Leftrightarrow \quad \frac{-D_1-\sqrt{D_2}}{G} \leq \frac{1}{\beta} \leq \frac{-D_1+\sqrt{D_2}}{G}\end{aligned}$$

where

$$\begin{aligned}D_1 &= \frac{1}{3}M\log(1-\delta) - \Gamma,\\D_2 &= \log(1-\delta)\cdot\left(\frac{1}{9}M^2\log(1-\delta) - 2\sigma^2\right),\end{aligned}$$

where G, Γ, M are defined as Equation (3.16), Equation (3.17), and Equation (3.18), respectively. Because we expect that the criterion SE can be used stably regardless of σ^2, we consider the case where the smallest range of β can be calculated. Here, if we assume that $\sigma^2 \to 0$, the range of β is the smallest as follows:

$$\frac{\Gamma}{G} \leq \frac{1}{\beta} \leq \frac{\Gamma}{G} - \frac{2M}{3G}\log(1-\delta),$$

where G, Γ, and M are defined as Equation (3.16), Equation (3.17), and Equation (3.18), respectively. □

3.3 Error Rate of Model Selection using SE

We examine how much error rate can be evaluated in MDL based model selection with SE. In evaluating the property of MDL based model selection with SE, we use two models K, K'.

At first, we consider the null hypothesis H_0 and the alternative hypothesis H_1 below:

H_0: The true model is K'.

H_1: The true model is K.

We decide to reject the null hypothesis H_0 when the condition of $L(K) < L(K')$ is satisfied. $L(K), L(K')$ are NML code lengths for models K, K', respectively. This is simple MDL based model selection, but by using SE, it is possible to determine that we do not use the result of model selection when there is uncertainty in model selection ($SE > \epsilon$). As a result, when it is difficult to make a decision, it becomes possible to select not to judge the result, and the probability of mistakes can be reduced.

Here, the Type-I error probability for simple MDL based model selection is calculated as follows:

$$P_{\mathrm{I}} = \int_{X\in\left\{X \,\middle|\, L(K)<L(K')\right\}} p(X; K', \theta)\mathrm{d}X.$$

Here, by introducing SE, it is possible to increase the reliability of the model selection result by selecting not to judge the result when uncertainty measured by SE in the model selection is high. Although the error in this case is not a strict Type-I error, the probability of selecting an incorrect model can be suppressed by model selection with reliability added. Here, we call this error probability as error probability with reliability (EPR). By introducing SE, when $SE > \epsilon$ (when uncertainty is high), we can avoid using the result of model selection. Then, EPR can be calculated as follows:

$$P_{\mathrm{EPR}} = \int_{X \in \left\{ X \middle| L(K) + \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} < L(K') \right\}} p(X; K', \theta) \mathrm{d}X.$$

By using the nature of NML code length, we can derive following theorem:

Theorem 5. *EPR is upper bounded as follows:*

$$P_{\mathrm{EPR}} \leq \exp\left\{ \ell(K') - \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} \right\}, \tag{3.20}$$

where $\ell(K')$ is the code length other than likelihood in total code length.

The content of the right side of Equation 3.20 has the following properties:

$$\begin{aligned} \ell(K') &= O(\log n), \\ \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} &= O(n). \end{aligned}$$

By these properties, EPR converges to zero when n goes to infinity. Since the term of $O(n)$ is generated by introducing the uncertainty determination by SE, we can say that SE significantly reduces EPR. The proof of this theorem 5 is given as follows:

Proof. By introducing SE, when $SE > \epsilon$ (when uncertainty is large), we can avoid using the result of model selection. Then, EPR can be calculated as follows:

$$P_{\mathrm{EPR}} = \int_{X \in \left\{ X \middle| L(K) + \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} < L(K') \right\}} p(X; \theta, \mathcal{M}(K')) \mathrm{d}X.$$

We then derive an upper bound on EPR with SE as follows:

$$\begin{aligned} P_{\mathrm{EPR}} &= \int_{X \in \left\{ X \middle| L(K) + \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} < L(K') \right\}} p(X; K', \theta) \mathrm{d}X \\ &\leq \int_{X \in \left\{ X \middle| L(K) + \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} < L(K') \right\}} p(X; K', \hat{\theta}(X)) \mathrm{d}X. \end{aligned} \tag{3.21}$$

Here, the inequality in the range of integration can be deformed as follows:

$$\begin{aligned}
& L(K) + \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} < L(K') \\
\Leftrightarrow\quad & L(K) + \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} < -\log p(X; K', \hat{\theta}(X)) + \ell(K') \\
\Leftrightarrow\quad & p(X; K', \hat{\theta}(X)) < \exp\left\{-L(K) + \ell(K') - \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta}\right\}.
\end{aligned}$$

By using this inequality, we can calculate an upper bound on Equation (3.21) as follows:

$$\begin{aligned}
& \mathrm{Equation(3.21)} \\
< \quad & \int_{X \in \left\{X \,\middle|\, L(K)+\log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta} < L(K')\right\}} \exp\left\{-L(K) + \ell(K') - \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta}\right\} \mathrm{d}X \\
< \quad & \exp\left\{\ell(K') - \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta}\right\} \cdot \int_{X \in \mathbb{R}^n} \exp\{-L(K)\}\, \mathrm{d}X. \qquad (3.22)
\end{aligned}$$

Here, using the Kraft's inequality, the integration in Equation (3.22) is not more than 1. Using this fact, the Equation (3.22) can be upper bounded as follows:

$$\mathrm{Equation(3.22)} \leq \exp\left\{\ell(K') - \log\left(\frac{1-g(\epsilon)}{g(\epsilon)}\right)^{1/\beta}\right\}.$$

□

3.4 Algorithm for Detecting Early Warning Signals of Changes using SE

We apply SE to detect early warning signals of changes. When a structure of a model of data changes, it can be considered that there is first a small movement in the dataset; then, a large change in structure is visible. Since SE is an index for measuring the uncertainty of structure of a model of data, it can be considered as an index that increases the uncertainty when the movement becomes large to change the structure. Using this SE index, an algorithm for detecting changes is presented in Algorithm 1. For simplicity, we express the dataset at each time as $X_t \in \mathbb{R}^{n \times m}$. $L_t(K_t)$ is the code length with model K_t at time t, for which we use the NML code length in Section 2.1 or the SDMS code length in Section 2.2. Thus, we denote the code length as follows:

$$L_t(K_t) \;=\; -\log p(X_t; \hat{\theta}(X_t), \mathcal{M}(K_t)) + \ell(K_t),$$

$$\ell(K_t) \quad = \quad \begin{cases} \log \mathcal{C}(\mathcal{M}(K_t)), & (L_t(K_t) = L_{\mathrm{NML}}), \\ \log \mathcal{C}(\mathcal{M}(K_t)) - \log p(K_t|\hat{K}_{t-1};\alpha), & (L_t(K_t) = L_{\mathrm{SDMS}}), \end{cases}$$

where $\mathcal{M}(K)$ is a model with model parameter K (e.g. a Gaussian mixture model with K components), and $p(K_t|\hat{K}_{t-1};\alpha)$ is defined by Equation (2.8).

Algorithm 1 Algorithm for detecting changes.

Calculate SE:

Calculate SE score defined by Equation (3.1) as follows:

$$SE_t \overset{\text{def}}{=} -\sum_{K \in \mathcal{K}} p(K_t = K) \log p(K_t = K),$$
$$p(K_t = K) \overset{\text{def}}{=} \frac{\exp(-\beta \cdot L_t(K))}{\sum_{K \in \mathcal{K}_t} \exp(-\beta \cdot L_t(K))},$$

Detection of early warning signals of changes:

Detect early warning signals of changes with following conditions:

$$a(t) \quad = \quad \begin{cases} 1 & \text{if } SE_t > \epsilon, \\ 0 & \text{otherwise.} \end{cases}$$

We can detect early warning signals of changes in the case where $a(t) = 1$.

3.5 Experimental Results

3.5.1 Defining Suitable Parameters

We first consider the properties of suitable parameter β derived from Theorem 4. We use a lower bound on β (in Theorem 4, an upper bound on $1/\beta$), in order to make it easier to raise alarms within the constraints of β. For example, in the case where the data size is equal to 1000 in GMM, we calculate a lower bound for β related to threshold ϵ and δ using Equation (3.15)). The result of the suitable parameters are shown in Figure 3.1 and Figure 3.2.

We define optimal parameter β for GMM, PMM, and AR as $\beta_{\mathrm{GMM}} = 0.05, \beta_{\mathrm{PMM}} = 0.05,$ and $\beta_{\mathrm{AR}} = 1.33$, respectively. Here, we set the threshold ϵ as a fixed value 0.1.

3.5.2 GMM for Artificial Dataset

Here, we consider the case where the model is a GMM. We created an artificial dataset and empirically demonstrated the usefulness of SE. In GMM, we consider the number

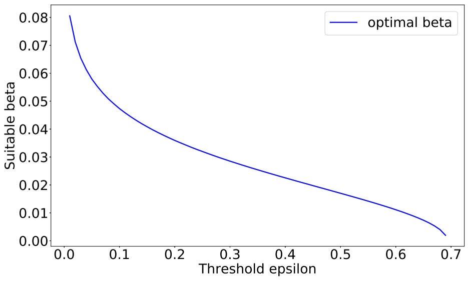


Figure 3.1: Optimal parameter β related to threshold ϵ.

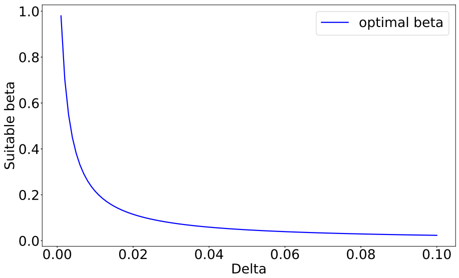


Figure 3.2: Optimal parameter β related to δ.

of clusters as a model and evaluate its changes. Because SDMS [9] and tracking the best expert (TBE) [8] methods can also be used to define model selection as the problem for determining the number of clusters of GMM, we introduce these two methods as comparison targets.

Single change

We consider the case where the number of change periods is equal to 1. The center of the generated cluster is generally changed over time as follows:

$$\begin{cases} K^* = 2,\ \mu = (\mu_1, \mu_2) & \text{if } 1 \leq t \leq \tau_1, \\ K^* = 3,\ \mu = (\mu_1, \mu_2, u(t)) & \text{if } \tau_1 + 1 \leq t \leq \tau_2, \\ K^* = 3,\ \mu = (\mu_1, \mu_2, \mu_3) & \text{if } \tau_2 + 1 \leq t \leq T, \end{cases} \tag{3.23}$$

$$\text{where } u(t) = \frac{(\tau_2 - t)\mu_2 + (t - \tau_1)\mu_3}{\tau_2 - \tau_1}.$$

We generated a dataset as in Equation (3.23) and detected the uncertainty of model selection using the proposed algorithm (SE). The values of SE_t at each time t and first uncertainty points detected were calculated as shown in Figure 3.3(a). In this figure, the SDMS algorithm was also plotted to compare clear and uncertainty changes. As seen in this figure, SE detected uncertainty points, and the uncertainty points is earlier than the clear change point.

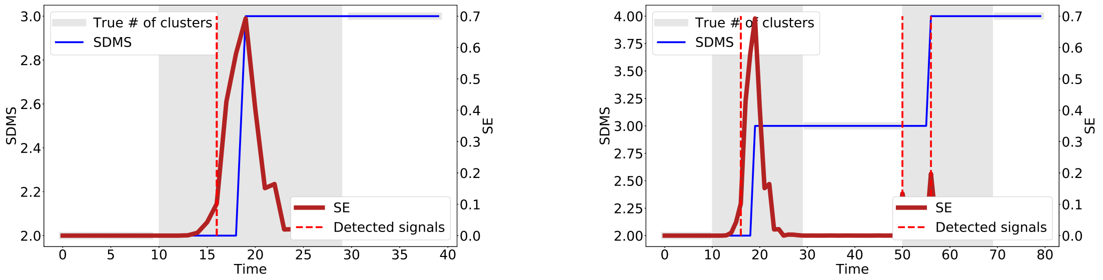
In addition to this, we calculated the benefit and delay scores to evaluate how well the algorithm detected uncertainty of model selection. The benefit and delay scores are defined in Equation (3.24) and Equation (3.25), respectively:

$$\text{benefit} \overset{\text{def}}{=} \begin{cases} 1 - (\hat{t} - t^*)/U & \text{if } t^* \leq \hat{t} \leq t^* + U, \\ 0 & \text{otherwise.} \end{cases}, \tag{3.24}$$

$$\text{delay} \overset{\text{def}}{=} \begin{cases} \hat{t} - t^* & \text{if } \hat{t} \in \text{Transition period}, \\ T & \text{otherwise}. \end{cases}, \tag{3.25}$$

where $\hat{t}$ is the first point where the algorithm detects uncertainties, and T is the length of transition period.

We evaluated how well the proposed method worked in comparison with other change point detection algorithms such as SDMS and TBE. The results are listed in TABLE 3.1(a). We generated 10 different datasets for the same model and detected uncertainty points for each. The table values are the average ones of benefit and delay. These results show that the proposed algorithm detected uncertainty points faster than other methods in terms of both of benefit and delay scores.



(a) Single change

(b) Multiple changes

Figure 3.3: Experiments for the data sequence with single and multiple change periods with GMM model. The transition periods are $10 \leq t \leq 29$ for the single pattern and $10 \leq t \leq 29, 50 \leq t \leq 69$ for the multiple pattern. In the change periods, the centers of the clusters gradually changed over time as given by Equation (3.23). This graph shows the values of SE_t, the number of clusters estimated by SDMS at each time t and detected uncertainty points with $\epsilon = 0.1$. We detected the uncertainty point at time $t = 16$ and found that the number of clusters changed at time $t = 19$ in the single change pattern; we detected similar points in the case of the multiple pattern.

Table 3.1: Benefit and delay scores for algorithms: SE, SDMS, and TBE (GMM)

(a) Single change

Methods	benefit	delay
SE	0.68	6.5
SDMS	0.55	9.0
TBE	0.45	11.0

(b) Multiple changes

Methods	benefit	delay
SE	0.76	4.8
SDMS	0.61	7.9
TBE	0.55	9.1

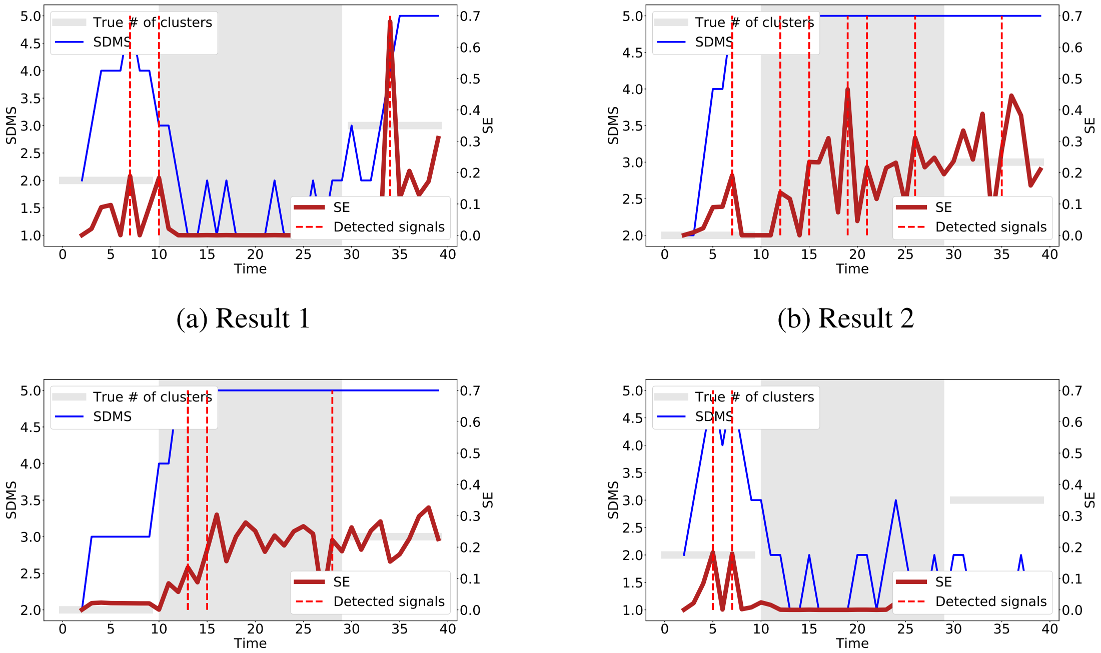
Multiple changes

We consider the case where the number of change periods is equal to 2. There exists two transition periods, in each of which the type of the change follows the single change case.

The values of SE_t at each time t and detected first uncertainty points were calculated as shown in Figure 3.3(b). In the evaluation, we used the benefit and delay scores defined in Equation (3.24) and Equation (3.25). The results are listed in TABLE 3.1(b).

As in the single change case, the proposed method detects uncertainty well even in the multiple changes case.

3.5.3 PMM for Artificial Dataset



(a) Result 1

(b) Result 2

(c) Result 3

(d) Result 4

Figure 3.4: Graphs showing the values of SE_t and estimated number of clusters by SDMS at each time t and detected uncertainty points with PMM model.

We consider the case where the structure is defined as a PMM. We created an artificial dataset and empirically demonstrated the usefulness of SE.

Considering the case where the mixture structure gradually changed over time, we generated the dataset such that the probability of the model class gradually changed over time from $t = \tau_1 + 1$ to $t = \tau_2$ as follows:

$$\begin{cases} Prob(K=2)=1 & \text{if } 1 \leq t \leq \tau_1, \\ Prob(K=2)=1-\frac{t-\tau_1}{\tau_2-\tau_1}, Prob(K=3)=\frac{t-\tau_1}{\tau_2-\tau_1} & \text{if } \tau_1+1 \leq t \leq \tau_2, \\ Prob(K=3)=1 & \text{if } \tau_2 \leq t \leq T. \end{cases}.$$

In this case, the dataset has the following parameters for each cluster:

$$\lambda = \begin{cases} \lambda_1, \lambda_2 & \text{if } 0 \leq t \leq \tau_1, \\ \lambda_1, \lambda_2, \lambda_3 & \text{if } \tau_1+1 \leq t \leq T. \end{cases}.$$

We obtained the results shown in Figure 3.4. As can be seen from the figures, it is difficult to define a true model with the PMM. However, we can see that there are data pattern where the uncertainty can be detected during the transition period. This is probably because the difficulty of selecting the true model increases the SE due to uncertainty.

3.5.4 AR Model

We next consider the case where the time-series dataset followed the AR model $AR(K)$ which denotes the autoregression model of order K. The order parameter K changed over time as follows:

$$\begin{cases} Prob(K=1)=1 & \text{if } 1 \leq t \leq \tau_1, \\ Prob(K=1)=1-\frac{t-\tau_1}{\tau_2-\tau_1}, Prob(K=3)=\frac{t-\tau_1}{\tau_2-\tau_1} & \text{if } \tau_1+1 \leq t \leq \tau_2, \\ Prob(K=3)=1 & \text{if } \tau_2 \leq t \leq T. \end{cases} \quad (3.26)$$

The dataset was generated as shown in Figure 3.5, for example.

We generated a dataset as in Equation (3.26) and detected the uncertainty of order selection using the proposed algorithm. We calculated SE for the case of $w = 50$ and $w = \infty$. In the case of $w = \infty$, we used all past data sequences for NML defined by Equation (2.7). The values of SE_t at each time t and detected first uncertainty points were calculated as shown in Figure 3.6. As seen in these figures, the SE detected the uncertainty points, and the uncertainty points appeared earlier than the clear change points.

As for evaluation metrics, we used the benefit and delay scores respectively defined in Equation (3.24) and Equation (3.25). The results are listed in TABLE 3.2. These results show that the SE method detected the uncertainty points earlier than the other methods in terms of both of benefit and delay scores.

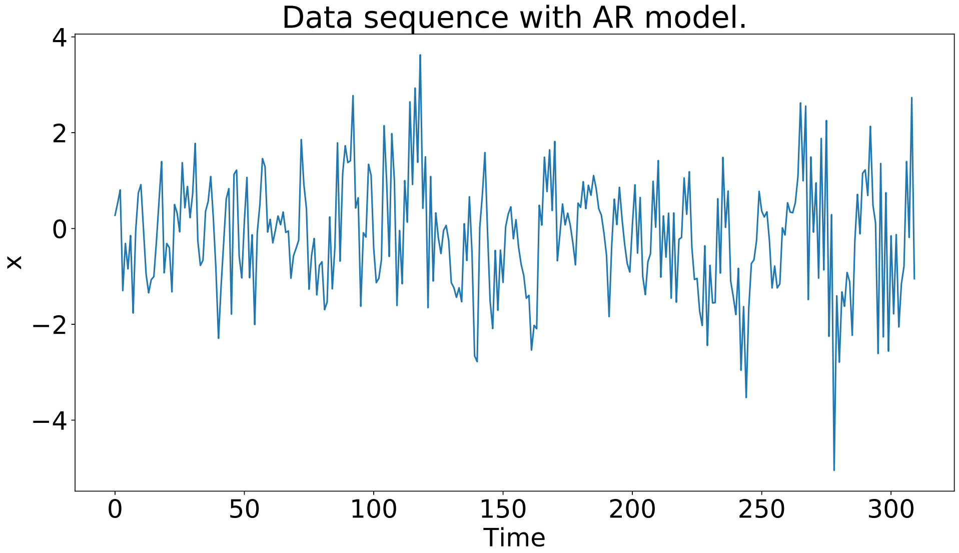


Figure 3.5: Example data sequence generated by Equation (3.26); this figure shows the value of x_t relative to t.

Table 3.2: Benefit and delay scores for algorithms: SE, SDMS, and TBE (AR model)

(a) Window size $= 50$

Methods	Benefit	Delay
SE	0.53	46.9
SDMS	0.19	80.9
TBE	0.00	99.0

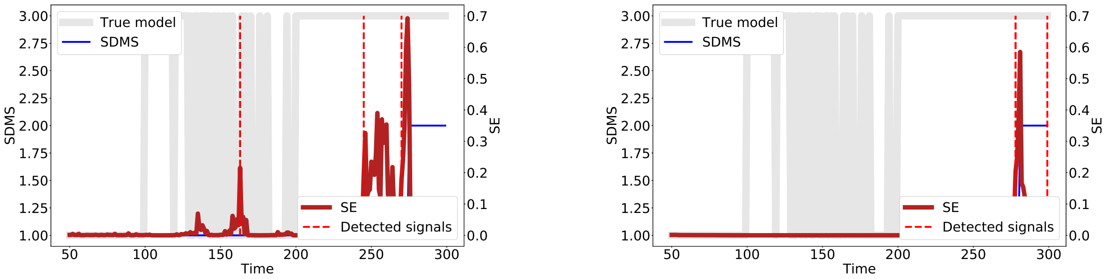
(b) Window size $= \infty$

Methods	Benefit	Delay
SE	0.20	79.1
SDMS	0.00	99.0
TBE	0.00	99.0

3.5.5 Real Data Using Gaussian Mixture Model

We evaluated our method using a real marketing dataset provided by Hakuhodo, Inc. This dataset consists of customers' beer purchasing behaviors, and covers beer from six manufacturers (called A to F here). At each time instant t ($t = \tau, \cdots, T$), the data point is defined as $\mathbf{x}_{i,t} \in \mathbb{R}^6$ ($i = 1, \cdots, N$), and each data point describes beer consumption from time $t - \tau + 1$ to t. In this experiment, the data size is $N = 1509$, window size is $\tau = 14$, and parameter value is $\beta = 0.03$.

Using SE, we detected the uncertainty before model change as shown in Figure 3.7. For example, SDMS detected the change point at $t = 24$. In comparison, the SE method captured the uncertainty at the time instant before the clear change occurred.



(a) Window size $= 50$

(b) Window size $= \infty$

Figure 3.6: Transition periods are $100 \leq t \leq 199$. In the change periods, the order of AR model varies over time as given by Equation (3.26). This graph shows the values of SE_t and estimated numbers of clusters by SDMS at each time t and detected uncertainty points. We detected uncertainty points at time instants $t = 163, 245, 270$ for window size $= 50$.

Looking at the clustering structure itself at time $t = 22$ to 24, we detected the uncertainty point at time $t = 23$ and clear change at time $t = 24$. As shown in TABLE 3.3, it can be qualitatively evaluated that the brand B consumption of cluster 2 was subtly lower at time $t = 23$, and a collapse of the cluster structure was indicated before the dormant user cluster (cluster 8) was created at time $t = 24$.

3.6 Conclusion

We proposed a new index called SE for measuring the uncertainty of model selection. We derived the features of SE using the probability that it exceeded threshold ϵ and selected the optimal parameter β using this feature. The experimental results showed the usefulness of the proposed method for two types of data; an artificial dataset and a real marketing dataset. For the artificial dataset, we showed that SE detected the uncertainty of a latent structure earlier than other change point detection algorithms such as SDMS and TBE. The real marketing dataset indicated customers' beer purchasing behaviors for a time-series clustering problem. We detected the uncertainty points in terms of the customers' purchase behaviors using SE. A few aspects of this method will be considered in a future study. In terms of evaluation, we can define data types for which SE is effective by increasing the variation of the experiments. In addition, the SE method is potentially applied to other models and real datasets, and the method may be extended for dynamic selection of algorithms to estimate the optimal parameter.

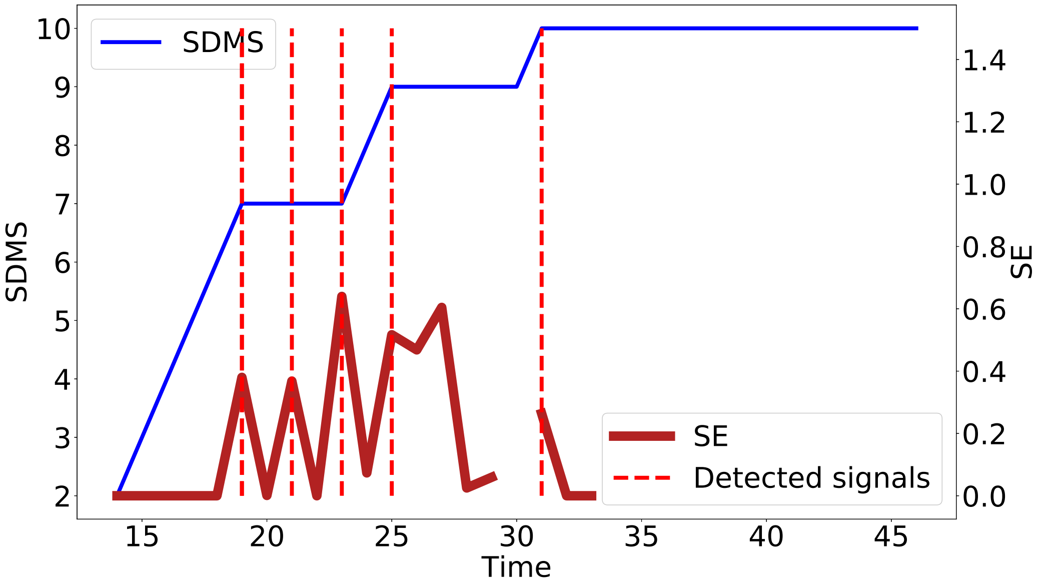


Figure 3.7: Values of SE_t at each time t and detected uncertainty points for real data. For example, SDMS detected the change point at $t = 24$. In comparison, the SE method captured the change in the uncertainty at a time instant before the clear change occurred.

Table 3.3: Estimated clusters for time $t = 22$ to 24.

Time 22

brand	clu-1	clu-2	clu-3	clu-4	clu-5	clu-6	clu-7
A	3397	0	16	14	22	6	21
B	12	126	19	7	49	13	36
C	0	0	2328	0	15	10	1815
D	0	0	0	3079	5	7	1551
E	0	0	0	0	559	0	0
F	0	0	0	0	0	2371	0
num	307	368	259	269	15	159	132

Time 23

brand	clu-1	clu-2	clu-3	clu-4	clu-5	clu-6	clu-7
A	3336	0	16	12	18	6	25
B	12	119	16	7	41	11	35
C	0	0	2398	0	18	9	1701
D	0	0	0	3171	3	6	1538
E	0	0	0	0	580	0	0
F	0	0	0	0	0	2495	0
num	305	373	257	270	14	158	132

Time 24

brand	clu-1	clu-2	clu-3	clu-4	clu-5	clu-6	clu-7	clu-8
A	3782	10	18	9	30	5	23	0
B	0	3118	14	0	26	10	136	0
C	0	0	2492	0	18	6	111	0
D	0	0	0	3296	0	5	1818	0
E	0	0	0	0	638	0	0	0
F	0	0	0	0	0	2466	0	0
num	206	319	248	197	12	156	202	169

Chapter 4

Sequential MDL Change Statistics

We consider the case where a dataset can be generated by a parametric model. Under this condition, we propose sequential MDL change statistics (SMCS) for measuring degree of change of the model and propose a novel algorithm for detecting changes and their early warning signals. In this chapter, Section 4.1 proposes SMCS. Section 4.2 discusses how to calculate suitable parameters in SMCS. For calculating suitable parameters, we use the evaluation of error probability rate which is derived in the same way as with [43]. Lastly, we give experimental results in Section 4.3. We present works of Section 4.1 to Section 4.3 to BigData 2019 [13].

4.1 Sequential MDL Change Statistics (SMCS)

We consider the index that represents the degree of change in the model. This index is an extension of the index proposed in [43] to the sequential setting. This index is based on the concept of SDMS that captures discrete changes, and the degree of change is converted into a continuous value as statistics. We first propose an index that only detects whether the model has changed.

Definition 1. *We define SMCS Φ_t as follows:*

$$\begin{aligned} \Phi_t \overset{\text{def}}{=} & \min_{\mathcal{M}} \{L_{\text{NML}}(X_{t-1} \cdot X_t, Z_{t-1} \cdot Z_t; \mathcal{M}) + \mathcal{L}(\mathcal{M}, \mathcal{M})\} \\ & - \min_{\mathcal{M}', \mathcal{M}''} \big\{L_{\text{NML}}(X_{t-1}, Z_{t-1}; \mathcal{M}') \\ & \quad + L_{\text{NML}}(X_t, Z_t; \mathcal{M}'') + \mathcal{L}(\mathcal{M}'', \mathcal{M}')\big\} - n\epsilon, \end{aligned} \tag{4.1}$$

where L_{NML} is an NML code length as in Equation (2.3), and $\epsilon(> 0)$ is a parameter. $\mathcal{M}$ represents a model defined by the number of clusters K. $L(\mathcal{M}_2, \mathcal{M}_1)$ means the code length required for encoding the model transition from $\mathcal{M}_1$ to $\mathcal{M}_2$. $Z_t = (z_{t1}, \cdots, z_{tn})$

is the latent variable, and each z_{ti} is the cluster index which the data point x_{ti} belongs to. In calculation of NML code length, Z_t is calculated by X_t using clustering algorithm (e.g. expectationmaximization (EM) algorithm).

SMCS can be regarded as a criterion for judging whether different models should be defined at times t and $t-1$. This enables the quantification of the degree of change in the model at each time t. However, when it is desired to observe changes in the model in a time series, it is also conceivable to introduce an index of whether the model from the previous time has changed. By setting the restriction of $\mathcal{M}' = \mathcal{M}$, the criteria of Equation (4.1) can be modified as follows:

$$\begin{aligned} \Phi_t \;\; = \;\; & \{L_{\mathrm{NML}}(X_{t-1} \cdot X_t, Z_{t-1} \cdot Z_t; \mathcal{M}) + \mathcal{L}(\mathcal{M}, \mathcal{M})\} \\ & - \min_{\mathcal{M}'} \big\{L_{\mathrm{NML}}(X_{t-1}, Z_{t-1}; \mathcal{M}) \\ & \quad + L_{\mathrm{NML}}(X_t, Z_t; \mathcal{M}') + \mathcal{L}(\mathcal{M}', \mathcal{M})\big\} - n\epsilon, \end{aligned} \tag{4.2}$$

where we can calculate $\mathcal{L}(\mathcal{M}, \mathcal{M})$ and $\mathcal{L}(\mathcal{M}', \mathcal{M})$ using Equation (2.8).

Using the SMCS criterion, we can detect both discrete change points and their early warning signals. The algorithm is summarized as Algorithm 2. At the discrete change points, we expect the value of SMCS to be greater than zero ($\Phi_t \geq 0$). In addition to this, as SMCS criterion is a continuous value, we can detect the early warning signals by observing changes in the value of SMCS itself. As in Equation (4.3) in Algorithm 2, when the SMCS value is larger than previous times, we can detect the early warning signals of changes because we can determine that the degree of change is large.

4.2 Calculation of a Suitable Parameter

4.2.1 Error Probability Rate with Definition (4.1)

We evaluate error probabilities when we use SMCS for detecting changes. In this evaluation, we consider the case wherein we use SMCS as in Equation (4.1), and consider the Type-I/II error probabilities on the hypothesis test below:

$$H_0 \;:\; D_{t-1} \sim \mathcal{M}_0^*,\ D_t \sim \mathcal{M}_0^* \tag{4.3}$$

$$H_1 \;:\; D_{t-1} \sim \mathcal{M}_1^*,\ D_t \sim \mathcal{M}_2^*. \tag{4.4}$$

Here, X_t, Z_t is written as D_t for simplicity. $\mathcal{M}_0^*, \mathcal{M}_1^*, and \mathcal{M}_2^*$ are unknown models. This represents the hypothesis test whether the time t is a change point.

First, we consider the Type-I error probability. On the hypothesis (4.3), by defining the sequential model statistics as Equation (4.1), we can calculate Type-I error probability

Algorithm 2 Algorithm for detecting discrete change points and their early warning signals.

```
**Calculate SMCS at each time:**
Calculate SMCS score $\Phi_t$ defined by Equation (4.1) or Equation (4.2).
**Detect the discrete change points and their early warning signals:**
  1. Detect the discrete change points with the following conditions:
     $a_1(t) = \begin{cases} 1 & \text{if } \Phi_t \geq 0, \\ 0 & \text{otherwise.} \end{cases}$
     Here, we can detect the discrete change points in the case where $a_1(t) = 1$.
  2. Detect the early warning signals with the following conditions:
     $a_2(t) = \begin{cases} 1 & \text{if } \Phi_t - E[\Phi_{\tau+1}^{t-1}] \geq \eta * Std[\Phi_{\tau+1}^{t-1}], \\ 0 & \text{otherwise,} \end{cases}$
     where $\Phi_{\tau+1}^{t-1} = \Phi_\tau, \cdots, \Phi_{t-1}$, and $\tau$ is the time when the most recent change was detected. $E[\cdot]$ and $Std[\cdot]$ represent an expectation and a standard deviation, respectively. Here, we can detect the early warning signals in the case where $a_2(t) = 1$.
```

as follows:

$$\begin{aligned} \text{Type-I error prob.} &= Prob_{H_0}\left[\Phi_t \geq 0\right] \\ &= \int_{D_{t-1,t} \in \{D : \Phi_t \geq 0\}} p(D_{t-1}; \theta_0^*, \mathcal{M}_0^*) p(D_t; \theta_0^*, \mathcal{M}_0^*) \, \mathrm{d}D_{t-1,t}, \end{aligned}$$

where θ_0^* is the true parameter of model $\mathcal{M}_0^*$.

Here, the following theorem is straightforwardly derived for [43]:

Theorem 6. *An upper bound on the Type-I error probability is calculated as follows:*

$$\textit{Type-I error prob.} \leq \exp\left\{ -n\left(\epsilon - \frac{\log \mathcal{C}_{2n}(\mathcal{M}_0^*) + \mathcal{L}(\mathcal{M}_0^*, \mathcal{M}_0^*)}{n}\right)\right\}$$

where $\mathcal{C}_{2n}(\mathcal{M}_0^*)$ is a normalization term for a model $\mathcal{M}_0^*$ with data size $2n$.

Here, the proofs for the bounds can be derived in the same way as in [43]. We describe the detail proofs in Appendix A.1. As seen in this theorem, we can reduce the Type-I error probability by deriving the adjustment parameter $\epsilon (> 0)$ under the condition below:

$$\epsilon > \frac{\log \mathcal{C}_{2n}(\mathcal{M}_0^*) + \mathcal{L}(\mathcal{M}_0^*, \mathcal{M}_0^*)}{n}.$$

Next, we consider Type-II error probability. On the hypothesis (4.4), by defining the sequential model statistics as Equation (4.1), we can calculate the Type-II error probability as follows:

$$\begin{aligned} & \text{Type-II error prob.} \\ = \; & Prob_{H_1}\left[\Phi_t < 0\right] \\ = \; & \int_{D_{t-1,t} \in \{D : \Phi_t < 0\}} p(D_{t-1}; \theta_1^*, \mathcal{M}_1^*) p(D_t; \theta_2^*, \mathcal{M}_2^*) \, \mathrm{d}D_{t-1,t}, \end{aligned}$$

where θ_1^*, θ_2^* are the true parameters of model $\mathcal{M}_1^*, \mathcal{M}_2^*$, respectively.

Here, the following theorem is straightforwardly derived for [43]:

Theorem 7. *An upper bound on the Type-II error probability is calculated as follows:*

$$\begin{aligned} & \textit{Type-II error prob.} \\ & < \exp\left\{ -n\left(2\alpha(1-\alpha) d_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*) - \frac{\alpha \ell_n(\mathcal{M}_1^*, \mathcal{M}_2^*, \epsilon)}{n} \right) \right\}, \end{aligned}$$

where α is the parameter, and $d_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*)$ and $\ell_n(\mathcal{M}_1^*, \mathcal{M}_2^*, \epsilon)$ is defined as follows:

$$d_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*) \overset{\text{def}}{=} \frac{1}{2\alpha(1-\alpha)}(1 - \delta_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*)), \tag{4.5}$$

$$\delta_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*) \overset{\text{def}}{=} \left(\int (p(D_{t-1}; \theta_1^*, \mathcal{M}_1^*) p(D_t; \theta_2^*, \mathcal{M}_2^*))^{1-\alpha} \times \tilde{p}_{\text{NML}}(D_{t-1,t})^{\alpha} \, \mathrm{d}D_{t-1,t} \right)^{\frac{1}{n}} \tag{4.6}$$

$$\ell_n(\mathcal{M}_1^*, \mathcal{M}_2^*, \epsilon) \overset{\text{def}}{=} \log \mathcal{C}_n(\mathcal{M}_1^*) + \log \mathcal{C}_n(\mathcal{M}_2^*) + \mathcal{L}(\mathcal{M}_2^*, \mathcal{M}_1^*) + \log \tilde{\mathcal{C}}_{2n} + n\epsilon. \tag{4.7}$$

Here, $d_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*)$ is the α-divergence between distribution $p(D_{t-1}; \theta_1^*, \mathcal{M}_1^*) \cdot p(D_t; \theta_2^*, \mathcal{M}_2^*)$ and $\tilde{p}_{\text{NML}}(D_{t-1,t})$, and $\tilde{p}_{\text{NML}}(D_{t-1,t}), \tilde{\mathcal{C}}_{2n}$ are defined as follows:

$$\tilde{p}_{\text{NML}}(D_{t-1,t}) \overset{\text{def}}{=} \frac{\max_{\mathcal{M}} \mathrm{e}^{-L_{\text{NML}}(D_{t-1,t}; \mathcal{M}) - \mathcal{L}(\mathcal{M}, \mathcal{M})}}{\tilde{\mathcal{C}}_{2n}} \tag{4.8}$$

$$\tilde{\mathcal{C}}_{2n} \overset{\text{def}}{=} \int_{D_{t-1,t}} \max_{\mathcal{M}} \mathrm{e}^{-L_{\text{NML}}(D_{t-1,t}; \mathcal{M}) - \mathcal{L}(\mathcal{M}, \mathcal{M})}. \tag{4.9}$$

Here, the proofs for the bounds can be derived in the same way as in [43]. We describe the detail proofs in Appendix A.2. As seen in this theorem, we can reduce the Type-II error probability under the condition below:

$$2\alpha(1-\alpha) d_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*) > \frac{\alpha \ell_n(\mathcal{M}_1^*, \mathcal{M}_2^*, \epsilon)}{n}. \tag{4.10}$$

As can be seen from Theorems 6 and 7, the sensitivity of the change point detection using SMCS can be controlled by setting the value of the adjustment parameter ϵ.

4.2.2 Calculation of a Suitable Parameter ϵ

We consider how to set the parameter ϵ. In practical applications, it is important to define a criterion so as to suppress the error described above to a certain extent, in order to prevent false detection. In this regard, we make conditions that the error probability is less than a value δ. We will discuss Type-I error probability in consideration of lowering false alarms so that there are not too many change detection points.

To make Type-I error probability less than a value δ, we aim to make an upper bound on Type-I error probability less than the value δ as follows:

$$\exp\left\{-n\left(\epsilon - \frac{\log \mathcal{C}_{2n}(\mathcal{M}_0^*) + \mathcal{L}(\mathcal{M}_0^*, \mathcal{M}_0^*)}{n}\right)\right\} \leq \delta. \tag{4.11}$$

This condition (4.11) can be expressed again as follows:

$$\begin{aligned} &\exp\left\{-n\left(\epsilon - \frac{\log \mathcal{C}_{2n}(\mathcal{M}_0^*) + \mathcal{L}(\mathcal{M}_0^*, \mathcal{M}_0^*)}{n}\right)\right\} \leq \delta \\ \Leftrightarrow \quad & \epsilon \geq \frac{\log \mathcal{C}_{2n}(\mathcal{M}_0^*) + \mathcal{L}(\mathcal{M}_0^*, \mathcal{M}_0^*) - \log\delta}{n}. \end{aligned} \tag{4.12}$$

By setting δ in the range of $0 < \delta < 1$, the range of the adjustment parameter ϵ can be lower bounded by Equation (4.12), and the sensitivity of change detection can be thereby controlled.

4.3 Experimental Results

In this section, we present the experimental results of SMCS for detecting change points and their early warning signals. In the experiments, we use SMCS defined by Equation (4.2).

4.3.1 Deciding the Suitable Parameter ϵ

We simulates the suitable value of the adjustment parameter ϵ as in Equation (4.12). We simulate using the following conditions: the total number of dataset $=$ 1000, the number of clusters (Model class $\mathcal{M}_0^*$) $= 2$. We set δ in the range of $10^{-10} \leq \delta \leq 10^{-1}$ and investigate the change in ϵ. Here, we calculate adjustment parameter ϵ only from the viewpoint of the Type-I error probability. The result of this simulation is depicted in Figure 4.1. In the experiments, we calculate the suitable parameter ϵ at each time t assuming that $\mathcal{M}_0^* = \hat{\mathcal{M}}_{t-1}$, where $\hat{\mathcal{M}}_{t-1}$ is the estimated model at time $t-1$.

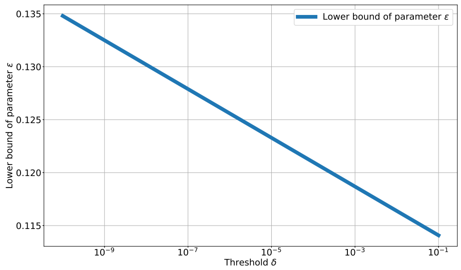


Figure 4.1: Suitable parameter ϵ related to threshold δ (in terms of the Type-I error probability).

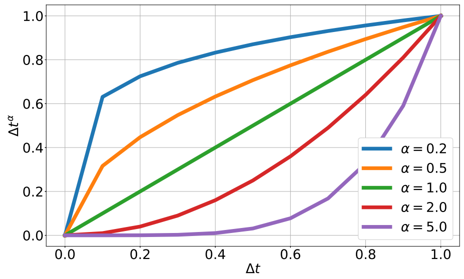
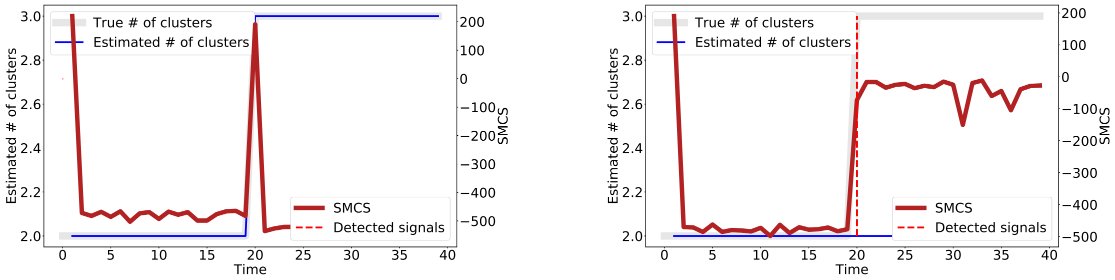


Figure 4.2: Characteristics of change with α.



(a) Dataset 0.

(b) Dataset 3.

Figure 4.3: Experiment for the data sequence using the GMM model. The true discrete change point is $t = 20$, and the centers of the clusters abruptly change as with Equation (4.13). This graph depicts the values of Φ_t and the estimated number of clusters at each time t.

4.3.2 GMM for Artificial Dataset

Here, we conducted an experiment using a synthetic dataset of the GMM. We created a synthetic data sequence and evaluated the usefulness of the proposed algorithm. In GMM, we consider the number of clusters as a model and evaluate its changes. As our algorithm can detect discrete change points and their early warning signals, we demonstrated the algorithm with two change patterns: abrupt change and gradual change.

Because the SDMS [9] and tracking the best expert (TBE) [8] methods can also be used to define model selection as the problem for deciding the number of clusters of GMM, we introduce these two methods as comparison targets for detecting a discrete change. In addition to these methods, we introduce structural entropy (SE) [11] as comparison targets for detecting the early warning signals.

Table 4.1: TPR and FAR scores for algorithms: SMCS, SDMS, and TBE (GMM).

Methods	TPR	FAR
SMCS (a_1)	0.3±0.5	0.02±0.01
SMCS ($a_1 \cup a_2$)	1.0±0.0	0.02±0.02
SDMS	1.0±0.0	0.00±0.00
TBE	1.0±0.0	0.03±0.00

Abrupt Change Pattern

Here, we created a dataset that changes abruptly in a time series. We define a "change" as a change in the number of clusters, where the center of the generated cluster abruptly changes as follows:

$$\begin{cases} K^* = 2,\ \mu = (\mu_1, \mu_2) & \text{if } 1 \leq t \leq \tau_1, \\ K^* = 3,\ \mu = (\mu_1, \mu_2, \mu_3) & \text{if } \tau_1 + 1 \leq t \leq T. \end{cases} \tag{4.13}$$

We aimed to detect the discrete change points using our proposed algorithm.

We generated a dataset that satisfied Equation (4.13) and detected discrete changes with the $a_1(t)$ in Algorithm 2. The values of Φ_t at each time t and the discrete change points detected were calculated as depicted in Figure 4.3.

In addition, we calculated the "true positive rate (TPR)" and "false alarm rate (FAR)" scores to evaluate whether the algorithm could detect the discrete changes. These scores are defined in Equation (4.14) and Equation (4.15), respectively:

$$\text{TPR} \overset{\text{def}}{=} \begin{cases} 1 & \text{if } \hat{t} = t^*, \\ 0 & \text{otherwise.} \end{cases}, \tag{4.14}$$

$$\text{FAR} \overset{\text{def}}{=} \frac{|\{\hat{t} | \hat{t} = t^*\}|}{|\text{not change point}|}, \tag{4.15}$$

where $\hat{t}$ is the detected change point.

The result is depicted in Figure 4.3. We found that the number of clusters changed at time $t = 20$ as in Figure 4.3(a). In Figure 4.3(b), the change point is not clearly detected, but the change statistic exhibits a high value. It can be understood that the change degree is large. Thus, by continuously grasping the degree of change, it is possible to recognize the degree of changes even when a discrete change cannot be detected.

In the evaluation, we calculated how well the proposed method worked in comparison with change point detection algorithms, such as SDMS and TBE. The results are listed in Table 4.1. We generated 10 different datasets for the same model and detected the early warning signals for each. The values in the table are the average and standard deviation values of TPR and FAR. These results demonstrate that the proposed algorithm detected discrete changes. However, by introducing the adjustment parameter ϵ, false positives were suppressed, but as seen in Figure 4.3(b), changes were judged more severely. For this reason, for some cases it was judged that there was no change (see SMCS (a_1)). Even in cases where there was no change (7 out of 10 cases), the value of Φ_t changes abruptly as depicted in Figure 4.3(b), which can be detected as an early warning signal of the change. Therefore, it can be considered that the feature of the change was captured even in a discrete change by using both $a_1(t)$ and $a_2(t)$ of Algorithm 2 as shown in Table 4.1 with SMCS ($a_1 \cup a_2$).

Gradual Change Pattern

Here, we created a dataset that changes gradually in a time series.

$$\begin{cases} K^* = 2,\ \mu = (\mu_1, \mu_2) & \text{if } 1 \leq t \leq \tau_1, \\ K^* = 3,\ \mu = (\mu_1, \mu_2, u(t)) & \text{if } \tau_1 + 1 \leq t \leq \tau_2, \\ K^* = 3,\ \mu = (\mu_1, \mu_2, \mu_3) & \text{if } \tau_2 + 1 \leq t \leq T, \end{cases} \tag{4.16}$$

where the function $u(t)$ is defined as follows:

$$\begin{aligned} u(t) &\overset{\text{def}}{=} (1 - \Delta t^{\alpha}) \cdot \mu_2 + \Delta t^{\alpha} \cdot \mu_3, \\ \Delta t &\overset{\text{def}}{=} \frac{t - \tau_1}{\tau_2 - \tau_1}, \end{aligned}$$

where α is a parameter that determines the characteristics of the change as shown in Figure 4.2. An example of the time series data with gradual change is shown in Figure 4.4.

We generated a dataset as in Equation (4.16) and detected discrete changes with $a_1(t)$, and detected their early warning signals with $a_2(t)$ in Algorithm 2 (SMCS). The values of Φ_t at each time t and the early warning signals detected were shown in Figure 4.5. In this figure, the discrete change points detected by SMCS algorithm also plotted. A discrete change point was defined as the point for which $\Phi_t \geq 0$. As seen in this figure, SMCS detected the discrete change points and their early warning signals.

In addition, we calculated the benefit, delay, and false alarm rate (FAR) scores to evaluate whether the algorithm detected the early warning signals. As it is difficult to define true discrete change points, we only evaluate how early we detected for the early warning signals. These scores are defined in Equation (4.17), Equation (4.18), and Equation (4.19), respectively:

$$\text{benefit} \overset{\text{def}}{=} \begin{cases} 1 - (\hat{t} - t^*)/U & \text{if } t^* \leq \hat{t} \leq t^* + U, \\ 0 & \text{otherwise.} \end{cases}, \tag{4.17}$$

Table 4.2: Benefit, delay, and FAR scores for algorithms: SMCS, SE, SDMS, and TBE (GMM). In this experiment, we set the parameter $\alpha = 1.0$.

Methods	benefit	delay	FAR
SMCS	0.840±0.062	3.200±1.249	0.005±0.014
SE	0.675±0.025	6.500±0.500	0.000±0.000
SDMS	0.550±0.039	9.000±0.775	0.000±0.000
TBE	0.445±0.042	11.100±0.831	0.000±0.000

$$\mathrm{delay} \overset{\mathrm{def}}{=} \begin{cases} \hat{t} - t^* & \text{if } \hat{t} \in \text{Transition period}, \\ T & \mathrm{otherwise}. \end{cases}, \tag{4.18}$$

$$\mathrm{FAR} \overset{\mathrm{def}}{=} \frac{|\{\hat{t}|\hat{t} \in \text{not transition period}\}|}{|\text{not transition period}|}, \tag{4.19}$$

where $\hat{t}$ is the first point where the algorithm detects the early warning signals in the transition period, or $\hat{t}$ is the detected change point at a time other than the transition period. T is the length of transition period.

We evaluated how well the proposed method worked in comparison with change point detection algorithms, such as SE, SDMS, and TBE. The results are listed in Table 4.2. We generated 10 different datasets for the same model and made alarms for the early warning signals for each. The values in the table are the average and standard deviation values of benefit, delay, and FAR. These results demonstrate that Algorithm 2 (SMCS) can detect the early warning signals faster than the other methods in terms of both of benefit and delay scores.

Next, we evaluated the benefit score with various parameters α. We changed α as $[0.2, 0.5, 1.0, 2.0]$, generated the dataset using these parameters, and made alarms for the early warning signals. The result is presented in Table 4.3. As shown in this result, the early warning signal points shift backward as α increases, and the early warning signals can be detected stably regardless of the value of α by using SMCS.

4.3.3 Real Marketing Dataset

Here, we used a real marketing dataset, which has a time series data of consumers' beer purchasing behaviors, and evaluated the usefulness of our algorithm qualitatively. This dataset has features of $540 (= n)$ consumers and $4 (= m)$ type of beer (we denote these types as A to D), and tracks all purchasing behaviors of the consumers for approximately three months. At each time $t (t = 0, \cdots, T - \tau)$, the data points are defined as $\mathbf{x}_{i,t} \in \mathbb{R}^m$ $(i = 1, \cdots, n)$, and each data point describes the beer consumption from time t to

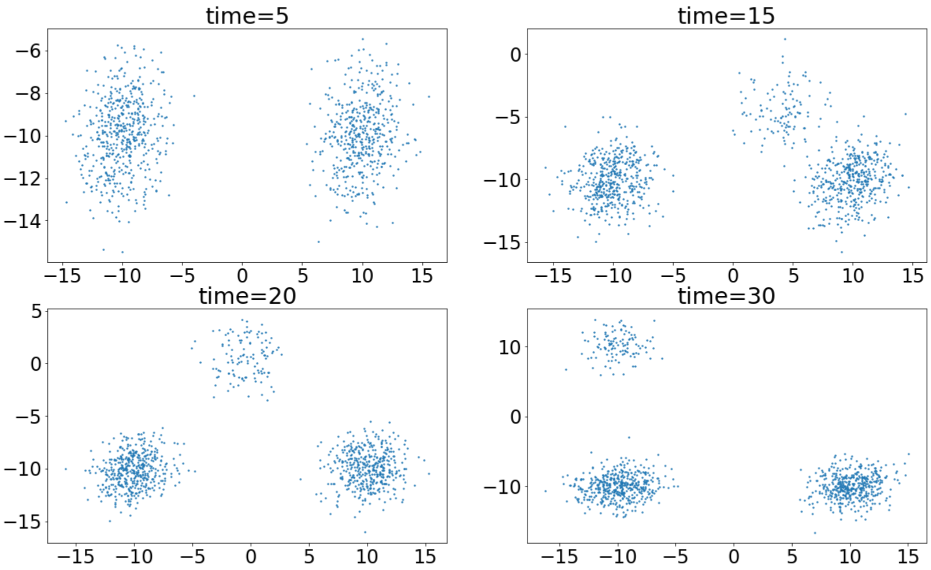
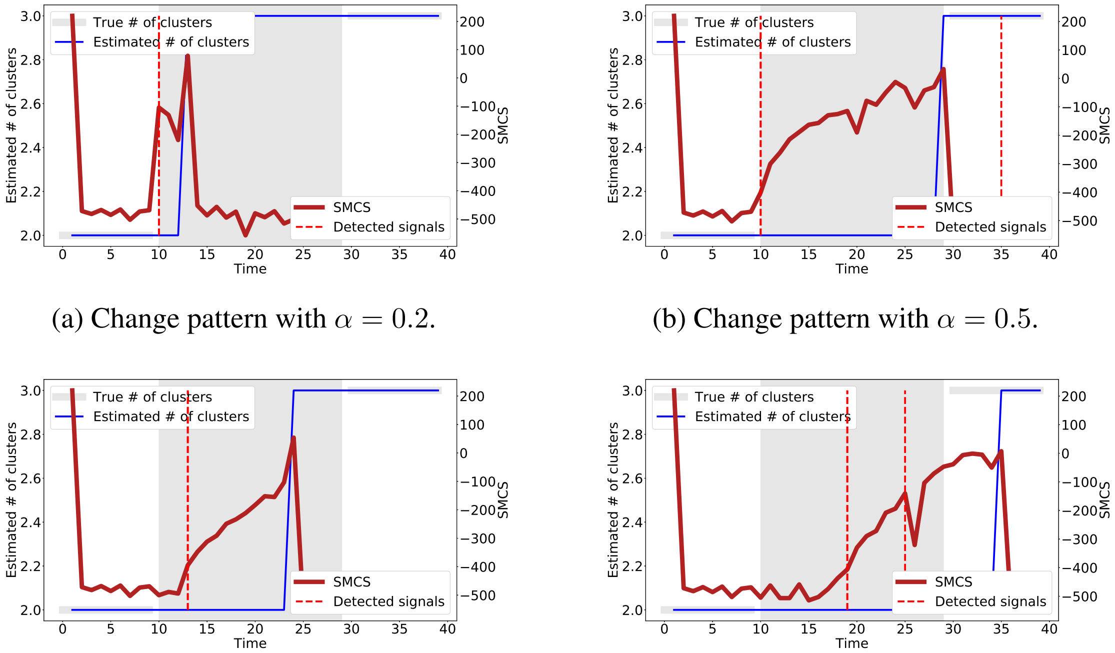


Figure 4.4: An example of time series data with gradual change.

$t+\tau-1$. We applied our method to the dataset with the parameter $\tau = 14, T = 92$ and a range of time from Nov. 14th to Jan. 31st.

Using SMCS, we detected the discrete change points in time $t = 2$ (Nov. 16th), $t = 6$ (Nov. 20th), and $t = 69$ (Jan. 22nd). By observing changes in SMCS as a whole, it can be seen that demand changed relatively frequently in the early days (November), and settled toward the end of the year. On January 22nd, the demand changed significantly and the number of clusters was increasing. At this time, as presented in Table 4.4, it can be considered that the number of users who purchased beer of types C and D at the same time was increasing to form a new cluster. The gradual change did not occur clearly; however, when observing time $t = 17$ (Dec. 1st), it seems that SMCS value increased slightly. It can be considered that the demand at this time needs to be carefully observed in addition to the change at the beginning of December, because the degree of change was a little greater due to the demand since the beginning of December. Qualitatively, the time point where SMCS value increased slightly may be considered as a change in the market accompanying year-end demand.



(a) Change pattern with $\alpha = 0.2$.

(b) Change pattern with $\alpha = 0.5$.

(c) Change pattern with $\alpha = 1.0$.

(d) Change pattern with $\alpha = 2.0$.

Figure 4.5: Experiment for the data sequence with multiple parameters α using the GMM model. The transition period is $10 \leq t \leq 29$. The centers of the clusters gradually change over time as in Equation (4.16). This graph depicts the values of Φ_t and estimated number of clusters at each time t. For example, in Figure 4.5(c), we detected the early warning signals at time $t = 13$ and find that the number of clusters changes at time $t = 24$. As seen in these figures, the smaller the α, the faster the change, and the algorithm can detect the early warning signals and the discrete change points.

4.3.4 Household Consumption Dataset

HWe employed the household consumption dataset, which has a time series data of power usage for a house, and evaluated the usefulness of our algorithm qualitatively. This dataset is available at [4]. There is one week's worth of power usage per hour. There are three types of power usage; kitchen, laundry room, and air-conditioner (see in detail at [4]). The dataset has all consumption behaviors for approximately three years. At each time $t(t = 0, \cdots, T - \tau)$, the data points are defined as $\mathbf{x}_{i,t} \in \mathbb{R}^3$ $(i = 1, \cdots, n)$. Each data point expresses the electric consumption from time t to $t + \tau - 1$ (Here, the unit time is one week.). We applied our method to the dataset with the parameter $\tau = 3$ (there are $504 (= n)$ data points at each time) and a range of time from Jan. 1st, 2007 to Dec. 31st, 2009.

Table 4.3: Benefit score for algorithms: SMCS, SE, SDMS, and TBE (GMM). In this experiment, we generated the dataset with various α.

Methods	α =0.2	α =0.5	α =1.0	α =2.0
SMCS	1.000±0.000	1.000±0.000	0.840±0.062	0.640±0.183
SE	0.900±0.300	0.905±0.035	0.675±0.025	0.430±0.024
SDMS	1.000±0.000	0.800±0.032	0.550±0.039	0.315±0.032
TBE	0.920±0.024	0.700±0.039	0.445±0.042	0.205±0.052

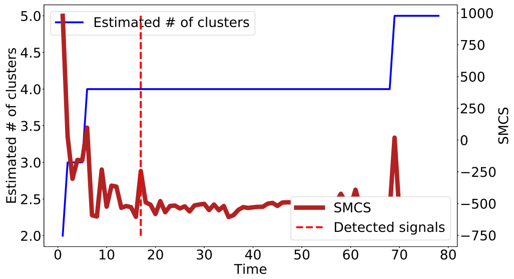


Figure 4.6: SMCS result for the real marketing dataset.

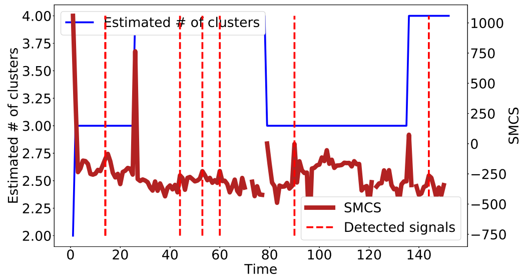


Figure 4.7: SMCS result for the household consumption dataset.

Using SMCS, we detected the discrete change points in time $t = 2$ (Feb. 4th, 2007), $t = 26$ (July 22nd, 2007), $t = 79$ (July 27th, 2008), and $t = 136$ (Aug. 30th, 2009). However, as SMCS values were not stably low, model selection at each time was difficult. Table 4.5 shows the cluster at time $t = 136$ when the model change was detected. It was thought that a new cluster was formed because there were many times when power was hardly used in the house. Looking at the detected warning signals, at the time $t = 90$ (Oct. 12th, 2008) for example, the times in which only air conditioning was used slightly increased in comparison with the previous time. From this result, in addition to detecting discrete changes, we were able to discover qualitatively meaningful times by observing early warning signals made by SMCS.

4.4 Conclusion

We have proposed SMCS, a unifying framework for detecting model changes and their early warning signals. In detecting the discrete change points, we have intended to produce reliable and stable alarms. For this purpose, we have evaluated the error probability rates and proposed a method to determine a suitable parameter. In our experiment, we

Table 4.4: Estimated clusters for time $t = 68$ to 69 (Real marketing dataset).

Time 68 (Jan. 21st)

type	clu-1	clu-2	clu-3	clu-4
A	2230	0	0	0
B	2	2308	0	0
C	13	20	3215	0
D	12	1	5	84
num	105	84	198	153

Time 69 (Jan. 22nd)

type	clu-1	clu-2	clu-3	clu-4	clu-5
A	2370	1	0	0	0
B	0	2187	0	0	0
C	6	26	346	0	2500
D	11	2	0	2848	1228
num	91	97	214	90	48

have simulated the suitable value of the adjustment parameter to produce a reliable detection alarm. The results of our evaluation have demonstrated the usefulness of the proposed method using an artificial dataset, real marketing dataset, and household consumption dataset. For the artificial dataset, we have demonstrated that the proposed SCMS method is able to simultaneously detect both discrete change points and their early warning signals. For the real marketing dataset and the household consumption dataset, we were able to detect meaningful change points, and some early warning signals. There are a number of issues regarding this SMCS method that will be considered in a future study. In terms of evaluation, we can define data types for which SMCS is effective by increasing the variation of the experiments. In addition, this method is potentially applied to other probability models (e.g., other mixture models), we consider an index that considers not only the previous time but also multiple times, and so on.

Table 4.5: Estimated clusters for time $t = 135$ to 136 (Household consumption dataset).

Time 135 (Aug. 23rd, 2009)

type	clu-1	clu-2	clu-3
Type-1	223	0	0
Type-2	23	37	0
Type-3	318	183	86
num	82	313	109

Time 136 (Aug. 30th, 2009)

type	clu-1	clu-2	clu-3	clu-4
Type-1	217	0	0	0
Type-2	17	33	0	4
Type-3	336	193	203	0
num	78	293	98	35

Chapter 5

Kernel Complexity

We consider the case where a dataset cannot be defined by a parametric model. Under this condition, we define a new index characterizing a structure of a nonparametric distribution and propose a method to detect its changes over time. This chapter, Section 5.1 expresses kernel complexity (KC) which is a new index for structural information and its nature; Section 5.2 discusses an algorithm for detecting changes and their early warning signals using KC. Lastly, we discuss experimental results in Section 5.3. We publish works of Section 5.1, Section 5.2, and Section 5.3 in preprint [14].

5.1 Kernel Complexity (KC)

5.1.1 Problem Setting

We consider a situation in which the distribution of the dataset X_t, which is observed at each time t, gradually changes over time. Each dataset X_t can be expressed as $X_t = \mathbf{x}^n = (\mathbf{x}_1, \cdots, \mathbf{x}_n)^\top \in \mathbb{R}^{n \times m}$, which consists of n data points of dimension m. We consider a situation where the distribution eof this dataset gradually changes over time. We aim to detect these changes. For this purpose, we think of the variation of the distribution as a sort of complexity and detect its changes. we measure the density variation as a complexity and detect change by detecting the change in density.

In this study, X_t is considered to follow a complex distribution. Thus, we aim to define the structure of the dataset without using a specific parametric distribution. In this section, for brevity, we express X_t as $X = \mathbf{x}^n = (\mathbf{x}_1, \cdots, \mathbf{x}_n)^\top \in \mathbb{R}^{n \times m}$. We do not use a specific parametric distribution, but rather we use the following kernel density:

$$p(\mathbf{x}; h) = \frac{1}{n} \sum_{j=1}^{n} K\left(\mathbf{x} - \mathbf{x}_j; h\right),$$

where $K(\cdot)$ is the kernel function. Specially we use the following Gaussian kernel:

$$K(\mathbf{x}; h) = \frac{1}{(2\pi h^2)^{m/2}} \exp\left\{-\frac{||\mathbf{x}||^2}{2h^2}\right\}.$$

5.1.2 Concept of the Proposed Method

Here we describe the concept of the proposed method. In order to measure the degree of concentration of a dataset at each time, we consider the amount of information contained in each data point included in the dataset. For example, we consider the density distribution in Figure 5.1(a). The density distribution indicated by the solid blue line is expected to be characterized by two peaks of density. Conversely, the structure of the density distribution indicated by the dotted orange lines can be characterized by extensive data points. To understand the structure of the density distribution in the figure, an index which defines spread of data would be useful to define a distribution. Then, we derive an index of the structural information termed KC.

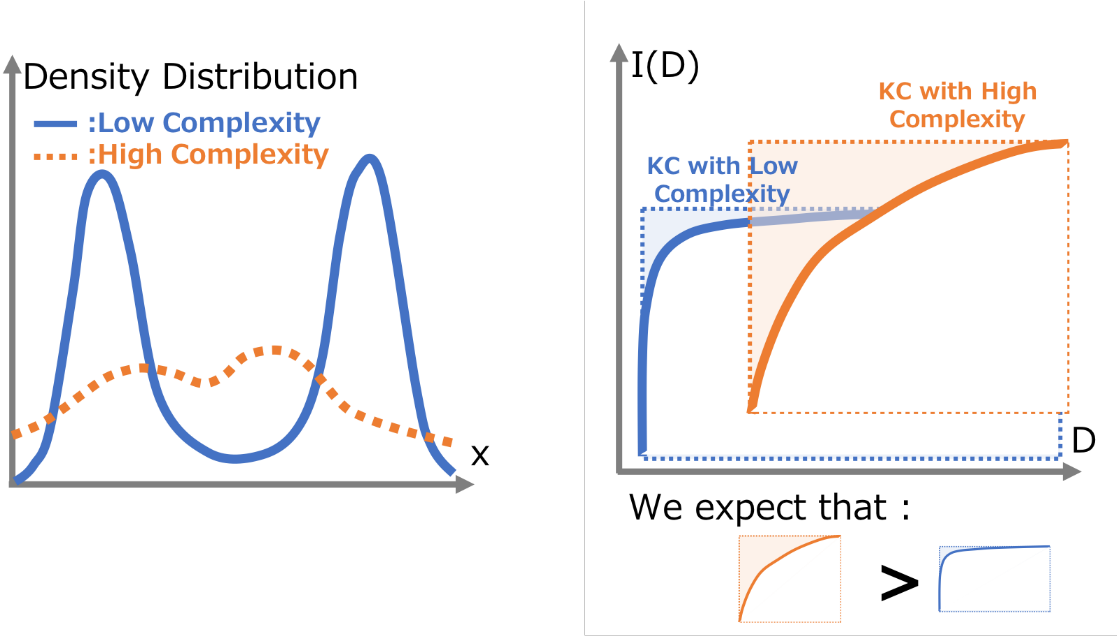
We measure the amount of information of the data in terms of MDL princilple as described in Chapter 2. In our setting, we employ Gaussian kernel density as a nonparametric model of the dataset. We then calculate the amount of information in terms of the NML code length of data associated with the class of Gaussian kernel densities.

The main process for defining KC can be summarized as follows:

1. We consider a subset of the dataset which defined by parameter D.
2. We calculate the amount of information for the subset with NML code length.
3. We measure the complexity of the distribution in terms of how the amount of information changes with respect to D.

5.1.3 Kernel Complexity

We propose an index that defines the complexity of a distribution in terms of its density. The index is illustrated in Figure 5.1b. We hypothesize that KC is relatively small when data points are densely distributed (solid blue line). In contrast, KC is relatively large when data points are sparsely distributed (solid orange line). In Figure 5.1b, the amount of information (denoted by I) increases with D. When the amount of information is biased for the data points in a specific dense area, I is initially considered to increase significantly as D increases, after which the change becomes gradual (see the blue line in Figure 5.1b). In comparison, when the amount of information varies throughout the dataset, I is considered to gradually increase as D increases (see the orange line in Figure 5.1b). In this way, this index can be understood as expressing the bias of the amount of



(a) Diagrammatic illustration of distribution complexity.

(b) Plot of the amount of information ($I(D)$) versus the parameter D.

Figure 5.1: Visualization of the concept of KC.

information possessed by each data point. By using the Gini coefficient [18], which is used to express the extent to which data is biased in economics etc., this index can be formulated as follows:

$$KC \overset{\text{def}}{=} 1 - \frac{\int_{\mathcal{D}} I(D)\,\mathrm{d}D - \frac{1}{2}\Delta L \cdot \Delta D - L_0 \cdot \Delta D}{\frac{1}{2}\Delta L \cdot \Delta D}, \tag{5.1}$$

where $I(D)$ is a function that defines the amount of information with D.

As the integral in Equation (5.1) is difficult to calculate using actual data, we approximate it as follows:

$$\int_{\mathcal{D}} I(D)\,\mathrm{d}D \approx \sum_{\ell_p \leq D_{\max}} (\ell_p - \ell_{p-1}) I(\ell_p),$$

where ℓ_p is the p-th numerical value with D at equal intervals.

5.1.4 NML Code Length associated with Kernel Density

In this section, we describe the NML code length for kernel density.

NML Code Length associated with Kernel Density

Let an observed data sequence be $\mathbf{x}^n = (\mathbf{x}_1, \cdots, \mathbf{x}_n),\ \mathbf{x}_i \in \mathbb{R}^m$. We use kernel density for a given dataset as follows:

$$f(\mathbf{x}; h) = \frac{1}{n}\sum_{j=1}^{n} K\left(\mathbf{x} - \mathbf{x}_j; h\right),$$

where the function $K(\cdot)$ is a kernel function. In the following discussion, we use the Gaussian kernel $K(\mathbf{x}; h) = 1/(2\pi h^2)^{m/2} \exp\{-||\mathbf{x}||^2/2h^2\}$. Aiming to capture the structural changes, we consider calculating the NML code length of the subset defined below:

$$\begin{aligned} A_i &= \left\{ j \,\middle|\, ||\mathbf{x}_i - \mathbf{x}_j||^2 \le (1+\gamma)D \right\}\ (i = 1, \cdots, n), && (5.2)\\ B &= \left\{ i \,\middle|\, \frac{1}{N(A_i)} \sum_{j \in A_i} ||\mathbf{x}_i - \mathbf{x}_j||^2 \le D \right\}, && (5.3) \end{aligned}$$

where $\gamma (> 0)$ is a parameter. For this subset, we can derive the following theorem.

Theorem 8. *The NML codelength for the subprobability disribution associated with kernel density is caluclated as follows:*

$$\frac{nm}{2}\log\left\{\sum_{i\in B}\frac{1}{N(A_i)}\sum_{j\in A_i}||\mathbf{x}_i - \mathbf{x}_j||^2\right\} - \sum_{i\in B}\left(\frac{m}{m+4}\log n - m\log\epsilon\right) \\ +nm\log\left(\frac{n^{\frac{1}{m+4}}}{\epsilon}\right) + \log\log\left(\frac{2\pi D \cdot n^{\frac{2}{m+4}}}{m\epsilon^2}\right) + \frac{nm}{2}\log(\pi) - \log\Gamma\left(\frac{nm}{2}\right). \tag{5.4}$$

Hereinafter, this NML code length is expressed as $L_{\mathrm{K-NML}}(\mathbf{x}^n; \gamma, D)$.

There are two aspects of the proof. The first is that we introduce a subprobability distribution with kernel density. The second is that we propose a method for calculating the NML for the subprobability distribution of kernel density.

Proof. First, we consider a bandwidth estimator with the kernel density function. We derive the log-likelihood as follows:

$$\log f(\mathbf{x}^n; h) = -\sum_{i\in B}\log\left(N(A_i)(2\pi h^2)^{\frac{m}{2}}\right) + \sum_{i\in B}\log\sum_{j\in A_i}\exp\left\{-\frac{1}{2h^2}||\mathbf{x}_i - \mathbf{x}_j||^2\right\},$$

where $N(A_i)$ and $N(B)$ are the number of A_i and B sets, respectively. Then, using the inequality $\log\left(\frac{1}{n}\sum_{i=1}^{n} x_i\right) \geq \frac{1}{n}\sum_{i=1}^{n}\log x_i$, the lower bound of this log-likelihood can be calculated as follows:

$$\begin{aligned}
\log f(\mathbf{x}^n; h) &= -\sum_{i\in B}\log\left((2\pi h^2)^{\frac{m}{2}}\right) + \sum_{i\in B}\log\left(\frac{1}{N(A_i)}\sum_{j\in A_i}\exp\left\{-\frac{1}{2h^2}||\mathbf{x}_i - \mathbf{x}_j||^2\right\}\right) \\
&\geq -\sum_{i\in B}\log\left((2\pi h^2)^{\frac{m}{2}}\right) - \sum_{i\in B}\frac{1}{N(A_i)}\sum_{j\in A_i}\frac{1}{2h^2}||\mathbf{x}_i - \mathbf{x}_j||^2. \qquad (5.5)
\end{aligned}$$

The bandwidth h of a kernel density was optimized to be $h = O(n^{-1/(m+4)})$ in the past work (e.g., refer to [36]) so that the generalization error for the maximum likelihood estimator for h is minimal. When considering the NML code length associated with the kernel, we set a constraint of $h \geq \epsilon \cdot n^{-1/(m+4)}/\sqrt{2\pi}$ using a positive constant ϵ. Under this condition, Equation (5.5) can be lower-bounded as follows:

$$\begin{aligned}
& \text{Equation}(5.5) \\
= \;& -N(B)\log\left((2\pi h^2)^{\frac{m}{2}}\right) - \sum_{i\in B}\frac{1}{N(A_i)}\sum_{j\in A_i}\frac{1}{2h^2}||\mathbf{x}_i - \mathbf{x}_j||^2 \\
= \;& -N(B)\log\left(\left(2\pi h^2 \cdot n^{\frac{2}{m+4}} \cdot \frac{1}{\epsilon^2}\right)^{\frac{m}{2}}\right) \\
& -\sum_{i\in B}\left\{\frac{1}{N(A_i)}\sum_{j\in A_i}\frac{1}{2h^2}||\mathbf{x}_i - \mathbf{x}_j||^2 - \frac{m}{m+4}\log n + m\log\epsilon\right\} \\
\geq \;& -n\log\left(\left(2\pi h^2 \cdot n^{\frac{2}{m+4}} \cdot \frac{1}{\epsilon^2}\right)^{\frac{m}{2}}\right) \\
& -\sum_{i\in B}\left\{\frac{1}{N(A_i)}\sum_{j\in A_i}\frac{1}{2h^2}||\mathbf{x}_i - \mathbf{x}_j||^2 - \frac{m}{m+4}\log n + m\log\epsilon\right\} \\
=: \;& \tilde{L}(\mathbf{x}^n; h). \qquad (5.6)
\end{aligned}$$

Then, we calculate the estimator $\tilde{h}$ such that $\tilde{L}(\mathbf{x}^n; h)$ is maximized as follows:

$$\tilde{h}(\mathbf{x}^n) = \sqrt{\frac{1}{nm}\sum_{i\in B}\frac{1}{N(A_i)}\sum_{j\in A_i}||\mathbf{x}_i - \mathbf{x}_j||^2}.$$

We define the distribution $\tilde{f}$ as follows:

$$\begin{aligned}
& \tilde{f}(\mathbf{x}^n; h) \\
\overset{\text{def}}{=} \;& \exp\left\{\tilde{L}(\mathbf{x}^n; h)\right\}
\end{aligned}$$

$$= \frac{1}{\left(2\pi h^2 \cdot n^{\frac{2}{m+4}} \cdot \frac{1}{\epsilon^2}\right)^{\frac{nm}{2}}} \exp\left\{-\frac{1}{2h^2}\sum_{i\in B}\frac{1}{N(A_i)}\sum_{j\in A_i}||\mathbf{x}_i - \mathbf{x}_j||^2\right\}$$
$$\times \exp\left\{\sum_{i\in B}\left(\frac{m}{m+4}\log n - m\log\epsilon\right)\right\}.$$

Using this equation and Equation (5.6), we find that the distribution $\tilde{f}$ is a subprobability distribution by the following formula:

$$\int \tilde{f}(\mathbf{x}^n; h)\mathrm{d}\mathbf{x}^n \leq \int f(\mathbf{x}^n; h)\mathrm{d}\mathbf{x}^n = 1.$$

Then, the NML distribution of $\tilde{f}$ can be calculated as follows:

$$\tilde{f}_{\text{NML}}(\mathbf{x}^n) \overset{\text{def}}{=} \frac{\tilde{f}(\mathbf{x}^n; \tilde{h}(\mathbf{x}^n))}{\mathcal{C}},$$
$$\mathcal{C} \overset{\text{def}}{=} \int \tilde{f}(\mathbf{y}^n; \tilde{h}(\mathbf{y}^n))\mathrm{d}\mathbf{y}^n,$$

where $\mathcal{C}$ is the normalization term of the NML distribution. In general, the normalization term cannot be calculated in a straightforward manner; instead, we calculate this term using the method described in Section 2.1.2. We can decompose $\tilde{f}(\mathbf{x}^n; h)$ as follows:

$$\tilde{f}(\mathbf{x}^n; h)\,\mathrm{d}\mathbf{x}^n = \bar{f}(z|\tilde{h}) \cdot g(\tilde{h}; h)\,\mathrm{d}z\,\mathrm{d}\tilde{h},$$

where the function g is the gamma distribution with the shape parameter $k = nm/2$ and scale parameter $\theta = 2h^2/nm$:

$$g(\tilde{h}; h) = \frac{2\tilde{h}}{\Gamma\left(\frac{nm}{2}\right)\left(\frac{2h^2}{nm}\right)^{\frac{nm}{2}}}\left(\tilde{h}^2\right)^{\frac{nm}{2}-1}\exp\left\{-\frac{\tilde{h}^2}{2h^2/nm}\right\}.$$

Then, we can define

$$g(\tilde{h}) \overset{\text{def}}{=} g(\tilde{h}; \tilde{h}) = \frac{2\exp\left(-\frac{nm}{2}\right)}{\Gamma\left(\frac{nm}{2}\right)\left(\frac{2}{nm}\right)^{\frac{nm}{2}}} \cdot \frac{1}{\tilde{h}}.$$

Then, we can calculate the normalization term $\mathcal{C}$ for integrating with respect to $\tilde{h}$ as follows:

$$\mathcal{C} = \int_Y \tilde{f}(\mathbf{y}^n; \tilde{h}(\mathbf{y}^n))\mathrm{d}\mathbf{y}^n$$
$$= \int_{\epsilon\cdot n^{-1/(m+4)}/\sqrt{2\pi}}^{\sqrt{D/m}} g(\tilde{h})\mathrm{d}\tilde{h}$$

$$= \frac{2\exp\left(-\frac{nm}{2}\right)}{\Gamma\left(\frac{nm}{2}\right)\left(\frac{2}{nm}\right)^{\frac{nm}{2}}} \cdot \log\left(\sqrt{\frac{2\pi D \cdot n^{\frac{2}{m+4}}}{m\epsilon^2}}\right),$$

where we define the range of $\tilde{h}$ as $Y = [1/\sqrt{2\pi}, \sqrt{D/m}]$. The upper bound on $\tilde{h}$ is calculated as follows:

$$\begin{aligned}
\tilde{h}(\mathbf{x}^n) &= \sqrt{\frac{1}{nm}\sum_{i\in B}\frac{1}{N(A_i)}\sum_{j\in A_i}||\mathbf{x}_i - \mathbf{x}_j||^2} \\
&\leq \sqrt{\frac{1}{nm}\sum_{i\in B} D} \\
&\leq \sqrt{\frac{D}{m}}.
\end{aligned}$$

Finally, we can calculate the the NML code length using the subprobability distribution $\tilde{f}$ as follows:

$$\begin{aligned}
&-\log \tilde{f}_{\mathrm{NML}}(\mathbf{x}^n) \\
= &-\log\left\{\frac{\exp(-\frac{nm}{2})\cdot\exp\left\{\sum_{i\in B}\left(\frac{m}{m+4}\log n - m\log\epsilon\right)\right\}}{\left(2\pi\tilde{h}^2\cdot n^{\frac{2}{m+4}}\cdot\frac{1}{\epsilon^2}\right)^{\frac{nm}{2}}}\right\} \\
&+\log\left\{\frac{2\exp\left(-\frac{nm}{2}\right)}{\Gamma\left(\frac{nm}{2}\right)\left(\frac{2}{nm}\right)^{\frac{nm}{2}}}\cdot\log\left(\sqrt{\frac{2\pi D\cdot n^{\frac{2}{m+4}}}{m\epsilon^2}}\right)\right\} \\
= &\frac{nm}{2}\log\left\{\sum_{i\in B}\frac{1}{N(A_i)}\sum_{j\in A_i}||\mathbf{x}_i-\mathbf{x}_j||^2\right\} - \sum_{i\in B}\left(\frac{m}{m+4}\log n - m\log\epsilon\right) \\
&+nm\log\left(\frac{n^{\frac{1}{m+4}}}{\epsilon}\right) + \log\log\left(\frac{2\pi D\cdot n^{\frac{2}{m+4}}}{m\epsilon^2}\right) + \frac{nm}{2}\log(\pi) - \log\Gamma\left(\frac{nm}{2}\right).
\end{aligned}$$

□

5.1.5 Kernel Complexity with the NML Code Length associated with the Kernel Density

We define KC by using the NML code length associated with the kernel density, which is calculated in Section 5.1.4. The definition of KC based on the NML code length is as follows:

$$KC_{\mathrm{K-NML}} \overset{\mathrm{def}}{=} 1 - \frac{\int_{\mathcal{D}} L_{\mathrm{K-NML}}(\mathbf{x}^n;\gamma,D)\,\mathrm{d}D - \frac{1}{2}\Delta L\cdot\Delta D - L_0\cdot\Delta D}{\frac{1}{2}\Delta L\cdot\Delta D}. \tag{5.7}$$

Let us discuss the theoretical nature of the NML code length $L_{\mathrm{K-NML}}(\mathbf{x}^n;\gamma,D)$. This NML code length contains the hyperparameters γ and D, and we evaluate the theoretical properties by tracking the change in the NML by D for fixed γ. First, we consider the case in which D changes by ΔD The amount of change in the NML code length can be approximated as follows:

$$\begin{aligned}\Delta L &= L_{\mathrm{K-NML}}(\mathbf{x}^n;\gamma,D+\Delta D)-L_{\mathrm{K-NML}}(\mathbf{x}^n;\gamma,D)\\ &\approx \frac{nm}{2}\cdot\frac{1}{1-\delta}\cdot\frac{N(\Delta B(D))}{N(B(D))},\end{aligned}\tag{5.8}$$

where $N(B(D))$ is the number of B sets in Equation (5.3), $\Delta B(D)$ is the increment of set B when D changes by ΔD. Additionally, we define a parameter $\delta, 0<\delta<1$. The calculation details are presented as follows:

Proof. We describe the approximation of ΔL discussed in Section 5.1.3. First, we can approximate ΔL as follows:

$$\begin{aligned}\Delta L &= L_{\mathrm{K-NML}}(\mathbf{x}^n;\gamma,D+\Delta D)-L_{\mathrm{K-NML}}(\mathbf{x}^n;\gamma,D)\\ &\approx \frac{nm}{2}\log\left\{\sum_{i\in B(D)}\frac{1}{N(A_i(D))}\sum_{j\in A_i(D)}||\mathbf{x}_i-\mathbf{x}_j||^2+\sum_{i\in\Delta B(D)}D\right\}\\ &\quad -\frac{nm}{2}\log\left\{\sum_{i\in B(D)}\frac{1}{N(A_i(D))}\sum_{j\in A_i(D)}||\mathbf{x}_i-\mathbf{x}_j||^2\right\},\end{aligned}\tag{5.9}$$

where $N(A_i(D))$ and $N(B(D))$ are the number of A_i and B sets in Equation (5.3), respectively, and $\Delta B(D)$ is the increment of B when changing D by ΔD. As we assume that ΔD is sufficiently small, we can naturally assume that

$$\sum_{i\in\Delta B(D)}D\ll\sum_{i\in B(D)}\frac{1}{N(A_i(D))}\sum_{j\in A_i(D)}||\mathbf{x}_i-\mathbf{x}_j||^2.$$

Using this assumption, we approximate Equation (5.9) as follows:

$$\text{Equation}(5.9)\approx\frac{nm}{2}\frac{\sum_{i\in\Delta B(D)}D}{\sum_{i\in B(D)}\frac{1}{N(A_i(D))}\sum_{j\in A_i(D)}||\mathbf{x}_i-\mathbf{x}_j||^2}.\tag{5.10}$$

Using a parameter $\delta, 0<\delta<1$, we can rewrite this equation as follows:

$$\sum_{i\in B(D)}\frac{1}{N(A_i(D))}\sum_{j\in A_i(D)}||\mathbf{x}_i-\mathbf{x}_j||^2=N(B(D))\cdot(1-\delta)D.\tag{5.11}$$

Using this formula, we can calculate Equation (5.10) as follows:

$$\begin{aligned}\text{Equation (5.10)} &= \frac{nm}{2}\frac{\sum_{i \in \Delta B(D)} D}{N(B(D)) \cdot (1-\delta) D} \\ &= \frac{nm}{2} \cdot \frac{1}{1-\delta} \cdot \frac{N(\Delta B(D))}{N(B(D))}. \qquad (5.12)\end{aligned}$$

□

Equation (5.8) is proportional to the rate of increase in B subject to the likelihood calculation. As the value of ΔL indicates the extent to which the data points subject to the code length increase as D increases, the width of the increase in the amount of information can be regarded as the amount of information of the newly added data points.

5.1.6 Property of KC

We calculated the value of KC for several generated datasets to evaluate the nature of KC. Specifically, we evaluated the behavior of the values of KC with respect to the number of sets (clusters in a parametric model) in the dataset and the behavior of the values of KC with respect to the extent of the dataset.

Aggregated Dataset

As a representative of a mixed dataset, we generated an artificial dataset, which we aggregated into several chunks we refer to as clusters. We used the same artificial dataset and performed the aggregation a few times to obtain a different number of clusters, and experimented with the characteristics of the KC values for each of the datasets we produced in this way. The different aggregations we generated are shown in Figure 5.2. Using this dataset, we calculated the value of KC, which is plotted in Figure 5.3 as a function of the number of clusters. This result indicates that KC increases as the number of clusters increases. Thus, from the viewpoint of data aggregation, KC is considered to capture the characteristics of the structure of the distribution. In addition to these results, we also experimented with the behavior of KC by changing the dimensions of the dataset. The results were shown in Figure 5.4(a) and Figure 5.4(b). These results showed that KC increases as the number of clusters increases, and demonstrated the ability of KC to grasp the structural information of data aggregation.

Circular Dataset

As nonparametric data cannot be expressed as a simple block, we investigated the characteristics of KC of a dataset of which the data are aggregated into circular chunks (clusters). As before, we varied the number of clusters, and investigated the extent to which the KC

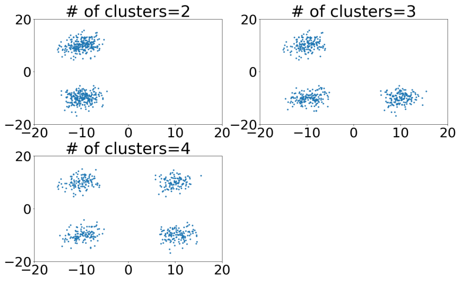


Figure 5.2: Dataset aggregated into several chunks, which we refer to as clusters.

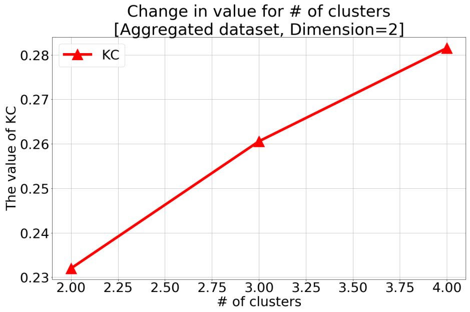
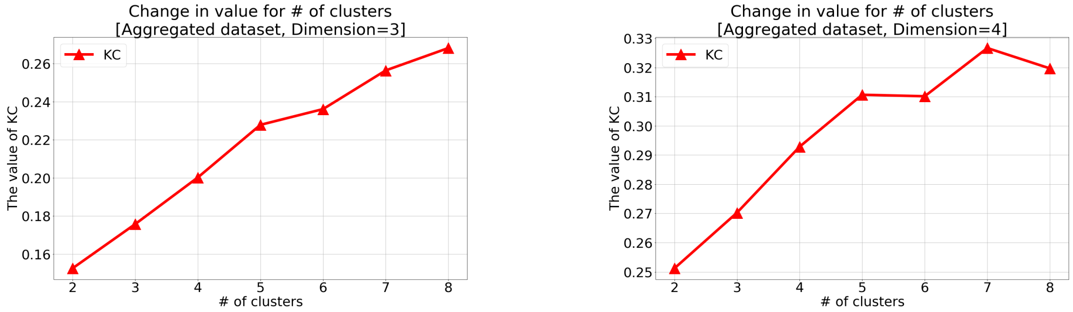


Figure 5.3: Value of KC as a function of the number of data chunks (clusters) in the dataset.



(a) Results for the dataset with a dimension of 3.

(b) Results for the dataset with a dimension of 4.

Figure 5.4: Value of KC for the dataset aggregated into several chunks (clusters) with a different number of dimensions.

value depends on the number of clusters. The dataset we generated is visualized in Figure 5.5. Using this dataset, the calculated values of KC are plotted in Figure 5.6 as a function of the number of circles (i.e., clusters). These results show that KC increases as the number of clusters increases; hence, KC is considered to capture the characteristics of the structure of the distribution in terms of the number of clusters.

5.2 Algorithm for Detecting Early Warning Signals of Changes using KC

Based on the above-mentioned findings, we proceeded to apply KC to further investigate its ability to detect structural changes in a dataset. A dataset that undergoes structural change first exhibits minor movement, after which a large change in the structure be-

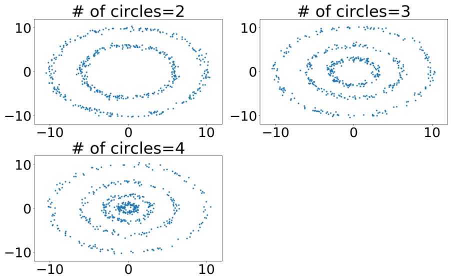


Figure 5.5: Circular dataset with various numbers of clusters.

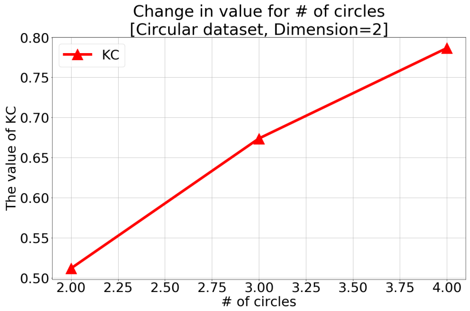


Figure 5.6: Value of KC for the circular dataset with aggregations into a different number of clusters.

comes visible. Because KC is an index for evaluating the complexity of the distribution of a dataset, it can be considered as an index of increasing complexity during substantial structural change. The algorithm we developed to compute the KC index for structural change detection is presented in Algorithm 3. For simplicity, we express the dataset as $X_t \in \mathbb{R}^{n\times m}$ and express $KC_{\mathrm{K-NML}}(X_t)$ as KC_t.

Algorithm 3 Algorithm for structural change detection.

Calculate KC:
Calculate KC score defined by Equation (5.7) as follows:

$$KC_{\mathrm{K-NML}}(X_t) = 1 - \frac{\int_{\mathcal{D}} L_{\mathrm{K-NML}}(X_t;\gamma, D)\,\mathrm{d}D - \frac{1}{2}\Delta L\cdot\Delta D - L_0\cdot\Delta D}{\frac{1}{2}\Delta L\cdot\Delta D}.$$

Detection of changes:
Detect changes with following conditions:

$$a(t) \;=\; \begin{cases} 1 & \text{if } |KC_t - E[KC^{t-1}]| > \eta * Std[KC^{t-1}], \\ 0 & \text{otherwise.} \end{cases}$$

We can detect changes in the case where $a(t) = 1$.

5.3 Experimental Results

We empirically demonstrated the usefulness of the algorithm using both artificial and practical datasets.

5.3.1 Artificial Dataset for Change Detection

Circular Dataset

We generated an artificial dataset distributed on the circumferences of several circles and considered the case in which the number of circles gradually changes over time. The parameters of this dataset are defined as follows:

$$\begin{cases} \text{\# of circles} = K,\ r = (r_1, r_2) & \text{if } 1 \le t \le \tau_1, \\ \text{\# of circles} = K',\ r = (r_1, r_2, u(t)) & \text{if } \tau_1 + 1 \le t \le \tau_2, \\ \text{\# of circles} = K',\ r = (r_1, r_2, r_3) & \text{if } \tau_2 + 1 \le t \le T, \end{cases} \tag{5.13}$$
$$\text{where } u(t) = \frac{(\tau_2 - t)r_2 + (t - \tau_1)r_3}{\tau_2 - \tau_1},$$

where the parameter r denotes the radius of each circle. An example of the generated dataset was shown in Figure 5.7.

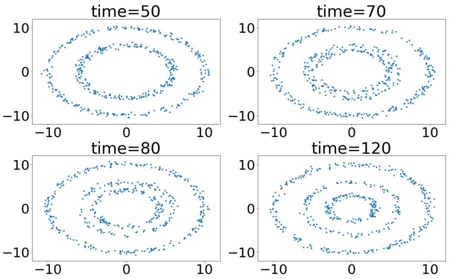


Figure 5.7: Dataset aggregated into circular data chunks over a period of time.

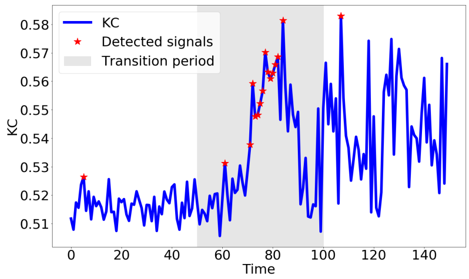


Figure 5.8: Calculated value of KC of the dataset aggregated into different circular chunks over time.

We found the points of change by using the index in Algorithm 3. The results were shown in Figure 5.8, which shows that KC is able to detect a transition in the structure of the data over time and this was interpreted as an increase in the complexity of the dataset.

In addition to the above qualitative interpretation, we calculated the benefit, delay, FAR, and AUC scores to evaluate the extent to which the algorithm detected changes. The benefit, delay, FAR, and (area under the curve) AUC scores, respectively, are defined as follows:

$$\text{benefit} \overset{\text{def}}{=} \begin{cases} 1 - (\hat{t} - t^*)/U & \text{if } t^* \le \hat{t} \le t^* + U, \\ 0 & \text{otherwise;} \end{cases} \tag{5.14}$$

$$\text{delay} \overset{\text{def}}{=} \begin{cases} \hat{t} - t^* & \text{if } \hat{t} \in \text{Transition period}, \\ None & \text{otherwise;} \end{cases} \tag{5.15}$$

$$\mathrm{FAR} \overset{\mathrm{def}}{=} \frac{\#\{\hat{t} \notin \text{Transition period}\}}{\#\{t \notin \text{Transition period}\}}, \tag{5.16}$$

$$\mathrm{AUC} \overset{\mathrm{def}}{=} \text{Area under the curve created by plotting the benefit against the FPR} \tag{5.17}$$

where $\hat{t}$ is the first point at which the algorithm detects a change in the transition period or the point at which the algorithm detects a change in any other period. In this study, the AUC was calculated in relation to the benefit. Using these evaluation scores, we evaluated the proposed algorithm in comparison with the density ratio estimation (DRE) algorithm [24], SDMS algorithm [9], SE algorithm [11], tracking the best expert (TBE) algorithm [8], and the entropy-based method (abbreviated as entropy), which is described in Algorithm 4. When using the DRE algorithm, we processed the two-dimensional data $X_t \in \mathbb{R}^{n \times m}$ at each time as one-dimensional data $X_t' \in \mathbb{R}^{nm}$ and used them as the input for the DRE algorithm. Even though this is not a perfect fit for the model of the DRE algorithm, it was added to the comparison as a model capable of detecting nonparametric change. The point at which the data complexity begins to rise is defined as the starting point of change in the model, and the proposed method was used to conduct a quantitative comparison experiment using the value of KC.

Algorithm 4 Entropy algorithm for detecting changes.

Calculate the density:
Calculate the density distribution as follows:

$$f(\mathbf{x}; \bar{h}) = \frac{1}{n} \sum_{i=1}^{n} K\left(\mathbf{x} - \mathbf{x}_i; \bar{h}\right),$$

where we calculate the estimator $\bar{h}$ on the basis of the method proposed by Scott [36], which is the default setting in the scipy.stats.gaussian_kde class (see [2]).

Calculate the entropy:
Calculate the entropy as follows:

$$\mathit{Entropy} = -\int f(\mathbf{y}; \bar{h}(\mathbf{x}^n)) \log f(\mathbf{y}; \bar{h}(\mathbf{x}^n)) \mathrm{d}\mathbf{y}.$$

Detection of changes:
Detect the change points with following conditions:

$$a(t) = \begin{cases} 1 & \text{if } |\mathit{Entropy}_t - E[\mathit{Entropy}^{t-1}]| > \eta * \mathit{Std}[\mathit{Entropy}^{t-1}], \\ 0 & \text{if otherwise.} \end{cases}$$

The results were listed in Table 5.1. We generated 10 different datasets with the same model and detected the points of change for each dataset. The values provided in the table are the average values of the benefit, delay, and FAR scores. These results showed that the proposed algorithm (KC) detected the points of change to a certain extent in terms of the benefit and delay scores. Although the FAR score of the proposed algorithm was slightly larger than that of the entropy method, this difference was insignificant. Because the SDMS, SE, and TBE algorithms are based on a parametric Gaussian mixture model (GMM), it was not possible to capture the changes in a circular distribution of data. The DRE algorithm assumes that the data at each time consist of only a single scalar value; thus, it was difficult to detect changes in this problem setting.

The main parameters of the generated data and algorithm are as follows.

- Dataset parameters:
  We generated a circular dataset with $K = 2, K' = 3$ clusters by using Equation (5.13). The radii of the circles were $r = (10, 6, 3)$. A chunk of data starts its transformation at time $t = 50 (= \tau_1 + 1)$ and finishes forming a new chunk of data (a circle with radius $r = 3$) at time $t = 100 (= \tau_2 + 1)$. Each data point contains noise that follows a normal distribution with the standard deviation $\sigma = 0.3$.

- Algorithm parameters:
  We used Algorithm 3 to calculate the index to determine the points of change. The detection parameter is $\eta = 3$, and the maximum value of D is 100. In the evaluation, we defined the parameter $U = 50$ to calculate the evaluation scores and the length of the transition period as 50.

Table 5.1: Benefit, delay, and FAR scores for the algorithms (Circular dataset).

Method	Benefit	Delay	FAR
KC	0.632	18.4	0.028
DRE	0.000	50.0	0.010
SDMS	0.000	50.0	0.000
SE	0.000	50.0	0.000
TBE	0.000	50.0	0.000
entropy	0.344	32.8	0.007

In addition to these results, we evaluated the experimental results by changing the noise level (from $\sigma = 0.1$ to $\sigma = 0.5$), focusing on the AUC scores. We calculated the AUC scores by changing the detection parameter η (from $\eta = 0.5$ to $\eta = 3.0$) in Algorithm 3. We generated aggregations with different patterns of the circular dataset with the following parameters:

1. The radii of the circles were $r = (10, 6, 3)$. The new data chunk is a circle with the radius $r_3 = 3$. The results are summarized in Table 5.2.

2. The radii of the circles were $r = (10, 3, 6)$. The new data chunk is a circle with the radius $r_3 = 6$. The results are summarized in Table 5.3.

3. The radii of the circles were $r = (3, 6, 10)$. The new data chunk is a circle with the radius $r_3 = 10$. The results are summarized in Table 5.4.

These results are similar to those in Table 5.1. On the basis of all the results, it is concluded that KC is able to detect changes stably, regardless of the difference in the generation model and the value of σ. The entropy method detected changes with the next highest score. However, as described in Section **??**, the entropy method requires computational time of an exponential order for the given dimensions, thus it may be difficult to apply in practice.

Table 5.2: AUC scores for the algorithms (circular dataset with $r = (10, 6, 3)$).

Method	σ =0.1	σ =0.2	σ =0.3	σ =0.4	σ =0.5
KC	**0.946±0.022**	**0.934±0.027**	0.918±0.030	**0.921±0.035**	**0.934±0.023**
DRE	0.495±0.000	0.495±0.000	0.495±0.000	0.495±0.000	0.495±0.000
SDMS	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
SE	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
TBE	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
entropy	0.930±0.041	0.928±0.050	**0.923±0.056**	0.911±0.057	0.920±0.050

Table 5.3: AUC scores for the algorithms (circular dataset with $r = (10, 3, 6)$).

Method	σ =0.1	σ =0.2	σ =0.3	σ =0.4	σ =0.5
KC	**0.964±0.018**	**0.965±0.015**	**0.968±0.017**	**0.966±0.018**	**0.969±0.015**
DRE	0.495±0.000	0.495±0.000	0.495±0.000	0.495±0.000	0.495±0.000
SDMS	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
SE	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
TBE	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
entropy	0.946±0.032	0.947±0.032	0.946±0.031	0.941±0.032	0.939±0.028

Table 5.4: AUC scores for the algorithms (circular dataset with $r = (3, 6, 10)$).

Method	σ =0.1	σ =0.2	σ =0.3	σ =0.4	σ =0.5
KC	**0.980±0.007**	**0.978±0.008**	**0.976±0.009**	**0.968±0.008**	**0.968±0.009**
DRE	0.495±0.000	0.495±0.000	0.495±0.000	0.495±0.000	0.495±0.000
SDMS	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
SE	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
TBE	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
entropy	0.973±0.011	0.965±0.017	0.961±0.016	0.962±0.017	0.952±0.016

Gamma-Distributed Dataset

We generated an artificial dataset that follows the gamma mixture model and considered the case in which the number of mixtures gradually changed over time. The parameters of this dataset are defined as follows:

$$\begin{cases} \text{\# of Gamma dist.} = K,\ k = (k_1, k_2) & \text{if } 1 \leq t \leq \tau_1, \\ \text{\# of Gamma dist.} = K',\ k = (k_1, k_2, u(t)) & \text{if } \tau_1 + 1 \leq t \leq \tau_2, \\ \text{\# of Gamma dist.} = K',\ k = (k_1, k_2, k_3) & \text{if } \tau_2 + 1 \leq t \leq T, \end{cases} \tag{5.18}$$
$$\text{where } u(t) = \frac{(\tau_2 - t)k_2 + (t - \tau_1)k_3}{\tau_2 - \tau_1},$$

where the parameter k denotes the shape of the gamma distribution. An example of the generated dataset is shown in Figure 5.9, which shows that data points that follow the gamma distribution with $k = 1$ are gradually generated over time.

We used Algorithm 3 to calculate the index to determine the points of change. The results are shown in Figure 5.10, which shows that KC gradually changes during the transition period and that it detected the changes at the beginning of the transition period.

In addition, we evaluated the experimental results by changing the scale parameter (from $\theta = 0.4$ to $\theta = 2.0$), by focusing on the AUC defined in Equation 5.17. We generated the dataset with the gamma distribution with different data patterns, using the following parameters:

1. The shape parameters of the gamma distribution were $k = (10, 5, 1)$. The new data chunk is a cluster with the shape parameter $k_3 = 1$. The results are summarized in Table 5.5.

2. The shape parameters of the gamma distribution were $k = (10, 1, 5)$. The new data chunk is a cluster with the shape parameter $k_3 = 5$. The results are summarized in Table 5.6.

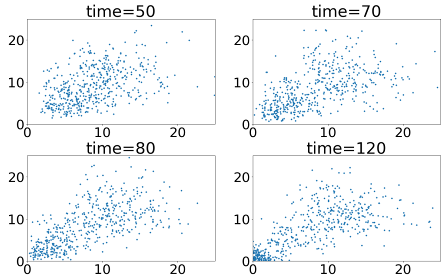


Figure 5.9: Dataset with the gamma-distributed data pattern evolving over time.

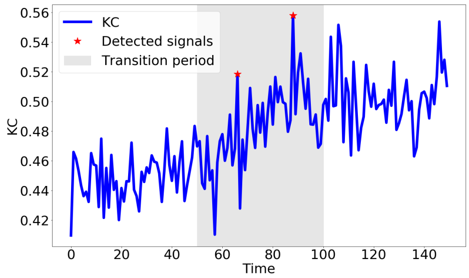


Figure 5.10: Value of KC over time for the gamma-distributed data pattern.

3. The shape parameters of the gamma distribution are $k = (1, 5, 10)$. The new data chunk is a cluster with the shape parameter $k_3 = 10$. The results are summarized in Table 5.7.

These results indicate that the value of KC and the entropy can be used to detect changes stably and effectively. The DRE and SE produced relatively good results, confirming that the gamma-distributed dataset is a model that easily enables both nonparametric and parametric changes to be detected. It should be noted that the value of KC and the AUC scores for KC vary depending on the scale parameter. This is probably because a constant value is used in this experiment. This suggests that the integration range of D should be appropriately selected according to the data spread; this is left for future study. In addition, the time required to calculate the entropy is of the exponential order with respect to the data dimension; hence, in practice, the calculation is limited to two dimensions. Therefore, this experiment was conducted using only two-dimensional data, and the behavior of KC when processing high-dimensional data is left for future study.

Table 5.5: AUC scores for the algorithms (gamma dataset with shape parameters $k = (10, 5, 1)$).

Method	θ =0.4	θ =0.8	θ =1.2	θ =1.6	θ =2.0
KC	0.873±0.021	0.876±0.020	**0.961±0.006**	**0.959±0.006**	0.948±0.018
DRE	0.900±0.105	0.900±0.105	0.900±0.105	0.900±0.105	0.900±0.105
SDMS	0.510±0.046	0.510±0.046	0.510±0.046	0.510±0.046	0.510±0.046
SE	0.903±0.002	0.903±0.002	0.903±0.002	0.903±0.002	0.903±0.002
TBE	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
entropy	**0.952±0.009**	**0.949±0.011**	0.950±0.008	0.950±0.008	**0.949±0.007**

Table 5.6: AUC scores for the algorithms (gamma dataset with shape parameters $k = (10, 1, 5)$).

Method	θ =0.4	θ =0.8	θ =1.2	θ =1.6	θ =2.0
KC	**0.965±0.018**	0.900±0.005	**0.940±0.030**	**0.977±0.006**	0.879±0.037
DRE	0.858±0.133	0.858±0.133	0.858±0.133	0.858±0.133	0.858±0.133
SDMS	0.543±0.136	0.543±0.136	0.543±0.136	0.543±0.136	0.543±0.136
SE	0.934±0.005	**0.934±0.005**	0.934±0.005	0.934±0.005	**0.934±0.005**
TBE	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
entropy	0.928±0.020	0.921±0.027	0.917±0.023	0.914±0.023	0.914±0.023

Table 5.7: AUC scores for the algorithms (gamma dataset with shape parameters $k = (1, 5, 10)$).

Method	θ =0.4	θ =0.8	θ =1.2	θ =1.6	θ =2.0
KC	0.838±0.020	0.893±0.005	0.945±0.009	**0.974±0.007**	**0.975±0.007**
DRE	0.805±0.228	0.805±0.228	0.805±0.228	0.805±0.228	0.805±0.228
SDMS	0.510±0.046	0.510±0.046	0.510±0.046	0.510±0.046	0.510±0.046
SE	0.910±0.005	0.910±0.005	0.910±0.005	0.910±0.005	0.910±0.005
TBE	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
entropy	**0.968±0.016**	**0.970±0.015**	**0.972±0.013**	0.973±0.012	0.973±0.012

Cross Dataset

We generated an artificial dataset distributed along straight lines for several chunks of data points and considered the case in which the number of chunks gradually changed over time. The parameters of this dataset are defined as follows:

$$\begin{aligned}\text{\# of lines} &= K \ (1 \le t \le T),\\ a &= (a_1, u(t)) \ (1 \le t \le T),\\ u(t) &= \frac{(T-t+1)\cdot a_1 + (t-1)\cdot a_2}{T}.\end{aligned}$$

We generated an artificial dataset with $K = 2$, $a_1 = -0.95$, and $a_2 = 0.95$, where the parameter a denotes the slope of a straight line. An example of the generated dataset is shown in Figure 5.11, which shows that one of the straight lines is gradually rotating over time.

We used the index in Algorithm 3 to detect the points of change. The results are shown in Figure 5.12, which indicates that KC is gradually decreasing. This is because the density of the distribution at the center of the distribution becomes relatively larger owing to the gradual movement of one of the straight lines, with an accompanying decrease in the complexity.

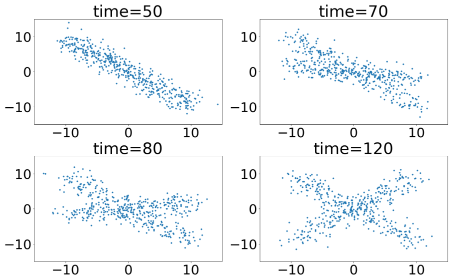


Figure 5.11: Dataset with data chunks in the form of a cross.

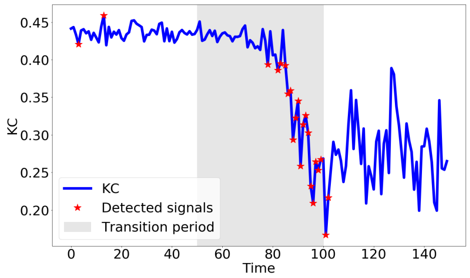


Figure 5.12: Value of KC over time for the dataset with datachunks in the form of a cross pattern.

In addition, we evaluated the experimental results by changing the noise level (from $\sigma = 0.5$ to $\sigma = 2.5$), by focusing on AUC defined in Equation 5.17.

Table 5.8: AUC scores for the algorithms (cross dataset).

Method	σ =0.5	σ =1.0	σ =1.5	σ =2.0	σ =2.5
KC	**0.908±0.007**	**0.912±0.005**	0.846±0.026	**0.915±0.012**	**0.914±0.013**
DRE	0.495±0.000	0.495±0.000	0.495±0.000	0.495±0.000	0.495±0.000
SDMS	0.613±0.018	0.495±0.005	0.500±0.000	0.500±0.000	0.500±0.000
SE	0.620±0.031	0.505±0.022	0.500±0.000	0.500±0.000	0.500±0.000
TBE	0.582±0.019	0.500±0.000	0.500±0.000	0.500±0.000	0.500±0.000
entropy	0.836±0.000	0.500±0.000	**0.916±0.006**	0.874±0.001	0.818±0.016

Considering the overall results, KC is able to detect changes stably, regardless of the difference in the value of σ. For small values of σ, the methods based on parametric models (the SDMS, SE, and TBE algorithms) were sometimes able to detect the points of change, probably because the dataset is relatively closely approximates a GMM.

Gaussian Mixture Model

We generated an artificial dataset distributed using a GMM, of which the parameters are defined as follows:

$$\begin{cases} K^* = 2,\ \mu = (\mu_1, \mu_2) & \text{if } 1 \leq t \leq \tau_1, \\ K^* = 3,\ \mu = (\mu_1, \mu_2, u(t)) & \text{if } \tau_1 + 1 \leq t \leq \tau_2, \\ K^* = 3,\ \mu = (\mu_1, \mu_2, \mu_3) & \text{if } \tau_2 + 1 \leq t \leq T, \end{cases} \tag{5.19}$$
$$\text{where } u(t) = \frac{(\tau_2 - t)\mu_2 + (t - \tau_1)\mu_3}{\tau_2 - \tau_1}.$$

A chunk of data starts undergoing transformation at time $t = 50$ and finishes forming a new chunk of data at time $t = 100$. An example of the generated dataset is shown in Figure 5.13.

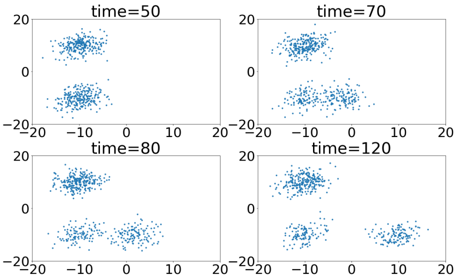


Figure 5.13: Dataset aggregated into data chunks using the Gaussian mixture model.

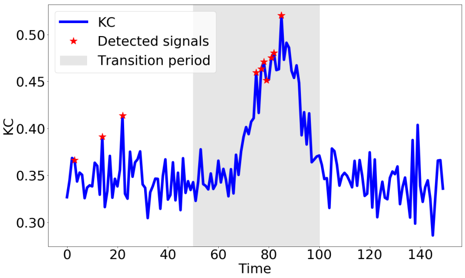


Figure 5.14: Value of KC over time for the dataset aggregated using the Gaussian mixture model.

We used the index in Algorithm 3 to detect the change points. The detection parameter is $\eta = 3$ in Figure 5.14. During the transition period, KC gradually increases at first, then gradually decreases after reaching its peak, before stabilizing in a certain range. Points of change are indicated by a gradual increase in the value of KC.

We also generated datasets by changing the variance of the Gaussian model from $\sigma^2 = 2.0$ to $\sigma^2 = 10.0$. The resultant AUC scores (see Equation (5.17)) are listed in Table 5.9. KC was able to indicate points of change earlier than the other methods assuming a GMM (SDMS, SE, and TBE algorithm) because it captures changes in the density distribution. In models that assume a GMM (SDMS, SE, and TBE algorithm), the number of clusters changes to a certain degree at a time when a chunk of the dataset undergoes transformation. In contrast, KC can be considered to indicate change in the complexity from the point at which a chunk of the dataset starts to gradually transform. Because of these characteristics, the nonparametric KC detected change at the earliest time.

Next, we observed the behavior of the benefit and FAR scores when calculating the AUC for different values of the sensitivity parameter. The results are shown in Figure 5.15, which indicates that KC yields high benefit values, but the corresponding FAR values are also relatively high. This is attributed to KC being more sensitive to change than the other methods. The results in the figure show that each algorithm becomes less sensitive to change as the change sensitivity parameter becomes larger (the change is judged more severely). For SDMS and TBE, which contain no sensitivity parameters, the scores of benefit and FAR were constant. The SE algorithm would be expected to enable change to be detected outside the transition period by reducing the change sensitivity parameter. The values of benefit was considered to be low because the subsequent time is not detected as a change.

Observing the curve plotting benefit against FAR as shown in Figure 5.16, the AUC score of KC was higher because the advantages of benefit exceed the disadvantages of FAR.

Table 5.9: AUC scores for the algorithms (GMM dataset).

Method	σ^2 =2.0	σ^2 =4.0	σ^2 =6.0	σ^2 =8.0	σ^2 =10.0
KC	**0.956±0.031**	**0.971±0.015**	**0.951±0.018**	**0.924±0.047**	**0.932±0.038**
DRE	0.664±0.130	0.671±0.133	0.674±0.134	0.679±0.137	0.684±0.141
SDMS	0.775±0.102	0.686±0.118	0.731±0.184	0.656±0.120	0.567±0.121
SE	0.817±0.094	0.844±0.119	0.872±0.050	0.890±0.064	0.857±0.107
TBE	0.721±0.115	0.604±0.139	0.682±0.138	0.562±0.054	0.495±0.025
entropy	0.584±0.178	0.922±0.031	0.921±0.034	0.913±0.030	0.914±0.037

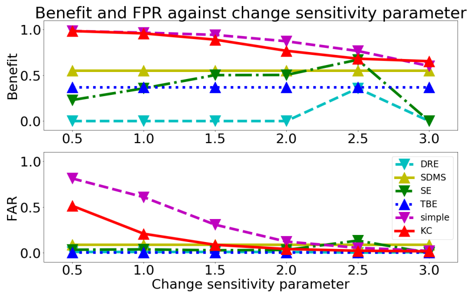


Figure 5.15: Evaluation of KC vs. the other algorithms by plotting benefit and FAR against the change sensitivity parameter.

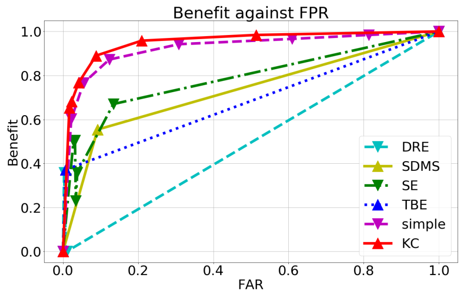


Figure 5.16: Evaluation of KC vs. the other algorithms by plotting the benefit against FAR.

5.3.2 Calculation Time

We evaluated the calculation time of KC and the entropy used above with a number of different dimensions. We used the dataset with the same settings as in Section **??**. To complete the calculation of the entropy in realistic time, the mesh in the calculation was set to rough. The results are shown in Figure 5.17, which shows that the time required to compute KC is independent of the dimension. However, the time required to calculate the entropy increases exponentially as the dimension increases.

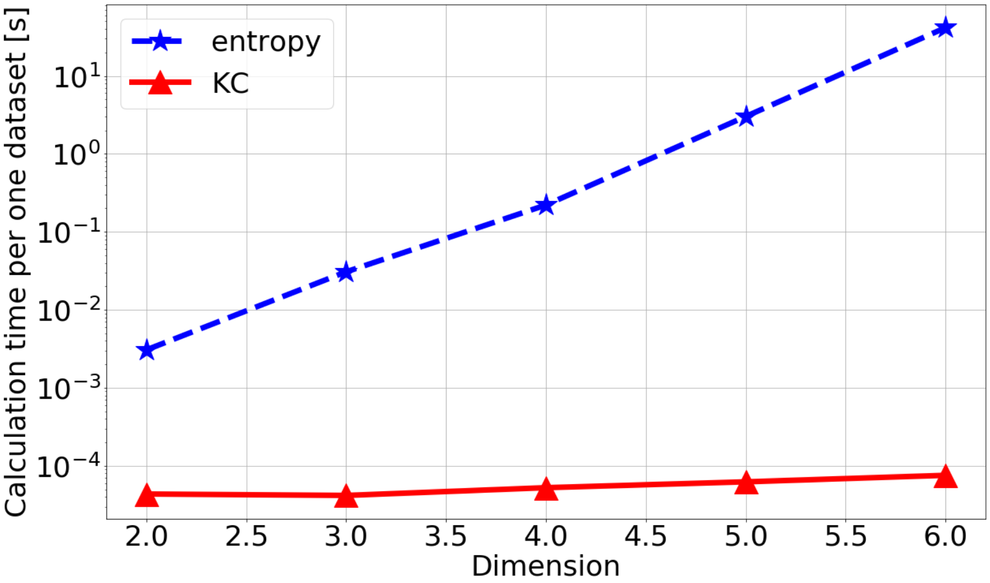


Figure 5.17: Time required by the two methods to perform the calculation for different dimensions.

5.3.3 Real Datasets

Sensor Dataset (Household Electricity Consumption)

We tested our method by using a dataset comprising household electricity consumption measurements provided by Dua and Graff [4]. This dataset contains three categories of electricity consumption data: 1. by the kitchen and laundry room, 2. by an electric water heater, 3. by the air conditioner. The data were acquired every other minute from December 17, 2006 to December 10, 2010. Using this dataset, we defined the features

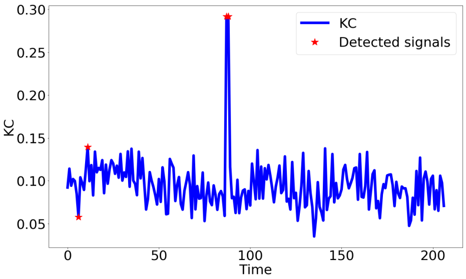


Figure 5.18: KC at each time (household).

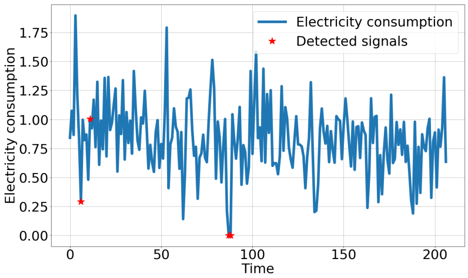


Figure 5.19: Average consumption of the first feature (in the kitchen and laundry room) for each week.

$X_t = (\mathbf{x}_1, \cdots, \mathbf{x}_n)$ and $\mathbf{x}_i = (x_{i1}, x_{i2}, x_{i3})$. Each $\mathbf{x}_i$ is the value of the total consumption in an hour for each of the three categories, and each X_t represents the consumption for one week (the number of datasets at each time was $n = 168$ or less).

The results obtained with the proposed method enabled us to observe changes in the data sequence at a specific time. The value of KC is shown in Figure 5.18 as a function of time. We detected large changes at $t = 87, 88$ (August 17–23, 2008 and August 24–30, 2008). As seen in Figure 5.19, the use of electricity in the kitchen and laundry room on a normal day is always greater than zero; however, the consumption is zero for the week in which the change is detected. This is because we were able to observe some lifestyle anomalies (or changes) by calculating the value of KC.

Marketing Dataset (Beer Purchasing Behavior)

We tested our method using a dataset containing data of beer purchases that was provided by Hakuhodo, Inc. and M-CUBE, Inc. QPR. The dataset is a record of customers' beer purchasing behavior and includes brands from six manufacturers (denoted A–F here). We captured the changes in the requirements of high-value customers by using simulation and analyzed the data using the top 50% of customers in terms of purchasing volume. The dataset at each instant represents the amount purchased for the four categories of beer in the last two weeks, where the target period was from November 1 to January 31 of the following year.

The results obtained with the proposed algorithm to calculate the value of KC to detect the points of change are shown in Figure 5.20. The proposed algorithm detected points of significantly large change on December 1, 3, and 31 and on January 1, 13, 14, and 15 (time=$17, 19, 47, 48, 60, 61, 62$). The detection of change in customers ' purchasing behavior in the last two weeks of November and December reflects the year-end demand, and the complexity of the structure decreases because the beer consumption stabilizes

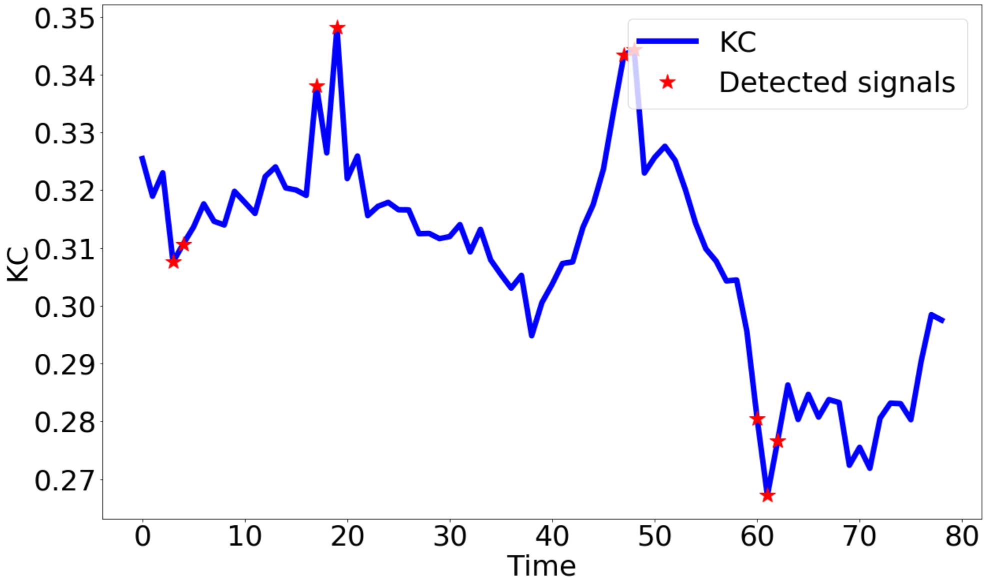


Figure 5.20: KC at each time (beer).

after the end of the year. These results confirm the ability of the proposed algorithm to effectively capture real market changes.

5.4 Conclusion

We proposed a method to calculate the value of KC to define new structural information for data with a nonparametric distribution and proposed a method to detect its change over time. This index is defined by measuring the density of data in terms of information bias and is based on the Gini index. We use the NML code length based on the MDL principle as a criterion to express information. We showed that this index, KC, is a value that characterizes the number of data chunks. Furthermore, we proposed an algorithm to detect the change in KC when the data are given as time series. By using this algorithm, we provide a framework for the detection of changes based on KC. However, KC has some limitations. First, since the parameters that define KC are not determined with theory, it is a future research issue to improve the reliability of KC by establishing parameter determination theory. Second, the assumed data structure is a structure with data chunks, and the usefulness of KC has only been shown in experiments focusing on the number of data chunks. For this reason, we consider that application to other data structures or proposal of new methods is a future research issue. The usefulness of the proposed method was experimentally demonstrated using both artificial and real datasets. For the artificial datasets, the proposed algorithm could detect the change points in terms of the benefit, delay, FAR, and AUC scores for specific kinds of datasets. Further, we showed the effectiveness of our method for analyzing a dataset containing household electricity consumption data. Specifically, our algorithm automatically detected changes at times when the electricity consumption in the kitchen and laundry room was likely to change. In addition, we also analyzed a dataset containing data relating to customers' beer purchasing behavior over time. The purchasing behavior significantly changed over the last part of the year and after the year ended, and the proposed algorithm effectively captured these changes. The ability of the proposed method to capture the complexity in a data structure that can be defined by the density has expanded the possibility of searching for new hidden value. In future, we aim to extend our work to other kernel functions, and to calculate exact values rather than the upper bound in the form of the NML. With respect to the KC index itself, analysis of the theoretical nature of this index (e.g., its properties by using the Gini coefficient) remains as an extension of current research. This may enable us to obtain appropriate values for the parameters (the features of the KC index depend on the parameters to some extent). In terms of evaluation, we can define data types for which KC is effective by increasing the variation of the experiments. In terms of applications, we consider adding more qualitative interpretations of actual data in combination with other methods. In addition, considering the application of our method, real data are rarely neatly arranged; thus, extending KC to an index that can handle missing data is a very important issue.

Chapter 6

Conclusion

In this thesis, we proposed new indices for measuring the structural information of a dataset. Using these indices, we proposed new methods for detecting change points and their early warning signals.

First, we proposed novel indices of structural information for parametric and nonparametric structures. For parametric models, we considered the problem of determining the number of clusters; we proposed an index called SE as the uncertainty of model selection. As a result, we were able to continuously grasp the uncertainty of model selection, which cannot be understood by simply determining the number of clusters. When using SE, to raise stable alarms of early warning signals, we proposed a method for selecting a suitable parameter for SE. For nonparametric models, we proposed an index, i.e., KC, to ascertain the structural information of aggregated data. It is defined by measuring the density of a dataset in terms of information bias with the Gini index; a larger KC indicates that the distribution of the dataset is wider and that the structure is more complex. Even though we adopted kernel density estimation as a nonparametric density estimation method, its NML code length cannot be directly calculated. For this, we proposed a method for calculating NML code length associated with kernel density.

Next, we proposed three methods for detecting change points and their early warning signals. The first method was an algorithm using SE in a parametric model. We proposed a method for detecting early warning signals in terms of the uncertainty of model selection using SE. The second method was an algorithm using SMCS in a parametric model. We proposed a unifying framework for detecting model changes and their early warning signals simultaneously. SMCS is a real-valued index that measures the degree of a model change. It is defined as the difference between the code lengths associated with the unchanged and changed models. Last, we explained the algorithm using KC in a nonparametric model. We proposed an algorithm to detect changes in KC when data are given as a time series. Our proposed algorithm only detects changes in global information measured by KC and provides a new view of nonparametric change detection. In addition, KC does not only detect abrupt changes but also gradual ones. Because KC is a continuous value, it is effective in detecting such gradual changes.

Last, we empirically evaluated the effectiveness of our methods. We employed synthetic datasets to empirically demonstrate that reliable alarms of model changes and their early warning signals using SE, SMCS, and KC could be achieved. Specifically, early warning signals can be detected significantly earlier than the alarms provided by the existing methods. We also employed two real datasets to validate SE, SMCS, and KC: a marketing dataset and household electric consumption dataset. For both datasets, we could detect meaningful change points corresponding to clear behavior changes. Regarding early warning signals, although these signals were not explicitly captured, it is possible to perceive them by changing the SE, SMCS, and KC values.

In this study, we focused only on the change in the number of clusters; however, we believe it will be useful to develop a change detection algorithm that uses not only the number of clusters but also the properties in the clusters. Moreover, for the detection of early warning signals, we believe that it is also valuable to derive an index that shows why early warning signals were detected. Last, we believe that it is also valuable to detect changes and give meaning by developing a method that mixes multiple indices. These will be a part of future work.

Acknowledgements

I want to express my sincere appreciation to my supervising professor, Kenji Yamanishi. He was willing to me to go on to the doctoral course while working in a company, and I was able to research at the doctoral course. Thanks to him, I was able to meet exciting topics: change detection, detection of its early warning signals, and MDL based model selection. It was very fortunate for me that I was able to lead the research life from the bachelor, master, and doctoral courses under him. Also, his in-depth and extensive knowledge and wisdom gave me much inspiration, and I was able to learn many important things through research and discussion. I am confident that these will have a significant impact on my professional life.

I am very grateful to the doctoral thesis review committee members, including Kenji Yamanishi, Fumiyasu Komaki, Hiroshi Kori, Taiji Suzuki, and Tomonari Sei, for taking the time to review my thesis. We received valuable opinions from lab seminars and other discussion venues and allowed me to consider the direction of research. I am very grateful to be able to research in the 6th Laboratory of Mathematical Informatics with excellent researchers. In conducting the research, we were able to obtain exciting results based on actual data by acquiring permission from Ryo Domoto in Hakuhodo Inc. to use beer data. We would also like to thank M-CUBE Inc. for allowing us to use the data of QPR. in the thesis.

I want to express my sincere appreciation to Kunihiko Shimizu and Yoshinori Okuda, my former bosses, who accepted my admission to the doctoral course while working. They had a significant influence on my professional life, including support at work. I am very grateful to the team members of the company who sent me out. The current supervisors, Atsushi Koga and Tomoaki Nishimura, are also very grateful for the approval of the continuation of the doctoral course.

Finally, I would like to thank my wife, Sayaka Hirai, and my daughter, Rika Hirai. The admission to the doctoral course was almost the same time as my daughter Rika was born, but Sayaka was kind enough to admit it. I believe that it was with the support of my family that I was able to overcome some hectic times. I will give them back over my life.

Bibliography

[1] Hirotogu Akaike. Information theory and an extension of the maximum likelihood principle. In *Selected papers of hirotugu akaike*, pages 199–213. Springer, 1998.

[2] The SciPy community. scipy.stats.gaussian_kde. `https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.gaussian_kde.html`.

[3] Richard A Davis, Thomas C M Lee, and Gabriel A Rodriguez-Yam. Structural break estimation for nonstationary time series models. *Journal of the American Statistical Association*, 101(473):223–239, 2006.

[4] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[5] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, 46(4):44, 2014.

[6] Lars Peter Hansen. Large sample properties of generalized method of moments estimators. *Econometrica: Journal of the Econometric Society*, pages 1029–1054, 1982.

[7] Zaïd Harchaoui, Eric Moulines, and Francis R. Bach. Kernel change-point analysis. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 609–616. Curran Associates, Inc., 2009.

[8] Mark Herbster and Manfred K Warmuth. Tracking the best expert. *Machine learning*, 32(2):151–178, 1998.

[9] So Hirai and Kenji Yamanishi. 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*, pages 343–351. ACM, 2012.

[10] So Hirai and Kenji Yamanishi. Efficient computation of normalized maximum likelihood codes for Gaussian mixture models with its applications to clustering. *IEEE Transactions on Information Theory*, 59(11):7718–7727, 2013.

[11] So Hirai and Kenji Yamanishi. Detecting Latent Structure Uncertainty with Structural Entropy. In *Proceedings of 2018 IEEE International Conference on Big Data (Big Data)*, pages 26–35. IEEE, 2018.

[12] So Hirai and Kenji Yamanishi. Correction to Efficient Computation of Normalized Maximum Likelihood Codes for Gaussian Mixture Models With Its Applications to Clustering. *IEEE Transactions on Information Theory*, 65(10):6827–6828, 2019.

[13] So Hirai and Kenji Yamanishi. Detecting Model Changes and their Early Warning Signals Using MDL Change Statistics. In *Proceedings of 2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019.

[14] So Hirai and Kenji Yamanishi. Kernel Complexity for Nonparametric Distributions and Detection of Its Changes. preprint, 2019.

[15] David Tse Jung Huang, Yun Sing Koh, Gillian Dobbie, and Russel Pears. Detecting volatility shift in data streams. In *Data Mining (ICDM), 2014 IEEE International Conference on*, pages 863–868. IEEE, 2014.

[16] Yu Ito, Shin-ichi Oeda, and Kenji Yamanishi. Rank selection for non-negative matrix factorization with normalized maximum likelihood coding. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, pages 720–728. SIAM, 2016.

[17] Daniel R Jeske, Veronica Montes De Oca, Wolfgang Bischoff, and Mazda Marvasti. Cusum techniques for timeslot sequences with applications to network surveillance. *Computational Statistics & Data Analysis*, 53(12):4332–4344, 2009.

[18] Nanak Kakwani et al. *Applications of Lorenz curves in economic analysis*. Number 12. International Bank for Reconstruction and Development, 1975.

[19] Yoshinobu Kawahara and Masashi Sugiyama. Sequential change-point detection based on direct density-ratio estimation. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 5(2):114–127, 2012.

[20] Andrei N Kolmogorov. On tables of random numbers. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 369–376, 1963.

[21] Petri Kontkanen and Petri Myllymäki. A linear-time algorithm for computing the multinomial stochastic complexity. *Information Processing Letters*, 103(6):227–233, 2007.

[22] Ivan O Kyrgyzov, Olexiy O Kyrgyzov, Henri Maître, and Marine Campedel. Kernel mdl to determine the number of clusters. In *Proceedings of International Workshop on Machine Learning and Data Mining in Pattern Recognition*, pages 203–217. Springer, 2007.

[23] Lev Davidovich Landau and Evgenii Mikhailovich Lifshitz. *Course of theoretical physics*. Elsevier, 2013.

[24] Song Liu, Makoto Yamada, Nigel Collier, and Masashi Sugiyama. Change-point detection in time-series data by relative density-ratio estimation. *Neural Networks*, 43:72–83, 2013.

[25] Geoffrey J McLachlan and Kaye E Basford. *Mixture models: Inference and applications to clustering*, volume 84. M. Dekker New York, 1988.

[26] Fionn Murtagh. Multidimensional clustering algorithms. *Compstat Lectures, Vienna: Physika Verlag, 1985*, 1985.

[27] Yukio Ohsawa. Graph-based entropy for detecting explanatory signs of changes in market. *The Review of Socionetwork Strategies*, pages 1–21, 2018.

[28] Ilya Prigogine. Time, structure, and fluctuations. *Science*, 201(4358):777–785, 1978.

[29] Jorma Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978.

[30] Jorma Rissanen. *Information and complexity in statistical modeling*. Springer Science & Business Media, 2007.

[31] Jorma Rissanen. *Optimal estimation of parameters*. Cambridge University Press, 2012.

[32] Jorma Rissanen, Teemu Roos, and Petri Myllymäki. Model selection by sequentially normalized least squares. *Journal of Multivariate Analysis*, 101(4):839–849, 2010.

[33] Jorma J Rissanen. Fisher information and stochastic complexity. *IEEE transactions on information theory*, 42(1):40–47, 1996.

[34] Yunus Saatçi, Ryan D Turner, and Carl Edward Rasmussen. Gaussian process change point models. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 927–934. Citeseer, 2010.

[35] Gideon Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

[36] David W Scott. *Multivariate density estimation: theory, practice, and visualization.* John Wiley & Sons, 2015.

[37] Yurii Mikhailovich Shtar'kov. Universal sequential coding of single messages. *Translated from Problems of Information Transmission*, 23(3):3–17, July-September 1987.

[38] Mingzhou Song and Hongbin Wang. Highly efficient incremental estimation of gaussian mixture models for online data stream clustering. In *Intelligent Computing: Theory and Applications III*, volume 5803, pages 174–183. International Society for Optics and Photonics, 2005.

[39] Toshimitsu Takahashi, Ryota Tomioka, and Kenji Yamanishi. Discovering emerging topics in social streams via link-anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 26(1):120–130, 2014.

[40] Bien Aik Tan, Peter Gerstoft, Caglar Yardim, and William S Hodgkiss. Change-point detection for recursive bayesian geoacoustic inversions. *The Journal of the Acoustical Society of America*, 137(4):1962–1970, 2015.

[41] Tim Van Erven, Peter Grünwald, and Steven De Rooij. Catching up faster by switching sooner: A predictive approach to adaptive estimation with an application to the aic–bic dilemma. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(3):361–417, 2012.

[42] Xiang Xuan and Kevin Murphy. Modeling changing dependency structure in multivariate time series. In *Proceedings of the 24th international conference on Machine learning*, pages 1055–1062. ACM, 2007.

[43] Kenji Yamanishi and Shintaro Fukushima. Model change detection with the mdl principle. *IEEE Transactions on Information Theory*, 64(9):6115–6126, 2018.

[44] Kenji Yamanishi and Yuko Maruyama. Dynamic syslog mining for network failure monitoring. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 499–508. ACM, 2005.

[45] Kenji Yamanishi and Yuko Maruyama. Dynamic model selection with its applications to novelty detection. *IEEE Transactions on Information Theory*, 53(6):2180–2189, 2007.

[46] Kenji Yamanishi and Kohei Miyaguchi. Detecting gradual changes from data stream using mdl-change statistics. In *2016 IEEE International Conference on Big Data (Big Data)*, pages 156–163. IEEE, 2016.

[47] Kenji Yamanishi, Tianyi Wu, Shinya Sugawara, and Makoto Okada. The decomposed normalized maximum likelihood code-length criterion for selecting hierarchical latent variable models. *Data Mining and Knowledge Discovery*, 33(4):1017–1058, 2019.

[48] Wei Zhang and Jana Ksecká. *Nonparametric estimation of multiple structures with outliers*, pages 60–74. Springer, 2006.

Appendices

A.1 Proof of Theorem 6

In this section, we show the detail proof of the theorem 6. This proof is calculated with reference to paper [43]. Here, X_t, Z_t is written as D_t for simplicity. We can calculate the type-I error probability as follows:

$$
\begin{aligned}
\text{Type-I error prob.} &= Prob_{H_0}\left[\Phi_t < 0\right] \\
&= \int_{D_{t-1,t} \in \{D : \Phi_t \geq 0\}} p(D_{t-1}; \theta_0^*, \mathcal{M}_0^*) p(D_t; \theta_0^*, \mathcal{M}_0^*) \, \mathrm{d}D_{t-1,t}, \quad \text{(A.1)}
\end{aligned}
$$

where θ_0^* is the true parameter of model $\mathcal{M}_0^*$. Here, using the condition of $\Phi_t \geq 0$, we get the following inequality:

$$
\begin{aligned}
& -\log p(D_{t-1,t}; \theta_0^*, \mathcal{M}_0^*) + \log \mathcal{C}_{2n}(\mathcal{M}_0^*) + \mathcal{L}(\mathcal{M}_0^*, \mathcal{M}_0^*) \\
\geq\ & -\log p(D_{t-1,t}; \hat{\theta}, \mathcal{M}_0^*) + \log \mathcal{C}_{2n}(\mathcal{M}_0^*) + \mathcal{L}(\mathcal{M}_0^*, \mathcal{M}_0^*) \\
\geq\ & \min_{\mathcal{M}} \{L_{\text{NML}}(D_{t-1,t}; \mathcal{M}) + \mathcal{L}(\mathcal{M}, \mathcal{M})\} \\
\geq\ & \min_{\mathcal{M}', \mathcal{M}''} \big\{L_{\text{NML}}(D_{t-1}; \mathcal{M}') + L_{\text{NML}}(D_t; \mathcal{M}'') + \mathcal{L}(\mathcal{M}'', \mathcal{M}')\big\} + n\epsilon \\
=:\ & \tilde{\mathcal{L}}(D_{t-1,t}) + n\epsilon \qquad \text{(A.2)}
\end{aligned}
$$

Using this inequality, we can derive an upper bound on the type-I error probability (A.1) as follows:

$$
\begin{aligned}
& \text{Type-I error prob.} \\
=\ & Prob_{H_0}\left[\Phi_t \geq 0\right] \\
=\ & \int_{D_{t-1,t} \in \{D : \Phi_t \geq 0\}} p(D_{t-1,t}; \theta_0^*, \mathcal{M}_0^*) \, \mathrm{d}D_{t-1,t} \\
\leq\ & \int_{D_{t-1,t} \in \{D : \Phi_t \geq 0\}} \exp\big\{ -\tilde{\mathcal{L}}(D_{t-1,t}) + \log \mathcal{C}_{2n}(\mathcal{M}_0^*) + \mathcal{L}(\mathcal{M}_0^*, \mathcal{M}_0^*) - n\epsilon \big\} \, \mathrm{d}D_{t-1,t} \quad \text{(A.3)} \\
\leq\ & \exp\left\{ -n\left(\epsilon - \frac{\log \mathcal{C}_{2n}(\mathcal{M}_0^*) + \mathcal{L}(\mathcal{M}_0^*, \mathcal{M}_0^*)}{n} \right) \right\} \qquad \text{(A.4)}
\end{aligned}
$$

Here, the calculation from Equation (A.3) to Equation (A.4) can be derived by Kraft's inequality as follows:

$$\int \exp\big\{ -\tilde{\mathcal{L}}(D)\big\}\, \mathrm{d}D \leq 1.$$

This is the end of the proof.

A.2 Proof of Theorem 7

In this section, we show the detail proof of the theorem 7. This proof is calculated with reference to paper [43]. Here, X_t, Z_t is written as D_t for simplicity. We can calculate the type-II error probability as follows:

$$\begin{aligned} \text{Type-II error prob.} &= Prob_{H_1}\left[\Phi_t \leq 0\right] \\ &= \int_{D_{t-1,t}\in\{D:\Phi_t<0\}} p(D_{t-1};\theta_1^*,\mathcal{M}_1^*)p(D_t;\theta_2^*,\mathcal{M}_2^*)\,\mathrm{d}D_{t-1,t}, \quad \text{(A.5)} \end{aligned}$$

where θ_1^*, θ_2^* are the true parameters of model $\mathcal{M}_1^*, \mathcal{M}_2^*$, respectively. Here, using the condition of $\Phi_t < 0$, we get the following inequality:

$$\begin{aligned} & -\log \tilde{p}(D_{t-1,t}) - \log \tilde{\mathcal{C}}_{2n} \\ =\; & \min_{\mathcal{M}} \{L_{\text{NML}}(D_{t-1,t};\mathcal{M}) + \mathcal{L}(\mathcal{M},\mathcal{M})\} \\ <\; & \min_{\mathcal{M}',\mathcal{M}''} \big\{L_{\text{NML}}(D_{t-1};\mathcal{M}') + L_{\text{NML}}(D_t;\mathcal{M}'') + \mathcal{L}(\mathcal{M}'',\mathcal{M}')\big\} + n\epsilon \\ \leq\; & L_{\text{NML}}(D_{t-1};\mathcal{M}_1^*) + L_{\text{NML}}(D_t;\mathcal{M}_2^*) + \mathcal{L}(\mathcal{M}_2^*,\mathcal{M}_1^*) + n\epsilon \\ \leq\; & -\log\big(p(D_{t-1};\theta_1^*,\mathcal{M}_1^*)p(D_t;\theta_2^*,\mathcal{M}_2^*)\big) \\ & \quad + \log \mathcal{C}_n(\mathcal{M}_1^*) + \log \mathcal{C}_n(\mathcal{M}_2^*) + \mathcal{L}(\mathcal{M}_2^*,\mathcal{M}_1^*) + n\epsilon, \qquad \text{(A.6)} \end{aligned}$$

where we define $\tilde{p}$ and $\tilde{\mathcal{C}}_{2n}$ as Equation (4.8) and Equation (4.9), respectively. Here, by Equation (A.6), we can derive the following inequality:

$$1 < \left(\frac{\tilde{p}(D_{t-1,t})}{p(D_{t-1};\theta_1^*,\mathcal{M}_1^*)p(D_t;\theta_2^*,\mathcal{M}_2^*)}\right)^{\alpha} \cdot \exp\left(\alpha \ell_n(\mathcal{M}_1^*,\mathcal{M}_2^*,\epsilon)\right) \qquad \text{(A.7)}$$

where α satisfies $0 < \alpha < 1$ and $\ell_n(\mathcal{M}_1^*,\mathcal{M}_2^*,\epsilon)$ is defined by Equation (4.7). Using Equation (A.7) with Equation (A.5), we have the following upper bound on type-II error probability:

$$\begin{aligned} & \text{Type-II error prob.} \\ =\; & \int_{D_{t-1,t}\in\{D:\Phi_t<0\}} p(D_{t-1};\theta_1^*,\mathcal{M}_1^*)p(D_t;\theta_2^*,\mathcal{M}_2^*)\,\mathrm{d}D_{t-1,t} \end{aligned}$$

$$
\begin{aligned}
< \; & \int_{D_{t-1,t} \in \{D : \Phi_t < 0\}} p(D_{t-1}; \theta_1^*, \mathcal{M}_1^*) p(D_t; \theta_2^*, \mathcal{M}_2^*) \, \mathrm{d}D_{t-1,t} \\
& \quad \times \left(\frac{\tilde{p}(D_{t-1,t})}{p(D_{t-1}; \theta_1^*, \mathcal{M}_1^*) p(D_t; \theta_2^*, \mathcal{M}_2^*)} \right)^{\alpha} \cdot \exp\left(\alpha \ell_n(\mathcal{M}_1^*, \mathcal{M}_2^*, \epsilon) \right) \\
< \; & \int_{D_{t-1,t} \in \{D : \Phi_t < 0\}} \left(p(D_{t-1}; \theta_1^*, \mathcal{M}_1^*) p(D_t; \theta_2^*, \mathcal{M}_2^*) \right)^{1-\alpha} \cdot \tilde{p}(D_{t-1,t})^{\alpha} \, \mathrm{d}D_{t-1,t} \\
& \quad \times \exp\left(\alpha \ell_n(\mathcal{M}_1^*, \mathcal{M}_2^*, \epsilon) \right)
\end{aligned}
$$

Then we derive an upper bound on type-II error probability as follows:

$$
\begin{aligned}
& \text{Type-II error prob.} \\
< \; & \exp(n \log \delta_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*)) \times \exp\left(\alpha \ell_n(\mathcal{M}_1^*, \mathcal{M}_2^*, \epsilon) \right) \\
= \; & \exp(n \log(1 - 2\alpha(1-\alpha) d_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*))) \times \exp\left(\alpha \ell_n(\mathcal{M}_1^*, \mathcal{M}_2^*, \epsilon) \right) \\
< \; & \exp(-2n\alpha(1-\alpha) d_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*)) \times \exp\left(\alpha \ell_n(\mathcal{M}_1^*, \mathcal{M}_2^*, \epsilon) \right) \\
= \; & \exp\left\{ -n \left(2\alpha(1-\alpha) d_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*) - \frac{\alpha \ell_n(\mathcal{M}_1^*, \mathcal{M}_2^*, \epsilon)}{n} \right) \right\},
\end{aligned}
$$

where we define $d_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*)$ and $\delta_n^{\alpha}(\mathcal{M}_1^*, \mathcal{M}_2^*)$ as Equation (4.5) and Equation (4.6), respectively. This is the end of the proof.